
Ana Rita Tiradentes Terra

PROGRAMAÇÃO DA PRODUÇÃO:
UMA ABORDAGEM POR REDES NEURAIS
ARTIFICIAIS

São Carlos
2000

Ana Rita Tiradentes Terra

PROGRAMAÇÃO DA PRODUÇÃO:
UMA ABORDAGEM POR REDES NEURAIIS
ARTIFICIAIS

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia de Produção da Universidade Federal de São Carlos, como parte dos requisitos para obtenção do título de Mestre em Engenharia de Produção.

Orientador: Prof. Dr. Néocles Alves Pereira

São Carlos
2000

**Ficha catalográfica elaborada pelo DePT da
Biblioteca Comunitária da UFSCar**

T323pp Terra, Ana Rita Tiradentes.
Programação da produção: uma abordagem por redes
neurais artificiais / Ana Rita Tiradentes Terra. –São Carlos :
UFSCar, 2000.

149 p.

Dissertação (Mestrado) – Universidade Federal de São
Carlos, 2000.

1.Programação da produção. 2. Redes neurais. 3.
Inteligência artificial. 4. Planejamento e controle da produção.
5. Simulação. 6. Programação heurística. I.Título.

Dedicado aos meus pais,

EVALDA e FÁBIO

Agradecimentos

Ao Daniel, pelo constante incentivo, compreensão, paciência e amor.

Ao Prof. Dr. Néocles Alves Pereira, pela orientação dispensada durante este trabalho e pela amizade ao longo desses anos.

À Sheila, pela orientação em redes neurais artificiais.

Ao Fabiano, pelo apoio na instalação dos softwares.

À Lunalva, pela hospitalidade em São Carlos.

Ao Prof. Miguel Bueno da Costa, pela utilização do laboratório Simucad.

Ao Departamento e ao Programa de Pós-Graduação em Engenharia de Produção da UFSCar, que possibilitaram a realização deste mestrado.

À CAPES, Coordenação de Aperfeiçoamento de Pessoal de Nível Superior, pela concessão de bolsa de estudo.

Aos amigos e a todos que colaboraram direta e indiretamente para a realização deste trabalho.

Aos meus pais, Evalda e Fábio, pela força, amor e carinho.

Resumo

A atividade da programação da produção, também chamada *scheduling*, encontra-se no nível mais detalhado e complexo de um sistema de planejamento e controle da produção. Devido à sua natureza combinatorial, vários métodos têm sido propostos como alternativas de solução para resolvê-la. Entre eles encontram-se a técnica de simulação de sistemas e abordagens por inteligência artificial. Este trabalho apresenta um procedimento de solução da programação da produção, através de um modelo híbrido de simulação de sistemas e redes neurais artificiais. Neste procedimento, o papel da rede neural artificial é aprender as relações entre as regras de prioridade designadas às máquinas de um ambiente de produção, e os valores das medidas de desempenho utilizados para avaliar as alternativas de programação. O objetivo é analisar a diferenciação entre um conjunto de combinações de regras de prioridade através da avaliação de quatro medidas de desempenho. Resultados são apresentados e comentados, destacando a capacidade de generalização do modelo híbrido em prescrever regras de prioridade às máquinas, a partir de valores de medidas de desempenho estabelecidos pelo usuário.

Palavras-Chave: Programação da produção. Redes neurais. Inteligência artificial. Planejamento e controle da produção. Simulação. Programação heurística.

Abstract

Production scheduling finds in the most detailed and complex level of production planning and control systems. Due to its nature combinatorial, several methods have been proposed to solve it. Among them there are the techniques of simulation systems and artificial intelligence approaches. This work presents a procedure of solution of production scheduling, through a hybrid model of simulation systems and artificial neural networks. In this procedure, the purpose of the artificial neural network is to learn the relationships between the priority rules designated to the machines of a production system, and the values of performance measures used to evaluate the scheduling. The objective is to analyze the differentiation among a group of combinations of priority rules through the evaluation of four performance measures. Results are presented and commented, highlighting the capacity of generalization of the hybrid model in prescribing priority rules to the machines, based on values of performance measures established by the user.

Keywords: Production scheduling. Neural networks. Artificial intelligence. Production planning and control. Simulation. Heuristic scheduling.

Lista de Figuras

| | |
|--|----|
| FIGURA 2.1 – Sistemas de produção intermitente..... | 5 |
| FIGURA 2.2 – Fluxo típico na manufatura celular..... | 8 |
| FIGURA 2.3 – Classificação dos problemas de programação da produção..... | 16 |
| FIGURA 3.1 – Modelo de neurônio humano..... | 22 |
| FIGURA 3.2 – Modelo de neurônio artificial..... | 23 |
| FIGURA 3.3 – Funções de ativação..... | 24 |
| FIGURA 3.4 – Organização em camadas..... | 26 |
| FIGURA 3.5 – Rede <i>back propagation</i> com uma camada interna..... | 30 |
| FIGURA 3.6 – Algoritmo de treinamento <i>back propagation</i> | 31 |
| FIGURA 3.7 – Ativação..... | 33 |
| FIGURA 5.1 – Procedimento de solução do sistema híbrido..... | 69 |
| FIGURA 5.2 – Representação simplificada do sistema..... | 72 |
| FIGURA 5.3 – Representação simplificada da rede MLP utilizada..... | 77 |
| FIGURA 6.1 – Estrutura da rede neural utilizada – tela do <i>software</i> SNNS..... | 85 |
| FIGURA 6.2 – Exemplo de treinamento com início de <i>overtraining</i> | 89 |

Lista de Tabelas

| | |
|---|-----|
| TABELA 2.1 – Natureza do processo de chegada e do tempo de processamento..... | 12 |
| TABELA 3.1 – Características de alguns <i>softwares</i> de RNA..... | 34 |
| TABELA 4.1 – Aspectos sobre programação da produção..... | 50 |
| TABELA 4.2 – Aspectos sobre redes neurais artificiais..... | 51 |
| TABELA 5.1 – Combinações de regras de prioridade às máquinas..... | 71 |
| TABELA 6.1 – Resultados da combinação 12..... | 83 |
| TABELA 6.2 – Resultados dos treinamentos com 17 padrões..... | 87 |
| TABELA 6.3 – Resultados dos treinamentos com 68 padrões..... | 90 |
| TABELA 6.4 – Resultados dos treinamentos com 272 padrões..... | 91 |
| TABELA 6.5 – Regras prescritas pela rede neural..... | 92 |
| TABELA 6.6 – Medidas de desempenho prescritas pela rede..... | 95 |
| TABELA 6.7 – Erro (%)..... | 97 |
| TABELA 6.8 – Média do Erro (%)..... | 99 |
| TABELA 6.9 – Erro Relativo (%)..... | 102 |
| TABELA 6.10 – Média do Erro Relativo (%)..... | 104 |
| TABELA 7.1 – Média e desvio padrão das medidas da simulação..... | 106 |
| TABELA 7.2 – Porcentagem de padrões x Erros..... | 110 |
| TABELA 7.3 – Porcentagem de padrões x Diferença algébrica..... | 111 |
| TABELA A1 – <i>Software</i> NeuralSIM..... | 123 |
| TABELA A2 – <i>Software</i> NeuralWorks Professional II/Plus..... | 124 |
| TABELA A3 – <i>Software</i> NeuroShell 2..... | 125 |
| TABELA A4 – <i>Software</i> SNNS – Stuttgart Neural Networks Simulator... | 126 |
| TABELA C1 – Seqüência de chegada dos lotes e datas de entrega..... | 128 |
| TABELA E1 – Tempos de <i>setup</i> em M1 (torno)..... | 139 |

| | |
|---|-----|
| TABELA E2 – Tempos de <i>setup</i> em M2 (fresa)..... | 139 |
| TABELA E3 – Tempos de <i>setup</i> em M3 (furadeira)..... | 139 |
| TABELA F1 – Resultados da combinação 1..... | 140 |
| TABELA F2 – Resultados da combinação 2..... | 141 |
| TABELA F3 – Resultados da combinação 3..... | 141 |
| TABELA F4 – Resultados da combinação 4..... | 142 |
| TABELA F5 – Resultados da combinação 5..... | 142 |
| TABELA F6 – Resultados da combinação 6..... | 143 |
| TABELA F7 – Resultados da combinação 7..... | 143 |
| TABELA F8 – Resultados da combinação 8..... | 144 |
| TABELA F9 – Resultados da combinação 9..... | 144 |
| TABELA F10 – Resultados da combinação 10..... | 145 |
| TABELA F11 – Resultados da combinação 11..... | 145 |
| TABELA F12 – Resultados da combinação 12..... | 146 |
| TABELA F13 – Resultados da combinação 13..... | 146 |
| TABELA F14 – Resultados da combinação 14..... | 147 |
| TABELA F15 – Resultados da combinação 15..... | 147 |
| TABELA F16 – Resultados da combinação 16..... | 148 |
| TABELA F17 – Resultados da combinação 17..... | 148 |

Sumário

Resumo

Abstract

| | |
|--|-----------|
| Capítulo 1 – Introdução | 1 |
| Capítulo 2 – Programação da Produção | 4 |
| 2.1 – Introdução..... | 4 |
| 2.2 – Sistemas de Produção..... | 4 |
| 2.3 – Célula de Manufatura..... | 7 |
| 2.4 – Planejamento e Controle da Produção..... | 8 |
| 2.5 – Programação da Produção..... | 10 |
| 2.5.1 – Classificação dos Problemas de Programação da Produção e Principais Métodos de Solução..... | 14 |
| 2.6 – Simulação de Sistemas..... | 18 |
| Capítulo 3 – Redes Neurais Artificiais | 20 |
| 3.1 – Introdução..... | 20 |
| 3.2 – Definição de Redes Neurais Artificiais (RNA)..... | 21 |
| 3.3 – O Neurônio Humano..... | 21 |
| 3.4 – O Modelo Artificial..... | 22 |
| 3.5 – Características das Redes Neurais Artificiais..... | 24 |
| 3.6 – Estrutura das Redes Neurais Artificiais..... | 25 |
| 3.7 – Principais Arquiteturas de Redes Neurais Artificiais..... | 27 |
| 3.8 – O Algoritmo <i>Back Propagation</i> | 29 |
| 3.9 – Ativação..... | 32 |
| 3.10 – <i>Softwares</i> de Redes Neurais Artificiais..... | 33 |

| | |
|--|------------|
| Capítulo 4 – Redes Neurais e a Engenharia de Produção | 36 |
| 4.1 – Aplicações de Redes Neurais Artificiais na Manufatura..... | 36 |
| 4.2 – Programação da Produção Segundo Redes Neurais Artificiais... | 38 |
| 4.2.1 – Aplicações com o Algoritmo <i>Back Propagation</i> | 38 |
| 4.2.2 – Aplicações com a Rede Hopfield..... | 43 |
| 4.3 – Análise sobre os Modelos Apresentados..... | 47 |
| | |
| Capítulo 5 – O Modelo Proposto | 53 |
| 5.1 – Introdução..... | 53 |
| 5.2 – O Ambiente da Programação da Produção..... | 53 |
| 5.3 – O Problema Específico..... | 58 |
| 5.4 – O Procedimento Utilizado..... | 65 |
| 5.5 – A Simulação..... | 70 |
| 5.6 – A Rede Neural Artificial..... | 74 |
| | |
| Capítulo 6 – Resultados | 81 |
| 6.1 – Introdução..... | 81 |
| 6.2 – Resultados da Simulação..... | 81 |
| 6.3 – Resultados de Treinamento e Teste da Rede Neural..... | 84 |
| 6.4 – Resultados da Verificação no Simulador..... | 94 |
| | |
| Capítulo 7 - Conclusões | 106 |
| | |
| Referências Bibliográficas | 115 |
| | |
| Bibliografia | 120 |
| | |
| Apêndice A – Softwares de Redes Neurais Artificiais | 123 |
| | |
| Apêndice B – Programa de Seqüência de Chegada de Lotes | 127 |

| | |
|--|------------|
| Apêndice C – Seqüência de Chegada dos Lotes | 128 |
| Apêndice D – Tempos de Processamento dos Lotes | 129 |
| Apêndice E – Tempos de Setup | 139 |
| Apêndice F – Resultados da Simulação | 140 |
| Apêndice G – Exemplo de Arquivo de Entrada da Rede Neural | 149 |

Capítulo 1

Introdução

Um sistema de planejamento e controle da produção envolve vários níveis de decisão. As decisões de nível operacional tratam de questões relacionadas diretamente ao chão-de-fábrica, como a designação dos pedidos de clientes às máquinas, a liberação, o processamento e a expedição desses pedidos, entre outras. Essas decisões estão relacionadas à atividade de programação da produção, que se encontra no nível mais detalhado e complexo de um sistema de planejamento e controle da produção (ANTHONY apud HAX e CANDEA, 1984). A programação da produção, também chamada de *scheduling*, estuda a ordem de processamento das peças através das máquinas ao longo do tempo (FRENCH, 1982). Dada a sua natureza combinatorial, vários métodos têm sido propostos como alternativas de solução para resolvê-la. Entre eles encontram-se a técnica de simulação de sistemas e abordagens por inteligência artificial (FERNANDES, 1991).

A simulação de sistemas, como o próprio nome indica, é uma técnica na qual é possível simular em computador o funcionamento de um ambiente de produção. Para isso é construído um modelo em que são definidos todos os parâmetros do ambiente que se deseja estudar, como por exemplo, a quantidade de máquinas e de operadores, os tipos de peças com respectivos roteiros de fabricação, os tempos de processamento das peças nas máquinas, e outros. O modo como as peças são selecionadas da fila de uma máquina particular é definido através de regras de prioridade. Por exemplo, entre várias peças na fila de uma máquina, a peça escolhida para ser processada, naquele instante, pode ser aquela cujo prazo de entrega é o mais crítico. Após a definição de todos os parâmetros e atribuição das regras às máquinas, é simulada a passagem das peças pelo ambiente de produção. Ao final da simulação, são coletados valores de medidas de

desempenho. Essas medidas, como por exemplo, o atraso das peças, o tempo gasto pelas peças em filas, são formas de avaliação do ambiente de produção. Entretanto, os valores das medidas obtidos podem não ser satisfatórios. Neste caso, são atribuídas novas regras de prioridade às máquinas, e simulada novamente a passagem das peças pelo ambiente de produção, num processo de tentativa e erro.

Redes neurais artificiais são técnicas computacionais, que juntamente com sistemas especialistas, algoritmos genéticos, lógica nebulosa e outras, constituem a inteligência artificial. Seu objetivo é simular algumas características de processos inteligentes através da aquisição de conhecimento por experiência. Redes neurais artificiais fazem analogia à estrutura e ao modo como ocorre a transferência de informações em neurônios biológicos (CARVALHO,1999).

Este trabalho combina as duas técnicas descritas anteriormente, simulação de sistemas e redes neurais artificiais, através de um procedimento de solução de um modelo híbrido, aplicado à programação da produção em um ambiente de produção intermitente. Neste procedimento, o papel da rede neural artificial é aprender as relações entre as regras de prioridade designadas às máquinas e os valores das medidas de desempenho utilizados para avaliar as alternativas de programação.

O objetivo é analisar a diferenciação entre um conjunto de combinações de regras de prioridade através da avaliação de quatro medidas de desempenho.

Como o ambiente de produção é um sistema hipotético, este trabalho refere-se a um estudo teórico. Entretanto, as características do ambiente de produção foram definidas visando a aproximação de um sistema real. Dessa forma, este trabalho possui grande potencial de aplicação prático.

Além da introdução, este trabalho contém outros seis capítulos. No capítulo 2 é apresentada uma visão geral sobre sistemas de produção, programação da produção e simulação de sistemas. No capítulo 3 são apresentados os principais conceitos sobre redes neurais artificiais (RNA) e algumas considerações sobre *softwares* de RNA. No capítulo 4 são apresentadas aplicações de redes neurais artificiais na Engenharia de Produção e mais especificamente na programação da produção. No capítulo 5 é apresentado o ambiente de produção e o procedimento de solução do modelo híbrido. No capítulo 6 são apresentados os resultados obtidos. Finalmente, no capítulo 7 são apresentadas as conclusões.

Capítulo 2

Programação da Produção

2.1 Introdução

Neste capítulo será apresentada uma visão geral sobre programação da produção, sua contextualização no ambiente de produção, e alguns conceitos sobre simulação de sistemas.

Também será apresentada uma classificação dos principais problemas de programação da produção e alguns métodos para a solução dos mesmos.

2.2 Sistemas de Produção

Os sistemas de produção industrial, segundo JOHNSON e MONTGOMERY (1974), podem ser classificados em quatro categorias, com base no tipo de produto e no tipo de processo:

- Sistema de produção contínua: processam um grande volume de determinado produto, como em indústrias químicas e refinarias de óleo;
- Sistema de produção intermitente: sempre ocorrem mudanças no tipo de produto fabricado, gerando grande variedade de produtos;
- Sistema de grandes projetos: os produtos são complexos e na maioria das vezes são únicos, como a construção de pontes e edifícios;
- Sistema de estoque puro: é um caso especial em que não há produção e sim distribuição. É o caso de supermercados e atacadistas.

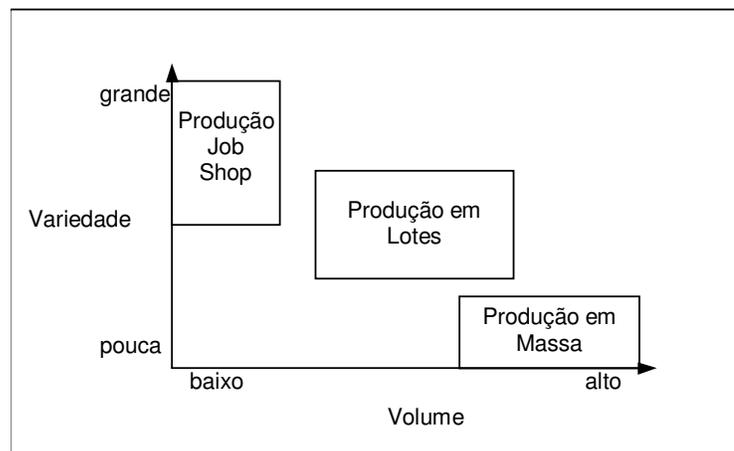
Segundo GROOVER (1987), os sistemas de produção intermitente podem ser classificados de acordo com a quantidade de produtos fabricados:

- Sistema de Produção *Job-Shop*: Caracteriza-se pelo baixo volume de produção. A produção é geralmente feita a partir de pedidos de clientes.

Os lotes são muito pequenos, geralmente unitários. As máquinas devem ser universais para permitir a produção de uma grande variedade de produtos. Para isso os funcionários devem ser qualificados e multifuncionais. Exemplos de produtos fabricados num sistema de produção *job-shop* incluem ferramentas especiais, tapeçarias, protótipos de produtos.

- Sistema de Produção em Lotes: Compreende a manufatura de lotes de tamanho médio. Os lotes podem ser produzidos somente uma vez ou em intervalos regulares. O objetivo é satisfazer continuamente a demanda de um produto, mesmo que para isso haja produção de estoques. As máquinas são de propósito geral, mas projetadas para taxas maiores de produção que no sistema *job-shop*. Produtos como freios, livros-texto, ferramentas manuais são fabricados nesse sistema.
- Sistema de Produção em Massa: Manufatura especializada de produtos idênticos. Caracteriza-se pelas altas taxas de produção, equipamentos dedicados à produção de um produto particular e pelas altas taxas de demanda desse produto. Eletrodomésticos e automóveis são fabricados nesse sistema.

A figura 2.1 apresenta de forma esquemática essa classificação dos sistemas de produção intermitente, considerando o volume de produção e a variedade dos produtos fabricados.



Fonte: GROOVER (1987)

FIGURA 2.1 – Sistemas de produção intermitente

Os sistemas de produção intermitente também podem ser caracterizados em termos do *layout* das máquinas. MCCLAIN e THOMAS (1985) os classificam em três categorias:

- *Layout* por processo: as máquinas são arranjadas de acordo com a sua função, por exemplo, todos os tornos em uma área e todas as furadeiras em outra. Este *layout* é também chamado de funcional. No sistema de produção *job-shop* as máquinas são arranjadas segundo esse *layout*.
- *Layout* por produto: as máquinas são arranjadas de acordo com a seqüência de operações pelas quais o produto passa. No sistema de produção em massa, cujo exemplo mais típico é a linha de montagem, as máquinas são arranjadas segundo esse *layout*.
- *Layout* de posições fixas: o produto é fixo em um local e os recursos se movem até ele para realizar o processamento. Este é o caso da construção de edifícios, navios e de áreas para estoque.

O modo como os produtos se movem no interior de um sistema de produção é denominado padrão de fluxo. Num sistema de produção intermitente, os padrões de fluxo são do tipo *flow-shop* e *job-shop*. Num sistema com padrão de fluxo *flow-shop*, todos os produtos possuem a mesma seqüência de operações nas diversas máquinas, ou seja, todos os produtos possuem o mesmo roteiro de fabricação. Num sistema com padrão de fluxo *job-shop*, cada produto possui uma seqüência de operações através das máquinas, ou seja, cada produto possui um roteiro de fabricação diferente.

No sistema de produção em massa, o padrão de fluxo é do tipo *flow-shop*. No sistema de produção *job-shop*, como o próprio nome indica, o padrão de fluxo é do tipo *job-shop*. No sistema de produção em lotes, tanto pode haver padrão de fluxo *flow-shop* como *job-shop*. É importante observar que a palavra *job-shop* tanto pode significar um tipo de sistema de produção intermitente como um padrão de fluxo de materiais através das máquinas.

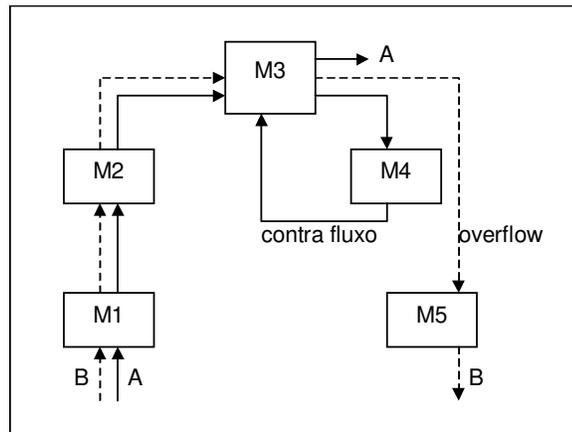
2.3 Célula de Manufatura

A célula de manufatura, segundo FERNANDES (1991), é formada por um conjunto de máquinas que devem estar agrupadas não por possuírem a mesma função, mas pelo fato de juntas serem capazes de realizar um conjunto de operações diferentes, suficientes para produzir uma gama de peças completas na saída.

Um dos princípios para a formação de uma célula de manufatura é a tecnologia de grupo (TG). De acordo com GROOVER (1987), a filosofia da TG é identificar peças com similaridades de projeto e de processo e agrupá-las em famílias, para que sendo processadas pelo mesmo conjunto de máquinas aumente a eficiência do sistema. Não é o objetivo deste trabalho discutir métodos para a formação de famílias de peças. GROOVER (1987) é uma boa referência sobre o tema.

A formação das famílias de peças permite a fabricação de pequenos lotes de produtos, e o aumento da diversidade dos mesmos. Isso é possível devido à redução do tempo de *setup* entre os lotes de peças. Dessa forma, a manufatura celular encontra-se num nível intermediário entre um sistema de produção *job-shop* e um sistema de produção em massa. É menos flexível que um sistema *job-shop*, mas por outro lado a produtividade é maior.

Tanto podem existir células com padrão de fluxo *flow-shop* como células com padrão de fluxo *job-shop*. Na manufatura celular com padrão de fluxo *job-shop*, segundo FERNANDES (1991) a mais usual, há *overflow* e contra-fluxo. No primeiro a peça pode “pular” uma máquina, e no segundo pode voltar para uma máquina, como representado na figura 2.2.



Fonte: Adaptado de FERNANDES (1991)

FIGURA 2.2 – Fluxo típico na manufatura celular

2.4 Planejamento e Controle da Produção

O termo Planejamento e Controle da Produção, segundo PIRES (1995), pode ser definido como um conjunto de atividades gerenciais fundamentais para que se concretize a produção de um item ou produto qualquer. Entretanto, como aborda PIRES (1995), não existe na literatura um consenso a respeito da terminologia sobre o tema. A seguir são apresentadas três abordagens.

Segundo BURBIDGE apud PIRES (1995), o planejamento da produção é uma função da administração que deve planejar os recursos físicos utilizados para a produção de bens ou serviços, incluindo a definição do roteiro de fabricação, do sistema de armazenagem e movimentação, e do *layout*. BURBIDGE (1983) define o controle da produção como uma função da administração que planeja, dirige e controla o suprimento de materiais e as atividades de processo de uma empresa. Este autor considera três níveis de controle da produção:

- Plano de produção: determina o volume de produção dos produtos acabados;

- Emissão de ordens: planeja a entrada de materiais vindos dos fornecedores considerando prazos, quantidades e o volume de produção das peças fabricadas.
- Liberação ou *dispatching*: programação diária do volume de produção por máquina, feita para completar as ordens na data estabelecida.

Para FERNANDES (1991), as questões de longo prazo com horizonte de pelo menos um ano devem ser tratadas pelo planejamento estratégico da empresa, como a expansão da capacidade produtiva, e alterações significativas na linha de produtos. O planejamento da produção deve ser encarregado das questões de médio prazo, de um a dezoito meses, como planejar a capacidade de produção de acordo com a previsão de demanda. O controle da produção deve tratar de questões de curto prazo, com horizonte de um a três meses. O sistema de controle da produção proposto por FERNANDES (1991) é constituído pelo programa mestre de produção (PMP), pelo sistema de emissão de ordens e pela programação da produção. O PMP decide o que, quanto e quando fabricar. A emissão de ordens é basicamente um sistema de informações que converte as necessidades colocadas no PMP e de produtos finais em termos de componentes. A programação da produção é a atividade mais específica e complexa do controle da produção, dada sua natureza combinatorial.

De acordo com a abordagem proposta por ANTHONY apud HAX e CANDEA (1984), os sistemas de planejamento e controle da produção envolvem três níveis de decisões: estratégico, tático e operacional.

As decisões de nível estratégico englobam a definição dos objetivos da organização, mudança de objetivos, os recursos utilizados para atingir tais objetivos, e as políticas que regem a aquisição, uso e disposição desses recursos. As decisões de nível estratégico envolvem questões de longo prazo como decisões sobre localização das fábricas e armazéns, compra de equipamentos, expansão da capacidade produtiva ou construção de novas unidades fabris.

As decisões de nível tático referem-se à forma como os gerentes asseguram que os recursos serão obtidos e utilizados eficaz e eficientemente na realização dos objetivos da organização. As decisões de nível tático envolvem questões de médio prazo como a definição dos canais de distribuição, níveis de estoque, utilização de máquinas e mão-de-obra, seleção de transporte, e outras.

As decisões de nível operacional referem-se ao processo que assegura que tarefas específicas serão executadas eficaz e eficientemente. As decisões de nível operacional envolvem questões de curto prazo como a designação de ordens de clientes à máquinas individuais, liberação, processamento e expedição dessas ordens no chão-de-fábrica. Essas decisões estão relacionadas à atividade de programação da produção, objeto deste trabalho e que será detalhada a seguir.

2.5 Programação da Produção

Na literatura, os termos programação de operações e programação da produção se confundem. O termo programação de operações é mais genérico, e pode ser usado tanto na manufatura como na área de serviços. Como este trabalho engloba especificamente a manufatura, será utilizado o termo programação da produção.

Segundo CONWAY et al. (1967), as hipóteses gerais de seqüenciamento e programação da produção são:

- as tarefas (peças) a serem executadas são previamente conhecidas. Uma tarefa é composta por um conjunto de operações;
- todas as tarefas deverão ser completadas;
- os recursos (máquinas, mão-de-obra, ferramentas) utilizados para a execução das tarefas são especificados anteriormente;

- o roteiro de fabricação de cada tarefa é conhecido, assim como o seu método de execução. E deve existir pelo menos um conjunto de recursos disponíveis para executar cada operação.

No estudo da programação da produção, é preciso diferenciar seqüenciamento da programação da produção propriamente dita (ou chamada de *scheduling*). Para CONWAY et al. (1967), seqüenciamento é a ordenação das tarefas em uma única máquina e programação da produção é a seqüência simultânea e sincronizada em várias máquinas. Para FRENCH (1982), seqüenciamento “é simplesmente a ordem na qual as tarefas são processadas através das máquinas. Ele não contém informações explícitas sobre o tempo no qual as várias tarefas iniciam e terminam. A programação da produção, entretanto, contém o tempo, assim como a informação do seqüenciamento”.

Segundo CONWAY et. al. (1967), a programação da produção em um ambiente *job-shop* pode ser caracterizada por algumas hipóteses básicas. A seguir são apresentadas as hipóteses que serão consideradas neste trabalho:

- cada máquina está continuamente disponível para a produção;
- as tarefas são seqüências de operações estritamente ordenadas, sem montagem ou partição. Isso significa que o roteiro de fabricação da tarefas é conhecido e não são permitidos roteiros alternativos. Ainda, cada operação deve ter no máximo uma operação que a precede diretamente e uma que a sucede diretamente;
- cada operação pode ser feita por somente uma máquina na fábrica, isto é, as máquinas não são intercambiáveis;
- existe somente uma máquina de cada tipo na fábrica;
- não é permitida a interrupção da operação, isto é, uma vez que uma operação é iniciada numa máquina ela deve ser completada, antes que outra operação inicie naquela máquina;

- os tempos de processamento de operações sucessivas de uma tarefa não podem ser sobrepostos, isto é, uma tarefa pode processar no máximo uma operação por vez e não são permitidas operações consecutivas de uma tarefa na mesma máquina.

HAX e CANDEA (1984) classificam o processo de chegada das tarefas em um sistema como estático ou dinâmico, e o tempo de processamento (p_i) das tarefas como determinístico ou probabilístico. Essa classificação é apresentada na tabela 2.1.

Tabela 2.1 – Natureza do processo de chegada e do tempo de processamento

| Processo de chegada | Natureza do p_i | Descrição |
|-------------------------|-------------------|--|
| Estático | Determinístico | n tarefas disponíveis em $t=0$ p_i conhecido quando as tarefas chegam ao sistema |
| Estático | Probabilístico | n tarefas disponíveis em $t=0$ p_i desconhecido qdo as tarefas chegam ao sistema (descrito por uma distribuição de probabilidade) |
| Dinâmico determinístico | Determinístico | n tarefas chegam em tempos futuros conhecidos p_i conhecido quando as tarefas chegam ao sistema |
| Dinâmico determinístico | Probabilístico | n tarefas chegam em tempos futuros conhecidos p_i desconhecido qdo as tarefas chegam ao sistema (descrito por uma distribuição de probabilidade) |
| Dinâmico probabilístico | Determinístico | Chegada Poisson (contínua, horizonte infinito) p_i conhecido quando as tarefas chegam ao sistema |
| Dinâmico probabilístico | Probabilístico | Chegada Poisson (contínua, horizonte infinito) p_i desconhecido qdo as tarefas chegam ao sistema (descrito por uma distribuição de probabilidade) |

Fonte: HAX e CANDEA (1984)

Vários modelos de programação da produção consideram o tempo de *setup* necessário para realizar a troca de peças ou lotes em uma máquina incluído no tempo de processamento. Isso permite a simplificação do modelo. Entretanto, em ambientes reais, o *setup* está presente e há uma preocupação na sua redução.

A escolha do critério de avaliação da programação da produção, ou também conhecido como medida de desempenho, dependerá do objetivo da programação, como por exemplo, redução do estoque em processo, atendimento aos prazos de entrega.

Segundo CONWAY et al. (1967), as medidas de desempenho geralmente utilizadas para avaliar a programação da produção estão associadas ao congestionamento do sistema, ao cumprimento aos prazos de entrega, e à utilização eficiente dos recursos. O congestionamento do sistema pode ser avaliado pela quantidade de estoque em processo, pelo tempo de permanência das peças no sistema ou também chamado tempo médio de fluxo, entre outras. O cumprimento aos prazos de entrega pode ser medido pelo número ou taxa de tarefas atrasadas, pelo tempo de atraso médio ou total das tarefas. Uma medida de desempenho associada à utilização eficiente dos recursos é a que objetiva minimizar a duração total da programação, também chamada de *makespan*.

O modo como as operações são designadas em tempos específicos às máquinas apropriadas, em um ambiente de programação da produção, é determinado de acordo com disciplinas ou regras de prioridade. O emprego dessas regras está relacionado às medidas de desempenho que se pretendem coletar num estudo.

As regras de prioridade, segundo HAX e CANDEA (1984), são classificadas de duas formas. A primeira refere-se à disponibilidade da informação. Nesse sentido as regras são classificadas em locais ou globais. Para as regras locais, a prioridade de uma operação é determinada por parâmetros das tarefas que se encontram na fila da máquina que realizará aquela operação. Esses parâmetros podem ser tempo de processamento, data de entrega, e outros. Para regras globais a prioridade é calculada considerando também as filas das outras máquinas. A segunda classificação refere-se à variação dos índices de prioridade. Para as regras estáticas os índices de prioridade não mudam com o tempo, como a data de entrega. Já

para as regras dinâmicas os índices de prioridade mudam em função do tempo, como a folga dinâmica.

Existe uma infinidade de regras de prioridade de tarefas às máquinas. Não é objetivo deste trabalho relacionar todas. A seguir são apresentadas algumas delas.

- FIFO: *First-in-first-out* - a primeira peça a entrar na fila é a primeira a ser atendida;
- SPT: *Shortest processing time* - seleciona da fila a peça com o menor tempo de processamento;
- EDD: *Earliest due date* - seleciona da fila a peça com menor data de entrega;
- SLACK: seleciona da fila a peça com menor folga dinâmica (data de entrega menos o tempo atual menos o tempo de processamento remanescente);
- STT: seleciona da fila a peça com menor soma do tempo de processamento e do tempo de *setup*.

2.5.1 Classificação dos Problemas de Programação da Produção e Principais Métodos de Solução

Para identificar os problemas de programação da produção, CONWAY et al. (1967) propuseram a notação A / B / C / D, na qual:

A : descreve o processo de chegada das tarefas. Para problemas estáticos, especifica o número de tarefas a serem programadas, e para dinâmicos, a distribuição de probabilidade dos tempos entre chegadas.

B : descreve o número de máquinas.

C : descreve o padrão de fluxo: *F* - *flow-shop*; *G* - *job-shop*; *O* - *open-shop*. Se B = 1, ou seja, somente uma máquina, C não irá existir.

D : descreve o critério para avaliar a programação. A escolha do critério de avaliação ou também conhecido como medida de desempenho dependerá do objetivo da programação, como redução do estoque em processo e

atendimento aos prazos de entrega. Segundo a notação de CONWAY et al. (1967), um problema do tipo 4/3/G/M refere-se à programação de quatro tarefas em três máquinas, em um ambiente *job-shop*, com o objetivo de minimizar o *makespan*, ou duração total da programação.

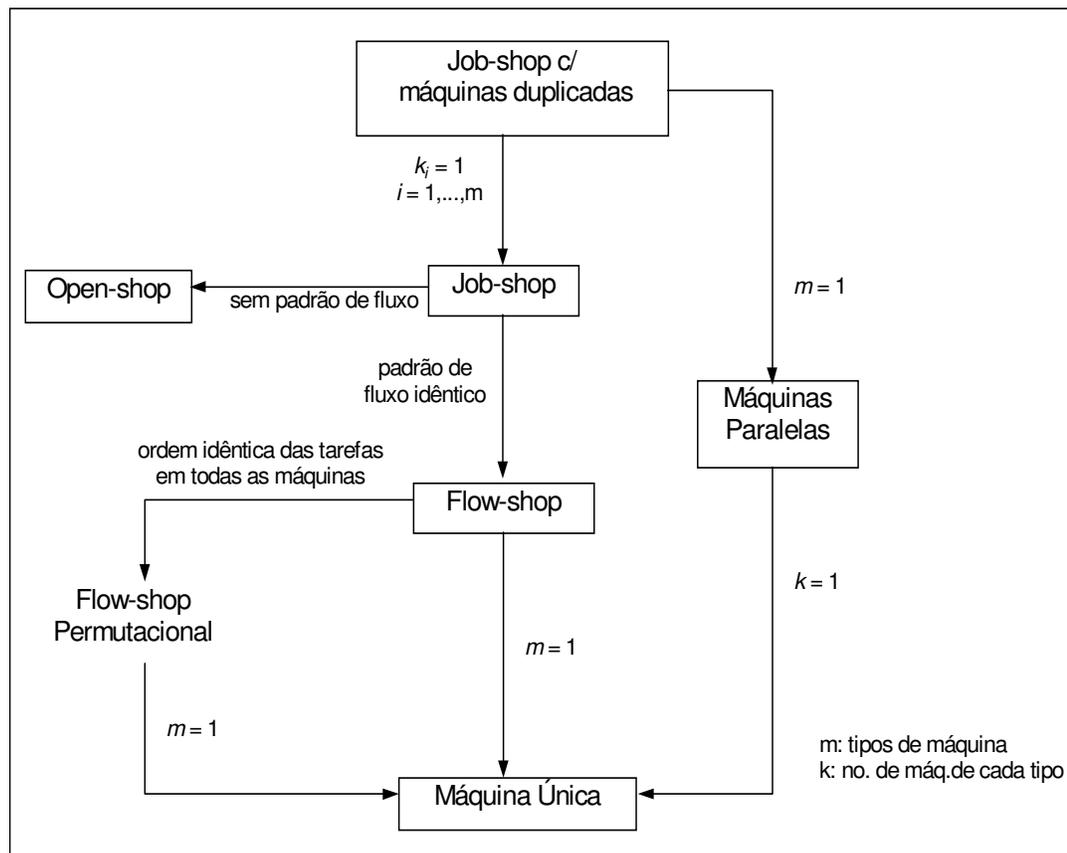
De acordo com MACCARTHY e LIU (1993), a classificação dos problemas de programação da produção deve considerar os objetivos da programação e as restrições tecnológicas, determinadas principalmente pelo padrão de fluxo das tarefas nas máquinas e pelo número e tipo de máquinas disponíveis. Neste contexto esses autores propuseram a seguinte classificação:

- *Job-shop*: existem m tipos de máquinas com uma máquina de cada tipo ($k_i = 1$ para qualquer i), com $i = 1, 2, \dots, m$. Cada tarefa tem seu próprio padrão de fluxo ou rota específica através das máquinas. Por exemplo, a fabricação de máquinas como tornos.
- *Flow-shop*: assim como no *job-shop* existem m tipos de máquinas com uma máquina de cada tipo ($k_i = 1$ para qualquer i). Aqui as tarefas têm o padrão de fluxo idêntico, ou seja, todas as tarefas passam pelas mesmas máquinas. Por exemplo, a montagem de geladeiras.
- *Open-shop*: não existe padrão de fluxo especificado para nenhuma tarefa, ou seja, a requisição das máquinas pelas tarefas é imprevista. Devido à diversidade de produtos e de mercado, a produção não é preestabelecida.
- *Flow-shop* permutacional: um *flow-shop* no qual a ordem de processamento das tarefas nas máquinas é a mesma.
- Máquina única: somente uma máquina está disponível ou quando os problemas de programação da produção podem ser críticos nela. Por exemplo, uma máquina “gargalo” em que há formação de estoques intermediários.
- Máquinas paralelas: existe somente um tipo de máquina ($m = 1$) com k_i máquinas idênticas. Cada tarefa necessita de uma e somente uma

dessas máquinas. Por exemplo, uma oficina onde há somente serras e todas são idênticas.

- *Job-shop* com máquinas duplicadas: um *job-shop* onde existem k_i máquinas idênticas em cada estágio ($i = 1, 2, \dots, m$) e cada tarefa requer somente uma máquina em cada estágio.

A figura 2.3 a seguir ilustra essa classificação.



Fonte: MACCARTHY e LIU (1993)

FIGURA 2.3 - Classificação dos problemas de programação da produção

Não foi encontrado na literatura (FERNANDES, 1991; MACCARTHY e LIU, 1993; MOCCELLIN, 1994; PINEDO, 1995; BLAZEWICZ et al., 1996; PORTMANN, 1997) um consenso quanto à classificação dos métodos para solução de problemas em programação da produção. Segundo FERNANDES (1991), as abordagens existentes para resolver tais problemas são:

- Para problemas estáticos:
 - métodos otimizantes (ou exatos)
 - métodos heurísticos (ou aproximados) com ou sem o emprego de técnicas de inteligência artificial (IA).
- Para problemas dinâmicos:
 - teoria dos processos estocásticos
 - e/ou simulação com ou sem o emprego de técnicas de IA.

Um método aproximado para solução de problemas da programação da produção que vem despertando interesse é a teoria das restrições. O enfoque é sobre os recursos críticos de um sistema de produção, também chamados de recursos gargalos (PIRES, 1995; GOLDRATT e COX, 1992).

PORTMANN (1997) faz uma classificação mais detalhada das metodologias de solução de problemas de programação da produção classificando-as em: métodos construtivos, de vizinhança, de decomposição, de troca de modelo, abordagens por inteligência artificial e abordagens mistas.

Os métodos construtivos referem-se a qualquer método iterativo que inicia com um conjunto vazio ou parcial de decisões e a cada passo uma programação parcial é completada por uma nova decisão: a posição de uma tarefa ou operação na seqüência. O processo termina quando uma programação completa é obtida. Entre eles PORTMANN (1997) cita: regras heurísticas SPT, EDD; algoritmos de Jonhson, Moore, CDS, Dannenbring.

Os métodos de vizinhança, ou também conhecidos como métodos de busca local, a cada passo estudam a vizinhança local de uma dada solução completa tentando melhorá-la. Os mais conhecidos são *best ascendant*, *hill climbing*, busca Tabu, *simulated annealing* e algoritmos genéticos (PORTMANN, 1997).

Os métodos de decomposição, como o próprio nome indica decompõem o problema original em subproblemas mais simples de resolver. Podem ser dos tipos hierárquico (decomposição do problema em níveis),

estrutural (decomposição do conjunto de restrições), e *branch and bound* (decomposição do conjunto de soluções) (PORTMANN, 1997).

Os métodos de troca de modelo modificam o modelo para simplificá-lo. Todos os métodos de relaxação pertencem a essa classe, como relaxação inteira, Lagrangeana (PORTMANN, 1997).

Entre as abordagens por inteligência artificial (IA) estão todas aquelas ferramentas usadas em IA que não pertencem às classes anteriores, como programação por restrições, sistemas especialistas, redes neurais, lógica nebulosa e outras (PORTMANN, 1997).

Entre as abordagens mistas estão o *shifting bottleneck method* (BLAZEWICZ et al., 1996) e aquelas que combinam por exemplo os métodos de relaxação e de decomposição (PORTMANN, 1997).

2.6 Simulação de Sistemas

A simulação de sistemas, segundo GORDON (1978), é a técnica de solução de problemas, pela observação do desempenho, no tempo, de um modelo dinâmico do sistema, e nesse sentido é um dos métodos de solução de problemas de programação da produção.

A técnica de simulação de sistemas pode ser aplicada no estudo da programação da produção tanto em sistemas reais como hipotéticos. No primeiro caso são implantadas regras de prioridade às máquinas e é avaliado o desempenho atual. Possui a vantagem do realismo, entretanto há perda de generalidade. Em sistemas hipotéticos são designadas às máquinas regras de prioridade, que são avaliadas através de várias medidas de desempenho. Uma das principais vantagens é a generalização dos resultados.

Num estudo de simulação primeiramente há a geração de dados sobre a forma de chegada das tarefas no sistema, os roteiros de fabricação, os tempos de processamento e as datas de entrega. Além disso, são definidos vários parâmetros como o número e tipo recursos (máquinas, mão-

de-obra), *layout* dos equipamentos, tamanho do lote de peças, tempos de *setup*, regras de prioridade e outros.

