

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
CURSO DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

**Modelos Analíticos para
Interconexão de Processadores:
Avaliação de Desempenho de
Alocação Simultânea de
Recursos**

por

Paulo Henrique Lemelle Fernandes

Dissertação submetida como requisito parcial
para a obtenção do grau de
Mestre em Ciência da Computação

DBW



UFRGS

SABi



05232072

Prof. Philippe Olivier Alexandre Navaux
Orientador

Prof. Maurizio Tazza
Co-orientador

Porto Alegre, dezembro de 1990

UFRGS
INSTITUTO DE INFORMÁTICA
BIBLIOTECA

CIP - CATALOGAÇÃO NA PUBLICAÇÃO

Fernandes, Paulo Henrique Lemelle

Modelos Analíticos para Interconexão de Processadores:
Avaliação de Desempenho de Alocação Simultânea de Re-
cursos / Paulo Henrique Lemelle Fernandes. -
Porto Alegre: CPGCC da UFRGS, 1990.

129 p. : il.

Dissertação (mestrado)—Universidade Federal do Rio
Grande do Sul, Curso de Pós-Graduação em Ciência da
Computação, Porto Alegre, 1990. Orientador: Navaux, Phi-
lippe Olivier Alexandre; Co-orientador: Tazza, Maurizio

Dissertação: Avaliação de Desempenho, Processamento Pa-
ralelo, Métodos Analíticos
Interconexões de Processadores, Redes de Filas de Espera,
Redes de Petri Temporizadas

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
Sistema de Biblioteca da UFRGS

5934

FERNANDES, PAULO HENRIQUE
LEMELLE

MODELOS ANALITICOS PARA
INTERCONEXAO DE PROCESSADORES

681.32.061(043)
F363M

INF
1993/60366-9
1993/08/02

Penso, logo existo.

René Descartes

Acho que penso; logo acho que existo.

Ambrose Bierce

Alcachofra não pensa, logo não existe.

Luiz Fernando Veríssimo

Solução. Por mais que você pense, não existe.

Anônimo

AGRADECIMENTOS

Esta é sem dúvida a melhor página para se escrever, por três motivos.

Primeiro: é a página mais lida. Qualquer um pode entendê-la e o conteúdo tem alguma chance de não ser chato. Ainda por cima, as coisas escritas nesta página tem aquele sabor de confiança, aquele ar de intimidade.

Segundo: ninguém corrige o seu conteúdo. É uma página a prova de comitês de leitura e regras de escrita. Evidentemente, o medo de expor um pouco mais de si a sabe-se lá quem venha a ler esta dissertação limita um pouco a expressão (sem falar na ortografia).

Terceiro: podemos deixar registrado para as pessoas que foram importantes o nosso agradecimento.

Curiosamente, acho que o terceiro motivo, além de possuir uma menor importância, é dúbio. Durante um trabalho do tamanho que uma dissertação de mestrado costuma ter, é humanamente impossível lembrar de todas as pessoas que realmente nos ajudaram. Infelizmente sempre existem pessoas iludidas, ou (pior) realmente auto-suficientes, que pensam que citar quatro ou cinco nomes basta para cobrir todos aqueles que os ajudaram.

Talvez meus motivos próprios para não adotar esta solução não sejam tão nobres, mas enfim decidi não citar um nome sequer nestes agradecimentos. Espero que mesmo assim, aqueles aos quais agradeço intimamente o saibam, seja pela consciência do auxílio direto a confecção deste volume, seja pela amizade e amor demonstrados em outras situações. Afinal para escrever uma dissertação é preciso, ou é pelo menos recomendável, ter uma vida a parte do trabalho de pesquisa.

Em outras palavras: "obrigado" (a quase todos).

SUMÁRIO

LISTA DE ABREVIATURAS	9
LISTA DE FIGURAS	10
LISTA DE TABELAS	12
RESUMO	13
ABSTRACT	15
RESUMÉ	17
1 INTRODUÇÃO	19
1.1 Métodos de Avaliação de Desempenho	20
1.2 Etapas de Avaliação de Desempenho	25
1.2.1 Modelagem	26
1.2.2 Extração	28
1.2.3 Interpretação	30
1.3 Classes de Problemas	31
1.4 Descrição dos Capítulos	33
2 MÉTODO RFE	36
2.1 Modelagem	37
2.1.1 Estações de Serviço	37
2.1.2 Redes de Filas de Espera	41
2.2 Extração	42
2.2.1 Redes Fechadas, Mono-classe, Mono-servidor	45
2.2.2 Redes Fechadas, Mono-classe, Multi-servidores	48

3	MÉTODO RdPt	52
3.1	Modelagem	53
3.1.1	Conceitos Básicos	54
3.1.2	Escopo de Aplicação do <i>Modelo Q</i>	58
3.2	Extração	60
3.2.1	Notação Adotada	61
3.2.2	<i>Modelo Q1</i>	62
3.3	Métodos Aproximativos	66
3.3.1	Utilização Não-uniforme de Recursos	67
3.3.2	Alocação Não-uniforme de Recursos	70
4	INTERCONEXÕES DE PROCESSADORES	74
4.1	Classificação de Interconexões	74
4.2	Interconexões Multiprocessador	77
4.2.1	Interconexões Multi-estágios	79
4.3	Análise Quantitativa	81
5	MODELAGEM ATRAVÉS DE RFE	84
5.1	Modelo Fundamental	85
5.1.1	Recurso Primário e Ciclo Básico	85
5.1.2	Resultados e Limitações	87
5.2	Modelo Refinado	88
5.2.1	Colisão sobre Memórias	89
5.2.2	Colisão sobre Arbitradores	91
5.3	Limitações	93

6	MODELAGEM ATRAVÉS DE RdPt	95
6.1	Modelo Fundamental	96
6.1.1	Tarefa Básica	96
6.1.2	Recursos	98
6.1.3	Resultados e Limitações	99
6.2	Modelo Refinado	100
6.2.1	Colisões sobre Caminhos e Memórias	101
6.2.2	Tempos de Utilização Variáveis	102
6.2.3	Limitações	103
7	ESTUDO COMPARATIVO	105
7.1	Facilidade de Modelagem	105
7.1.1	Primitivas de Modelagem	105
7.1.2	Técnicas de Construção de Modelos	107
7.2	Velocidade de Extração	108
7.2.1	Modelos RFE	108
7.2.2	Modelos RdPt	109
7.2.3	Orientação	111
7.3	Comparação Numérica	112
7.3.1	Paradigma de Comparação	112
7.3.2	Estudo de um Caso Prático	114
8	CONCLUSÃO	117
8.1	Diferenciação dos Métodos	118
8.2	Trabalhos Práticos Associados	120

8.3 Trabalhos Futuros	122
ANEXO A-1 ALGORITMO PARA CÁLCULO DA CONSTANTE DE NORMALIZAÇÃO	124
BIBLIOGRAFIA	126

LISTA DE ABREVIATURAS

ADMP	Avaliação de Desempenho de Máquinas Paralelas
FIFO	<i>First In, First Out</i> – Disciplina de Fila
RdPt	Método baseado em Redes de Petri Temporizadas
RFE	Método baseado em Redes de Filas de Espera
SARA	Sistema de Análise de Redes e Arquiteturas
SARF	Sistema de Análise de Redes de Filas Fechadas
SQ1	Sistema de Análise de Modelos Q1
SSIP	Sistema Simulador de Interconexões de Processadores

LISTA DE FIGURAS

Figura 2.1	Estação de Serviço	38
Figura 2.2	Redes de Filas de Espera	41
Figura 2.3	Redes Abertas e Fechadas	42
Figura 2.4	Exemplo de Rede Mono-servidor	46
Figura 3.1	Estruturas de Redes de Petri	54
Figura 3.2	Redes de Petri	55
Figura 3.3	Disparos de Transições	56
Figura 3.4	Temporização de Lugares	57
Figura 3.5	<i>Modelo Q</i>	58
Figura 3.6	Cadeias Principais	59
Figura 3.7	Modelos <i>Q1</i>	60
Figura 3.8	Exemplo do <i>Modelo Q1</i>	63
Figura 3.9	Rede com Tempos de Utilização Variáveis	69
Figura 4.1	Taxonomia de Interconexões de Processadores	75
Figura 4.2	Classes de Multiprocessador	77
Figura 4.3	Crossbar e Multi-estágios	79
Figura 4.4	Multi-estágios Uniformes e Não-uniformes	80
Figura 4.5	Interconexões Multi-estágios Tipo Omega	80
Figura 5.1	Modelo Fundamental em RFE	87
Figura 5.2	Modelo de Colisão sobre Memórias em RFE	90
Figura 5.3	Modelo de Colisão sobre Arbitradores em RFE	91
Figura 5.4	Funcionamento de Arbitradores	94
Figura 6.1	Modelo da Tarefa Principal em RdPt	97

Figura 6.2	Modelo Fundamental em RdPt	99
Figura 6.3	Modelo de Colisão de Caminhos	101
Figura 7.1	Resultados Numéricos	116

LISTA DE TABELAS

Tabela 1.1	Classificação de Métodos de Avaliação de Desempenho	21
Tabela 1.2	Características Genéricas dos Métodos de Avaliação	23
Tabela 1.3	Classificação Genérica de Problemas	31
Tabela 2.1	Resultados Obtidos para o Exemplo Mono-servidor	48
Tabela 2.2	Resultados Obtidos para o Exemplo Multi-servidores	51
Tabela 3.1	Resultados para a Rede com Utilização Variável de Recursos	68
Tabela 3.2	Resultados para a Rede com Duas Tarefas de Tempos Variáveis	70
Tabela 3.3	Resultados para a Rede com Alocação Variável de Recursos	71
Tabela 5.1	Resultados para o Modelo Fundamental de RFE	88
Tabela 5.2	Resultados para o Modelo com Colisão sobre Memórias em RFE	91
Tabela 5.3	Resultados para o Modelo com Colisão sobre Arbitradores em RFE	93
Tabela 6.1	Resultados para o Modelo Fundamental em RdPt	100
Tabela 6.2	Resultados para o Modelo com Colisões em RdPt	102
Tabela 7.1	Resultados para o Exemplo Prático	115

RESUMO

1,5 a 2 folhas

Descreve-se a aplicação de dois métodos analíticos para um mesmo problema com alocação simultânea de recursos: a avaliação de desempenho de interconexões de processadores do tipo multi-estágios. Aplica-se o método analítico tradicional com redes de filas de espera e cadeias de Markov. Utiliza-se após um método com modelagem através de redes de Petri temporizadas e extração através de ponderação de resultados determinísticos. Os modelos desenvolvidos por cada um dos métodos são comparados segundo três diferentes aspectos: facilidade de modelagem, velocidade de extração e acuracidade de resultados.

O primeiro método de avaliação empregado, método analítico tradicional, consiste no uso de redes de filas de espera (teoria de filas) para a modelagem e do algoritmo de convolução (cálculo da constante de normalização) para a extração de resultados. A abordagem de resolução é clássica (abordagem estocástica) resultando nas fórmulas derivadas do modelo de cadeias de Markov.

O método de avaliação aplicado em seguida está baseado no desenvolvimento de modelos em redes de Petri temporizadas. Este método se divide na aplicação de um método analítico puro (*modelo Q*) e na aplicação de métodos aproximativos. Estes métodos aproximativos consistem na decomposição de um problema com comportamento variável em diversos problemas com comportamento determinístico e probabilidades associadas.

Os modelos para as interconexões multi-estágios em ambos os métodos já citados são desenvolvidos com diferentes graus de refinamento. Modelos básicos com imprecisões de descrição são refinados até ser estabelecido um modelo tão próximo quanto possível da realidade descrita (interconexão multi-estágios).

A comparação de todos os modelos desenvolvidos é feita em três etapas visando estabelecer quais modelos são adequados à avaliação de desempenho de interconexões multi-estágios. A adequação deve considerar aspectos como facilidade de desenvolvimento do modelo, rapidez e precisão na obtenção de resultados. Estas informações são consideradas segundo o tamanho do problema (dimensão da interconexão), o enfoque de modelagem (parâmetros relevantes) e os objetivos qualitativos da avaliação (o que se pretende descobrir).

→ PALAVRAS-CHAVE:
M NEG

A bcd, —

ABSTRACT

Main's N
Title: Analytic Models for Processors Interconnections: Performance Evaluation of Simultaneous Resource Allocation

Main's cells
1/2/1/2/1/2

This work describes the application of two analytic models to the multistage processor interconnections, a simultaneous resource allocation problem. First, the traditional method based on queueing networks and Markov chains is applied. After, a Petri nets based method with deterministic weighed results is used. The models developed in each method are compared in three levels: modeling facility, evaluation speed and precision.

The first evaluation method used is based on the queueing network classic method (queueing theory) to model and the convolution algorithm (normalization constant calculus) to obtain results. The classic stochastic approach is used through formules derived from Markov chains.

The second method used is based on temporised Petri nets models. This method has two parts: the application of a pure analitical method (model Q) and the application of approximatives methods. This approximatives methods decompose a problem with non-fixed behaviour in several weighted problems with deterministic behaviour.

The multistage interconnections models are developped with different degrees of accuracy for both methods cited. Basic models without description precision are refined until as accurate as possible model is obtained for the choosen reality (multistage interconnections).

The comparison of all developped models is made in three steps aiming the qualitative judgment of the models for multistages performance evaluation. This judgment should consider the dificulty to develop the model, the speed

and the precision of the results. These informations are considered taking into account the the problem size (number of elements in the interconnection), the modelling approach (relevant parameters) and the the qualitative evaluation goals (the evaluation purpose).

KEYWORDS: -

RESUMÉ

Titre: Modèles Analytiques pour les Interconnexions de Processeurs: Evaluation des Performances d'Allocation Simultanée de Ressources

On décrit l'application de deux méthodes analytiques à un même problème avec l'utilisation simultanée de ressources, l'évaluation des performances des interconnexions de processeurs du type multistage. On applique d'abord la méthode traditionnelle utilisant les réseaux de files d'attente et les chaînes de Markov. Après, on utilise une méthode avec modélisation basée sur les réseaux de Petri et l'extraction à travers des pondérations des résultats déterministes. Les modèles développés par chacun des méthodes sont comparés selon trois aspects distincts: facilité de modélisation, vitesse d'évaluation et précision des résultats.

La première méthode employée, la méthode analytique traditionnelle, utilise les réseaux de files d'attente (théorie des réseaux) pour faire la modélisation et l'algorithme de convolution (constante de normalisation) pour faire l'extraction des résultats. La résolution est faite à travers l'approche stochastique traditionnelle avec les formules dérivées des chaînes de Markov.

La méthode d'évaluation appliquée après est basée sur les développement des modèles en réseaux des Petri temporisées. Cette méthode est composée de deux parties: l'application d'une méthode analytique pure (modèle Q) et l'application des méthode approximatives. Ces méthodes approximatives font la décomposition d'un problème avec comportement variable en plusieurs problèmes avec comportement déterministes et probabilités associées.

Les modèles des interconnexions multistage ont été développées avec plusieurs degrés de raffinement dans tout les deux méthodes déjà citées. Les

modèles de base avec des imprecisions ont été raffinés jusqu'à avoir une description la plus proche possible de la réalité (les interconnexions multistage).

La comparaison de tout les modèles développés est faite en trois parties pour établir les modèles les plus adéqués à l'évaluation des performances des interconnexions multistage. L'adequation doit envisager la facilité de développement du modèle, la vitesse et la précision des résultats. Ces informations sont prises selon la taille du problème (dimension de l'interconnexion), l'enfoque de modélisation (les paramètres à considerer) et les objectifs qualitatifs de l'évaluation (ce qu'on veut savoir).

1 INTRODUÇÃO

Uma exigência de nossa sociedade tecnológica é o planejamento de todos os empreendimentos. Uma das bases deste planejamento é a previsão das consequências das eventuais decisões. Com base nestas duas premissas, é possível afirmar que a avaliação de desempenho, na qualidade de ferramenta científica de previsão, é uma necessidade de nossos dias. Na realidade, a avaliação de desempenho, que é o tema desta dissertação, não é necessariamente uma ferramenta de previsão. Trata-se na verdade de uma ferramenta de análise quantitativa, cuja função precípua é exprimir, através de grandezas, certas características que permitam um julgamento qualitativo de um sistema real (uma realidade) qualquer.

A existência de diversos métodos para a avaliação de desempenho de diversos tipos de problemas sugere a possibilidade de uma maior ou menor adequação de determinados métodos a determinados tipos de problemas. Um estudo detalhado de cada método é interessante para a determinação daquele que será adequado ao problema a resolver. Neste sentido, o objetivo deste trabalho é o estabelecimento de uma comparação entre dois métodos de avaliação de desempenho de sistemas. Trata-se na verdade de dois métodos analíticos que possuem processos de resolução completamente diferentes.

O primeiro método utiliza uma abordagem estocástica tradicional baseada na descrição de estados do sistema através de *cadeias de Markov*. O segundo método baseia-se em uma abordagem probabilística *ad hoc* para a ponderação da análise determinística de diversos modos de utilização de recursos do sistema. As ferramentas gráficas de modelagem (descrição do sistema a ser analisado) são igualmente distintas. Respectivamente, os métodos utilizam redes de filas de espera e redes de Petri temporizadas. No decorrer desta dissertação serão utilizadas as seguintes denominações: **RFE** e **RdPt**. Tais abreviações foram escolhidas com

base nos formalismos de modelagem adotados (Redes de Filas de Espera e Redes de Petri temporizadas).

Ambos os métodos são aplicados a um mesmo problema onde existe alocação simultânea de recursos. Um tipo de interconexão de processadores foi escolhido para representar esta classe de problema. Avaliam-se, especificamente, interconexões classificadas como multiprocessador do tipo multi-estágios [AND75]. Dois modelos, baseados cada um deles sobre um dos métodos, são desenvolvidos para esta realidade e resultados numéricos de um caso prático real são obtidos. Com base nos dois modelos, bem como nos resultados gerados, desenvolve-se a comparação dos métodos.

A comparação propriamente dita se dá em três níveis distintos: a capacidade de modelagem, velocidade de extração (obtenção de resultados) e precisão numérica. Uma comparação subjetiva é feita no nível de capacidade de modelagem. A comparação de velocidade de extração é feita com base no número de iterações necessário para extração de resultados de cada um dos modelos. Finalmente os resultados obtidos por cada um dos modelos são comparados aos resultados de simulação, especificando a precisão de cada método. Desta forma é possível estabelecer algumas conclusões sobre a adequação de cada método aos problemas com alocação simultânea de recursos. Cabe salientar que todas estas considerações não estão diretamente vinculadas aos métodos em si, mas aos modelos neles baseados para uma mesma realidade exemplo (multiprocessadores multi-estágios).

1.1 Métodos de Avaliação de Desempenho

Esta seção tem por objetivo expor uma caracterização genérica dos tipos de métodos de avaliação de desempenho de sistemas frequentemente empregados. Não pretende-se encaixar cada método numa classe (tipo) específica,

mas sim enumerar algumas características marcantes de cada tipo de método de avaliação. Cabe salientar que os métodos encontrados na prática são frequentemente híbridos, possuindo características comuns a dois ou mais tipos de métodos distintos. A tabela 1.1 apresenta a classificação genérica dos grandes grupos de métodos (monitoração, simulação e análise) frequentemente adotada [ALL78, FER78, GEL80, LAZ84, BRA87].

Métodos de Avaliação de Desempenho		
Métodos Elementares	Métodos Indiretos	
Monitoração Benchmark, etc	Simulação p/ eventos, etc	Análise filas, Petri, etc

Tabela 1.1: Classificação de Métodos de Avaliação de Desempenho

A maneira mais simples de avaliar uma certa realidade é medi-la diretamente durante seu funcionamento. Independente da complexidade física envolvida na medição direta, a nível de complexidade de método de avaliação, esta opção é a solução mais elementar. Exatamente por isto este gênero de avaliação denomina-se métodos elementares. Quando aplicados durante o desenvolvimento de um projeto, os métodos elementares tem como desvantagem a necessidade de um protótipo para avaliar. Este tipo de avaliação é feito sem nenhuma abstração da realidade, podendo ser empregada a denominação métodos diretos. Note-se que mesmo se considerarmos o uso de um protótipo este será sempre uma cópia física da realidade.

Além dos métodos elementares, existem métodos que exigem o desenvolvimento de modelos (modelagem). Nestes métodos o modelo desenvolvido é o alvo da medição numa etapa posterior (extração). Estas duas etapas, modelagem e extração, irão compor a aplicação dos métodos com desenvolvimento de modelos, denominados então métodos indiretos. A qualidade destes métodos dependerá tanto do modelo desenvolvido (modelagem) quanto do tratamento a ele dado (extração). Para englobar as etapas de modelagem e extração o termo avaliação será empregado. Os métodos indiretos dividem-se em dois grandes

grupos: as simulações e os métodos analíticos. Estes dois tipos tem, em princípio, o mesmo tipo de modelagem, sua diferença significativa se manifesta a nível de extração.

Os métodos de simulação [FER78, BRA87] são semelhantes aos métodos elementares, pois a extração baseia-se na observação e medição do funcionamento do modelo. Tanto o funcionamento, quanto a medição são, no entanto simulados de maneira automática (não-física). A grande vantagem das simulações em comparação aos métodos elementares é a nível da facilidade de extração (medição). Cabe salientar porém que este tipo de medição é feito sobre um modelo que não considera a totalidade dos aspectos da realidade. No mínimo uma abstração dos aspectos físicos mais elementares é assumida, mesmo que restrições físicas (tipicamente atrasos) possam ser consideradas.

Os métodos analíticos [LAZ84, BRA83, SIF80, AJM84, TAZ85], por sua vez, não fazem nenhuma simulação do funcionamento. A partir de um modelo definido segundo algumas hipóteses de funcionamento, um conjunto de equações é obtido. Estas equações descrevem matematicamente a relação entre o dimensionamento do modelo (parâmetros de entrada) e seu desempenho (resultados). Desta forma, tais equações são a expressão matemática do modelo, cujo desenvolvimento pode seguir um formalismo qualquer. Em comparação com os métodos elementares e simulações, os métodos analíticos têm o mais alto nível de abstração. A extração é feita totalmente a partir da descrição matemática do modelo (equações) abstraindo-se completamente a realidade física avaliada.

Infelizmente os métodos analíticos puros só são aplicáveis a poucos casos reais, pois suas hipóteses de funcionamento costumam ser restritivas demais. Para atenuar este problema é frequente a utilização de métodos ditos aproximativos. Estes métodos, ao contrário do método analítico puro que lhes serve de base, não fornecem resultados precisos, mas sim aproximações razoáveis para a realidade estudada. Cabe salientar que as considerações sobre a precisão dos métodos analíticos puros e dos métodos aproximativos refere-se à exatidão dos

resultados frente ao modelo. Desta forma, o método analítico fornece resultados exatos considerando a realidade funcionando segundo as hipóteses adotadas na modelagem. No uso de métodos aproximativos, por outro lado, os resultados fornecidos não são exatos, mas consideram outros aspectos da realidade que haviam sido desprezados pelas hipóteses do método analítico puro.

A grosso modo pode-se dizer que a aplicação de um método analítico não puro (com a aplicação de métodos aproximativos) compõe-se de três etapas distintas: modelagem, extração e aplicação do método aproximativo. Cabe salientar que estas fases não são necessariamente de fácil distinção, podendo estar cronologicamente alternadas. A fase de extração, referida anteriormente, corresponde com frequência a cálculos extremos, como máximos e mínimos teóricos. O método aproximativo, então, parte destes resultados buscando valores mais reais (nem otimistas, nem pessimistas).

Cada método possui suas características intrínsecas, porém de acordo com seu tipo, os métodos apresentam uma série de particularidades comparáveis (critérios de comparação). Evidentemente, para cada caso real a analisar, os diversos métodos serão mais ou menos adequados. Portanto, é possível estabelecer uma comparação genérica entre os tipos de métodos (tabela 1.2).

	Métodos Elementares	Simulações	Métodos Aproximativos	Métodos Analíticos
Sujeito da Avaliação	realidade	modelo funcional	modelo híbrido	modelo comportamental
Nível de Abstração	nenhum	baixo	médio	alto
Velocidade de Extração	real	baixa	média	alta
Fator de Dependência	tempo de observação	tempo de simulação	tempo de convergência	iterações do algoritmo
Precisão dos Resultados	real	real	alta	exata

Tabela 1.2: Características Genéricas dos Métodos de Avaliação

O nível de abstração, como já foi visto, é mais alto para os métodos analíticos. No entanto, quando os métodos aproximativos são utilizados, o nível de abstração deixa de ser absoluto. Os métodos aproximativos partem de hipóteses de funcionamento que se aproximam mais da realidade em questão. Porém, seu nível de abstração continua sendo mais alto que as simulações. Estas podem ser vistas como um método puramente aproximativo, o qual é aplicado desde o início da análise e não a partir de um método analítico puro.

O nível de abstração pode também ser visto como um indicador da flexibilidade de um método, isto é, a facilidade de adaptá-lo a uma realidade similar. A flexibilidade de um método é, portanto, diretamente proporcional a seu nível de abstração, pois este representa também a distância lógica entre a realidade e seu modelo correspondente [GEL80].

A velocidade de avaliação dos métodos elementares é função do tempo de observação, o qual determina também o grau de precisão de seus resultados. Quanto mais tempo observarmos (monitorarmos) a realidade maior será a confiabilidade das conclusões tiradas. Infelizmente não é possível estabelecer uma relação em termos absolutos da velocidade de avaliação dos métodos elementares em comparação aos demais métodos, pois os métodos elementares trabalham sobre uma realidade concreta e não sobre um modelo avaliado em um computador. No entanto, podemos dizer que a velocidade de avaliação das simulações é, analogamente aos métodos elementares, função do tempo de observação. Porém neste caso a observação não é física, então o tempo é chamado tempo de simulação. Exatamente por este tempo, a velocidade das simulações é baixa se comparada aos métodos analíticos puros. Estes possuem uma velocidade alta, pois trata-se simplesmente da resolução de equações. Desta forma, nem a velocidade de avaliação, nem a precisão dos resultados são função de qualquer tempo específico. O algoritmo do método em si, através do seu número de iterações, fornece a velocidade de avaliação. A aplicação dos métodos aproximativos, ao contrário, baseia-se no desdobramento do cálculo básico e frequentemente necessita cálculos iterativos

com tempos de convergência mais ou menos definidos para cada caso particular. No entanto, como os métodos aproximativos só são aplicados em um momento da análise, a velocidade de avaliação dos métodos aproximativos é considerada média.

