

UNIVERSIDADE DE SÃO PAULO  
INSTITUTO DE FÍSICA DE SÃO CARLOS

Proposta de Rede Óptica de Interconexão para  
Arquiteturas a Fluxo de Dados

*João Angelo Martini*

OK

USP/IFSC/SBI



8-2-001309

Tese apresentada ao Instituto de Física de  
São Carlos, da Universidade de São Paulo,  
para obtenção do título de **Doutor** em  
Ciências: Física Aplicada - opção:  
Física Computacional

Orientador: Prof. Dr. Álvaro Garcia Neto

DEPARTAMENTO DE FÍSICA E INFORMÁTICA

SÃO CARLOS - 1999

IFSC-USP SERVIÇO DE BIBLIOTECA  
INFORMAÇÃO

Martini, João Angelo

Proposta de Rede Óptica de Interconexão para Arquiteturas a Fluxo de Dados/João Angelo Martini. - São Carlos, 1999.  
132p.

Tese (Doutorado) - Instituto de Física de São Carlos, 1999.

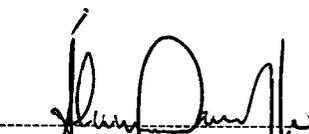
Orientador: Prof. Dr. Álvaro Garcia Neto

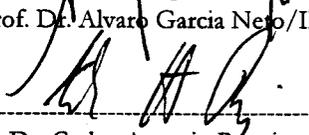
1. Arquiteturas Paralelas não Convencionais. 2. Redes de Interconexão. I. Título.

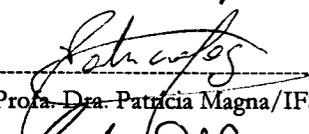


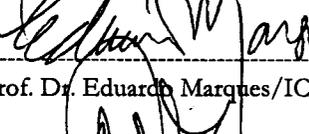
MEMBROS DA COMISSÃO JULGADORA DA TESE DE DOUTORADO DE JOÃO ANGELO MARTINI APRESENTADA AO INSTITUTO DE FÍSICA DE SÃO CARLOS, DA UNIVERSIDADE DE SÃO PAULO, EM 07 DE DEZEMBRO DE 1999.

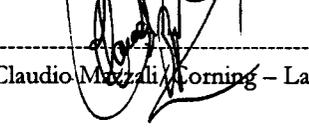
COMISSÃO JULGADORA:

  
-----  
Prof. Dr. Alvaro Garcia Neto/IFSC/USP

  
-----  
Prof. Dr. Carlos Antonio Ruggiero/IFSC/USP

  
-----  
Profa. Dra. Patricia Magna/IFSC/USP

  
-----  
Prof. Dr. Eduardo Marques/ICMC/USP

  
-----  
Dr. Claudio Mezzi/Corning - Latina América

## **Dedicatória**

À minha esposa Vera e à minha filha Mariane pelo amor, compreensão e paciência. À minha mãe pelo amor, incentivo e oportunidade.

## Agradecimentos

Ao Prof. Dr. Álvaro Garcia Neto, cuja orientação e amizade muito contribuíram para a elaboração deste trabalho e para minha formação profissional.

Aos integrantes do grupo Dataflow do FFI.

Ao professor João Eduardo pelas importantes discussões a respeito de sistemas WDM.

Ao professor Marcos Cavenaghi pelo apoio e pelas discussões sobre arquiteturas a fluxo de dados.

À professora Patrícia Magna pelo incentivo e amizade.

À Cláudia Tofaneli pela amizade e pelo competente trabalho de secretaria.

À Wladerez A. G. Caiado pela atenção e amizade.

Ao amigo Osvaldo Luis Asato pelo incentivo e apoio.

Aos colegas do Departamento de Informática da UEM.

A Deus pela vida.

Ao Reverendo Ricardo A. C. Makino e aos irmãos da Igreja Presbiteriana Calvário  
pelo amor, pelos ensinamentos e pelo apoio.

Em tudo somos atribulados, porém não angustiados;  
perplexos, porém não desanimados;  
perseguidos, porém não desamparados;  
abatidos, porém não destruídos.