Depois de definidos todos os parâmetros a simulação inicia com a passagem das peças pelo sistema segundo as regras de prioridade designadas às máquinas. Ao final da simulação são coletados valores de medidas de desempenho sob aquelas regras e é analisada a influência das diversas regras no desempenho da fábrica.

Os desenvolvimentos ocorridos na área da programação da produção são extensos. Neste capítulo foram apresentados somente os conceitos necessários para a compreensão do ambiente de produção em estudo.

A seguir será apresentada uma visão geral da teoria sobre redes neurais artificiais.

Capítulo 3

Redes Neurais Artificiais

3.1 Introdução

A busca da compreensão de como ocorre a transferência de informações no cérebro humano e dos processos de aprendizagem é antigo. As pesquisas em redes neurais artificiais (RNA) iniciaram com McCulloch e Pitts em 1943, que propuseram um modelo lógico binário para descrever os neurônios. Este trabalho teve grande impulso em 1958 com Rosenblatt que propôs o modelo chamado *Perceptron*. Mas em 1969, o trabalho de Minsky e Papert levou a área de redes neurais praticamente à estagnação, por provar as limitações dos modelos até então existentes. Esse desaquecimento durou até a década de 80, quando em 1982 Hopfield relatou a utilização de redes para otimização, e em 1986 Rumelhart, McClelland e Williams propuseram o algoritmo *back propagation* que será abordado adiante, ressurgindo então o interesse por redes neurais artificiais (BEALE e JACKSON, 1990).

Redes neurais artificiais têm sido aplicadas em diversas áreas do conhecimento como: diagnóstico de doenças em medicina; detecção de fraudes em sistemas de crédito e previsão do mercado de ações na área financeira, reconhecimento de caracteres e imagens em computação, robótica e controle motor em engenharia (SHARDA e WANG, 1996). No capítulo 4 deste trabalho são apresentadas algumas aplicações de RNA em Engenharia de Produção e especificamente na programação da produção.

Redes neurais artificiais correspondem a uma ferramenta computacional que juntamente com outras como sistemas especialistas, lógica nebulosa, algoritmos genéticos, constituem a inteligência artificial. Segundo ZAHEDI (1991), RNA e sistemas especialistas têm o objetivo comum de simular a inteligência humana, mas utilizam métodos diferentes. Sistemas especialistas consideram o cérebro como uma “caixa preta”, isto é, tratam-no de forma macroscópica sem analisar sua estrutura biológica.

Processam o conhecimento seqüencialmente e utilizam o raciocínio dedutivo humano. Redes neurais consideram o cérebro como uma “caixa branca”, isto é, procuram entender seu aspecto estrutural e funcional. Processam o conhecimento em paralelo, o que é conseguido na maioria das vezes através de simulações em *software*.

3.2 Definição de Redes Neurais Artificiais (RNA)

Redes neurais artificiais são elementos computacionais inspirados em neurônios biológicos, que conectados em rede podem reproduzir algumas características de processos inteligentes (ARAÚJO, 1996).

Segundo HANSON e BURR (1991), RNA, também chamadas de modelos conexionistas, possuem três características gerais:

- são formadas por elementos computacionais básicos ou unidades de processamento (UP);
- as UP são arranjadas em várias arquiteturas possíveis (camadas), com padrões arbitrários de interconexões;
- a atualização da intensidade das conexões entre as UP ocorre segundo regras recursivas (ou regras de aprendizagem).

Essas três características - elementos computacionais, arquitetura e regras de aprendizagem - são independentes da tarefa a ser realizada.

3.3 O Neurônio Humano

Um neurônio, de modo simplificado, é formado pelos dendritos, que são um conjunto de terminais de entrada, pelo corpo celular, e pelo axônio (terminal de saída) (figura 3.1). Os dendritos recebem as informações de outros neurônios e as enviam para o corpo celular que é responsável pelo processamento biológico da célula. A informação é transmitida através do

axônio, conectado a outros neurônios, numa ligação chamada sinapse. Essa ligação permite ou não a passagem da informação, na forma de impulso elétrico, para a outra célula. Essa transferência ou não de informação é determinada por um mecanismo de troca de íons sódio e potássio, responsável pelo potencial de ação do neurônio (pulso de voltagem). Se esse potencial exceder um valor limitante acontece o disparo, transmitindo o impulso elétrico.

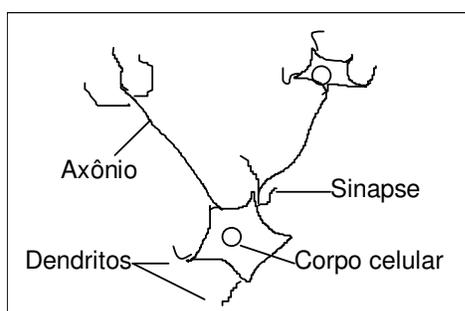


FIGURA 3.1 - Modelo de neurônio humano

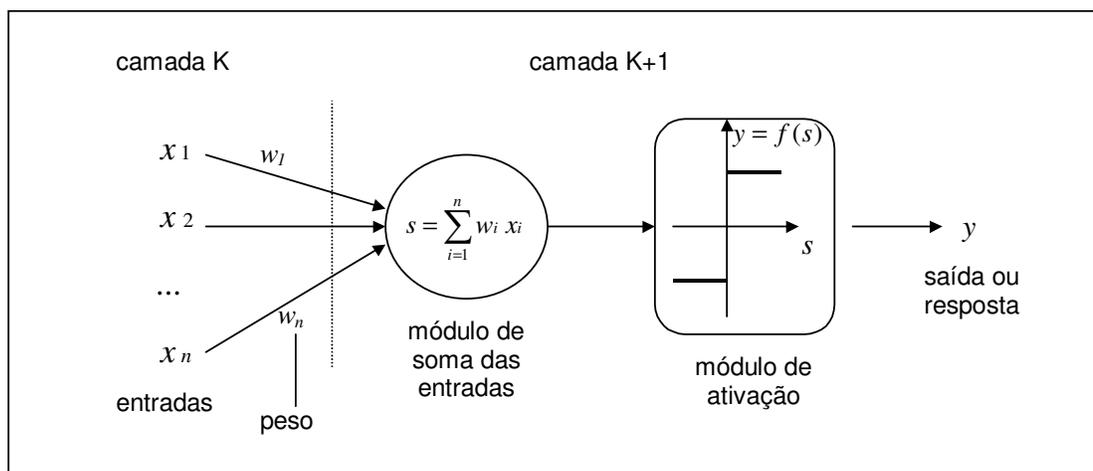
O cérebro humano contém cerca de 10^{11} neurônios. Cada neurônio forma entre mil e dez mil sinapses, o que possibilita a formação de redes muito complexas (CARVALHO, 1999).

3.4 O Modelo Artificial

Primeiramente é preciso salientar que existem grandes diferenças entre a célula nervosa humana e o modelo de neurônio artificial, mas o princípio de transferência das informações é o mesmo.

Os neurônios artificiais são interligados de forma semelhante como acontece nos neurônios humanos. Um neurônio recebe os sinais (informações) de entrada, x_i , de outros neurônios. É feita uma média ponderada entre os sinais de entrada e os pesos das conexões, w_i . O resultado, s , é aplicado a uma função de ativação (ou função de transferência). O resultado final é chamado de ativação do neurônio. Existe um valor s que quando atingido provoca um “disparo”, ocasionado pela

função de ativação, transferindo a informação. Este valor é chamado de limitante, L . A figura 3.2 apresenta este modelo de neurônio artificial, chamado *Perceptron* (LOESCH e SARI, 1996).



Fonte: LOESCH e SARI (1996)

FIGURA 3.2 - Modelo de neurônio artificial

A equação (3.1) mostra o processamento efetuado na figura 3.2.

$$s = \sum_{i=1}^n w_i x_i \quad \text{e} \quad y = f(s) = \begin{cases} +1 & \text{se } s \geq L \\ -1 & \text{se } s < L \end{cases}, \text{ na qual} \quad (3.1)$$

L = limitante

$f(s)$ = função de ativação

Uma outra forma de representação do modelo de neurônio artificial é adicionar uma entrada sempre ativa de valor igual a 1 (um) à UP, conhecida como *bias*. Embora não exista no neurônio biológico, a função do *bias* é transladar o valor do limitante, L , da função de ativação. Dessa forma, a equação (3.1) pode ser reescrita da forma:

$$s = \sum_{i=1}^n w_i x_i - L \quad \text{e} \quad y = f(s) = \begin{cases} +1 & \text{se } s \geq 0 \\ -1 & \text{se } s < 0 \end{cases} \quad (3.2)$$

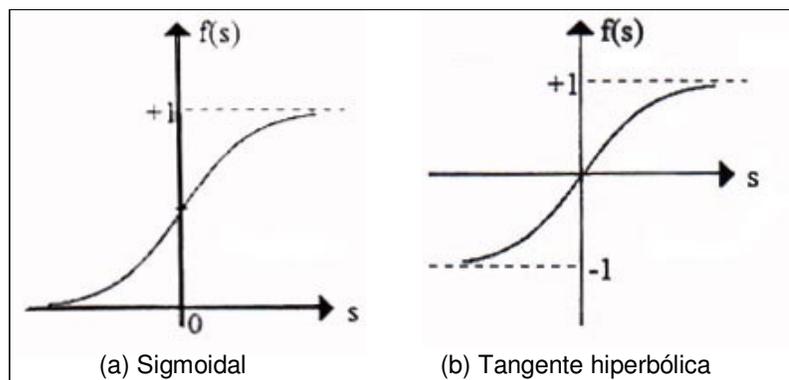
Dessa forma, $-L$, atua como um peso para a entrada adicional (*bias*). A atuação do limitante será mais detalhada na seção 3.8.

A função de ativação apresentada no módulo de ativação da figura 3.2 e utilizada na equação (3.1) é chamada função degrau. Entretanto, segundo BISHOP (1995), a maioria das funções de ativação empregadas em aplicações atuais são não-lineares, como:

a função sigmoidal: $f(s) = \frac{1}{1 + e^{-s}}$

e a função tangente hiperbólica: $f(s) = \tanh(s) = \frac{e^s - e^{-s}}{e^s + e^{-s}}$

Essas duas funções são apresentadas na figura 3.3.



Fonte: LOESCH e SARI (1996)

FIGURA 3.3 - Funções de ativação

3.5 Características das Redes Neurais Artificiais

As características mais importantes das redes neurais artificiais são (ARAÚJO, 1996 e HRYCEJ, 1992):

- Capacidade de “aprender” as relações entre um conjunto de dados de entrada, também chamados de exemplos de treinamento, e com isso

melhorar o seu desempenho. Essa capacidade deve-se ao algoritmo de treinamento ou aprendizagem, que será abordado adiante.

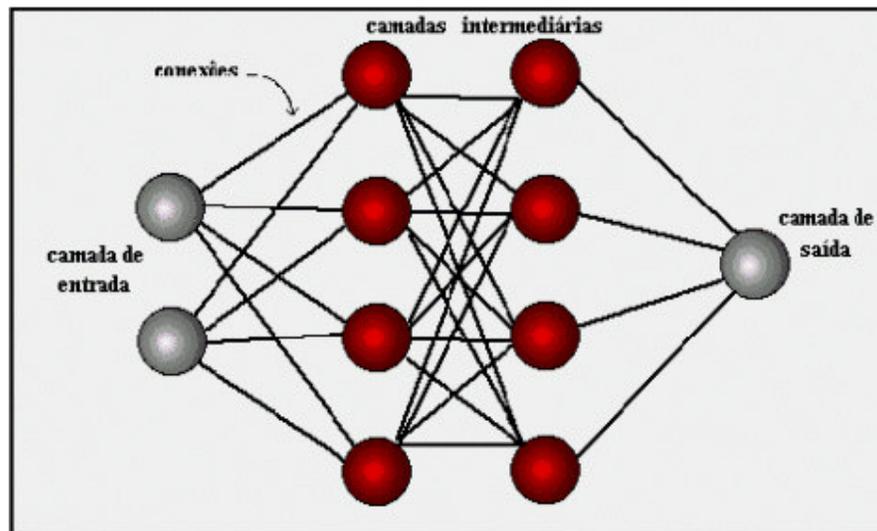
- Capacidade de generalizar o aprendizado para novos exemplos. As redes podem fornecer respostas similares àsquelas para as quais foi treinada, para exemplos não apresentados no treinamento.
- Capacidade de extrair a essência de um conjunto de dados e aprender a partir de informações incompletas.

3.6 Estrutura das Redes Neurais Artificiais

As redes neurais artificiais podem ser descritas de acordo com as propriedades da rede, da célula, propriedades dinâmicas e propriedades de aprendizagem (GALLANT, 1993).

Propriedades da Rede

Em vários modelos, os neurônios, ou unidades de processamento (UP), são organizados em camadas, denominadas camada de entrada, escondida ou interna e de saída. Em geral, as UP da camada de entrada somente transmitem o valor da entrada para as próximas UP. Não ocorre nenhum processamento nelas. A camada interna permite a modelagem não linear da rede. Dependendo do modelo pode haver mais de uma camada interna. A camada de saída apresenta os resultados da rede, as respostas para o problema analisado. A figura 3.4 apresenta um exemplo de organização em camadas.



Fonte: CARVALHO (1999)

FIGURA 3.4 – Organização em camadas

As RNA são classificadas como redes *feedforward* ou redes unidirecionais quando as conexões possuem somente um sentido, da entrada para a saída, ou seja, se não possuem ciclos. São classificadas como redes de *feedback* ou recorrentes se contém tais ciclos.

Propriedades da Célula

Cada UP calcula uma única saída. As saídas podem ser discretas (0 e 1 ou -1 e +1) ou contínuas assumindo valores dentro de um intervalo, por exemplo, [0,1].

Propriedades Dinâmicas

Um modelo de RNA deve especificar quando cada UP calcula sua nova ativação. Geralmente as UP avaliam e atualizam sua ativação antes que a próxima UP seja “visitada”. Este modo de atualização é chamado de ordem topológica, ou seja, da entrada para a saída da rede. Existem outras formas de atualização das ativações como a simultânea, em que as ativações de todas as UP são mudadas ao mesmo tempo, e a aleatória, em que a ordem de atualização das ativações é definida por um gerador de números aleatórios. Neste último caso algumas UP podem ser atualizadas várias vezes e outras não (SNNS, 1995).

Propriedades de Aprendizagem

Os métodos de aprendizagem da rede, segundo HRYCEJ (1992), podem ser supervisionado ou não-supervisionado. O método de aprendizagem supervisionado utiliza um "professor", ou seja, para cada padrão (exemplo) de entrada apresentado à rede, é fornecida a saída desejada para aquele padrão. A aprendizagem ocorre através de um processo de minimização do erro (diferença entre a saída obtida e a desejada) e do ajuste dos pesos das conexões. Este modo de aprendizagem é utilizado pelas redes *Perceptron* de Múltiplas Camadas - MLP- que serão vistas adiante. No método de aprendizagem não-supervisionado não é fornecida a saída desejada da rede. A partir da regra de aprendizagem as UP devem descobrir sozinhas correlações entre os dados de entrada e evoluir, para soluções ótimas ou quase ótimas. As redes de Kohonen utilizam esse método de aprendizagem (GALLANT, 1993).

3.7 Principais Arquiteturas de Redes Neurais Artificiais

Diversas arquiteturas de redes neurais já foram desenvolvidas. A seguir são apresentadas as principais arquiteturas com aplicações, vantagens e desvantagens (LOESCH e SARI, 1996).

Rede *Perceptron* de Múltiplas Camadas (*Multilayer Perceptron* - MLP)

Esta rede é uma das mais empregadas em aplicações de RNA. Utiliza como regra de aprendizagem o algoritmo *back propagation*, que será detalhado adiante. A aprendizagem é supervisionada. A rede é formada por camadas de entrada, escondida e de saída, totalmente conectadas, com conexões unidirecionais. Principais aplicações: classificação, reconhecimento de padrões visuais, otimização, controle robótico. Vantagens: realiza boa representação interna dos dados de entrada e é bem compreendida. Desvantagem: tempo de treinamento longo.

Rede de Hopfield

Esta rede é de aprendizagem supervisionada e foi aplicada primeiramente na resolução do problema do caixeiro viajante (BEALE e JACKSON, 1990). Principais aplicações: otimização e auto-associação. Vantagem: facilidade de implementação em circuitos integrados. Desvantagens: armazenamento pobre de memória, tendência em estabilizar em mínimos locais de energia e não atingir o mínimo global (LIPPMAN, 1987).

Rede de Kohonen

A rede de Kohonen é de aprendizagem não-supervisionada. Dado um conjunto de entrada, a rede realiza sua própria classificação desse conjunto, agrupando exemplos com características comuns dentro de uma mesma classe. Principais aplicações: reconhecimento da fala (discurso), processamento de informações semânticas, otimização (BEALE e JACKSON, 1990). Vantagem: capacidade de auto-organização. Desvantagem: treinamento longo.

Máquina de Boltzmann

Esta rede combina o algoritmo de Hopfield com a técnica *simulated annealing*. Esta técnica baseia-se em um modelo termodinâmico de aquecimento e resfriamento de metais para encontrar um mínimo global em uma superfície de energia. Principais aplicações: otimização, reconhecimento de imagens. Vantagem: boa representação das características dos padrões. Desvantagem: tempo de aprendizado longo (BEALE e JACKSON, 1990).

Redes ART (*Adaptative Resonance Theory*)

As redes ART são de aprendizagem não-supervisionada. Principais aplicações: reconhecimento de padrões (radar/sonar), sistemas de visão e controle motor (ARAÚJO, 1996; LIPPMAN, 1987). Vantagem: capacidade de aprender novas classes de padrões sem detrimento às já aprendidas.

Desvantagens: sensibilidade a ruídos nos dados de entrada e estrutura complexa.

Rede BAM (*Bidirectional Associative Memory*)

Principal aplicação: associa pares de padrões, como por exemplo o nome e telefone de uma pessoa (BEALE e JACKSON, 1990). Vantagens: dispensa o pré-processamento dos dados de entrada e é de fácil treinamento. Desvantagem: baixa capacidade de armazenamento.

3.8 O Algoritmo *Back Propagation*

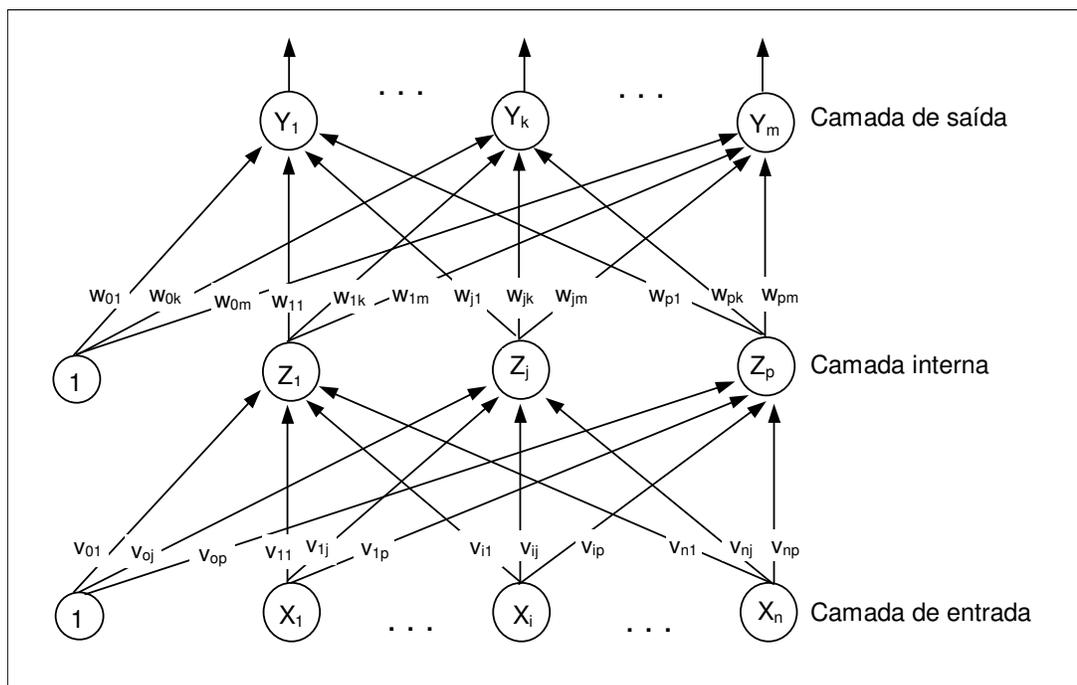
O algoritmo *back propagation* utiliza a aprendizagem supervisionada. O treinamento de uma rede neural com este algoritmo envolve duas fases. Primeiro, um padrão (exemplo) é apresentado à camada de entrada da rede. A atividade resultante das ativações dos neurônios flui através da rede, camada por camada, até que seja produzida uma resposta pela camada de saída. Esta fase é chamada de fase de propagação para frente (*forward*). Na segunda fase, a saída obtida é comparada à saída desejada para aquele padrão. A comparação é feita através do cálculo do erro. O erro é propagado da camada de saída para a camada de entrada, e os pesos das conexões vão sendo modificados conforme o erro é propagado. Esta fase é chamada de fase de propagação para trás (*backward*) (CARVALHO, 1999).

O algoritmo *back propagation* utiliza a informação do gradiente descendente do erro para modificar os valores dos pesos das conexões. Dessa forma, as unidades de processamento (UP) devem possuir funções de ativação contínuas e diferenciáveis, como por exemplo, a função sigmoideal (BEALE e JACKSON, 1990).

A taxa de aprendizagem, η , é uma constante de proporcionalidade no intervalo $[0,1]$, utilizada no mecanismo de ajuste dos pesos. Ela mede o “passo” do gradiente na superfície de erro. Quanto maior essa taxa, maior

será a mudança nos pesos, aumentando a velocidade de aprendizagem. Por outro lado, valores altos de η podem levar à oscilação do modelo na superfície de erro (CARVALHO, 1999).

Para melhor compreensão do algoritmo, é apresentada na figura 3.5 a seguir, uma rede neural com três camadas de unidades de processamento (UP), totalmente conectadas.



Fonte: FERREIRA (1998)

FIGURA 3.5 – Rede *back propagation* com uma camada interna

Na figura 3.5, X_i , Z_j e Y_k representam respectivamente, as ativações das UP de entrada, internas e de saída. V_{ij} representam os pesos das conexões entre a camada de entrada e a camada interna, e W_{jk} os pesos das conexões entre a camada interna e a de saída. O limitante sobre uma UP de saída é denotado W_{ok} , e o limitante sobre uma UP interna é denotado V_{oj} (FERREIRA, 1998). Como comentado na seção 3.4, os limitantes atuam como pesos sobre as conexões das UP cuja entrada é sempre 1 (*bias*).

A seguir é apresentado o algoritmo de treinamento *back propagation*.

Passo 0. Inicie os pesos, onde W_{jk} e $V_{ij} \in \mathbf{R}$ (estabeleça valores aleatórios pequenos).

Inicie os limitantes W_{ok} e $V_{oj} \in \mathbf{R}$.

Inicie a taxa de aprendizagem η , onde $0 < \eta \leq 1$.

Determine a condição de parada. Esta condição pode ser dada por um valor de erro desejado, ou pelo número de ciclos executados (um ciclo compreende os passos 1-10).

Passo 1. Enquanto a condição de parada for falsa, faça os passos 2-10.

Passo 2. Para cada padrão de treinamento, faça os passos 3-9.

Propagação para frente:

Passo 3. Cada unidade de entrada recebe um sinal de entrada x_i e propaga esse sinal para todas as unidades na camada interna.

Passo 4. Cada unidade interna ($Z_j, j=1, \dots, p$) calcula sua ativação,

$$z_{-s_j} = v_{0j} + \sum_{i=1}^n x_i v_{ij},$$

aplica a função de ativação para calcular o sinal de saída,

$$z_j = f(z_{-s_j}),$$

e envia esse sinal a todas as unidades na camada de saída.

Passo 5. Cada unidade de saída ($Y_k, k=1, \dots, m$) calcula sua ativação,

$$y_{-s_k} = w_{ok} + \sum_{j=1}^p z_j w_{jk}$$

e aplica a função de ativação para calcular seu sinal de saída,

$$y_k = f(y_{-s_k}).$$

Passo 6. Cada unidade de saída ($Y_k, k=1, \dots, m$) calcula seu erro (δ_k) em relação à saída desejada (t_k),

$$\delta_k = (t_k - y_k) f'(y_{-s_k}),$$

Se a condição de parada foi estabelecida em relação ao erro, então, teste a condição de parada.

Propagação para trás do erro:

Passo 7. Cada unidade de saída ($Y_k, k=1, \dots, m$) calcula seu termo de correção do peso (usado para atualizar w_{jk} mais tarde),

$$\Delta w_{jk} = \eta \delta_k z_j,$$

calcula seu termo de correção do limitante (para atualizar w_{ok} mais tarde),

$$\Delta w_{ok} = \eta \delta_k,$$

e envia δ_k para as unidades na camada interna.

Fonte: FAUSETT apud FERREIRA (1998)

continua

FIGURA 3.6 – Algoritmo de treinamento *back propagation*

continuação

Passo 8. Cada unidade interna ($Z_j, j=1, \dots, p$) calcula seu erro,

$$\delta_j = \sum_{k=1}^m \delta_k w_{jk} * f'(z_{sj}),$$

calcula seu termo de correção do peso (usado para atualizar v_{ij} mais tarde),

$$\Delta v_{ij} = \eta \delta_j x_i, \quad e$$

calcula seu termo de correção do limitante (para atualizar v_{0j} mais tarde),

$$\Delta v_{0j} = \eta \delta_j.$$

Atualiza pesos e limitantes

Passo 9. Cada unidade de saída ($Y_k, k=1, \dots, m$) atualiza seus limitantes e pesos ($j=0, \dots, p$):

$$w_{jk}(\text{new}) = w_{jk}(\text{old}) + \Delta w_{jk}.$$

Cada unidade interna ($Z_j, j=1, \dots, p$) atualiza seus limitantes e pesos ($i=0, \dots, n$):

$$v_{ij}(\text{new}) = v_{ij}(\text{old}) + \Delta v_{ij}.$$

Passo 10. Se a condição de parada foi estabelecida em relação ao número de ciclos, então teste a condição de parada.

Fonte: FAUSETT apud FERREIRA (1998)

FIGURA 3.6 – Algoritmo de treinamento *back propagation*

3.9 Ativação

Após o treinamento da rede neural, ela poderá ser utilizada como ferramenta para a classificação de novos dados. Para isto, a rede deverá ser utilizada apenas no modo *forward* (fase de propagação para frente do algoritmo de treinamento). A figura 3.7 apresenta o procedimento da ativação.

Passo 0. Carregue os pesos resultantes da fase de treinamento.

Passo 1. Para cada padrão de entrada, faça os Passos 2-4.

Passo 2. Para $i=1, \dots, n$: estabeleça a ativação da unidade de entrada x_i ;

Passo 3. Para $j=1, \dots, p$:

$$z_{-s_j} = v_{0j} + \sum_{i=1}^n x_i v_{ij};$$

$$z_j = f(z_{-in_j}).$$

Passo 4. Para $k=1, \dots, m$:

$$y_{-s_k} = w_{0k} + \sum_{j=1}^p z_j w_{jk};$$

$$y_k = f(y_{-in_k}).$$

Fonte: FAUSETT apud FERREIRA (1998)

FIGURA 3.7 – Ativação

3.10 Softwares de Redes Neurais Artificiais

Para o desenvolvimento do modelo de RNA, foram pesquisados alguns *softwares* disponíveis no mercado. A tabela 3.1 apresenta características desses *softwares*. A análise mais detalhada dos mesmos encontra-se no apêndice A. Para desenvolver uma aplicação com RNA não é necessário desenvolver um programa computacional, nem esse é o objetivo deste trabalho. Esses *softwares* possuem os algoritmos já implementados.

TABELA 3.1 - Características de alguns *softwares* de RNA

| Características | Softwares | | | |
|-------------------------|--|-------------------------------------|--|--------------------------------|
| | NeuralWorks Professional II/Plus | NeuralSIM | NeuroShell2 | SNNS |
| Fabricante | NeuralWare, Inc | NeuralWare, Inc | Ward Systems Group, Inc | University of Stuttgart |
| Propósito | Geral | Geral | Geral | Geral |
| Principais arquiteturas | MLP, BAM, Hopfield, ART, Kohonen (28 tipos) | MLP | MLP, Kohonen Probabilística (16 tipos) | MLP, Kohonen ART (17 tipos) |
| Avaliação do desempenho | gráficos de erro histogramas | gráficos de erro histogramas | gráfico de erro, de correlação | gráfico de erro, de pesos |
| Plataformas | PC, SUN, RS6000, SGI | PC | PC | PC, SUN, RS6000, HP, SGI |
| Entrada de dados | teclado, arquivo ASCII, binário, linguagem C | teclado via Excel, ASCII | teclado, arquivo ASCII, binário, planilha | teclado, arquivo ASCII |
| Saída de dados | tela ou arquivos | tela ou arquivos | tela, ASCII, binário, planilha | tela, arquivo ASCII |
| Preço | US\$ 2.995 a US\$ 4.995 | US\$ 1.995 | US\$ 595 a US\$ 1.395 | Freeware |
| Home-page | www.neuralware.com | | www.wardsystems.com | ver apêndice A |

O *software* NeuralWorks Professional II/Plus é indicado para aplicações de crédito, previsão de falências, seguros, controle de processo, inspeção industrial, marketing, diagnóstico de máquinas. Implementa 28 arquiteturas de redes neurais. Pode ser utilizado com sistemas especialistas e lógica nebulosa, e suporta várias plataformas.

O *software* NeuralSIM implementa somente redes MLP. Soluciona problemas de predição e classificação. Já foi utilizado para inspeção industrial, diagnóstico médico, previsões financeiras, qualidade de processos químicos, seguros. Está integrado ao Microsoft Excel. Possui dois algoritmos

de treinamento, um para dados limpos e outro para dados ruidosos. Converte dados em C, Fortran, e Visual Basic.

O *software* NeuroShell2 é indicado para estudos acadêmicos. Implementa 16 algoritmos de rede neurais. Permite a escolha de várias funções de ativação, conversão de dados alfanuméricos e *strings* em números, permite o pré-processamento dos dados de entrada, prevenção contra retreinamento. Possui um módulo para iniciantes e outro avançado.

O *software* SNNS – Stuttgart Neural Networks Simulator, foi desenvolvido em 1989 pela Universidade de Stuttgart, Alemanha. Seu objetivo foi criar um ambiente eficiente e flexível para pesquisa e aplicações de RNA. É um software de propósito geral e permite a implementação de 17 arquiteturas de RNA e variações. Possibilita a utilização de mais de 20 funções de ativação além de funções definidas pelo usuário. Por ser também uma ferramenta para pesquisa em RNA, o *software* SNNS vem sendo bastante utilizado em aplicações acadêmicas em áreas como reconhecimento de caracteres, classificação de padrões, robótica, e outras.

Capítulo 4

Redes Neurais e a Engenharia de Produção

4.1 Aplicações de Redes Neurais Artificiais na Manufatura

Os trabalhos pioneiros de Hopfield no início da década de 80, nos quais foi utilizado como exemplo o problema do caixeiro viajante, demonstraram a capacidade de redes neurais em fornecer soluções rápidas e de qualidade para problemas complexos, e precipitaram uma série de aplicações de redes neurais na manufatura (UDO e GUPTA, 1994).

ZHANG e HUANG (1995) abordam o estado da arte das aplicações de redes neurais na manufatura. Citam como principais desenvolvimentos aqueles ocorridos na aplicação de tecnologia de grupo, na engenharia de projeto, para monitorização e diagnose, no planejamento e controle do processo, no controle de qualidade e na programação da produção.

A tecnologia de grupo explora as similaridades do projeto do produto e do processo através da formação de famílias de peças visando a economia de escala. Um dos problemas na sua implementação é a classificação e a formação dessas famílias. KAO e MOON apud ZHANG e HUANG (1995) desenvolveram uma rede com algoritmo *back propagation* utilizando a capacidade de redes neurais em aprender critérios consistentes para a formação das famílias de peças e automatizar o processo.

CHRYSSOLOURIS et al. (1990) propuseram a utilização combinada de redes neurais com um simulador para o dimensionamento de um sistema de manufatura. A rede aprendia a função inversa da simulação, ou seja, a partir de um desempenho desejado dado pelo usuário, ela fornecia a quantidade de recursos. Assim pode-se determinar o número de recursos apropriado para cada centro de trabalho num ambiente *job-shop*.

Uma rede neural com algoritmo *back propagation* foi utilizada por SMITH apud ZHANG e HUANG (1995) no controle de qualidade de moldes de injeção. Os moldes requerem vários materiais e diferentes condições de

máquina e ambiente, já que as condições da linha mudam durante a operação e afetam a qualidade do produto. Nesse problema, cujos dados não seguem uma distribuição ou padrão conhecidos, redes neurais são especialmente aplicáveis. Os resultados foram comparados com cartas de controle e outras técnicas estatísticas apresentando desempenho superior.

ANSUJ et al. apud PORTO (1995) desenvolveram um modelo de redes neurais para previsão do volume de vendas de uma empresa em Santa Maria, RS. Para treinamento e teste da rede foram utilizados dados de demanda entre 1979 e 1989. O algoritmo utilizado foi o *back propagation*. Os resultados da rede foram comparados aos obtidos através de um método matemático e apresentaram previsões mais exatas.

PORTO (1995) faz também uma análise das aplicações de redes neurais na área financeira. Foram encontrados trabalhos na área de previsão de falência e inadimplência de instituições, na previsão do comportamento do mercado de ações e outros investimentos financeiros, análise de risco para concessão de crédito, contratos de seguros e hipotecas. Essas aplicações utilizam a capacidade de previsão e classificação das redes neurais, principalmente nos modelos de aprendizagem supervisionada, já que o volume de dados numéricos disponíveis nessa área é enorme (índices econômicos, cotações).

A seguir são apresentadas algumas aplicações de redes neurais artificiais na programação da produção.

4.2 Programação da Produção Segundo Redes Neurais Artificiais

Segundo SABUNCUOGLU e GURGUN (1996), as aplicações de redes neurais na programação da produção utilizam principalmente as redes *back propagation* e de Hopfield. Este fato pode ser observado na revisão bibliográfica realizada, da qual serão apresentadas as contribuições mais significativas.

Como será observado adiante, a maioria dos artigos revisados abordam o problema de programação da produção em ambiente *job-shop*. Isso vem ao encontro do fato que este problema é np-completo (não determinístico em tempo polinomial). Esse tipo de problema exibe um crescimento exponencial em tempo computacional quando o tamanho do problema cresce linearmente. A abordagem por RNA é especialmente indicada para esse tipo de problema devido à capacidade de modelar problemas de grande porte que envolvem diversas restrições e variáveis.

4.2.1 Aplicações com o Algoritmo *Back Propagation*

KIM e LEE (1993) realizaram vários estudos sobre utilização de RNA para melhorar o desempenho de regras de prioridade (heurísticas) em máquina única e em máquinas paralelas. O desempenho das heurísticas que utilizavam a abordagem por RNA foi superior ao das heurísticas puras.

TOURE et al. (1993) analisaram a capacidade de aplicação de redes Hopfield, Kohonen e *Perceptron* de Múltiplas Camadas (MLP) em Sistemas Flexíveis de Manufatura - FMS. O objetivo era comparar o desempenho de seis regras de prioridade frente ao critério de minimização do atraso das tarefas. Os resultados gerados pela rede foram superiores aqueles obtidos pela aplicação direta das regras de prioridade.

WATANABE et al. (1993) propuseram uma melhoria numa rede neural *back propagation* para estimação da folga dinâmica de uma tarefa (tempo até a data de entrega menos o tempo de processamento remanescente), em ambiente *job-shop*. As medidas de desempenho relacionadas ao atraso médio e máximo foram coletadas na saída da rede. O índice de confiabilidade dessas medidas foi calculado através da adição de uma segunda rede neural em seqüência. Com esta segunda rede houve melhoria de até 14% no atraso máximo das tarefas.

A metodologia de que uma rede neural de múltiplas camadas - MLP - poderia ser usada como a função inversa de um sistema de simulação foi sugerida por CHRYSSOLOURIS et. al. (1990). A simulação foi utilizada para determinar medidas de desempenho, dada a configuração do sistema e certas estratégias de controle. Neste estudo a rede neural foi treinada para aprender a função inversa da simulação, ou seja, estimar parâmetros do sistema a partir das medidas de desempenho fornecidas. Os resultados mostraram que a RNA poderia ser utilizada como uma ferramenta para o projeto de um sistema de manufatura, e poderia reduzir os erros das rodadas de simulação.

LIU e DONG (1996) propuseram um algoritmo para programação da produção em ambiente *job-shop* que consiste de três estágios:

- simulação com o objetivo de avaliar a eficiência de regras de prioridade em diferentes situações de programação;
- utilização dos resultados da simulação para treinar a rede neural com o objetivo de adquirir conhecimento para selecionar a regra mais eficiente para cada situação de programação;
- utilização da rede treinada como seletor de regras de prioridade num sistema de programação em tempo real.

O sistema é composto por cinco máquinas pelas quais são processadas cinco tipos de peças. As peças chegam aleatoriamente em lotes. As peças de um lote possuem a mesma data de entrega. O tamanho do lote, o número

de operações por tipo de peça e o tempo de processamento variam segundo uma distribuição uniforme. As tarefas são designadas às máquinas segundo regras de prioridade. Para cada situação de programação são avaliadas as regras SPT, LPT (é selecionada da fila a peça com maior tempo de processamento), FCFS (a primeira peça a entrar no sistema é a primeira a sair), LCFC (a última peça a entrar no sistema é a primeira a sair), EDD, LDD (a peça com data de entrega mais longe é seqüenciada primeiro) e CR (razão crítica entre o tempo disponível e o tempo de processamento remanescente). As medidas de desempenho coletadas ao final da simulação são: tempo médio de permanência no sistema, utilização da máquina e espera média para cada máquina.

As medidas de desempenho obtidas pela melhor regra de prioridade em cada situação de programação foram utilizadas para treinar a rede neural. A rede utilizada foi do tipo *Perceptron* de Múltiplas Camadas (MLP) com algoritmo *back propagation*. Os resultados da RNA (regras de prioridade recomendadas) foram comparados aos das regras utilizadas no simulador. Esses resultados mostraram que em 70% dos casos as regras fornecidas pela rede foram mais efetivas que as do simulador. Esse trabalho apresentou o potencial de utilização de redes neurais artificiais como seletor de regras de prioridade.

CHEN e YIH (1996) propuseram a utilização de redes neurais para selecionar atributos essenciais de um FMS. Atributos são características usadas para representar um sistema. Entre os vinte atributos analisados estão: estoque em processo, utilização média das máquinas, atraso médio das tarefas. Primeiramente foi realizada a simulação com o objetivo de coletar dados contendo a relação entre as medidas de desempenho e os atributos sob várias regras de prioridade. Foram analisadas seis medidas de desempenho, entre elas: atraso médio, *makespan* (duração total da programação), tempo médio de permanência no sistema. Foram testadas nove regras de prioridade, entre elas: SPT, FIFO, EDD, SLACK.

O FMS era composto por quatro centros de usinagem, uma máquina de lavar, um guindaste e área para estocagem. Cada centro de usinagem possuía seu próprio *buffer* (local de espera) de entrada e de saída com capacidade de uma peça. Esse sistema era capaz de processar quarenta diferentes operações e cada operação podia ser feita em exatamente dois centros diferentes. Como o transporte das peças era feito pelo guindaste, os pontos de decisão eram definidos quando o guindaste estivesse livre.

Para cada regra foram feitas várias rodadas de simulação e ao final foram coletados os valores das medidas de desempenho e a variabilidade dos atributos. Foram desenvolvidas várias redes neurais (uma para cada regra) com o objetivo de mapear as relações entre as medidas de desempenho e os valores dos atributos. A rede utilizada foi a *Perceptron* de Múltiplas Camadas, com o algoritmo *back propagation* para o treinamento. Foram testadas várias topologias de rede e o desempenho de cada uma foi medido através do cálculo do erro. As topologias com menor erro foram escolhidas para a seleção dos atributos. Os resultados mostraram que a habilidade de generalização da rede neural depende de quais atributos são usados.

SUN e YIH (1996) propuseram um sistema de controle baseado em redes neurais para adaptar diferentes estratégias de programação em uma célula de manufatura. O sistema de controle é constituído por um módulo de ajuste e por um equipamento de controle para cada máquina da célula e para um robô. O módulo de ajuste determina a importância relativa de cada medida de desempenho em cada ponto de decisão. O equipamento de controle, implementado por uma rede neural *back propagation*, é alimentado pelo módulo de ajuste, e determina a regra de prioridade baseado no seu *status* e nos níveis de importância relativa.

A célula é composta por cinco máquinas e um robô para o transporte. Existe um *buffer* de entrada e de saída com capacidade limitada em cada máquina, e um *buffer* de entrada no sistema. Quando uma peça chega, espera no *buffer* de entrada do sistema e requisita o robô para o

transporte. O robô leva a peça para o seu destino. Não é permitida interrupção nesse transporte. Depois de visitar cada máquina de acordo com seu roteiro, a peça deixa a célula e as medidas de desempenho são coletadas, como: tempo de ciclo, tempo de espera médio, atraso médio e taxa de tarefas atrasadas. São investigadas regras de prioridade para as máquinas e para o robô. As regras associadas às máquinas selecionam uma peça do *buffer* de entrada, entre elas, SPT, EDD, FIFO, SLACK, STT. As regras associadas ao robô selecionam a próxima peça quando o robô se torna ocioso. Entre as regras do robô estão: SDIST (seleciona a tarefa com menor tempo de transporte), NEARQ (seleciona a tarefa que estiver na fila mais próxima).

As entradas de cada rede são a importância relativa das medidas de desempenho e características do estado atual como tamanho da fila, *setup*, tempo de folga. As saídas das redes são notas para cada regra de prioridade. Pelos resultados obtidos concluiu-se que não existe uma única regra que satisfaça simultaneamente todas as medidas de desempenho. Isso provou a capacidade de adaptação do sistema de controle em ambiente multicritério, e de resposta a mudanças de objetivos.

SAAD (1996) propôs um modelo híbrido de simulação de sistemas com redes neurais artificiais cujo objetivo era testar a capacidade do modelo em realizar a diferenciação entre um dado conjunto de regras de um sistema de produção intermitente, de modo a operar num nível de desempenho preestabelecido.

O sistema era formado por cinco máquinas onde eram processados três tipos de peças, A, B e C. As peças A eram consideradas lentas (possuíam maior tempo de processamento), as peças B médias e as C rápidas. Apesar de cada peça possuir um roteiro de fabricação próprio, a primeira operação sempre era realizada em M1 e a última em M5. Com o objetivo de caracterizar três níveis de carga (alta, média e baixa) variou-se a porcentagem de cada tipo de peça no conjunto de ordens a serem

simuladas. Cada simulação consistia da passagem de 90 peças pelo sistema, com chegada estática.

As regras analisadas foram: SPT, EDD, FIFO, SLACK, MFSE (seleciona da fila a peça com menor fila na próxima máquina). A cada passagem das peças pelo sistema foram coletadas três medidas de desempenho: *makespan*, estoque médio em processo e tempo total de atraso.

As regras foram atribuídas individualmente a cada máquina, de modo aleatório. Foram simuladas 1% de todas as combinações possíveis de regras às máquinas. As medidas de desempenho coletadas do simulador foram as entradas da rede neural. A RNA utilizada foi a *Perceptron* de Múltiplas Camadas, com algoritmo de treinamento *back propagation*. Foram analisadas várias topologias de rede com variação dos parâmetros como: número de camadas escondidas, taxa de aprendizagem, termo de momento. As saídas da rede (regras de prioridade recomendadas) foram validadas no simulador, ou seja, as regras recomendadas pela rede foram atribuídas às máquinas e as medidas de desempenho obtidas foram comparadas às desejadas. Um resultado satisfatório significa que a RNA prescreveu bem as regras de prioridade às máquinas.