A precisão dos métodos elementares e das simulações é considerada como referência para os outros métodos, por isto seus resultados são admitidos como reais. No entanto, um tempo muito grande de observação/simulação é necessário para que se possa evitar o perigo de uma amostra pouco significativa do funcionamento do sistema. Os métodos analíticos, em contra partida, não são afetados por este tipo de problema. Não existem influências aleatórias (escolha de uma amostra de funcionamento) para a avaliação. Desta forma, os métodos analíticos puros fornecem resultados perfeitos para a previsão do desempenho. Esta característica é parcialmente passada aos métodos aproximativos. A precisão destes mantém um bom nível (alta), pois esta é uma condição necessária para sua utilização. Nenhum método de avaliação de desempenho pode ter precisão de resultados baixa.

Além das características já citadas, outras podem ser relevantes para a classificação de um método de avaliação de desempenho. Por exemplo, a facilidade de desenvolvimento do modelo é frequentemente considerada no momento da escolha de um método. Estas características, no entanto, são bem mais sensíveis às técnicas (formalismos de modelagem, linguagens de simulação, etc) e a definição mais específica do método em questão.

1.2 Etapas de Avaliação de Desempenho

A aplicação de um método consiste via de regra de três etapas: modelagem, extração e interpretação [LAZ84]. A modelagem ou construção do modelo é obviamente exclusiva dos métodos indiretos. A extração, ao contrário, é comum

a todos os métodos de avaliação, Segundo alguns autores esta etapa é denominada simplesmente de *avaliação*, pois é nesta fase da avaliação que os resultados são obtidos. A terceira fase, interpretação, engloba todo o tratamento que é feito sobre os dados obtidos. A grosso modo é possível dizer que:

- na modelagem definimos o que será avaliado;
- na extração resultados numéricos são obtidos;
- todo tratamento posterior é interpretação.

Embora exista uma forte relação temporal entre estas três etapas, diversas iterações podem ser feitas para obter uma boa avaliação. Por *forte relação temporal* entenda-se que após uma etapa de modelagem (ou remodelagem) sempre virá uma etapa de extração, que por sua vez será seguida de uma etapa de interpretação. Fechando o ciclo, após cada interpretação podemos considerar os resultados satisfatórios e encerrar a avaliação ou reiniciar o ciclo remodelando a realidade tantas vezes quanto for necessário.

1.2.1 Modelagem

Modelar é descrever através de um formalismo uma realidade qualquer. Um modelo é uma descrição de algo que ele próprio não é (a realidade), porém ao conhecê-lo é possível ter informações sobre a realidade descrita. Modelar é abstrair.

Nenhuma parte substancial do universo é tão simples que possa ser compreendida e controlada sem abstração. A abstração consiste em substituir a parte do universo em estudo por um modelo semelhante, porém de estrutura mais simples. Os modelos constituem, portanto, uma necessidade primordial de qualquer procedimento científico.

A modelagem não é um privilégio da avaliação de desempenho, pois o ato de modelar se basta por si só. Sempre que for necessária a compreensão de algo (um sistema) a modelagem deve ser usada. Independente do desenvolvimento posterior das demais etapas de avaliação de desempenho, o processo de modelagem não precisa ser necessariamente alterado.

Para o desenvolvimento de um modelo antes de mais nada é necessário definir o formalismo adotado para a modelagem. Este formalismo compõe-se de um conjunto de primitivas e um conjunto de regras de utilização. Uma linguagem de computador, por exemplo, é um formalismo para modelar algoritmos. Numa linguagem de programação os comandos são as primitivas e a gramática é a regra de utilização. O formalismo é também o responsável pela ausência de ambiguidade no modelo.

A realidade e o formalismo adotado para descrevê-la podem ter uma maior ou menor adequação. Por exemplo, para um algoritmo com muito processamento científico uma linguagem do tipo COBOL não é um bom formalismo para a descrição deste algoritmo. Na verdade a inadequação entre realidade e formalismo pode dificultar e mesmo impossibilitar o desenvolvimento de um modelo satisfatório. Neste caso teríamos um caso extremo de inadequação do formalismo à realidade, teríamos uma realidade não modelável por este formalismo.

O ponto de vista pelo qual observamos uma realidade é tão importante quanto a própria realidade a ser modelada ou mesmo o formalismo adotado para descrevê-la. Este ponto de vista é chamado enfoque de modelagem, ou seja, o aspecto da realidade que é de interesse à avaliação de desempenho. Segundo o enfoque de modelagem, a adequação de um formalismo pode alterar-se. Por exemplo, se quisermos modelar um quadrado usando como primitivas de modelagem pequenos círculos podemos obter uma descrição satisfatória segundo o aspecto área ocupada, mas não segundo o aspecto contiguidade. Neste sentido, toda modelagem deve preocupar-se em definir um modelo adequado segundo o formalismo a utilizar e o enfoque de modelagem a ser considerado.

1.2.2 Extração

Definido um modelo para a realidade, começa a fase de extração. Nesta fase, bem mais objetiva (não subjetiva) que a modelagem, um conjunto de parâmetros de entrada é fornecido e um conjunto de parâmetros de saída é obtido. Esta caracterização genérica de algo que recebe um conjunto de números (avaliação quantitativa) e fornece um outro conjunto de números é que permite identificar a avaliação de desempenho como uma caixa preta. Este é, via de regra, o ponto de vista típico do usuário de avaliação de desempenho. Por esta característica, a fase de extração é com frequência denominada simplesmente de *avaliação*.

A definição de um modelo implica na especificação de uma realidade, porém esta especificação pode ser mais ou menos estanque. Durante a modelagem, alguns pontos da realidade podem ser definidos como variáveis e somente na fase de extração ter um valor associado. Tipicamente características numéricas são especificadas a tempo de extração. A definição destas é chamada de parametrização, pois diz respeito à definição dos parâmetros de entrada. Geralmente, estes parâmetros definem a carga do sistema (realidade) ou suas características de fácil alteração.

A rigor um modelo pode ser completamente fechado e não necessitar de parametrização, porém um modelo assim definido não possui nenhuma flexibilidade. Um modelo é flexível quando podemos obter resultados para realidades semelhantes com um mínimo esforço. Note-se que o nível de abstração do modelo está diretamente relacionado com a flexibilidade de todo o método, logo modelos que suportam um grande número de parâmetros tendem a possuir um maior nível de abstração e vice-versa.

Na simulação, a extração corresponde à experimentação do modelo alimentando-o com estímulos semelhantes aos estímulos que o sistema real re-

cebe durante seu próprio funcionamento (uso). Como toda experimentação, o funcionamento do modelo é observado através de medições de certos componentes do modelo que caracterizam os pontos de interesse da avaliação. Tipicamente, são alvo das medições tamanhos de filas, tempos entre mudanças de estado, etc.

A parametrização típica das simulações é a definição da carga de trabalho/funcionamento ao qual o modelo é submetido, por exemplo, os tempos médios de utilização de recursos. No entanto, outros valores numéricos em geral podem ser adotados como parâmetros de entrada, por exemplo número de recursos, etc. Um parâmetro adicional de controle de simulação é sempre encontrado na prática: o tempo de simulação. Via de regra, este tempo especifica o tamanho do intervalo de tempo que o modelo será experimentado.

Nos métodos analíticos, por sua vez, a extração consiste da aplicação de um algoritmo, um sistema de equações ou até uma simples fórmula. O processo de extração para todos os casos segue os mesmos princípios. A parametrização, por exemplo, consiste da atribuição de valores às variáveis, sejam estas de um algoritmo, de um sistema de equações ou de uma fórmula. Já a obtenção de resultados é ligeiramente diferenciada, sendo fruto da execução do algoritmo, da solução do sistema de equações ou do cálculo da fórmula.

Na maioria dos algoritmos uma solução de mão única é empregada. Os parâmetros de entrada são fornecidos e só então é possível obter os parâmetro saída (resultados). Nos dois outros casos esta característica não se reproduz. Um sistema de equações ou uma fórmula não é somente uma forma de obter resultados, mas uma expressão matemática da relação existente entre diversas grandezas do sistema analisado. De posse de um subconjunto suficiente destas grandezas é possível obter as demais. A diferença entre parâmetros de entrada e saída passa a ser puramente conceitual e portanto relativa. Por exemplo, tanto é possível calcular o tempo de resposta de um terminal em função da sua taxa de transmissão, quanto descobrir o valor da taxa de transmissão necessária para um valor fixo de tempo de resposta. Cabe salientar que esta vantagem possibilita

o cálculo direto dos valores. No entanto, em qualquer outro método é possível chegar às mesmas conclusões, sendo necessário fazer repetidas avaliações até a obtenção dos resultados desejados.

1.2.3 Interpretação

Após possuir os resultados numéricos da avaliação, inicia-se um processo inverso ao processo de modelagem. Na modelagem o sistema real foi traduzido em um modelo que continha informações numéricas para especificar suas características. Na fase de interpretação o processo consiste em traduzir as informações numéricas obtidas em opiniões, juízos sobre o sistema avaliado.

A avaliação quantitativa fornece apenas resultados numéricos. A interpretação traduz estes resultados em informações úteis ao usuário da avaliação de desempenho, pois são estabelecidas conclusões sobre o sistema avaliado. Em suma, interpretar consiste em obter propriedades qualitativas (qualidades e defeitos) a partir do comportamento numérico obtido na extração. Interpretar é concluir.

Com muita frequência, a fase de interpretação pode ter ainda como função testar a confiabilidade da avaliação desenvolvida. Neste caso o alvo do teste pode ser tanto a precisão do modelo desenvolvido, quanto a confiabilidade do método de avaliação empregado. Estas tarefas são genericamente denominadas de validação de modelos e verificação de métodos, respectivamente.

Na validação de modelos o objetivo básico é determinar se a modelagem desenvolvida está correta. O modelo desenvolvido pode possuir imprecisões diversas que variam desde a adoção de hipóteses falsas de funcionamento (comportamento), até simples descuidos na etapa de parametrização. O usual para este tipo de validação é uma comparação de resultados de um outro modelo já verificado, ou mesmo modelos de menor complexidade, por exemplo monitorações.

Na verificação de métodos de avaliação de desempenho os objetivos são mais profundos, pois não só a correção do modelo é questionada, mas o próprio método não é admitido como correto. Neste caso, nem a modelagem, nem as primitivas de extração (fórmulas, algoritmos, equações) estão isentas do teste. O procedimento usual para esta verificação é desenvolver uma comparação com um método já verificado. Usualmente, desenvolve-se a avaliação de uma mesma realidade para ambos os métodos. Cabe salientar que não é frequente a verificação dos métodos. Um método só deve ser verificado se existe dúvida da sua correção, tipicamente na proposta de um novo método.

1.3 Classes de Problemas

Diversas classes de problemas podem ser encontradas na literatura de avaliação de desempenho de sistemas [ALL78, FER78, GEL80, TAZ85]. Basicamente, estas classes descrevem agrupamentos de problemas com características comuns. Via de regra, o padrão de utilização de recursos é o aspecto diferenciado entre as classes de problemas. Não obstante, subdivisões mais detalhadas sempre podem ser adotadas, por exemplo problemas com bloqueio ou capacidade limitada de filas, etc. Dois níveis de observação são frequentemente encontrados para a classificação de problemas no que se refere ao padrão de utilização de recursos. Respectivamente, estes níveis identificam utilização temporalmente fixa ou variável e serial ou simultânea. Quatro tipos elementares de problemas são então definidos (tabela 1.3).

		Alocação de Recursos	
		Serial	Simultânea
Tempos de Utilização	Fixos	Classe 1	Classe 2
	Variáveis	Classe 3	Classe 4

Tabela 1.3: Classificação Genérica de Problemas

Quanto ao tempo de utilização de recursos temos a possibilidade de encontrar um padrão fixo onde, sempre que um recurso for alocado, ele será utilizado de forma idêntica. Por outro lado, a utilização pode ser variável. Esta variação pode ocorrer, seja a nível de unidades de tempo que o recurso é utilizado, seja a nível da quantidade de unidades do recurso que são alocadas. Por exemplo, quando há a necessidade de um número variável de indivíduos para a execução de uma tarefa, ou um tempo variável para a execução desta tarefa, configura-se tal tarefa como um problema com utilização variável de recursos.

Dentro dos problemas com utilização variável de recursos (problemas das classes 2 e 4) é possível definir diversas subdivisões que dizem respeito ao tipo de variação existente. Usualmente os conceitos de variáveis aleatórias são utilizados para classificar os problemas segundo este aspecto. Desta forma, temos problemas com utilização variável de recursos segundo processos exponenciais, normais, etc.

A rigor é possível enxergar a macro divisão de problemas com utilização fixa de recursos como uma das subdivisões citadas. No entanto, a diferença de complexidade dos problemas com utilização fixa justifica a classificação à parte destes problemas. Neste caso também é frequente rotular tais problemas como determinísticos. Nos problemas determinísticos a solução independe de suposições de funcionamento (tempos de utilização), pois é possível prever com exatidão o comportamento (fixo).

Segundo o ponto de vista da distribuição temporal da utilização de recursos, as duas possibilidades diferem segundo a necessidade de uso concomitante de recursos. Nos problemas com utilização serial de recursos uma mesma tarefa pode ser dividida em diversas subtarefas que utilizam diversos recursos. Porém, em nenhuma destas subtarefas dois ou mais recursos de tipos distintos poderão ser utilizados simultaneamente. Em contrapartida, nos problemas com alocação simultânea de recursos existe pelo menos uma subtarefa que necessita de dois tipos de recursos distintos ao mesmo tempo. Cabe salientar que a utilização

simultânea de recursos não implica numa alocação conjunta (alocação no mesmo instante de tempo) dos recursos simultaneamente utilizados. Basta que em algum momento da execução da tarefa pelo menos dois recursos sejam necessários.

Os problemas com alocação simultânea de recursos (problemas 3 e 4) podem se subdividir em dois tipos de problemas segundo a existência de uma relação hierárquica entre os recursos ou não. Nos problemas onde existe tal hierarquia parte-se da premissa que existe um recurso dito básico que utiliza os demais. Nestes problemas é possível identificar no recurso básico a figura de um cliente. Em virtude desta característica este tipo de problema é também denominado de problemas segundo o modelo de clientes.

Antagonicamente, nos problemas onde não existe esta hierarquia entre recursos parte-se do princípio que todos os recursos são indistintamente alocados para a execução das tarefas. Nestes problemas é frequente o emprego da expressão modelo de solicitações de tarefa para designá-los. Cabe salientar que esta subdivisão de problemas com alocação simultânea de recursos não implica que um mesmo problema não possa ser visto de ambas as formas.

1.4 Descrição dos Capítulos

Após a definição dos conceitos básicos feita nesta introdução, os dois próximos capítulos descrevem os dois métodos utilizados no desenvolvimento dos modelos desta dissertação.

O capítulo dois descreve o método analítico tradicional de avaliação de desempenho, aqui denominado RFE (Redes de Filas de Espera). Apesar deste capítulo possuir duas partes descrevendo a modelagem e a extração através do método RFE, a tônica deste capítulo é a descrição do formalismo de modelagem baseado em redes de filas de espera [LAZ84], sem uma maior preocupação com os

formalismos de extração. Um caráter prático visando possibilitar a obtenção de resultados numéricos é o objetivo deste capítulo. Nenhuma consideração teórica é feita a respeito das *cadeias de Markov* [KLE75] que são a base teórica das fórmulas fornecidas neste capítulo.

O terceiro capítulo descreve o método analítico de avaliação de desempenho baseado em Redes de Petri Temporizadas, método **RdPt**. Este capítulo possui três partes que descrevem brevemente as técnicas de modelagem através de redes de Petri [REI85], a extração de resultados através do *modelo Q* [TAZ85] e os métodos aproximativos aplicáveis a este método.

O capítulo quatro descreve especificamente o problema que serve de base para o desenvolvimento dos modelos desta dissertação. Esta descrição se dá através da localização do problema no contexto de arquiteturas de processadores [HWA84]. Define-se então as interconexões multi-estágios com a determinação dos seus parâmetros relevantes à avaliação de desempenho.

O quinto capítulo apresenta um modelo para interconexões multi-estágio desenvolvido através do método **RFE**. Este modelo é apresentado em dois passos. Inicialmente o problema é visto como uma simples alternância de estados no modelo fundamental. Após, sofisticções de descrição são incluídas ao modelo fundamental. Finalmente as vantagens e as limitações mais evidentes do modelo desenvolvido são brevemente discutidas.

A exemplo do capítulo anterior, o sexto capítulo apresenta o modelo para interconexões multi-estágios desenvolvido através do método **RdPt**. Uma primeira abordagem utilizando somente o método analítico puro (*modelo Q*) é apresentada. Num segundo momento refinamentos utilizando os métodos aproximativos são aplicados ao modelo fundamental. Uma crítica do modelo refinado obtido é feita com o propósito de mostrar as suas principais limitações e vantagens.

O capítulo sete desenvolve a comparação propriamente dita. Este capítulo está dividido em três partes que correspondem aos três níveis nos quais é feita a comparação. A primeira parte discute a facilidade de modelagem encontrada no desenvolvimento de modelos através de ambos os métodos. A segunda parte tece breves considerações sobre a velocidade de extração de resultados referentes aos dois modelos desenvolvidos. Finalmente, a terceira parte fornece a comparação numérica dos resultados obtidos em comparação a resultados de um caso prático. O caso prático escolhido trata-se de uma máquina paralela com interconexão multi-estágios, oito processadores e oito módulos de memória global. Os resultados numéricos de referência são fornecidos pelo uso de um simulador de interconexões de processadores que é brevemente descrito [CAV89].

Na conclusão são discutidas as principais diferenças e semelhanças verificadas entre os dois métodos. As conclusões são expressas segundo um ponto de vista genérico, bem como no caso específico da avaliação de desempenho de interconexões de processadores. Após são descritos os trabalhos práticos associados ao desenvolvimento desta dissertação, que integram o projeto *ADMP - Avaliação de Desempenho de Máquinas Paralelas*, ao qual esta dissertação também está vinculada. Finalmente, trabalhos futuros são sugeridos como continuação à pesquisa desenvolvida até o momento.

2 MÉTODO RFE

Neste capítulo descreve-se um método de avaliação de desempenho de sistemas composto de duas partes distintas: modelagem e extração. A modelagem utiliza as redes de filas de espera para obter um modelo de funcionamento do sistema a avaliar. A extração consiste na aplicação do método analítico baseado na abordagem estocástica pura.

Apesar das redes de filas de espera serem utilizadas somente para a modelagem, o método descrito neste capítulo será denominado método **RFE**. Na verdade este método é a aplicação típica dos modelos com redes de filas de espera. Os conceitos citados aqui não esgotam as informações sobre o método **RFE**. Eles se resumem ao estritamente necessário à compreensão da análise desenvolvida no capítulo 5.

A primeira parte deste capítulo contém a descrição de redes de filas de espera somente como ferramenta de modelagem. Os conceitos básicos e algumas sofisticções são apresentados. O leitor familiarizado com os conceitos referentes a redes forma-produto pode dispensar a leitura desta parte. Por outro lado, o leitor interessado em maiores informações sobre este assunto poderá encontrar vasto material em [KLE75, LAZ84].

Na segunda parte, apresentam-se os cálculos associados à ferramenta de modelagem vista anteriormente. Para tanto utiliza-se a solução através da abordagem estocástica baseada no cálculo da constante de normalização. A bibliografia referente às redes de filas de espera já citada contém igualmente informações mais detalhadas sobre a extração do método **RFE**. Não obstante, é interessante adicionar o trabalho de Bruel e Balbo [BRU80] pela sua conotação bem mais pragmática para o caso das análises aqui desenvolvidas.

2.1 Modelagem

A avaliação de desempenho através de um sistema de filas é uma técnica de ampla utilização nos meios científicos. Particularmente, o ramo prático da teoria de filas utilizado para a avaliação de desempenho é denominado Redes de Filas de Espera (*Queueing Networks*). Estas redes definem um formalismo de modelagem ao qual um formalismo de extração analítica de resultados pode ser aplicado. Não obstante, um modelo em redes de filas de espera pode sofrer um outro tratamento qualquer, por exemplo servir a uma simulação. Note-se que em ambas as opções, simulação e métodos analíticos, o propósito é a avaliação de desempenho. No entanto, esta parte apresenta as redes de filas de espera unicamente como formalismo de descrição de sistemas. Serão para tanto vistos os conceitos de estações de serviço e rotação (rede de filas).

2.1.1 Estações de Serviço

O conceito de estações de serviço (figura 2.1) é a base da compreensão das redes de filas de espera. Uma estação compõe-se de um ou mais servidores e de uma fila de espera. O funcionamento de uma estação consiste na passagem de clientes através da estação. Os clientes chegam, eventualmente esperam a disponibilidade de um servidor, recebem o serviço prestado e deixam a estação. A caracterização de uma estação é composta de cinco informações precisas:

- O processo de chegada (A);
 exemplo: processo exponencial;
- O processo de serviço (B);
 exemplo: processo poissônico;
- O número de servidores (C);

exemplo: mono-servidor;

- A capacidade da fila de espera (K);

exemplo: capacidade infinita;

- A disciplina de serviço;

exemplo: FIFO (First In, First Out);

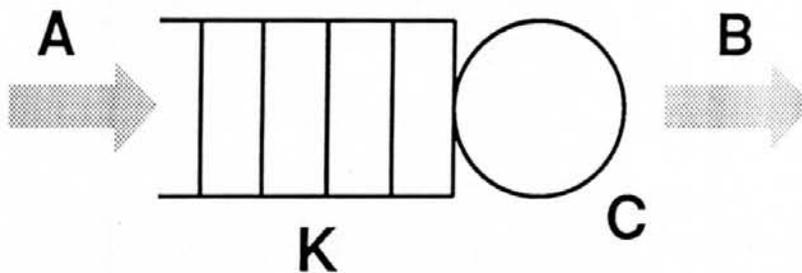


Figura 2.1: Estação de Serviço

O processo de chegada de clientes define o comportamento de chegada de clientes à estação. O processo de serviço define o tempo gasto com cada cliente quando da prestação do serviço. Estes dois processos serão definidos segundo uma lei de distribuição qualquer. Por exemplo, podemos ter a chegada de clientes definida segundo um processo exponencial com média 5, ou seja a cada 5 minutos, em média, um novo cliente chegará à estação. O número de servidores corresponde ao número de clientes que podem ser simultaneamente atendidos. Este número é um número natural (exceto zero) ou mesmo infinito. No caso de um número infinito de servidores, os clientes nunca esperarão, ou seja, não haverá fila de espera. Similarmente, a capacidade da fila de espera pode ser infinita ou um número natural qualquer, também exceto zero. Finalmente a disciplina de serviço define a política de administração da fila. Em princípio qualquer disciplina pode ser adotada, porém para a maioria dos modelos a disciplina FIFO é utilizada.

A definição específica de uma estação de serviço é feita através de certos parâmetros numéricos. Tais parâmetros vão dizer com precisão o funcionamento esperado para a estação de serviço. Por exemplo supondo:

- uma estação de serviço mono-servidor;
- a capacidade da fila infinita;
- a chegada de clientes segundo um processo exponencial;
- a prestação de serviço segundo um processo poissônico;

Duas informações numéricas devem ser definidas:

- a taxa de chegada de clientes (λ_i), ou seja, a média do processo exponencial que descreve a chegada de clientes na estação i ;
- o tempo médio de prestação de serviço (S_i), ou seja, a média do processo poissônico que descreve o atendimento de clientes na estação i .

Para estações multi-servidores, o número de servidores da estação deve ser conhecido (C_i). Por razões de cálculo, a definição da prestação de serviço deve ser feita também através de um processo exponencial (à semelhança da chegada de clientes). Neste caso a definição da taxa de prestação de serviço é feita através de um conjunto de variáveis ($\mu_i(k)$) dependentes do número de clientes (k) presentes na estação i .

Basicamente, o tempo médio de prestação de serviço (S_i) equivale ao inverso da taxa de prestação de serviço quando somente um ^{cliente} servidor está presente ($\mu_i(1)$). Para o caso de um número maior de clientes na estação ($k > 1$) até a capacidade de atendimento da estação (número de servidores - C_i) a taxa de prestação de serviço decresce proporcionalmente. A fórmula abaixo ¹ define o cálculo de $\mu_i(k)$ para uma estação i com C_i servidores e tempo médio de atendimento S_i :

$$\mu_i(k) = \frac{\min(k, C_i)}{S_i}$$

¹A notação $\min(a, \dots, z)$ significa o menor valor numérico entre os valores a até z .

Por exemplo, supondo uma estação onde cada servidor pode atender um cliente em duas unidades de tempo ($S_i = 2$). No caso da estação possuir três servidores ($C_i = 3$), é possível dizer que quando houverem três ou mais clientes na estação, esta terá um comportamento análogo ao de uma estação onde há somente um servidor com tempo de atendimento três vezes menor. Estas informações são definidas através de taxas de serviço distintas segundo o número de clientes presentes na estação, para este caso temos:

- $\mu_i(0) = 0,0$ cl./u.t. (por definição)
- $\mu_i(1) = 0,5$ cl./u.t. $\min\left(\frac{1,3}{2}\right) = 0,5$
- $\mu_i(2) = 1,0$ cl./u.t.
- $\mu_i(3) = 1,5$ cl./u.t.
- $\mu_i(4) = 1,5$ cl./u.t.
- ...

Ainda é possível definir uma distinção entre classes de clientes. Neste caso, para cada classe de clientes é necessário definir os processos de chegada e de serviço, bem como a disciplina de serviço entre as classes. As estações com tratamento distinto para as classes de clientes são chamadas estações multi-classes, ao contrário das estações elementares ditas mono-classe. Em uma estação multi-classes todos os servidores estão aptos a atender a todos os clientes, segundo a disciplina e os processos de serviço especificados. Igualmente na fila de espera qualquer limitação diz respeito ao total de clientes na fila. Nenhuma restrição individual ao número de clientes em uma determinada classe pode ser especificada.