2 Coríntios 4: 8-9

# Sumário

<b>Lista de Figuras</b> .....	<b>vii</b>
<b>Lista de Tabelas</b> .....	<b>ix</b>
<b>Resumo</b> .....	<b>.xi</b>
<b>Abstract</b> .....	<b>.xii</b>
<b>Introdução</b> .....	<b>.xiii</b>
<b>Capítulo 1</b>	
<b>Arquitetura a Fluxo de Dados e Redes de Interconexão</b> .....	<b>1</b>
<b>1.1 Introdução</b> .....	<b>1</b>
<b>1.2 Modelo de Fluxo de Dados</b> .....	<b>2</b>
<b>1.3 Contexto das Arquiteturas a Fluxo de Dados</b> .....	<b>4</b>
<b>1.4 Formalismo de Interconexão</b> .....	<b>9</b>
1.4.1 Redes .....	10
1.4.2 Topologia .....	11
1.4.3 Elementos de Chaveamento .....	13
1.4.4 Características Operacionais .....	14
1.4.5 Exemplos de Redes de Interconexão .....	15
<b>1.5 Tecnologia WDM (Wavelength Division Multiplexing)</b> .....	<b>18</b>
1.5.1 Arquitetura WDM .....	20
1.5.2 Taxonomia de Mukherjee .....	22
<b>1.6 Conclusões</b> .....	<b>23</b>
<b>Capítulo 2</b>	
<b>Propostas de redes de interconexão para a arquitetura APO</b>	
<b>2.1 Introdução</b> .....	<b>24</b>
<b>2.2 Arquitetura APO</b> .....	<b>24</b>
2.2.1 Descrição Funcional .....	25
2.2.2 Simulador da Arquitetura APO .....	27
<b>2.3 Requisitos de Comunicação</b> .....	<b>27</b>
2.3.1 Destinos para E/S .....	28
2.3.2 Destinos para CF .....	28
2.3.3 Destinos para ME .....	28
2.3.4 Destinos para as MDs .....	29
2.3.5 Destinos para EPs .....	29
<b>2.4 Rede de Interconexão</b> .....	<b>30</b>
2.4.1 Protocolo de Controle baseado na técnica TDM .....	31
2.4.1.1 Descrição dos Nós para o Protocolo de Controle TDM .....	33
2.4.2 Protocolo de Controle com Nó Central .....	37
2.4.2.1 Descrição dos Nós para o Protocolo com Controlador Central .....	39
<b>2.6 Conclusões</b> .....	<b>42</b>

6.4.1 Emprego dos modelos de redes de interconexão da arquitetura APO em outras arquiteturas paralelas .....	114
6.4.2 Incorporação dos modelos de redes de interconexão da arquitetura APO na arquitetura SEED .....	114
6.4.3 Investigação do comportamento dos índices de atividade das unidades da arquitetura APO .....	114
6.4.4 Investigação da distribuição da unidade CF nas entradas das MDs na arquitetura APO .....	115
6.4.5 Incorporar a metodologia de distribuição de carga do modelo TDM no modelo com Nó Central .....	115
6.4.6 Investigação do balanceamento de velocidade da Chave para o modelo com Nó Central .....	115
<b>6.5 Contribuições deste Trabalho .....</b>	<b>116</b>

#### **Anexo A**

<b>Comportamento da Chave: Distribuição de Carga para o Protocolo de Controle TDM .....</b>	<b>118</b>
<b>A.1 Introdução .....</b>	<b>118</b>
<b>A.2 Distribuição de carga dos processadores para a arquitetura APO com controle TDM .....</b>	<b>118</b>

#### **Anexo B**

<b>Comportamento da Chave: Distribuição de Carga para o Protocolo de controle com Nó Central .....</b>	<b>122</b>
<b>B.1 Introdução .....</b>	<b>122</b>
<b>B.2 Distribuição de carga dos processadores para o protocolo de controle com Nó Central .....</b>	<b>122</b>