4.2.2 Aplicações com a Rede Hopfield

A rede de Hopfield sofreu várias extensões e alterações à medida que novas pesquisas foram desenvolvidas. Uma das modificações mais significativas foi a proposta por FOO e TAKEFUJI (1988 I e II), e teve vários seguidores. Primeiramente serão apresentadas as aplicações que utilizam como base a rede de Hopfield e a seguir aquelas baseadas no modelo de Foo e Takefuji.

ARIZONO et al. (1992) desenvolveram uma rede neural estocástica baseada no modelo de Hopfield para solucionar o problema de

seqüenciamento em máquina única em ambiente *just-in-time*, com o objetivo de minimizar tempo total de fluxo. O modelo de Hopfield é determinístico, e por isso tende para um mínimo local de energia dependendo do estado inicial da rede. Visando escapar desses mínimos locais é fornecida aleatoriedade ao processo através da adição de ruído no sistema. Para solucionar o problema de otimização, a função objetivo e as restrições são descritas na forma de uma função de energia, que deve ser minimizada. As restrições asseguram as relações de precedência entre as operações de uma mesma tarefa, e evitam o conflito de operações em uma mesma máquina. O resultado gerado pela rede foi comparado com o valor ótimo obtido por método de enumeração completa. O modelo nem sempre chegou à programação ótima, mas forneceu soluções com média superior a 90% da ótima para cinquenta tipos de problemas testados.

SATAKE et al. (1994) desenvolveram uma abordagem baseada no modelo de Hopfield com o objetivo de minimizar o *makespan* da programação da produção em ambiente *job-shop*. A diferença do modelo de Hopfield está no valor do limitante de cada neurônio, que não é pré-determinado, mas revisado a cada atualização dos neurônios. A RNA empregada é a Máquina de Boltzmann. A função energia é composta por apenas uma restrição, a que cada operação deve ser iniciada somente uma vez, e o restante das restrições e a função objetivo estão refletidas nos valores dos limitantes. O estado do neurônio depende de seu estado atual e do valor do limitante, que é mudado a cada iteração. Portanto, as mudanças nos limitantes devem satisfazer as restrições de precedência entre operações de uma mesma tarefa, evitar o conflito de operações na mesma máquina e minimizar o *makespan*. A simulação foi feita sobre quinze problemas de *job-shop* e os resultados foram comparados aos obtidos pelo método *branch and bound*. A rede produziu soluções ótimas para dez tipos de problemas e próximas da ótima para cinco tipos, confirmando sua efetividade.

WILLEMS e BRANDTS (1995) propuseram uma arquitetura de RNA, semelhante à de Hopfield, em que o critério de otimização é implementado na própria arquitetura. O objetivo é a minimização do *makespan* para o problema de programação da produção em ambiente *job-shop*. Foram empregadas seis regras de prioridade como critério de otimização local. Entre elas, SPT, EDD, MWKR (seleciona a operação com maior quantidade de trabalho remanescente). O problema é representado por formulação linear inteira. A rede neural é formada por três camadas. Na primeira os neurônios representam os tempos de início das operações (função objetivo). Na segunda, as restrições de precedência e as restrições de recursos. Na terceira, as unidades binárias da representação inteira. As regras de prioridade são implementadas através de realimentações existentes entre as unidades da rede, que podem alterar as datas de início das unidades da primeira camada. Foi desenvolvida uma rede para cada regra. Os autores apresentaram um exemplo com cinco tipos de produtos e quatro máquinas. Os experimentos mostraram que a rede com a regra MWKR forneceu o melhor resultado em relação ao *makespan*.

SABUNCUOGLU e GURGUN (1996) desenvolveram um modelo de RNA aplicado ao problema de seqüenciamento em máquina única com o objetivo de minimizar o atraso médio, e em ambiente *job-shop* com o objetivo de minimizar o *makespan*. As restrições de viabilidade estão representadas tanto na função de energia como em um processador externo. Os neurônios, que representam as tarefas no problema de máquina única e as operações no problema de *job-shop*, competem entre si para pegar a primeira posição disponível na seqüência. A seqüência das tarefas numa máquina é representada de forma matricial. A linha refere-se a uma tarefa e a coluna à posição dessa tarefa na seqüência, formando uma matriz $n \times n$. O processador realiza trocas aleatórias de duas linhas, correspondendo a posições de duas tarefas na seqüência, e avalia o valor da função de energia da rede após a troca, visando minimizá-la. Para máquina única foram testados problemas com 50 e 100 tarefas. Houve melhoria de até

4,46% no atraso médio. Para o *job-shop* foram testados 25 problemas dos quais a rede encontrou a solução ótima para 18 deles.

FOO e TAKEFUJI (1988 I e II) propuseram uma RNA estocástica inspirada no modelo de Hopfield, com o objetivo de minimizar o tempo de término das tarefas, para a programação da produção em *job-shop*. A arquitetura da rede é análoga a um circuito elétrico, com resistências e capacitores. O problema de *job-shop* é formulado via representação linear inteira com o objetivo de minimizar o tempo de início de todas as tarefas (função custo), sujeitas a um conjunto de restrições de precedência e de recursos. Entradas externas (excitações e inibições) representando as restrições do problema são introduzidas nos neurônios da rede. As saídas, juntamente com informações dos tempos de processamento das operações são introduzidas num circuito que calcula a função custo e determina o tempo de início de cada tarefa. A diferença entre o custo atual e o anterior é adicionada à função de energia. A finalidade dessa realimentação é promover a busca de pontos aleatórios na função de energia que venham alterar o estado do neurônio e, se possível, minimizar a energia. Na convergência da rede a função custo é minimizada.

ZHOU et al. (1991) propuseram uma rede com formulação linear inteira semelhante à de FOO e TAKEFUJI (1988 I e II) com o objetivo de minimizar o tempo de término das tarefas em ambiente *job-shop*. Aqui a função custo e as restrições são colocadas na função de energia, que deve ser minimizada. A importância desse modelo deve-se à utilização de uma função custo linear em que o número de conexões cresce linearmente com o tamanho do problema. No modelo de Foo e Takefuji, como a função de energia possuía um termo quadrático, o número de conexões crescia exponencialmente com o tamanho do problema. Para o problema 4/3/G/M houve redução de 92% no número de neurônios e de 99,8% no número de conexões em relação ao modelo do caixeiro viajante de Hopfield. Em relação ao modelo de Foo e Takefuji, houve redução de 84% no número de

neurônios e de 98% no número de conexões, mostrando viabilidade para implementação em *hardware*.

YEH et al. (1995) desenvolveram uma abordagem que modifica o algoritmo de FOO e TAKEFUJI (1988 I e II) para solucionar o problema de programação da produção em ambiente *job-shop*. A diferença está em três pontos: arranjo dos neurônios, forma da função de energia e forma da matriz de entradas externas. Na rede de Hopfield para o problema do caixeiro viajante, a posição de uma operação de uma tarefa está representada pelo índice na seqüência. Se o valor da saída de um neurônio V_{xi} é um ($V_{xi}=1$), significa que a tarefa x está designada à posição i . No modelo de Foo e Takefuji, a posição de uma operação de uma tarefa está representada pela interdependência das tarefas, isto é, a tarefa x está posicionada depois da tarefa i e nenhuma outra tarefa pode ser posicionada entre x e i quando $V_{xi}=1$. O modelo proposto por YEH et al. (1995) acrescenta à definição de V_{xi} de Foo e Takefuji o fato que uma e somente uma tarefa possa seguir outra na seqüência. A RNA apresentou porcentagem de soluções ótimas e a porcentagem do erro em relação à solução ótima melhores que um método heurístico e programação linear inteira (YEH et al., 1995).

4.3 Análise sobre os Modelos Apresentados

No editorial de uma edição especial da revista *European Journal of Operational Research* sobre trabalhos que utilizavam redes neurais para solucionar problemas de pesquisa operacional (PO), SHARDA e WANG (1996) consideram que redes neurais oferecem três oportunidades do ponto de vista da Pesquisa Operacional / Ciência da Gestão: são alternativas aos algoritmos e heurísticas convencionais da PO para problemas de otimização, são técnicas de mapeamento estatístico biologicamente inspiradas, e é um campo no qual podem ser aplicados algoritmos da PO para melhorar o desempenho da rede.

Apesar das redes neurais do tipo Hopfield e das baseadas no algoritmo *back propagation* serem as mais utilizadas em aplicações na programação da produção, a forma de modelagem do problema em cada uma delas é completamente diferente.

Na rede de Hopfield a modelagem torna-se bastante complexa, já que os parâmetros do modelo como tempo de início das operações, seqüência em que são realizadas, são representadas na estrutura interna da rede, principalmente através de uma representação linear inteira com restrições e função objetivo. As aplicações com a rede de Hopfield apresentadas traduzem essa complexidade e tratam de programação da produção em sistemas produtivos mais simples como máquina única e *job-shop* básico, que possui várias considerações e simplificações.

Nas aplicações com o algoritmo *back propagation*, o sistema produtivo se aproxima mais de um sistema real com considerações sobre filas, tempo de *setup*, lotes de produção, estoque em processo, datas de entrega e outras. Na maioria das aplicações, como as de LIU e DONG (1996), SAAD (1996), CHEN e YIH (1996) e SUN e YIH (1996), o ambiente de produção é modelado através de um sistema de simulação. As saídas da simulação são entradas para a rede neural. Além da técnica de simulação de sistemas permitir a inclusão de diversos aspectos presentes em um ambiente real, retira a carga da modelagem do sistema produtivo ser representado na estrutura interna da rede.

A minimização da função de energia no modelo de Hopfield e em suas extensões é equivalente à minimização da função de erro no algoritmo *back propagation*. Segundo BEALE e JACKSON (1990), nas redes *back propagation* também existe o perigo da convergência para um mínimo local indesejado, dificultando o processo de aprendizagem. Isso pode ser evitado através de algumas alternativas de manipulação dos parâmetros da rede como: variação da taxa de aprendizagem, η , presente no mecanismo de ajuste dos pesos e adição de neurônios às camadas escondidas. É

interessante que no início da aprendizagem o valor de η seja alto e decresça progressivamente. Isso faz com que os pesos sejam alterados mais rapidamente nos estágios iniciais, aumentando a probabilidade de não se atingir mínimos locais nessa fase, e mais suavemente à medida que o tempo passa. A adição de neurônios às camadas escondidas, cria mais limites para possíveis soluções através da melhor codificação dos dados de entrada. Como abordam CORSTEN e MAY (1996), com pequeno número de neurônios na camada escondida, a rede não consegue construir uma representação interna para o problema. Entretanto, deve-se tomar cuidado com o fenômeno da sobreposição, ou seja, a memorização dos dados de entrada prejudicando a generalização dos resultados. Portanto, existe um ponto ótimo na determinação do número de neurônios na camada escondida, que varia de acordo com o problema.

A adição de um termo de momento, α , na equação de atualização dos pesos no algoritmo *back propagation* é outra forma de se evitar os mínimos locais. Este termo permite acelerar o treinamento da rede. A determinação dos melhores valores para η e α depende do problema e da experiência de quem conduz o treinamento. Segundo STEPHANOPOULOUS e HAN (1996), embora esta seja uma técnica empírica, pesquisas relevantes têm sido desenvolvidas para sistematizá-la.

A seguir são apresentadas duas tabelas contendo alguns aspectos sobre programação da produção e sobre RNA presentes nas aplicações mais significativas revisadas, ou seja, as que utilizam a rede *Perceptron* de Múltiplas Camadas - MLP - com algoritmo *back propagation*.

TABELA 4.1 – Aspectos sobre programação da produção

| Aspectos | Aplicações | | | |
|------------------------|---|--|---|--|
| | SAAD | LIU e DONG | SUN e YIH | CHEN e YIH |
| Ambiente | Job-shop | Job-shop | Célula | FMS |
| No. Máquinas | 5 | 5 | 5 máquinas 1 robô | 4 máquinas 1 guindaste |
| Tipos de peça | 3 | 5 | 15 | _____ |
| No. operações por peça | 5 | U[1, 5] | _____ | 2 a 10 |
| Chegada | determinística | aleatória em lotes | _____ | _____ |
| Tempo de Processamento | determinístico (1 a 12 min) | determinístico U[20,50] un.tmp | _____ | determinístico (5 a 20 min) |
| Setup | não considera | não considera | considera | _____ |
| Capacidade das Filas | ilimitada | ilimitada | Limitada | Limitada (1 peça) |
| Data de Entrega | Sorteio aleat. entre 100 e 600 | _____ | _____ | _____ |
| Regras | 1 por máquina | 1 p/ o sistema | 1 p/ o sistema | 1 p/ o sistema |
| Medidas de Desempenho | Makespan No. méd. pç. sist Atraso total | Tmp méd. fluxo No. méd. pç. fila Utiliz. da máq. | Tempo de ciclo Espera média % tarefas atrasadas | WIP Tmp méd fluxo Utiliz. das máqs |

Existe muita semelhança entre os modelos de *job-shop* da tabela 4.1, como o número de máquinas, tempo de processamento determinístico, capacidade ilimitada das filas, *setup* incluído no tempo de processamento. A principal diferença entre eles é a forma de atribuição das regras de prioridade. No modelo de SAAD (1996) as regras são atribuídas individualmente às máquinas, ou seja, existe uma regra para cada máquina. Para o modelo de LIU e DONG (1996) e demais é considerada uma regra para todo o sistema. São realizadas várias simulações, cada uma com uma regra. Este aspecto do modelo irá influenciar na forma de representação das unidades da camada de saída da rede neural. No primeiro caso as unidades de saída irão representar as regras de prioridade e no segundo caso representarão as regras associadas às máquinas.

Como abordado no capítulo 2 e mostrado na tabela 4.1, as medidas de desempenho avaliadas estão associadas à congestão do sistema e ao cumprimento aos prazos de entrega.

A tabela 4.2 a seguir apresenta alguns aspectos sobre redes neurais artificiais daqueles mesmos trabalhos.

TABELA 4.2 – Aspectos sobre redes neurais artificiais

| Aspectos | Aplicações | | | |
|---------------------------------------|------------|----------------------------------|------------------------------|-------------------------|
| | SAAD | LIU e DONG | SUN e YIH | CHEN e YIH |
| Entradas | ID | No. da máquina Tnp processam. | Atributos do sistema e ID | Atributos do sistema |
| Saídas | Regras | Regras | Regras | ID |
| Função de Ativação | sigmoidal | sigmoidal | _____ | sigmoidal |
| No. neurônios na camada de entrada | 3 | 50 | _____ | 20 |
| No. neurônios na camada de saída | 5 | 7 | 8 | 6 |
| Conj. Treinamento (dados entrada) | contínuos | discretos; contínuos | contínuos | contínuos |

As entradas e saídas da rede refletem a forma de modelagem do sistema. O número de neurônios nas camadas de entrada e de saída está associado à configuração do sistema, como o número de medidas de desempenho e de regras analisadas. A função de ativação utilizada nos quatro trabalhos citados na tabela 4.2 é a sigmoidal, comprovando a afirmação apresentada no capítulo 3 de que essa função é uma das mais utilizadas.

Outros parâmetros de uma rede neural do tipo *Perceptron* de Múltiplas Camadas - MLP - como número de camadas escondidas, número de neurônios nessas camadas, taxa de aprendizagem, termo de momento, intervalo de iniciação dos pesos, parâmetro de época, são avaliados ao realizar o treinamento da rede. O desempenho das topologias é analisado e as redes com as melhores configurações são as utilizadas para a modelagem do problema. Os parâmetros de uma rede MLP serão detalhados no capítulo 5 deste trabalho.

Questões não tratadas nos trabalhos *supra* citados como forma de apresentação dos padrões de treinamento, modo de atualização dos pesos, critério de parada do treinamento, devem ser pensadas *a priori* durante a modelagem da rede neural.

Segundo SHARDA e WANG (1996), as pesquisas sobre redes neurais artificiais abriram uma nova dimensão para a pesquisa científica e aplicações industriais e de negócios. Devido ao seu potencial de utilização em diferentes áreas do conhecimento como engenharia, neurologia, psicologia, matemática, computação, e outras, RNA têm contribuído na solução de numerosos problemas em diversas aplicações.

Neste capítulo foram apresentadas algumas aplicações de redes neurais artificiais na programação da produção. Essas aplicações utilizaram as redes *Perceptron* de Múltiplas Camadas - MLP - com algoritmo *back propagation* e redes baseadas no modelo de Hopfield, que são as mais empregadas na solução de problemas de programação da produção e otimização.

A seguir serão apresentadas as características do ambiente de produção e o procedimento de aplicação de redes neurais artificiais neste ambiente.

Capítulo 5

O Modelo Proposto

5.1 Introdução

Como abordado anteriormente, o objetivo deste trabalho é analisar capacidade de um modelo híbrido de simulação de sistemas e de redes neurais artificiais em realizar a diferenciação entre um conjunto de combinações de regras de prioridade designadas às máquinas no ambiente de produção, através da avaliação de quatro medidas de desempenho.

Este capítulo está estruturado da seguinte forma: na seção 5.2 será caracterizado de forma genérica o ambiente da programação da produção em relação a alguns parâmetros; na seção 5.3 será apresentado o problema de programação da produção específico a ser considerado neste trabalho, através da particularização daqueles parâmetros; a seção 5.4 trará o procedimento de utilização combinada da técnica de simulação de sistemas com redes neurais artificiais; na seção 5.5 será apresentada a dinâmica da simulação e na seção 5.6 serão apresentadas as características e configurações da rede neural artificial utilizada.

5.2 O Ambiente da Programação da Produção

Um ambiente de programação da produção pode ser caracterizado em relação a alguns parâmetros presentes em qualquer sistema, independente da sua configuração. Esses parâmetros são apresentados a seguir.

a) Fator Limitante do Sistema

Segundo GARGEYA e DEANE (1996), um sistema produtivo pode ser restringido pelas máquinas, mão-de-obra, recursos auxiliares como

ferramentas, ou uma combinação dos anteriores. Essas restrições regulam a saída do sistema e influenciam no desempenho da fábrica.

b) Número de Máquinas

Como apresentado no capítulo 2, a programação da produção pode ser desenvolvida em uma única máquina, em máquinas paralelas ou em sistemas com uma máquina de cada tipo. Na programação da produção em máquina única somente uma máquina está disponível ou então existe uma máquina que representa o gargalo do sistema. Na programação da produção em máquinas paralelas existem várias máquinas do mesmo tipo na fábrica. Essa situação pode ocorrer quando uma sessão da fábrica é dedicada a uma função específica, por exemplo, uma área para torneamento manual. Entretanto, em sistemas nos quais é produzida uma peça completa na saída, geralmente existe uma máquina de cada tipo, cada uma processando diferentes operações. Este é o caso de uma célula de manufatura. Segundo FERNANDES (1991), o número de máquinas em uma célula varia muito, em geral de 1 a 25 máquinas, entretanto quanto menor for o grupo de máquinas, melhor será o controle.

c) Operador

Devido à automatização dos processos produtivos, a quantidade de operadores em uma célula de manufatura passa a ser bem reduzida. Os operadores devem ser polivalentes ou multifuncionais capazes de operar todas ou a maioria das máquinas.

d) Tipos de Peças

Na tecnologia de grupo, cujo *layout* é a célula de manufatura, as peças com similaridades de projeto e processo são agrupadas e processadas pelo mesmo conjunto de máquinas, constituindo uma família de peças. Portanto, o número de tipos de peça numa célula irá depender das similaridades entre as peças.

e) Processo de Chegada

De acordo com a tabela 2.1, o processo de chegada das tarefas num sistema produtivo com padrão de fluxo *job-shop* pode ser estático, dinâmico determinístico ou dinâmico probabilístico. No processo estático todas as tarefas chegam juntas no instante zero. No processo dinâmico determinístico as tarefas chegam no sistema em tempos futuros conhecidos. No processo dinâmico probabilístico a chegada das tarefas ocorre segundo uma distribuição de probabilidade, por exemplo, Poisson, em que os tempos entre chegadas são exponencialmente distribuídos. Segundo HAX e CANDEA (1984), a hipótese típica em um sistema com padrão de fluxo *job-shop* é a chegada dinâmica probabilística.

f) Tamanho do Lote

Como abordado no capítulo 2, em um sistema intermitente de produção em lotes, o tamanho do lote é variável, e são produzidos em intervalos regulares. Entretanto, em sistemas reais, com o objetivo de diversificar os tipos de produtos existe uma tendência em reduzir tal tamanho.

g) Datas de Entrega

A data de entrega refere-se ao prazo máximo para uma peça sair de um sistema sem que seja considerada atrasada. As datas de entrega, em ambientes reais, são em geral estabelecidas a partir do pedido realizado pelo cliente e também considerando o tempo envolvido para a fabricação da peça completa.

h) Tempo de Processamento

Como apresentado na tabela 2.1, a natureza dos tempos de processamento pode ser determinística ou probabilística. No primeiro caso os tempos de processamento são conhecidos quando a tarefa chega na fábrica. No segundo caso isso não ocorre, ou seja, os tempos de

processamento são descritos de acordo com uma distribuição de probabilidade.

i) Roteiro de Fabricação

O roteiro de fabricação das peças em um sistema produtivo está associado ao padrão de fluxo presente neste sistema. Como tratado no capítulo 2, em sistemas com padrão de fluxo *flow-shop*, todas as peças possuem o mesmo roteiro de fabricação através das máquinas. Em sistemas com padrão de fluxo *job-shop* cada peça possui seu próprio roteiro de fabricação através das máquinas.

j) *Setup*

Vários modelos de programação da produção, para efeito de simplificação, consideram o tempo de *setup* incluído no tempo de processamento. Entretanto, o *leadtime* de produção de um item ou produto está diretamente relacionado com o tempo de *setup*. Portanto, este é um parâmetro que deve ser considerado em sistemas que procuram uma representação próxima da realidade.

k) Capacidade das Filas

Num ambiente de produção real, a capacidade máxima de peças em filas está associada ao meio de transporte utilizado para a movimentação das peças. Quando o modo de transporte das peças é manual, ou seja, realizado por um operador, a capacidade das filas é em geral restringida pelo espaço disponível entre as máquinas. Quando as peças são transportadas automaticamente de uma máquina para outra, ou seja, quando existem calhas ou esteiras, a capacidade das filas é determinada pelo comprimento do meio de transporte.

l) Medidas de Desempenho

Segundo CONWAY et al. (1967), as medidas de desempenho geralmente utilizadas para avaliar a programação da produção estão

associadas ao congestionamento do sistema e ao cumprimento aos prazos de entrega. O congestionamento do sistema pode ser avaliado pela quantidade de estoque em processo, pelo tempo que as peças gastam no sistema, entre outras. O cumprimento aos prazos de entrega pode ser medido pelo número ou taxa de tarefas atrasadas, pelo tempo de atraso médio ou total das tarefas. Entretanto, as medidas de desempenho devem estar relacionadas com a configuração do sistema de produção. Por exemplo, em um ambiente em que a capacidade da fila é limitada em uma peça, não faz sentido ter o número médio de peças em filas como uma medida de desempenho.

m) Regras de Prioridade

Vários autores como CONWAY (1965), BAKER e DZIELINSKI apud HAX e CANDEA (1984), NANOT apud BUFFA e MILLER (1979), realizaram estudos extensivos comparando um grande número de regras de prioridade em diversos sistemas produtivos, e comprovaram que diferentes regras são recomendadas conforme as medidas de desempenho que se pretendem otimizar.

Como abordado no capítulo 2, as regras de prioridade podem ser classificadas em locais ou globais, ou em estáticas ou dinâmicas. Para as regras locais, a prioridade de uma operação é determinada por parâmetros das peças que se encontram na fila da máquina que realizará aquela operação. Para regras globais a prioridade é calculada considerando também as filas das outras máquinas. Para as regras estáticas os índices de prioridade não mudam com o tempo. Já para as regras dinâmicas os índices de prioridade mudam em função do tempo.

No estudo da programação da produção existem dois enfoques na alocação das regras de prioridade. A escolha de um dos dois enfoques, descritos a seguir, depende do objetivo do problema analisado.

No primeiro enfoque é atribuída uma regra para todo o sistema. São realizadas simulações, e em cada uma delas é testada uma regra de prioridade para todas as máquinas. O objetivo é determinar a melhor regra

de prioridade para o sistema, de modo a minimizar algumas medidas de desempenho. Uma vantagem deste enfoque é a facilidade de implementação das regras de prioridade. Por outro lado, trabalha como se o fluxo de produção estivesse balanceado. Não considera, por exemplo, as diferenças entre uma máquina ociosa e uma máquina gargalo.

No segundo enfoque as regras de prioridade são atribuídas às máquinas, ou seja, é atribuída uma regra de prioridade para cada máquina. O objetivo é realizar a diferenciação entre as regras. Este enfoque, por atribuir as regras diretamente às máquinas, possui a vantagem de permitir a análise de recursos gargalos e ociosos. Por outro lado, existem múltiplas combinações de alocação de regras às máquinas, dificultando a implementação das mesmas.

5.3 O Problema Específico

O ambiente de produção em estudo é um sistema intermitente de produção em lotes, formado por um grupo de máquinas representando uma célula de manufatura. Como abordado no capítulo 2, as máquinas em uma célula são agrupadas de forma a permitir que peças com similaridades de projeto e de processo sejam fabricadas e completadas por esse grupo de máquinas.

Esse ambiente de produção será caracterizado a seguir através da particularização dos parâmetros descritos na seção 5.2 deste capítulo. Os valores dos parâmetros foram escolhidos de modo a representar características universais de uma célula de manufatura.

a) Fator Limitante do Sistema

Na tentativa de aproximação do ambiente de produção com uma situação real, foram considerados máquina e a mão-de-obra como fatores limitantes do sistema. SAAD (1996) e LIU e DONG (1996) utilizaram somente a máquina como fator limitante. SUN e YIH (1996) e CHEN e YIH

(1996), além da máquina consideraram respectivamente um robô e um guindaste para o transporte das peças.

b) Número de Máquinas

Segundo BAKER (1974), não existe evidência que o número de máquinas tenha influência crucial no desempenho relativo das regras de prioridade na programação da produção. NANOT apud BUFFA e MILLER (1979) também constatou que o tamanho da fábrica foi muito pouco significativo no mecanismo de análise de variância associado ao tempo médio de permanência no sistema para dez regras de decisão.

Dessa forma, a célula de manufatura estudada foi composta por 5 (cinco) máquinas, M1, M2, M3, M4 e M5. Essas máquinas são respectivamente um torno, uma fresa, uma furadeira, uma estação de trabalho para retirar as rebarbas das peças e uma máquina de lavar. A primeira operação de cada tipo de peça pode ser executada em qualquer uma das máquinas M1 (torno), M2 (fresa) e M3 (furadeira). A última operação de cada tipo de peça pode ser executada em M4 (rebarbas) ou M5 (lavar).

Nos modelos de LIU e DONG (1996), SUN e YIH (1996), CHEN e YIH (1996) e SAAD (1996) sobre aplicações de redes neurais artificiais na programação da produção o número de máquinas utilizadas esteve também em torno de cinco.

c) Operador

Segundo FERNANDES (1991), na manufatura celular o número de operadores é menor que o número de máquinas e, além disso, um operador pode operar mais de um equipamento (operário multifuncional). Dessa forma, na célula em estudo foram considerados três operadores, capazes de operar todas as máquinas.

d) Tipos de Peças

No ambiente de produção foi considerado o processamento de 5 (cinco) tipos de peças com similaridades de projeto e de processo: A, B, C, D e E. SAAD (1996) considerou uma família de peças formada por três tipos de peças. LIU e DONG (1996) simularam a passagem de cinco tipos de peças pelo sistema.

e) Processo de Chegada

Segundo HAX e CANDEA (1984), a maioria dos sistemas de produção *job-shop* são dinâmicos, novas ordens são periodicamente liberadas para o chão-de-fábrica enquanto outras são completadas. A hipótese típica é de chegadas das tarefas segundo uma distribuição de Poisson.

No ambiente de produção em estudo, o processo de chegada foi aleatório, em lotes. Um lote é formado por um único tipo de peça. Todas as peças que compõem um lote possuem a mesma data de entrega, d_i .

Com o objetivo de identificar um modelo físico o mais próximo da realidade de um chão-de-fábrica, com algumas restrições, é claro, o processo de chegada dos lotes foi dinâmico probabilístico, ou seja, os lotes chegaram na célula segundo uma distribuição Poisson, com intervalo entre chegadas distribuído exponencialmente.

Os lotes chegavam na célula num local chamado recepção. Não era realizado nenhum processamento neste local. Não foi permitida a chegada de dois lotes consecutivos do mesmo tipo de peça. A frequência entre chegadas dos lotes foi estabelecida em 5 minutos (média da distribuição exponencial). Este valor foi dimensionado em rodadas preliminares da simulação de modo a existir sempre um pequeno número de lotes na recepção.

f) Tamanho do Lote

Para satisfazer os desejos individuais dos clientes, a diversificação dos tipos de produtos fabricados nos processos de produção atuais torna-se

praticamente uma exigência de mercado. A consequência direta dessa tendência num ambiente produtivo é a redução do tamanho de lote. Dessa forma, neste trabalho considerou-se o tamanho do lote variável, segundo uma distribuição uniforme entre oito e vinte peças, ou seja, $U[8,20]$. Essa mesma variação foi proposta por LIU e DONG (1996).

Foram geradas chegadas de 60 (sessenta) lotes no sistema. Este número foi estabelecido nas rodadas preliminares de simulação de forma a simular um período mínimo de oito horas de trabalho. No apêndice B encontra-se o programa desenvolvido em linguagem Pascal para geração da seqüência de chegada dos lotes. A dinâmica da simulação será apresentada na seção 5.5 deste capítulo.

g) Datas de Entrega

CONWAY (1965) em estudo sobre regras de prioridade em um *job-shop* analisou várias formas de determinação de datas de entrega e a que se mostrou mais eficaz foi a estabelecida em relação ao conteúdo total de trabalho (TWK – *total-work due-dates*). Aquele autor considerou a permanência ($a_i = d_i - r_i$) de uma tarefa no sistema igual 9 (nove) vezes o seu tempo de processamento.

No presente trabalho, a data de entrega (d_i) de um lote foi estabelecida como a soma entre a data de liberação (instante de chegada do lote no sistema) e 5 (cinco) vezes o tempo de processamento do lote, ou seja:

$$d_i = r_i + 5p_i, \quad \text{onde}$$

d_i = data de entrega do i -ésimo lote

r_i = data de liberação do i -ésimo lote

p_i = tempo de processamento do i -ésimo lote

Dessa forma, os lotes somente foram considerados atrasados após um tempo equivalente a cinco vezes o seu tempo de processamento. Este valor, tendo como base o estudo de CONWAY (1965), foi o que melhor apresentou a ocorrência de adiantamentos e atrasos, testados nas rodadas

preliminares de simulação. O apêndice C apresenta a seqüência de chegada dos lotes e suas respectivas datas de entrega.

h) Tempo de Processamento

Com o objetivo de modelar o ambiente de produção o mais próximo de um sistema real, foram considerados tempos de processamento determinísticos. Devido à tendência de automatização dos processos de produção, a variabilidade do tempo de processamento passa a ser muito pequena de uma peça para outra. Para as máquinas M1 (torno), M2 (fresa) e M3 (furadeira), os tempos de processamento de cada peça em cada uma das máquinas, foram determinados *a priori* por uma distribuição uniforme entre 0,5 (meio minuto) e 1,5 (um minuto e meio), ou seja, $U [0,5; 1,5]$ minutos. Para a máquina M4 (rebarbas), o tempo de processamento de cada peça foi determinado segundo uma distribuição uniforme entre 15 (quinze) e 30 (trinta) segundos, ou seja, $U [0,25; 0,5]$ minutos. O tempo de processamento na máquina de lavar, M5, foi considerado constante e igual a 15 (quinze) segundos para todos os lotes. Os cálculos dos tempos de processamento de todos os lotes encontram-se no apêndice D.

LIU e DONG (1996) utilizaram tempos de processamento determinísticos com distribuição uniforme $U[5, 20]$ unidades de tempo. SAAD (1996) também considerou os tempos de processamento determinísticos, com um mínimo de 1 (um) e um máximo de 12 (doze) minutos.

i) Roteiro de Fabricação

O padrão de fluxo no interior da célula de manufatura em estudo é *job-shop*. Dessa forma, cada tipo de peça possui seu próprio roteiro de fabricação através das máquinas. Além disso, os roteiros de fabricação são conhecidos *a priori* e não são permitidos roteiros alternativos. O roteiro de fabricação de cada um dos cinco tipos de peça fabricados na célula é apresentado a seguir:

- Peça A: M1 M2 M3 M4 M5
- Peça B: M1 M3 M2 M4

- Peça C: M3 M2 M5
- Peça D: M2 M3 M4
- Peça E: M1 M2 M4 M5

j) *Setup*

O tempo de *setup* é um importante parâmetro na programação da produção em ambientes reais. Neste trabalho o *setup* foi considerado independente do tempo de processamento, ou seja, dependente da seqüência de processamento dos lotes nas máquinas. Segundo FERNANDES (1991), em células de manufatura, a necessidade de produzir grande variedade de produtos em pequenos lotes forçou a queda nos tempos de *setup*. No presente trabalho, como as peças processadas na célula possuem certa similaridade, foi estabelecido um *setup* pequeno, entre 2% a 6% do tempo de processamento total do lote que irá entrar na máquina. Foi considerado tempo de *setup* para as máquinas M1 (torno), M2 (fresa) e M3 (furadeira). Para as máquinas M4 (rebarbas) e M5 (lavar) não foi estabelecido *setup*, pelo fato de não existir ajustes ou troca de ferramentas de um lote para outro nessas máquinas. As tabelas com os tempos de *setup* encontram-se no apêndice E. SUN e YIH (1996) também consideraram o tempo de *setup* independente do tempo de processamento.

k) Capacidade das Filas

Foi considerada uma fila de entrada e uma de saída para cada máquina. Após o processamento da última operação de cada peça não foi considerada fila de saída, pois as peças já deixavam o sistema. A capacidade das filas foi estabelecida em 5000 lotes, ou seja, suficientemente grande modo a não prejudicar a dinâmica de formação de filas na célula.

SAAD (1996) e LIU e DONG (1996) consideraram somente filas de entrada, de capacidade ilimitada.

l) Medidas de Desempenho

Como comentado anteriormente, de acordo com CONWAY et al. (1967), as medidas de desempenho utilizadas para avaliar a programação da produção estão associadas ao congestionamento do sistema e ao cumprimento aos prazos de entrega. Nesse sentido, as medidas de desempenho analisadas foram:

1. \bar{F} : tempo médio de fluxo ou tempo médio de permanência, por lote, no sistema – refere-se à média sobre os cinco tipos de peças;
2. \bar{T}_f : tempo médio de espera, por lote, em filas – refere-se à média sobre as filas nas cinco máquinas;
3. %U: Porcentagem de utilização média das máquinas – refere-se à média das porcentagens de utilização de cada máquina;
4. \bar{L} : Atraso médio por lote – refere-se à média sobre os cinco tipos de peças. O atraso é calculado a partir da data de entrega do lote.

As medidas 1, 2 e 3 estão associadas ao congestionamento do sistema. A medida 4 está associada ao cumprimento aos prazos de entrega.

m) Regras de Prioridade

No ambiente de produção em estudo, e de acordo com o objetivo do trabalho, foi atribuída uma regra de prioridade a cada máquina. Foram analisadas as seguintes regras de prioridade:

- FIFO: o primeiro lote a entrar na fila é o primeiro a ser atendido;
- SPT: seleciona da fila o lote com o menor tempo de processamento;
- EDD: seleciona da fila o lote com menor data de entrega;
- SLACK: seleciona da fila o lote com menor folga dinâmica (data de entrega menos o tempo atual, menos o tempo de processamento remanescente);
- STT: seleciona da fila o lote com menor soma do tempo de processamento e de *setup*.

Às máquinas M1 (torno), M2 (furadeira) e M3 (fresa) foram alocadas qualquer uma das cinco regras anteriores. Como a máquina M4 é uma estação de trabalho para retirar rebarbas, não exige troca de ferramentas ou

ajustes na troca de lotes (não existe *setup*). Portanto, para M4 não foi alocada a regra STT. À máquina M5 (lavar) somente foi atribuída a regra FIFO, na tentativa de aproximação do modelo de um sistema real.

SUN e YIH (1996) analisaram sete regras de prioridade: FIFO, SPT, EDD, SLACK, STT, CR - razão crítica: tempo que resta até a data de entrega dividido pelo tempo de processamento remanescente, e SST que considera apenas o *setup* da peça.

Como tanto as regras SLACK como CR são proporções da data de entrega do lote, neste trabalho foi analisada somente a primeira, a folga dinâmica, que é uma medida da viabilidade de se executar uma determinada operação antes do lote se tornar atrasado. Como o *setup* é uma proporção do tempo de processamento total do lote, neste trabalho foi considerada a regra STT, que engloba essas duas grandezas.

5.4 O Procedimento Utilizado

O procedimento de utilização combinada da técnica de simulação de sistemas com a tecnologia de redes neurais artificiais foi proposto por CHRYSSOLOURIS et al. em 1990. Como abordado no capítulo 4, estes autores sugeriram que uma rede neural do tipo *Perceptron* de Múltiplas Camadas - MLP - poderia ser utilizada como uma função inversa da simulação. Este procedimento motivou o desenvolvimento de várias aplicações de redes neurais na programação da produção como os trabalhos de SAAD (1996), LIU e DONG (1996), SUN e YIH (1996) e CHEN e YIH (1996), já revisados no capítulo 4.

Na abordagem convencional de simulação de sistemas primeiramente é atribuída uma regra de prioridade a cada máquina, simulada a passagem das peças pelo sistema segundo essas regras e ao final coletados valores de algumas medidas de desempenho. Caso os valores das medidas não

sejam satisfatórios são atribuídas novas regras, num processo de tentativa e erro.

O procedimento aqui descrito opera ao contrário do modo convencional, ou seja, o modelo de redes neurais trabalha de maneira inversa ao simulador. Para tanto, primeiro é atribuída uma regra de prioridade a cada máquina, simulada a passagem dos lotes pelo sistema e são coletados valores de medidas de desempenho, da forma convencional como descrito anteriormente. A seguir, os resultados da simulação (valores de medidas de desempenho) são apresentados como entradas da rede neural. Para cada entrada da rede é também apresentada uma saída desejada (regras de prioridade atribuídas às máquinas). Este processo é chamado treinamento da rede neural, no qual a rede aprende as relações entre os valores das medidas de desempenho e as regras de prioridade. Após o treinamento, a rede deve ser testada. Na fase de testes são apresentados à rede valores de medidas de desempenho diferentes daqueles apresentados no treinamento. A rede então, deve ser capaz de prescrever regras de prioridade para os novos valores de medidas, ou seja, deve ser capaz de fazer a generalização para novos exemplos. As regras prescritas pela rede deverão ainda ser verificadas no simulador. Os valores das medidas de desempenho resultantes das regras prescritas pela rede devem ser comparados com os valores das medidas de desempenho obtidos pela simulação, de acordo com o objetivo do trabalho. Esse procedimento, adaptado de SAAD (1996), é descrito passo a passo a seguir:

Passo 1

Atribua uma regra de prioridade para cada máquina.

Passo 2

Simule a passagem dos lotes pelo sistema. A prioridade dos lotes em cada máquina depende da regra atribuída no passo 1.

Passo 3

Colete os valores das medidas de desempenho (resultados da simulação). Os passos 1, 2 e 3 devem ser repetidos para cada combinação de regras de prioridade às máquinas.

Passo 4

Prepare os valores das medidas de desempenho obtidos no passo 3 para que sirvam como entradas para o treinamento da rede neural. Essa preparação consiste em deixar todas as grandezas na mesma escala para evitar distorções, e será comentada na seção 5.6.

Passo 5

Defina todos os parâmetros da rede neural, como número de neurônios em cada camada, taxa de aprendizagem, e outros. Esses parâmetros serão detalhados na seção 5.6.

Passo 6

Faça o treinamento da rede neural. O treinamento é realizado pela aplicação do algoritmo *back propagation* apresentado no capítulo 3.

Passo 7

Caso os resultados do treinamento não sejam satisfatórios execute os passos 5 e 6 novamente com a alteração de alguns parâmetros. Quando os resultados do treinamento forem satisfatórios vá para o passo 8.

Passo 8

Faça o teste de generalização da rede neural com valores de medidas de desempenho não apresentados durante o treinamento. As saídas obtidas pela rede neural serão as regras de prioridade prescritas para cada máquina.

Passo 9

Verifique no simulador as regras de prioridade prescritas pela rede, ou seja, atribua as regras prescritas às máquinas, simule a passagem dos lotes e colete os novos valores de medidas de desempenho.

Passo 10

Compare os novos valores das medidas de desempenho (obtidos no passo 8) com os valores das medidas de desempenho coletados no passo 3. Se os resultados não forem satisfatórios, volte ao passo 5, altere alguns parâmetros da rede neural e execute os passos 6 a 9. Se os resultados forem satisfatórios vá para o passo 11.

Passo 11

A rede neural poderá ser utilizada para prescrição de regras de prioridade.

A figura 5.1 ilustra sinteticamente essa seqüência de passos e ao mesmo tempo fornece uma visão do procedimento para a solução do problema de programação da produção.

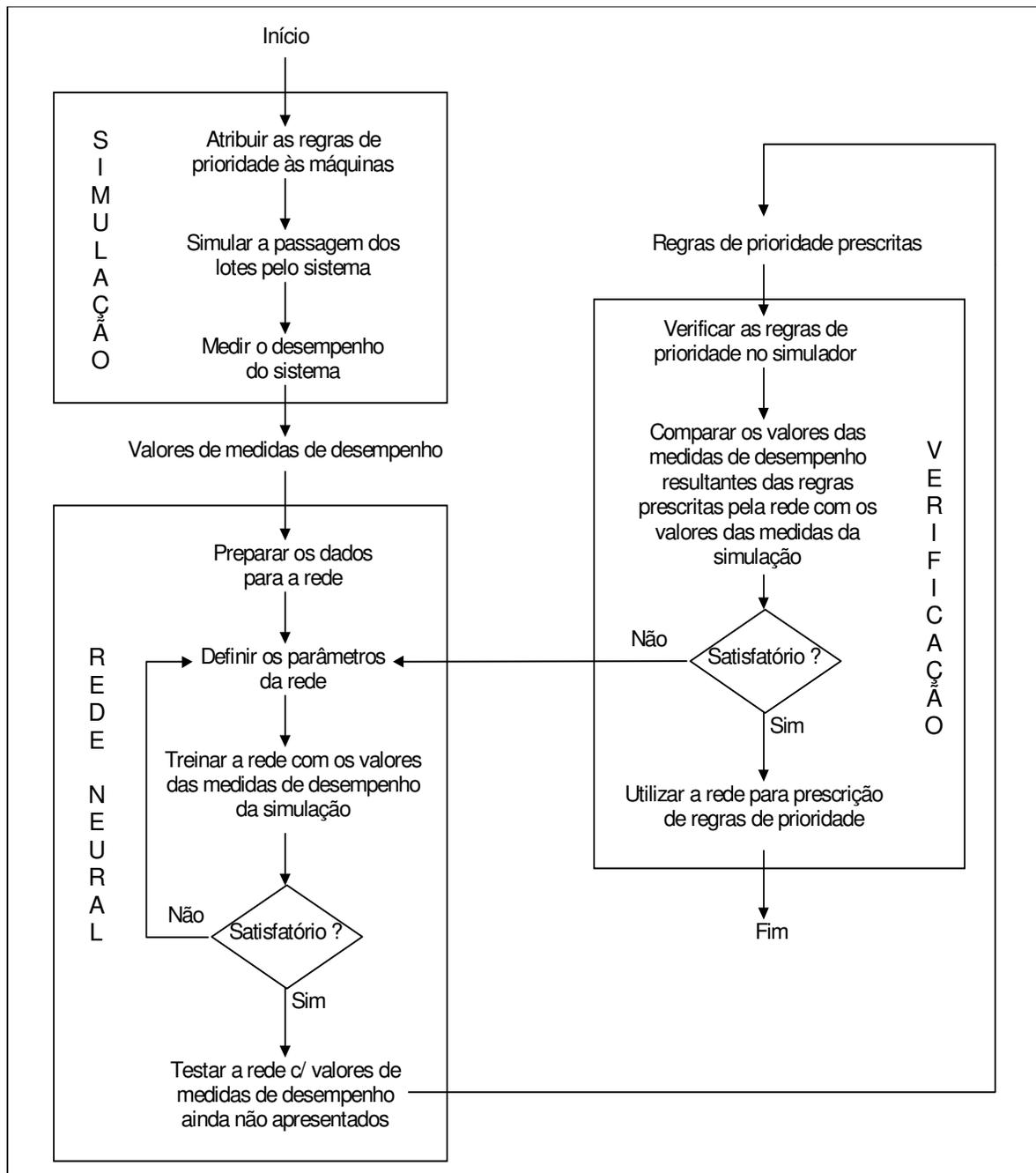


FIGURA 5.1 – Procedimento de solução do modelo híbrido

Como o próprio nome indica, o objetivo do módulo de verificação apresentado na figura 5.1 é verificar se as regras de prioridade prescritas pela rede neural irão produzir valores de medidas de desempenho compatíveis com aqueles obtidos na simulação e utilizados para treinar a rede. Se isso acontecer significa que a rede neural aprendeu as relações

entre as regras de prioridade e os valores das medidas de desempenho, e dessa forma poderá ser utilizada para a determinação de regras de prioridade às máquinas, a partir de valores de medidas de desempenho estabelecidos pelo usuário.