2.1.2 Redes de Filas de Espera

Definidas as estações, definem-se as redes de filas de espera como sendo um conjunto de estações ligadas de acordo com um caminho determinado (figura 2.2). Este caminharmento é descrito através da probabilidade de rotação da rede ($p_{i,j}$). Esta probabilidade define a possibilidade de um cliente saindo de uma estação qualquer i , ir para uma outra estação da rede ou retornar a ela mesma j . Em suma, a probabilidade de rotação de uma rede é definida através de uma matriz quadrada de ordem igual ao número de estações que compõem a rede. Esta matriz chama-se p e é indexada por dois índices denominados i e j .

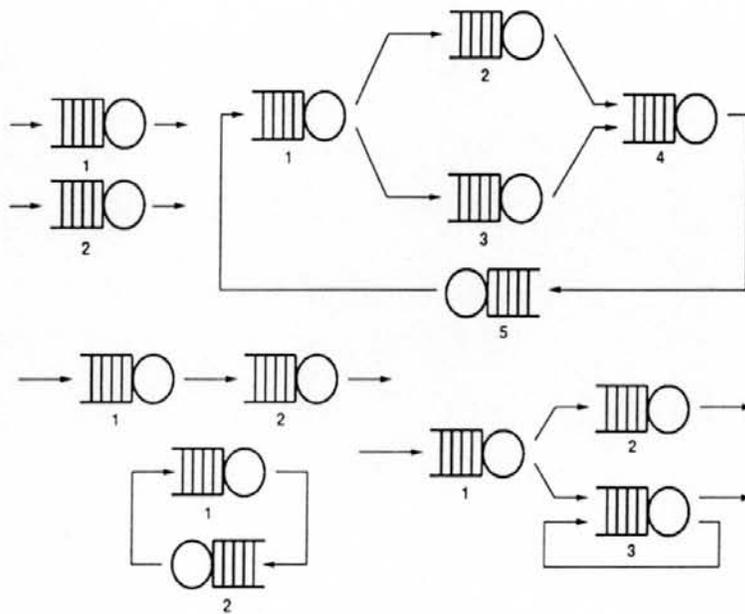


Figura 2.2: Redes de Filas de Espera

As redes com estações multi-classes podem possuir probabilidades de rotação distintas para cada classe de clientes. Se existe pelo menos uma estação multi-classes na rede, toda a rede é dita multi-classes (senão mono-classe).

Segundo a forma das ligações (probabilidade de rotação) as redes podem se dividir ainda em redes abertas e fechadas (figura 2.3). Uma rede é denominada aberta quando existem pontos de contacto com o exterior do modelo (uma entrada e uma saída). Neste caso é necessário definir probabilidades de

rotação para estes pontos de maneira análoga às demais estações da rede. Na prática é como se tivéssemos uma estação a mais representando tudo que é exterior ao modelo. Se ao contrário a rede não possui nenhuma comunicação com o exterior, a rede é classificada como uma rede fechada. Neste caso o número de clientes presente na rede é constante. Este número deve ser definido inicialmente e não pode ser alterado durante o funcionamento da rede. Nas redes abertas, ao contrário, este pode variar sempre de acordo com o fluxo de entrada e saída de clientes.

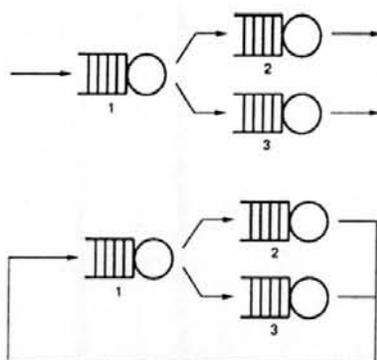


Figura 2.3: Redes Abertas e Fechadas

2.2 Extração

Neste momento vê-se o método analítico para a extração de resultados a partir de modelos desenvolvidos segundo as primitivas definidas anteriormente. O método analítico utilizado é a abordagem estocástica através do cálculo da constante de normalização. Este método é definido para uma classe de redes de filas de espera denominada redes forma-produto (*product-form networks*). Para esta classe de redes algumas hipóteses de funcionamento são definidas para permitir a obtenção de resultados exatos. Uma rede é considerada forma-produto quando quatro hipóteses se verificam. Estas hipóteses dizem respeito à capacidade das filas, aos processos de chegada e serviço, à probabilidade de rotação da rede e ao equilíbrio de fluxo das estações.

Hipóteses:

- Filas de espera com capacidade ilimitada;
- Chegada de clientes e taxa de serviço segundo uma lei exponencial;
- Probabilidade de rotação de clientes segundo um processo Markoviano;
- Estações com comportamento estacionário;

Segundo os princípios de redes forma-produto a capacidade das filas é ilimitada. Desta forma, um cliente sempre será admitido na fila de uma estação. A consequência prática desta hipótese é descartar a possibilidade de bloqueio de uma estação por falta de espaço em qualquer uma das estações seguintes.

A chegada de clientes e a taxa de serviço das estações são expressas segundo uma lei exponencial. A chegada de clientes é descrita por um processo Poissoniano e o tempo de serviço é descrito por um processo Exponencial. Desta forma, basta definir numericamente o valor médio destes processos.

A rotação dos clientes é totalmente probabilística. O caminho de um cliente é independente do passado. Cada vez que um cliente sai de uma estação com duas ou mais possibilidades de caminho, este será definido aleatoriamente segundo a probabilidade de rotação. Esta decisão é um evento independente, logo a alternância de estados da rede pode ser descrita através de uma cadeia de Markov [KLE75].

Por fim, uma hipótese de estacionariedade é admitida. Esta define que o fluxo de clientes através de uma estação é equilibrado. Desta forma, o número de clientes que entra é, em média, igual ao número de clientes que sai. Para as redes abertas esta última hipótese ainda define que o fluxo de clientes em toda rede também é equilibrado.

Para as redes forma-produto existem algoritmos formalmente definidos para a obtenção da taxa de saída de cada estação, da taxa de utilização de cada servidor, do número médio de clientes presentes em cada estação e do tempo médio de resposta de cada estação [KLE75, BRU80, LAZ84]. Outros parâmetros podem ser obtidos, no entanto, os parâmetros acima enumerados são suficientes para a maioria dos casos reais e para a totalidade dos casos propostos nesta dissertação. Pela mesma razão será vista somente a análise de redes de filas fechadas mono-classe. A notação empregada para a descrição dos modelos será a seguinte:

Variáveis de entrada (modelo):

- M número de estações da rede;
- N número de clientes presentes na rede;
- S_i tempo médio de serviço na estação i (mono-servidor);
- $\mu_i(k)$ taxa de serviço da estação i com k clientes (multi-servidores);
- $p_{i,j}$ probabilidade de um cliente sair da estação i e ir à estação j (probabilidade de rotação);

Variáveis auxiliares:

- V_i taxa média de visita da estação i ;
- $F_i(N)$ caracterização concisa da estação i na rede com N clientes;
- $G(N)$ constante de normalização na rede com N clientes;
- $G_i^M(N)$ constante de normalização independente da estação i na rede com N clientes;
- $p_i(n, N)$ probabilidade que n clientes estejam na estação i na rede com N clientes (probabilidade marginal);

Índices de desempenho:

- $D_i(N)$ taxa de saída da estação i na rede com N clientes;
- $U_i(N)$ índice de utilização dos servidores da estação i na rede com N clientes;
- $N_i(N)$ número médio de clientes presentes na estação i na rede com N clientes;
- $W_i(N)$ tempo médio de resposta da estação i na rede com N clientes;

A configuração da rede estando definida pelas variáveis de entrada, um conjunto de equações permite a obtenção das variáveis intermediárias e por consequência os parâmetros de saída (índices de desempenho). A seguir as equações de resolução são dadas através de exemplos para os dois casos previstos, redes fechadas a forma-produto: mono-classe/mono-servidor e mono-classe/multi-servidores.

2.2.1 Redes Fechadas, Mono-classe, Mono-servidor

Supondo o exemplo da figura 2.4, onde temos uma rede fechada composta de cinco estações mono-servidor ($M = 5$) e onde 10 clientes ($N = 10$) estão presentes, seja o tempo de prestação de serviço de cada estação e a probabilidade de rotação definida como:

$S_1 = 2$		$p_{i,j}$	1	2	3	4	5
$S_2 = 1$	$N = 10$	1	0	0,5	0	0,5	0
$S_3 = 2$		2	0	0	1	0	0
$S_4 = 2$	$M = 5$	3	1	0	0	0	0
$S_5 = 1$		4	0	0	0	0	1
		5	1	0	0	0	0

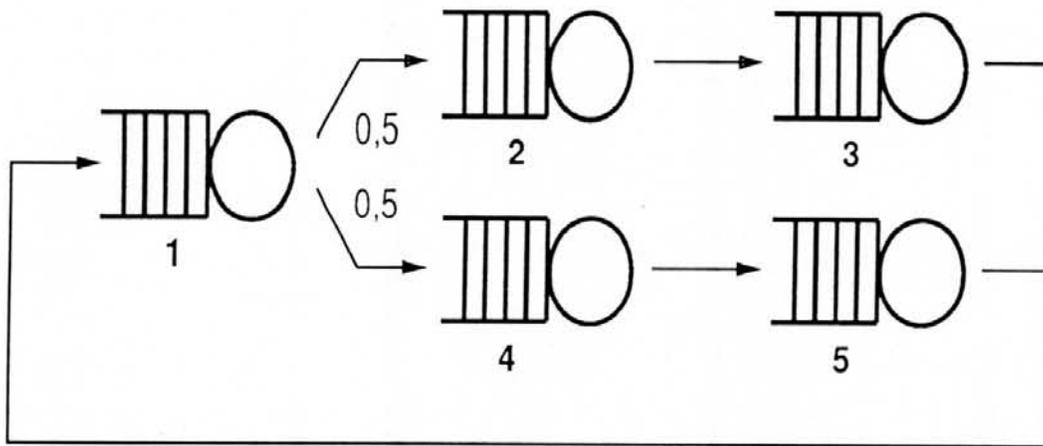


Figura 2.4: Exemplo de Rede Mono-servidor

A relação entre as taxas médias de visita das estações é dada pelas M equações definidas abaixo:

$$V_i = \sum_{j=1}^M V_j p_{j,i} \quad \text{para } i = 1, \dots, M$$

Tendo a relação entre os V_i 's definida, podemos atribuir um valor arbitrário a uma estação, por exemplo $V_1 = 1$. A interpretação de cada taxa de visita será, neste caso, o número médio de passagens pela estação i entre duas passagens pela estação 1. Para o exemplo escolhido:

$$V_1 = 1,0 \quad V_2 = 0,5 \quad V_3 = 0,5 \quad V_4 = 0,5 \quad V_5 = 0,5$$

A solução das sistemas de redes de filas de espera adotada no contexto desta dissertação é através do cálculo da constante de normalização ($G(N)$ [BRU80]). Os fundamentos teóricos desta abordagem definem que os diversos estados da cadeia de Markov que representa o sistema da rede de filas devem possuir uma probabilidade associada a eles. A constante de normalização tem por função assegurar que a soma da probabilidade destes estados é igual a um (1). Desta forma, os resultados obtidos são, além de proporcionais, coerentes com a realidade. Por exemplo, a soma do número médio de clientes presentes ($N_i(N)$) em todas as estações da rede deve ser igual ao total de clientes da rede (N).

A constante de normalização para todos números possíveis de clientes na rede ($G(0), \dots, G(N)$) são obtidas através do algoritmo de Buzen [BUZ73]. A

descrição deste algoritmo foge ao escopo desta dissertação. No entanto, o leitor interessado encontrará uma implementação eficiente deste algoritmo em anexo a este volume (anexo A1). Para o exemplo escolhido a aplicação do algoritmo de Buzen fornece os seguintes resultados:

$$\begin{array}{lll}
 G(0) = 1 & G(1) = 5 & G(2) = 15,75 \\
 G(3) = 40,5 & G(4) = 93,56 & G(5) = 203,4375 \\
 G(6) = 427,0469 & G(7) = 878,1875 & G(8) = 1784,4258 \\
 G(9) = 3600,8789 & G(10) = 7237,7725 &
 \end{array}$$

Para este primeiro caso de estação mono-servidor e rede mono-classe somente as variáveis auxiliares já calculadas são suficientes para a obtenção dos índices de desempenho.

A taxa de saída de cada estação é dada pela fórmula abaixo:

$$D_i(N) = V_i \frac{G(N-1)}{G(N)}$$

A taxa de utilização do servidor de cada estação é o simples produto abaixo:

$$U_i(N) = S_i D_i(N)$$

O número médio de clientes presentes numa estação é calculado através da fórmula recursiva abaixo, onde a condição inicial é $N_i(0) = 0$:

$$N_i(N) = U_i(N)(1 + N_i(N-1))$$

Finalmente, segundo a lei de Little [LAZ84], o valor médio do tempo de resposta é obtido pela fórmula abaixo:

$$W_i(N) = \frac{N_i(N)}{D_i(N)}$$

Para o exemplo escolhido os valores obtidos são exibidos na tabela 2.1.

i	$D_i(10)$	$U_i(10)$	$N_i(10)$	$W_i(10)$
1	0,4975	0,9950	7,3912	14,856
2	0,2488	0,2488	0,3304	1,3281
3	0,2488	0,4875	0,9740	3,9156
4	0,2488	0,4875	0,9740	3,9156
5	0,2488	0,2488	0,3304	1,3281

Tabela 2.1: Resultados Obtidos para o Exemplo Mono-servidor

2.2.2 Redes Fechadas, Mono-classe, Multi-servidores

Na seção precedente o tempo médio de serviço (S_i) foi considerado constante. As redes forma-produto admitem, porém, que este tempo seja função do número de clientes presente na estação. A aplicação típica desta facilidade é a definição de estações multi-servidores.

Formalmente as estações com tempo de serviço sensível ao número de clientes são chamadas estações dependentes da carga. Por motivos técnicos, o tempo médio de atendimento (S_i) para estações dependentes da carga é expresso através da taxa média de clientes atendidos por unidade de tempo ($\mu_i(k)$), uma função do número de clientes presentes na estação (k).

Nesta seção é adotado como exemplo uma variante do exemplo anterior (figura 2.4), onde a estação 1 possui, não mais um único servidor, mas dois servidores idênticos. As informações permanecem as mesmas, exceto a especificação do tempo médio de atendimento da estação 1, que é definido abaixo:

$S_1 = 2$	$C_1 = 2$	$\mu_1(0) = 0.0$ cl./u.t		$p_{i,j}$	1	2	3	4	5
$S_2 = 1$	$C_2 = 1$	$\mu_1(1) = 0.5$ cl./u.t	$N = 10$	1	0	0,5	0	0,5	0
$S_3 = 2$	$C_3 = 1$	$\mu_1(2) = 1.0$ cl./u.t		2	0	0	1	0	0
$S_4 = 2$	$C_4 = 1$	$\mu_1(3) = 1.0$ cl./u.t	$M = 5$	3	1	0	0	0	0
$S_5 = 1$	$C_5 = 1$...		4	0	0	0	0	1
				5	1	0	0	0	0

As taxas de visita (V_i) continuam as mesmas, os valores das constantes de normalização, porém, não são mais válidos. No entanto, o algoritmo de

Buzen (anexo A1) continua válido, pois ele aceita ambos os tipos de estações (dependentes e independentes da carga). Para este novo exemplo o algoritmo fornece os valores abaixo:

$$\begin{array}{lll}
 G(0) = 1 & G(1) = 5 & G(2) = 13.75 \\
 G(3) = 28.5 & G(4) = 50.06 & G(5) = 78.9375 \\
 G(6) = 115.4219 & G(7) = 159.6875 & G(8) = 211.8320 \\
 G(9) = 271.9102 & G(10) = 339.9521 &
 \end{array}$$

Para o cálculo dos índices de desempenho para as estações independentes da carga (seção anterior) algumas simplificações foram adotadas. Para as estações dependentes da carga, ao contrário, é necessário trabalhar mais próximo das definições matemáticas (cadeias de Markov). No entanto, algumas simplificações ainda podem ser mantidas para evitar complicações desnecessárias à aplicação de redes de filas de espera. Desta forma, definem-se três novas variáveis auxiliares: a caracterização concisa, a constante de normalização independente e a probabilidade marginal.

A caracterização concisa de uma estação define a importância da sua taxa de atendimento ($\mu_i(k)$) de acordo com sua taxa de visita (V_i). O seguinte cálculo iterativo fornece a caracterização concisa da estação i para a rede com N clientes:

$$F_i(N) = \frac{V_i}{\mu_i(N)} F_i(N - 1)$$

com a condição inicial: $F_i(0) = 1$

A exemplo da constante de normalização que é calculada levando-se em conta todas estações da rede, a constante de normalização independente é calculada desprezando o efeito de uma determinada estação. Esta variável auxiliar tem por função possibilitar a análise do comportamento da rede considerando estática a estação em questão (i). A constante de normalização independente

da estação i para a rede com N clientes também é obtida através de um cálculo iterativo:

$$G_i^M(N) = G(N) - \sum_{k=1}^N F_i(k) G_i^M(N-k)$$

com a condição inicial: $G_i^M(0) = G(0) = 1$

A probabilidade marginal do número de clientes na estação i para a rede com N clientes é obtida pela seguinte fórmula:

$$p_i(n, N) = \frac{F_i(n)}{G(N)} G_i^M(N-n)$$

A probabilidade marginal do número de clientes numa estação é utilizada por alguns autores como um índice de desempenho. Porém na maioria dos casos, este parâmetro é somente uma variável auxiliar. Obviamente, quando existe o interesse específico na determinação de um tamanho crítico para a fila de espera de uma das estações, a probabilidade marginal deve ser encarada como um índice final de desempenho e não apenas uma variável auxiliar.

O cálculo da taxa de saída de cada estação permanece idêntico, exceto pela adoção de novas constantes de normalização, ou seja:

$$D_i(N) = V_i \frac{G(N-1)}{G(N)}$$

A taxa de utilização dos servidores de estações multi-servidores é dada pela seguinte fórmula:

$$U_i = \frac{G_i^M(N)}{G(N)}$$

O cálculo do número médio de clientes presentes em cada estação não pode ser simplificado. Sua definição formal, a soma ponderada das probabilidades marginais, é utilizada como fórmula de cálculo:

$$N_i(N) = \sum_{n=1}^N n p_i(n, N)$$

O tempo médio de espera de clientes em cada estação continua a ser obtido pela lei de Little, ou seja:

$$W_i(N) = \frac{N_i(N)}{D_i(N)}$$

Para o exemplo escolhido os valores obtidos são expressos na tabela 2.2.

i	$D_i(10)$	$U_i(10)$	$N_i(10)$	$W_i(10)$
1	0,7998	0,8941	3,2008	4,0018
2	0,3999	0,3999	0,6460	1,6154
3	0,3999	0,7998	2,7536	6,8854
4	0,3999	0,7998	2,7536	6,8854
5	0,3999	0,3999	0,6460	1,6154

Tabela 2.2: Resultados Obtidos para o Exemplo Multi-servidores

Cabe salientar que o cálculo das variáveis auxiliares dado para as estações dependentes da carga não pode ser utilizado tal e qual para as estações independentes da carga. Maiores detalhes, tipicamente o cálculo da probabilidade marginal, podem ser encontrados em [BRU80].

3 MÉTODO RdPt

Neste capítulo descreve-se um método para avaliação de desempenho de sistemas composto de três partes distintas: modelagem, extração através do método analítico puro e aplicação de métodos aproximativos. Trata-se portanto de um método híbrido onde a aplicação de um método analítico puro é seguida do uso de métodos aproximativos.

A modelagem tem sua base na Teoria Geral de Redes (redes de Petri [PET77]), um formalismo de modelagem bastante difundido. As redes de Petri utilizadas são denominadas redes lugar/transição temporizadas (*timed P/T-nets*). Eventualmente redes mais complexas são citadas para melhor visualizar os modelos dos métodos aproximativos.

A extração, por sua vez, utiliza o *modelo Q* [TAZ88], um conjunto de fórmulas para a obtenção de resultados numéricos a partir de modelos definidos em redes de Petri, segundo certas restrições. Na verdade, utiliza-se um subconjunto do *modelo Q* chamado *Q1*, um método determinístico para problemas com alocação simultânea de recursos. Cabe salientar que nesta dissertação entende-se por *modelo Q* a proposta original formalizada em [TAZ85], pois apenas esta pode ser classificada como um método analítico puro.

Na terceira etapa métodos aproximativos baseados em ponderações (teoria das probabilidades) são empregados. Estes métodos tem por objetivo a obtenção de resultados mais precisos através da maior adequação do modelo à realidade. A aplicação de tais métodos tem por função considerar aspectos que envolvam aleatoriedade de comportamento.

Este capítulo divide-se em três partes para descrever a modelagem, a extração segundo o *modelo Q* e o uso dos métodos aproximativos associados. A primeira parte contém alguns conceitos básicos sobre redes de Petri em geral e o

conceito de temporização. Desta forma, o leitor familiarizado com estes conceitos pode dispensar a leitura desta parte. Por outro lado, o leitor interessado em maiores informações sobre esta matéria poderá encontrar vasto material bibliográfico em [PET77, PET81, REI85]. Especificamente as redes de Petri temporizadas são descritas em [BRA83, RIC85]. Compõe ainda a primeira parte exemplos simples de modelagem segundo o escopo de aplicação do *modelo Q*. A segunda parte apresenta o *modelo Q* em geral com ênfase especial no seu subconjunto de interesse, o *modelo Q1*. O leitor interessado em maiores detalhes nos demais subconjuntos do *modelo Q* e sua fundamentação teórica, poderá encontrar tais informações em [TAZ85, TAZ85a, TAZ88]. Finalmente, a terceira parte define os métodos aproximativos que podem ser utilizados em conjunto com o método analítico puro anteriormente descrito (*modelo Q*). Os dois métodos aproximativos apresentados não possuem maior documentação exceto casos práticos de avaliação de desempenho onde estes foram empregados informalmente. O leitor interessado pode encontrá-los em [TAZ87, FER88, FER88a].

3.1 Modelagem

A Teoria Geral das Redes define diversos formalismos gráficos de modelagem denominados genericamente redes de Petri. Entre estes formalismos gráficos de descrição, o mais utilizado são as redes ditas lugar/transição. No decorrer deste trabalho, somente este tipo de rede será utilizado. No entanto, a definição mais completa destas pode ser vista em [REI85] onde um tipo mais elementar de rede (redes condição/evento) é visto com maior detalhe.

3.1.1 Conceitos Básicos

Os conceitos vistos nesta seção são uma breve descrição prática e informal do mínimo necessário ao entendimento de redes de Petri lugar/transição temporizadas.

Os elementos que definem a estrutura dos modelos para as redes do tipo lugar/transição são os lugares (\circ), as transições (\square) e os arcos (\rightarrow) que interligam estes elementos básicos. Desta forma, trata-se de um grafo orientado, onde os nodos são divididos em dois conjuntos disjuntos: lugares e transições. Os arcos orientados possíveis são definidos sempre entre dois nodos de conjuntos distintos. A figura 3.1 apresenta alguns exemplos de estruturas de redes de Petri.

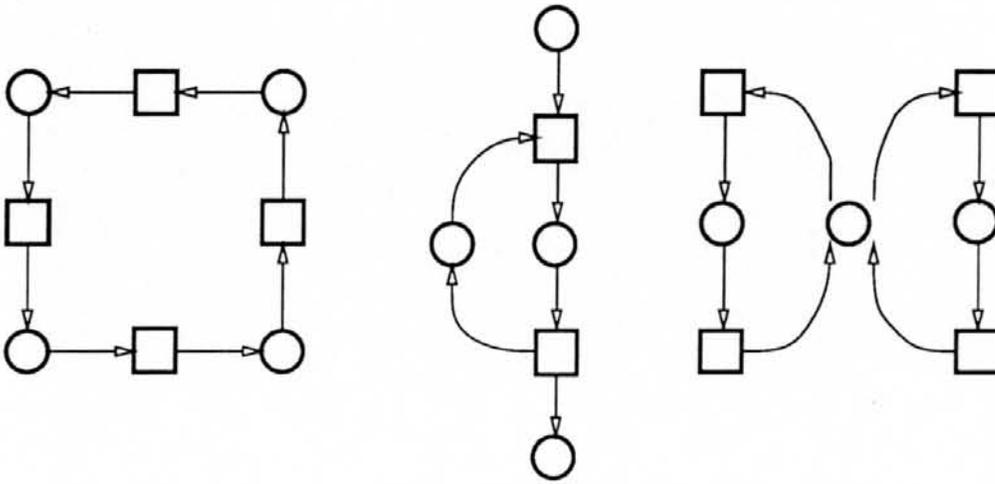


Figura 3.1: Estruturas de Redes de Petri

Além da estrutura, outra noção básica nas redes lugar/transição é o conceito de marcação. A uma mesma estrutura é possível associar diversas marcações através da atribuição de um número natural a cada lugar da rede. A representação gráfica usual para as marcações é o desenho nos lugares de tantas marcas (pontos) quanto for o número natural a ele associado. A figura 3.2 apresenta alguns exemplos de redes, ou seja, estruturas com uma marcação associada.

Podemos ver as marcações possíveis como sendo a representação de estados nos quais a estrutura da rede se encontra. Desta forma, a mudança de

estado em uma rede de Petri equivale à mudança na sua marcação. Esta mudança de estados é feita através do disparo de uma ou mais transições. Seguindo o mesmo raciocínio, temos que a regra para o disparo de transições é a interpretação primeira do caráter dinâmico dos modelos descritos através de redes de Petri.