<b>Bibliografia .....</b>	<b>126</b>
---------------------------	------------

## Lista de Figuras

FIGURA 1.1 - Grafo para $a = (b + 1) * (b - c)$ .....	2
FIGURA 1.2 - Arquitetura Wolf .....	5
FIGURA 1.3 - Redes de Interconexão em Sistemas Paralelos .....	10
FIGURA 1.4 - Rede de Interconexão Ideal .....	12
FIGURA 1.5 - Barramento Compartilhado .....	16
FIGURA 1.6 - Rede Crossbar .....	16
FIGURA 1.7 - Controle do Elemento de Chaveamento .....	17
FIGURA 1.8 - Rede de Interconexão multi-estágio Delta .....	19
FIGURA 1.9 - Rede de Interconexão baseada em WDM .....	22
FIGURA 2.1 - Arquitetura APO .....	25
FIGURA 2.2 - Destinos para as fichas do nó E/S .....	28
FIGURA 2.3 - Destino para as fichas do nó CF .....	28
FIGURA 2.4 - Destinos para as fichas do nó ME .....	29
FIGURA 2.5 - Destinos para as fichas dos nós MDs .....	29
FIGURA 2.6 - Destinos para as fichas dos nós EPs .....	29
FIGURA 2.7 - Ciclos de Controle e Transmissão para o Protocolo de Controle TDM .....	32
FIGURA 2.8 - Tempo de controle .....	33
FIGURA 2.9 - Estrutura do nó E/S para controle TDM .....	34
FIGURA 2.10 - Estrutura do nó CF para controle TDM .....	35
FIGURA 2.11 - Estrutura do nó ME para controle TDM .....	36
FIGURA 2.12 - Estrutura dos nós MDs para controle TDM .....	36
FIGURA 2.13 - Estrutura dos nós EPs para controle TDM .....	37
FIGURA 2.14 - Rede com Controlador Central .....	38
FIGURA 2.15 - Ciclos de Controle e Transmissão para o Protocolo de Controle com Controlador Central .....	39
FIGURA 2.16 - Estrutura do nó E/S para controle central .....	40
FIGURA 2.17 - Estrutura do nó CF para controle central .....	40
FIGURA 2.18 - Estrutura do nó ME para controle central .....	41
FIGURA 2.19 - Estrutura dos nós MDs para controle central .....	42
FIGURA 2.20 - Estrutura dos nós EPs para controle central .....	42
FIGURA 3.1 - Distribuição de Carga da arquitetura Wolf para Gauss .....	49
FIGURA 3.2 - Distribuição de Carga da arquitetura APO com o controle TDM para todas as chaves para Gauss .....	50

FIGURA 3.3 - Distribuição de Carga da arquitetura APO com o controle de Nó Central para todas as chaves para Gauss .....	51
FIGURA 4.1 - Tempo de execução para chave com controle TDM - Ackermann .....	70
FIGURA 4.2 - Tempo de execução para chave com controle TDM - Somatorial .....	72
FIGURA 4.3 - Tempo de execução para chave com controle TDM - Integração Binária ...	72
FIGURA 4.4 - Tempo de execução para chave com controle TDM - Gauss .....	73
FIGURA 4.5 - Tempo de execução para chave com controle TDM - Interpolação .....	74
FIGURA 4.6 - Tempo de execução para chave com controle TDM - Laplace .....	75
FIGURA 4.7 - Tempo de execução para chave com controle TDM - Multiplicação de Matrizes .....	76
FIGURA 4.8 - Tempo de execução para chave com controle TDM - N-Queens .....	76
FIGURA 4.9 - Tempo de execução para chave com controle TDM - Quick Sort .....	77
FIGURA 4.10 - Tempo de execução para chave com controle TDM - Simple .....	77
FIGURA 4.11 - Tempo de execução para chave com Nó Central - Ackermann .....	79
FIGURA 4.12 - Tempo de execução para chave com Nó Central - Somatorial .....	79
FIGURA 4.13 - Tempo de execução para chave com Nó Central - Integração Binária ....	80
FIGURA 4.14 - Tempo de execução para chave com Nó Central - Gauss .....	80
FIGURA 4.15 - Tempo de Execução para chave com Nó Central - Interpolação .....	81
FIGURA 4.16 - Tempo de Execução para chave com Nó Central - Laplace .....	82
FIGURA 4.17 - Tempo de Execução para chave com Nó Central - Multiplicação de Matrizes .....	82
FIGURA 4.18 - Tempo de Execução para chave com Nó Central - N-Queens .....	83
FIGURA 4.19 - Tempo de Execução para chave com Nó Central - Quick Sort .....	83
FIGURA 4.20 - Tempo de Execução para chave com Nó Central - Simple .....	84