A seguir é apresentada a dinâmica da simulação.

5.5 A Simulação

O objetivo da simulação é obter os valores das medidas de desempenho resultantes da designação de regras de prioridade às máquinas, segundo a especificação do ambiente de produção apresentada na seção 5.3 deste capítulo. Como a simulação expressa a dinâmica de funcionamento do sistema, os valores das medidas de desempenho obtidos representam a realidade sob análise.

Como detalhado na seção 5.3, às máquinas M1 (torno), M2 (fresa) e M3 (furadeira) foram designadas qualquer uma das cinco regras de prioridade analisadas. À máquina M4 (rebarbas) foram designadas quatro regras de prioridade, e a máquina M5 (lavar) somente trabalhou com uma regra. Essas alocações geram um total de 500 possíveis combinações de regras de prioridade às máquinas. Não é o objetivo deste trabalho testar todas as combinações para escolher a melhor dentre as possíveis, e sim analisar a capacidade do modelo híbrido em realizar a diferenciação de um conjunto de combinações de regras de prioridade.

Foram analisadas 17 (dezessete) combinações de regras às máquinas, o que corresponde a 3,4% do universo de regras possíveis. SAAD (1996) analisou, de um total de 2500 combinações de regras às máquinas, 25 delas, ou seja, 1% das combinações possíveis. As combinações analisadas estão apresentadas na tabela 5.1.

Tabela 5.1 – Combinações de regras de prioridade às máquinas

| Combinação | M1 | M2 | M3 | M4 | M5 |
|------------|-------|-------|-------|-------|------|
| 1 | SLACK | SLACK | SLACK | SLACK | FIFO |
| 2 | FIFO | FIFO | FIFO | FIFO | FIFO |
| 3 | SPT | SPT | SPT | SPT | FIFO |
| 4 | EDD | EDD | EDD | EDD | FIFO |
| 5 | STT | STT | STT | FIFO | FIFO |
| 6 | FIFO | SPT | FIFO | FIFO | FIFO |
| 7 | FIFO | EDD | FIFO | FIFO | FIFO |
| 8 | FIFO | SLACK | FIFO | FIFO | FIFO |
| 9 | FIFO | STT | FIFO | FIFO | FIFO |
| 10 | SPT | FIFO | SPT | SPT | FIFO |
| 11 | SPT | EDD | SPT | SPT | FIFO |
| 12 | SPT | SLACK | SPT | SPT | FIFO |
| 13 | SPT | STT | SPT | SPT | FIFO |
| 14 | EDD | FIFO | EDD | EDD | FIFO |
| 15 | EDD | SPT | EDD | EDD | FIFO |
| 16 | EDD | SLACK | EDD | EDD | FIFO |
| 17 | EDD | STT | EDD | EDD | FIFO |

Para as combinações 1 a 5 foi designada a mesma regra de prioridade para todas as máquinas, salvo as restrições de alocação de regras para as máquinas M4 e M5.

A regra SPT está relacionada ao tempo de processamento das peças, e é conhecido da literatura (CONWAY et al., 1967, HAX e CANDEA, 1984), que esta regra é indicada para minimizar o tempo médio de fluxo, ou seja, o tempo médio de permanência das peças no sistema.

A regra EDD está relacionada à data de entrega das peças, portanto é indicada para minimizar medidas de desempenho relacionadas ao atraso (CONWAY, 1965).

A regra SLACK mede a folga dinâmica das peças, que relaciona a data de entrega e o tempo de processamento.

A regra FIFO é independente tanto do tempo de processamento como da data de entrega, e foi designada para todas as máquinas por corresponder a uma situação comum em fábricas.

Inicialmente foram feitas algumas rodadas preliminares de simulação com as regras FIFO, SPT, EDD e SLACK, nas quais pode-se constatar que a fresa, M2, representava a máquina gargalo do sistema. Dessa forma, as combinações de 6 a 17 foram escolhidas de forma a possibilitar também um estudo sobre a máquina gargalo, já que representam variações das regras sobre M2.

A figura 5.2 ilustra a combinação 12, com os roteiros de fabricação dos tipos de peças A e C.

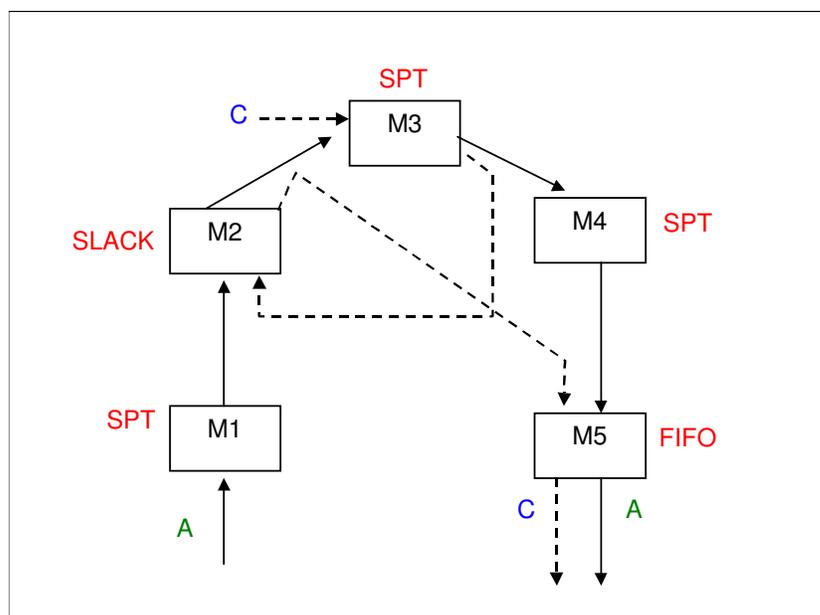


Figura 5.2 – Representação simplificada do sistema

Inicialmente pensou-se em utilizar o método de simulação *batch means*. Segundo LAW (1977), este método consiste em realizar uma longa rodada de simulação com k simulações independentes de comprimento m cada uma. Portanto, o tamanho dessa longa rodada é $n = k.m$. Cada uma das k simulações é chamada *batch*. Para se usar este método, o comprimento de cada *batch*, m , deve ser grande o suficiente para dissipar a dependência entre os dados próximos. Nas rodadas preliminares de

simulação foram feitos testes com o método *batch means*, entretanto, nem todos os 5 tipos de lotes (peças A, B, C, D e E) passavam por todos os *batches*, o que comprometiam as estatísticas. Portanto este método foi descartado, e foi utilizado o método da replicação.

No método da replicação, em cada rodada de simulação é especificado o número de replicações desejado. Em cada replicação todos os lotes passam pelo sistema e as estatísticas são coletadas ao término de cada replicação. Segundo SALLES (1998), o número de replicações por rodada é geralmente aceito como sendo em torno de 30 (trinta). O período de iniciação da simulação considerado, ou *warm-up*, foi de 2,5 h (duas horas e trinta minutos). O *warm-up* corresponde ao tempo em que o sistema ainda não entrou em equilíbrio, e portanto deve ser descartado. As estatísticas da simulação começam a ser contabilizadas após esse período. Este valor foi estabelecido pela observação gráfica do número médio de lotes em filas nas rodadas preliminares de simulação.

Para cada uma das 17 combinações de regras apresentadas na tabela 5.1, foram feitas 20 (vinte) rodadas de simulação com o método descrito acima. Este número foi estabelecido com o objetivo de fornecer um conjunto representativo de exemplos para o treinamento e teste da rede neural artificial. Na chegada dos lotes, em cada uma dessas rodadas utilizou-se uma *stream* diferente (posição inicial na seqüência de números aleatórios), devido à natureza dinâmica probabilística do processo de chegadas. Os valores das quatro medidas de desempenho, \bar{F} , \bar{T}_f , %U e \bar{L} , coletados ao final de cada rodada referem-se à média sobre as 30 (trinta) replicações.

Portanto, foram feitas 340 (trezentas e quarenta) rodadas de simulação, 20 para cada uma das 17 combinações de regras. Dentre as 20 rodadas de simulação, os valores das medidas de desempenho resultantes de 16 rodadas foram utilizados para o treinamento da rede neural. Os valores das medidas de desempenho resultantes de 4 (quatro) rodadas foram reservados para o teste da rede neural. A seguir é apresentada a dinâmica da rede neural.

5.6 A Rede Neural Artificial

A rede neural utilizada foi a *Perceptron* de Múltipla Camadas - MLP - que utiliza como regra de aprendizagem o algoritmo *back propagation*, já apresentado no capítulo 3. A rede MLP é indicada principalmente para classificação e reconhecimento de padrões. Esta rede realiza uma boa representação interna dos dados de entrada e já foi empregada em muitas aplicações de sucesso (LOESCH e SARI, 1996).

Segundo LOESCH e SARI (1996), os parâmetros da rede neural MLP, como o número de camadas escondidas, número de UP nessas camadas, valores dos pesos iniciais da rede, taxa de aprendizagem, termo de momento, número de ciclos, podem ser determinados por algumas heurísticas. De acordo com CHEN e YIH (1996), uma arquitetura de rede neural apropriada não pode somente reduzir rapidamente a função de erro para os dados de treinamento, mas também deve permitir uma generalização aceitável para os dados não apresentados (dados de teste).

Os parâmetros da rede MLP utilizada são apresentados a seguir.

a) Entradas da Rede

Como abordado anteriormente, as entradas da rede foram os valores das medidas de desempenho obtidos da simulação. Portanto, a camada de entrada é formada por 4 (quatro) neurônios ou unidades de processamento (UP). As UP de entrada representam:

UP1: \bar{F} – tempo médio de permanência, por lote, no sistema;

UP2: %U – porcentagem de utilização média das máquinas;

UP3: \bar{T}_f – tempo médio de espera, por lote, em filas;

UP4: \bar{L} – atraso médio por lote.

A preparação dos dados de entrada da rede, comentada no passo 4 do procedimento de solução do modelo híbrido, é também chamada de normalização. Os valores das entradas da rede, ou seja, das medidas de

desempenho, são números reais. A normalização consiste em transladar esses valores para um mesmo intervalo. De acordo com CHEN e YIH (1996), baseado no teorema de *Kolmogorov*, que será apresentado a seguir, os valores de entrada devem ser normalizados no intervalo $[0,1]$, de forma que a função de mapeamento entre as entradas e saídas possa ser facilmente encontrada. Mas, segundo a propriedade de saturação da função da sigmoideal, cuja figura encontra-se no capítulo 3, a maioria das saídas da rede estão entre 0,1 e 0,9. Portanto, os dados de entrada para cada medida de desempenho foram normalizados no intervalo $[0,1; 0,9]$.

b) Saídas da Rede

As saídas da rede são as regras de prioridade associadas a cada máquina no sistema. Como a célula de manufatura é composta por cinco máquinas, a rede possui 5 saídas, cada uma representando uma regra associada a cada máquina. As cinco regras de prioridade foram representadas por números inteiros positivos, de 1 a 5 (antes da normalização), da forma:

1. FIFO
2. SLACK
3. SPT
4. EDD
5. STT

A combinação 12, por exemplo, que utiliza respectivamente nas máquinas M1 a M5 as regras SPT, SLACK, SPT, SPT, FIFO, seria representada por:

$$\begin{matrix} M1 \\ M2 \\ M3 \\ M4 \\ M5 \end{matrix} \begin{pmatrix} 3 \\ 2 \\ 3 \\ 3 \\ 1 \end{pmatrix}$$

Da mesma forma que para as entradas da rede, as saídas também foram normalizadas no intervalo $[0,1; 0,9]$.

c) Número de Camadas Escondidas

Segundo FAUSETT (1994), resultados teóricos mostram que uma camada escondida é suficiente para aproximar qualquer mapeamento contínuo dos padrões de entrada em padrões de saída. Entretanto, duas camadas escondidas podem fazer o treinamento mais fácil. Dessa forma, no modelo de rede neural foram feitos testes com uma e com duas camadas escondidas.

d) Número de UP na Camada Escondida

As camadas escondidas de uma rede MLP são responsáveis pela modelagem não linear da rede. De acordo com HECHT-NIELSEN apud FERREIRA (1998), o número de UP na camada escondida pode ser definido inicialmente de acordo com o teorema de *Kolmogorov*: “qualquer função contínua $f: I^n \rightarrow R^m$ definida num espaço n -dimensional e com valores num espaço de dimensão m , pode ser representada exatamente por uma rede neural com n neurônios na camada de entrada, $2n+1$ na camada interna e m na camada de saída”. Portanto, este teorema foi utilizado inicialmente na determinação do número de UP na camada escondida. Além de $2n+1$, ou seja, 9 neurônios na camada escondida, também foram testadas topologias com 15, 20 e com 30 neurônios na camada escondida. Os resultados serão apresentados no capítulo 6.

A figura 5.3 apresenta uma representação simplificada da rede MLP utilizada neste trabalho.

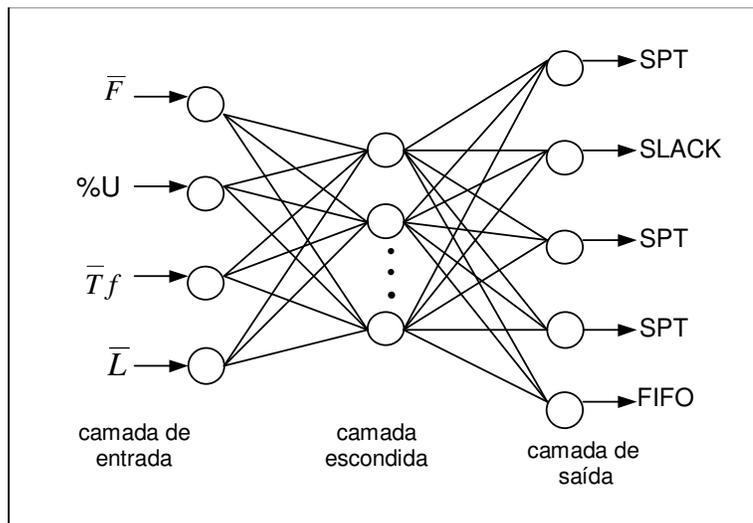


FIGURA 5.3 – Representação simplificada da rede MLP utilizada

e) Função de Ativação

Segundo FAUSETT (1994), a função de ativação utilizada no algoritmo *back propagation* deve ser contínua, diferenciável e monotonicamente não decrescente, e para eficiência computacional sua derivada deve ser fácil de calcular. A função de ativação utilizada neste estudo foi a sigmoideal, apresentada no capítulo 3.

De acordo com LOESCH e SARI (1996), na função sigmoideal, sinais de entrada de grande amplitude são acomodados sem saturação, e sinais baixos passam sem excessiva modificação. A derivada é sempre positiva, e próxima a zero para valores positivos ou negativos de x de grande valor absoluto. Essas propriedades são importantes para a lei de aprendizagem do algoritmo *back propagation*, pois as trocas nos pesos são proporcionais à derivada da ativação. Se as derivadas são pequenas as trocas nos pesos serão pequenas, para valores ao meio as trocas serão maiores.

f) Taxa de Aprendizagem

O valor da taxa de aprendizagem, η , influencia na velocidade de convergência da rede e na sua estabilidade. De acordo com LOESCH e SARI (1996), altas taxas de aprendizagem aceleram o processo de convergência pois forçam a rede a dar passos maiores na direção do menor

gradiente do erro a cada iteração. Por outro lado, podem produzir oscilações em torno de um ponto e não permitir a estabilização. Valores pequenos de taxa de aprendizagem estabilizam o processo, mas tornam a aprendizagem lenta, aumentando o tempo computacional. Neste trabalho foram testados dois valores de taxa de aprendizagem, 0,8 e 0,2.

g) Termo de Momento

Uma forma de acelerar o treinamento da rede é adicionar o termo de momento, α , na equação de atualização dos pesos no passo 7 do algoritmo *back propagation* apresentado no capítulo 3 (BEALE e JACKSON, 1990). Segundo LOESCH e SARI (1996), esse termo mede a persistência da variação dos pesos ao longo das iterações da rede. Neste trabalho foi atribuído o valor 0,9 para o termo de momento. Este valor foi o que apresentou melhores resultados nos trabalhos de SAAD (1996) e FERREIRA (1998).

h) Iniciação dos Pesos

O primeiro passo do algoritmo de treinamento *back propagation* é atribuir valores aleatórios pequenos aos pesos da rede. Esse procedimento é chamado de iniciação dos pesos. A atualização dos pesos entre duas UP depende da derivada da função de ativação, como apresentado no passo 7 do algoritmo *back propagation* no capítulo 3. Segundo FAUSETT (1994), os valores dos pesos iniciais não devem ser muito grandes, ou os sinais de entrada em cada unidade escondida ou de saída irão cair na região onde a derivada da função sigmoideal tem um valor muito pequeno (chamada região de saturação). Por outro lado, se os pesos iniciais forem muito pequenos, a ativação das unidades escondidas e de saída será próxima a zero, que também causa aprendizagem lenta. Neste trabalho foram utilizados valores iniciais de pesos entre -1,0 e 1,0. Também foram feitos alguns testes com pesos entre -0,5 e 0,5 e entre -4,0 e 4,0.

i) Escolha dos Padrões para o Treinamento

Segundo LOESCH e SARI (1996), os exemplos ou padrões de treinamento utilizados para treinar a rede devem possuir variações de características que a rede deve abstrair, ou seja, devem incluir casos fronteira para que a rede possa aprender a estabelecer limites de decisão. Cada padrão de treinamento é formado por uma porção de entrada e uma de saída. Neste trabalho, a porção de entrada refere-se aos valores das quatro medidas de desempenho obtidos da simulação. A porção de saída refere-se às regras associadas às cinco máquinas, que na simulação geraram os valores das medidas de desempenho. Vale lembrar que a rede neural MLP trabalha com o método de aprendizagem supervisionada, ou seja, na fase de treinamento são apresentadas as entradas e as saídas desejadas para cada entrada.

j) Tamanho do Conjunto de Treinamento

Como comentado na seção 5.5 para o treinamento da rede foram utilizados 16 exemplos (padrões) de cada uma das 17 combinações de regras de prioridade. Portanto, o conjunto de treinamento foi formado por 272 (duzentos e setenta e dois) padrões.

k) Forma de Apresentação dos Padrões de Treinamento

A seqüência de apresentação dos padrões de treinamento pode também influenciar no processo de aprendizagem da rede neural. Segundo LOESCH e SARI (1996), a estratégia mais utilizada é apresentar padrões de classes alternadas. Este procedimento evita com que a rede memorize a seqüência de padrões apresentados e facilita a generalização dos resultados. Neste trabalho os padrões foram apresentados em diferentes ordens nos diferentes ciclos (todos os padrões são apresentados uma vez durante cada ciclo).

l) Modo de Atualização dos Estados de Ativação

Segundo GALLANT (1993), numa rede MLP, cada UP calcula e atualiza sua ativação antes que a próxima UP seja examinada. Este modo de atualização das ativações é chamado ordem topológica. A ativação das UP de saída é interpretada como a saída da rede.

m) Critério de Parada do Treinamento

De acordo com o algoritmo de treinamento *back propagation* apresentado no capítulo 3, o critério de parada do treinamento pode ser estabelecido tanto em relação ao erro como em relação ao número de ciclos.

Um procedimento comum é estabelecer inicialmente um número pequeno de ciclos, por exemplo, 15, e observar o gráfico do erro durante o treinamento. Segundo FAUSETT (1994), quando o erro da rede começa a aumentar, significa o início de um processo de memorização dos padrões de treinamento, ou seja, a rede estará perdendo a capacidade de generalização. Neste trabalho foram feitos vários testes variando o número de ciclos e observando o gráfico do erro. Os resultados são apresentados no capítulo 6.

CHEN e YIH (1996) consideraram como aceitável para o treinamento um erro RMS (*root mean square*) de 4%, de 12% para o conjunto de testes. SAAD (1996) considerou como aceitável para o treinamento um erro de até 7% e para os testes de até 17%.

n) Número de Iterações para o Descarte da Rede

Uma rede neural deve ser descartada quando não houver convergência após um certo número de iterações. Esse número de iterações para o descarte varia de um modelo para outro e deve ser determinado, por exemplo, a partir de observações do gráfico do erro.

No capítulo 6 são apresentados os resultados obtidos.

Capítulo 6

Resultados

6.1 Introdução

Neste capítulo primeiramente serão apresentados os resultados da simulação correspondendo à aplicação dos passos 1, 2 e 3 do procedimento de solução do modelo híbrido apresentado no capítulo 5.

A seguir serão apresentados os resultados obtidos no treinamento e teste da rede neural artificial, correspondentes aos passos 4 a 8 do procedimento de solução.

Na seqüência serão apresentados os resultados da fase de verificação das regras prescritas pela rede no simulador (passo 9).

Por fim será apresentada a comparação entre os valores das medidas de desempenho resultantes de regras prescritas pela rede e os valores das medidas de desempenho obtidos inicialmente na simulação (passo 10).

6.2 Resultados da Simulação

O *software* de simulação de sistemas de manufatura utilizado para modelar o sistema de produção descrito no capítulo 5 foi o ProModel 4.0. Este *software* é bastante difundido tanto em aplicações comerciais como acadêmicas. Possui boa interface com o usuário e conta com um módulo de animação (PROMODEL, 1997).

Conforme comentado no capítulo 5, foram feitas 20 rodadas de simulação para cada uma das 17 combinações de regras às máquinas. Ao final de cada rodada de simulação foram coletados valores de quatro medidas de desempenho:

\bar{F} - tempo médio de permanência, por lote, no sistema (minutos);

\bar{T}_f - tempo médio de espera, por lote, em filas (minutos);

%U - porcentagem de utilização média das máquinas;

\bar{L} - atraso médio por lote (minutos).

As medidas \bar{F} , \bar{T}_f e \bar{L} são medidas regulares de desempenho, ou seja, são funções do tempo de término das tarefas em um sistema, e nesse sentido é interessante que sejam minimizadas (HAX e CANDEA, 1984). A porcentagem de utilização média das máquinas, %U, está relacionada à utilização eficiente ou não dos recursos. A utilização intensiva e eficiente de um recurso aumenta a produtividade do sistema, reduz a necessidade de altos níveis de estoque de segurança e do emprego de horas extras (HAX e CANDEA, 1984).

Os resultados das simulações referentes às 17 combinações de regras às máquinas estão apresentados no apêndice F. A tabela 6.1 a seguir ilustra os valores obtidos com a seguinte combinação de regras: M1: spt; M2: slack; M3: spt; M4: spt e M5:fifo. No capítulo 5 estão apresentados os significados de todas as regras de prioridade analisadas.

Tabela 6.1 – Resultados da combinação 12

M1: spt M2: slack M3: spt M4: spt M5:fifo

| Rodada | \bar{F} | \bar{T}_f | %U | \bar{L} |
|--------|------------|-------------|-----------|-----------|
| 1 | 277,102527 | 26,364534 | 54,547500 | 81,876013 |
| 2 | 275,087717 | 25,954838 | 54,745000 | 81,109619 |
| 3 | 279,870607 | 26,104028 | 54,465000 | 84,042076 |
| 4 | 274,953634 | 26,630640 | 54,797500 | 86,053609 |
| 5 | 275,185978 | 26,246477 | 54,812500 | 80,653697 |
| 6 | 282,224319 | 27,663737 | 54,487500 | 80,409210 |
| 7 | 276,363194 | 27,089603 | 54,892500 | 81,443506 |
| 8 | 274,581219 | 26,443688 | 54,300000 | 79,214085 |
| 9 | 284,187486 | 27,486034 | 54,535000 | 79,666469 |
| 10 | 223,901596 | 26,461150 | 54,550000 | 81,216212 |
| 11 | 279,416259 | 27,677688 | 54,407500 | 77,486675 |
| 12 | 275,301943 | 27,258039 | 54,737500 | 78,683564 |
| 13 | 278,669832 | 27,480647 | 54,690000 | 80,867501 |
| 14 | 275,026072 | 26,684851 | 54,537500 | 79,167317 |
| 15 | 273,185114 | 26,578685 | 54,862500 | 79,944268 |
| 16 | 272,554514 | 26,279314 | 54,912500 | 86,883366 |
| 17 | 275,965040 | 26,800413 | 54,730000 | 86,070370 |
| 18 | 276,570030 | 26,403512 | 55,142500 | 94,008353 |
| 19 | 281,593789 | 27,774304 | 54,705000 | 89,026755 |
| 20 | 274,769447 | 27,009090 | 54,690000 | 79,236480 |

Como abordado no capítulo 5, foram realizadas 20 rodadas de simulação para cada uma das 17 combinações de regras às máquinas. Tomando como exemplo a rodada número 1 da tabela 6.1, os resultados obtidos são interpretados da forma a seguir. O tempo médio de permanência por lote no sistema, \bar{F} , foi de 277 minutos, ou seja, aproximadamente 4h30min. O tempo médio de espera por lote em filas, \bar{T}_f , foi cerca de 26min. A utilização média das máquinas foi de 55%. O atraso médio de cada lote foi de 81 minutos, ou aproximadamente 1h30min.

6.3 Resultados de Treinamento e de Teste da Rede Neural

De posse dos valores das medidas de desempenho apresentados no apêndice F, e conforme o passo 4 do procedimento de solução do modelo híbrido foi feita a normalização por medida de desempenho (entrada da rede neural). Também foi feita a normalização da saída da rede, conforme descrito na seção 5.6. No apêndice G encontra-se um exemplo de parte do arquivo de entrada da rede neural com os dados já normalizados.

Para realizar o treinamento da rede neural artificial, correspondente aos passos 5 a 7 do procedimento de solução do modelo híbrido, foi utilizado o *software* SNNS – Stuttgart Neural Networks Simulator. Este *software* é bem difundido em aplicações acadêmicas e foi utilizado por FERREIRA (1998) para reconhecimento de caracteres. Este *software* dispõe de recursos como visualização do gráfico de erro durante o treinamento e da distribuição de pesos da rede. No capítulo 3 são apresentadas algumas características deste *software*. Uma análise mais detalhada sobre quatro *softwares* de redes neurais artificiais é apresentada no apêndice A.

A figura 6.1 apresenta uma tela do *software* SNNS.

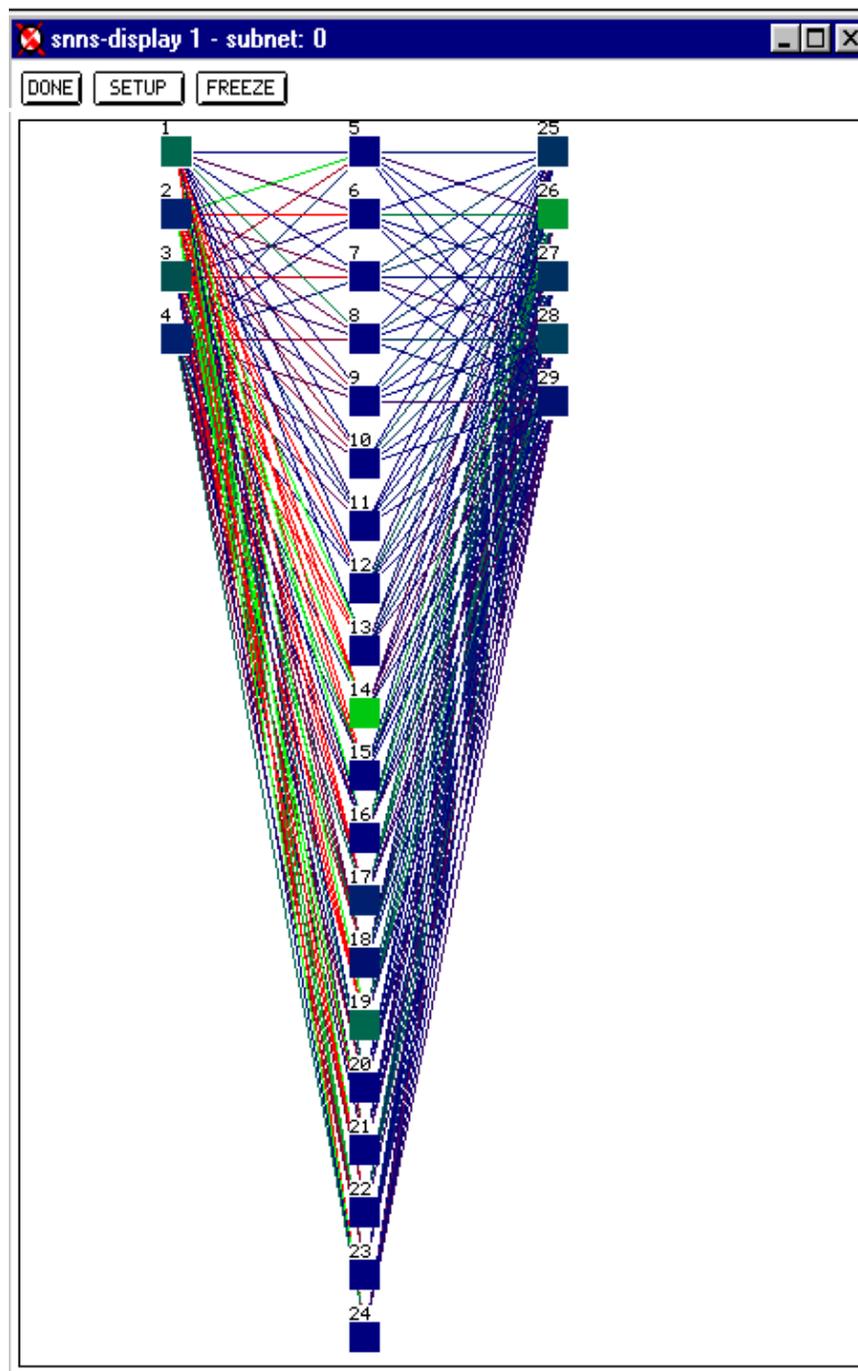


Figura 6.1 – Estrutura da rede neural utilizada – tela do *software* SNNS

A figura 6.1 ilustra a estrutura da rede neural utilizada neste trabalho. Como comentado no capítulo 5, a rede é constituída por 4 neurônios na camada de entrada, correspondente aos valores das quatro medidas de

desempenho, e 5 neurônios na camada de saída. As saídas representam as regras de prioridade associadas a cada máquina. O número de neurônios na camada interna assim como os testes realizados com diversas topologias de rede serão detalhados adiante.

Para analisar a sensibilidade do modelo de redes neurais foram feitos vários treinamentos, nos quais aumentou-se a quantidade de padrões (exemplos) apresentados à rede gradativamente. Primeiramente foram feitos treinamentos com 17 padrões, a seguir com 68 padrões e por fim com todos os 272 padrões. Apesar dos resultados com 17 e 68 padrões não terem sido verificados no simulador, esta experiência será relatada a seguir, pois foi importante para o conhecimento do comportamento do modelo de redes neurais.

O treinamento de uma rede neural, conforme o passo 6 do procedimento de solução do modelo híbrido consiste em treinar a rede até que o erro estabilize em um valor baixo.

Inicialmente foram feitas várias tentativas de treinamento com 17 padrões, sendo um padrão de cada combinação de regras. Foi escolhido o padrão mais representativo de cada combinação (sobre as respectivas rodadas de simulação). Este padrão refere-se aquele cujos valores das medidas de desempenho encontravam-se numa posição intermediária à variação de cada medida. Esta escolha foi feita de forma a caracterizar bem cada combinação. A tabela 6.2 apresenta os resultados dos treinamentos com 17 padrões. Os resultados estão ordenados por ordem crescente de erro.

Tabela 6.2 – Resultados dos treinamentos com 17 padrões

| No. | Topologia | No. Ciclos | Pesos | η | α | Erro % |
|-----|-----------|------------|--------|--------|----------|--------|
| 1 | 4-9-5 | 400 | -1 a 1 | 0,2 | 0,9 | 8,0 |
| 2 | 4-9-5 | 320 | -1 a 1 | 0,2 | 0,9 | 10,0 |
| 3 | 4-9-5 | 200 | -1 a 1 | 0,2 | 0,9 | 14,0 |
| 4 | 4-9-5 | 50 | -1 a 1 | 0,2 | 0,9 | 23,0 |
| 5 | 4-9-5 | 25 | -1 a 1 | 0,2 | 0,9 | 26,5 |
| 6 | 4-9-5 | 14 | -1 a 1 | 0,2 | 0,9 | 27,4 |
| 7 | 4-6-5 | 14 | -1 a 1 | 0,2 | 0,9 | 28,0 |

Na tabela 6.2, a topologia refere-se ao número de neurônios em cada camada: entrada, interna e de saída. O número de ciclos, de acordo com o algoritmo *back propagation* apresentado na capítulo 3, refere-se ao número de vezes que o conjunto de treinamento é apresentado à rede. Os pesos, a taxa de aprendizagem, η , e o termo de momento, α , são parâmetros da rede *back propagation* também comentados no capítulo 3.

Inicialmente foram treinadas duas redes com 14 ciclos cada. A primeira com seis neurônios na camada interna, e a segunda com nove neurônios. O erro de treinamento para essas duas rede foi extremamente alto em relação à literatura (vide capítulo 5). O número de ciclos foi aumentado progressivamente (25, 50, 200, 320 e 400). A rede treinada com 400 ciclos foi a que apresentou o menor erro, 8,0%.

Foi feito um teste de generalização com essa rede, no qual foram apresentados dois novos padrões de cada combinação de regras não pertencentes ao conjunto de treinamento. A generalização da rede não foi satisfatória, ou seja, ela classificou a maioria dos dados como pertencentes a uma única combinação de regras. Este fato confirma a literatura (LOESCH e SARI, 1996), pois aumentar o número de ciclos implica em apresentar os mesmos padrões mais vezes à rede. Isso faz com que a rede se torne especialista, ou seja, ela reproduz muito bem o conjunto de dados para os quais foi treinada, mas em compensação “memoriza” esses dados. Dessa forma, a rede perde sua capacidade de generalização, ou seja, capacidade

de apresentar respostas similares às respostas cujas entradas foram treinadas, para entradas não treinadas, como comentado no capítulo 3.

Foram realizadas novas tentativas de treinamento, dessa vez com 68 padrões, sendo 4 padrões de cada uma das 17 combinações de regras. Em outras palavras, foram determinados dois conjuntos de dados. O primeiro, chamado conjunto de treinamento, com os 68 padrões, como o próprio nome indica, foi usado para treinar a rede. O segundo, chamado conjunto de validação, foi usado para monitorar o desempenho da rede sobre padrões não apresentados no treinamento. Este conjunto era constituído por 34 outros padrões, sendo 2 de cada combinação de regras. O segundo conjunto é importante para monitorar *on-line* o treinamento. O treinamento deve ser interrompido quando o erro do conjunto de validação atingir um mínimo. Neste ponto a rede generaliza melhor. Quando o treinamento não é interrompido, ocorre o fenômeno chamado *overtraining*. Apesar do erro do treinamento continuar a diminuir, o desempenho da rede é reduzido e ela começa a “memorizar” os padrões.

A figura 6.2 apresenta um exemplo de treinamento com início de *overtraining*. A linha em vermelho apresenta a evolução do erro do conjunto de validação, e a linha preta, a evolução do erro do conjunto de treinamento.

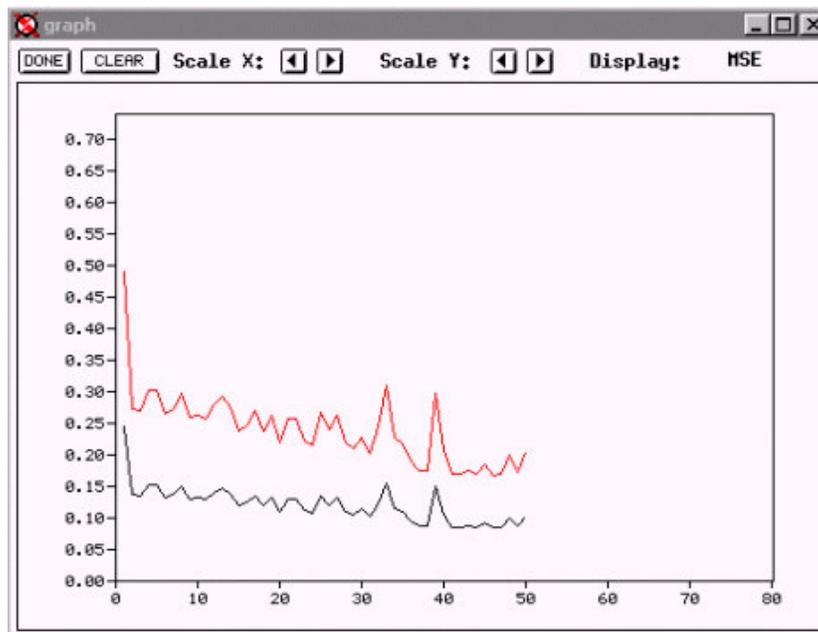


Figura 6.2 – Exemplo de treinamento com início de *overtraining*

Na figura 6.2, o eixo y apresenta o erro RMS (*root mean square*), e o eixo x apresenta o número de ciclos utilizados no treinamento, conforme comentado no capítulo 5.

A tabela 6.3 apresenta os resultados dos treinamentos com os 68 padrões. Os resultados estão ordenados por ordem crescente do erro do conjunto de treinamento. Nas topologias assinaladas com “n/c” não houve convergência durante o treinamento.

Tabela 6.3 – Resultados dos treinamentos com 68 padrões

| No. | Topologia | No. Ciclos | Pesos | η | α | Erro % | |
|-----|-----------|------------|--------|--------|----------|-------------|-----------|
| | | | | | | Treinamento | Validação |
| 1 | 4-20-5 | 50 | -1 a 1 | 0,2 | 0,9 | 8,0 | 15,0 |
| 2 | 4-20-5 | 38 | -1 a 1 | 0,2 | 0,9 | 8,5 | 17,0 |
| 3 | 4-30-5 | 50 | -1 a 1 | 0,2 | 0,9 | 9,5 | 18,0 |
| 4 | 4-20-5 | 38 | -1 a 1 | 0,1 | 0,9 | 10,0 | 20,0 |
| 5 | 4-15-5 | 50 | -1 a 1 | 0,2 | 0,9 | 10,0 | 20,0 |
| 6 | 4-15-15-5 | 50 | -1 a 1 | 0,2 | 0,9 | 10,0 | 20,0 |
| 7 | 4-20-5 | 38 | -1 a 1 | 0,3 | 0,9 | 11,0 | 22,0 |
| 8 | 4-20-20-5 | 50 | -1 a 1 | 0,2 | 0,9 | 11,0 | 22,0 |
| 9 | 4-9-5 | 50 | -1 a 1 | 0,2 | 0,9 | 12,0 | 23,0 |
| 10 | 4-4-5 | 15 | -1 a 1 | 0,2 | 0,9 | 14,0 | 17,0 |
| 11 | 4-15-5 | 25 | -1 a 1 | 0,2 | 0,9 | 14,0 | 25,0 |
| 12 | 4-9-5 | 15 | -1 a 1 | 0,2 | 0,9 | 15,0 | 25,0 |
| 13 | 4-9-5 | 25 | -1 a 1 | 0,2 | 0,9 | n/c | --- |
| 14 | 4-15-5 | 15 | -1 a 1 | 0,2 | 0,9 | n/c | --- |
| 15 | 4-20-5 | 38 | -1 a 1 | 0,8 | 0,9 | n/c | --- |

Para as tentativas de treinamento com 68 padrões, a rede que apresentou melhores resultados foi a com 20 neurônios na camada interna e 50 ciclos. Como apresentado na tabela 6.3, para esta rede o erro do conjunto de treinamento foi de 8% e do conjunto de validação de 15%.

Finalmente foram realizados os treinamentos com os 272 padrões. Esses padrões, que formam o conjunto de treinamento, correspondem aos valores de 16 rodadas de simulação para cada uma das 17 combinações de regras apresentadas nas tabelas do apêndice F. Para o conjunto de validação foram reservados valores de 4 rodadas de cada uma das 17 combinações de regras, não presentes no conjunto de treinamento. Portanto, o conjunto de validação foi constituído por 68 padrões.

A tabela 6.4 apresenta os resultados dos treinamentos com os 272 padrões. Foram realizados vários treinamentos variando-se além da topologia de rede, como o número de neurônios na camada interna, a taxa

de aprendizagem, η , o número de ciclos e o intervalo de iniciação dos pesos. Como comentado no capítulo 5, o termo de momento, α , utilizado, foi 0,9. Os dados são apresentados por erro de treinamento crescente.

Tabela 6.4 – Resultados dos treinamentos com 272 padrões

| No. | Topologia | No. Ciclos | Pesos | η | α | Erro % | |
|-----|-----------|------------|------------|--------|----------|-------------|-----------|
| | | | | | | Treinamento | Validação |
| 1 | 4-20-5 | 50 | -1 a 1 | 0,2 | 0,9 | 3,3 | 13,0 |
| 2 | 4-9-5 | 38 | -4 a 4 | 0,2 | 0,9 | 3,5 | 14,0 |
| 3 | 4-15-5 | 50 | -1 a 1 | 0,2 | 0,9 | 3,8 | 13,8 |
| 4 | 4-20-5 | 50 | -4 a 4 | 0,2 | 0,9 | 3,8 | 14,3 |
| 5 | 4-30-5 | 50 | -1 a 1 | 0,2 | 0,9 | 3,9 | 14,0 |
| 6 | 4-9-5 | 38 | -1 a 1 | 0,2 | 0,9 | 4,2 | 16,0 |
| 7 | 4-9-5 | 15 | -1 a 1 | 0,2 | 0,9 | 4,8 | 18,5 |
| 8 | 4-9-5 | 50 | -1 a 1 | 0,2 | 0,9 | 5,0 | 19,5 |
| 9 | 4-9-5 | 38 | -0,5 a 0,5 | 0,2 | 0,9 | 5,8 | 22,0 |
| 10 | 4-15-5 | 38 | -1 a 1 | 0,2 | 0,9 | 7,0 | 28,0 |
| 11 | 4-9-5 | 15 | -1 a 1 | 0,8 | 0,9 | 10,2 | 41,0 |

No treinamento com todos os 272 padrões, a rede que apresentou os melhores resultados foi a com 20 neurônios na camada interna, com erro do conjunto de treinamento de 3,3% e do conjunto de validação de 13,0%. Como abordado no capítulo 5, SAAD (1996) considerou como aceitável para o conjunto de treinamento um erro de até 7% e para o conjunto de validação de até 17%.

Pelas tabelas 6.2, 6.3 e 6.4, observa-se a importância do tamanho do conjunto de treinamento. Os erros de treinamento e de validação, de um modo geral, somente tiveram uma redução significativa quando a rede foi treinada com os 272 padrões.