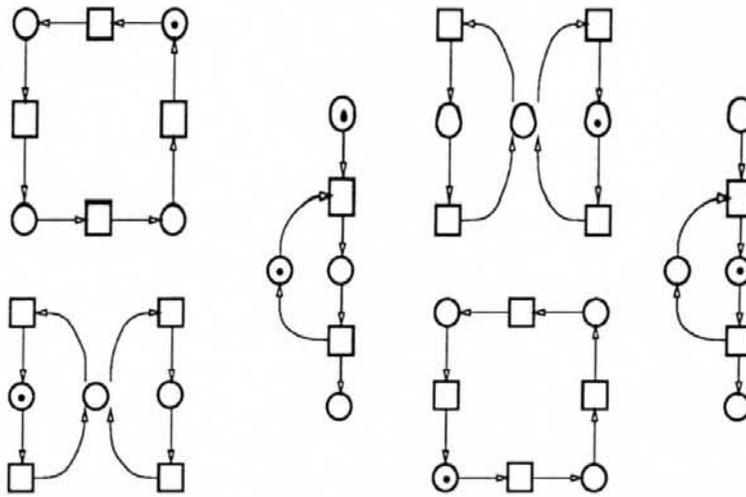


Figura 3.2: Redes de Petri

A regra básica de disparo de transições implica que sejam consumidas marcas dos lugares que alimentam uma transição e sejam produzidas marcas nos lugares alimentados por esta mesma transição. O disparo de uma transição consome uma marca de cada um dos lugares de entrada (lugares que alimentam) da transição disparada. Analogamente, cada um dos lugares de saída (lugares alimentados) da transição recebe uma marca. A figura 3.3 apresenta alguns exemplos do disparo de transições em redes de Petri.

Concomitantemente ao conceito de disparo de transições, existe o conceito de habilitação. Uma transição está habilitada quando existem marcas suficientes para o disparo nos seus lugares de entrada (se houver). Note-se que não é feita nenhuma restrição quanto aos lugares de saída, pois admite-se que todos os lugares da rede possuem uma capacidade infinita. Esta característica não é compatível com a definição original de [TAZ85]. Não obstante, esta simplificação é adotada, pois não altera as propriedades do modelo Q .

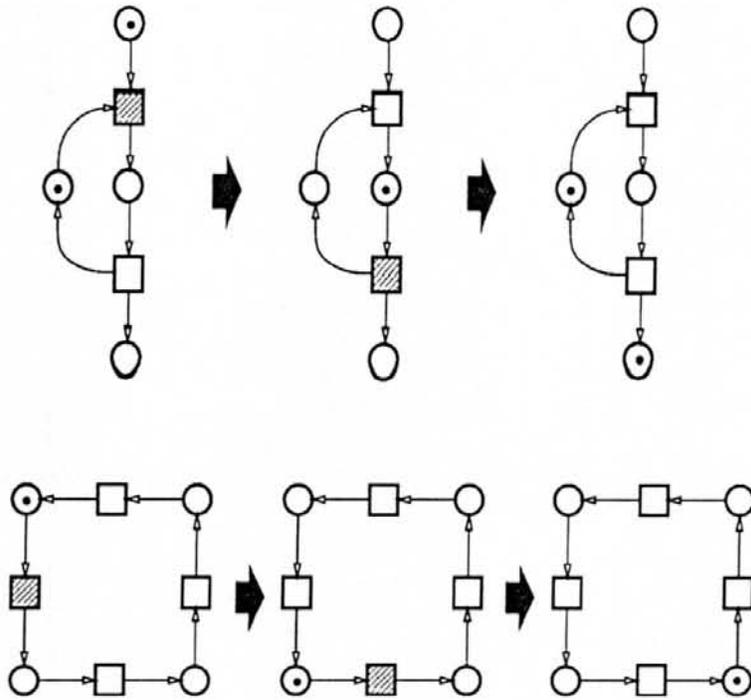


Figura 3.3: Disparos de Transições

A definição completa de um modelo em rede de Petri possui ainda o conceito de marcação inicial. A marcação inicial define um estado inicial no qual a rede de Petri deverá iniciar seu funcionamento. Esta marcação como as demais marcações é definida através do número de marcas correspondente a cada lugar da rede. Sua notação é $M_0(l) = m$ onde l é um lugar que inicialmente possui m marcas. Desta forma, uma rede de Petri lugar/transição é definida através de sua estrutura e de sua marcação inicial.

O propósito final da modelagem aqui descrita é a avaliação de desempenho de sistemas. Portanto, é necessário introduzir o conceito de tempo nos modelos, através de redes temporizadas. Duas opções equivalentes de redes (mesmo poder de descrição) existem para representar o conceito de temporização: temporização de lugares e de transições. Neste trabalho será adotada a temporização de lugares, ou seja, a cada lugar um tempo de indisponibilidade pode ser atribuído. Esta decisão foi tomada para manter a compatibilidade com o *modelo Q*.

O funcionamento das redes temporizadas é idêntico àquele das redes ordinárias, exceto pela inclusão de um conceito de indisponibilidade de marcas. Desta forma, quando uma marca entra em um lugar temporizado, esta marca

permanece indisponível para o disparo de todas as transições da rede durante o tempo atribuído a este lugar. A notação adotada para representar a temporização de um lugar é definida como $Z(l) = t$, onde l é um lugar que possui um tempo de indisponibilidade de t unidades de tempo. De forma análoga às redes ordinárias, nas redes temporizadas o momento de disparo de transições também não é determinado. Quando as marcas de um certo lugar tornam-se disponíveis, as transições que dependem da marcação deste lugar tornam-se habilitadas podendo ou não disparar imediatamente. A figura 3.4 ilustra a temporização de lugares através da abstração de indisponibilidade de marcas.

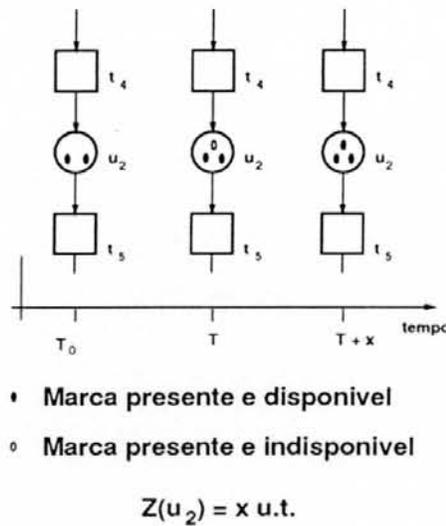


Figura 3.4: Temporização de Lugares

Esta definição da temporização é considerada uma extensão às redes lugar/transição originais. Esta abordagem é simples e está fundamentada em [BRA83]. No entanto, a temporização pode ser vista não como uma extensão, mas como uma abreviação de uma rede lugar/transição. Neste caso assume-se a existência de um relógio lógico (uma sub-rede) que controla a indisponibilidade das marcas. Segundo esta abordagem [RIC85] algumas pequenas diferenças são encontradas. A única diferença relevante no contexto desta dissertação refere-se ao disparo das transições que é obrigatório quando existem marcas disponíveis.

3.1.2 Escopo de Aplicação do *Modelo Q*

Os conceitos vistos nesta seção dizem respeito a modelagem visando a aplicação dos modelos desenvolvidos ao *modelo Q*. Não obstante, também estes modelos podem ter qualquer aplicação distinta, inclusive simulação. O *modelo Q*, independente do seus níveis de aplicação ($Q0$, $Q1$, $Q2$ ou $Q3$), utiliza os conceitos aqui descritos de cadeia principal e recursos. Os conceitos desta seção, porém, dizem respeito ao nível $Q1$ que é o foco de atenção deste capítulo.

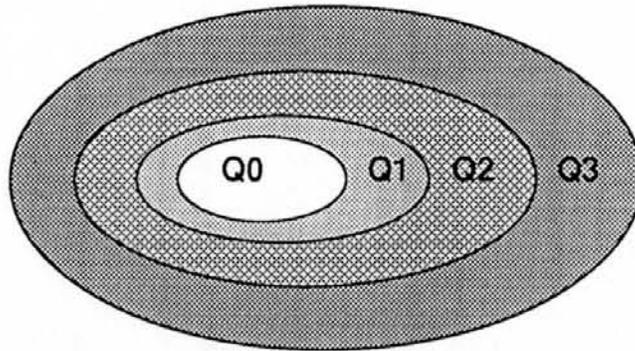


Figura 3.5: *Modelo Q*

O escopo de aplicação do *modelo Q* é um subconjunto das redes de Petri lugar/transição temporizadas definidas na seção anterior. O *modelo Q* está dividido em quatro níveis de utilização superpostos segundo uma ordem crescente de complexidade (figura 3.5). Cabe salientar que a complexidade referenciada não diz respeito ao poder computacional necessário para a solução. Trata-se somente do maior poder descritivo dos modelos dos níveis superiores ($Q2$ e $Q3$). Desta forma, cada nível possui um escopo de aplicação, o qual contém aqueles dos níveis mais simples.

O primeiro conceito no *modelo Q* é o conceito de Cadeia Principal, ou Cadeia Aberta. Uma determinada tarefa, que é o alvo da avaliação de desempenho (tarefa alvo), é representada através de uma Cadeia Principal. Esta, por sua vez, é representada através de diversos lugares representando cada um deles uma subtarefa que compõe a tarefa alvo. A figura 3.6 apresenta alguns exemplos

de Cadeias Principais, onde temos tarefas distintas com um número distinto de subtarefas representadas.

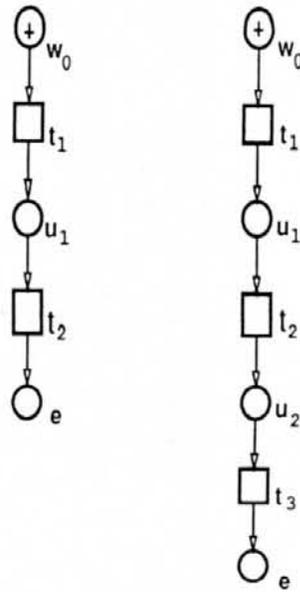


Figura 3.6: Cadeias Principais

Podem ser consideradas subtarefas de uma Cadeia Principal tanto as etapas necessárias à efetivação da tarefa alvo (u_i), quanto os possíveis pontos de espera. Estes pontos de espera podem representar eventuais esperas por sincronismo, ou mesmo disponibilidade de recursos. Notadamente, o primeiro (w_0) e o último (e) lugares da cadeia são lugares de espera. O primeiro representa diversas solicitações da tarefa alvo esperando efetivação. Desta forma, os modelos pertencentes ao escopo de aplicação do *modelo Q* têm como pressuposto básico o funcionamento em capacidade máxima, ou seja, quantas requisições da tarefa alvo forem possíveis, serão atendidas. O último lugar da rede é simplesmente um acumulador de requisições atendidas, onde estas esperam indefinidamente.

Cada subtarefa necessita de recursos para ser efetivada. Desta forma, os recursos são representados como marcas que são alocadas durante a passagem das solicitações através da Cadeia Principal. A disponibilidade dos recursos é então representada por lugares específicos fora da cadeia conectados a ela em dois pontos, a alocação e a liberação dos recursos. Todas as subtarefas contidas desde a alocação dos recursos até a sua liberação necessitam deste recurso para sua

efetivação. A figura 3.7 apresenta algumas estruturas de redes de Petri compostas por uma Cadeia Principal e lugares representando a disponibilidade de Recursos, bem como seus arcos de alocação e liberação. Note-se que toda alocação de recursos é precedida de um lugar de espera que significa a possível espera pela disponibilidade dos recursos que neste ponto são alocados.

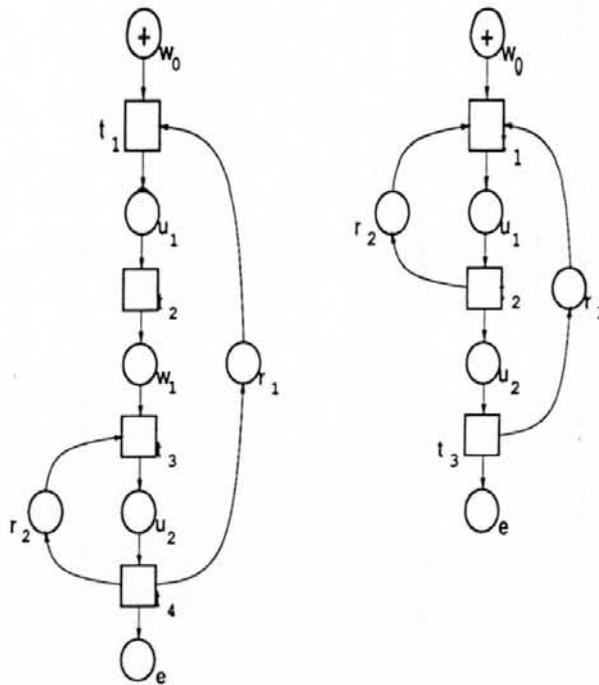


Figura 3.7: Modelos $Q1$

3.2 Extração

A mais típica das utilizações de redes de Petri é, sem dúvida, a modelagem de sistemas. Na maior parte das vezes, esta modelagem tem por objetivo único a descrição de sistemas. Em virtude disto, o uso de redes de Petri na avaliação de desempenho de sistemas não foi suficientemente explorado [SIF80, BRA83, TAZ85]. Por outro lado, o uso de um mesmo formalismo para a modelagem com propósitos descritivos e para a avaliação de desempenho traz vantagens óbvias. Esta característica é tão forte que trabalhos como [AJM84] utilizam conceitos de redes de Petri para modelagem inicial de métodos de avaliação

de desempenho. No entanto, no trabalho de [AJM84], modelos em Redes de Filas de Espera ainda são utilizados para a extração. Estes trabalhos referem-se a redes estocásticas e, portanto, não são compatíveis com as redes utilizadas nesta dissertação.

As redes de Petri são um formalismo de descrição de sistemas que suporta definições genéricas do tipo sequenciamento e sincronismo. O *modelo Q* busca valer-se das facilidades de modelagem das redes de Petri para dar ao método de avaliação as mesmas primitivas de modelagem. O escopo de aplicação do *modelo Q* são redes de Petri que possuem determinadas características. Na verdade quatro níveis de aplicação do *modelo Q* estão disponíveis. Cada um destes níveis possui primitivas específicas para a modelagem de problemas distintos, como alocação simultânea de recursos, utilização variável de recursos, etc.

Os conceitos vistos nesta parte não têm a função de descrever detalhadamente o *modelo Q*, ou mesmo seu subconjunto de interesse nesta dissertação (*modelo Q1*). Busca-se somente possibilitar ao leitor a compreensão do *modelo RdPt*. A propósito do *modelo Q*, cabe salientar que trata-se, na verdade, de um método de avaliação de desempenho analítico. No entanto, a palavra *modelo* é utilizada com o propósito de permanecer compatível com sua denominação original em [TAZ85].

3.2.1 Notação Adotada

A notação adotada para as variáveis e os índices de desempenho do método *RdPt* nesta dissertação não são os mesmos utilizados por [TAZ85] na definição do *modelo Q*. Esta distinção tem por objetivo evitar confusões quando da comparação com o método *RFE*. Cabe salientar que mesmo os conceitos apresentados posteriormente são, como esta notação, um subconjunto simplificado do *modelo Q*.

Notação adotada:

Variáveis de entrada:

- R número de recursos distintos;
- $M_0(r_i)$ marcação inicial do lugar r_i ;
- $Z(u_i)$ temporização associada ao lugar u_i ;

Variáveis auxiliares:

- $P_0(R_i)$ período básico dos recursos do tipo R_i ;
- $d_0(R_i)$ desempenho máximo teórico dos recursos do tipo R_i ;
- $D_0(R_i)$ desempenho isolado dos recursos do tipo R_i ;

Índices de desempenho:

- $D(N)$ desempenho do modelo descrito pela rede N ;
- $U(R_i)$ taxa de utilização dos recursos do tipo R_i ;
- $Z_w(w_i)$ tempo induzido de espera no lugar w_i ;
- $P_m(w_i)$ população média do lugar w_i ;

3.2.2 Modelo Q1

O modelo Q1 define equações básicas para a determinação de três tipos de informações básicas: o índice de desempenho geral do modelo, as taxas de utilização dos recursos e os pontos de indução de tempos de espera (gargalos). O índice de desempenho do modelo fornece quantas tarefas descritas pela cadeia

principal podem ser atendidas por unidade de tempo. As taxas de utilização dos recursos fornecem as informações referentes à carga ociosa de cada um dos recursos envolvidos. Finalmente, os pontos de indução de tempos de espera são identificados pelo tempo induzido de espera e pela população média no gargalo.

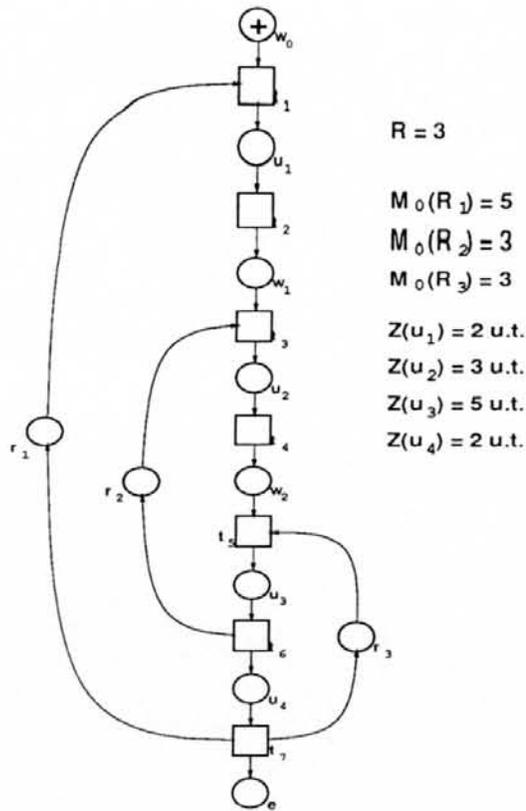


Figura 3.8: Exemplo do Modelo Q1

Seja a figura 3.8 um modelo Q1 composto de uma cadeia principal descrevendo uma tarefa dividida em quatro subtarefas que são executadas por três tipos de recursos. A primeira subtarefa necessita de apenas uma unidade do recurso R_1 . Após a execução desta subtarefa espera-se a disponibilidade de uma unidade do recurso R_2 para que a segunda subtarefa seja executada com o uso simultâneo dos recursos R_1 e R_2 . A terceira subtarefa deve ser executada com o uso de mais uma unidade do recurso R_3 . Após, a unidade do recurso R_2 é liberada, e a quarta subtarefa é executada com o uso simultâneo dos recursos R_1 e R_3 . Encerrada esta última subtarefa, ambos recursos são liberados. Note-se que um comportamento cíclico está descrito. A ocorrência da transição t_7 não encerra o funcionamento do modelo, mas apenas a execução de mais uma tarefa.

Define-se ainda, na figura 3.8, os tempos envolvidos na execução das subtarefas ($Z(u_i)$) e a disponibilidade de recursos de cada tipo ($M_0(r_i)$). Cabe salientar, por fim, a distinção entre os tipos de recursos (R_i) e os lugares da rede de Petri que representam a sua disponibilidade (r_i).

Define-se o período básico (P_0) de um recurso como sendo a soma dos tempos associados a todos os lugares que simbolizam subtarefas nas quais o recurso é necessário. Nos modelos *Q1* equivale dizer que são todos os tempos da cadeia principal entre suas transições de alocação e liberação.

Para o exemplo da figura 3.8 temos:

$$P_0(R_1) = Z(u_1) + Z(u_2) + Z(u_3) + Z(u_4) = 12u.t.$$

$$P_0(R_2) = Z(u_2) + Z(u_3) = 8u.t.$$

$$P_0(R_3) = Z(u_3) + Z(u_4) = 7u.t.$$

O índice de desempenho isolado de um recurso do sistema é função da marcação inicial (M_0) do lugar que simboliza sua disponibilidade e do seu período básico.

Desta forma, a equação do índice do desempenho isolado de um recurso i é:

$$D_0(R_i) = \frac{M_0(r_i)}{P_0(R_i)}$$

Para o exemplo dado temos:

$$D_0(R_1) = 0,4167$$

$$D_0(R_2) = 0,375$$

$$D_0(R_3) = 0,4286$$

Note-se que o período básico refere-se aos recursos (R_i) e a marcação inicial refere-se aos lugares que simbolizam a disponibilidade de recurso (r_i). O cálculo do desempenho de um recurso é a base para todas as demais informações que podem ser extraídas do modelo.

Expandindo o cálculo do desempenho para o modelo como um todo, temos que o índice de desempenho geral do modelo é função dos desempenhos isolados. O desempenho máximo do sistema é dado pelo menor desempenho de um recurso isolado. No exemplo em discussão temos que o desempenho do sistema será:

$$D(N) = \min(D_0(R_1), D_0(R_2), D_0(R_3)) = D_0(R_2) = 0,375$$

Generalizando a definição para outros sistemas $Q1$, temos o desempenho geral do modelo como ¹:

$$D(N) = \min(D_0(R_i))$$

Note-se que, quando é definido o desempenho final do sistema, consideramos que os componentes mais rápidos deverão ser atrasados para igualarem-se ao componente mais lento. Esta adequação dos componentes mais rápidos causará uma sub-utilização dos recursos destes componentes. Analogamente, podemos concluir que a razão entre o desempenho real do modelo e o desempenho individual dos recursos fornecerá suas taxas de utilização:

$$U(R_i) = \frac{D(N)}{D_0(R_i)}$$

Para o exemplo dado, teremos:

$$U(R_1) = 0,9 \quad U(R_2) = 1,0 \quad U(R_3) = 0,875$$

Note-se que o recurso crítico (o mais lento) possui utilização máxima. Para todos os demais recursos a fórmula fornecerá um valor entre 1 (um) e 0 (zero), variando entre a utilização máxima e a ausência de utilização, respectivamente.

Uma vez que um tipo de recurso é o componente mais lento, os recursos já alocados durante sua transição de alocação serão retardados. Além de justificar

¹A notação $\min(D_0(R_i))$ significa o menor valor numérico entre os desempenhos individuais dos recursos R_i para i de 1 a R .

a sub-utilização de recursos, este fato provoca a formação de gargalos. Antes de tudo, porém, o retardo induzido aos recursos mais rápidos pode ser visto como uma adição ao período básico destes recursos. Esta adição pode ser quantificada, considerando que é feita uma equalização do desempenho isolado dos recursos mais rápidos ao desempenho da rede. Por exemplo, no modelo da figura 3.8, os componentes mais lentos são os recursos do tipo R_2 , logo, um tempo de espera será induzido no lugar w_1 , que precede a alocação destes recursos (gargalo).

$$D(N) = \frac{M_0(r_1)}{P_0(R_1) + Z_w(w_1)} \quad \text{ou} \quad Z_w(w_1) = \frac{M_0(r_1)}{D(N)} P_0(R_1)$$

Finalmente, a População média de qualquer lugar da rede de Petri será fornecida pela multiplicação do tempo associado ao lugar, pelo desempenho geral da rede. Desta forma, para os gargalos (lugares w_i), teremos:

$$Pm(w_i) = Z_w(w_i)D(N)$$

e para os lugares de utilização (lugares u_i):

$$Pm(u_i) = Z(u_i)D(N)$$

3.3 Métodos Aproximativos

Nesta parte, dois métodos aproximativos são vistos, para complementar o método analítico puro desenvolvido na parte anterior. Ambos os métodos aproximativos baseiam-se na obtenção de diversos valores intermediários para o índice de desempenho geral da rede, $D(N)$. A seguir, estes valores são ponderados segundo especificações complementares ao modelo em rede de Petri, e um novo valor para o índice de desempenho da rede é adotado. A partir deste momento todos os demais parâmetros (referentes à utilização e aos gargalos) são calculados com base no novo valor do índice de desempenho.

O primeiro método aproximativo está baseado na possibilidade da utilização não-uniforme de recursos. Este método aplica-se quando temos problemas onde o tempo de utilização de recursos é variável. Cabe lembrar que o *modelo Q* parte do princípio de que os tempos de utilização ($Z(u_i)$) são fixos.

O segundo método baseia-se na alocação não-uniforme de recursos. Este método é utilizado para a solução de problemas onde existe distinção entre recursos de um mesmo tipo. Tipicamente, estes problemas envolvem um casamento de padrões entre os recursos. Ressalta-se que o *modelo Q* não possui nenhum tipo de distinção entre recursos de um mesmo tipo.

3.3.1 Utilização Não-uniforme de Recursos

Este método fornece resultados levando em conta possíveis variações nos tempos de utilização de recursos. O *modelo Q1* não pode avaliar um problema onde a utilização de um recurso é temporalmente variável. A única alternativa dentro do escopo do *modelo Q1* é avaliar o problema supondo um único tempo médio de utilização de recursos. Esta alternativa, porém, pode fornecer resultados consideravelmente distantes daqueles encontrados na prática.

O método aproximativo, descrito nesta seção, aceita a definição de tempos de utilização através de um conjunto de valores possíveis com probabilidades associadas. Desta forma, a utilização de recursos de um determinado tipo pode ser feita segundo diversos padrões de comportamento, tendo valores distintos de tempo de utilização para cada um deles. Obviamente, a probabilidade de cada padrão é expressa como a probabilidade de cada valor do tempo de utilização.

Supondo que no exemplo visto anteriormente (figura 3.8), o tempo de utilização da quarta subtarefa ($Z(u_4)$) não seja mais determinístico, mas variável entre 0 (zero) e 4 (quatro) unidades de tempo, com probabilidades distintas para cada um dos casos, ou seja:

$Z(u_4)$	probabilidade
0 u.t.	0,3
1 u.t.	0,15
2 u.t.	0,1
3 u.t.	0,15
4 u.t.	0,3

Cabe salientar que o comportamento médio desta subtarefa permanece idêntico ao valor fixo utilizado anteriormente no cálculo com o *modelo Q*. O cálculo utilizando o método aproximativo revela, porém, uma imprecisão razoável na simplificação do uso do valor médio.

Basicamente, uma nova rede é calculada para cada possibilidade de tempo de utilização. A tabela 3.1 apresenta resumidamente os cálculos para este novo exemplo².