## Lista de Tabelas

TABELA 1.1 - Arquiteturas a Fluxo de Dados .....	8
TABELA 1.2 - Características Ideais de uma Rede de Interconexão .....	13
TABELA 3.1 - Distribuição de carga para a arquitetura Wolf .....	52
TABELA 3.2 - Distribuição de carga para a arquitetura APO para TDM .....	52
TABELA 3.3 - Distribuição de carga para a arquitetura APO para Nó Central .....	52
TABELA 3.4 - Distribuição de Carga para Ackermann - TDM .....	54
TABELA 3.5 - Distribuição de Carga para Somatorial - TDM .....	55
TABELA 3.6 - Distribuição de Carga para Integração Binária - TDM .....	55
TABELA 3.7 - Distribuição de Carga para Interpolação - TDM .....	56
TABELA 3.8 - Distribuição de Carga para Laplace - TDM .....	57
TABELA 3.9 - Distribuição de Carga para Multiplicação de Matrizes - TDM .....	57
TABELA 3.10 - Distribuição de Carga para N-Queens - TDM .....	58
TABELA 3.11 - Distribuição de Carga para Quick Sort - TDM .....	59
TABELA 3.12 - Distribuição de Carga para Simple - TDM .....	59
TABELA 3.13 - Distribuição de Carga para Ackermann - Nó Central .....	60
TABELA 3.14 - Distribuição de Carga para Somatorial - Nó Central .....	61
TABELA 3.15 - Distribuição de Carga para Integração Binária - Nó Central .....	61
TABELA 3.16 - Distribuição de Carga para Interpolação - Nó Central .....	62
TABELA 3.17 - Distribuição de Carga para Laplace - Nó Central .....	63
TABELA 3.18 - Distribuição de Carga para Multiplicação de Matrizes - Nó Central .....	63
TABELA 3.19 - Distribuição de Carga para N-Queens - Nó Central .....	64
TABELA 3.20 - Distribuição de Carga para Quick Sort - Nó Central .....	64
TABELA 3.21 - Distribuição de Carga para Simple - Nó Central .....	65
TABELA 4.1 - Diferenças entre os tempos de execução dos protocolos de controle TDM e Nó Central .....	88
TABELA 5.1 - Índices de Atividade para Ackermann - TDM .....	91
TABELA 5.2 - Índices de Atividade para Somatorial - TDM .....	92
TABELA 5.3 - Índices de Atividade para Integração Binária - TDM .....	93
TABELA 5.4 - Índices de Atividade para Gauss - TDM .....	93
TABELA 5.5 - Índices de Atividade para Interpolação - TDM .....	94
TABELA 5.6 - Índices de Atividade para Laplace - TDM .....	94
TABELA 5.7 - Índices de Atividade para Multiplicação de Matrizes - TDM .....	95
TABELA 5.8 - Índices de Atividade para N-Queens - TDM .....	95

TABELA 5.9 - Índices de Atividade para Quick Sort - TDM .....	96
TABELA 5.10 - Índices de Atividade para Simple - TDM .....	96
TABELA 5.11 - Índices de Atividade para Ackermann - Nó Central .....	98
TABELA 5.12 - Índices de Atividade para Somatorial - Nó Central .....	99
TABELA 5.13 - Índices de Atividade para Integração Binária - Nó Central .....	99
TABELA 5.14 - Índices de Atividade para Gauss - Nó Central .....	99
TABELA 5.15 - Índices de Atividade para Interpolação - Nó Central .....	100
TABELA 5.16 - Índices de Atividade para Laplace - Nó Central .....	101
TABELA 5.17 - Índices de Atividade para Multiplicação de Matrizes - Nó Central .....	101
TABELA 5.18 - Índices de Atividade para N-Queens - Nó Central .....	102
TABELA 5.19 - Índices de Atividade para Quick Sort - Nó Central .....	102
TABELA 5.20 - Índices de Atividade para Simple - Nó Central .....	103
TABELA 5.21 - Índices de Atividade da Chave para controle TDM .....	105
TABELA 5.22 - Índices de Atividade da Chave para controle com Nó Central .....	106