Os resultados apresentados nas tabelas 6.2, 6.3 e 6.4, correspondem à execução dos passos 5, 6 e 7 do procedimento de solução do modelo híbrido apresentado no capítulo 5. O passo 8 consiste em realizar o teste de generalização.

O teste de generalização corresponde à fase de ativação da rede neural. Como abordado no capítulo 3, a ativação consiste em apresentar à rede neural já treinada um padrão (ou conjunto de padrões) para ser classificado. A rede utiliza apenas a fase de propagação para frente do algoritmo de treinamento *back propagation*. O teste de generalização somente foi realizado com a rede número1 da tabela 6.4, pois foi a que apresentou o menor erro de treinamento (3,3%) e de validação (13,0%). No teste de generalização foi apresentado à rede o mesmo conjunto de 68 padrões utilizado na validação. Vale lembrar que a validação é apenas um mecanismo para monitorar o treinamento. O conjunto de 68 padrões era formado pelos valores das medidas de desempenho de 4 rodadas de simulação (não presentes no conjunto de treinamento) de cada uma das 17 combinações de regras.

Dessa forma, o teste de generalização consistiu em apresentar à rede já treinada valores de medidas de desempenho não apresentados no treinamento. As saídas da rede foram as regras de prioridade prescritas para cada máquina. A tabela 6.5 apresenta as regras prescritas pela rede.

Tabela 6.5 – Regras prescritas pela rede neural

| Regras Simulador | | Regras Prescritas pela Rede | | | | |
|---|--------|-----------------------------|-------|-------|-------|------|
| | Rodada | M1 | M2 | M3 | M4 | M5 |
| Combinação 1 slack slack slack slack fifo | 3 | slack | slack | slack | spt | fifo |
| | 5 | slack | slack | slack | spt | fifo |
| | 12 | slack | slack | slack | spt | fifo |
| | 13 | spt | slack | spt | spt | fifo |
| Combinação 2 fifo fifo fifo fifo fifo | 1 | fifo | slack | fifo | fifo | fifo |
| | 5 | fifo | slack | fifo | fifo | fifo |
| | 15 | fifo | slack | fifo | fifo | fifo |
| | 20 | fifo | slack | fifo | fifo | fifo |
| Combinação 3 spt spt spt spt fifo | 1 | slack | edd | slack | slack | fifo |
| | 3 | edd | stt | edd | spt | fifo |
| | 7 | spt | edd | spt | slack | fifo |
| | 10 | edd | edd | edd | spt | fifo |
| Combinação 4 edd edd edd edd fifo | 8 | slack | spt | slack | slack | fifo |
| | 12 | slack | spt | slack | spt | fifo |
| | 14 | slack | spt | slack | spt | fifo |
| | 15 | fifo | spt | fifo | slack | fifo |

continua

continuação Tabela 6.5 – Regras prescritas pela rede neural

| Regras Simulador | | Regras Prescritas pela Rede | | | | |
|--|--------|-----------------------------|-------|-------|-------|------|
| | Rodada | M1 | M2 | M3 | M4 | M5 |
| Combinação 5 stt stt stt fifo fifo | 7 | spt | edd | spt | spt | fifo |
| | 14 | edd | edd | edd | spt | fifo |
| | 15 | spt | stt | spt | spt | fifo |
| | 20 | spt | edd | spt | slack | fifo |
| Combinação 6 fifo spt fifo fifo fifo | 1 | fifo | spt | fifo | fifo | fifo |
| | 3 | slack | edd | slack | slack | fifo |
| | 8 | fifo | spt | fifo | fifo | fifo |
| | 14 | fifo | spt | fifo | slack | fifo |
| Combinação 7 fifo edd fifo fifo fifo | 1 | fifo | spt | fifo | slack | fifo |
| | 6 | slack | edd | slack | slack | fifo |
| | 7 | slack | edd | slack | slack | fifo |
| | 14 | slack | edd | slack | slack | fifo |
| Combinação 8 fifo slack fifo fifo fifo | 1 | slack | spt | slack | slack | fifo |
| | 7 | fifo | slack | fifo | fifo | fifo |
| | 16 | slack | slack | slack | slack | fifo |
| | 20 | slack | spt | slack | slack | fifo |
| Combinação 9 fifo stt fifo fifo fifo | 5 | fifo | spt | fifo | fifo | fifo |
| | 7 | fifo | spt | fifo | fifo | fifo |
| | 10 | fifo | spt | fifo | fifo | fifo |
| | 20 | fifo | spt | fifo | fifo | fifo |
| Combinação 10 spt fifo spt spt fifo | 5 | spt | fifo | spt | spt | fifo |
| | 8 | spt | fifo | spt | spt | fifo |
| | 10 | spt | fifo | spt | spt | fifo |
| | 14 | spt | fifo | spt | spt | fifo |
| Combinação 11 spt edd spt spt fifo | 1 | spt | edd | spt | spt | fifo |
| | 3 | slack | edd | slack | spt | fifo |
| | 7 | spt | edd | spt | spt | fifo |
| | 20 | slack | edd | slack | slack | fifo |
| Combinação 12 spt slack spt spt fifo | 1 | slack | slack | slack | spt | fifo |
| | 2 | slack | slack | slack | spt | fifo |
| | 14 | slack | slack | slack | slack | fifo |
| | 20 | slack | slack | slack | slack | fifo |
| Combinação 13 spt stt spt spt fifo | 2 | spt | edd | spt | slack | fifo |
| | 10 | edd | stt | edd | spt | fifo |
| | 13 | spt | edd | spt | spt | fifo |
| | 16 | edd | stt | edd | slack | fifo |
| Combinação 14 edd fifo edd edd fifo | 1 | slack | slack | slack | slack | fifo |
| | 7 | slack | slack | slack | spt | fifo |
| | 15 | fifo | slack | fifo | slack | fifo |
| | 19 | slack | slack | slack | slack | fifo |
| Combinação 15 edd spt edd edd fifo | 3 | stt | stt | stt | edd | fifo |
| | 8 | slack | edd | slack | spt | fifo |
| | 12 | edd | edd | edd | edd | fifo |
| | 15 | slack | edd | slack | slack | fifo |

continua

continuação Tabela 6.5 – Regras prescritas pela rede neural

| Regras Simulador | | Regras Prescritas pela Rede | | | | |
|--|--------|-----------------------------|-------|-------|-------|------|
| | Rodada | M1 | M2 | M3 | M4 | M5 |
| Combinação 16 edd slack edd edd fifo | 6 | edd | slack | edd | stt | fifo |
| | 7 | spt | slack | spt | spt | fifo |
| | 10 | spt | slack | spt | edd | fifo |
| | 20 | spt | slack | spt | edd | fifo |
| Combinação 17 edd stt edd edd fifo | 1 | edd | edd | edd | edd | fifo |
| | 12 | edd | edd | edd | edd | fifo |
| | 15 | slack | edd | slack | slack | fifo |
| | 19 | spt | edd | spt | spt | fifo |

O significado de todas as regras da tabela 6.5 está detalhado no capítulo 2.

A rede neural realizou somente uma prescrição inviável, designando a regra STT para M4, na combinação 16 rodada 6. Esta prescrição não foi verificada no simulador. Como pode ser observado pela tabela 6.5, para todos os padrões testados a rede prescreveu a regra FIFO para M5 (lavar), como esperado.

A seguir serão apresentados os resultados da verificação no simulador.

6.4 Resultados da Verificação no Simulador

As regras de prioridade prescritas pela rede, conforme o passo 9 do procedimento de solução do modelo híbrido do capítulo 5, foram verificadas no simulador. A verificação consiste em comparar os valores das medidas de desempenho resultantes das regras de prioridade prescritas pela rede neural, com os valores das medidas de desempenho obtidos da simulação.

A comparação é feita através do cálculo do erro. Como o objetivo do trabalho é testar a capacidade do modelo híbrido em realizar a diferenciação entre um conjunto de regras de prioridade às máquinas, o erro é calculado em relação ao intervalo de variação de cada medida de desempenho obtida das regras prescritas pela rede neural.

Dessa forma, o erro é calculado por:

$$Erro = \frac{|mr - ms|}{v} \quad , \text{ onde} \quad (6.1)$$

mr = valor da medida de desempenho prescrito pela rede

ms = valor da medida de desempenho obtido pelo simulador

v = variação da medida de desempenho da rede

Os valores das medidas de desempenho resultantes das regras prescritas pela rede são apresentados na tabela 6.6. A coluna Combinação.Rodada serve apenas para fazer referência aos valores das medidas anteriores do simulador (apêndice F).

Tabela 6.6 – Medidas de desempenho prescritas pela rede

| Combinação. Rodada | \bar{F} | \bar{T}_f | %U | \bar{L} |
|-----------------------|-----------|-------------|---------|-----------|
| 1.3 | 271,4759 | 26,0473 | 54,9425 | 75,6315 |
| 1.5 | 269,8725 | 26,0628 | 54,9875 | 75,7130 |
| 1.12 | 268,2980 | 26,5478 | 54,6425 | 71,0204 |
| 1.13 | 271,4870 | 28,0427 | 54,4650 | 65,4034 |
| 2.1 | 267,3444 | 28,1478 | 53,8900 | 59,7248 |
| 2.5 | 266,5747 | 27,7482 | 54,2225 | 59,3684 |
| 2.15 | 265,1523 | 29,2196 | 53,8975 | 58,5973 |
| 2.20 | 267,2976 | 28,0503 | 54,1525 | 58,4482 |
| 3.1 | 337,1714 | 33,1963 | 57,0850 | 102,4334 |
| 3.3 | 271,3260 | 26,6249 | 52,7750 | 41,9535 |
| 3.7 | 341,3011 | 33,9295 | 56,7575 | 106,0330 |
| 3.10 | 337,7533 | 33,9760 | 57,0900 | 101,7194 |
| 4.8 | 259,8860 | 25,6174 | 53,8350 | 62,3441 |
| 4.12 | 262,0827 | 26,9743 | 53,3925 | 64,5644 |
| 4.14 | 263,1120 | 25,7686 | 53,5600 | 64,6919 |
| 4.15 | 257,8695 | 28,7712 | 53,6900 | 53,5959 |
| 5.7 | 340,2087 | 33,7682 | 56,7775 | 104,9406 |
| 5.14 | 336,1161 | 33,5664 | 57,0950 | 98,8290 |
| 5.15 | 262,1334 | 26,0273 | 53,0600 | 68,5294 |
| 5.20 | 344,6287 | 34,4848 | 56,8725 | 110,5743 |
| 6.1 | 262,0428 | 28,9104 | 53,3575 | 56,5302 |
| 6.3 | 338,4208 | 33,5595 | 57,1425 | 102,0350 |
| 6.8 | 260,2745 | 28,8474 | 53,6825 | 56,5258 |
| 6.14 | 260,4835 | 28,4230 | 53,7875 | 51,9455 |
| 7.1 | 261,6384 | 28,9864 | 53,3850 | 56,4156 |
| 7.6 | 342,2060 | 33,6073 | 57,3000 | 101,4038 |
| 7.7 | 338,6133 | 33,9502 | 57,3050 | 103,0286 |
| 7.14 | 339,0356 | 33,6473 | 56,8675 | 102,4002 |

continua

Tabela 6.6 – Medidas de desempenho prescritas pela rede
continuação

| Combinação. Rodada | \bar{F} | \bar{T}_f | %U | \bar{L} |
|-----------------------|---------------------------|-------------|---------|-----------|
| 8.1 | 263,8871 | 26,4908 | 53,5250 | 69,2579 |
| 8.7 | 265,5199 | 29,0548 | 54,3725 | 59,4285 |
| 8.16 | 262,3837 | 25,7507 | 54,9825 | 76,6221 |
| 8.20 | 266,8799 | 25,4431 | 53,5800 | 70,3443 |
| 9.5 | 260,7223 | 28,9472 | 53,6325 | 56,1195 |
| 9.7 | 260,7812 | 29,2746 | 53,7900 | 56,6453 |
| 9.10 | 262,0038 | 28,8339 | 53,4325 | 54,1865 |
| 9.20 | 262,9334 | 28,0744 | 53,4200 | 55,8237 |
| 10.5 | 282,6439 | 29,4001 | 57,0225 | 76,2731 |
| 10.8 | 280,4995 | 29,4919 | 56,7875 | 73,3709 |
| 10.10 | 285,3574 | 29,1292 | 56,8600 | 74,7443 |
| 10.14 | 279,7481 | 28,8494 | 57,4325 | 70,3449 |
| 11.1 | 340,1787 | 32,6169 | 56,6625 | 105,3305 |
| 11.3 | 337,3686 | 33,4030 | 57,1525 | 100,9829 |
| 11.7 | 340,2087 | 33,7682 | 56,7775 | 104,9406 |
| 11.20 | 338,2421 | 33,1196 | 57,0675 | 103,6826 |
| 12.1 | 273,3653 | 26,0281 | 54,9025 | 79,3596 |
| 12.2 | 264,3679 | 26,3041 | 55,0000 | 69,6254 |
| 12.14 | 274,7007 | 26,3276 | 54,5300 | 79,0595 |
| 12.20 | 275,2894 | 26,1912 | 54,8950 | 80,5413 |
| 13.2 | 334,6369 | 33,2181 | 56,9350 | 101,9593 |
| 13.10 | 275,9671 | 27,1812 | 53,2550 | 53,6812 |
| 13.13 | 341,2335 | 33,5098 | 56,8275 | 104,7231 |
| 13.16 | 266,6974 | 27,2592 | 53,1025 | 58,2201 |
| 14.1 | 273,3806 | 26,0301 | 54,9050 | 79,3749 |
| 14.7 | 268,7722 | 26,2981 | 55,1650 | 74,0050 |
| 14.15 | 265,9895 | 28,7124 | 53,9850 | 60,0978 |
| 14.19 | 273,8379 | 27,1075 | 54,5775 | 80,5100 |
| 15.3 | 262,9483 | 25,5572 | 53,2850 | 65,6777 |
| 15.8 | 336,1111 | 33,6837 | 57,2525 | 102,4641 |
| 15.12 | 337,8272 | 34,1405 | 57,2475 | 102,5881 |
| 15.15 | 336,5639 | 33,1003 | 57,0500 | 102,8502 |
| 16.7 | 271,9508 | 27,3364 | 54,7750 | 70,1354 |
| 16.10 | 276,9782 | 26,7098 | 54,5825 | 79,9291 |
| 16.20 | 276,2139 | 27,0498 | 54,5625 | 81,2808 |
| 17.1 | 337,3342 | 34,3929 | 56,9575 | 101,4042 |
| 17.12 | 337,8272 | 34,1405 | 57,2475 | 102,5881 |
| 17.15 | 339,2525 | 34,8081 | 57,0175 | 105,6359 |
| 17.19 | 339,1837 | 33,8601 | 56,6650 | 103,7368 |
| | Intervalo de Variação - v | | | |
| | \bar{F} | \bar{T}_f | %U | \bar{L} |
| | 86,7592 | 9,3649 | 4,6575 | 68,6206 |

Na tabela 6.6, o intervalo de variação, v , corresponde à diferença entre o maior e o menor valor de cada medida de desempenho, que foi utilizado para o cálculo do erro (tabela 6.7).

A tabela 6.7 apresenta os erros (calculados pela equação 6.1) entre os valores das medidas de desempenho resultantes de regras prescritas pela rede (tabela 6.6) e os valores das medidas de desempenho obtidos pelo simulador (apêndice F).

Tabela 6.7 – Erro (%)

| Combinação. Rodada | \bar{F} | \bar{T}_f | %U | \bar{L} |
|-----------------------|-----------|-------------|---------|-----------|
| 1.3 | 0,9143 | 1,9713 | 3,3816 | 0,7215 |
| 1.5 | 0,0217 | 1,8471 | 0,9662 | 0,2019 |
| 1.12 | 0,6723 | 2,8277 | 0,3221 | 0,8644 |
| 1.13 | 0,0910 | 21,0077 | 10,6817 | 10,4320 |
| 2.1 | 42,8684 | 51,2367 | 49,0607 | 46,8670 |
| 2.5 | 46,5293 | 53,7648 | 45,7327 | 52,2337 |
| 2.15 | 47,5164 | 37,8994 | 49,1143 | 52,5033 |
| 2.20 | 45,4173 | 56,3787 | 48,7923 | 47,8632 |
| 3.1 | 87,0340 | 65,1771 | 82,5550 | 52,4281 |
| 3.3 | 8,5804 | 14,5029 | 8,6420 | 35,2521 |
| 3.7 | 94,0441 | 79,4594 | 72,1417 | 62,2060 |
| 3.10 | 86,5641 | 88,2272 | 81,7499 | 55,6173 |
| 4.8 | 88,8165 | 91,1695 | 65,4321 | 59,4932 |
| 4.12 | 86,4012 | 75,1718 | 83,9506 | 54,5904 |
| 4.14 | 85,2375 | 84,7778 | 76,1675 | 51,1281 |
| 4.15 | 92,3185 | 66,6329 | 72,8932 | 70,6359 |
| 5.7 | 92,1090 | 79,8330 | 76,5969 | 60,1707 |
| 5.14 | 88,4284 | 81,8210 | 79,1197 | 55,2184 |
| 5.15 | 3,7499 | 1,0976 | 3,2206 | 5,5934 |
| 5.20 | 94,9991 | 84,4206 | 79,4418 | 66,3067 |
| 6.1 | 0,5281 | 3,4283 | 0,7515 | 0,6746 |
| 6.3 | 85,4082 | 58,0317 | 81,0521 | 64,3899 |
| 6.8 | 0,4198 | 0,1715 | 1,0199 | 0,3801 |
| 6.14 | 0,7499 | 0,1783 | 1,2346 | 1,1038 |
| 7.1 | 93,5265 | 69,9679 | 83,6822 | 75,5347 |
| 7.6 | 4,1754 | 13,9265 | 0,9125 | 6,7490 |
| 7.7 | 7,3933 | 12,7538 | 2,3618 | 9,5653 |
| 7.14 | 6,4694 | 15,7605 | 9,1787 | 8,3004 |

continua

continuação Tabela 6.7 – Erro (%)

| Combinação. Rodada | \bar{F} | \bar{T}_f | %U | \bar{L} |
|-----------------------|-----------|-------------|---------|-----------|
| 8.1 | 3,8353 | 16,2466 | 8,0515 | 14,2032 |
| 8.7 | 0,1265 | 0,3666 | 1,2882 | 0,4195 |
| 8.16 | 1,9145 | 21,7470 | 12,6141 | 16,1924 |
| 8.20 | 0,8227 | 25,2404 | 12,0236 | 16,7824 |
| 9.5 | 0,2369 | 0,6719 | 5,9045 | 2,2389 |
| 9.7 | 0,7308 | 2,5589 | 0,5368 | 0,1814 |
| 9.10 | 0,9739 | 3,9917 | 7,1927 | 1,3982 |
| 9.20 | 0,0572 | 4,6132 | 0,7515 | 0,7603 |
| 10.5 | 0,4999 | 0,3182 | 1,5030 | 0,9319 |
| 10.8 | 0,2646 | 0,9894 | 1,0199 | 0,5040 |
| 10.10 | 0,4401 | 0,0510 | 0,7515 | 0,4621 |
| 10.14 | 1,2160 | 1,1293 | 5,2603 | 1,5078 |
| 11.1 | 0,0826 | 2,0602 | 0,3757 | 0,0513 |
| 11.3 | 1,4979 | 1,2496 | 7,1390 | 0,8631 |
| 11.7 | 0,5588 | 3,8082 | 1,5030 | 0,5661 |
| 11.20 | 1,8323 | 8,5273 | 9,0177 | 3,1172 |
| 12.1 | 4,3075 | 3,5921 | 7,6221 | 3,6671 |
| 12.2 | 12,3558 | 3,7302 | 5,4750 | 16,7357 |
| 12.14 | 0,3751 | 3,8144 | 0,1610 | 0,1571 |
| 12.20 | 0,5994 | 8,7335 | 4,4015 | 1,9016 |
| 13.2 | 88,0248 | 74,8782 | 77,2410 | 54,9829 |
| 13.10 | 16,2882 | 13,8995 | 3,3816 | 14,4019 |
| 13.13 | 90,1742 | 74,1922 | 75,8991 | 59,1719 |
| 13.16 | 9,7964 | 19,7487 | 1,3956 | 18,1062 |
| 14.1 | 57,2585 | 78,1933 | 41,6532 | 29,3174 |
| 14.7 | 62,7189 | 65,4911 | 39,5062 | 38,6595 |
| 14.15 | 68,9423 | 58,2652 | 62,5335 | 62,5018 |
| 14.19 | 59,4266 | 64,0228 | 53,7305 | 29,4279 |
| 15.3 | 10,9166 | 9,5201 | 13,2045 | 20,0951 |
| 15.8 | 72,9844 | 56,8513 | 88,0837 | 73,2737 |
| 15.12 | 76,1918 | 74,2731 | 89,0499 | 75,1954 |
| 15.15 | 75,2236 | 47,0911 | 80,4616 | 75,2498 |
| 16.7 | 0,1202 | 0,6945 | 0,5368 | 22,1327 |
| 16.10 | 1,8504 | 3,3991 | 0,9662 | 35,7204 |
| 16.20 | 1,7956 | 3,1798 | 3,1669 | 37,9285 |
| 17.1 | 75,8558 | 81,0376 | 78,2072 | 71,0466 |
| 17.12 | 78,2283 | 76,2519 | 85,9903 | 77,3696 |
| 17.15 | 76,8170 | 61,2763 | 83,4138 | 78,4080 |
| 17.19 | 78,7483 | 62,6408 | 76,9189 | 79,1864 |

A coluna Combinação.Rodada da tabela 6.7, como comentado anteriormente, serve apenas para fazer referência aos valores das medidas

do simulador. Por exemplo, a primeira linha da tabela 6.7 deve ser interpretada da forma a seguir. Na coluna Combinação.Rodada, 1.3 refere-se à rodada 3 combinação 1 do simulador. Pela tabela 6.5 observa-se que as regras prescritas pela rede neural para essa rodada foram: M1:slack, M2:slack, M3:slack, M4:spt e M5:fifo. Essas regras resultaram num erro de menos de 1% para o tempo médio de permanência por lote no sistema, \bar{F} , e para o atraso médio por lote, \bar{L} . Para o tempo médio de espera por lote em filas, \bar{T}_f , o erro foi menor que 2%. Para a porcentagem de utilização média das máquinas, %U, o erro foi de 3,38%.

Foi calculada também a média do erro sobre as quatro medidas de desempenho para cada rodada da tabela 6.7. A média é apenas uma indicação do erro sobre as quatro medidas. Os resultados são apresentados na tabela 6.8, por ordem crescente de erro.

Tabela 6.8 – Média do Erro (%)

| Combinação. Rodada | Média do Erro (%) |
|-----------------------|-------------------|
| 10.20 | 0,4261 |
| 6.18 | 0,4978 |
| 8.17 | 0,5502 |
| 11.11 | 0,6424 |
| 10.18 | 0,6944 |
| 1.15 | 0,7592 |
| 10.15 | 0,8132 |
| 6.24 | 0,8166 |
| 9.17 | 1,0019 |
| 12.24 | 1,1269 |
| 1.22 | 1,1716 |
| 6.11 | 1,3456 |
| 9.30 | 1,5455 |
| 11.17 | 1,6090 |
| 1.13 | 1,7471 |
| 9.15 | 2,2630 |
| 10.24 | 2,2783 |
| 11.13 | 2,6874 |
| 9.20 | 3,3891 |
| 5.25 | 3,4153 |

continua

Tabela 6.8 – Média do Erro (%)
continuação

| Combinação. Rodada | Média do Erro (%) |
|-----------------------|-------------------|
| 12.30 | 3,9090 |
| 12.11 | 4,7972 |
| 11.30 | 5,6236 |
| 16.17 | 5,8710 |
| 7.16 | 6,4408 |
| 7.17 | 8,0185 |
| 12.12 | 9,5741 |
| 7.24 | 9,9272 |
| 16.20 | 10,4840 |
| 1.23 | 10,5531 |
| 8.11 | 10,5841 |
| 16.30 | 11,5177 |
| 13.20 | 11,9928 |
| 13.26 | 12,2617 |
| 8.26 | 13,1170 |
| 15.13 | 13,4340 |
| 8.30 | 13,7172 |
| 3.13 | 16,7443 |
| 2.25 | 46,7583 |
| 2.11 | 47,5082 |
| 2.15 | 49,5651 |
| 2.30 | 49,6128 |
| 14.17 | 51,5939 |
| 14.11 | 51,6056 |
| 14.29 | 51,6519 |
| 14.25 | 63,0607 |
| 15.25 | 69,5065 |
| 3.11 | 71,7985 |
| 6.13 | 72,2204 |
| 15.18 | 72,7982 |
| 13.12 | 73,7817 |
| 4.24 | 74,3277 |
| 17.29 | 74,3736 |
| 13.23 | 74,8593 |
| 17.25 | 74,9787 |
| 4.22 | 75,0285 |
| 4.25 | 75,6201 |
| 5.24 | 76,1468 |
| 4.18 | 76,2278 |
| 17.11 | 76,5368 |
| 3.17 | 76,9628 |
| 5.17 | 77,1774 |
| 3.20 | 78,0396 |

continua

Tabela 6.8 – Média do Erro (%)
continuação

| Combinação. Rodada | Média do Erro (%) |
|-----------------------|-------------------|
| 15.22 | 78,6775 |
| 17.22 | 79,4600 |
| 7.11 | 80,6778 |
| 5.30 | 81,2920 |

Os valores das medidas de desempenho resultantes das regras prescritas pela rede (tabela 6.6), para vários padrões, foram menores que os valores das medidas de desempenho obtidos pelo simulador (apêndice F). Dessa forma, também foi calculado o erro relativo, considerando se a diferença entre os valores das medidas de desempenho prescritos pela rede e os valores das medidas de desempenho do simulador foi positiva ou negativa.

$$Erro\ relativo = \frac{mr - ms}{v} , \quad \text{onde} \quad (6.2)$$

mr = valor da medida de desempenho prescrito pela rede

ms = valor da medida de desempenho obtido pelo simulador

v = variação da medida de desempenho da rede

A tabela 6.9 apresenta os erros relativos (calculados pela equação 6.2) entre os valores das medidas de desempenho resultantes de regras prescritas pela rede e os valores das medidas de desempenho obtidos pelo simulador.

Tabela 6.9 – Erro Relativo (%)

| Combinação. Rodada | \bar{F} | \bar{T}_f | %U | \bar{L} |
|-----------------------|-----------|-------------|----------|-----------|
| 1.3 | -0,9143 | -1,9713 | 3,3816 | -0,7976 |
| 1.5 | 0,0217 | -1,8471 | -0,96618 | 0,2232 |
| 1.12 | 0,6723 | -2,8277 | 0,3220 | 0,95572 |
| 1.13 | 0,0909 | 21,0077 | -10,6817 | -11,5337 |
| 2.1 | -42,8684 | -51,2367 | -49,0607 | -51,8166 |
| 2.5 | -46,5293 | -53,7648 | -45,7327 | -57,7500 |
| 2.15 | -47,5164 | -37,8994 | -49,1143 | -58,0480 |
| 2.20 | -45,4173 | -56,3787 | -48,7923 | -52,9179 |
| 3.1 | 87,0339 | 65,1770 | 82,5550 | 57,9649 |
| 3.3 | 8,5804 | 14,5028 | -8,6419 | -38,975 |
| 3.7 | 94,0440 | 79,4594 | 72,1417 | 68,7755 |
| 3.10 | 86,5641 | 88,2271 | 81,7498 | 61,4909 |
| 4.8 | -88,8165 | -91,1695 | -65,4321 | -65,7762 |
| 4.12 | -86,4012 | -75,1718 | -83,9506 | -60,3556 |
| 4.14 | -85,2375 | -84,7778 | -76,1675 | -56,5276 |
| 4.15 | -92,3185 | -66,6329 | -72,8932 | -78,0956 |
| 5.7 | 92,1090 | 79,8329 | 76,5968 | 66,5252 |
| 5.14 | 88,4284 | 81,8210 | 79,1197 | 61,0498 |
| 5.15 | 3,7499 | 1,0976 | -3,2206 | 6,1840 |
| 5.20 | 94,9991 | 84,4205 | 79,4417 | 73,3092 |
| 6.1 | -0,5280 | 3,4283 | -0,7514 | -0,7458 |
| 6.3 | 85,4081 | 58,0317 | 81,0520 | 71,1900 |
| 6.8 | -0,4198 | 0,1715 | 1,0198 | -0,4202 |
| 6.14 | 0,7499 | 0,1783 | 1,2345 | 1,2204 |
| 7.1 | -93,5265 | -69,9679 | -83,6822 | -83,5118 |
| 7.6 | -4,1754 | -13,9265 | -0,9125 | -7,4617 |
| 7.7 | -7,3932 | -12,7538 | 2,3617 | -10,5755 |
| 7.14 | -6,4693 | -15,7605 | -9,1787 | -9,1770 |
| 8.1 | -3,8352 | -16,2466 | -8,0515 | 15,7031 |
| 8.7 | 0,1264 | -0,3666 | -1,2882 | 0,4638 |
| 8.16 | -1,9145 | -21,747 | 12,6140 | 17,9025 |
| 8.20 | -0,82272 | -25,2404 | -12,0236 | 18,5547 |
| 9.5 | -0,2368 | 0,6718 | 5,9044 | -2,4753 |
| 9.7 | 0,7308 | -2,5588 | 0,5367 | -0,2005 |
| 9.10 | 0,9739 | 3,9917 | -7,1927 | -1,5458 |
| 9.20 | 0,0572 | -4,6131 | 0,7514 | -0,8406 |
| 10.5 | -0,4998 | 0,3181 | 1,5029 | -1,0303 |
| 10.8 | 0,2646 | 0,9894 | -1,0198 | 0,5572 |
| 10.10 | -0,4401 | 0,0509 | 0,7514 | -0,5109 |
| 10.14 | -1,2159 | -1,1293 | 5,2603 | -1,6670 |
| 11.1 | 0,0826 | 2,0601 | -0,3757 | 0,0567 |
| 11.3 | -1,4978 | -1,2496 | 7,1390 | -0,9542 |
| 11.7 | 0,5587 | 3,8081 | -1,5029 | 0,6258 |
| 11.20 | -1,8323 | -8,5272 | 9,0177 | -3,4464 |

continua

continuação

Tabela 6.9 – Erro Relativo (%)

| Combinação. Rodada | \bar{F} | \bar{T}_f | %U | \bar{L} |
|-----------------------|-----------|-------------|----------|-----------|
| 12.1 | -4,3075 | -3,5921 | 7,6221 | -4,0543 |
| 12.2 | -12,3558 | 3,7302 | 5,4750 | -18,5031 |
| 12.14 | -0,3750 | -3,8143 | -0,1610 | -0,1737 |
| 12.20 | 0,5993 | -8,7335 | 4,4015 | 2,1024 |
| 13.2 | 88,0248 | 74,8781 | 77,2410 | 60,7895 |
| 13.10 | 16,2882 | 13,8995 | 3,3816 | -15,9228 |
| 13.13 | 90,1742 | 74,1922 | 75,8990 | 65,4209 |
| 13.16 | 9,7963 | 19,7486 | -1,3956 | -20,0184 |
| 14.1 | -57,2585 | -78,1933 | -41,6532 | -32,4135 |
| 14.7 | -62,7189 | -65,4911 | -39,5062 | -42,7422 |
| 14.15 | -68,9423 | -58,2652 | -62,5335 | -69,1024 |
| 14.19 | -59,4266 | -64,0228 | -53,7305 | -32,5358 |
| 15.3 | -10,9166 | -9,5200 | 13,2045 | 22,2172 |
| 15.8 | 72,9844 | 56,8512 | 88,0837 | 81,0119 |
| 15.12 | 76,1918 | 74,2731 | 89,0499 | 83,1366 |
| 15.15 | 75,2235 | 47,0911 | 80,4616 | 83,1967 |
| 16.7 | -0,1201 | 0,6945 | -0,5367 | 24,4700 |
| 16.10 | 1,8503 | -3,3991 | -0,9661 | 39,4927 |
| 16.20 | 1,7956 | 3,1797 | -3,1669 | 41,9339 |
| 17.1 | 75,8557 | 81,0375 | 78,2071 | 78,5496 |
| 17.12 | 78,2283 | 76,2519 | 85,9903 | 85,5404 |
| 17.15 | 76,8169 | 61,2763 | 83,4138 | 86,6884 |
| 17.19 | 78,7482 | 62,6407 | 76,9189 | 87,5490 |

Na tabela 6.9, valores negativos de erro significam que os valores das medidas de desempenho resultantes de regras prescritas pela rede foram menores que os obtidos pelo simulador.

A tabela 6.10 apresenta a média do erro relativo, para cada rodada da tabela 6.9. A média é uma indicação do erro relativo sobre as quatro medidas de desempenho. Os resultados são apresentados em ordem crescente de erro.

Tabela 6.10 – Média do Erro Relativo (%)

| Combinação. Rodada | Média do Erro Relativo (%) |
|-----------------------|-------------------------------|
| 7.11 | -82,6721 |
| 4.18 | -77,7986 |
| 4.25 | -77,4850 |
| 4.22 | -76,4698 |
| 4.24 | -75,6776 |
| 14.25 | -64,7109 |
| 14.17 | -52,6146 |
| 14.29 | -52,4289 |
| 14.11 | -52,3796 |
| 2.15 | -50,9442 |
| 2.30 | -50,8765 |
| 2.11 | -48,7456 |
| 2.25 | -48,1445 |
| 7.24 | -10,1464 |
| 7.17 | -7,0901 |
| 7.16 | -6,6190 |
| 3.13 | -6,1334 |
| 12.12 | -5,4134 |
| 8.30 | -4,8830 |
| 8.11 | -3,1075 |
| 11.30 | -1,1970 |
| 9.30 | -1,1612 |
| 12.24 | -1,1310 |
| 12.11 | -1,0829 |
| 9.20 | -0,9432 |
| 1.15 | -0,6420 |
| 12.30 | -0,4075 |
| 9.17 | -0,3729 |
| 1.23 | -0,2791 |
| 8.17 | -0,2661 |
| 1.22 | -0,2194 |
| 1.13 | -0,0754 |
| 10.20 | -0,0371 |
| 10.15 | 0,0727 |
| 6.18 | 0,0878 |
| 10.18 | 0,1978 |
| 10.24 | 0,3120 |
| 6.11 | 0,3507 |
| 11.11 | 0,4559 |
| 6.24 | 0,8458 |
| 11.13 | 0,8593 |
| 11.17 | 0,8724 |
| 9.15 | 0,9660 |
| 8.26 | 1,7137 |

continua

Tabela 6.10 – Média do Erro Relativo (%)
continuação

| Combinação. Rodada | Média do Erro Relativo (%) |
|-----------------------|-------------------------------|
| 5.25 | 1,9527 |
| 13.26 | 2,0327 |
| 15.13 | 3,7462 |
| 13.20 | 4,4116 |
| 16.17 | 6,1269 |
| 16.20 | 9,2444 |
| 16.30 | 10,9356 |
| 15.25 | 71,4932 |
| 3.11 | 73,1827 |
| 6.13 | 73,9204 |
| 15.18 | 74,7328 |
| 13.12 | 75,2333 |
| 13.23 | 76,4216 |
| 17.29 | 76,4642 |
| 17.25 | 77,0489 |
| 5.24 | 77,6047 |
| 17.11 | 78,4125 |
| 3.17 | 78,6051 |
| 5.17 | 78,7660 |
| 3.20 | 79,5080 |
| 15.22 | 80,6628 |
| 17.22 | 81,5027 |
| 5.30 | 83,0426 |

A primeira linha da tabela 6.10 é interpretada da seguinte forma: os valores das medidas de desempenho resultantes das regras prescritas pela rede foram em média, 48% menores que os valores das medidas de desempenho obtidos do simulador.

As conclusões do trabalho são apresentadas a seguir.

Capítulo 7

Conclusões

Neste trabalho foi apresentado um procedimento de solução da programação da produção através de um modelo híbrido de simulação de sistemas e de redes neurais artificiais. O objetivo é analisar a diferenciação entre um conjunto de combinações de regras de prioridade designadas às máquinas em um ambiente de produção através da avaliação de quatro medidas de desempenho.

Primeiramente foi feita a análise dos valores das medidas de desempenho obtidos da simulação (apêndice F), correspondendo à aplicação dos passos 1, 2 e 3 do procedimento de solução do modelo híbrido apresentado no capítulo 5.

Foi calculada a média (μ) e o desvio padrão (σ) das 20 rodadas de simulação para cada uma das 17 combinações de regras, em relação a cada medida de desempenho. Os resultados estão apresentados na tabela 7.1.

Tabela 7.1 – Média e desvio padrão das medidas da simulação

| Combinação | | \bar{F} | \bar{T}_f | %U | \bar{L} |
|------------------------------|----------|-----------|-------------|---------|-----------|
| combinação 1 | μ | 270,4403 | 26,38161 | 54,8675 | 75,7063 |
| slack slack slack slack fifo | σ | 3,7752 | 0,4050 | 0,1647 | 4,2962 |
| combinação 2 | μ | 307,3365 | 32,9645 | 56,3763 | 93,7385 |
| fifo fifo fifo fifo fifo | σ | 3,8405 | 0,6466 | 0,1721 | 2,3472 |
| combinação 3 | μ | 261,9272 | 26,1310 | 53,2647 | 65,9873 |
| spt spt spt spt fifo | σ | 3,6884 | 0,5701 | 0,1589 | 2,7349 |
| combinação 4 | μ | 337,3744 | 34,0308 | 57,0950 | 101,9877 |
| edd edd edd edd fifo | σ | 4,2460 | 0,4823 | 0,1758 | 1,1562 |
| combinação 5 | μ | 261,1936 | 25,9734 | 53,2962 | 65,4447 |
| stt stt stt fifo fifo | σ | 4,6292 | 0,5451 | 0,1789 | 2,7026 |
| combinação 6 | μ | 261,7065 | 28,6933 | 53,4997 | 55,9458 |
| fifo spt fifo fifo fifo | σ | 3,7147 | 0,6493 | 0,2022 | 2,2874 |
| combinação 7 | μ | 342,8282 | 35,3876 | 57,2297 | 108,4761 |
| fifo edd fifo fifo fifo | σ | 3,2794 | 0,5795 | 0,1186 | 1,3343 |

continua

Tabela 7.1 – Média e desvio padrão das medidas da simulação
continuação

| Combinação | | \bar{F} | \bar{T}_f | %U | \bar{L} |
|---------------------------|----------|-----------|-------------|---------|-----------|
| combinação 8 | μ | 266,8756 | 28,5464 | 54,0786 | 59,5767 |
| fifo slack fifo fifo fifo | σ | 2,8830 | 0,6731 | 0,2049 | 2,8245 |
| combinação 9 | μ | 261,5750 | 28,6951 | 53,5202 | 56,7535 |
| fifo stt fifo fifo fifo | σ | 3,7418 | 0,6326 | 0,1981 | 1,9325 |
| combinação 10 | μ | 282,2610 | 29,4911 | 57,0475 | 74,2046 |
| spt fifo spt spt fifo | σ | 3,9191 | 0,5422 | 0,2189 | 2,7179 |
| combinação 11 | μ | 337,9131 | 33,4289 | 56,7561 | 103,7528 |
| spt edd spt spt fifo | σ | 3,9441 | 0,4506 | 0,1266 | 1,5520 |
| combinação 12 | μ | 274,3255 | 26,8195 | 54,6773 | 82,3529 |
| spt slack spt spt fifo | σ | 11,9541 | 0,5561 | 0,1957 | 4,0244 |
| combinação 13 | μ | 261,6288 | 26,1413 | 53,2280 | 65,7005 |
| spt stt spt spt fifo | σ | 3,5961 | 0,4833 | 0,1332 | 2,9021 |
| combinação 14 | μ | 323,9564 | 33,0263 | 56,9815 | 100,5918 |
| edd fifo edd edd fifo | σ | 4,1778 | 0,6960 | 0,1878 | 2,0830 |
| combinação 15 | μ | 269,0289 | 27,4885 | 53,1903 | 51,1651 |
| edd spt edd edd fifo | σ | 10,7486 | 0,5786 | 0,2319 | 2,6129 |
| combinação 16 | μ | 270,0779 | 26,7250 | 54,5570 | 52,5593 |
| edd slack edd edd fifo | σ | 3,6568 | 0,4214 | 0,1597 | 2,4686 |
| combinação 17 | μ | 271,5144 | 27,3483 | 53,2030 | 51,2474 |
| edd stt edd edd fifo | σ | 3,6308 | 0,6422 | 0,2426 | 2,4307 |

Pela tabela 7.1, observa-se que as combinações 15, 16 e 17, apresentaram em relação à média, μ , o menor atraso médio por lote, \bar{L} . Essas combinações utilizam a regra EDD para M1, M3 e M4 e variam em M2, a máquina gargalo do sistema, as regras SPT, SLACK e STT. Este fato confirma a literatura (CONWAY, 1965), já que a regra EDD está relacionada à data de entrega das peças e portanto é indicada para minimizar medidas de desempenho relacionadas ao atraso. É interessante notar que as combinações 4 e 14, que aplicam respectivamente as regras EDD e FIFO para M2, não obtiveram bons resultados em relação à \bar{L} .

Para o tempo médio de permanência por lote no sistema, \bar{F} , as combinações que resultaram em menores valores foram: 3, 5, 6, 9 e 13. Essas combinações ou utilizam as regras SPT e STT para a maioria das

máquinas ou as utilizam na máquina gargalo, M2. A regra STT seleciona da fila a peça com menor soma entre o tempo de processamento e de *setup*. A regra SPT seleciona a peça com menor tempo de processamento, e é indicada para minimizar justamente o tempo médio de permanência no sistema, \bar{F} (CONWAY et al., 1967). Portanto, os resultados obtidos confirmam a literatura. Para a medida \bar{F} não houve diferença nos resultados em aplicar as regras SPT ou STT. As combinações 6 e 9 além de minimizar \bar{F} , também produziram bons resultados para o atraso médio dos lotes \bar{L} . Essas combinações aplicam as regras SPT ou STT somente na máquina gargalo, M2, utilizando a regra FIFO para as demais máquinas.

Em relação ao tempo médio de espera por lote em filas, \bar{T}_f , os melhores resultados foram observados nas combinações 3, 5 e 13 e também 1, 12 e 16. O primeiro conjunto utiliza as regras SPT e/ou STT. O segundo conjunto aplica a regra SLACK para a máquina gargalo, M2, variando com SPT ou EDD ou SLACK para as demais. O tempo médio de espera por lote em filas é uma medida do congestionamento do sistema, e como tal, a regra SPT também é indicada para minimizá-la (HAX e CANDEA, 1984). A regra SLACK é utilizada principalmente para atender aos prazos de entrega (HAX e CANDEA, 1984), entretanto, no sistema em análise, quando utilizada na máquina gargalo, M2, produziu bons resultados em relação à \bar{T}_f .

A porcentagem de utilização média das máquinas, %U, se manteve praticamente a mesma independente da combinação de regras utilizada, Isso pode ser observado na tabela 7.1 pelo pequeno desvio padrão da medida.

As combinações 2, 7 e 14 utilizaram a regra FIFO na maioria das máquinas ou a utilizaram na máquina gargalo, M2. A regra FIFO é independente tanto do tempo de processamento das tarefas quanto da data de entrega, e sua aplicação corresponde a uma situação comum em fábricas. Entretanto, pela tabela 7.1, observa-se que essas combinações foram as que produziram os piores resultados.

Os passos 4 a 8 do procedimento de solução do modelo híbrido englobam a fase de treinamento e teste da rede neural artificial. Os resultados obtidos da fase de treinamento correspondentes às tabelas 6.2, 6.3 e 6.4 mostram a importância do número de rodadas de simulação no desempenho do modelo de redes neurais artificiais. Somente quando o treinamento foi realizado com os 272 padrões, correspondentes às 16 rodadas de cada uma das 17 combinações de regras, que os resultados para a maioria das topologias de rede tiveram melhora significativa.