$Z(u_4)$	prob.	$D_0(R_1)$	$D_0(R_2)$	$D_0(R_3)$	$D(N)$	pond.
0 u.t.	0,3	0,5	0,375	0,6	0,375	0,1125
1 u.t.	0,15	0,4545	0,375	0,5	0,375	0,0563
2 u.t.	0,1	0,4167	0,375	0,4286	0,375	0,0375
3 u.t.	0,15	0,3846	0,375	0,375	0,375	0,0563
4 u.t.	0,3	0,3571	0,375	0,3333	0,3333	0,1
Total (desempenho real da rede) - $D(N)$ final						0,3626

Tabela 3.1: Resultados para a Rede com Utilização Variável de Recursos

Note-se que o valor obtido como desempenho final da rede (0,3626), levando em conta o comportamento variável do tempo da quarta subtarefa, é inferior ao valor originalmente gerado pelo *modelo Q* (0,375). Este fato deve-se principalmente ao desprezo do efeito nocivo da flutuação do tempo de utilização. Quando este tempo é superior à média, todo o modelo deve se adequar a ele (atrasar). Quando, ao contrário, este tempo é inferior à média, o ganho de desempenho não pode ser aproveitado pelas limitações do recurso crítico original

²Nesta tabela a coluna pond. significa ponderação, ou seja, desempenho ($D(N)$) vezes probabilidade (prob.).

do modelo. Evidentemente, se o tempo variável influenciar o período básico do recurso crítico, não haverá diferença entre os resultados do método aproximativo e do modelo Q .

Casos mais sofisticados, como a existência de vários tempos variáveis, também são analisáveis por este método aproximativo. Nestes casos, uma rede para cada combinação de possibilidades de tempos deve ser calculada. Supondo uma rede onde existam dois tempos de utilização variáveis, cada um com três valores possíveis, será preciso o cálculo de nove redes distintas. A figura 3.9 apresenta um exemplo onde os tempos de utilização (u_1 e u_2) são variáveis.

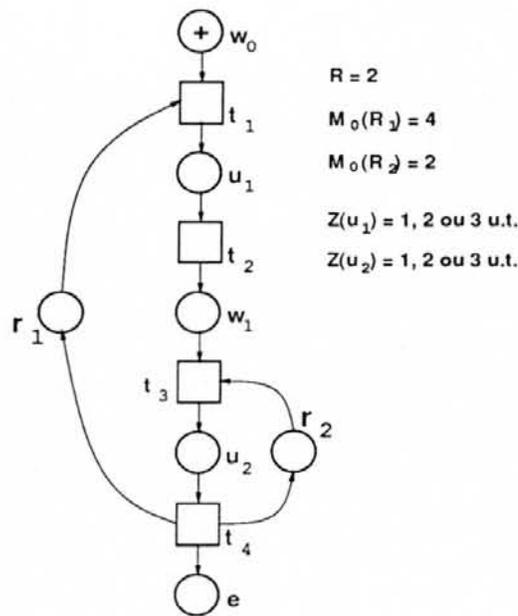


Figura 3.9: Rede com Tempos de Utilização Variáveis

Uma vez obtido o desempenho real da rede (ponderação dos desempenhos das diversas redes), todos os demais parâmetros são calculados normalmente. Cabe lembrar que todos os parâmetros calculados (utilização de recursos e gargalos) são calculados a partir do desempenho da rede. A tabela 3.2 apresenta os resultados obtidos para o exemplo da figura 3.9.

Note-se que o resultado obtido (0,99583) é ligeiramente inferior ao resultado esperado por uma média simples dos valores (1). Atribui-se esta diferença a perda causada pela flutuação do funcionamento dos componentes.

$Z(u_1)$	$Z(u_2)$	prob.	$D_0(R_1)$	$D_0(R_2)$	$D(N)$	pond.
1 u.t.	1 u.t.	0,0625	2	2	2	0,125
1 u.t.	2 u.t.	0,125	1,3333	1	1	0,125
1 u.t.	3 u.t.	0,0625	1	0,6666	0,666	0,04166
2 u.t.	1 u.t.	0,125	1,3333	2	1,333	0,16666
2 u.t.	2 u.t.	0,25	1	1	1	0,25
2 u.t.	3 u.t.	0,125	0,8	0,6666	0,666	0,08333
3 u.t.	1 u.t.	0,0625	1	2	1	0,0625
3 u.t.	2 u.t.	0,125	0,8	1	0,8	0,1
3 u.t.	3 u.t.	0,0625	0,6666	0,6666	0,666	0,04166
Total (desempenho real da rede) - $D(N)$ final						0,99583

Tabela 3.2: Resultados para a Rede com Duas Tarefas de Tempos Variáveis

3.3.2 Alocação Não-uniforme de Recursos

Este método fornece resultados considerando possíveis restrições à alocação de recursos. O *modelo Q* parte do princípio de que sempre que houver uma unidade de recurso disponível, não existirão solicitações bloqueadas à espera de tais recursos. No entanto, se admitirmos que nem todas as unidades de recursos de um mesmo tipo são idênticas, o princípio adotado pelo *modelo Q* não é verdade.

Supondo que existam relacionamentos entre recursos de dois tipos distintos que serão utilizados simultaneamente, é possível admitir que uma situação de bloqueio possa ocorrer. Por exemplo, seja uma tarefa onde operários e ferramentas são utilizados simultaneamente. Após a alocação de um operário para a execução de uma subtarefa, este não poderia executá-la por falta da ferramenta de sua preferência, mesmo que outras ferramentas distintas possam estar sem uso no momento.

O método aproximativo de resolução deste tipo de problema baseia-se na divisão do problema em diversas redes onde a disponibilidade de recursos é distinta. A distinção da disponibilidade das unidades de recurso é feita através da variação da marcação inicial (M_0) do lugar que representa a disponibilidade

do recurso em questão. Neste método, a ponderação dos resultados de cada uma das redes com marcação inicial distinta também é feita através de especificações externas à rede original analisável pelo *modelo Q*.

Supondo que no exemplo do *modelo Q* (figura 3.8) o uso de recursos do tipo R_3 dependa da unidade específica do recurso do tipo R_2 alocado anteriormente. Neste caso é preciso definir a probabilidade de choques de preferência. Hipoteticamente admitamos os seguintes valores:

- Seja de três décimos (0,3) a probabilidade de que não exista disputa entre os recursos do tipo R_2 por recursos do tipo R_3 , ou seja, nenhuma das unidades R_2 deseja a mesma unidade R_3 que outra unidade R_2 ;
- Seja de três quintos (0,6) a probabilidade de que a disputa entre os recursos do tipo R_2 pelos recursos do tipo R_3 seja parcial, ou seja, duas unidades de recursos R_2 desejam utilizar a mesma unidade de R_3 ;
- Seja de um décimo (0,1) a probabilidade de que a disputa entre os recursos do tipo R_2 pelos recursos do tipo R_3 seja total, ou seja, todas as unidades do tipo R_2 desejam a mesma unidade do tipo R_3 .

Para esta configuração de probabilidade, o cálculo do desempenho através deste método aproximativo está resumidamente descrito na tabela 3.3.

$M_0(r_3)$	prob.	$D_0(R_1)$	$D_0(R_2)$	$D_0(R_3)$	$D(N)$	pond.
1	0,1	0,4167	0,375	0,1429	0,1429	0,0143
2	0,6	0,4167	0,375	0,2857	0,2857	0,1714
3	0,3	0,4167	0,375	0,4286	0,375	0,1125
Total (desempenho real da rede) - $D(N)$ final						0,2982

Tabela 3.3: Resultados para a Rede com Alocação Variável de Recursos

O valor obtido pelo método analítico considerando o comportamento preferencial das unidades do recurso do tipo R_2 frente à alocação das unidades do recurso do tipo R_3 é consideravelmente mais baixo que o valor obtido pelo *modelo*

Q (0,375). Este resultado é esperado, pois se o *modelo Q* parte do princípio de que a alocação de recursos R_3 é indistinta, toda queda de desempenho proporcionada pelas disputas de recursos especializados (R_3) é desprezada. Neste sentido, os resultados do *modelo Q* são demasiadamente otimistas frente aos resultados fornecidos pelo método aproximativo. Salienta-se que os valores escolhidos para as probabilidades são essenciais para a determinação do valor final do desempenho.

Neste método aproximativo, a exemplo do método da seção anterior, o novo valor obtido para o índice de desempenho da rede deve ser utilizado para calcular todos os demais parâmetros já citados na descrição do *modelo Q*. Ilustra-se com o cálculo dos demais parâmetros o fenômeno modelado por este método aproximativo. Tipicamente, encontra-se em todos os recursos uma taxa de utilização inferior a 1 (um). O significado destes números é o de que todos os recursos, mesmo o recurso crítico, possuem uma carga ociosa, que por sua vez é fruto da perda de desempenho causada pelas disputas entre recursos de um mesmo tipo. Para o exemplo da figura 3.8 considerando as ponderações desta seção temos:

$$U(R_1 = \frac{0,2982}{0,4167} = 0,7156$$

$$U(R_2 = \frac{0,2982}{0,375} = 0,7952$$

$$U(R_3 = \frac{0,2982}{0,4286} = 0,6958$$

Os tempos induzidos de espera são calculados com base no acréscimo do período básico dos recursos devido a adequação dos demais componentes aos recursos mais lentos. No entanto, neste exemplo o desempenho real da rede é inferior aos desempenhos individuais dos componentes. Esta situação provoca o fenômeno de adequação dos períodos básicos (indução de tempos de espera) devido a limitação do recurso com distinção de preferência (no exemplo o recurso R_3). Desta forma, existirá indução de tempos de espera em ambos os lugares anteriores a alocação deste recurso (u_2 e r_3). Por fim, o período do recurso

mais externo (R_1) ainda pode precisar de ajustes, causando indução de tempos de espera também na alocação do segundo recurso (w_1). Para o exemplo apresentado temos os seguintes resultados:

$$\begin{aligned} Z_w(r_1) &= 0 & Z_w(r_2) &= 0 \\ Z_w(r_3) &= \frac{3}{0,2982} - 7 = 3,0604 & Z_w(u_2) &= \frac{3}{0,2982} - 8 = 2,0604 \\ Z_w(w_1) &= \frac{5}{0,2982} - 12 = 4,7673 \end{aligned}$$

Finalmente, a população média dos lugares da cadeia principal são calculados a partir dos seus tempos associados (induzidos ou não) e do desempenho final da rede. Para o exemplo dado temos os resultados abaixo. Note-se que a soma da população média dos lugares que compõem o período básico de cada recurso deve ser igual ao número de unidades disponíveis do recurso.

$$\begin{aligned} P_m(r_1) &= 0 & P_m(r_2) &= 0 & P_m(r_3) &= 0,9126 \\ P_m(u_1) &= 0,5964 & P_m(u_2) &= 0,8946 & P_m(u_3) &= 1,4191 \\ P_m(u_4) &= 0,5964 & P_m(w_1) &= 1,4216 & P_m(w_2) &= 0,6144 \end{aligned}$$

4 INTERCONEXÕES DE PROCESSADORES

Este capítulo apresenta a realidade para a qual os modelos de avaliação de desempenho são desenvolvidos. A realidade escolhida foram as interconexões de processadores da classe multiprocessador. Especificamente, dentro desta classe foram escolhidas as interconexões do tipo multi-estágios. Por questões práticas os multi-estágios escolhidos como alvo foram aqueles ditos uniformes. No entanto, os modelos desenvolvidos são suficientemente flexíveis para adaptarem-se a qualquer tipo de multi-estágios, sendo necessárias pequenas alterações.

A primeira parte deste capítulo descreve as interconexões de processadores genericamente e cita a taxonomia adotada para a classificação das interconexões de processadores. A segunda parte define as interconexões multi-estágios diferenciando-as das demais interconexões multiprocessador. Finalmente, a terceira parte faz a análise quantitativa das interconexões multi-estágios através da definição precisa de quais parâmetros devem ser considerados relevantes ao desempenho destas interconexões.

4.1 Classificação de Interconexões

A forma pela qual diversos processadores são interconectados em uma máquina paralela é denominada interconexão. Desta forma, uma interconexão de processadores é o meio pelo qual processadores podem trocar informações. Os componentes fixos de uma interconexão são os processadores, eventualmente outros componentes básicos são utilizados. Entre eles é frequente o uso de memórias de uso compartilhado, barramentos de comunicação e arbitradores (chaveadores).

Visando facilitar o seu estudo, as interconexões podem ser classificadas em diversos grupos segundo suas características. Estas características podem va-

riar desde sua funcionalidade, tolerância a falhas ou mesmo seu aspecto físico, sua topologia. Nesta dissertação adotou-se uma taxonomia baseada principalmente na topologia da interconexão.

A taxonomia adotada baseia-se no trabalho de Anderson e Jensen [AND75]. Este trabalho classifica interconexões de processadores segundo diversos critérios. Note-se que para efeitos de classificação não importa se a interconexão é aplicada a computadores afastados geograficamente (redes), ou a simples processadores agindo internamente em uma máquina (arquiteturas).

Na taxonomia adotada optou-se pela topologia da interconexão como critério básico de classificação. Na figura 4.1 temos a representação genérica adotada para as interconexões incluídas nesta taxonomia. Esta taxonomia foi adotada por ser a mais adequada aos propósitos do projeto *ADMP* [ADM90], no qual esta dissertação se insere. A seguir são brevemente descritos os grupos que compõem esta taxonomia e suas respectivas classes.

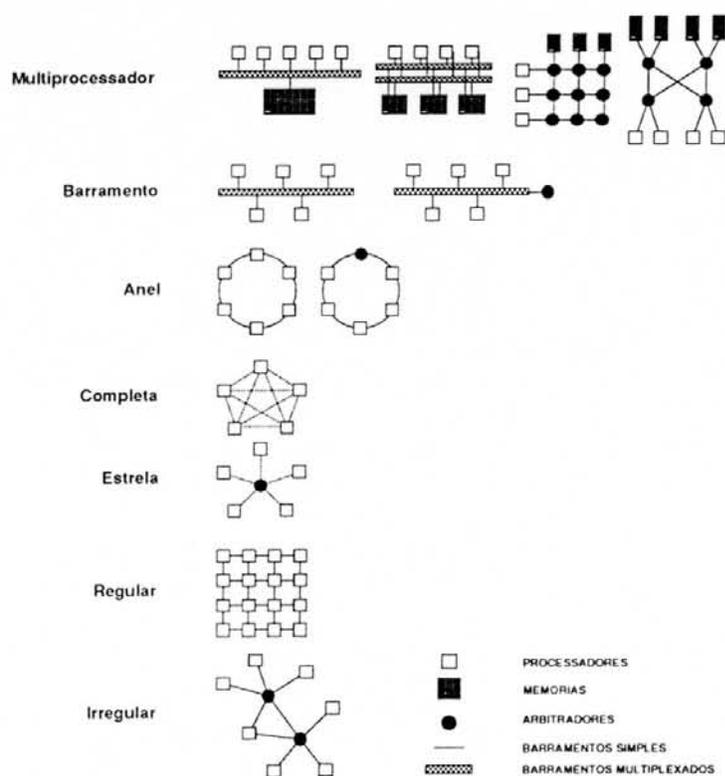


Figura 4.1: Taxonomia de Interconexões de Processadores

- **MULTIPROCESSADOR:** Todos os processadores são conectados a um ou mais módulos de memória global por onde mensagens são trocadas. Este grupo subdivide-se em quatro classes segundo a forma de conexão entre processadores e módulos de memória: barramento único, múltiplos barramentos, crossbar e multi-estágios.
- **BARRAMENTO:** A comunicação ocorre através de um barramento global ao qual todos os processadores estão conectados. Este grupo subdivide-se em duas classes: barramento standard e barramento com arbitrador central.
- **ANEL:** A característica básica deste grupo é uma arquitetura circular onde cada processador está conectado a dois outros processadores vizinhos. Pelo anel circulam tokens responsáveis pela troca de mensagens. Este grupo subdivide-se em duas classes: anel standard e anel com arbitrador.
- **COMPLETA:** Todos os processadores possuem um caminho dedicado que os interconecta a todos os demais processadores da interconexão. A comunicação entre dois processadores se dá através do caminho direto entre eles. Este grupo não possui subdivisão em classes.
- **ESTRELA:** Todos os processadores estão conectados por barramentos dedicados a um arbitrador central pelo qual mensagens são trocadas. Este grupo também não se subdivide em classes.
- **REGULAR:** - Os processadores são conectados entre si por caminhos dedicados que possuem algum padrão de ligação. A comunicação se dá através da difusão de mensagens na rede desde o processador fonte até o processador destino. Utiliza-se como representação genérica deste grupo a classe Retangular. Porém, este grupo subdivide-se em tantas classes quantos forem os possíveis padrões de ligação, por exemplo: árvore, hipercubo.

- **IRREGULAR:** A semelhança do grupo regular os processadores são conectados por uma malha de ligações, porém esta malha não possui nenhum padrão que a caracterize. As classes podem ser estabelecidas neste grupo de acordo com critérios específicos. No entanto, tais critérios são dependentes da aplicação dada à topologia em questão.

4.2 Interconexões Multiprocessador

As interconexões multi-estágios fazem parte do grupo multiprocessador. Neste grupo existem três outras classes de interconexões que diferenciam-se dos multi-estágios pela malha de interconexão utilizada. Desta forma no grupo multiprocessador a definição das classes de interconexão é feita por intermédio da observação da malha de interconexão. Note-se que a malha de interconexão dos multiprocessadores é utilizada para ligar processadores a módulos de memória e não processadores entre si.

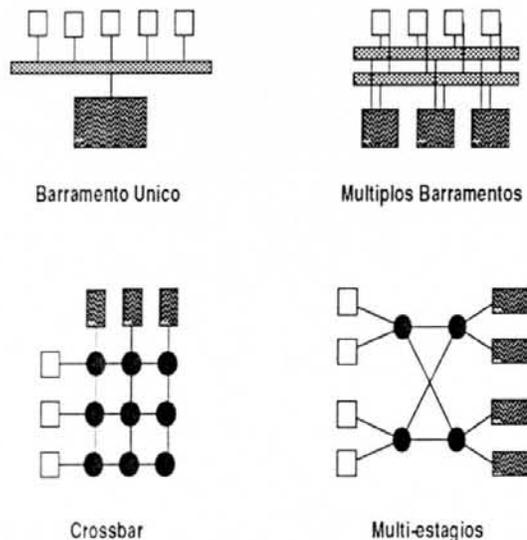


Figura 4.2: Classes de Multiprocessador

Dentro do grupo multiprocessador, quatro classes são definidas: barramento único, múltiplos barramentos, crossbar e multi-estágios. Sob o ponto de vista do desempenho, a diferença básica entre estas classes é a contenção do

acesso dos processadores às memórias pela malha de interconexão. A figura 4.2 apresenta as classes pertencentes ao grupo multiprocessador.

- Na classe **barramento único**, a malha consiste de um único barramento arbitrado que possibilita o acesso de todos os processadores a todos os módulos de memória. Nesta classe a contenção do acesso é máxima, pois não é possível o acesso simultâneo de processadores a módulos de memória.
- A classe **múltiplos barramentos** é apenas uma extensão da classe anterior, pois o princípio de funcionamento é o mesmo. Neste caso porém a contenção não é máxima, e sim limitada de acordo com o número de barramentos utilizados.
- A terceira classe, **crossbar**, possui uma diferença fundamental das demais classes, pois possui uma malha com arbitradores dedicados para cada par processador/módulo de memória. Desta forma não existe nenhuma contenção do acesso aos módulos de memória causada pela malha de interconexão.
- Analogamente às interconexões da classe crossbar, a classe **multi-estágios** possui uma malha de arbitradores. Nesta classe, porém, utilizam-se arbitradores compartilhados. A escolha dos arbitradores a utilizar é feita de acordo com o módulo de memória requisitado. Desta forma, a contenção do acesso aos módulos de memória é ligeiramente mais forte que nas interconexões crossbar. Nesta classe não só as memórias são compartilhadas, mas também os caminhos. Apesar do caminho entre um processador e um módulo de memória ser único, ele não é de uso exclusivo. Este compartilhamento, no entanto, é feito em partes representadas pelos caminhos entre cada estágio.

4.2.1 Interconexões Multi-estágios

A classe de interconexões multi-estágios é definida como sendo o conjunto de interconexões de processadores onde estes estão conectados aos módulos de memória através de uma malha de arbitradores compartilhados [HWA84]. Estes arbitradores podem nitidamente ser agrupados em dois ou mais estágios.

Uma interconexão crossbar é composta de um único estágio que interconecta diretamente processadores e módulos de memória. Supondo um crossbar com quatro processadores e quatro módulos de memória, o estágio único é composto da malha crossbar 4x4 de arbitradores. Se dividirmos esta malha em dois estágios de arbitração com dois arbitradores em cada estágio podemos conseguir uma interconexão multi-estágios para interconectar um número igual de processadores e módulos de memória (figura 4.3).

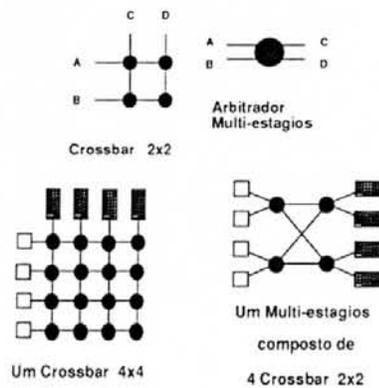


Figura 4.3: Crossbar e Multi-estágios

As interconexões multi-estágios, no entanto, não tem necessariamente um mesmo número de processadores e módulos de memória. Neste caso é feita uma distinção entre multi-estágios uniformes e não-uniformes (figura 4.4). A distinção básica entre estes dois tipos de multi-estágios verifica-se pela homogeneidade dos arbitradores utilizados. Cada arbitrador é na verdade uma chave crossbar que possui um certo número de entradas (fan-in) e saídas (fan-out). Caso um multi-estágio possua arbitradores distintos na sua malha, ou mesmo

arbitradores com fan-in e fan-out diferentes, será considerado um multi-estágios não-uniforme.

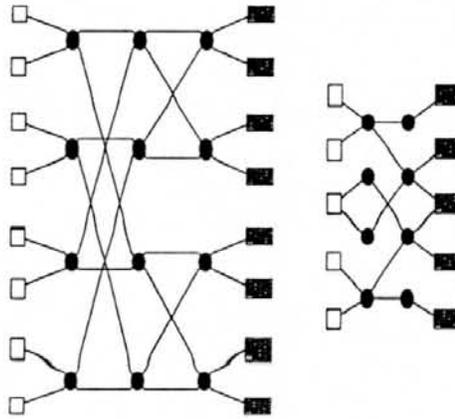


Figura 4.4: Multi-estágios Uniformes e Não-uniformes

Dentre os multi-estágios uniformes foram escolhidos para servirem de exemplo nesta dissertação os multi-estágios binários. Estes distinguem-se por possuir um fan-in e fan-out igual a dois. Particularmente, as redes Omega (figura 4.5) foram escolhidas. Redes Omega são interconexões multi-estágios uniformes, binários com embaralhamento perfeito (*perfect-shuffle*).

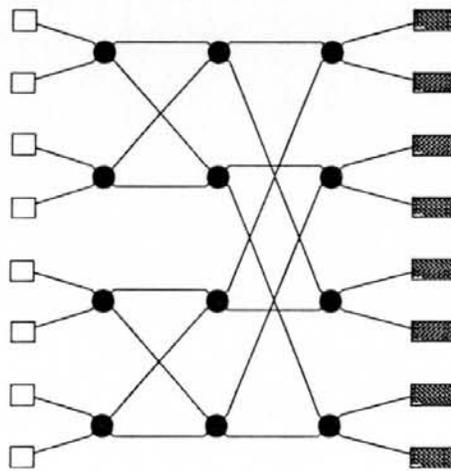


Figura 4.5: Interconexões Multi-estágios Tipo Omega

Nas redes Omega três restrições básicas são impostas:

- número de processadores igual ao número de módulos de memória é uma potência de dois;

- arbitradores binários (duas entradas e duas saídas - fan-in e fan-out igual a dois);
- embaralhamento perfeito entre estágios (topologia de conexão embaralhando dois na N processadores até o N -ésimo estágio);

Estas restrições visam simplificar o processo de utilização das interconexões. Adicionalmente esta simplificação facilita a definição da topologia da interconexão. Por exemplo, se sabemos o número de processadores (n), temos imediatamente o número de módulos de memória (também n), podemos calcular o número de arbitradores por estágio (metade de n) e o número de estágios (raiz quadrada de n). Finalmente, devido ao embaralhamento perfeito sabemos que a malha de arbitradores embaralha os dois primeiros processadores no primeiro estágio, os quatro primeiros no segundo estágio e assim por diante até embaralhar a totalidade de processadores.

4.3 Análise Quantitativa

Todas as interconexões multi-estágios apresentam características comuns que são consideradas como relevantes ao desempenho. Para sistematizar, estas características são agrupadas em aspectos estruturais, físicos e comportamentais. A definição da classe e os aspectos estruturais definem a topologia da interconexão em si e seus componentes (processadores, memórias e arbitradores). Os aspectos físicos definem as restrições físicas associadas aos componentes da interconexão. Os aspectos comportamentais definem genericamente o ambiente de utilização previsto, ou seja, a carga de trabalho esperada para a interconexão.

A definição dos aspectos estruturais, físicos e comportamentais especifica a interconexão a ser analisada e o seu ambiente genérico de uso. Segundo alguns autores da área [FER78, LAZ84], a carga do sistema (*system workload*) deve

ser diferenciada dos parâmetros descritivos da interconexão. No entanto, neste trabalho todos os aspectos são considerados indistintamente como parâmetros de entrada para a análise quantitativa proposta. Assume-se desta forma a análise conjunta da máquina (*hardware*) e do seu ambiente de execução (aplicação).

Para todas as interconexões pertencentes à classe multi-estágios citada é possível definir um elenco de parâmetros comuns que são, então, considerados para análise desta classe. Cabe salientar que a interconexão Omega diferencia-se das demais pelo tratamento dado a estas informações comuns, bem como à relação implícita entre alguns aspectos estruturais. Desta forma, a adaptação das análises desenvolvidas nesta dissertação para outras interconexões multi-estágio envolve basicamente um tratamento distinto para este conjunto de informações primitivas.

Segundo a divisão proposta temos:

Aspectos Estruturais:

- Número de processadores;
- Número de módulos de memória;
- Número de estágios;
- Cardinalidade dos arbitradores (fan-in e fan-out);
- Topologia de conexões (embaralhamento);

Cabe lembrar que para a rede Omega, a nível de aspectos estruturais, basta definirmos o número de processadores. O número de módulos de memória é idêntico ao número de processadores. A cardinalidade dos arbitradores é 2x2 (arbitradores binários - duas entradas, duas saídas). O número de estágios é o logaritmo na base dois do número de processadores já definido. Finalmente, a

topologia específica para as conexões entre processadores e módulos de memória global é definida como sendo de embaralhamento perfeito.