## Resumo

Este trabalho apresenta dois modelos de redes de interconexão para a arquitetura a fluxo de dados APO (Arquitetura Paralela com rede de interconexão Óptica) e investiga o comportamento dessas redes para diferentes taxas de transmissão. Três parâmetros são considerados: distribuição de carga, tempo de execução e índice de atividade. A arquitetura APO é uma evolução da arquitetura Wolf proposta pelo Grupo de Pesquisas em Arquiteturas a Fluxo de Dados do Departamento de Física e Informática da Universidade de São Paulo.

A primeira rede de interconexão proposta utiliza o protocolo de controle TDM (*Time Division Multiplexing*). Ela emprega controle distribuído e evita colisões. A segunda rede de interconexão proposta utiliza um protocolo de controle centralizado e também evita colisões.

Para se investigar o comportamento dessas duas redes de interconexão, foi implementado um simulador dirigido a tempo desenvolvido em GNU C++ . Os resultados dessa investigação mostram que ambas as redes de interconexão propostas para a arquitetura APO proporcionam distribuições de carga muito mais homogêneas do que as atuais chaves da arquitetura Wolf. Mostram também que o protocolo com controle centralizado tem desempenho melhor que o protocolo TDM tanto em relação ao tempo de execução quanto ao índice de atividade.

## **Abstract**

This work presents two models for the APO architecture interconnection network and investigates the two networks behaviour for different transmission rates. Three parameters are considered: load distribution, execution time and active time. The APO architecture is an evolution of the Wolf architecture proposed by the Dataflow Architecture Research Group of the Physics and Informatics Department at University of São Paulo.

The first interconnection network proposed uses the TDM control protocol. It employs distributed control and avoids collisions. The second interconnection network proposed uses a centralized control protocol and also avoids collisions.

To investigate the behaviour of the two interconnection networks, a time driven simulator was implemented using GNU C++. The investigation results show that both interconnection networks proposed for the APO architecture provided better load distribution than the current Wolf interconnection networks. They also show that the centralized control protocol has better performance than the TDM protocol for execution time and active time.

## **Introdução**

Redes de interconexão constituem uma área importante no contexto de arquiteturas paralelas, pois viabilizam a comunicação entre os seus elementos. Arquiteturas a fluxo de dados necessitam de alta largura de banda para seus subsistemas de comunicação. Essas arquiteturas têm sido questionadas por exigirem uma capacidade de comunicação excessivamente grande.

Devido às limitações das redes de interconexão eletrônicas como barramento compartilhado e redes multi-estágio e ao desenvolvimento e amadurecimento da tecnologia de comunicação óptica, a investigação da aplicabilidade de comunicação óptica em arquiteturas a fluxo de dados é uma possibilidade promissora.

Há um amplo campo de investigação de sistemas de comunicação óptica para arquiteturas a fluxo de dados e a tecnologia WDM provê alta largura de banda, sendo então uma candidata natural para desempenhar o papel de sistema de comunicação para arquiteturas a fluxo de dados.

O objetivo deste trabalho consiste na proposta e simulação de redes de interconexão para a arquitetura a fluxo de dados APO (Arquitetura Paralela com rede de interconexão Óptica). Esta arquitetura é uma proposta evolutiva desenvolvida pelo Grupo de Pesquisas em Arquiteturas a Fluxo de Dados do Departamento de Física e Informática do Instituto de Física de São Carlos e é baseada na arquitetura Wolf.

A tese é estruturada em seis capítulos. O primeiro capítulo apresenta os conceitos fundamentais do paradigma de fluxo de dados bem como um resumo histórico desde os trabalhos pioneiros em arquiteturas baseadas no modelo de fluxo de dados. Apresenta também os principais conceitos da área de redes de interconexão que é o foco central do trabalho. Aborda também os aspectos fundamentais da tecnologia WDM, discutindo suas principais características e apresentando também a taxonomia desenvolvida por Mukherjee para sistemas WDM.