Pelas tabelas 6.2 e 6.4 observa-se que os resultados obtidos com a taxa de aprendizagem, η , igual a 0,2, foram melhores que os obtidos com $\eta=0,8$. Este último provocou muita oscilação e levou até à não convergência da rede, confirmando a literatura (LOESCH e SARI, 1996).

Inicialmente, o número de neurônios na camada interna foi determinado de acordo com o teorema de *Kolmogorov* (capítulo 5). Entretanto, este teorema não se mostrou eficaz. Os resultados das tabelas 6.2, 6.3 e 6.4 mostraram que, em geral, topologias maiores de rede, com mais neurônios na camada interna geram resultados melhores. Como comentado no capítulo 4, a adição de mais neurônios à camada interna cria mais limites para possíveis soluções através da melhor codificação dos dados de entrada.

A taxa de aprendizagem, η , e o número de neurônios na camada interna foram os parâmetros da rede neural que tiveram maior impacto nos resultados. Como a determinação dos parâmetros de uma rede neural é realizada de forma empírica, a experiência realizada neste trabalho mostrou que aqueles dois parâmetros devem ser determinados e avaliados cuidadosamente, antes de qualquer outro parâmetro.

A rede neural utilizada para realizar o teste de generalização foi a rede número 1 da tabela 6.4. Esta rede apresentou um erro de treinamento de 3,3% e de teste de validação de 13,0%. Comparando com a literatura, esses resultados são considerados bastante satisfatórios. CHEN e YIH (1996) consideraram como aceitável para o treinamento um erro de 4%, e de

12% para os testes. SAAD (1996) considerou aceitável para o treinamento um erro de até 7% e para testes de até 17%.

As saídas do teste de generalização (passo 8 do procedimento de solução) são as regras de prioridade prescritas pela rede neural para cada máquina. Pela tabela 6.5, observa-se que a rede realizou somente uma prescrição inviável, designando a regra STT para M4, na combinação 16 rodada 6. Este fato pode ser atribuído à decodificação da saída da rede. A decodificação consiste em transformar as saídas da rede (números reais) em números inteiros e associar a regra correspondente. Para todos os padrões testados, a rede prescreveu a regra FIFO para M5 (lavar), como esperado.

Os passos 9 e 10 do procedimento de solução do modelo híbrido correspondem à verificação no simulador das regras prescritas pela rede e à comparação entre os valores das medidas de desempenho resultantes das regras prescritas e os valores das medidas de desempenho obtidos pelo simulador. A comparação é feita através dos cálculos dos erros apresentados nas tabelas 6.7 a 6.10.

No capítulo 6, a tabela 6.7 apresenta o erro percentual entre os valores das medidas prescritas e os valores das medidas do simulador. Este erro é calculado de acordo com a equação 6.1. A tabela 7.2 apresenta a análise sumarizada da tabela 6.7. Foram escolhidos para análise dois níveis de erro: abaixo de 20% e abaixo de 10%. A tabela 7.2 mostra a porcentagem de padrões do teste de generalização com aqueles dois níveis de erro.

Tabela 7.2 – Porcentagem de padrões x Erros

| Erro abaixo de | \bar{F} | \bar{T}_f | %U | \bar{L} |
|----------------|-----------|-------------|-------|-----------|
| 20% | 56,7% | 52,2% | 56,7% | 50,7% |
| 10% | 52,2% | 41,7% | 50,7% | 40,3% |

Na tabela 7.2, em relação à medida \bar{F} , tempo médio de permanência por lote no sistema, a seguinte interpretação deve ser feita: 56,7% dos

padrões testados tiveram erro abaixo de 20%, e 52,2% dos padrões possuíram erro inferior a 10%. A mesma interpretação deve ser feita para as demais medidas.

A tabela 6.8, no capítulo 6, apresenta a média do erro percentual sobre as quatro medidas de desempenho. A análise desta tabela mostra que 56,7% dos padrões testados possuíram um erro médio inferior a 20%, e 41,7% dos padrões tiveram erro médio inferior a 10%. No modelo desenvolvido por SAAD (1996), um erro médio de até 20% foi obtido por 28,7% da amostra, e um erro médio de até 10% foi obtido por 9,6% das amostras. Isso equivale a dizer que, a capacidade do modelo híbrido em realizar a diferenciação entre regras de prioridade foi eficiente para 56,7% da amostra testada, o que pode ser considerado bastante satisfatório se comparado à literatura (SAAD, 1996).

A tabela 6.9 apresenta o erro relativo percentual (calculado pela equação 6.2) entre os valores das medidas de desempenho resultantes das regras prescritas pela rede neural e os valores das medidas obtidos pelo simulador. Como comentado no capítulo 6, o erro relativo considera a diferença algébrica entre os valores das medidas de desempenho.

A tabela 7.3 apresenta a análise sumarizada da tabela 6.9. Na tabela 7.3, a diferença algébrica negativa significa que os valores das medidas de desempenho prescritos pela rede foram menores que os valores obtidos do simulador. A diferença algébrica “zero” significa que os valores das medidas de desempenho prescritos pela rede foram bem próximos os valores obtidos do simulador (erro percentual próximo a zero).

Tabela 7.3 – Porcentagem de padrões x Diferença algébrica

| Diferença Algébrica | \bar{F} | \bar{T}_f | %U | \bar{L} |
|---------------------|-----------|-------------|-----|-----------|
| Negativa | 49% | 49% | 47% | 52% |
| “Zero” | 18% | 10% | 6% | 9% |

Na tabela 7.3, em relação à medida \bar{F} , tempo médio de permanência por lote no sistema, a seguinte interpretação deve ser feita: em 49% dos padrões testados, o valor de \bar{F} resultante de regras prescritas pela rede, foi menor que o dado pelo simulador. Para 18% dos padrões, o valor de \bar{F} prescrito pela rede foi praticamente idêntico ao obtido pelo simulador. A mesma interpretação deve ser feita para as demais medidas.

Pela tabela 7.3, observa-se que a porcentagem de utilização média das máquinas, %U, foi a medida de desempenho que apresentou a menor diferença algébrica negativa (47%). Isso equivale a dizer, em relação a %U, 53% dos padrões testados apresentaram valores prescritos pela rede maiores que os obtidos pelo simulador. Este resultado também pode ser considerado bastante satisfatório, pois como comentado no capítulo 6, a utilização intensiva e eficiente de um recurso, aumenta a produtividade do sistema, reduz a necessidade de altos níveis de estoque de segurança e do emprego de horas extras (HAX e CANDEA, 1984).

A tabela 6.10, no capítulo 6, apresenta a média do erro relativo percentual sobre as quatro medidas de desempenho. A análise desta tabela mostra que para 49% dos padrões testados, os valores das medidas de desempenho prescritos pela rede foram menores que os valores obtidos pelo simulador (média do erro relativo percentual negativa). Para 15% dos padrões testados os valores das medidas de desempenho resultantes de regras prescritas pela rede foram muito próximos aos do simulador (média do erro relativo percentual próxima a zero).

SAAD (1996) obteve médias de erro relativo percentual negativas para 65% da amostra testada. Comparado à literatura (SAAD, 1996), o valor obtido neste trabalho também pode ser considerado satisfatório, pois como abordado anteriormente, não é interessante que o valor da porcentagem de utilização média das máquinas, %U, seja minimizado.

Segundo LOESCH e SARI (1996), a rede neural *back propagation* é utilizada principalmente para problemas de reconhecimento e de classificação de padrões. Neste trabalho foram analisadas 17 combinações de regras de prioridade às máquinas, equivalentes a 17 classes. Considerando que as combinações de regras não diferiam muito entre si, pois foram realizadas variações principalmente sobre a máquina gargalo do sistema, os resultados obtidos de um modo geral podem ser considerados bastante satisfatórios.

Este trabalho apresentou o potencial de aplicação de um modelo híbrido de simulação de sistemas e de redes neurais artificiais na programação da produção. Os resultados mostraram a capacidade de generalização do modelo híbrido em prescrever regras de prioridade às máquinas a partir de valores de medidas de desempenho estabelecidos pelo usuário.

A aplicação do procedimento de solução do modelo híbrido de simulação de sistemas e de redes neurais artificiais em um ambiente produtivo real deve obedecer à seguinte ordem:

- Estabelecer as medidas de desempenho convenientes para avaliar a programação da produção;
- Estabelecer as regras de prioridade viáveis de serem implantadas no ambiente produtivo;
- Estabelecer uma amostra de combinações de regras de prioridade às máquinas;
- Simular o funcionamento do ambiente de produção;
- Separar os resultados da simulação em dois conjuntos de dados (um para o treinamento da rede neural artificial e outro para os testes de generalização);
- Realizar o treinamento da rede neural;
- Realizar os testes de generalização da rede neural e obter as regras de prioridade prescritas para cada máquina;

- Verificar os resultados da rede neural no simulador, ou seja, simular novamente o ambiente de produção segundo as regras prescritas pela rede;
- Analisar a capacidade de generalização do modelo através da comparação entre os valores das medidas de desempenho prescritos pela rede e os obtidos anteriormente na simulação;
- Implantar as regras de prioridade designadas às máquinas pela rede neural no ambiente produtivo.

Tendo como base este trabalho, os seguintes pontos poderiam ser investigados em pesquisas futuras:

- Utilizar uma segunda rede neural para realizar uma pré-classificação dos valores das medidas de desempenho antes de apresentá-los à rede *back propagation*. O trabalho de FERREIRA (1998) é uma boa referência sobre o assunto;
- Aumentar o número de combinações de regras de prioridade analisadas para avaliar as implicações nos resultados;
- Analisar todas as 500 possíveis combinações de regras designadas às máquinas, para estabelecer qual a melhor combinação teórica. Entretanto, a melhor combinação teórica poderia implicar em aumentos substanciais de investimentos, como por exemplo, alterações significativas no ambiente de produção para contemplá-la;
- Aplicar o procedimento de solução do modelo híbrido de simulação de sistemas e de redes neurais artificiais em um ambiente produtivo real.

Referências Bibliográficas

ARAÚJO, A. **Redes neurais artificiais**. São Carlos : EESC-USP, 1996. 28 p. Apostila.

ARIZONO, I., YAMAMOTO, A., OHTA, H. Scheduling for minimizing total actual flow by neural networks. **International Journal of Production Research**, v. 30, n. 3, p. 503-512, 1992.

BEALE, R., JACKSON, T. **Neural computing: an introduction**. Bristol and New York : Adam Hilger, 1990. 240 p.

BISHOP, C.M. **Neural networks for pattern recognition**. 4.ed. Oxford : Oxford University Press, 1995. 482 p.

BLAZEWICZ, J., DOMSCHKE, W., PESCH, E. The job shop sequencing problem: conventional and new solution techniques. **European Journal of Operational Research**, v. 93, n. 1, p. 1-33, 1996.

BUFFA, E.S., MILLER, J.G. **Production-inventory systems: planning and control**. Homewood : Richard D. Irwin Inc., 1979. 744 p.

BURBIDGE, J.L. **Planejamento e Controle da Produção**. 12.ed. São Paulo : Atlas, 1983. 555 p.

CARVALHO, A.P.L. Redes Neurais Artificiais. Página da Internet. Endereço: <http://www.icmsc.sc.usp.br/%7Eandre/>. Acessado em 27/01/2000.

CHEN, C.C., YIH, Y. Identifying attributes for knowledge-based development in dynamic scheduling environments. **International Journal of Production Research**, v. 34, n. 6, p. 1739-1755, 1996.

CHRYSSOLOURIS, G. et al. Use of neural networks for the design of manufacturing systems. **Manufacturing Review**, v. 3, n. 3, p. 187-194, 1990.

CONWAY, R.W. Priority dispatching and job lateness in a job shop. **Journal of Industrial Engineering**, v. 16, n. 4, p. 228-237, 1965.

CONWAY, R.W., MAXWELL, W.L., MILLER, L.W. **Theory of scheduling**. Reading : Addison-Wesley, 1967. 294 p.

CORSTEN, H., MAY, C. Artificial neural networks for supporting production planning and control. **Technovation**, v. 16, n. 2, p. 67-76, 1996.

COSTA, J.A.F. **Sistema de reconhecimento de padrões visuais invariante a transformações geométricas utilizando redes neurais artificiais de múltiplas camadas**. São Carlos, 1996. 157p. Dissertação

(Mestrado em Engenharia) - Escola de Engenharia de São Carlos, Universidade de São Paulo.

FAUSETT, L. **Fundamentals of Neural Networks**. 1. ed. New Jersey : Prentice-Hall Inc.,1994. 461 p.

FERNANDES, F.C.F. **Concepção de um sistema de controle para a manufatura celular**. São Carlos, 1991. 239p. Tese (Doutorado em Engenharia) - **Escola de Engenharia de São Carlos**, Universidade de São Paulo.

FERREIRA, S. **Um modelo simbólico-conexionista para interpretação de cenas**. São José dos Campos, SP, 1998. 106 p. Dissertação (Mestrado em Computação) – **Instituto Nacional de Pesquisas Espaciais**, Ministério da Ciência e Tecnologia.

FOO, S.Y., TAKEFUJI, Y. Stochastic neural networks for solving job-shop scheduling. Part I: Problem representation. In: 2nd INTERNATIONAL CONFERENCE ON NEURAL NETWORKS, 1988, **Proceedings...** IEEE, 1988. v. II, p. 275-282.

FOO, S.Y., TAKEFUJI, Y. Stochastic neural networks for solving job-shop scheduling. Part II: Architecture and simulation. In: 2nd INTERNATIONAL CONFERENCE ON NEURAL NETWORKS, 1988, **Proceedings...** IEEE, 1988. v. II, p. 283-290.

FRENCH, S. **Sequencing and scheduling: an introduction to the mathematics of the job-shop**. 1.ed. Chichester : Ellis Horwood Limited, 1982. 245 p.

GALLANT, S.I. **Neural networks learning and expert systems**. 1.ed. Cambridge : The MIT Press, 1993. 365 p.

GARGEYA, V.B., DEANE, R.H. Scheduling research in multiple resource constrained job shops: a review and critique. **International Journal of Production Research**, v. 34, n. 8, p. 2077-2097, 1996.

GORDON, G. **System Simulation**. 2. ed. New Jersey : Prentice-Hall, 1978. 324 p.

GROOVER, M.P. **Automation, Production Systems and Computer Integrated Manufacturing**. New Jersey : Prentice-Hall, 1987. 808 p.

HANSON, S.J., BURR, D.J. What connectionist models learn: learning and representation in connectionist networks. In: MAMMONE, R.J., ZEEVI, Y. **Neural Networks: theory and applications**. Piscataway : Academic Press, 1991. p. 169-208.

HAX, A.C., CANDEA, D. **Production and inventory management**. 1.ed. New Jersey : Prentice-Hall, 1984. 513 p.

HRYCEJ, T. **Modular learning in neural networks: a modularized approach to neural network classification**. New York : Jonh Wiley & Sons, 1992. 235 p.

JOHNSON, L.A., MONTGOMERY, D.C. **Operations research in production planning, scheduling, and inventory control**. 4.ed. New York : John Wiley & Sons, 1974. 525 p.

KIM, S., LEE, Y-H. Enhancement of a job sequencing rule using an artificial neural network. In: SECOND INDUSTRIAL ENGINEERING RESEARCH CONFERENCE, 1993, Los Angeles. **Proceedings...** Los Angeles : IEEE, 1993. p. 842-846.

LAW, A.M. Confidence intervals in discrete event simulation: a comparison of replication and batch means. **Naval Research Logistic Quartely**, v.24, p. 667-678, 1977.

LIPPMANN, R.P. An introduction to computing with neural nets. **IEEE ASSP Magazine**, April, p. 4-21, 1987.

LIU, H., DONG, J. Dispatching rule selection using artificial neural networks for dynamic planning and scheduling. **Journal of Intelligent Manufacturing**, v. 7, n. 3, p. 243-250, 1996.

LOESCH, C., SARI, S. **Redes neurais artificiais: fundamentos e modelos**. São José dos Campos : INPE, 125 p. Apostila.

MACCARTHY, B.L., LIU, J. Addressing the gap in scheduling reserch: a review of optimization and heuristic methods in production scheduling. **International Journal of Production Research**, v. 31, n. 1, p. 59-79, 1993.

MCCLAIN, J.O., THOMAS, L.J. Facility Design. In: _____. **Operations Management: production of goods and services**. New Jersey : Prentice-Hall, 1985. p. 108-125.

MOCCELLIN, J.V. **Técnicas de sequenciamento e programação de operações em máquinas**. São Carlos : EESC-USP, 1994. 80p. Apostila.

NEURALWARE. Pittsburgh. *Using Nworks: an extended tutorial for NeuralWorks Professional II/Plus and NeuralWorks Explorer*. Pittsburgh, 1991(a), 138 p.

NEURALWARE. Pittsburgh. *NeuralWorks Predict: complete solution for neural data modeling*. Pittsburgh, 1991(b), 165 p.

PINEDO, M. **Scheduling: theory, algorithms and systems**. New Jersey : Prentice Hall, 1995. 378 p.

PIRES, S. **Gestão estratégica da produção**. Piracicaba : UNIMEP, 1995. 269 p.

PORTMANN, M-C. Scheduling methodology: optimization and compu-search approaches I. In ARTIBA, A., ELMAGHRABY, S.E. **The planning and scheduling of production systems**. London : Chapman & Hall, 1997. p. 271-300.

PORTO, A. **Redes neurais como uma alternativa de solução para a engenharia de produção**. São Carlos : Departamento de Engenharia de Produção/UFSCar, 1995. 95 p. Trabalho de Conclusão de Curso.

PROMODEL – Manufacturing Simulation Software. User's Guide, Version 4.0. **Promodel Corporation**, Utah, USA, 1997, 645p.

SAAD, C.S. **Aplicação de um modelo de redes neurais à programação da produção**. São Paulo, 1996. 124 p. Dissertação (Mestrado em Engenharia) - **Escola Politécnica**, Universidade de São Paulo.

SABUNCUOGLU, I., GURGUN, B. A neural network model for scheduling problems. **European Journal of Operational Research**, v. 93, n. 3, p. 288-299, 1996.

SALLES, M.C.B. **Minicurso de simulação**. XXX SIMPÓSIO BRASILEIRO DE PESQUISA OPERACIONAL, 1998, Curitiba-PR. Notas de Aula.

SATAKE, T., MORIKAWA, K., NAKAMURA, N. Neural network approach for minimizing the makespan of the general job-shop. **International Journal of Production Economics**, v. 33, n. 1/3, p. 67-74, 1994.

SHARDA, R., WANG, J. Neural networks and operations research / management science. **European Journal of Operational Research**, v. 93, n. 3, p. 227-229, 1996.

SNNS – Stuttgart Neural Networks Simulator. User Manual, Version 4.1. **University of Stuttgart**, Stuttgart, Germany, 1995, 312 p.

STEPHANOPOULOS, G., HAN, C. Intelligent systems in process engineering: a review. **Computers & Chemical Engineering**, v. 20, n. 6/7, p. 743-791, 1996.

SUN, Y-L., YIH, Y. Intelligent controller for manufacturing cells. **International Journal of Production Research**, v. 34, n. 8, p. 2353-2373, 1996.

TOURE, S., RABELO, L., VELASCO, T. Artificial neural networks for flexible manufacturing systems scheduling. **Computers and Industrial Engineering**, v. 25, n. 1/4, p. 385-388, 1993.

UDO, G.D., GUPTA, Y.P. Application of neural network in manufacturing management systems. **Production Planning and Control**, v. 5, n. 3, p. 258-270, 1994.

WATANABE, H., TOKUMARU, H., HASHIMOTO, Y. Job shop scheduling using neural networks. **Control Engineering Practice**, v. 1, n. 6, p. 957-961, 1993.

WILLEMS, T.M., BRANDTS, L.E.M.W. Implementing heuristics as an optimization criterion in neural networks for job-shop scheduling. **Journal of Intelligent Manufacturing**, v. 6, n. 6, p. 377-387, 1995.

YEH, J-M., LIN, C., CHANG, K-J. Dynamical neural network approach to the job-shop scheduling. **Journal of Information & Optimization Sciences**, v. 16, n. 1, p. 17-38, 1995.

ZAHEDI, F. An introduction to neural network and a comparison with artificial intelligence and expert systems. **Interfaces**, v. 21, n. 2, p. 25-38, 1991.

ZHANG, H.C., HUANG, S.H. Applications of neural networks in manufacturing: a state-of-art survey. **International Journal of Production Research**, v. 33, n. 3, p. 705-728, 1995.

ZHOU, D.N. et al. A neural network approach to job-shop scheduling. **IEEE Transactions on Neural Networks**, v. 2, n. 1, p. 175-179, 1991.

Bibliografia

ALEXANDRE, F. et al. Compu-search methodologies II: scheduling using genetic algorithms and artificial neural networks. In: ARTIBA, A., ELMAGHRABY, S.E. **The planning and scheduling of production systems**. London : Chapman & Hall, 1997. p. 301-336.

BAKER, K.R. **Introduction to sequencing and scheduling**. New York : John Wiley & Sons, 1974. 305 p.

BANKS, J., CARSON, J.S. **Discrete-event system simulation**. New Jersey : Prentice Hall, 1984. 514 p.

BLAZEWICZ, J. et al. **Scheduling computer and manufacturing process**. Berlin : Springer, 1996. 491 p.

BURKE, L.I., IGNIZIO, J. Neural networks and operations research: an overview. **Computers & Operations Research**, v. 19, n. 3/4, p. 179-189, 1992.

CHANG, C.Y., JENG, M.D. Experimental study of a neural model for scheduling jobs shops. In: IEEE INTERNATIONAL CONFERENCE ON SYSTEMS, MAN AND CYBERNETICS, 1., 1995, Vancouver. **Proceedings...** Vancouver : IEEE, 1995. p. 536-540.

CHANG, S.H., NAM, B.H. Linear programming neural networks for job-shop scheduling. In: INTERNATIONAL JOINT CONFERENCE ON NEURAL NETWORKS, 2., 1993, Nagoya. **Proceedings...** Nagoya : IEEE, 1993. p. 1557-1560.

CLARK, K.B., HAYES, R.H. Recapturing America's manufacturing heritage. **California Management Review**, v. 30, n. 4, p. 9-33, 1988.

EMMONS, H. One machine sequencing to minimize mean flow time with minimum number tardy. **Naval Research Logistics Quarterly**, v. 22, n. 3, 1975.

FOO, S.Y., TAKEFUJI, Y. SZU, H. Scaling properties of neural networks for job-shop scheduling. **Neurocomputing**, v. 8, n. 1, p. 79-91, 1995.

GALLONE, J-M., CHARPILLET, F., ALEXANDRE, F. Anytime scheduling with neural networks. In: 1995 INRIA/IEEE SYMPOSIUM ON EMERGING TECHNOLOGIES AND FACTORY AUTOMATION, 1., 1995, Paris. **Proceedings...** Paris : IEEE, 1995. p. 509-520.

GOLDRATT, E.M., COX, J. **A Meta**. São Paulo : IMAM, 1992. 260 p.

GRAVES, S.C. A review of production scheduling. **Operations Research**, v. 29, n. 4, p. 646-675, 1981.

HARRELL, C.R. et al. **System improvement using simulation**. 3.ed. Orem : PROMODEL Corporation, 1995. 155 p.

HILLIER, F.S., LIEBERMAN, G.J. Teoria das filas. In: _____. **Introdução à pesquisa operacional**. 1.ed. Rio de Janeiro : Campus, 1988. p. 394-449.

HOFFMAN, K., KAPLINSKY, R. The point of transition - from machinofacture to systemofacture. In: _____. **Driving force: the global restructuring of technology, labour, and investment in the automobile and components industries**. London : Westview, 1988. p. 31-72.

HUAJIE, L., DONG, J. Dispatching rule selection using artificial neural networks for dynamic planning and scheduling. **Journal of Intelligent Manufacturing**, v. 7, n. 3, p. 243-250, 1996.

KIM, S., LEE, Y-H. A hybrid approach to sequencing jobs using heuristics rules and neural networks. **Production Planning and Control**, v. 6, n. 5, p. 445-453, 1995.

LAW, A.M., KELTON, W.D. **Simulation modelling and analysis**. McGraw-Hill. 2 ed. 1991.

LEE, Y-H., KIM, S. Neural network aplications for scheduling jobs on parallel machines. **Computers and Industrial Engineering**, v. 25, n. 1/4, p. 227-230, 1993.

LO, C-C., HSU, C-C. A parallel distributed processing technique for job-shop scheduling problems. In: INTERNATIONAL JOINT CONFERENCE ON NEURAL NETWORKS, 2., 1993, Nagoya. **Proceedings...** Nagoya : IEEE, 1993. p. 1602-1605.

MOCCELLIN, J.V. **Uma contribuição à programação de operações em sistemas de produção intermitente 'flow-shop'**. São Carlos, 1992. 125 p. Tese (Livre **Docência**) - Escola de Engenharia de São Carlos, Universidade de São Paulo.

MOREIRA, D.A. **Administração da produção e operações**. 1.ed. São Paulo : Pioneira, 1993. 619 p.

NAGANO, M.S. **Novos procedimentos de busca tabu para o problema de programação de operações 'flow-shop' permutacional**. São Carlos, 1995. 118 p. Dissertação (Mestrado em Engenharia) - Escola de Engenharia de São Carlos, Universidade de São Paulo.

PACHECO, R.F. **Um sistema de emissão de ordens e programação de operações para a manufatura celular**. São Carlos, 1995. 97 p. Dissertação (Mestrado em Engenharia) - Departamento de Engenharia de Produção, Universidade Federal de São Carlos.

PARKER, R.G. **Deterministic scheduling theory**. London : Chapman & Hall, 1995. 290 p.

PIZZOLATO, E.B., KIRNER, C. Ambiente de emulação de redes neurais. In: XXIV CONGRESSO NACIONAL DE INFORMÁTICA, 1991, **Local. Anais... Local : Editora**, 1991. **no.pág totais** p. 268-275.

PROMODEL CORPORATION. <http://www.promodel.com>.

SELLERS, D. A survey of approaches to the job shop scheduling problem. In: TWENTY-EIGHTH SOUTHEASTERN SYMPOSIUM ON SYSTEMS THEORY, 1994, Baton Rouge. **Proceedings...** Baton Rouge : Naraghi-Pour, M., 1994. p. 396-400.

TAFNER, M.A., XEREZ, M. de, FILHO, I.W.R. **Redes neurais artificiais: introdução e princípios de neurocomputação**. Blumenau : FURB, 1995. 199 p.

THE MATHWORKS, INC. Natick. MATLAB User's Guide. Natick, 1992(a), 213 p.

THE MATHWORKS, INC. Natick. MATLAB Reference Guide. Natick, 1992(b), 548 p.

VAITHYANATHAN, S., IGNIZIO, J.P. A stochastic neural network for resource constrained scheduling. **Computers & Operations Research**, v. 19, n. 3/4, p. 241-254, 1992.

Apêndice A

Softwares de Redes Neurais Artificiais

TABELA A1 - *Software NeuralSIM**

| | |
|---------------------------|--|
| Fabricante | NeuralWare, Inc. |
| Distribuidor Brasil | SMI - Software Marketing International – São Paulo - SP |
| Endereço | Av. Brigadeiro Faria Lima, 613 - 10 Andar Tel.: (011) 820-0388 Fax.: (011) 820-7361 E-mail: smisp@embratel.net.br |
| Preço (USA) | US\$1995,00 |
| Validação | |
| Plataformas | IBM PC e compatíveis |
| Hardware | Win 95, 98 ou NT, 8MB RAM, espaço disco: 10MB, 520KB livre, Excel 7 ou 97, CD-ROM |
| Linguagem | C |
| Estatísticas geradas | Histogramas de pesos, gráficos de erro |
| Propósito | Geral |
| Documentação | On-line e impressa |
| Verificação | |
| Arquiteturas de RNA | MLP - com 2 algoritmos de treinamento, um para dados limpos e outro para ruidosos |
| Entrada e saída dos dados | Entrada: via Excel, arquivo ASCII Saída: na tela, ou gravada em arquivos |
| Capacidade | Campos de entrada: 256/4000 |
| Funções ativação | Tangente hiperbólica, sigmoidal, seno, gaussiana, linear |
| Definição parâmetros | Limitantes, pesos, taxa de aprendizagem é ajustada automaticamente |
| Outros Recursos | Análise de sensibilidade, módulo explicativo, raciocínio baseado em casos Converte dados em C, Visual Basic e Fortran |

* Fontes:
NEURALWARE, INC. <http://www.neuralware.com>. Consultado em 27/01/2000.
NEURALWARE, INC (1991)

TABELA A2 - *Software NeuralWorks Professional II/Plus**

| | |
|---------------------------|---|
| Fabricante | NeuralWare, Inc. |
| Distribuidor Brasil | SMI - Software Marketing International – São Paulo - SP |
| Endereço | Av. Brigadeiro Faria Lima, 613 - 10 Andar Tel.: (011) 820-0388 Fax.: (011) 820-7361 E-mail: smisp@embratel.net.br |
| Preço (USA) | US\$2995,00 a US\$4995,00 Completo ¹ : US\$9995,00 a US\$14995,00 |
| Validação | |
| Plataformas | PC, SUN, RS6000, SGI |
| Hardware | PC: 640K base c/ 2MB estendida, 8MB RAM, Win 95, 98 ou NT, HD: 8MB SUN: 24MB memória, sist.oper: SunOS 4.1.3 ou sup, Solaris 1.x,2.x, HD: 11MB RS6000: memória: 24MB, sist.oper: AIX 3.2, HD: 11MB SGI: 24MB memória, sist.oper: IRIX 5.3 ou sup, HD: 11MB |
| Linguagem | C |
| Estatísticas geradas | Histogramas de pesos, gráficos de erro, de correlação, de funções de ativação |
| Propósito | Geral |
| Documentação | On-line e impressa |
| Verificação | |
| Arquiteturas | Implementa 28 arquiteturas de RNA, entre elas: Backpropagation, Kohonen, Boltzmann, Hopfield, BAM, ART. |
| Entrada e saída dos dados | Entrada: teclado, arquivo ASCII, binário, linguagem C. Saída: na tela, ou gravada em arquivos. Converte arquivos em C. |
| Capacidade | Número de UP: ilimitada |
| Funções ativação | Permite a escolha de vários tipos |
| Definição parâmetros | Taxa de aprendizagem, termo de momento, pesos, número de ciclos, etc. |
| Outros Recursos | Otimiza no. de UP na camada interna p/ backpropagation Prevenção contra overtraining Módulo explicativo Utilização com Sistemas Especialistas e Lógica Nebulosa Permite ao usuário modificar ou criar arquiteturas Compatibilidade entre plataformas através de arquivos ASCII |

¹ O Pacote completo constitui-se do NeuralWorks Professional II/Plus, NeuralSIM, Designer Pack, UDND e treinamento incluso.

* Fontes:

NEURALWARE, INC. <http://www.neuralware.com>. Consultado em 27/01/2000.

NEURALWARE, INC (1991)

TABELA A3 - *Software* NeuroShell 2*

| | |
|---------------------------|---|
| Fabricante | Ward Systems Group, Inc |
| Distribuidor Brasil | IBL – Informática Brasileira Ltda |
| Telefone | São Paulo - SP. Telefone : (011) 5561- 6972 Fax : (011) 533- 5734 |
| Preço | De US\$ 595,00 a US\$ 1395,00 |
| Validação | |
| Plataforma | PC |
| Hardware | PC: HD:8MB, 4MB RAM, Windows 95 / NT |
| Linguagem | C |
| Estatísticas geradas | Gráficos 3D, indicadores de séries temporais e financeiros |
| Propósito | Geral |
| Documentação | On-line, impressa |
| Verificação | |
| Arquiteturas de RNA | Implementa 16 arquiteturas de RNA, entre elas: MLP, Kohonen, Probabilística. |
| Entrada e saída dos dados | Entrada: teclado, arquivos ASCII, binários, planilhas Saída: na tela, arquivos ASCII e binários |
| Capacidade | Dado não disponível |
| Funções ativação | Permite a escolha de vários tipos |
| Definição parâmetros | Taxa de aprendizagem, termo de momento, pesos, número de ciclos, etc. |
| Outros Recursos | Possui módulo para iniciantes e avançado. Gera código fonte em Visual Basic ou C. Permite criar regras If/Then/Else para o pré-processamento dos dados. Prevenção contra retreinamento. Converte dados alfanuméricos e <i>strings</i> em números. |

* Fontes:

WARD SYSTEMS GROUP, INC. <http://www.wardsystems.com>. Consultado em 27/01/2000
WARD SYSTEMS GROUP, INC. Neural Network Demonstration, versão 1.2. 1 disquete 3 ½.

TABELA A4 - *Software* SNNS – Stuttgart Neural Networks Simulator

| | |
|---------------------------|--|
| Fabricante | Universidade de Stuttgart, Alemanha |
| Endereço Internet | http://inf.informatik.uni-stuttgart.de/ipvr/bv/projekte/snns/snns.html |
| Preço | <i>Freeware</i> |
| Validação | |
| Plataformas | SUN SparcSt. ELC, IPC, 2, 5, 10, 20: sistema operacional: Sun OS 4.1.2 ou superior IBM PC 80486, Pentium: sistema operacional: Linux IBM RS 6000 / 320, 320H, 530H: sistema operacional: AIX V3.1, V3.2 SGI Indigo: sistema operacional IRIX 4.0.5, 5.3 HP 9000 / 720, 730: sistema operacional HP-UX 8.0.7 |
| Linguagem | C |
| Estatísticas geradas | Gráficos de erro, de função de ativação, diagramas de pesos |
| Propósito | Geral |
| Documentação | On-line, manual |
| Verificação | |
| Arquiteturas de RNA | Implementa 17 arquiteturas de RNA entre elas: Backpropagation, BAM, Kohonen, ART |
| Entrada e saída dos dados | Entrada: teclado, arquivo ASCII Saída: na tela, arquivo ASCII |
| Funções ativação | Permite a escolha de vários tipos e definidas pelo usuário |
| Definição parâmetros | Taxa de aprendizagem, termo de momento, pesos, número de ciclos, etc. |
| Outros Recursos | Permite adicionar UP, camadas e apagá-las. Permite implementação de algoritmos de <i>simulated annealing</i> . Possui ferramentas para monitorar o desempenho da rede. Análise do impacto das UP de entrada no resultado da rede. Gera arquivos compatíveis entre as várias plataformas. |

* Fonte:

UNIVERSITY OF STUTTGART. Stuttgart. SNNS – Stuttgart Neural Networks Simulator. User Manual, Version 4.1. Stuttgart, 1995, 312 p.

Apêndice B

Programa de Seqüência de Chegada de Lotes

Este programa gera uma seqüência de chegada de lotes, com a restrição de que não poderão chegar dois lotes consecutivos do mesmo tipo de peça. Variáveis a definir: número de lotes que chegarão no sistema e número de tipos de peças.

```

uses crt;
var texto: text;
var i,m,aux,y,x: integer;

begin
  randomize;
  clrscr;
  assign(texto, 'sequencia.txt' );
  rewrite (texto);

  write('digite a qtde de lotes -> '); readln (m);
  write('digite o numero de tipos de pecas -> ');
  readln(x);

  y:=random(x); write(texto, y, ' ');
  write(y, ' ');
  aux:=y;

  for i:=2 to m do
  begin
    y:=random(x);

    while y=aux do
      y:=random (x);

    write (texto,y,' ');
    write (y, ' ');
    aux:=y;
  end;
  close(texto);
  readln;
end.

```

Apêndice C

Seqüência de Chegada dos Lotes

TABELA C1 - Seqüência de chegada dos lotes e datas de entrega

| Lote Número | Tipo de Peça | Data de Entrega (min) | Lote Número | Tipo de Peça | Data de Entrega (min) |
|-------------|--------------|-----------------------|-------------|--------------|-----------------------|
| 1 | E | 188,98 | 31 | C | 286,87 |
| 2 | C | 92,72 | 32 | D | 391,71 |
| 3 | D | 174,87 | 33 | C | 346,07 |
| 4 | E | 151,17 | 34 | B | 451,19 |
| 5 | C | 176,57 | 35 | E | 370,28 |
| 6 | E | 125,44 | 36 | D | 350,08 |
| 7 | B | 229,11 | 37 | E | 306,09 |
| 8 | A | 301,64 | 38 | C | 369,04 |
| 9 | C | 228,07 | 39 | D | 310,35 |
| 10 | D | 183,04 | 40 | B | 332,98 |
| 11 | C | 238,32 | 41 | C | 324,03 |
| 12 | A | 222,51 | 42 | E | 316,33 |
| 13 | C | 173,01 | 43 | D | 364,11 |
| 14 | B | 317,86 | 44 | C | 337,59 |
| 15 | C | 167,55 | 45 | D | 339,45 |
| 16 | E | 250,87 | 46 | A | 362,75 |
| 17 | C | 156,38 | 47 | B | 428,18 |
| 18 | E | 273,29 | 48 | E | 390,93 |
| 19 | D | 289,13 | 49 | C | 426,30 |
| 20 | A | 442,81 | 50 | A | 471,31 |
| 21 | B | 401,48 | 51 | B | 495,19 |
| 22 | A | 367,59 | 52 | C | 391,31 |
| 23 | E | 290,22 | 53 | A | 415,02 |
| 24 | C | 222,53 | 54 | C | 425,52 |
| 25 | B | 379,50 | 55 | A | 491,55 |
| 26 | E | 351,90 | 56 | C | 428,09 |
| 27 | D | 337,25 | 57 | B | 539,74 |
| 28 | B | 432,82 | 58 | A | 496,36 |
| 29 | E | 330,20 | 59 | C | 409,08 |
| 30 | D | 235,98 | 60 | A | 463,84 |