Aspectos Físicos:

- Atraso dos arbitradores;

Aspectos Comportamentais:

- Tempo médio de processamento local (quanto tempo um processador fica em média sem acessar um módulo de memória global);
- Tempo médio de processamento global (quanto tempo um processador fica em média acessando um módulo de memória global);
- Probabilidade de acesso de todos os processadores a todos os módulos de memória;

Certas restrições foram adotadas com o intuito de facilitar o processo de especificação (definição dos aspectos estruturais, físicos e comportamentais) da interconexão. Por exemplo, considera-se um tempo único de processamento local. Talvez pareça mais natural definir especificamente o tempo de execução de operações locais e o tempo de inatividade (*idle time*). No entanto, esta decisão foi tomada com base na pouca importância de diferenciação entre o tempo útil (execução de operações locais) e o tempo ocioso inerente a cada processador. Em ambos os casos, trata-se de tempos onde não é feito acesso a módulos de memória global.

5 MODELAGEM ATRAVÉS DE RFE

Este capítulo descreve o desenvolvimento de um modelo para as interconexões multi-estágios do tipo Omega através do método RFE. A análise desenvolvida é suficientemente genérica para aplicar-se, sem alterações, a qualquer variação dos parâmetros expostos na análise quantitativa do capítulo anterior (seção 4.3). No entanto, para facilitar a explicação do modelo desenvolvido, um conjunto de parâmetros será fixado. O modelo apresentado terá as seguintes características:

- número de processadores: 4
- número de módulos de memória: 4
- número de estágios: 2
- cardinalidade dos arbitradores (*fan-in* e *fan-out*): 2
- atraso dos arbitradores: 0,1 u.t.
- tempo médio de processamento local: 1 u.t.
- tempo médio de processamento global: 1 u.t.
- probabilidade de acesso dos processadores aos módulos de memória: 25% (equiprovável)

A primeira parte deste capítulo contém a descrição do modelo fundamental em redes de filas. Este modelo extremamente simples tem por função expor a técnica básica de modelagem, bem como dirimir as possíveis dúvidas ainda existentes na compreensão do sujeito da avaliação. A segunda parte apresenta incrementalmente o modelo refinado. São expostas as duas sofisticações do modelo fundamental adotadas para proporcionar um ajuste fino do modelo ao

funcionamento esperado das interconexões multi-estágios. Em ambas as partes, resultados são exibidos e críticas às limitações do modelo são feitas.

5.1 Modelo Fundamental

O modelo fundamental descreve da forma mais simples possível as interconexões multi-estágios através do método RFE. A preocupação básica deste modelo é a descrição do ciclo de funcionamento da interconexão. Desta forma, apresenta-se o recurso primário e seu ciclo de funcionamento. Após, resultados numéricos são apresentados e limitações do modelo fundamental são discutidas.

5.1.1 Recurso Primário e Ciclo Básico

O método RFE parte do princípio que existe um recurso básico em cada sistema. Este recurso básico é, então, adotado como cliente do modelo de filas, pois ele utiliza os demais recursos segundo seu ciclo de funcionamento.

Nas interconexões multi-estágios, como nas demais classes do grupo multiprocessador, os recursos primários são, sem dúvida, os processadores. Os demais recursos, arbitradores e memórias, são vistos como recursos passivos que o processador aloca segundo suas necessidades. Esta abordagem tem eco na realidade, pois em uma interconexão multi-estágios somente o processador é dotado de inteligência, no sentido computacional da palavra.

Sendo os processadores adotados como clientes, resta definir o seu ciclo básico de funcionamento. Uma interconexão do tipo multiprocessador tem duas tarefas básicas: processamento local e processamento global. Na primeira, o processador trabalha com recursos a ele dedicados e portanto não necessita de nenhum outro recurso. A segunda tarefa básica consiste no uso dos módulos de

memória. Estes são de uso compartilhado e portanto o processador deve alocar recursos para poder efetivar esta tarefa. Tratando-se de uma interconexão multi-estágios, mais uma tarefa é necessária. Esta tarefa consiste em alocar o caminho para o acesso a memória, ou seja, alocar os arbitradores necessários à conexão. Desta forma, o ciclo básico de um processador é composto de três tarefas:

- processamento local (PL)
- conexão (Cx)
- processamento global (PG)

A figura 5.1 apresenta o modelo fundamental em redes de filas. Neste modelo as seguintes informações numéricas são associadas:

- número de clientes na rede:

$$N = 4;$$

- probabilidade de rotação:

$p_{i,j}$	PL	Cx	PG
PL	0	1	0
Cx	0	0	1
PG	1	0	0

- tempos médios de prestação de serviço (em u.t.):

$$S_{PL} = 1 \quad S_{Cx} = 0,2 \quad S_{PG} = 1$$

- número de servidores: $C_{PL} = 4 \quad C_{Cx} = 4 \quad C_{PG} = 4$

- taxas de prestação de serviço (em clientes por u.t.):

$$\begin{aligned} \mu_{PL}(0) = 0 \quad \mu_{PL}(1) = 1 \quad \mu_{PL}(2) = 2 \quad \mu_{PL}(3) = 3 \quad \mu_{PL}(4) = 4 \\ \mu_{Cx}(0) = 0 \quad \mu_{Cx}(1) = 5 \quad \mu_{Cx}(2) = 10 \quad \mu_{Cx}(3) = 15 \quad \mu_{Cx}(4) = 20 \\ \mu_{PG}(0) = 0 \quad \mu_{PG}(1) = 1 \quad \mu_{PG}(2) = 2 \quad \mu_{PG}(3) = 3 \quad \mu_{PG}(4) = 4 \end{aligned}$$

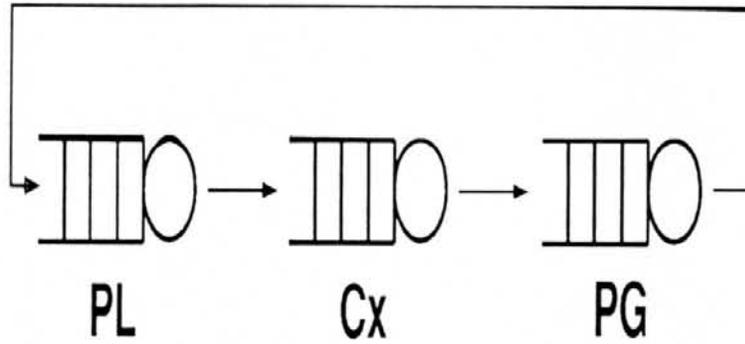


Figura 5.1: Modelo Fundamental em RFE

Note-se que o tempo de conexão (S_{Cx}) é o dobro do atraso dos arbitradores, pois cada conexão envolve o uso de tantos arbitradores quanto forem o número de estágios. Note-se ainda que, tratando-se de um modelo em redes forma-produto, os tempos de utilização são definidos por uma distribuição exponencial. Os valores adotados (S_{PL} , S_{Cx} e S_{PG}) são apenas médias destas distribuições.

Os servidores na estação *PL* correspondem ao número de clientes, pois os processadores não necessitam esperar nenhum recurso para executar suas tarefas locais. Desta forma sempre que um cliente chegar a esta estação será imediatamente atendido. Os servidores da estação *Cx* correspondem ao máximo possível de caminhos simultâneos na malha de arbitradores. Os servidores da estação *PG* correspondem ao número de memórias disponíveis ao acesso compartilhado.

5.1.2 Resultados e Limitações

O modelo fundamental para o caso definido apresenta os resultados da tabela 5.1. Note-se que a taxa de saída de todas as estações é idêntica. Em um sistema fechado este fato é razoável, pois mesmo que exista um recurso crítico (gargalo), todos os demais recursos devem ajustar-se a sua velocidade. No entanto, as taxas de utilização das estações são distintas, pois não existe equivalência entre o tempo dispendido pelos clientes em cada estação. Analogamente, o tempo

de resposta e a população média de clientes em cada estação também são distintos. As limitações mais evidentes deste modelo referem-se ao tratamento otimista dado às possíveis colisões na disputa por memórias e por arbitradores.

i	$D_i(N)$	$U_i(N)$	$N_i(N)$	$W_i(N)$
PL	1,8182	0,9115	1,8182	1,0000
Cx	1,8182	0,3170	0,3636	0,2000
PG	1,8182	0,9115	1,8182	1,0000

Tabela 5.1: Resultados para o Modelo Fundamental de RFE

No que se refere a colisões por memória, os resultados referentes à estação PG são idênticos aos resultados da estação PL . Este é um resultado visivelmente errôneo. Na estação PL está representada uma tarefa (processamento local) que não necessita de outros recursos exceto os próprios processadores. Em contrapartida, na estação PG utilizam-se as memórias de uso compartilhado, logo é razoável esperar a ocorrência de colisões que reduziriam a utilização dos servidores desta estação. Estas colisões seriam traduzidas no modelo em RFE ainda como clientes presentes na fila de espera, mesmo quando existem servidores disponíveis.

Os arbitradores, por sua vez, não estão representados diretamente no modelo fundamental. O conceito abstrato de caminhos, que são dois pares de arbitradores, é utilizado para estabelecer o número de servidores da estação Cx . Admitindo também a disputa dos arbitradores, é possível considerar otimista, e portanto impreciso, o modelo fundamental que sempre possibilita a formação de quatro caminhos simultâneos na malha de arbitradores.

5.2 Modelo Refinado

O objetivo da definição de um modelo refinado é atacar as limitações do modelo fundamental. Desta forma, os problemas referentes às disputas por memória e arbitradores são atacados nas duas próximas seções. No entanto,

outras limitações inerentes ao próprio método e ao modelo desenvolvido ainda existem e são apresentadas ao final deste capítulo.

Cabe lembrar que todo refinamento aplicado ao modelo fundamental tem um custo associado. Este custo manifesta-se, no mínimo, em razão do tempo dispendido para achar um modelo mais adequado. Nos refinamentos desenvolvidos a seguir, os modelos recebem um aumento no número de estações e portanto são ligeiramente mais lentos para a obtenção de resultados.

5.2.1 Colisão sobre Memórias

O princípio básico da modelagem da disputa de memórias é a divisão da estação PG (figura 5.1) em quatro estações distintas ($M1$, $M2$, $M3$ e $M4$) representado cada uma delas um dos módulos de memória disponíveis. Desta forma, a probabilidade de rotação associada a saída da estação Cx é responsável pela ocorrência de colisões na disputa de módulos de memória.

A figura 5.2 apresenta este refinamento aplicado ao modelo fundamental. Neste modelo refinado as seguintes informações numéricas são associadas:

- número de clientes na rede:

$$N = 4;$$

- probabilidade de rotação:

$p_{i,j}$	PL	Cx	$M1$	$M2$	$M3$	$M4$
PL	0	1	0	0	0	0
Cx	0	0	0.25	0.25	0.25	0.25
$M1$	1	0	0	0	0	0
$M2$	1	0	0	0	0	0
$M3$	1	0	0	0	0	0
$M4$	1	0	0	0	0	0

- tempos médios de prestação de serviço (em u.t.):

$$S_{PL} = 1 \quad S_{Cx} = 0.2 \quad S_{M1} = 1 \quad S_{M2} = 1 \quad S_{M3} = 1 \quad S_{M4} = 1$$

- número de servidores:

$$C_{PL} = 4 \quad C_{Cx} = 4 \quad C_{M1} = 1 \quad C_{M2} = 1 \quad C_{M3} = 1 \quad C_{M4} = 1$$

- taxas de prestação de serviço (em clientes por u.t.):

$$\begin{aligned} \mu_{PL}(0) = 0 \quad \mu_{PL}(1) = 1 \quad \mu_{PL}(2) = 2 \quad \mu_{PL}(3) = 3 \quad \mu_{PL}(4) = 4 \\ \mu_{Cx}(0) = 0 \quad \mu_{Cx}(1) = 5 \quad \mu_{Cx}(2) = 10 \quad \mu_{Cx}(3) = 15 \quad \mu_{Cx}(4) = 20 \end{aligned}$$

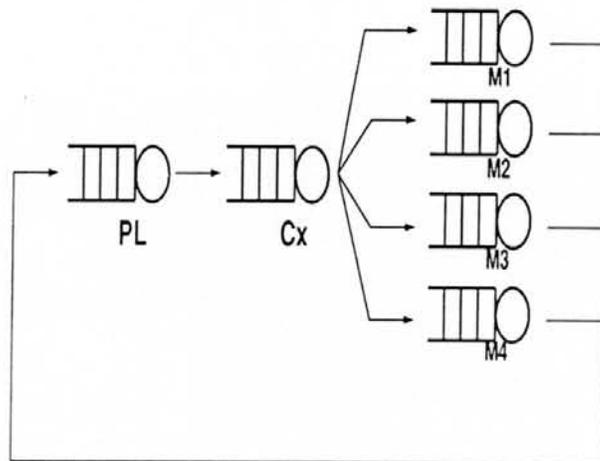


Figura 5.2: Modelo de Colisão sobre Memórias em RFE

Basicamente, a estação *PG*, uma estação multi-servidores, foi dividida em quatro estações mono-servidor. As preferências dos processadores pelos módulos de memória são expressas pela probabilidade de rotação. Tratando-se de uma rede forma-produto a escolha dos caminhos é feita segundo um processo Markoviano. Portanto podem existir desequilíbrios momentâneos na distribuição de clientes para cada uma das estações memórias (*M*'s). Estes desequilíbrios provocam o eventual desvio de mais de um cliente para uma mesma estação, representando assim a possibilidade de colisões.

Os resultados da tabela 5.2 mostram a maior precisão obtida com este refinamento.

i	$D_i(N)$	$U_i(N)$	$N_i(N)$	$W_i(N)$
PL	1,5496	0,8405	1,5496	1,0000
C_x	1,5496	0,2740	0,3099	0,2000
$M1$	0,3874	0,3874	0,5351	1,3812
$M2$	0,3874	0,3874	0,5351	1,3812
$M3$	0,3874	0,3874	0,5351	1,3812
$M4$	0,3874	0,3874	0,5351	1,3812

Tabela 5.2: Resultados para o Modelo com Colisão sobre Memórias em RFE

5.2.2 Colisão sobre Arbitradores

A escolha dos arbitradores é feita de acordo com o módulo de memória desejado. Desta forma, de um a quatro caminhos simultâneos são permitidos. A representação escolhida para o conflito da disputa pelos arbitradores é uma malha de estações similar àquela encontrada na interconexão multi-estágio original.

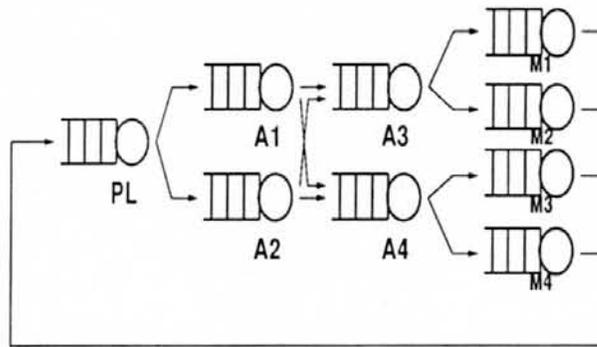


Figura 5.3: Modelo de Colisão sobre Arbitradores em RFE

A figura 5.3 apresenta o modelo fundamental com os dois refinamentos propostos, a modelagem da colisão sobre memórias e a modelagem da colisão sobre arbitradores. Neste modelo as informações numéricas associadas são:

- número de clientes na rede:

$$N = 4;$$

- probabilidade de rotação:

$p_{i,j}$	PL	$A1$	$A2$	$A3$	$A4$	$M1$	$M2$	$M3$	$M4$
PL	0	0,5	0,5	0	0	0	0	0	0
$A1$	0	0	0	0,5	0,5	0	0	0	0
$A2$	0	0	0	0,5	0,5	0	0	0	0
$A3$	0	0	0	0	0	0,5	0,5	0	0
$A4$	0	0	0	0	0	0	0	0,5	0,5
$M1$	1	0	0	0	0	0	0	0	0
$M2$	1	0	0	0	0	0	0	0	0
$M3$	1	0	0	0	0	0	0	0	0
$M4$	1	0	0	0	0	0	0	0	0

- tempos médios de prestação de serviço (em u.t.):

$$S_{PL} = 1 \quad S_{A1} = 0,1 \quad S_{A2} = 0,1 \quad S_{A3} = 0,1 \quad S_{A4} = 0,1$$

$$S_{M1} = 1 \quad S_{M2} = 1 \quad S_{M3} = 1 \quad S_{M4} = 1$$

- número de servidores:

$$C_{PL} = 4 \quad C_{A1} = 2 \quad C_{A2} = 2 \quad C_{A3} = 2 \quad C_{A4} = 2$$

$$C_{M1} = 1 \quad C_{M2} = 1 \quad C_{M3} = 1 \quad C_{M4} = 1$$

- taxas de prestação de serviço (em clientes por u.t.):

$$\mu_{PL}(0) = 0 \quad \mu_{PL}(1) = 1 \quad \mu_{PL}(2) = 2 \quad \mu_{PL}(3) = 3 \quad \mu_{PL}(4) = 4$$

$$\mu_{A1}(0) = 0 \quad \mu_{A1}(1) = 10 \quad \mu_{A1}(2) = 20 \quad \mu_{A1}(3) = 20 \quad \mu_{A1}(4) = 20$$

$$\mu_{A2}(0) = 0 \quad \mu_{A2}(1) = 10 \quad \mu_{A2}(2) = 20 \quad \mu_{A2}(3) = 20 \quad \mu_{A2}(4) = 20$$

$$\mu_{A3}(0) = 0 \quad \mu_{A3}(1) = 10 \quad \mu_{A3}(2) = 20 \quad \mu_{A3}(3) = 20 \quad \mu_{A3}(4) = 20$$

$$\mu_{A4}(0) = 0 \quad \mu_{A4}(1) = 10 \quad \mu_{A4}(2) = 20 \quad \mu_{A4}(3) = 20 \quad \mu_{A4}(4) = 20$$

Note-se que o tempo médio de atendimento das estações A ($A1$, $A2$, $A3$ e $A4$) é igual ao atraso dos arbitradores, pois neste novo modelo, cada estação corresponde a um único arbitrador. O número de servidores das estações arbitradores é igual à capacidade máxima de atendimento dos arbitradores (dois). As demais características permanecem similares ao modelo refinado anterior (colisão sobre memórias).

Os resultados da tabela 5.3 apontam a evolução obtida com os dois refinamentos adicionados ao modelo fundamental. Note-se que a modelagem da colisão sobre caminhos fornece informações mais precisas exclusivamente para as estações que representam os arbitradores. O fluxo da rede de filas como um todo não se altera com a política interna de administração dos arbitradores.

i	$D_i(N)$	$U_i(N)$	$N_i(N)$	$W_i(N)$
<i>PL</i>	1,5496	0,8405	1,5496	1,0000
<i>A1</i>	0,7748	0,0751	0,0775	0,1001
<i>A2</i>	0,7748	0,0751	0,0775	0,1001
<i>A3</i>	0,7748	0,0751	0,0775	0,1001
<i>A4</i>	0,7748	0,0751	0,0775	0,1001
<i>M1</i>	0,3874	0,3874	0,5351	1,3812
<i>M2</i>	0,3874	0,3874	0,5351	1,3812
<i>M3</i>	0,3874	0,3874	0,5351	1,3812
<i>M4</i>	0,3874	0,3874	0,5351	1,3812

Tabela 5.3: Resultados para o Modelo com Colisão sobre Arbitradores em RFE

5.3 Limitações

O modelo refinado, apesar de bastante adequado, contém ainda duas imprecisões de menor importância. O método utilizado é responsável por uma destas imprecisões. Em contrapartida, a outra imprecisão deve-se a um problema específico da modelagem desenvolvida. Ambas imprecisões eventualmente podem inserir pequenos erros numéricos. Cabe lembrar, no entanto, que o propósito das análises desenvolvidas nesta dissertação não é a obtenção de resultados numéricos exatos.

Os valores adotados como média dos tempos de utilização das estações do modelo de filas são erroneamente considerados segundo uma distribuição exponencial. Esta limitação deve-se a uma das hipóteses das redes forma-produto. Na verdade, uma interconexão multi-estágios possui tempos fixos para os atrasos

dos arbitradores e tempos discretamente variáveis para o processamento local e global.

A outra limitação do modelo obtido deve-se à decisão de modelar de forma otimista o funcionamento dos arbitradores. Cada arbitrador pode encontrar-se em uma de quatro situações descritas na figura 5.4. Em duas destas situações (*a* e *b*), dois acessos podem ser atendidos. No entanto, nas duas outras situações (*c* e *d*) somente um dos acessos pode ser atendido.

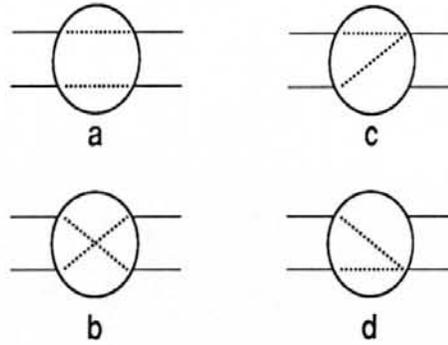


Figura 5.4: Funcionamento de Arbitradores

Desta forma, o modelo desenvolvido traz imprecisões por ser otimista no que se refere à modelagem do funcionamento dos arbitradores. Esta imprecisão tem sua importância diminuída na mesma razão que diminui o tempo de processamento global.

6 MODELAGEM ATRAVÉS DE RdPt

Este capítulo apresenta o desenvolvimento, através do método **RdPt**, do modelo para interconexões multi-estágios do tipo Omega. Qualquer conjunto de valores coerentes atribuídos aos parâmetros citados na análise quantitativa (seção 4.3) é aplicável a este modelo. No entanto, visando facilitar a compreensão, um multi-estágios com características idênticas ao exemplo do capítulo anterior será utilizado neste capítulo:

- número de processadores: 4
- número de módulos de memória: 4
- número de estágios: 2
- cardinalidade dos arbitradores (*fan-in* e *fan-out*): 2
- atraso dos arbitradores: 0,1 u.t.
- tempo médio de processamento local: 1 u.t.
- tempo médio de processamento global: 1 u.t.
- probabilidade de acesso dos processadores aos módulos de memória: 25 % (equiprovável)

Um modelo analisável diretamente pelo *modelo Q1* é apresentado na primeira parte deste capítulo. Este modelo bastante simples apresenta o princípio básico de modelagem utilizado nos modelos **RdPt**. A segunda parte apresenta incrementalmente o modelo refinado. São expostas as duas sofisticções do modelo fundamental adotadas para proporcionar um maior ajuste do modelo à realidade. Para ambos os modelos resultados são exibidos e críticas às suas limitações são feitas.

6.1 Modelo Fundamental

A preocupação básica do modelo fundamental é descrever a tarefa básica executada pelas interconexões multi-estágios. Compõe ainda a definição do modelo fundamental a especificação dos recursos utilizados e sua ordem de utilização. Desta forma, apresenta-se a cadeia principal e os recursos do modelo, ou seja, um modelo do tipo *Q1* completo. Após, são fornecidos os resultados referentes ao modelo fundamental e apresentam-se suas limitações.

6.1.1 Tarefa Básica

A classe multi-estágios, como as demais classes do grupo multiprocessador, tem o acesso a um módulo de memória como tarefa básica. O desempenho destas interconexões é expresso através do número de acessos que são suportados por unidade de tempo. Desta forma, a tarefa a ser modelada pela cadeia principal do modelo em **RdPt** é um acesso ao módulo de memória.

Denomina-se processamento global o acesso a um módulo de memória. Um acesso a um módulo de memória é feito quando existem dois pré-requisitos fundamentais (óbvios): necessidade de acesso e possibilidade de acesso.

A necessidade de acesso se manifesta quando um processador tem interesse em acessar uma memória de uso compartilhado. Para que isto ocorra é necessário que o processador já tenha executado o seu processamento local. Portanto, é possível afirmar que um dos pré-requisitos para o acesso a um módulo de memória é o processamento local dos processadores.

A possibilidade de acesso se dá quando existe um caminho entre o processador que necessita o acesso e o módulo de memória que será acessado. Por-

tanto, o estabelecimento de uma conexão entre processador e memória também é um dos pré-requisitos para o acesso ao módulo de memória.

Define-se então três subtarefas que devem ser executadas para o fechamento do ciclo que envolve o acesso aos módulos de memória:

- o processamento local (PL)
- a conexão (Cx)
- o processamento global (PG)

A figura 6.1 apresenta um modelo em rede de Petri das subtarefas definidas. Note-se que não é feita menção a eventuais tempos de espera no modelo da figura 6.1. Não é possível definir os pontos onde podem ocorrer tais esperas sem a especificação dos recursos e seus pontos de alocação e liberação. Pois, somente os recursos (na verdade sua ausência) causam tais tempos de espera na execução da tarefa básica.

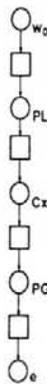


Figura 6.1: Modelo da Tarefa Principal em RdPt

Os tempos de utilização devem ser associados a cada um dos lugares que representam as subtarefas (PL , Cx e PG). Desta forma, teremos para o exemplo assumido as seguintes informações associadas ao modelo em redes de Petri temporizadas:

$$Z(PL) = 1u.t. \quad Z(Cx) = 0.2u.t. \quad Z(PG) = 1u.t.$$

Note-se que o tempo de utilização de conexão ($Z(Cx)$) envolve o uso de dois arbitradores, portanto deve considerar o dobro do atraso do arbitrador.

6.1.2 Recursos

As entidades físicas presentes numa interconexão multi-estágios são os processadores, os arbitradores e os módulos de memória. No entanto, os recursos envolvidos na modelagem proposta não precisam ser necessariamente representações diretas destas entidades.