O segundo capítulo apresenta as características da arquitetura APO e discute os aspectos de comunicação da arquitetura. Apresenta e discute as duas propostas de redes de interconexão para a arquitetura.

Os três capítulos seguintes apresentam os resultados obtidos na investigação do comportamento das redes de interconexão da arquitetura. O terceiro capítulo apresenta os resultados das distribuições de carga para os dois modelos de chave de interconexão da arquitetura APO em função da variação de velocidade de transmissão das chaves de interconexão. Esses resultados são também analisados comparativamente aos obtidos pela arquitetura Wolf. O capítulo apresenta também os programas de teste empregados nessa investigação.

O quarto capítulo apresenta os resultados dos tempos de execução dos programas em função dos modelos e das velocidades das chaves de interconexão da arquitetura. Apresenta também uma proposta de classificação de desempenho do sistema.

O quinto capítulo apresenta os resultados dos índices de atividade das unidades da arquitetura para os dois modelos de chaves em função da variação de velocidade das chaves.

O sexto capítulo sintetiza as conclusões com base nos resultados e discussões apresentados nos capítulos anteriores. Aponta caminhos futuros, discutindo oportunidades de continuidade de investigações futuras para este trabalho.

# **Capítulo 1**

## **Arquitetura a Fluxo de Dados e Redes de Interconexão**

### **1.1 Introdução**

Este é um trabalho evolutivo que se insere dentro do espectro das linhas de investigação propostas pelo Grupo de Pesquisas em Arquiteturas a Fluxo de Dados do Departamento de Física e Informática da Universidade de São Paulo. Ele explora as possibilidades de convergência entre mecanismos de comunicação óptica e as arquiteturas a fluxo de dados.

Este capítulo apresenta os principais conceitos do paradigma de fluxo de dados: regra de disparo, modelos estático e dinâmico e mecanismo de rotulação de fichas. Apresenta também um resumo histórico desde os trabalhos pioneiros em arquiteturas baseadas no paradigma de fluxo de dados.

Para situar as redes de interconexão no contexto da pesquisa em arquiteturas paralelas, este capítulo também aborda as características fundamentais das redes de interconexão.

## 1.2 Modelo de Fluxo de Dados

Os princípios fundamentais do modelo de fluxo de dados foram propostos por Jack B. Dennis [1] na década de 70. O conceito fundamental do modelo de fluxo de dados baseia-se na chamada *regra de disparo*. Segundo essa regra, uma instrução é escalonada para execução quando todos os operandos de entrada necessários para sua execução ficarem disponíveis. Quando essa condição é satisfeita diz-se que a instrução foi disparada. Os dados de entrada são então consumidos e gera-se outros dados de saída.

A seqüência de execução das instruções está intrinsecamente relacionada à dependência dos dados. Um programa em fluxo de dados não é representado por uma seqüência de instruções com controle central que impõe a ordem de execução, como no modelo de von Neumann, mas é representado através de um grafo bidimensional. No grafo cada instrução é representada por um nó e os arcos que ligam os nós representam os caminhos que os dados percorrem até a instrução subsequente.

A figura 1.1 mostra um exemplo de grafo para o trecho de programa que calcula a expressão  $a = (b+1) * (b-c)$ . Nesse grafo percebe-se a possibilidade de execução concorrente dos nós de adição e subtração, pois não há dependência de dados nesse caso.

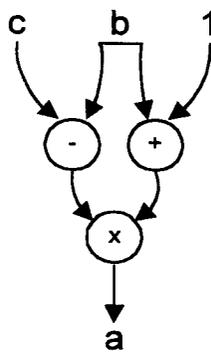


FIGURA 1.1 - Grafo para  $a = (b + 1) * (b - c)$

Os dados são transmitidos através de fichas<sup>1</sup> que percorrem os arcos entre os nós. Como os nós podem possuir mais de uma entrada, sua execução só é disparada após a chegada de todas as fichas nas correspondentes entradas. No exemplo da figura 1.1, o nó que executa a operação de subtração possui duas entradas e somente é executado a partir da chegada das fichas correspondentes aos dados **b** e **c** em suas entradas. Supondo que apenas a ficha do dado **b** esteja presente na sua respectiva entrada, então essa ficha deve esperar pelo seu par correspondente, a ficha relativa ao dado **c**, que é denominada ficha parceira ou ficha par. A partir da chegada da parceira, as fichas são emparelhadas e o nó pode ser executado.