Apêndice D

Tempos de Processamento dos Lotes

| Lote 1 - Peça E - 16 peças | | | | | Lote 2 - Peça C - 8 peças | | | | |
|----------------------------|-----------|-----------------|----------|-----------|----------------------------|-----------|-----------------|----------|-----------|
| M1 | M2 | M3 | M4 | M5 | M1 | M2 | M3 | M4 | M5 |
| 0,840458 | 1,445820 | | 0,335115 | 15 seg | 0,600876 | 1,151410 | | | 15 seg |
| 1,460450 | 0,850002 | | 0,490112 | | 0,793165 | 1,352550 | | | |
| 0,927810 | 0,667789 | | 0,356952 | | 0,794979 | 1,099140 | | | |
| 1,246410 | 0,557270 | | 0,436602 | | 0,933304 | 1,056080 | | | |
| 0,844936 | 0,674955 | | 0,336234 | | 1,343690 | 1,228770 | | | |
| 1,403610 | 0,847821 | | 0,475902 | | 0,616065 | 1,483640 | | | |
| 1,121720 | 0,921236 | | 0,405431 | | 1,244990 | 1,163560 | | | |
| 0,879290 | 1,215410 | | 0,344823 | | 1,491670 | 1,067670 | | | |
| 1,474230 | 0,856505 | | 0,493558 | | | | | | |
| 0,900570 | 0,636931 | | 0,350142 | | | | | | |
| 0,653667 | 0,672084 | | 0,288417 | | | | | | |
| 1,150560 | 0,745313 | | 0,412639 | | | | | | |
| 1,195230 | 0,797415 | | 0,423808 | | | | | | |
| 0,930383 | 1,436280 | | 0,357596 | | | | | | |
| 1,273440 | 1,157500 | | 0,443359 | | | | | | |
| 0,947081 | 0,502271 | | 0,361770 | | | | | | |
| | | | | pi lote | | | | | pi lote |
| | | | | 37,796907 | | | | | 17,671559 |
| 17,249845 | 13,984602 | | 6,312460 | 0,250000 | 7,818739 | 9,602820 | | | 0,250000 |
| | | pi remanescente | | | | | pi remanescente | | |
| 37,796907 | 20,547062 | 6,562460 | 6,562460 | 0,250000 | 17,671559 | 17,671559 | 9,852820 | 0,250000 | 0,250000 |
| Lote 3 - Peça D - 15 peças | | | | | Lote 4 - Peça E - 11 peças | | | | |
| M1 | M2 | M3 | M4 | M5 | M1 | M2 | M3 | M4 | M5 |
| 0,787224 | 0,558666 | | 0,486456 | | 1,264350 | 0,835122 | | 0,250568 | 15 seg |
| 1,051650 | 0,787023 | | 0,337500 | | 0,768103 | 0,793921 | | 0,275219 | |
| 1,025410 | 1,193270 | | 0,291947 | | 1,445340 | 1,183450 | | 0,323291 | |
| 1,037560 | 0,861008 | | 0,264318 | | 0,515737 | 0,989574 | | 0,323745 | |
| 1,374960 | 0,947598 | | 0,293739 | | 0,687415 | 1,343900 | | 0,358326 | |
| 0,782035 | 0,738873 | | 0,336955 | | 1,385370 | 1,343490 | | 0,460922 | |
| 1,072920 | 1,370620 | | 0,355309 | | 1,490390 | 1,010190 | | 0,279016 | |
| 0,850960 | 1,235440 | | 0,428852 | | 1,461850 | 1,101730 | | 0,436248 | |
| 0,883982 | 0,615173 | | 0,339126 | | 1,312710 | 0,665640 | | 0,497917 | |
| 1,054900 | 0,616320 | | 0,284233 | | 0,985499 | 1,251580 | | 0,412853 | |
| 1,160660 | 0,579719 | | 0,293021 | | 0,689111 | 0,517529 | | 0,463138 | |
| 1,276010 | 0,628257 | | 0,311328 | | | | | | |
| 0,682538 | 0,864338 | | 0,324354 | | | | | | |
| 1,385650 | 0,866313 | | 0,484069 | | | | | | |
| 0,899687 | 0,513547 | | 0,414374 | | | | | | |
| | | | | pi lote | | | | | pi lote |
| | | | | 32,947892 | | | | | 27,373244 |
| 15,326146 | 12,376165 | | 5,245581 | | 12,005875 | 11,036126 | | 4,081243 | 0,250000 |
| | | pi remanescente | | | | | pi remanescente | | |
| 32,947892 | 32,947892 | 17,621746 | 5,245581 | 0,000000 | 27,373244 | 15,367369 | 4,331243 | 4,331243 | 0,250000 |
| Lote 5 - Peça C - 16 peças | | | | | Lote 6 - Peça E - 9 peças | | | | |
| M1 | M2 | M3 | M4 | M5 | M1 | M2 | M3 | M4 | M5 |
| 0,809178 | 1,050310 | | | 15 seg | 0,516818 | 0,929263 | | 0,399785 | 15 seg |
| 0,972270 | 1,049310 | | | | 0,904085 | 1,453150 | | 0,389019 | |
| 0,841119 | 0,818951 | | | | 0,524666 | 1,136940 | | 0,432192 | |
| 0,580943 | 1,265220 | | | | 1,159870 | 0,709448 | | 0,495910 | |
| 0,942357 | 1,148560 | | | | 0,642572 | 1,291920 | | 0,415891 | |
| 1,416830 | 1,409340 | | | | 0,734472 | 0,905174 | | 0,391918 | |
| 1,124610 | 0,614692 | | | | 0,748790 | 0,790090 | | 0,321806 | |
| 1,321220 | 0,841899 | | | | 1,319570 | 0,760318 | | 0,387911 | |
| 0,710857 | 0,570704 | | | | 1,056880 | 0,720438 | | 0,381353 | |
| 1,191730 | 0,719267 | | | | | | | | |
| 1,392710 | 0,686469 | | | | | | | | |
| 0,877700 | 0,884556 | | | | | | | | |
| 1,495480 | 0,618446 | | | | | | | | |
| 0,883693 | 0,996760 | | | | | | | | |
| 0,561910 | 1,127440 | | | | | | | | |
| 1,401740 | 0,797900 | | | | | | | | |
| | | | | pi lote | | | | | pi lote |
| | | | | 31,374171 | | | | | 20,170249 |
| 16,524347 | 14,599824 | | | 0,250000 | 7,607723 | 8,696741 | | 3,615785 | 0,250000 |
| | | pi remanescente | | | | | pi remanescente | | |
| 31,374171 | 31,374171 | 14,849824 | 0,250000 | 0,250000 | 20,170249 | 12,562526 | 3,865785 | 3,865785 | 0,250000 |

| Lote 7 - Peça B - 12 peças | | | | | Lote 8 - Peça A - 16 peças | | | | | |
|------------------------------------|------------------|------------------|-----------------|------------------|------------------------------------|------------------|------------------|------------------|------------------|-----------------|
| M1 | M2 | M3 | M4 | M5 | M1 | M2 | M3 | M4 | M5 | |
| 0,628410 | 1,002280 | 0,960438 | 0,384390 | | 0,558712 | 1,118150 | 0,541441 | 0,264666 | 15 seg | |
| 0,663759 | 0,772783 | 0,658623 | 0,468740 | | 0,784688 | 0,613876 | 0,833265 | 0,321756 | | |
| 1,289390 | 0,559725 | 1,411050 | 0,320509 | | 1,115100 | 1,386710 | 1,057300 | 0,423316 | | |
| 1,306010 | 0,546127 | 0,590863 | 0,393230 | | 1,240390 | 1,385540 | 0,628485 | 0,340252 | | |
| 1,274370 | 1,222970 | 1,235170 | 0,337740 | | 1,305730 | 0,581644 | 0,630649 | 0,361899 | | |
| 0,638553 | 0,564530 | 0,993282 | 0,345995 | | 1,138840 | 0,527341 | 1,013110 | 0,309718 | | |
| 0,932625 | 1,320040 | 1,333020 | 0,388725 | | 1,072640 | 0,702375 | 1,107250 | 0,467655 | | |
| 1,160760 | 0,867600 | 0,965726 | 0,415165 | | 0,542425 | 0,983307 | 1,263380 | 0,433859 | | |
| 1,417860 | 0,889566 | 1,442010 | 0,444002 | | 0,765622 | 0,886756 | 0,511702 | 0,278793 | | |
| 0,953738 | 0,943652 | 0,783004 | 0,295634 | | 1,184900 | 1,364320 | 1,442380 | 0,279080 | | |
| 0,885810 | 1,304370 | 1,467920 | 0,471414 | | 0,860374 | 0,681466 | 1,098710 | 0,269930 | | |
| 0,554515 | 0,958502 | 0,729594 | 0,349922 | | 1,401440 | 1,370850 | 1,099830 | 0,282064 | | |
| | | | | | 1,049060 | 1,045870 | 1,103430 | 0,341085 | | |
| | | | | | 1,191950 | 1,385320 | 1,189970 | 0,341578 | | |
| | | | | | 1,277850 | 1,402720 | 1,079320 | 0,253387 | | |
| | | | | | 0,678434 | 0,556491 | 0,912090 | 0,441088 | | |
| | | | | pi lote | | | | | pi lote | |
| | | | | 39,844111 | | | | | 53,333329 | |
| 11,705800 | 10,952145 | 12,570700 | 4,615466 | | 16,168155 | 15,992736 | 15,512312 | 5,410126 | 0,250000 | |
| | pi remanescente | | | | | pi remanescente | | | | |
| 39,844111 | 28,138311 | 17,186166 | 4,615466 | 0,000000 | 53,333329 | 37,165174 | 21,172438 | 5,660126 | 0,250000 | |
| Lote 9 - Peça C - 19 peças | | | | | Lote 10 - Peça D - 12 peças | | | | | |
| M1 | M2 | M3 | M4 | M5 | M1 | M2 | M3 | M4 | M5 | |
| | 0,828554 | 1,478130 | | 15 seg | 0,649847 | 1,120830 | 0,317026 | | | |
| | 1,483760 | 0,514599 | | | 0,992649 | 0,738960 | 0,486334 | | | |
| | 0,884666 | 1,294520 | | | 0,501181 | 0,869990 | 0,253934 | | | |
| | 0,576787 | 1,392020 | | | 0,821941 | 0,921527 | 0,296854 | | | |
| | 1,200680 | 0,667533 | | | 0,852515 | 1,342850 | 0,471342 | | | |
| | 0,540040 | 0,794680 | | | 1,192640 | 0,737580 | 0,497597 | | | |
| | 0,584587 | 1,419280 | | | 0,592868 | 1,089320 | 0,490463 | | | |
| | 1,021790 | 1,492490 | | | 0,640001 | 1,257250 | 0,453178 | | | |
| | 0,852525 | 0,795607 | | | 1,092020 | 0,558952 | 0,371375 | | | |
| | 0,768153 | 0,731566 | | | 1,399850 | 1,440900 | 0,297278 | | | |
| | 1,472340 | 1,036850 | | | 1,034280 | 0,759369 | 0,333780 | | | |
| | 1,036880 | 1,101060 | | | 1,267870 | 1,115570 | 0,323480 | | | |
| | 0,723589 | 1,400440 | | | | | | | | |
| | 1,004240 | 0,830822 | | | | | | | | |
| | 0,629601 | 1,383480 | | | | | | | | |
| | 0,889653 | 0,612121 | | | | | | | | |
| | 0,817055 | 1,329920 | | | | | | | | |
| | 0,570927 | 0,754695 | | pi lote | | | | | pi lote | |
| | 1,354890 | 1,273420 | | 37,793950 | | | | | 27,583401 | |
| | 17,240717 | 20,303233 | | 0,250000 | | 11,037662 | 11,953098 | 4,592641 | | |
| | pi remanescente | | | | | pi remanescente | | | | |
| 37,793950 | 37,793950 | 20,553233 | 0,250000 | 0,250000 | 27,583401 | 27,583401 | 16,545739 | 4,592641 | 0,000000 | |
| Lote 11 - Peça C - 19 peças | | | | | Lote 12 - Peça A - 9 peças | | | | | |
| M1 | M2 | M3 | M4 | M5 | M1 | M2 | M3 | M4 | M5 | |
| | 1,095270 | 1,132070 | | 15 seg | 1,369440 | 1,353960 | 1,285610 | 0,420864 | 15 seg | |
| | 0,515986 | 1,062400 | | | 0,865589 | 1,225100 | 1,258860 | 0,372394 | | |
| | 1,112870 | 0,892137 | | | 1,058980 | 0,946219 | 1,016750 | 0,460975 | | |
| | 0,820543 | 0,978041 | | | 0,822526 | 1,005200 | 0,899320 | 0,460874 | | |
| | 0,650694 | 0,776288 | | | 1,078770 | 1,399360 | 0,761479 | 0,377547 | | |
| | 1,312460 | 0,883035 | | | 1,075260 | 0,663816 | 1,319600 | 0,400433 | | |
| | 0,676251 | 0,955943 | | | 0,795150 | 1,215990 | 1,327560 | 0,291410 | | |
| | 1,287000 | 1,149160 | | | 1,466720 | 1,239900 | 0,973016 | 0,437894 | | |
| | 1,099250 | 1,355350 | | | 0,654733 | 1,373000 | 1,292280 | 0,254382 | | |
| | 0,927996 | 0,512005 | | | | | | | | |
| | 1,238610 | 0,865432 | | | | | | | | |
| | 1,478210 | 1,202530 | | | | | | | | |
| | 0,993041 | 1,057870 | | | | | | | | |
| | 0,799428 | 1,106720 | | | | | | | | |
| | 0,587039 | 0,574214 | | | | | | | | |
| | 0,693874 | 1,269990 | | | | | | | | |
| | 1,192780 | 1,192460 | | | | | | | | |
| | 1,124390 | 0,824747 | | pi lote | | | | | pi lote | |
| | 1,209210 | 0,939221 | | 37,794515 | | | | | 33,470961 | |
| | 18,814902 | 18,729613 | | 0,250000 | | 9,187168 | 10,422545 | 10,134475 | 3,476773 | 0,250000 |
| | pi remanescente | | | | | pi remanescente | | | | |
| 37,794515 | 37,794515 | 18,979613 | 0,250000 | 0,250000 | 33,470961 | 24,283793 | 13,861248 | 3,726773 | 0,250000 | |

| Lote 13 - Peça C - 11 peças | | | | | Lote 14 - Peça B - 15 peças | | | | |
|------------------------------------|------------------|------------------|----------|------------------|------------------------------------|------------------|------------------|-----------------|------------------|
| M1 | M2 | M3 | M4 | M5 | M1 | M2 | M3 | M4 | M5 |
| | 0,833635 | 0,683848 | | 15 seg | 1,101240 | 0,646083 | 1,415040 | 0,327295 | |
| | 1,390010 | 0,925041 | | | 1,202120 | 1,028880 | 1,136680 | 0,368067 | |
| | 1,292240 | 0,905383 | | | 0,950912 | 1,203690 | 1,234500 | 0,335280 | |
| | 1,086130 | 1,463830 | | | 1,391760 | 0,837397 | 1,049220 | 0,270236 | |
| | 0,943826 | 0,524588 | | | 0,761531 | 0,728094 | 1,091370 | 0,360589 | |
| | 1,278120 | 0,709747 | | | 0,629936 | 1,245590 | 0,706411 | 0,479207 | |
| | 1,493790 | 1,315210 | | | 0,757178 | 1,445260 | 0,539262 | 0,406151 | |
| | 1,268810 | 1,245260 | | | 0,521350 | 1,240690 | 0,913069 | 0,455304 | |
| | 0,599876 | 0,687318 | | | 0,573464 | 1,382610 | 0,727719 | 0,302714 | |
| | 1,203560 | 0,546054 | | | 1,059720 | 0,568757 | 1,147220 | 0,422933 | |
| | 0,684824 | 1,167320 | | | 0,885495 | 0,528518 | 0,780035 | 0,473177 | |
| | | | | | 1,162220 | 1,370310 | 1,473950 | 0,344425 | |
| | | | | | 0,501297 | 0,814125 | 1,227700 | 0,498871 | |
| | | | | | 1,291690 | 1,248170 | 0,870107 | 0,345923 | |
| | | | | | 1,299930 | 0,664762 | 1,392820 | 0,265478 | |
| | | | | pi lote | | | | | pi lote |
| | | | | 22,498420 | | | | | 50,403532 |
| | 12,074821 | 10,173599 | | 0,250000 | 14,089843 | 14,952936 | 15,705103 | 5,655650 | |
| | pi remanescente | | | | pi remanescente | | | | |
| 22,498420 | 22,498420 | 10,423599 | 0,250000 | 0,250000 | 50,403532 | 36,313689 | 21,360753 | 5,655650 | 0,000000 |
| Lote 15 - Peça C - 10 peças | | | | | Lote 16 - Peça E - 15 peças | | | | |
| M1 | M2 | M3 | M4 | M5 | M1 | M2 | M3 | M4 | M5 |
| | 0,736201 | 0,682346 | | 15 seg | 1,440530 | 0,580970 | | 0,475436 | 15 seg |
| | 1,262920 | 1,133840 | | | 1,039980 | 1,223160 | | 0,387579 | |
| | 0,770486 | 1,324280 | | | 0,601909 | 1,020240 | | 0,387328 | |
| | 0,928807 | 1,238020 | | | 1,399240 | 0,993139 | | 0,329738 | |
| | 0,715871 | 1,279120 | | | 0,848829 | 1,396730 | | 0,441304 | |
| | 0,772698 | 1,328520 | | | 1,231930 | 0,808508 | | 0,412140 | |
| | 0,777616 | 0,999029 | | | 1,464000 | 1,297110 | | 0,477335 | |
| | 0,760236 | 1,041320 | | | 0,874891 | 0,843782 | | 0,278673 | |
| | 0,875321 | 1,114130 | | | 0,535283 | 0,775020 | | 0,335475 | |
| | 0,912805 | 0,622854 | | | 0,834725 | 0,943483 | | 0,267676 | |
| | | | | | 0,544136 | 1,040760 | | 0,304817 | |
| | | | | | 0,772811 | 1,402430 | | 0,296617 | |
| | | | | | 0,661104 | 0,564813 | | 0,346139 | |
| | | | | | 1,364570 | 1,092960 | | 0,279612 | |
| | | | | | 0,909091 | 0,909839 | | 0,374190 | |
| | | | | pi lote | | | | | pi lote |
| | | | | 19,526420 | | | | | 35,060032 |
| | 8,512961 | 10,763459 | | 0,250000 | 14,523029 | 14,892944 | | 5,394059 | 0,250000 |
| | pi remanescente | | | | pi remanescente | | | | |
| 19,526420 | 19,526420 | 11,013459 | 0,250000 | 0,250000 | 35,060032 | 20,537003 | 5,644059 | 5,644059 | 0,250000 |
| Lote 17 - Peça C - 8 peças | | | | | Lote 18 - Peça E - 16 peças | | | | |
| M1 | M2 | M3 | M4 | M5 | M1 | M2 | M3 | M4 | M5 |
| | 1,297310 | 0,623733 | | 15 seg | 1,352460 | 1,161660 | | 0,406861 | |
| | 1,393030 | 0,718464 | | | 0,802149 | 0,614199 | | 0,324475 | |
| | 0,594487 | 0,693425 | | | 1,148340 | 1,477800 | | 0,254205 | |
| | 0,547533 | 0,988874 | | | 0,928510 | 1,093440 | | 0,351021 | |
| | 0,987967 | 1,182010 | | | 1,130990 | 1,060500 | | 0,256167 | |
| | 0,756722 | 0,801549 | | | 0,724438 | 0,894598 | | 0,414968 | |
| | 1,058400 | 1,225280 | | | 0,610998 | 0,516471 | | 0,285643 | |
| | 0,890901 | 1,024580 | | | 1,129190 | 0,751249 | | 0,308618 | |
| | | | | | 1,131560 | 1,321570 | | 0,312197 | |
| | | | | | 1,165720 | 1,324850 | | 0,454894 | |
| | | | | | 0,541407 | 0,889148 | | 0,389221 | |
| | | | | | 1,330720 | 1,268750 | | 0,357316 | |
| | | | | | 1,066890 | 0,901901 | | 0,488289 | |
| | | | | | 1,162590 | 0,755471 | | 0,409234 | |
| | | | | | 0,539498 | 0,526608 | | 0,302362 | |
| | | | | | 1,470280 | 0,575812 | | 0,447979 | |
| | | | | pi lote | | | | | pi lote |
| | | | | 15,034265 | | | | | 37,383217 |
| | 7,526350 | 7,257915 | | 0,250000 | 16,235740 | 15,134027 | | 5,763450 | 0,250000 |
| | pi remanescente | | | | pi remanescente | | | | |
| 15,034265 | 15,034265 | 7,507915 | 0,250000 | 0,250000 | 37,383217 | 21,147477 | 6,013450 | 6,013450 | 0,250000 |

| Lote 19 - Peça D - 18 peças | | | | | Lote 20 - Peça A - 20 peças | | | | |
|------------------------------------|------------------|------------------|------------------|------------------|------------------------------------|------------------|------------------|-----------------|------------------|
| M1 | M2 | M3 | M4 | M5 | M1 | M2 | M3 | M4 | M5 |
| | 0,998613 | 0,793184 | 0,351293 | | 0,997640 | 1,282460 | 1,193800 | 0,264931 | 15 seg |
| | 0,708248 | 1,415500 | 0,322523 | | 0,965775 | 0,505452 | 0,803280 | 0,261532 | |
| | 0,623153 | 1,320190 | 0,315079 | | 0,573475 | 1,307590 | 1,019130 | 0,430743 | |
| | 1,412180 | 0,531321 | 0,305109 | | 0,578428 | 0,947303 | 0,967930 | 0,266133 | |
| | 1,222650 | 1,133230 | 0,282102 | | 1,245580 | 0,619839 | 1,471850 | 0,455011 | |
| | 0,780355 | 0,788809 | 0,290940 | | 0,503729 | 0,907669 | 0,811461 | 0,341900 | |
| | 0,949274 | 0,694660 | 0,447347 | | 1,212400 | 0,907178 | 1,406500 | 0,347391 | |
| | 1,328920 | 0,630023 | 0,451503 | | 0,673179 | 1,195830 | 1,374460 | 0,360913 | |
| | 0,628738 | 0,653255 | 0,443591 | | 1,145280 | 1,369720 | 1,162490 | 0,451092 | |
| | 0,740773 | 0,705906 | 0,284638 | | 0,617819 | 0,676144 | 1,220340 | 0,364626 | |
| | 0,693018 | 0,508991 | 0,358156 | | 1,157530 | 1,207550 | 1,137410 | 0,365110 | |
| | 1,203250 | 0,510946 | 0,415189 | | 0,987289 | 0,949087 | 1,253540 | 0,289656 | |
| | 0,841744 | 0,806346 | 0,479466 | | 1,035260 | 1,169420 | 1,300200 | 0,477762 | |
| | 0,696986 | 0,592387 | 0,363434 | | 1,056500 | 0,955754 | 1,259290 | 0,272716 | |
| | 0,847374 | 1,237680 | 0,346452 | | 0,637369 | 1,419860 | 0,818665 | 0,433794 | |
| | 0,926911 | 0,731060 | 0,263629 | | 0,667383 | 1,346950 | 0,987684 | 0,373320 | |
| | 1,308040 | 1,319710 | 0,375571 | | 1,416530 | 0,732154 | 1,081820 | 0,458255 | |
| | 1,405970 | 1,446400 | 0,318196 | pi lote | 0,718485 | 1,101690 | 1,431330 | 0,366432 | pi lote |
| | | | | 39,550013 | 0,996057 | 0,652319 | 1,231200 | 0,485502 | 69,288349 |
| | | | | | 1,252810 | 1,452940 | 0,572972 | 0,320751 | |
| | 17,316197 | 15,819598 | 6,414218 | | 18,438518 | 20,706909 | 22,505352 | 7,387570 | 0,250000 |
| | | | | pi remanescente | | | | | pi remanescente |
| 39,550013 | 39,550013 | 22,233816 | 6,414218 | 0,000000 | 69,288349 | 50,849831 | 30,142922 | 7,637570 | 0,250000 |
| Lote 21 - Peça B - 18 peças | | | | | Lote 22 - Peça A - 16 peças | | | | |
| M1 | M2 | M3 | M4 | M5 | M1 | M2 | M3 | M4 | M5 |
| | 0,700253 | 0,962252 | 1,159820 | 0,491980 | 1,078530 | 1,052430 | 0,583578 | 0,404538 | 15 seg |
| | 1,246030 | 0,935410 | 1,357080 | 0,307399 | 0,785750 | 1,283970 | 1,111170 | 0,278469 | |
| | 0,719126 | 1,159410 | 0,604054 | 0,264678 | 0,988174 | 1,147690 | 1,069650 | 0,471677 | |
| | 1,444900 | 0,768016 | 0,994499 | 0,321172 | 0,668762 | 1,035890 | 0,807271 | 0,471385 | |
| | 1,247230 | 0,618299 | 0,951652 | 0,403775 | 0,597134 | 0,660576 | 1,170280 | 0,270411 | |
| | 0,667371 | 0,506929 | 0,651953 | 0,435098 | 1,310790 | 1,148120 | 1,312300 | 0,256835 | |
| | 0,981548 | 1,320290 | 1,261360 | 0,451433 | 1,484520 | 0,724882 | 1,221910 | 0,300594 | |
| | 0,647103 | 1,030600 | 1,146810 | 0,409710 | 0,757779 | 1,398630 | 0,831570 | 0,370827 | |
| | 0,696753 | 0,939536 | 1,337560 | 0,393161 | 0,644939 | 1,122390 | 1,086680 | 0,346689 | |
| | 0,569897 | 0,644003 | 1,168910 | 0,260606 | 0,607831 | 0,935699 | 0,614974 | 0,466079 | |
| | 0,985409 | 1,007750 | 1,063170 | 0,316406 | 1,197080 | 0,884116 | 0,502255 | 0,295367 | |
| | 1,088530 | 1,070730 | 1,382750 | 0,421224 | 0,952499 | 0,904590 | 0,622518 | 0,467713 | |
| | 0,930903 | 0,562964 | 0,794407 | 0,340093 | 1,086780 | 0,928198 | 1,258100 | 0,386469 | |
| | 1,112310 | 1,010850 | 0,808940 | 0,475360 | 0,883447 | 1,169500 | 1,193500 | 0,471330 | |
| | 1,387970 | 0,774146 | 0,772376 | 0,387265 | 1,441560 | 0,696713 | 0,593863 | 0,475681 | |
| | 1,276410 | 1,099690 | 1,031180 | 0,422988 | 1,282390 | 0,563162 | 0,586177 | 0,264123 | |
| | 1,320170 | 1,201240 | 1,406900 | 0,444462 | | | | | |
| | 0,636963 | 1,007540 | 1,117560 | 0,294609 | | | | | pi lote |
| | | | | 60,130931 | | | | | 52,238504 |
| | 17,658876 | 16,619655 | 19,010981 | 6,841419 | 15,767965 | 15,656556 | 14,565796 | 5,998187 | 0,250000 |
| | | | | pi remanescente | | | | | pi remanescente |
| 60,130931 | 42,472055 | 25,852400 | 6,841419 | 0,000000 | 52,238504 | 36,470539 | 20,813983 | 6,248187 | 0,250000 |
| Lote 23 - Peça E - 15 peças | | | | | Lote 24 - Peça C - 11 peças | | | | |
| M1 | M2 | M3 | M4 | M5 | M1 | M2 | M3 | M4 | M5 |
| | 1,479810 | 0,872491 | 0,260360 | 15 seg | | 0,604356 | 0,754896 | | 15 seg |
| | 0,697117 | 1,110570 | 0,333316 | | | 0,570067 | 0,640684 | | |
| | 0,806051 | 0,681681 | 0,389326 | | | 1,071130 | 1,316660 | | |
| | 0,895431 | 1,455290 | 0,282121 | | | 0,805561 | 1,391410 | | |
| | 1,096580 | 0,733097 | 0,282662 | | | 1,216010 | 0,848292 | | |
| | 0,525604 | 1,291620 | 0,378276 | | | 0,864059 | 0,698450 | | |
| | 1,092880 | 1,411200 | 0,401812 | | | 0,802526 | 1,009590 | | |
| | 1,364700 | 1,236010 | 0,440845 | | | 1,028770 | 1,013790 | | |
| | 1,162150 | 0,874466 | 0,252925 | | | 0,818365 | 0,552000 | | |
| | 0,528283 | 1,394070 | 0,485594 | | | 1,398450 | 1,411510 | | |
| | 1,442150 | 0,575962 | 0,399679 | | | 1,271150 | 1,079900 | | |
| | 0,691582 | 0,963493 | 0,399958 | | | | | | |
| | 1,256270 | 1,383430 | 0,400857 | | | | | | |
| | 1,060870 | 0,527896 | 0,422492 | | | | | | |
| | 0,664915 | 0,676311 | 0,394829 | | | | | | |
| | | | | pi lote | | | | | pi lote |
| | | | | 35,727032 | | | | | 21,417626 |
| | 14,764393 | 15,187587 | 5,525052 | 0,250000 | 10,450444 | 10,717182 | | | 0,250000 |
| | | | | pi remanescente | | | | | pi remanescente |
| 35,727032 | 20,962639 | 5,775052 | 5,775052 | 0,250000 | 21,417626 | 21,417626 | 10,967182 | 0,250000 | 0,250000 |

| Lote 25 - Peça B - 16 peças | | | | | Lote 26 - Peça E - 19 peças | | | | |
|-----------------------------|------------------|------------------|-----------------|-----------------|-----------------------------|------------------|------------------|-----------------|-----------------|
| M1 | M2 | M3 | M4 | M5 | M1 | M2 | M3 | M4 | M5 |
| 1,460890 | 0,968088 | 0,943879 | 0,353022 | | 1,363200 | 1,497400 | | 0,347413 | 15 seg |
| 1,298510 | 0,779214 | 0,543642 | 0,332138 | | 0,773081 | 0,835266 | | 0,329264 | |
| 1,279740 | 0,599689 | 0,611589 | 0,495941 | | 0,883003 | 0,783521 | | 0,267732 | |
| 0,862852 | 1,328060 | 0,859776 | 0,346167 | | 1,466650 | 1,355120 | | 0,463723 | |
| 0,614253 | 0,982187 | 0,963295 | 0,269197 | | 1,236860 | 0,987250 | | 0,494533 | |
| 0,576125 | 1,348930 | 1,232400 | 0,425170 | | 1,353480 | 0,549258 | | 0,253650 | |
| 1,082640 | 0,578589 | 0,526230 | 0,260010 | | 0,892704 | 0,826955 | | 0,448630 | |
| 0,982085 | 0,859198 | 1,061180 | 0,271147 | | 0,645068 | 0,713873 | | 0,473006 | |
| 1,374940 | 0,567222 | 0,629553 | 0,380448 | | 1,091510 | 1,131850 | | 0,291883 | |
| 1,497870 | 0,600291 | 1,045270 | 0,338131 | | 0,815794 | 1,196710 | | 0,323670 | |
| 0,502261 | 1,050040 | 0,883842 | 0,317038 | | 0,533431 | 0,579843 | | 0,479819 | |
| 1,132770 | 0,604907 | 0,923059 | 0,493085 | | 1,309050 | 1,397790 | | 0,498123 | |
| 0,542148 | 1,356900 | 1,070690 | 0,384220 | | 0,754647 | 1,101310 | | 0,323902 | |
| 0,930434 | 0,804144 | 1,465100 | 0,305897 | | 0,507994 | 0,501832 | | 0,307892 | |
| 0,559315 | 1,237820 | 1,292640 | 0,376060 | | 0,629845 | 1,359320 | | 0,384213 | |
| 1,063980 | 1,240680 | 1,391810 | 0,282400 | | 0,818377 | 1,169030 | | 0,400266 | |
| | | | | pi lote | 1,121680 | 0,668827 | | 0,475109 | |
| | | | | 51,740798 | 1,164190 | 0,800460 | | 0,332705 | pi lote |
| | | | | | 1,487910 | 1,129750 | | 0,470870 | 45,050242 |
| 15,760813 | 14,905959 | 15,443955 | 5,630071 | | 18,848474 | 18,585365 | | 7,366403 | 0,250000 |
| | pi remanescente | | | | | pi remanescente | | | |
| 51,740798 | 35,979985 | 21,074026 | 5,630071 | 0,000000 | 45,050242 | 26,201768 | 7,616403 | 7,616403 | 0,250000 |
| Lote 27 - Peça D - 18 peças | | | | | Lote 28 - Peça B - 17 peças | | | | |
| M1 | M2 | M3 | M4 | M5 | M1 | M2 | M3 | M4 | M5 |
| | 1,458120 | 1,055870 | 0,278030 | | 1,218010 | 1,422160 | 0,684277 | 0,342498 | |
| | 1,319780 | 1,416920 | 0,457480 | | 1,158670 | 1,205310 | 1,211520 | 0,355382 | |
| | 0,720320 | 0,741276 | 0,313674 | | 1,456340 | 1,462160 | 0,721127 | 0,460712 | |
| | 0,969589 | 0,746626 | 0,443355 | | 1,125340 | 0,897642 | 1,419830 | 0,309395 | |
| | 0,769295 | 0,585235 | 0,287462 | | 0,915035 | 0,691664 | 1,383370 | 0,397330 | |
| | 1,454200 | 0,776095 | 0,373162 | | 1,494940 | 1,336180 | 1,189830 | 0,439311 | |
| | 0,778697 | 0,522648 | 0,250295 | | 0,511415 | 1,046000 | 0,846509 | 0,264738 | |
| | 1,341930 | 0,792634 | 0,330485 | | 0,616675 | 0,874321 | 1,052780 | 0,485225 | |
| | 1,368280 | 0,992261 | 0,338129 | | 0,566206 | 0,547699 | 1,478760 | 0,314842 | |
| | 1,234690 | 0,906005 | 0,423160 | | 0,960208 | 1,117770 | 0,592931 | 0,403893 | |
| | 0,961836 | 0,741671 | 0,273217 | | 1,337200 | 1,439200 | 0,808700 | 0,398817 | |
| | 1,376160 | 0,747452 | 0,285000 | | 1,041790 | 1,055750 | 1,412010 | 0,253997 | |
| | 0,639233 | 1,137580 | 0,398006 | | 1,012040 | 1,040560 | 1,027140 | 0,403217 | |
| | 0,577689 | 0,523672 | 0,474962 | | 0,750202 | 0,871226 | 1,262780 | 0,330136 | |
| | 0,617131 | 0,999888 | 0,383570 | | 0,884998 | 0,825547 | 0,942344 | 0,287674 | |
| | 1,220680 | 1,419630 | 0,441966 | | 1,287200 | 0,812375 | 0,746412 | 0,453115 | |
| | 0,546407 | 1,323520 | 0,405206 | | 0,678118 | 1,160690 | 1,348430 | 0,294063 | |
| | 1,115710 | 0,757730 | 0,309740 | pi lote | | | | | pi lote |
| | | | | 41,123359 | | | | | 59,143736 |
| | 18,469747 | 16,186713 | 6,466899 | | 17,014387 | 17,806254 | 18,128750 | 6,194345 | |
| | pi remanescente | | | | | pi remanescente | | | |
| 41,123359 | 41,123359 | 22,653612 | 6,466899 | 0,000000 | 59,143736 | 42,129349 | 24,323095 | 6,194345 | 0,000000 |
| Lote 29 - Peça E - 17 peças | | | | | Lote 30 - Peça D - 8 peças | | | | |
| M1 | M2 | M3 | M4 | M5 | M1 | M2 | M3 | M4 | M5 |
| 0,610909 | 1,288510 | | 0,446750 | 15 seg | 0,589580 | 1,353730 | 0,463838 | | |
| 0,773709 | 0,678791 | | 0,399813 | | 0,631019 | 1,246340 | 0,253001 | | |
| 1,220810 | 1,428560 | | 0,356999 | | 0,725768 | 0,714611 | 0,341358 | | |
| 1,385270 | 1,314810 | | 0,434652 | | 0,686413 | 0,536325 | 0,425633 | | |
| 0,635124 | 0,881398 | | 0,494553 | | 1,260110 | 1,458360 | 0,389467 | | |
| 1,155280 | 0,966914 | | 0,373260 | | 0,646207 | 0,811916 | 0,401681 | | |
| 1,112260 | 0,567600 | | 0,324857 | | 1,269690 | 0,754439 | 0,268554 | | |
| 1,474030 | 0,603202 | | 0,271760 | | 1,463330 | 0,610005 | 0,442497 | | |
| 0,643278 | 0,667779 | | 0,298469 | | | | | | |
| 0,982997 | 0,646484 | | 0,408018 | | | | | | |
| 1,307010 | 0,654204 | | 0,390600 | | | | | | |
| 0,646721 | 0,514135 | | 0,348034 | | | | | | |
| 0,661807 | 0,615647 | | 0,369510 | | | | | | |
| 1,447470 | 0,746296 | | 0,319072 | | | | | | |
| 0,911230 | 1,444610 | | 0,345759 | | | | | | |
| 0,601458 | 0,634074 | | 0,363986 | | | | | | |
| 0,792886 | 1,099470 | | 0,412289 | pi lote | | | | | pi lote |
| | | | | 37,723114 | | | | | 17,743872 |
| 16,362249 | 14,752484 | | 6,358381 | 0,250000 | 7,272117 | 7,485726 | 2,986029 | | |
| | pi remanescente | | | | | pi remanescente | | | |
| 37,723114 | 21,360865 | 6,608381 | 6,608381 | 0,250000 | 17,743872 | 17,743872 | 10,471755 | 2,986029 | 0,000000 |

| Lote 43 - Peça D - 13 peças | | | | | Lote 44 - Peça C - 13 peças | | | | | |
|------------------------------------|------------------|------------------|------------------|------------------|------------------------------------|------------------|------------------|-----------------|------------------|-----------------|
| M1 | M2 | M3 | M4 | M5 | M1 | M2 | M3 | M4 | M5 | |
| | 1,169620 | 1,378730 | 0,434506 | | | 1,499140 | 0,921164 | | 15 seg | |
| | 0,943064 | 0,974931 | 0,444780 | | | 0,848405 | 1,379480 | | | |
| | 1,171260 | 0,753785 | 0,457131 | | | 0,657284 | 0,563301 | | | |
| | 1,204470 | 0,606006 | 0,374757 | | | 1,111090 | 0,505220 | | | |
| | 0,875482 | 0,893996 | 0,385330 | | | 1,256460 | 0,782036 | | | |
| | 1,321770 | 0,838790 | 0,403532 | | | 0,685257 | 1,031570 | | | |
| | 0,675185 | 0,916172 | 0,280713 | | | 1,159360 | 1,417490 | | | |
| | 0,777302 | 1,150070 | 0,485133 | | | 0,561233 | 0,838371 | | | |
| | 1,167520 | 1,067770 | 0,384995 | | | 0,952301 | 1,188740 | | | |
| | 1,154990 | 0,775272 | 0,275477 | | | 0,811196 | 0,706126 | | | |
| | 1,134990 | 0,810046 | 0,474809 | | | 0,683785 | 0,547907 | | | |
| | 1,063120 | 0,673143 | 0,337207 | | | 0,859513 | 1,197880 | | | |
| | 1,487790 | 0,669161 | 0,432981 | | | 1,362530 | 0,990612 | | | |
| | | | | pi lote | | | | | pi lote | |
| | | | | 30,825786 | | | | | 24,767451 | |
| | 14,146563 | 11,507872 | 5,171351 | | | 12,447554 | 12,069897 | | 0,250000 | |
| | pi remanescente | | | | | pi remanescente | | | | |
| 30,825786 | 30,825786 | 16,679223 | 5,171351 | 0,000000 | 24,767451 | 24,767451 | 12,319897 | 0,250000 | 0,250000 | |
| Lote 45 - Peça D - 11 peças | | | | | Lote 46 - Peça A - 8 peças | | | | | |
| M1 | M2 | M3 | M4 | M5 | M1 | M2 | M3 | M4 | M5 | |
| | 0,743071 | 0,722377 | 0,490999 | | | 0,593745 | 0,608934 | 0,910083 | 0,380059 | 15 seg |
| | 0,692293 | 0,745942 | 0,343723 | | | 1,100720 | 1,290690 | 1,068510 | 0,373285 | |
| | 0,727821 | 1,206050 | 0,258821 | | | 1,497680 | 0,617850 | 0,857722 | 0,474181 | |
| | 0,503239 | 0,925366 | 0,333681 | | | 0,818878 | 1,469220 | 1,008530 | 0,327127 | |
| | 1,246270 | 0,611096 | 0,261034 | | | 1,142450 | 1,135940 | 1,432430 | 0,449278 | |
| | 0,564931 | 0,818969 | 0,318203 | | | 1,441930 | 0,632478 | 0,823638 | 0,335945 | |
| | 1,060390 | 1,419750 | 0,290276 | | | 0,562883 | 0,637902 | 1,465580 | 0,318755 | |
| | 0,625063 | 1,412440 | 0,466142 | | | 1,408000 | 0,582140 | 1,133350 | 0,360871 | |
| | 1,497510 | 1,405680 | 0,352273 | | | | | | | |
| | 0,659943 | 0,532156 | 0,270242 | | | | | | | |
| | 0,651547 | 1,483030 | 0,430791 | | | | | | | |
| | | | | pi lote | | | | | pi lote | |
| | | | | 24,071119 | | | | | 27,510784 | |
| | 8,972078 | 11,282856 | 3,816185 | | | 8,566286 | 6,975154 | 8,699843 | 3,019501 | 0,250000 |
| | pi remanescente | | | | | pi remanescente | | | | |
| 24,071119 | 24,071119 | 15,099041 | 3,816185 | 0,000000 | 27,510784 | 18,944498 | 11,969344 | 3,269501 | 0,250000 | |
| Lote 47 - Peça B - 12 peças | | | | | Lote 48 - Peça E - 13 peças | | | | | |
| M1 | M2 | M3 | M4 | M5 | M1 | M2 | M3 | M4 | M5 | |
| | 1,013290 | 1,045540 | 1,314610 | 0,385191 | | 1,258350 | 1,382360 | | 0,347725 | 15 seg |
| | 0,984325 | 1,174640 | 0,832269 | 0,475608 | | 0,651903 | 0,953775 | | 0,280933 | |
| | 0,538546 | 0,663199 | 0,871841 | 0,266203 | | 1,044380 | 1,375320 | | 0,304616 | |
| | 1,209340 | 0,613922 | 0,671070 | 0,398241 | | 0,667964 | 1,289010 | | 0,298356 | |
| | 0,636711 | 0,946068 | 1,242640 | 0,352460 | | 1,030760 | 1,128360 | | 0,372219 | |
| | 0,942404 | 0,926650 | 1,374210 | 0,449329 | | 0,948151 | 0,962644 | | 0,420502 | |
| | 1,314410 | 1,292530 | 0,536123 | 0,473259 | | 1,095740 | 1,211890 | | 0,325387 | |
| | 1,161170 | 0,516068 | 0,668249 | 0,273622 | | 0,887841 | 0,517171 | | 0,431319 | |
| | 0,759760 | 1,115120 | 0,850295 | 0,261883 | | 0,541672 | 0,959117 | | 0,381145 | |
| | 1,475920 | 1,389600 | 0,696932 | 0,371992 | | 1,230800 | 1,134790 | | 0,463115 | |
| | 1,279420 | 1,043360 | 0,947792 | 0,314181 | | 0,896698 | 1,371060 | | 0,325537 | |
| | 1,102230 | 1,154840 | 0,673433 | 0,389600 | | 0,952163 | 0,567935 | | 0,412085 | |
| | | | | pi lote | | 0,703293 | 1,360090 | | 0,357128 | |
| | | | | 39,390096 | | | | | pi lote | |
| | | | | | | | | | 31,093304 | |
| | 12,417526 | 11,881537 | 10,679464 | 4,411569 | | 11,909715 | 14,213522 | | 4,720067 | 0,250000 |
| | pi remanescente | | | | | pi remanescente | | | | |
| 39,390096 | 26,972570 | 15,091033 | 4,411569 | 0,000000 | 31,093304 | 19,183589 | 4,970067 | 4,970067 | 0,250000 | |

| Lote 49 - Peça C - 17 peças | | | | | Lote 50 - Peça A - 13 peças | | | | |
|------------------------------------|------------------|------------------|-----------------|------------------|------------------------------------|------------------|------------------|-----------------|------------------|
| M1 | M2 | M3 | M4 | M5 | M1 | M2 | M3 | M4 | M5 |
| | 1,314060 | 1,344190 | | 15 seg | 0,724704 | 1,052510 | 1,044470 | 0,407747 | 15 seg |
| | 0,619969 | 1,253490 | | | 1,123380 | 1,219500 | 0,654981 | 0,306109 | |
| | 1,223130 | 1,013480 | | | 0,687332 | 1,459900 | 1,336100 | 0,277750 | |
| | 0,975527 | 0,884922 | | | 0,729598 | 1,467990 | 1,455270 | 0,407297 | |
| | 1,156400 | 0,989175 | | | 1,297210 | 0,774238 | 1,454370 | 0,407891 | |
| | 1,413040 | 0,753336 | | | 0,568049 | 0,578285 | 1,189630 | 0,416430 | |
| | 1,324530 | 1,069710 | | | 1,024480 | 1,275150 | 0,925089 | 0,260352 | |
| | 1,211370 | 0,931180 | | | 1,183580 | 0,745120 | 1,332070 | 0,457681 | |
| | 1,104680 | 0,805043 | | | 0,850541 | 0,551273 | 1,464200 | 0,391723 | |
| | 0,944573 | 1,273050 | | | 0,599234 | 0,747019 | 0,850293 | 0,415649 | |
| | 1,456360 | 1,135370 | | | 1,380690 | 1,294350 | 1,363620 | 0,259874 | |
| | 1,473040 | 0,998427 | | | 1,109940 | 0,628433 | 1,365520 | 0,492569 | |
| | 0,939582 | 0,964620 | | | 0,630219 | 0,924761 | 0,867318 | 0,415415 | |
| | 0,967793 | 0,861689 | | | | | | | |
| | 1,306370 | 0,635663 | | | | | | | |
| | 1,017680 | 1,485750 | | | | | | | |
| | 0,909453 | 1,353040 | | | | | | | |
| | | | | pi lote | | | | | pi lote |
| | | | | 37,359692 | | | | | 45,096904 |
| | 19,357557 | 17,752135 | | 0,250000 | 11,908957 | 12,718529 | 15,302931 | 4,916487 | 0,250000 |
| | pi remanescente | | | | pi remanescente | | | | |
| 37,359692 | 37,359692 | 18,002135 | 0,250000 | 0,250000 | 45,096904 | 33,187947 | 20,469418 | 5,166487 | 0,250000 |
| Lote 51 - Peça B - 14 peças | | | | | Lote 52 - Peça C - 14 peças | | | | |
| M1 | M2 | M3 | M4 | M5 | M1 | M2 | M3 | M4 | M5 |
| 0,993500 | 0,572975 | 1,334740 | 0,278550 | | 1,224060 | 1,312730 | | | 15 seg |
| 1,111660 | 1,274290 | 0,760726 | 0,494451 | | 0,753559 | 0,755031 | | | |
| 0,648666 | 1,432760 | 1,313500 | 0,398361 | | 1,374970 | 0,586712 | | | |
| 1,421220 | 0,625765 | 1,016690 | 0,390124 | | 1,494750 | 1,015740 | | | |
| 0,766193 | 1,036760 | 1,183990 | 0,348650 | | 1,419820 | 0,608112 | | | |
| 1,427500 | 1,123010 | 0,921651 | 0,254118 | | 0,786587 | 0,627927 | | | |
| 1,308260 | 0,663999 | 0,602587 | 0,312812 | | 0,800598 | 0,634457 | | | |
| 1,312730 | 0,859565 | 0,961489 | 0,455393 | | 0,629452 | 1,087140 | | | |
| 1,463560 | 0,742683 | 0,527162 | 0,456214 | | 1,344840 | 0,943599 | | | |
| 1,335490 | 1,499500 | 0,881410 | 0,347287 | | 0,511829 | 0,787055 | | | |
| 1,085320 | 1,032160 | 0,539065 | 0,442188 | | 0,526019 | 0,539329 | | | |
| 0,555991 | 1,063520 | 1,479820 | 0,350475 | | 0,878309 | 1,059830 | | | |
| 0,785084 | 1,155240 | 1,150430 | 0,313868 | | 0,914085 | 1,476080 | | | |
| 1,499740 | 1,095610 | 1,033020 | 0,256652 | | 1,332130 | 1,480490 | | | |
| | | | | pi lote | | | | | pi lote |
| | | | | 48,698174 | | | | | 27,155240 |
| 15,714914 | 14,177837 | 13,706280 | 5,099143 | | 13,991008 | 12,914232 | | | 0,250000 |
| pi remanescente | | | | | pi remanescente | | | | |
| 48,698174 | 32,983260 | 18,805423 | 5,099143 | 0,000000 | 27,155240 | 27,155240 | 13,164232 | 0,250000 | 0,250000 |
| Lote 53 - Peça A - 9 peças | | | | | Lote 54 - Peça C - 17 peças | | | | |
| M1 | M2 | M3 | M4 | M5 | M1 | M2 | M3 | M4 | M5 |
| 0,594857 | 1,022510 | 0,640696 | 0,268953 | 15 seg | 1,331800 | 1,230240 | | | 15 seg |
| 1,447860 | 0,584745 | 0,546844 | 0,423194 | | 0,654294 | 0,970839 | | | |
| 0,605937 | 1,201000 | 0,854441 | 0,406098 | | 1,168300 | 0,699087 | | | |
| 1,416230 | 1,026720 | 1,050230 | 0,427302 | | 0,609697 | 0,764897 | | | |
| 1,427060 | 1,458100 | 1,459740 | 0,423114 | | 0,986553 | 1,074360 | | | |
| 1,384620 | 0,695407 | 1,358450 | 0,331187 | | 1,282240 | 1,337840 | | | |
| 1,438140 | 1,468600 | 1,211070 | 0,359805 | | 0,948859 | 1,250190 | | | |
| 0,715427 | 1,102220 | 0,639749 | 0,374653 | | 0,713639 | 1,477380 | | | |
| 0,618641 | 0,639230 | 0,883107 | 0,302062 | | 0,562137 | 0,740461 | | | |
| | | | | pi lote | | 1,031030 | 0,810097 | | |
| | | | | 31,057999 | | 0,893962 | 1,102250 | | |
| | | | | | | 0,748024 | 1,499530 | | |
| | | | | | | 0,523660 | 1,494790 | | |
| | | | | | | 1,133820 | 0,655295 | | |
| | | | | | | 0,799604 | 0,585309 | | |
| | | | | | | 0,885100 | 0,780481 | | |
| | | | | | | 0,584626 | 0,815007 | | |
| | | | | pi lote | | | | | pi lote |
| | | | | 31,057999 | | | | | 32,395398 |
| 9,648772 | 9,198532 | 8,644327 | 3,316368 | 0,250000 | 14,857345 | 17,288053 | | | 0,250000 |
| pi remanescente | | | | | pi remanescente | | | | |
| 31,057999 | 21,409227 | 12,210695 | 3,566368 | 0,250000 | 32,395398 | 32,395398 | 17,538053 | 0,250000 | 0,250000 |