Sob o ponto de vista da tarefa principal, os recursos necessários são:

- para o processamento local:
 - um processador
- para o estabelecimento da conexão:
 - um processador
 - um caminho através dos arbitradores¹
- para o processamento global:
 - um processador
 - um caminho através dos arbitradores
 - um módulo de memória

Desta forma, os três tipos de recursos são identificados. A figura 6.2 apresenta o modelo fundamental completo com a tarefa básica e os recursos necessários a sua execução. Neste modelo além da cadeia principal estão presentes os lugares que representam a disponibilidade dos recursos e os pontos onde estes

¹Entenda-se por *caminho através dos arbitradores* um dos possíveis pares de arbitradores, onde cada um destes pertence a um estágio.

são alocados e liberados. Note-se que foram inseridos na cadeia principal lugares para eventuais esperas sempre que recursos são alocados. Estes lugares aumentam a cadeia principal sem, no entanto, alterar o seu tempo mínimo dispendido (somatório dos tempos de utilização).

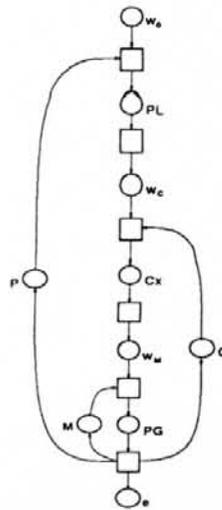


Figura 6.2: Modelo Fundamental em RdPt

A marcação associada a cada lugar representando a disponibilidade de recursos é relativa ao número de processadores (P), número de módulos de memórias (M) e número de caminhos simultâneos possíveis (C). Note-se que os caminhos são uma abstração feita a partir das entidades arbitradores. Cabe salientar que é puramente casual o fato de quatro arbitradores possibilitarem quatro caminhos. Por exemplo, se estivéssemos desenvolvendo um modelo para um multi-estágios com oito processadores e oito módulos de memória, teríamos doze arbitradores e apenas oito caminhos simultâneos possíveis.

6.1.3 Resultados e Limitações

Os resultados da tabela 6.1 são obtidos a partir do modelo fundamental.

*Estes resultados são uma primeira aproximação dos resultados exatos. Note-se que os recursos críticos são os processadores. Esta conclusão é tirada da utilização máxima dos recursos processador ($U(P) = 1$). Por consequência, o número de

caminhos e de memórias é suficiente para a demanda dos processadores e os tempos induzidos de espera são nulos.

$$\begin{array}{l|l|l} D(N) = 1,8182 & Z_w(w_C) = 0 & Z_w(w_M) = 0 \\ U(PL) = 1 & U(Cr) = 0,5455 & U(PG) = 0,4545 \end{array}$$

Tabela 6.1: Resultados para o Modelo Fundamental em RdPt

No entanto, o modelo fundamental não considera a possibilidade de conflitos nem por memórias, nem por caminhos. Na possibilidade de disputa de processadores por memórias ou por caminhos é possível haver uma queda de desempenho e utilização não máxima dos recursos. Quando dois ou mais processadores desejarem utilizar um mesmo módulo de memória ou uma mesma parte de um caminho (uma mesma saída de um arbitrador) existirão um ou mais processadores não atendidos. Esta situação onde as preferências de acesso influenciam o desempenho não é modelada. O modelo fundamental, por desconsiderar esta possibilidade de colisões, fornece resultados otimistas em relação ao funcionamento das interconexões.

O modelo Q1 prevê somente tempos de utilização fixos. Logo, os tempos de processamento local e global também podem estar sendo modelados de forma não satisfatória. Em geral, tais tempos são discretamente variáveis e somente uma média fixa destes tempos pode ser considerada no modelo fundamental. Desta forma, a modelagem dos tempos de processamento é uma das limitações mais evidentes deste modelo.

6.2 Modelo Refinado

O modelo refinado tem por função básica solucionar as limitações do modelo fundamental. Os métodos aproximativos vistos no capítulo três são utilizados para este propósito. Os métodos aproximativos aumentam o custo da obtenção de resultados, pois a técnica utilizada baseia-se no desdobramento

do modelo fundamental em diversas redes com estrutura idêntica e tempos de utilização e marcações iniciais distintas. Logo, o cálculo de diversas redes $Q1$ deve ser feito para a obtenção dos índices de desempenho.

São fornecidos resultados numéricos que ilustram o aumento de precisão decorrente das sofisticações propostas. O modelo refinado, apesar de solucionar as limitações mais elementares do modelo fundamental, possui ainda algumas limitações. Estas limitações são discutidas ao final deste capítulo.

6.2.1 Colisões sobre Caminhos e Memórias

A qualquer momento do funcionamento da interconexão multi-estágios uma disputa entre processadores pode ocorrer. Esta disputa acontece quando dois processadores necessitam utilizar simultaneamente o mesmo trecho de caminho, ou seja, uma mesma saída de um arbitrador. Este fenômeno pode ser visto como uma disponibilidade variável do recurso caminho. No modelo refinado esta característica será implementada através da adoção de uma marcação variável no lugar C . A figura 6.3 apresenta a idéia do desdobramento proposto.

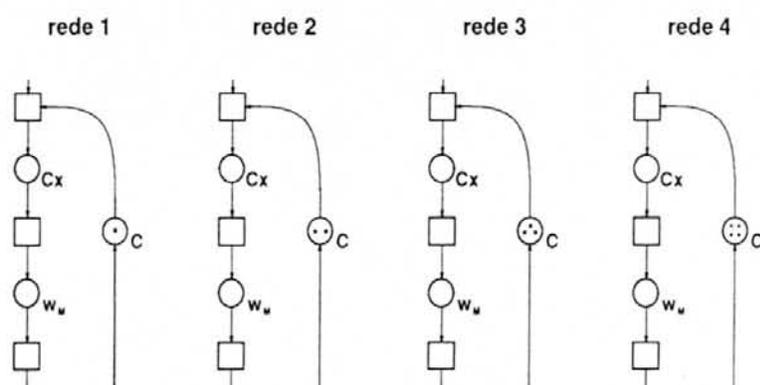


Figura 6.3: Modelo de Colisão de Caminhos

A ponderação de cada uma das redes com marcações distintas é feita de acordo com as probabilidades de colisão decorrentes da matriz de preferências de acesso dos processadores aos módulos de memória. O cálculo envolvido para

determinar os pesos de cada situação de colisão envolvem a classificação de cada combinação particular de processadores e memórias.

A tabela 6.2 fornece os resultados para o exemplo proposto com colisões. Os resultados obtidos demonstram que uma modelagem mais realista foi utilizada, pois existe uma coerência com o efeito das colisões. Nenhum recurso possui utilização máxima, logo mesmo o recurso crítico (recurso com maior utilização) possui uma carga ociosa.

$$\begin{array}{l|l|l} D(N) = 1,7188 & Z_w(w_C) = 0,1272 & Z_w(w_M) = 0 \\ U(PL) = 0,9454 & U(Cx) = 0,5157 & U(PG) = 0,4297 \end{array}$$

Tabela 6.2: Resultados para o Modelo com Colisões em RdPt

Resolvido o problema de modelagem das disputas de caminhos automaticamente as disputas de módulos de memória estão resolvidas. Visto que todo módulo de memória está conectado somente a uma saída de um arbitrador do último estágio, todas as disputas de memória ocorrem antes que o processador tenha chance de acessá-la. Equivale dizer que todo conflito por módulos de memória não ocorre diretamente sobre as memórias, mas sobre o último estágio dos caminhos. Em suma, a modelagem da contenção de caminhos é suficiente para considerar tanto disputas sobre caminhos, quanto disputas sobre módulos de memória. Numericamente este fato se verifica pelo tempo induzido de espera nulo antes da alocação dos módulos de memória ($Z_w(w_M) = 0$).

6.2.2 Tempos de Utilização Variáveis

Considerando que os tempos de processamento local e global não são fixos, conclui-se que a modelagem destes tempos é inadequada no modelo fundamental. A natureza destes tempos pede uma modelagem mais sofisticada, onde estes são considerados discretamente variáveis (valores discretos). Na verdade

trata-se de tempos que assumem um número finito de valores distintos segundo uma probabilidade de ocorrência.

O método aproximativo utilizado para modelar este fenômeno consiste em atribuir pesos a diversas redes com tempos de utilização determinísticos e assim calcular um novo valor para o índice de desempenho. Logo, o esforço de modelagem consiste em determinar que valores devem ser assumidos para cada tempo de processamento (local e global).

O tempo de processamento local é o caso mais simples e pode ser resumido em dois valores. Estes valores simbolizam as duas possibilidades básicas do processamento local: execução de operações locais e inatividade (*idle*). Para cada uma destas possibilidades, uma probabilidade de ocorrência é fixada. A análise pode, então, ser feita. Geralmente, o tempo de execução de operações locais varia segundo o tipo de aplicação dado à interconexão. Por outro lado, o tempo de inatividade costuma ser menos variável, pois envolve somente operações de consulta a outros processadores através de acessos a áreas pré-estabelecidas nos módulos de memória.

O tempo de processamento global, por sua vez, deve ser dividido em diversas faixas segundo a necessidade de precisão dos resultados. Cabe lembrar que o método aproximativo empregado desdobra a rede do modelo fundamental em diversas redes $Q1$ segundo o número de tempos discretos considerados. Em virtude disto, quanto maior for o número de faixas consideradas, maior será o custo em tempo de extração para a obtenção de resultados.

6.2.3 Limitações

A principal limitação da modelagem proposta é a nível de descrição do processo de conexão. A modelagem desenvolvida, tanto no modelo fundamental, quanto no refinado, pressupõe conexões monolíticas. Desta forma, os

processadores que buscam uma conexão tem somente dois tipos de resultados: ou a conseguem ou não. Não são modeladas situações onde um processador consegue um caminho através de diversos estágios e mesmo assim não efetiva a conexão. Na prática, esta imperfeição de modelagem não traz maiores imprecisões nos resultados, pois quando uma conexão é negada o processador deve aguardar até que todo o caminho através dos diversos estágios esteja disponível.

Cabe salientar que a modelagem da disputa de arbitradores passo a passo, em contraposição à modelagem proposta, também traz limitações. Basicamente, o modelo de disputa passo a passo deveria especificar quantos caminhos seriam possíveis para cada grupo de arbitradores de um mesmo estágio. Tal número é dependente do caminho escolhido por cada processador e ainda variável para cada estado. Desta forma, o aumento da complexidade do modelo para distinguir a disputa passo a passo seria bastante grande e seus resultados não seriam seguramente mais satisfatórios.

7 ESTUDO COMPARATIVO

Neste capítulo é feita a comparação entre os dois modelos apresentados nos capítulos anteriores. Esta comparação se dá em três níveis: facilidade de modelagem, velocidade de extração e comparação numérica. As apresentações destes níveis compõem as partes deste capítulo, respectivamente. Toda comparação feita refere-se ao paralelo traçado entre os dois modelos analíticos desenvolvidos. No entanto, as considerações expressas neste capítulo aplicam-se a qualquer modelo desenvolvido através dos métodos **RFE** e **RdPt**. Desta forma, o propósito deste capítulo é comparar os dois métodos utilizados.

7.1 Facilidade de Modelagem

A facilidade de modelagem é, antes de tudo, função da desenvoltura do modelador no método utilizado. A experiência no uso do formalismo de modelagem dá mais liberdade para a definição do modelo. No entanto, podem ser tecidas algumas considerações inerentes à adequação do método em si ao tipo de problema. Basicamente, dois aspectos desta adequação podem ser identificados: adequação das primitivas de modelagem e das técnicas de construção do modelo.

7.1.1 Primitivas de Modelagem

O objetivo básico desta seção é uma análise subjetiva da adaptação das primitivas de modelagem de ambos os modelos (**RFE** e **RdPt**) às situações existentes nas interconexões multi-estágios. Algumas destas situações não são modeláveis por limitações das primitivas dos métodos. O interesse desta comparação é muito mais analisar as soluções encontradas e os seus custos do que determinar

quais são as limitações encontradas. Em suma, um método não precisa ter todas as primitivas necessárias ao problema. A facilidade de desenvolvimento, bem como a qualidade do modelo, no entanto, são consideravelmente maiores quando o método possuir as primitivas necessárias para representar todos os aspectos do problema considerado.

As interconexões multi-estágios, um problema típico de alocação simultânea de recursos, devem ser modeladas com primitivas de sincronismo. A representação explícita do sincronismo contribui tanto para a clareza do modelo, quanto para a sua fidelidade à realidade.

Os tempos de processamento local e global, por sua vez, podem ser variáveis. Desta forma, trata-se de um problema com tempos de utilização variáveis. Os métodos devem, portanto, ter primitivas para representar o caráter aleatório dos tempos de utilização de recursos.

O método **RFE** possui primitivas para a representação de tempos aleatórios, ainda que estes só possam estar distribuídos exponencialmente (segundo a forma-produto). Por outro lado, a idéia de sincronismo não existe como um conceito primário. O recurso básico utiliza os demais recursos sem que haja uma definição explícita do sincronismo. Existe um sincronismo implícito sempre que uma unidade de recurso básico (um cliente) estiver sendo atendido em uma estação. Este sincronismo será entre os recursos representados pelo cliente e pelo servidor da estação. Uma representação mais explícita da alocação simultânea de recursos poderia ser incluída através do uso de algum método aproximativo [FDI 86 e DAL 89]. No entanto, tais métodos dificultariam bastante a extração, pois eles utilizam uma abordagem distinta, valendo-se de operações de ordem superior de complexidade (cálculo integral).

O método **RdPt**, devido a sua base em redes de Petri, suporta facilmente a descrição de processos com vários pontos de sincronismo. As transições são os pontos naturais para a representação de sincronismo. Por outro lado,

a representação do comportamento aleatório só é possível através do uso de métodos aproximativos. A principal desvantagem destes métodos é o alto custo em tempo de processamento para a obtenção de resultados. Para certos casos, os resultados obtidos podem ser desvantajosos, pois eles não são obtidos a partir de um método verdadeiramente aleatório. Na verdade, eles são obtidos pela ponderação estatística dos resultados determinísticos. Felizmente, esta característica dos métodos aproximativos corresponde ao funcionamento real das interconexões multi-estágios.

7.1.2 Técnicas de Construção de Modelos

O objetivo desta seção é tecer considerações sobre a adequação das abstrações usuais dos métodos em relação ao comportamento genérico das interconexões multi-estágios. Sob o ponto de vista funcional, estas interconexões podem ser vistas segundo diversas abordagens. Este fato facilita a adequação de ambos os métodos à realidade proposta (multi-estágios).

A construção do modelo através do método RFE é bastante natural. O modelo de clientes pode ser facilmente associado aos processadores da interconexão. Este fato é reforçado pela hierarquia natural dos processadores sobre os demais componentes que normalmente integram um circuito digital [SIE82].

Analogamente, o modelo abstrato de tarefa principal também é bastante natural. Principalmente devido ao ciclo de funcionamento processamento local, conexão e processamento global. Desta forma, a construção do modelo em RdPt também é intuitiva. Somente a idéia de infinitas requisições pode não ser tão imediata. Este pequeno problema é atenuado pela adoção de uma representação hierarquicamente indistinta dos processadores, memórias e caminhos. Esta representação define as requisições de uso de módulos de memória como necessidade básica do modelo.

7.2 Velocidade de Extração

O objetivo deste nível de comparação é verificar a agilidade de aplicação da extração de resultados a ambos os modelos. Especificamente, a velocidade de extração é estabelecida em função do tamanho do problema (número de processadores da interconexão) e a precisão desejada. Cabe salientar que a comparação se refere à extração aplicada aos modelos de multi-estágios desenvolvidos. No entanto, também os métodos utilizados estão sendo indiretamente comparados.

7.2.1 Modelos RFE

Todo modelo desenvolvido segundo o método RFE tem o tempo de extração de resultados expresso como uma função linear do número de clientes e do número de estações. Ambas informações podem ser diretamente concluídas do tamanho do problema (N - número de processadores). Para os modelos desenvolvidos no capítulo cinco teremos:

- Modelo fundamental:

3 estações (PL , Cx e PG)

N clientes

- Modelo refinado com colisão sobre memórias:

$2 + N$ estações (PL , Cx e N estações M)

N clientes

- Modelo refinado com colisão sobre memórias e arbitradores:

$1 + (\frac{N}{2} \log_2 N) + N$ estações (PL , $\frac{N}{2} \log_2 N$ estações A e N estações M)

N clientes

Considerando as informações acima, conclui-se que o modelo refinado com colisão sobre memórias é o mais recomendado. O custo de extração deste modelo é mais baixo que aquele do segundo modelo refinado (colisão sobre arbitradores) e os seus resultados são de igual qualidade, exceto no que se refere a valores específicos dos arbitradores. O modelo fundamental, apesar do seu baixo custo, apresenta resultados demasiadamente simplificados. Por fim, o modelo refinado com colisão sobre memórias e arbitradores só acrescenta precisão nos resultados referentes ao desempenho dos arbitradores trazendo um aumento considerável do custo.

7.2.2 Modelos RdPt

Os modelos desenvolvidos através do método **RdPt** têm o seu custo de tempo expresso em função de duas grandezas: o número de redes a ser calculado e o tamanho da matriz necessária ao cálculo dos pesos de cada rede (matriz de preferências). O número de redes é função do número de processadores e da quantidade de tempos discretos que serão utilizados na modelagem dos tempos de processamento local e global.

A matriz de preferências estabelece qual a probabilidade de cada processador acessar cada módulo de memória. Portanto, o tamanho desta matriz é função do número de processadores (N). O cálculo dos pesos a partir da matriz de preferências implica na obtenção de um número de valores exponencialmente proporcional à ordem da matriz. O método tradicional para este cálculo implica na obtenção de N^N valores obtidos através da multiplicação de N fatores ($N - 1$ multiplicações). Métodos alternativos [KIT89] podem ser utilizados em matrizes particulares para reduzir este custo excessivo.

Não obstante, por tratar-se do cálculo de pesos baseados em probabilidades de acesso (valores entre zero e um) um problema adicional limita o

cálculo de matrizes muito grandes. Valores muito pequenos devem ser processados e uma grande precisão computacional é necessária à máquina encarregada da obtenção de tais valores. Note-se que este problema é semelhante ao encontrado em simulações muito longas onde valores muito pequenos devem ser somados a valores muito grandes para o cálculo de índices de desempenho.

A experiência prática da implementação do cálculo da matriz de probabilidades de acesso em computador da linha IBM-PC tipo AT 286 fornece um ponto de referência sobre os tempos envolvidos. Utilizando o método tradicional do cálculo de probabilidades são necessárias cinco horas de processamento para uma matriz de ordem oito (oito processadores e oito módulos de memória). Através do método alternativo para matrizes equiprováveis [KIT89] é possível o cálculo quase imediato de matrizes de ordem quarenta. Neste caso porém, a limitação é causada pelo limite de precisão da máquina e da linguagem utilizada proporcionando o fenômeno da acumulação catastrófica de erros no cálculo.

Seja N o número de processadores. Sejam L e G os números de valores discretos utilizados para a modelagem dos tempos de processamento local e global, respectivamente. Para cada um dos modelos desenvolvidos no capítulo seis teremos:

- Modelo fundamental:
 - 1 rede (sem matriz de preferências)
- Modelo refinado com colisão de caminhos:
 - N redes
 - matriz de preferências de ordem N
- Modelo refinado com tempos variáveis:
 - $L * G$ redes (sem matriz de preferências)
- Modelo refinado com ambas sofisticções:

$N * L * G$ redes

matriz de preferências de ordem N

Com base nas informações acima é possível recomendar o uso do modelo fundamental para obtenção de valores iniciais. A extração através do modelo fundamental é praticamente imediata e seus resultados fornecem uma primeira aproximação razoável. De acordo com a necessidade particular de cada aplicação, os modelos refinados devem ser utilizados. Este é por exemplo o caso de avaliações onde o efeito das colisões é muito importante e o modelo refinado com colisões sobre caminhos deve ser utilizado.

7.2.3 Orientação

Cada um dos modelos desenvolvidos tem uma aplicação mais apropriada. Desta forma, é possível identificar os casos práticos onde as características (vantagens e defeitos) podem melhor adequar-se à aplicação.

Para a obtenção de primeiras informações sobre uma interconexão multi-estágios deve-se utilizar o modelo fundamental em **RdPt**. Este modelo apresenta uma melhor relação custo benefício, pois seu cálculo é praticamente imediato e seus resultados são melhores que aqueles do modelo elementar em **RFE**. No entanto quando uma maior precisão é necessária este método não fornece resultados suficientemente acurados devido às suas limitações de modelagem.

Resultados mais acurados só podem ser obtidos através do uso de um dos modelos refinados. Nestes casos os modelos do método **RdPt** possuem melhores resultados. A sua aplicação a problemas grandes é, no entanto, fortemente prejudicada. Resta assim utilizar os modelos em **RFE** que apesar de menos precisos possuem uma maior velocidade. Cabe lembrar que o cálculo dos resultados nos métodos em **RFE** é linearmente proporcional ao tamanho do problema, en-

quanto nos métodos aproximativos em **RdPt** uma relação exponencial pode ser encontrada.

Em suma, o método **RdPt** possui uma velocidade maior para níveis mais baixos de precisão de resultados (modelo fundamental). Com custos intermediários, uma melhor precisão é obtida através dos modelos refinados em **RFE**. Finalmente, os modelos refinados em **RdPt** apresentam a maior acuracidade tendo uma velocidade de extração bastante baixa para problemas grandes. Note-se que o compromisso com o tamanho do problema é tanto função do número de processadores, quanto dos recursos computacionais disponíveis para sua solução.

7.3 Comparação Numérica

O objetivo da comparação numérica é assegurar a coerência dos modelos desenvolvidos. No entanto, resultados exatos não são o objetivo dos modelos desenvolvidos. Busca-se somente provar que, para o caso adotado (multi-estágios), os resultados fornecidos são suficientemente corretos. Desta forma, desenvolve-se um trabalho de validação dos modelos. Note-se, porém, que não é o objetivo desta dissertação assegurar a validade dos métodos, mas apenas compará-los para um caso particular. Para uma validação dos modelos desenvolvidos seria necessário uma análise mais profunda dos resultados. Esta análise deveria incluir um maior número de comparações numéricas com outros métodos e não apenas com os resultados de um único simulador.

7.3.1 Paradigma de Comparação

A comparação numérica necessita de resultados reais. Neste sentido optou-se pelo uso de um simulador de interconexões de processadores (SSIP

[CAV89]). Este simulador é genérico para interconexões do grupo multiprocessador. Os parâmetros utilizados pelo simulador são mais completos do que aqueles utilizados pelos métodos analíticos. Os parâmetros utilizados pelo SSIP são os seguintes:

- número de processadores
- número de módulos de memória
- número de arbitradores
- tipo da malha de interconexão
- atrasos dos arbitradores
- tempo de processamento local
- tempo de processamento global
- matriz de preferências
- tipo de geração de tempos de processamento
- política de escalonamento

As informações utilizadas pelo SSIP são idênticas àquelas dos modelos analíticos, exceto a malha de interconexão, o tipo de geração de tempos de processamento e a política de escalonamento. Na verdade, o tipo de malha define a classe de interconexão multiprocessador, no caso multi-estágios.

O tipo de geração de tempos de processamento define a distribuição aleatória dos tempos de utilização. As opções fornecidas são distribuição uniforme, exponencial e constante. Desta forma, os tempos de processamento local e global deverão ser informados através de intervalos de geração (uniforme), médias (exponencial) e valores exatos (constante).

A política de escalonamento define qual o tratamento de prioridade é empregado quando da existência de colisões. Este parâmetro só tem função no SSIP porque a simulação desenvolvida é individualizada para cada processador. O efeito das diferentes políticas de escalonamento se manifesta somente nos parâmetros individuais de cada processador. Na análise proposta o interesse é nos processadores como um conjunto de recursos homogêneos.

A simulação desenvolvida pelo SSIP é orientada a eventos. Os resultados são gerados segundo um mecanismo interno de crítica que verifica a estabilização dos resultados. Basicamente, o simulador permanece em funcionamento até que resultados gerados sejam supostamente confiáveis, ou seja, estabilizáveis e estatisticamente significativos. Cabe salientar que a confiabilidade dos resultados é suposta, pois não é possível provar quando um simulador está fornecendo resultados cem por cento confiáveis [BRA87].

7.3.2 Estudo de um Caso Prático

Seja uma interconexão de processadores com as seguintes características:

- número de processadores: 8
- número de módulos de memória: 8
- número de estágios: 3
- atraso dos arbitradores: 0,1 u.t.
- tempo médio de processamento local: 0 a 4 u.t.
- tempo médio de processamento global: 2 u.t.
- probabilidade de acesso dos processadores aos módulos de memória: 25 % (equiprovável)

Para esta realidade obtemos os resultados definidos na tabela 7.1. Os parâmetros de interesse são o número de acessos à memória global por unidade de tempo e o índice de utilização dos recursos processador e módulos de memória. Os resultados são apresentados para cada uma das possibilidades discretas de tempo de processamento local. Os modelos utilizados são os modelos mais acurados de cada método. Utiliza-se o modelo refinado com colisão de caminhos e tempos variáveis em **RdPt** e o modelo refinado com colisão de memórias e arbitradores em **RFE**. O simulador é configurado para gerar valores com tempos de processamento gerados segundo uma distribuição exponencial.

proc. local	Acessos por u.t.			Utilização de Processadores			Utilização de Memórias		
	simulação	RFE	RdPt	simulação	RFE	RdPt	simulação	RFE	RdPt
0 u.t.	1,850982	2,0476	1,9452	0,480880	0,5887	0,5074	0,463896	0,5119	0,5012
1 u.t.	1,643	1,7681	1,7504	0,632481	0,7294	0,7307	0,390793	0,442	0,4296
2 u.t.	1,387732	1,5205	1,4480	0,736309	0,8173	0,7983	0,339808	0,3801	0,3537
3 u.t.	1,225045	1,3159	1,2803	0,842358	0,8718	0,8611	0,272593	0,329	0,2821
4 u.t.	1,211626	1,1510	1,1306	0,890877	0,9064	0,8990	0,246605	0,2878	0,2552

Tabela 7.1: Resultados para o Exemplo Prático

Os dados obtidos confirmam as expectativas. Os modelos apresentam resultados da mesma ordem de grandeza, apesar de possuírem diferenças relevantes a nível de modelagem. Os resultados do modelo **RFE** mostram-se otimistas em praticamente todos os casos analisados. Este fato era previsível, pois os modelos em **RFE** não conseguem modelar exatamente as disputas pelos arbitradores. Acredita-se, então, que este aspecto (colisão sobre arbitradores) é um dos mais relevantes ao desempenho da realidade estudada.

Os resultados de simulação demonstram as hipóteses de modelagem de ambos os modelos aproximam-se bastante do comportamento real das interconexões multi-estágios. Não obstante, os resultados de simulação mantêm-se abaixo dos valores gerados pelos métodos analíticos. Este fenômeno justifica-se pela melhor descrição da realidade feita no simulador. Detalhes como solicitações a módulos de memória que são bloqueados no último estágio da malha de arbitradores só são considerados corretamente pelo modelo de simulação.

A figura 7.1 apresenta graficamente os resultados fornecidos anteriormente sob a forma de tabelas. A visualização das curvas reforça a validação dos modelos, pois o comportamento relativo da interconexão é semelhante tanto para o modelo em RFE, quanto para o modelo em RdPt. A inclinação das curvas demonstra este fato. Igualmente o uso do teste Kolmogorov-Smirnov [BRA87] atesta que todas as três curvas são da mesma ordem, pois seus resultados são exponenciais.

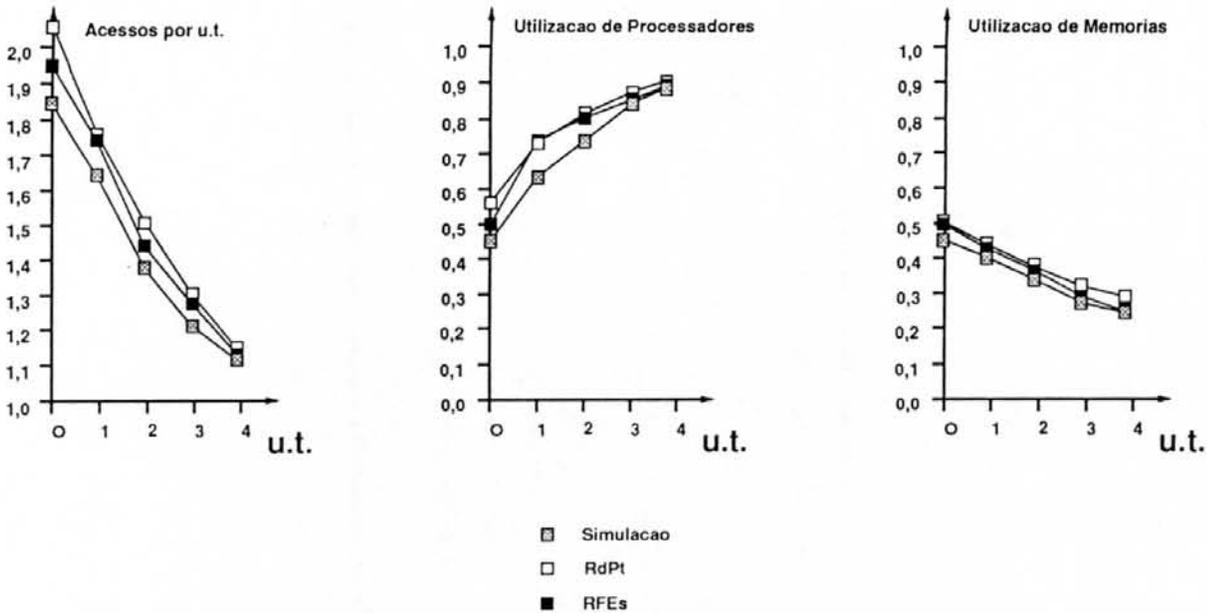


Figura 7.1: Resultados Numéricos

A única exceção numérica aos resultados otimistas do modelo em RFE se dá no índice de utilização dos recursos processador para o tempo de processamento local igual a um. Os resultados do modelo em RdPt apresentam uma característica a parte dos demais. Observando as curvas deste modelo, percebemos que existe um ponto de inflexão exatamente onde o tempo de processamento local é igual a um. Esta inflexão não é verificada nas demais curvas (simulação e RFE), logo, é possível que uma mudança do recurso crítico (gargalo) esteja sendo super estimada. Estas afirmações não podem, todavia, ser verificadas sem um maior estudo dos resultados gerados pelo modelo.

8 CONCLUSÃO

As primeiras utilizações práticas do *modelo Q* [TAZ87] mostraram a necessidade de métodos aproximativos para a obtenção de resultados mais precisos. Esta necessidade foi a principal razão do desenvolvimento do método **RdPt**. Após a utilização prática deste método, optou-se pela comparação do método **RdPt** com o método analítico tradicional (**RFE**) para a sua validação. Desta forma, o objetivo inicial desta dissertação era a divulgação e defesa do método **RdPt**.

No decorrer dos trabalhos verificou-se, porém, que o método **RFE** possui certas características bastante distintas do método **RdPt**. Tais características impossibilitaram uma comparação absoluta da qualidade de ambos os métodos, mesmo para um só tipo de problema. Na verdade, suas primitivas de modelagem são tão diferentes entre si que não é possível representar modelos exatamente iguais segundo os dois métodos. Basta para tanto citar a diferença de enfoque na modelagem dos tempos de utilização variáveis.

Baseado em tal conclusão, definiu-se como objetivo genérico desta dissertação tecer um estudo comparativo entre os métodos **RFE** e **RdPt**. Este estudo se propõe a auxiliar a escolha e utilização destes métodos segundo as características individuais do problema a avaliar. Nos problemas com alocação simultânea de recursos exemplificou-se o uso de ambos os métodos, particularmente nas interconexões multi-estágios. Note-se que o desenvolvimento de tais exemplos é vital para a comparação proposta. Isto porque não se trata de definir o melhor dos métodos, mas proporcionar ao leitor as informações necessárias à formação do seu julgamento pessoal, sua escolha.

8.1 Diferenciação dos Métodos

No sentido de fornecer as informações necessárias ao conhecimento dos métodos **RFE** e **RdPt**, apresentam-se as diferenças filosóficas e teóricas entre os métodos. Discute-se em um primeiro momento a abordagem de cada método, ou seja, a forma pela qual os componentes do sistema são identificados. Após identifica-se o embasamento teórico existente e as características contingenciais de cada método.

O princípio de descrição adotado pelo método **RFE** é o modelo de clientes. Desta forma, o modelo é visto como um conjunto de caminhos por onde os clientes se deslocam. Durante este "passeio" dos clientes através do modelo, diversas atividades são disparadas segundo a presença dos clientes em determinados pontos do modelo (estações de serviço). Esta abordagem é fortemente hierarquizada, pois atribui-se aos clientes toda atenção e controle do funcionamento. Todo atraso é realizado sobre os clientes e os seus efeitos também são sobre eles manifestados. A abordagem adotada é bastante útil, pois diversas realidades a modelar são efetivamente hierarquizadas. Este é por exemplo o caso das interconexões de processadores onde estes dominam o funcionamento dos demais componentes.

O método **RdPt**, ao contrário, adota um princípio de descrição distinto, baseado no modelo de solicitação de tarefas. Um modelo em **RdPt** tem descrita uma tarefa básica que é executada infinitamente, tão rápido quanto os recursos disponíveis puderem suportar. Os recursos são vistos de forma não hierárquica, sendo alocados e liberados em algum ponto da tarefa básica. Esta abordagem tem um caráter cooperativo, através do qual os recursos são utilizados. A atenção do processo de funcionamento é voltada para a tarefa básica. Os recursos em si são vistos de forma descentralizada. A abordagem adotada é bastante flexível, pois qualquer sistema pode ter identificada uma tarefa básica na qual todos os recursos são necessários à execução. Soma-se a esta possibilidade os sistemas onde um

ciclo de funcionamento pode ser identificado, pois este ciclo pode ser visto como uma tarefa composta de diversas partes (subtarefa).

Sob o ponto de vista teórico o método **RFE** é completamente embasado na teoria de filas e cadeias de Markov. Exatamente por esta sua origem histórica, ele é considerado um método clássico. Se por um lado esta característica lhe confere credibilidade, por outro lado o seu uso é mistificado como excessivamente complexo. Na prática, apesar de clássico, o método **RFE** é pouco utilizado. Mesmo entre os pesquisadores da área de informática, o número de usuário deste método é reduzido. Este fato deve-se em parte a preferência natural dos informatas pela construção de simuladores no lugar da manipulação de fórmulas matemáticas. Obviamente para informatas com uma formação matemática mais sólida esta tendência tende a atenuar-se.

Os conhecimentos necessários para a avaliação de redes forma-produto, no entanto, são de fácil compreensão e implementação trivial em linguagens de programação procedurais. Cabe salientar, porém, que o uso de métodos aproximativos (fora da forma-produto) para o método **RFE** não é simples [FDI 86 e DAL 89]. Justifica-se assim a utilização de outros métodos nos casos onde as redes forma-produto não são satisfatórias.

Em contrapartida, o método **RdPt** é bem recente, Por isto não carrega consigo preconceitos sobre o seu uso. Exceto o *modelo Q* que só fornece resultados determinísticos, o método **RdPt** não possui o mesmo nível de embasamento teórico que o método **RFE**. No entanto, esta sua falha é compensada pela sua facilidade de utilização. Os resultados do *modelo Q* podem ser calculados mentalmente por simples operações de divisão. Esta característica torna-o de fácil compreensão, cuja utilização inicial é facilmente adotada. Os métodos aproximativos, por outro lado, pecam pelo alto custo normalmente associado à adaptação dos parâmetros de entrada. Este, por exemplo, é o caso do cálculo dos pesos a partir da matriz de preferências do método refinado com colisão de caminhos apresentado no capítulo seis.

8.2 Trabalhos Práticos Associados

Esta dissertação é parte integrante do projeto ADMP Avaliação de Desempenho de Máquinas Paralelas [ADM90]. Este projeto constitui-se em um grupo de pesquisa cujo objetivo comum é o estudo de métodos para avaliação de desempenho de sistemas. Especificamente, o grupo concentra seus estudos na avaliação de arquiteturas paralelas. Esta avaliação se dá em diversos níveis: interconexão de processadores, organização interna de nodos de processamento, etc [NAV88a]. Para estes níveis são desenvolvidos modelos de avaliação, ferramentas são implementadas e casos particulares são analisados. Pesquisas diretamente relacionadas com estes ramos principais de atividades são igualmente desenvolvidas, por exemplo: o estudo de métodos de avaliação de desempenho e o desenvolvimento de interfaces gráficas de alto conforto (user-friendly) para as ferramentas implementadas. Porém o projeto abrange também áreas correlatas como a avaliação de desempenho de redes de computadores e ambientes industriais automatizados, pela semelhança da solução de tais problemas com as atividades básicas deste projeto.

As ferramentas atualmente em desenvolvimento no projeto ADMP são os trabalhos práticos associados a esta dissertação. Estas ferramentas são orientadas à avaliação de interconexões de processadores e de modelos genéricos segundo os métodos **RFE** e **RdPt**. As técnicas utilizadas pelas ferramentas são simulação e aplicação de métodos analíticos (**RFE** e **RdPt**). Diretamente relacionados com esta dissertação existem as seguintes ferramentas, atualmente em desenvolvimento: SARA - Sistema de Análise de Redes e Arquiteturas, SSIP - Sistema Simulador de Interconexões de Processadores, SQ1 - Sistema Analisador de Modelos Q1, SARF - Sistema de Análise de Redes de Filas.

SARA, Sistema de Análise de Redes e Arquiteturas [NAV88], é uma ferramenta dedicada à análise quantitativa de interconexões de processadores, sejam estas utilizadas em redes de computadores ou em arquiteturas paralelas.

O usuário pode definir interconexões através de informações como a topologia, número de processadores, tempo de processamento, etc. A ferramenta oferece uma análise interativa de diversos aspectos relevantes ao desempenho de interconexões com saídas gráficas e numéricas. Os modelos implementados foram desenvolvidos através do método **RdPt**. Em particular, o modelo apresentado no capítulo seis foi integrado à ferramenta SARA.

SSIP, Sistema Simulador de Interconexões de Processadores [CAV89], é uma ferramenta dedicada à simulação orientada a eventos de interconexões de processadores, sendo estas da classe multiprocessador (barramento único, múltiplos barramentos, crossbar e multi-estágios). A definição e visualização de resultados das interconexões na ferramenta SSIP é análoga àquela utilizada na ferramenta SARA. Os resultados de simulação utilizados na capítulo sete foram obtidos através da ferramenta SSIP.

SQ1, Sistema Analisador de Modelos Q1, é uma ferramenta para auxílio na definição de modelos baseados no método **RdPt**. Desta forma o usuário poderá definir e obter resultados relativos à análise de diversas redes de Petri temporizadas através do *modelo Q1*. A ferramenta possui uma interface de alto conforto segundo os padrões das demais ferramentas desenvolvidas no projeto. Os resultados dos modelos serão exibidos de forma gráfica ou numérica. Todos os resultados referentes ao método **RdPt** apresentados nesta dissertação foram obtidos através da ferramenta SQ1.

SARF, Sistema de Análise de Redes de Filas, é uma ferramenta para auxílio na definição de modelos através do método **RFE**. O usuário pode definir e obter resultados relativos à análise de diversos modelos em redes de filas fechadas. A interface desta ferramenta segue os mesmos padrões das demais desenvolvidas no projeto ADMP, tanto a nível de edição quanto exibição de resultados. Todos os resultados referentes a modelos em **RFE** apresentados nesta dissertação foram calculados pela ferramenta SARF.

8.3 Trabalhos Futuros

Os trabalhos desenvolvidos nesta dissertação não esgotam o esforço de caracterização dos métodos **RFE** e **RdPt**. Somente um caso particular dos problemas com alocação simultânea de recursos foi desenvolvido. Neste exemplo, as interconexões multi-estágios, a existência de um tipo de recurso hierarquicamente superior aos demais facilitou o processo de modelagem em ambos os métodos. Desta forma, o estudo de outros problemas desta mesma classe com características distintas é uma complementação natural ao trabalho desenvolvido nesta dissertação.

Analogamente, o desenvolvimento de um estudo comparativo semelhante para classes distintas de problemas também é uma continuação viável para o estudo iniciado. Especificamente, a comparação dos métodos **RFE** e **RdPt** quando aplicados a problemas com alocação serial de recursos seria interessante. Tais problemas tem necessidades distintas, logo a flexibilidade dos métodos seria evidenciada. Um exemplo prático, como linhas de manufatura, pode ser adotado para ilustrar esta classe de problemas. Uma vantagem adicional desta escolha é a análise de uma realidade onde ocorrem outros tipos de situações aleatórias, por exemplo pane de estações, etc.

Aprofundando no campo de avaliação de desempenho, outro tipo de trabalho poderia dar sequência à pesquisa desta dissertação. Trata-se da proposta de novos métodos aproximativos, tanto ao método **RdPt**, que já possui alguns destes métodos, quanto ao **RFE**, que necessita de um maior poder de descrição. Este trabalho de cunho mais inovador é uma decorrência natural do estudo aprofundado de diversos métodos analíticos.

Finalmente, um objetivo maior seria o estabelecimento de taxonomias para a classificação de métodos analíticos e classes de problemas. Tais tarefas, no entanto, demandariam o levantamento de praticamente todos os estudos relevan-

tes da área de avaliação de desempenho, após o qual um trabalho classificatório deve ser feito. Apesar da dificuldade de execução, este trabalho seria de grande importância. As taxonomias por ele obtidas facilitariam enormemente qualquer pesquisa da avaliação de desempenho de sistemas.

ANEXO A-1 ALGORITMO PARA CÁLCULO DA CONSTANTE DE NORMALIZAÇÃO

Este anexo apresenta uma implementação na linguagem PASCAL do cálculo da constante de normalização para redes de fila de espera fechadas, mono-classe, com estações dependentes ou independentes da carga. A presente implementação deste algoritmo, chamado algoritmo de Buzen ou algoritmo de convolução, foi proposta por Bruell e Balbo [BRU80]. Maiores detalhes podem ser encontrados nesta bibliografia.

```
{ Segmento de programa para calcular a constante de normalizacao
  para uma rede com M estacoes e N clientes.
  Assume-se como entrada:

  STATION[I] = 'LI' para estacoes independentes da carga
              = 'LD' para estacoes dependentes da carga
  S[I]       = taxa media de servico independente da
              carga para a estacao I
  K[I,J]     = taxa de servico da estacao I quando
              a carga e' de J clientes
  V[I]       = taxa de visita para a estacao I

  e assume-se como saida:

  G[N] = constante de normalizacao para a rede com N clientes }

var { variaveis temporarias utilizadas }
    NI,          { ponteiro para numero de clientes }
    MI,          { ponteiro para estacoes }
    L: integer;  { indice auxiliar ao calculo }
    ACUM: real;  { acumulador da constante de normalizacao }
    TEMP: array [1..N] of real;
                { valor temporario para a caracterizacao das estacoes }
```

```

begin
  G[0] := 1.0;
  if STATION[1] = 'LI'   { estacoes independentes da carga }
  then begin
    Y := V[1] * S[1];
    for NI := 1 to N
      do G[NI] := G[NI-1] * Y;
    end
  else begin             { estacoes dependentes da carga }
    Y := V[1];
    for NI := 1 to N
      do G[NI] := G[NI-1] * Y / K[1,NI];
    end;
  for MI := 2 to M
  do if STATION[MI] = 'LI'
    then begin
      Y := V[MI] * S[MI];
      for NI := 1 to N
        do G[NI] := G[NI] + Y * G[NI - 1];
      end
    else begin
      TEMP[0] := 1.0;
      Y := V[MI];
      for NI := 1 to N
        do TEMP[NI] := TEMP[NI - 1] * Y / K[MI,NI];
      for NI := N downto 1
        do begin
          ACUM := G[NI];
          for L := 1 to NI
            do ACUM := ACUM + TEMP[L] * G[NI - L];
          G[NI] := ACUM;
        end
      end
    end
  end;
end;

```

BIBLIOGRAFIA

- [ADM90] **ADMP - Avaliação de Desempenho de Máquinas Paralelas.** Porto Alegre: CPGCC da UFRGS, 1990. Relatório de Projeto.
- [AJM84] AJMONE MARSAN, M.; BALBO, G.; CONTE, G. A class of generalized stochastic Petri nets for performance evaluation of multiprocessor systems. **ACM Transactions on Computer Systems**, New York, v.2, n.2, p.93-122, May 1984.
- [ALL78] ALLEN, A. O. **Probability, statistics, and queueing theory; With computer science applications.** London: Academic Press, 1978.
- [AND75] ANDERSON, G. A.; JENSEN, E. D. Computer interconnection structures: taxonomy, characteristics, and examples. **Computing Surveys**, New York, v.7, n.4, p.197-213, Dec. 1975.
- [BRA83] BRAHMS, G. W. **Réseaux de Petri: théorie et pratique.** Paris: Gauthier-Villars, 1983. 2v.
- [BRA87] BRATLEY, P.; FOX, B.; SCHRAGEL, L. **A guide to simulation.** New York: Springer-Verlag, 1987.
- [BRU80] BRUEL, S.; BALBO, G. **Algorithms for closed queueing networks.** Cambridge: The MIT Press, 1980.
- [BUZ73] BUZEN, J. Computational algorithms for closed queueing networks with exponential servers. **Communications of ACM**, New York, v.16, n.9, p.527-531, Sept. 1973.
- [CAV89] CAVALHEIRO, G.; KITAJIMA, J.; NAVAU, P.; TAZZA, M. SSIP - Sistema simulador de interconexões de processadores. In: CONGRESSO NACIONAL DE MATEMÁTICA APLICADA E COMPUTACIONAL, 12., 4-8 set. 1989, São José do Rio Preto. **Anais...** São José do Rio Preto: SBMAC, 1989. p.207-210.

- [DAL89] DALLERY, Y. **Approximate analysis of general open queueing networks with restricted capacity.** Grenoble: Laboratoire d'Automatique de Grenoble, 1989. (Rapport de Recherche)
- [FDI86] FDIDA, S.; PUJOLE, G.; MAILLES, D. Réseaux de files d'attente avec sémaphores. **Technique et Science Informatique - TSI**, Paris, mars 1986.
- ✓ [FER78] FERRARI, D. **Computer systems performance evaluation.** Englewood Cliffs; Prentice-Hall, 1978. 554p.
- ✓ [FER88] FERNANDES, P.; TAZZA, M.; NAVAUX, P. Análise quantitativa de interconexões estrela. In: CONGRESSO DA SOCIEDADE BRASILEIRA DE COMPUTAÇÃO, 8., jul. 17-22, 1988, Rio de Janeiro. **Anais...** Rio de Janeiro: SBC, 1988. p.125-142
- ✓ [FER88a] FERNANDES, P. **Análise quantitativa de interconexões regulares.** Porto Alegre: CPGCC da UFRGS, 1988. 49p. (Trabalho individual, 57)
- ✓ [FER89] FERNANDES, P. **Evaluation de performances de systèmes: une étude comparative.** Grenoble: Laboratoire d'Automatique de Grenoble, 1989. 46p. (Note interne, 89/57)
- ✓ [GEL80] GELENBE, E. & MITRANI, I. **Analysis and synthesis of computer systems.** London: Academic Press, 1980.
- [KIT89] KITAJIMA, J.; NAVAUX, P.; TAZZA, M. Um problema combinatorial sobre matrizes. In: CONGRESSO NACIONAL DE MATEMÁTICA APLICADA E COMPUTACIONAL, 12., 4-8 set. 1989, São José dos Campos. **Anais...** São José do Rio Preto: SBMAC, 1989. p.241-244.
- [KLE75] KLEINROCK, L. **Queueing systems.** New York: Wiley, 1975. 2v.

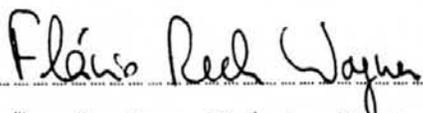
- [LAZ84] LAZOWSKA, E.; ZAHORJAN, J.; GRAHAM, G.; SEVCIK, K. **Quantitative system performance: computer system analysis using queueing networks models**. Englewood Cliffs: Prentice-Hall, 1984. 417p.
- [HWA84] HWANG, K.; BRIGGS, F. A. **Computer architecture and parallel processing**. New York: McGraw-Hill, 1984. 846p.
- [NAV88] NAVAUX, P.; FERNANDES, P.; TAZZA, M. SARA: a processor interconnection performance analysis tool. In: EUROMICRO SYMPOSIUM ON MICROPROCESSING AND MICROPROGRAMMING, 14., Aug. 29-Sept. 1, 1988, EUROMICRO'88, Zurich. **Proceedings...** Amsterdam: North-Holland, 1988. p.197-204.
- ✓ [NAV88a] NAVAUX, P.; FERNANDES, P.; TAZZA, M. **Avaliação de desempenho de máquinas paralelas**. In: SIMPÓSIO BRASILEIRO DE ARQUITETURA DE COMPUTADORES - PROCESSAMENTO PARALELO, 1., set. 1988, Águas de Lindóia. **Anais...** Águas de Lindóia: SBC, 1988. v.2. p.1111-1117.
- [PET77] PETRI, C. A. **General net theory**. In: PROCEEDINGS OF THE JOINT IBM UNIVERSITY OF NEWCASTLE UPON TYNE SEMINAR, Sept. 7-10, 1976, Newcastle upon Tyne. **Proceedings...** New Castle upon Tyne: University of Newcastle upon Tyne, 1977. p.131-169.
- [PET81] PETERSON, J. L. **Petri net theory and the modelling of systems**. Englewood Cliffs: Prentice-Hall, 1981. 485p.
- [REI85] REISIG, W. **Petri nets: an introduction**. Berlin: Springer-Verlag, 1985. 161p.
- [RIC85] RICHTER, G. Clocks and their use for time modelling. In: **Information systems: Theoretical and formal aspects**. Amsterdam: North-Holland, 1985. p.49-66.

- [SIE82] SIEWIOREK, D.; BELL, C.; NEWELL, A. **Computer structures: principles and examples**. London: McGraw-Hill, 1982.
- [SIF80] SIFAKIS, J. Use of Petri nets for performance evaluation. In: INTERNATIONAL SYMPOSIUM ON MEASURING, MODELLING AND EVALUATION OF COMPUTER SYSTEMS, 3., Oct. 3-5, 1977, Bonn-Bad Godesberg. **Proceedings...** Amsterdam: North-Holland, 1980. p.75-94.
- [TAZ85] TAZZA, M. **Ein netztheoretisches modell zur quantitativen analyse von systemen (Q-modell)**. München: R. Oldenbourg Verlag, 1985. 134p. (Bericht, 149)
- [TAZ85a] TAZZA, M. **Performance evaluation using a net theory based model**. Porto Alegre: CPGCC da UFRGS, 1985. 41p. (Relatório, 23)
- [TAZ87] TAZZA, M.; FERNANDES, P.; NAVAU, Ph. Análise quantitativa de interconexões cross-bar. In: CONGRESSO DA SOCIEDADE BRASILEIRA DE COMPUTAÇÃO, 7., jul. 11-17, 1987, Salvador. **Anais...** Salvador: SBC, 1987. p.241-254.
- ↳ [TAZ88] TAZZA, M. **Análise quantitativa de sistemas**. Curitiba: EBAI. 1988. 169p. (III Escola Brasileiro-Argentina de Informática).

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
CURSO DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

Avaliação de desempenho de interconexão de processadores-
-modelos analíticos para alocação simultânea de recursos.

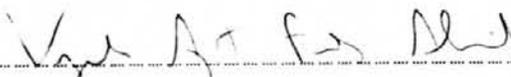
Dissertação apresentada aos Srs.



Prof. Dr. Flávio Rech Wagner



Prof. Dr. Claudio Walter



Prof. Dr. Virgílio Almeida



Prof. Dr. Maurizio Tazza

Visto e permitida a impressão

Porto Alegre, 11.../02..../.93...



Prof. Dr. Philippe O. A. Navaux
Orientador



Prof. Dr. Ricardo Augusto da L. Reis
Coordenador do Curso de Pós-Graduação
em Ciência da Computação