| Lote 55 - Peça A - 13 peças | | | | | Lote 56 - Peça C - 16 peças | | | | |
|------------------------------------|------------------|------------------|-----------------|------------------|------------------------------------|------------------|------------------|-----------------|------------------|
| M1 | M2 | M3 | M4 | M5 | M1 | M2 | M3 | M4 | M5 |
| 0,959749 | 1,445660 | 1,491490 | 0,280788 | 15 seg | 1,304550 | 1,251330 | | | 15 seg |
| 1,234540 | 1,414400 | 0,742005 | 0,478044 | | 1,484470 | 0,556600 | | | |
| 0,964000 | 0,981025 | 0,684015 | 0,430662 | | 1,021920 | 0,835578 | | | |
| 0,950661 | 0,632829 | 1,017620 | 0,320089 | | 0,641723 | 0,702642 | | | |
| 1,161190 | 0,500010 | 1,094610 | 0,362319 | | 1,315760 | 0,527615 | | | |
| 1,197240 | 0,880385 | 0,801172 | 0,457231 | | 1,274490 | 1,038490 | | | |
| 0,661796 | 1,059790 | 1,196990 | 0,282184 | | 0,508420 | 0,932846 | | | |
| 1,019950 | 1,289670 | 0,583461 | 0,310193 | | 0,503665 | 1,217440 | | | |
| 1,120720 | 0,844905 | 1,395930 | 0,298255 | | 1,152160 | 1,109620 | | | |
| 1,246580 | 1,020050 | 0,642283 | 0,425811 | | 1,030000 | 0,755597 | | | |
| 1,162480 | 1,354260 | 0,684691 | 0,335436 | | 1,292130 | 1,458540 | | | |
| 1,409400 | 0,984929 | 1,283780 | 0,299246 | | 0,545488 | 0,939310 | | | |
| 0,553142 | 1,479340 | 0,683930 | 0,336844 | | 0,672744 | 0,638234 | | | |
| | | | | | 1,187690 | 1,127040 | | | |
| | | | | | 1,337430 | 0,742536 | | | |
| | | | | | 0,726555 | 0,706077 | | | |
| | | | | pi lote | | | | | pi lote |
| | | | | 44,697780 | | | | | 30,788690 |
| 13,641448 | 13,887253 | 12,301977 | 4,617102 | 0,250000 | 15,999195 | 14,539495 | | | 0,250000 |
| | | | | pi remanescente | | | | | pi remanescente |
| 44,697780 | 31,056332 | 17,169079 | 4,867102 | 0,250000 | 30,788690 | 30,788690 | 14,789495 | 0,250000 | 0,250000 |
| Lote 57 - Peça B - 16 peças | | | | | Lote 58 - Peça A - 13 peças | | | | |
| M1 | M2 | M3 | M4 | M5 | M1 | M2 | M3 | M4 | M5 |
| 1,228530 | 0,586027 | 0,881976 | 0,356728 | | 0,610213 | 0,612734 | 0,812316 | 0,273097 | 15 seg |
| 0,914864 | 1,430110 | 0,879089 | 0,452009 | | 0,760414 | 1,373550 | 0,768236 | 0,434421 | |
| 1,135130 | 1,292010 | 0,534195 | 0,476492 | | 0,576503 | 1,485410 | 1,165080 | 0,307765 | |
| 0,561497 | 1,485010 | 0,917575 | 0,323296 | | 0,534541 | 1,175820 | 1,274520 | 0,454926 | |
| 1,250770 | 0,708727 | 0,714296 | 0,478875 | | 0,690981 | 1,493390 | 0,862366 | 0,486600 | |
| 0,623092 | 0,742656 | 0,680736 | 0,455048 | | 0,526159 | 1,033570 | 0,596442 | 0,374410 | |
| 1,106030 | 1,388840 | 0,739755 | 0,257830 | | 1,095410 | 0,764491 | 0,683865 | 0,366444 | |
| 0,986437 | 0,911537 | 0,515347 | 0,408309 | | 1,238690 | 0,995966 | 0,988139 | 0,268369 | |
| 1,199270 | 0,655216 | 0,823970 | 0,322202 | | 1,257130 | 1,249220 | 1,150910 | 0,269607 | |
| 1,262440 | 1,065130 | 1,496060 | 0,298665 | | 0,975971 | 0,558350 | 1,058500 | 0,436394 | |
| 0,562850 | 0,855616 | 1,114950 | 0,282506 | | 0,941186 | 0,975485 | 1,188080 | 0,250932 | |
| 0,647845 | 0,806406 | 1,237500 | 0,288314 | | 0,963167 | 0,813510 | 0,727069 | 0,428099 | |
| 0,893309 | 0,505958 | 1,398050 | 0,301476 | | 1,399920 | 0,985862 | 0,852108 | 0,293295 | |
| 0,704740 | 0,915647 | 1,196690 | 0,252248 | | | | | | |
| 0,685506 | 1,489890 | 0,815602 | 0,252736 | | | | | | |
| 1,430510 | 1,065480 | 1,399850 | 0,326586 | | | | | | |
| | | | | pi lote | | | | | pi lote |
| | | | | 51,976041 | | | | | 42,109633 |
| 15,192820 | 15,904260 | 15,345641 | 5,533320 | | 11,570285 | 13,517358 | 12,127631 | 4,644359 | 0,250000 |
| | | | | pi remanescente | | | | | pi remanescente |
| 51,976041 | 36,783221 | 20,878961 | 5,533320 | 0,000000 | 42,109633 | 30,539348 | 17,021990 | 4,894359 | 0,250000 |
| Lote 59 - Peça C - 12 peças | | | | | Lote 60 - Peça A - 10 peças | | | | |
| M1 | M2 | M3 | M4 | M5 | M1 | M2 | M3 | M4 | M5 |
| | 1,013910 | 0,791973 | | 15 seg | 0,899730 | 1,182830 | 0,572892 | 0,411320 | 15 seg |
| | 1,170180 | 1,203810 | | | 0,563056 | 1,133070 | 0,998086 | 0,279455 | |
| | 0,592278 | 0,603181 | | | 0,643360 | 1,096510 | 1,168510 | 0,414383 | |
| | 0,823765 | 1,011340 | | | 1,271950 | 0,793343 | 0,505233 | 0,371822 | |
| | 1,039180 | 1,263480 | | | 0,751796 | 1,474400 | 1,391020 | 0,383816 | |
| | 1,451090 | 1,463410 | | | 1,401200 | 1,147700 | 1,450560 | 0,389125 | |
| | 1,214830 | 0,591585 | | | 1,432830 | 0,702063 | 0,856754 | 0,284342 | |
| | 0,818131 | 0,684487 | | | 1,199510 | 0,585440 | 1,119580 | 0,291846 | |
| | 0,615312 | 1,162040 | | | 1,020150 | 1,410720 | 0,613129 | 0,479134 | |
| | 1,256800 | 0,623664 | | | 0,500632 | 1,487840 | 0,728959 | 0,304621 | |
| | 0,723309 | 0,865757 | | | | | | | |
| | 1,332920 | 1,190570 | | | | | | | |
| | | | | pi lote | | | | | pi lote |
| | | | | 23,757002 | | | | | 33,962717 |
| | 12,051705 | 11,455297 | | 0,250000 | 9,684214 | 11,013916 | 9,404723 | 3,609864 | 0,250000 |
| | | | | pi remanescente | | | | | pi remanescente |
| 23,757002 | 23,757002 | 11,705297 | 0,250000 | 0,250000 | 33,962717 | 24,278503 | 13,264587 | 3,859864 | 0,250000 |

Apêndice E

Tempos de Setup

TABELA E1 – Tempos de *setup* em M1 (torno). Unidade: minutos.

| Tipo de Peça | A | B | C | D | E |
|--------------|-----|-----|-----|-----|-----|
| A | --- | 1,5 | --- | --- | 1,0 |
| B | 1,4 | --- | --- | --- | 0,9 |
| C | --- | --- | --- | --- | --- |
| D | --- | --- | --- | --- | --- |
| E | 1,3 | 1,4 | --- | --- | --- |

TABELA E2 – Tempos de *setup* em M2 (fresa) . Unidade: minutos.

| Tipo de Peça | A | B | C | D | E |
|--------------|-----|-----|-----|-----|-----|
| A | --- | 1,0 | 0,9 | 0,8 | 0,5 |
| B | 1,2 | --- | 1,0 | 1,2 | 1,7 |
| C | 1,1 | 0,9 | --- | 1,0 | 1,3 |
| D | 1,1 | 1,0 | 1,5 | --- | 1,8 |
| E | 1,1 | 1,6 | 1,5 | 1,0 | --- |

TABELA E3 – Tempos de *setup* em M3 (furadeira) . Unidade: minutos.

| Tipo de Peça | A | B | C | D | E |
|--------------|-----|-----|-----|-----|-----|
| A | --- | 1,1 | 1,2 | 0,8 | --- |
| B | 1,2 | --- | 1,2 | 1,4 | --- |
| C | 1,7 | 1,8 | --- | 1,6 | --- |
| D | 1,3 | 1,5 | 1,0 | --- | --- |
| E | --- | --- | --- | --- | --- |

Apêndice F

Resultados da Simulação

Nas tabelas F1 a F17 são apresentados os valores das medidas de desempenho obtidos da simulação para cada uma das 17 combinações de regras de prioridade às máquinas. As medidas de desempenho avaliadas são:

\bar{F} - tempo médio de permanência, por lote, no sistema (minutos);

\bar{T}_f - tempo médio de espera, por lote, em filas (minutos);

%U - porcentagem de utilização média das máquinas;

\bar{L} - atraso médio por lote (minutos).

Tabela F1 – Resultados da combinação 1
M1:slack M2: slack M3: slack M4: slack M5:fifo

| Rodada | \bar{F} | \bar{T}_f | %U | \bar{L} |
|--------|------------|-------------|-----------|-----------|
| 1 | 272,532573 | 26,461863 | 54,905000 | 78,775140 |
| 2 | 264,418606 | 26,055216 | 55,045000 | 69,920077 |
| 3 | 272,269126 | 26,231988 | 54,785000 | 76,126593 |
| 4 | 268,179600 | 25,954686 | 54,880000 | 78,899113 |
| 5 | 269,853674 | 26,235849 | 55,032500 | 75,574470 |
| 6 | 277,191649 | 27,030196 | 54,710000 | 75,763236 |
| 7 | 268,926969 | 26,533756 | 55,172500 | 73,759580 |
| 8 | 270,933068 | 26,354411 | 54,785000 | 77,093622 |
| 9 | 278,065708 | 27,268989 | 54,612500 | 75,584140 |
| 10 | 267,983747 | 26,074596 | 54,827500 | 68,792393 |
| 11 | 272,641849 | 26,801269 | 54,782500 | 69,937971 |
| 12 | 267,714716 | 26,812694 | 54,627500 | 70,427297 |
| 13 | 271,408100 | 26,075419 | 54,962500 | 72,561949 |
| 14 | 273,693500 | 26,355077 | 54,577500 | 78,022538 |
| 15 | 265,534286 | 26,056970 | 55,072500 | 71,991547 |
| 16 | 263,012578 | 25,681975 | 55,027500 | 77,600652 |
| 17 | 268,303783 | 26,264145 | 54,937500 | 77,832221 |
| 18 | 269,751068 | 26,458352 | 55,007500 | 87,582885 |
| 19 | 273,364018 | 26,978412 | 54,687500 | 79,954434 |
| 20 | 273,027129 | 25,946271 | 54,912500 | 77,926977 |

Tabela F2 – Resultados da combinação 2
M1: fifo M2: fifo M3: fifo M4: fifo M5: fifo

| Rodada | \bar{F} | \bar{T}_f | %U | \bar{L} |
|--------|------------|-------------|-----------|------------|
| 1 | 304,536656 | 32,946191 | 56,175000 | 91,885318 |
| 2 | 305,350295 | 34,066949 | 56,395000 | 93,127143 |
| 3 | 309,209865 | 32,886807 | 56,132500 | 94,322139 |
| 4 | 302,071684 | 32,376188 | 56,285000 | 93,060662 |
| 5 | 306,943134 | 32,783297 | 56,352500 | 95,211591 |
| 6 | 313,678109 | 32,061665 | 56,697500 | 94,160441 |
| 7 | 306,478373 | 32,009371 | 56,550000 | 95,161772 |
| 8 | 307,169007 | 33,082995 | 56,462500 | 94,357800 |
| 9 | 317,264327 | 33,878220 | 56,155000 | 94,258024 |
| 10 | 307,106783 | 32,620597 | 56,237500 | 91,698400 |
| 11 | 312,497383 | 32,416496 | 56,270000 | 91,617248 |
| 12 | 310,117475 | 34,260511 | 56,480000 | 94,372860 |
| 13 | 310,215310 | 33,586459 | 56,290000 | 92,564397 |
| 14 | 305,897381 | 32,072906 | 56,482500 | 90,728816 |
| 15 | 306,377173 | 32,768906 | 56,185000 | 94,625435 |
| 16 | 303,422430 | 32,871210 | 56,637500 | 100,195798 |
| 17 | 302,676457 | 33,395932 | 56,170000 | 92,629780 |
| 18 | 301,944010 | 32,420332 | 56,512500 | 98,578353 |
| 19 | 307,072038 | 33,456513 | 56,632500 | 90,921814 |
| 20 | 306,701275 | 33,330239 | 56,425000 | 91,292308 |

Tabela F3 – Resultados da combinação 3
M1: spt M2: spt M3: spt M4: spt M5: fifo

| Rodada | \bar{F} | \bar{T}_f | %U | \bar{L} |
|--------|------------|-------------|-----------|-----------|
| 1 | 261,661432 | 27,092575 | 53,240000 | 66,456877 |
| 2 | 258,628516 | 26,245345 | 53,055000 | 64,066223 |
| 3 | 263,881712 | 25,266778 | 53,177500 | 66,143818 |
| 4 | 259,435062 | 25,889761 | 53,400000 | 69,202604 |
| 5 | 263,664465 | 25,118037 | 53,545000 | 69,787697 |
| 6 | 268,579868 | 26,617256 | 53,202500 | 66,680028 |
| 7 | 259,709184 | 26,488208 | 53,397500 | 63,346769 |
| 8 | 259,628007 | 26,500576 | 52,967500 | 62,481273 |
| 9 | 269,608192 | 27,099448 | 53,200000 | 64,088311 |
| 10 | 262,650950 | 25,713601 | 53,282500 | 63,554495 |
| 11 | 265,947914 | 26,876419 | 52,967500 | 62,043858 |
| 12 | 260,606706 | 26,519874 | 53,212500 | 62,672829 |
| 13 | 265,286057 | 26,368456 | 53,150000 | 66,002346 |
| 14 | 261,039721 | 25,357738 | 53,322500 | 62,548078 |
| 15 | 261,612546 | 25,825154 | 53,420000 | 68,328197 |
| 16 | 256,219671 | 25,635500 | 53,332500 | 68,449702 |
| 17 | 258,344885 | 25,849558 | 53,242500 | 67,157485 |
| 18 | 254,685059 | 25,666570 | 53,357500 | 71,075226 |
| 19 | 263,522506 | 26,467872 | 53,575000 | 69,787388 |
| 20 | 263,831526 | 26,022622 | 53,247500 | 65,873768 |

Tabela F4 – Resultados da combinação 4
M1: edd M2: edd M3: edd M4: edd M5: fifo

| Rodada | \bar{F} | \bar{T}_f | %U | \bar{L} |
|--------|------------|-------------|-----------|------------|
| 1 | 337,505157 | 34,367932 | 56,937500 | 101,575260 |
| 2 | 332,880689 | 33,934847 | 57,142500 | 99,681151 |
| 3 | 338,757443 | 34,274241 | 56,932500 | 101,490947 |
| 4 | 332,551030 | 33,184965 | 57,257500 | 102,035189 |
| 5 | 336,853115 | 34,019407 | 57,077500 | 102,152380 |
| 6 | 342,071451 | 33,634669 | 57,337500 | 100,935206 |
| 7 | 337,249398 | 33,255402 | 57,417500 | 100,823478 |
| 8 | 336,942473 | 34,155407 | 56,882500 | 103,168810 |
| 9 | 347,568056 | 34,016177 | 57,127500 | 102,808357 |
| 10 | 338,434136 | 33,936108 | 57,035000 | 102,430994 |
| 11 | 340,118887 | 34,591610 | 56,900000 | 100,372843 |
| 12 | 337,043761 | 34,014150 | 57,302500 | 102,024769 |
| 13 | 341,415690 | 34,775552 | 56,715000 | 103,179442 |
| 14 | 337,063391 | 33,708040 | 57,107500 | 99,776403 |
| 15 | 337,964320 | 35,011415 | 57,085000 | 102,066695 |
| 16 | 328,549159 | 33,930027 | 57,270000 | 102,195547 |
| 17 | 334,910652 | 33,620304 | 57,077500 | 103,300036 |
| 18 | 329,504435 | 33,309154 | 57,235000 | 102,761718 |
| 19 | 341,269317 | 34,488651 | 56,887500 | 103,962944 |
| 20 | 338,834925 | 34,388594 | 57,172500 | 103,012392 |

Tabela F5 – Resultados da combinação 5
M1: stt M2: stt M3: stt M4: fifo M5: fifo

| Rodada | \bar{F} | \bar{T}_f | %U | \bar{L} |
|--------|------------|-------------|-----------|-----------|
| 1 | 260,407527 | 26,203767 | 53,397500 | 65,103601 |
| 2 | 256,069428 | 25,594565 | 53,622500 | 62,182591 |
| 3 | 263,552932 | 25,428225 | 53,390000 | 66,219739 |
| 4 | 258,331505 | 24,969901 | 53,400000 | 66,936227 |
| 5 | 262,414553 | 25,187382 | 53,520000 | 68,018149 |
| 6 | 270,472654 | 26,376667 | 53,075000 | 68,868493 |
| 7 | 260,295650 | 26,291948 | 53,210000 | 63,651022 |
| 8 | 257,919642 | 25,999339 | 53,315000 | 60,919478 |
| 9 | 271,492483 | 26,986346 | 53,210000 | 67,268170 |
| 10 | 259,773312 | 25,977883 | 52,840000 | 61,315125 |
| 11 | 267,240333 | 26,518770 | 53,240000 | 65,315431 |
| 12 | 262,389822 | 26,882454 | 53,252500 | 66,841027 |
| 13 | 264,130302 | 26,238604 | 53,122500 | 65,873819 |
| 14 | 259,396261 | 25,903972 | 53,410000 | 60,937838 |
| 15 | 258,879973 | 25,924509 | 53,210000 | 64,691217 |
| 16 | 254,758363 | 25,439505 | 53,475000 | 67,187136 |
| 17 | 254,577569 | 26,042658 | 53,410000 | 63,306557 |
| 18 | 255,576340 | 25,740298 | 53,147500 | 71,158111 |
| 19 | 263,985298 | 25,184017 | 53,505000 | 68,026345 |
| 20 | 262,208203 | 26,578849 | 53,172500 | 65,074160 |

Tabela F6 – Resultados da combinação 6
M1: fifo M2: spt M3: fifo M4: fifo M5: fifo

| Rodada | \bar{F} | \bar{T}_f | %U | \bar{L} |
|--------|------------|-------------|-----------|-----------|
| 1 | 262,500945 | 28,589409 | 53,392500 | 56,993179 |
| 2 | 256,504307 | 28,463064 | 53,280000 | 54,971245 |
| 3 | 264,321290 | 28,124846 | 53,367500 | 57,850196 |
| 4 | 257,939625 | 28,050688 | 53,375000 | 56,940434 |
| 5 | 260,612173 | 29,157430 | 53,690000 | 55,917805 |
| 6 | 267,570629 | 27,756499 | 53,737500 | 54,704085 |
| 7 | 261,171697 | 29,753632 | 53,765000 | 56,361700 |
| 8 | 260,638670 | 28,831429 | 53,635000 | 56,786629 |
| 9 | 270,857483 | 29,102399 | 52,970000 | 54,695676 |
| 10 | 260,971331 | 29,042469 | 53,570000 | 53,620533 |
| 11 | 266,638340 | 29,993891 | 53,260000 | 53,134809 |
| 12 | 262,310666 | 29,124356 | 53,375000 | 54,447497 |
| 13 | 265,062969 | 28,904472 | 53,407500 | 56,437721 |
| 14 | 259,832842 | 28,406375 | 53,730000 | 51,188053 |
| 15 | 258,824262 | 29,630434 | 53,692500 | 53,925711 |
| 16 | 257,863641 | 27,692532 | 53,742500 | 60,353855 |
| 17 | 257,165390 | 27,995035 | 53,465000 | 56,370134 |
| 18 | 257,322503 | 28,160963 | 53,595000 | 61,551577 |
| 19 | 262,702304 | 28,995698 | 53,562500 | 56,611308 |
| 20 | 263,319692 | 28,091910 | 53,382500 | 56,054642 |

Tabela F7 – Resultados da combinação 7
M1: fifo M2: edd M3: fifo M4: fifo M5: fifo

| Rodada | \bar{F} | \bar{T}_f | %U | \bar{L} |
|--------|------------|-------------|-----------|------------|
| 1 | 342,781313 | 35,538954 | 57,282500 | 108,248059 |
| 2 | 341,131222 | 36,569618 | 57,242500 | 107,622846 |
| 3 | 343,281345 | 36,475194 | 57,175000 | 107,501039 |
| 4 | 338,510091 | 35,397606 | 57,382500 | 108,119476 |
| 5 | 339,937824 | 34,663930 | 57,362500 | 106,746717 |
| 6 | 345,828536 | 34,911576 | 57,342500 | 106,034982 |
| 7 | 345,027670 | 35,144642 | 57,195000 | 109,592387 |
| 8 | 342,017999 | 34,870158 | 57,210000 | 108,893076 |
| 9 | 347,543830 | 35,645908 | 57,320000 | 106,047549 |
| 10 | 343,434070 | 34,134068 | 57,107500 | 108,332215 |
| 11 | 346,690955 | 35,782707 | 57,200000 | 108,021806 |
| 12 | 345,488461 | 35,296733 | 57,150000 | 110,498480 |
| 13 | 344,233260 | 35,346662 | 57,120000 | 108,116186 |
| 14 | 344,648432 | 35,123309 | 57,295000 | 108,095975 |
| 15 | 342,729758 | 35,703465 | 57,415000 | 107,850588 |
| 16 | 334,588131 | 35,372147 | 57,425000 | 109,555669 |
| 17 | 341,255486 | 35,584810 | 57,010000 | 109,683954 |
| 18 | 336,728496 | 34,992881 | 57,222500 | 110,551173 |
| 19 | 344,984432 | 34,975334 | 57,092500 | 109,336892 |
| 20 | 345,723648 | 36,223480 | 57,045000 | 110,673827 |

Tabela F8 – Resultados da combinação 8
M1: fifo M2: slack M3: fifo M4: fifo M5:fifo

| Rodada | \bar{F} | \bar{T}_f | %U | \bar{L} |
|--------|------------|-------------|-----------|-----------|
| 1 | 267,214529 | 28,012287 | 53,900000 | 59,511594 |
| 2 | 261,588260 | 28,798358 | 53,877500 | 57,312875 |
| 3 | 268,608494 | 28,279406 | 54,007500 | 60,335647 |
| 4 | 264,890477 | 27,683066 | 54,130000 | 63,197088 |
| 5 | 266,147989 | 27,948816 | 54,220000 | 59,176914 |
| 6 | 272,104090 | 29,724144 | 53,845000 | 58,946805 |
| 7 | 265,410192 | 29,089144 | 54,432500 | 59,140714 |
| 8 | 265,626770 | 29,021906 | 53,907500 | 57,921161 |
| 9 | 272,638958 | 29,419427 | 53,960000 | 56,412373 |
| 10 | 266,246233 | 28,032521 | 54,275000 | 56,671122 |
| 11 | 271,138034 | 29,410521 | 53,702500 | 56,048366 |
| 12 | 267,309393 | 28,908876 | 53,980000 | 59,435356 |
| 13 | 269,667659 | 29,407571 | 53,972500 | 59,918526 |
| 14 | 265,558584 | 28,025104 | 54,340000 | 57,774888 |
| 15 | 264,975563 | 29,132018 | 53,862500 | 58,098414 |
| 16 | 264,044765 | 27,787372 | 54,395000 | 65,510783 |
| 17 | 262,977216 | 28,411785 | 54,082500 | 59,789528 |
| 18 | 264,560686 | 27,438553 | 54,377500 | 67,631373 |
| 19 | 269,210374 | 28,591469 | 54,165000 | 59,873407 |
| 20 | 267,593637 | 27,806924 | 54,140000 | 58,828203 |

Tabela F9 – Resultados da combinação 9
M1: fifo M2: stt M3: fifo M4: fifo M5:fifo

| Rodada | \bar{F} | \bar{T}_f | %U | \bar{L} |
|--------|------------|-------------|-----------|-----------|
| 1 | 261,417314 | 29,227604 | 53,290000 | 57,382414 |
| 2 | 256,266010 | 28,638109 | 53,197500 | 56,258560 |
| 3 | 262,445750 | 28,592371 | 53,590000 | 56,765176 |
| 4 | 257,894847 | 28,224435 | 53,492500 | 57,382585 |
| 5 | 260,927778 | 28,884286 | 53,357500 | 57,655883 |
| 6 | 269,194582 | 28,586007 | 53,620000 | 56,046504 |
| 7 | 260,147182 | 29,514318 | 53,765000 | 56,769845 |
| 8 | 260,146550 | 29,454810 | 53,645000 | 57,214934 |
| 9 | 270,589252 | 29,362243 | 53,060000 | 55,654108 |
| 10 | 261,158770 | 28,460094 | 53,767500 | 55,146003 |
| 11 | 266,860030 | 29,405245 | 53,267500 | 54,661892 |
| 12 | 261,516046 | 28,747488 | 53,585000 | 54,219727 |
| 13 | 264,054155 | 28,404115 | 53,630000 | 56,945427 |
| 14 | 260,486474 | 28,601715 | 53,680000 | 52,679348 |
| 15 | 260,329295 | 29,508507 | 53,475000 | 56,861574 |
| 16 | 258,251853 | 27,220629 | 53,822500 | 61,879817 |
| 17 | 258,397289 | 27,632384 | 53,532500 | 58,139585 |
| 18 | 255,860107 | 27,764535 | 53,627500 | 60,292832 |
| 19 | 262,672539 | 29,168120 | 53,615000 | 56,768836 |
| 20 | 262,883711 | 28,506458 | 53,385000 | 56,345500 |

Tabela F10 – Resultados da combinação 10
M1: spt M2: fifo M3: spt M4: spt M5: fifo

| Rodada | \bar{F} | \bar{T}_f | %U | \bar{L} |
|--------|------------|-------------|-----------|-----------|
| 1 | 282,034906 | 29,645807 | 57,085000 | 74,664129 |
| 2 | 275,722272 | 29,731432 | 57,385000 | 69,617023 |
| 3 | 285,650333 | 29,049280 | 56,895000 | 75,795804 |
| 4 | 279,287882 | 28,148750 | 57,357500 | 75,055902 |
| 5 | 283,077621 | 29,370302 | 56,952500 | 76,912579 |
| 6 | 291,161692 | 29,792260 | 57,077500 | 76,588934 |
| 7 | 279,233272 | 30,228149 | 57,115000 | 73,268908 |
| 8 | 280,269865 | 29,399254 | 56,835000 | 73,025117 |
| 9 | 288,088850 | 29,799597 | 56,992500 | 70,390001 |
| 10 | 285,739292 | 29,124475 | 56,825000 | 75,061415 |
| 11 | 288,123271 | 29,985221 | 56,572500 | 72,669277 |
| 12 | 284,706486 | 29,229265 | 57,167500 | 75,239658 |
| 13 | 283,134282 | 29,698494 | 57,012500 | 71,426012 |
| 14 | 280,803023 | 28,955255 | 57,187500 | 71,379640 |
| 15 | 280,684713 | 29,737251 | 56,815000 | 73,995675 |
| 16 | 278,495882 | 29,858580 | 57,497500 | 79,645848 |
| 17 | 276,543063 | 30,514831 | 56,912500 | 72,398074 |
| 18 | 278,794176 | 28,889395 | 56,960000 | 80,071303 |
| 19 | 281,879113 | 28,776731 | 57,305000 | 75,030969 |
| 20 | 281,790532 | 29,887877 | 57,000000 | 71,855848 |

Tabela F11 – Resultados da combinação 11
M1: spt M2: edd M3: spt M4: spt M5: fifo

| Rodada | \bar{F} | \bar{T}_f | %U | \bar{L} |
|--------|------------|-------------|-----------|------------|
| 1 | 340,106984 | 32,423964 | 56,680000 | 105,295296 |
| 2 | 333,850176 | 33,040134 | 56,942500 | 101,172784 |
| 3 | 338,668173 | 33,520116 | 56,820000 | 101,575146 |
| 4 | 335,233679 | 33,259917 | 56,750000 | 105,396411 |
| 5 | 334,696684 | 33,732649 | 56,875000 | 101,724487 |
| 6 | 344,368214 | 33,684631 | 56,640000 | 104,469428 |
| 7 | 339,723923 | 33,411653 | 56,847500 | 104,552127 |
| 8 | 338,257626 | 34,616195 | 56,592500 | 104,682475 |
| 9 | 344,106192 | 33,642623 | 56,780000 | 102,045672 |
| 10 | 337,593955 | 33,195057 | 56,527500 | 103,131013 |
| 11 | 342,427085 | 33,180014 | 56,730000 | 104,403218 |
| 12 | 338,818313 | 32,924913 | 56,932500 | 104,430208 |
| 13 | 341,450631 | 33,716320 | 56,807500 | 104,825917 |
| 14 | 339,543685 | 33,360317 | 56,612500 | 102,208090 |
| 15 | 333,947934 | 33,006648 | 56,980000 | 100,764475 |
| 16 | 330,411782 | 33,415296 | 56,805000 | 105,681427 |
| 17 | 335,783859 | 33,472159 | 56,822500 | 104,130427 |
| 18 | 329,659232 | 33,109666 | 56,765000 | 104,410650 |
| 19 | 339,782404 | 33,949301 | 56,565000 | 104,335349 |
| 20 | 339,831794 | 33,918223 | 56,647500 | 105,821645 |

Tabela F12 – Resultados da combinação 12
M1: spt M2: slack M3: spt M4: spt M5:fifo

| Rodada | \bar{F} | \bar{T}_f | %U | \bar{L} |
|--------|------------|-------------|-----------|-----------|
| 1 | 277,102527 | 26,364534 | 54,547500 | 81,876013 |
| 2 | 275,087717 | 25,954838 | 54,745000 | 81,109619 |
| 3 | 279,870607 | 26,104028 | 54,465000 | 84,042076 |
| 4 | 274,953634 | 26,630640 | 54,797500 | 86,053609 |
| 5 | 275,185978 | 26,246477 | 54,812500 | 80,653697 |
| 6 | 282,224319 | 27,663737 | 54,487500 | 80,409210 |
| 7 | 276,363194 | 27,089603 | 54,892500 | 81,443506 |
| 8 | 274,581219 | 26,443688 | 54,300000 | 79,214085 |
| 9 | 284,187486 | 27,486034 | 54,535000 | 79,666469 |
| 10 | 223,901596 | 26,461150 | 54,550000 | 81,216212 |
| 11 | 279,416259 | 27,677688 | 54,407500 | 77,486675 |
| 12 | 275,301943 | 27,258039 | 54,737500 | 78,683564 |
| 13 | 278,669832 | 27,480647 | 54,690000 | 80,867501 |
| 14 | 275,026072 | 26,684851 | 54,537500 | 79,167317 |
| 15 | 273,185114 | 26,578685 | 54,862500 | 79,944268 |
| 16 | 272,554514 | 26,279314 | 54,912500 | 86,883366 |
| 17 | 275,965040 | 26,800413 | 54,730000 | 86,070370 |
| 18 | 276,570030 | 26,403512 | 55,142500 | 94,008353 |
| 19 | 281,593789 | 27,774304 | 54,705000 | 89,026755 |
| 20 | 274,769447 | 27,009090 | 54,690000 | 79,236480 |

Tabela F13 – Resultados da combinação 13
M1: spt M2: stt M3: spt M4: spt M5:fifo

| Rodada | \bar{F} | \bar{T}_f | %U | \bar{L} |
|--------|------------|-------------|-----------|-----------|
| 1 | 259,266141 | 26,410436 | 53,352500 | 63,683300 |
| 2 | 258,267221 | 26,205849 | 53,337500 | 64,229704 |
| 3 | 262,785212 | 25,388649 | 53,395000 | 64,673429 |
| 4 | 257,794920 | 25,986373 | 53,212500 | 66,342405 |
| 5 | 263,382837 | 25,573178 | 53,192500 | 70,160308 |
| 6 | 267,369641 | 26,650948 | 53,025000 | 65,075060 |
| 7 | 259,997858 | 26,110304 | 53,362500 | 63,380828 |
| 8 | 259,762163 | 26,637842 | 52,900000 | 63,133099 |
| 9 | 270,319975 | 27,013233 | 53,157500 | 64,298253 |
| 10 | 261,835521 | 25,879531 | 53,097500 | 63,563886 |
| 11 | 265,658118 | 26,989155 | 53,227500 | 62,702144 |
| 12 | 261,900487 | 25,825163 | 53,297500 | 64,139356 |
| 13 | 262,999022 | 26,561792 | 53,292500 | 64,118897 |
| 14 | 259,202139 | 25,905665 | 53,265000 | 61,317001 |
| 15 | 262,949141 | 25,805523 | 53,082500 | 69,226528 |
| 16 | 258,198143 | 25,409739 | 53,167500 | 70,644782 |
| 17 | 257,845378 | 26,684772 | 53,172500 | 65,978651 |
| 18 | 255,023753 | 25,569831 | 53,237500 | 70,348771 |
| 19 | 264,754175 | 26,216994 | 53,485000 | 71,200399 |
| 20 | 263,264273 | 26,002145 | 53,300000 | 65,794049 |

Tabela F14 – Resultados da combinação 14
M1: edd M2: fifo M3: edd M4: edd M5: fifo

| Rodada | \bar{F} | \bar{T}_f | %U | \bar{L} |
|--------|------------|-------------|-----------|------------|
| 1 | 323,057593 | 33,352918 | 56,845000 | 99,492748 |
| 2 | 321,102630 | 33,458802 | 57,090000 | 97,511940 |
| 3 | 322,919564 | 33,116331 | 56,722500 | 98,815146 |
| 4 | 320,264510 | 32,476679 | 57,022500 | 100,459926 |
| 5 | 323,862439 | 32,455600 | 56,777500 | 102,482582 |
| 6 | 326,818231 | 31,919226 | 57,045000 | 97,920216 |
| 7 | 323,186642 | 32,431386 | 57,005000 | 100,533414 |
| 8 | 323,057873 | 33,568099 | 57,440000 | 99,818794 |
| 9 | 328,737335 | 33,625764 | 57,145000 | 95,624195 |
| 10 | 325,970056 | 32,670544 | 56,837500 | 101,297617 |
| 11 | 333,512478 | 34,409779 | 56,662500 | 102,424803 |
| 12 | 325,854103 | 33,249508 | 57,292500 | 102,648523 |
| 13 | 326,819378 | 32,704481 | 57,127500 | 99,399212 |
| 14 | 326,936239 | 33,929840 | 57,047500 | 102,895385 |
| 15 | 325,803317 | 34,169010 | 56,897500 | 102,986953 |
| 16 | 316,056559 | 32,035749 | 57,050000 | 104,446513 |
| 17 | 321,274536 | 33,294923 | 56,845000 | 100,905443 |
| 18 | 313,886230 | 32,152404 | 56,887500 | 101,559583 |
| 19 | 325,395941 | 33,103231 | 57,080000 | 100,703725 |
| 20 | 324,611789 | 32,402265 | 56,810000 | 99,910086 |

Tabela F15 – Resultados da combinação 15
M1: edd M2: spt M3: edd M4: edd M5: fifo

| Rodada | \bar{F} | \bar{T}_f | %U | \bar{L} |
|--------|------------|-------------|-----------|-----------|
| 1 | 271,361414 | 27,172888 | 53,452500 | 51,109670 |
| 2 | 269,638944 | 28,144547 | 53,232500 | 48,274037 |
| 3 | 272,419443 | 26,448802 | 52,670000 | 51,888409 |
| 4 | 265,678674 | 26,771343 | 53,365000 | 48,886611 |
| 5 | 269,572493 | 27,742396 | 53,112500 | 50,979961 |
| 6 | 224,558472 | 27,541326 | 53,587500 | 50,951848 |
| 7 | 270,797492 | 27,745954 | 53,477500 | 51,308943 |
| 8 | 272,790345 | 28,359601 | 53,150000 | 52,183246 |
| 9 | 279,993530 | 28,097914 | 52,962500 | 49,125905 |
| 10 | 275,086579 | 26,884190 | 53,332500 | 53,728353 |
| 11 | 274,441950 | 27,863122 | 52,672500 | 47,451413 |
| 12 | 271,723755 | 27,184844 | 53,100000 | 50,988539 |
| 13 | 271,113777 | 26,863959 | 53,040000 | 47,230764 |
| 14 | 273,898581 | 27,557269 | 53,185000 | 51,519382 |
| 15 | 271,300550 | 28,690250 | 53,302500 | 51,213251 |
| 16 | 267,678608 | 27,389129 | 53,190000 | 59,058673 |
| 17 | 268,778123 | 27,044243 | 53,172500 | 50,699789 |
| 18 | 264,325384 | 27,117486 | 53,430000 | 53,565834 |
| 19 | 270,571654 | 28,080701 | 53,272500 | 49,063310 |
| 20 | 274,848478 | 27,070298 | 53,100000 | 54,075862 |

Tabela F16 – Resultados da combinação 16
M1: edd M2: slack M3: edd M4: edd M5:fifo

| Rodada | \bar{F} | \bar{T}_f | %U | \bar{L} |
|--------|------------|-------------|-----------|-----------|
| 1 | 267,831672 | 26,014840 | 54,322500 | 50,706151 |
| 2 | 266,970850 | 26,666979 | 54,545000 | 50,215441 |
| 3 | 272,519975 | 26,707574 | 54,400000 | 52,286664 |
| 4 | 264,324893 | 26,474840 | 54,927500 | 50,573183 |
| 5 | 268,855913 | 26,768770 | 54,452500 | 53,556545 |
| 6 | 272,436100 | 26,286971 | 54,482500 | 48,931445 |
| 7 | 272,055037 | 27,271378 | 54,800000 | 54,947883 |
| 8 | 271,761134 | 26,800052 | 54,690000 | 55,153188 |
| 9 | 275,916496 | 27,538065 | 54,385000 | 52,048498 |
| 10 | 275,372800 | 27,028200 | 54,627500 | 55,417543 |
| 11 | 273,160635 | 26,879199 | 54,490000 | 50,784333 |
| 12 | 271,037024 | 26,784678 | 54,535000 | 51,649525 |
| 13 | 271,837396 | 27,092354 | 54,320000 | 48,250050 |
| 14 | 267,756780 | 26,604019 | 54,645000 | 52,105378 |
| 15 | 269,950282 | 27,380620 | 54,445000 | 52,086925 |
| 16 | 264,586043 | 26,128310 | 54,725000 | 57,588189 |
| 17 | 265,277998 | 26,407571 | 54,540000 | 51,251526 |
| 18 | 263,584306 | 25,945447 | 54,675000 | 56,573235 |
| 19 | 271,666293 | 26,968568 | 54,422500 | 51,807640 |
| 20 | 274,655993 | 26,752107 | 54,710000 | 55,254082 |

Tabela F17 – Resultados da combinação 17
M1: edd M2: stt M3: edd M4: edd M5:fifo

| Rodada | \bar{F} | \bar{T}_f | %U | \bar{L} |
|--------|------------|-------------|-----------|-----------|
| 1 | 271,522258 | 26,803842 | 53,315000 | 52,651546 |
| 2 | 268,075395 | 27,656185 | 53,287500 | 47,886576 |
| 3 | 273,020441 | 26,795659 | 52,847500 | 51,623746 |
| 4 | 265,853190 | 27,391907 | 53,320000 | 49,210208 |
| 5 | 269,979692 | 27,610751 | 53,430000 | 51,481854 |
| 6 | 277,746995 | 27,337395 | 53,525000 | 50,467323 |
| 7 | 272,062680 | 27,809907 | 53,602500 | 52,757237 |
| 8 | 272,905786 | 27,791980 | 53,270000 | 52,075822 |
| 9 | 277,309292 | 27,296251 | 52,972500 | 48,540322 |
| 10 | 275,728500 | 26,945343 | 53,257500 | 53,788000 |
| 11 | 277,264249 | 28,310729 | 52,685000 | 48,772162 |
| 12 | 269,956878 | 26,999531 | 53,242500 | 49,496558 |
| 13 | 270,493259 | 26,099805 | 53,027500 | 49,150580 |
| 14 | 272,091264 | 27,124068 | 53,045000 | 50,106640 |
| 15 | 272,606746 | 29,069633 | 53,132500 | 51,831844 |
| 16 | 267,233582 | 27,462771 | 53,235000 | 58,530901 |
| 17 | 267,640776 | 26,957803 | 52,945000 | 50,198276 |
| 18 | 264,611127 | 26,758999 | 53,680000 | 53,903419 |
| 19 | 270,862331 | 27,993829 | 53,082500 | 49,398606 |
| 20 | 273,323156 | 26,749756 | 53,157500 | 53,077736 |

Apêndice G

Exemplo de Arquivo de Entrada da Rede Neural

SNNS pattern definition file V3.2
generated at Qui Jan 13 00:15:00 2000

No. of patterns : 17
No. of input units : 04
No. of output units : 05

```
# Input pattern 1:
0.3864350.4662350.1679180.499329
# Output pattern 1:
0.30.30.30.30.1
# Input pattern 2:
0.6304320.7317970.5898740.648499
# Output pattern 2:
0.10.10.10.10.1
# Input pattern 3:
0.3439530.2242880.1589840.366033
# Output pattern 3:
0.50.50.50.50.1
# Input pattern 4:
0.8349030.8071980.7481560.78527
# Output pattern 4:
0.70.70.70.70.1
# Input pattern 5:
0.3361570.2205590.1850960.325372
# Output pattern 5:
0.90.90.90.10.1
# Input pattern 6:
0.3484690.2168310.3865210.191001
# Output pattern 6:
0.10.50.10.10.1
...
# Input pattern 14:
0.7414430.8904710.6929940.763121
# Output pattern 14:
0.70.10.70.70.1
# Input pattern 15:
0.4070180.2296740.2519340.148912
# Output pattern 15:
0.70.50.70.70.1
# Input pattern 16:
0.4145130.3866910.2198420.163754
# Output pattern 16:
0.70.30.70.70.1
# Input pattern 17:
0.4014020.1592440.1779260.124208
# Output pattern 17:
0.70.90.70.70.1
```