Dependendo do modo como as fichas são tratadas, os computadores de fluxo de dados podem ser classificados em estáticos ou dinâmicos [2]. No modelo estático apenas uma ficha por vez pode ocupar determinado arco. Isso gera uma regra de disparo estática em que uma instrução é escalonada se uma ficha está presente em cada um de seus arcos de entrada e nenhuma ficha está presente em qualquer um de seus arcos de saída. Essa implementação usa sinais de sincronismo, que são fichas de controle usadas para indicar que os arcos de saída estão disponíveis.

No modelo dinâmico as fichas são rotuladas para permitir que múltiplas fichas apareçam simultaneamente em qualquer arco de entrada de um nó. A função do rótulo é identificar o contexto no qual a ficha foi gerada. Fichas pertencentes a diferentes instâncias, por exemplo, os diversos níveis de sucessivas chamadas recursivas, são distinguidas através desse mecanismo. Essa estratégia evita a necessidade de sinais de sincronismo, que são usados no modelo estático. Cada arco pode conter várias fichas com diferentes rótulos. A regra de disparo habilita um nó

---

<sup>1</sup> Em arquiteturas a fluxo de dados os dados e as informações de controle referentes a eles são representados por entidades denominadas fichas.

quando fichas com mesmo rótulo estão presentes em cada um dos seus arcos de entrada. Para que as fichas destinadas à mesma instrução sejam emparelhadas elas devem possuir o mesmo rótulo.

### **1.3 Contexto das Arquiteturas a Fluxo de Dados**

As arquiteturas a fluxo de dados suscitaram interesse desde os projetos pioneiros desenvolvidos na década de 1970. Muitos projetos experimentais foram investigados a partir de então e têm servido como base para subseqüentes computadores a fluxo de dados.

O interesse em relação às pesquisas na área de fluxo de dados tem tido altos e baixos, apresentando um comportamento cíclico de problemas e soluções em que novos experimentos e estudos trazem à tona problemas anteriormente não detectados e suscitam a busca de soluções através do desenvolvimento de novas pesquisas. O presente trabalho está no contexto de ciclo problema-solução.

O interesse pela área diminuiu quando se identificou que o subsistema de interconexão era tão ou mais complexo do que todo o restante da arquitetura. Conforme a velocidade dos elementos computacionais foi aumentando, cada vez mais o gargalo foi recaindo sobre a interconexão entre os elementos [3, 4]. Dentro desse ciclo problema-solução se inserem dois trabalhos de investigação na área de redes de interconexão. O primeiro desenvolvido por Martini [5] discute a influência das chaves de interconexão na arquitetura Wolf [6, 7, 8].

A arquitetura a fluxo de dados Wolf, apresentada na figura 1.2, é baseada no modelo de fluxo de dados dinâmico. O funcionamento da arquitetura é descrito resumidamente a seguir: as fichas são inseridas na arquitetura através da Chave de Coleta, esta chave transmite as fichas para a Memória de Dados, se esta unidade

estiver disponível ou então transmite para a Fila de Fichas para posterior distribuição. A Memória de Dados gera um pacote formado por um par de fichas dirigidas para um mesmo nó. Esse pacote é enviado para a Memória de Instruções Externa. Dependendo do tipo do pacote recebido por essa memória, ela pode gerar fichas destinadas a operações de tratamento de estruturas de dados ou transmitir o pacote inalterado para a Chave de Distribuição. A Chave de Distribuição pode transmitir o pacote para a Memória de Estruturas ou para uma das Memórias de Instruções Internas. No caso de pacotes destinados à Memória de Instruções Interna, esses pacotes são enviados para a primeira memória que estiver disponível. A partir do pacote recebido a Memória de Instruções Interna cria um pacote executável com todas as informações necessárias para a execução da instrução enviando-o à sua correspondente Unidade Funcional, a qual processa esse pacote executável, produzindo novos pacotes de dados. A Memória de Estruturas é responsável pelo armazenamento de dados estruturados.

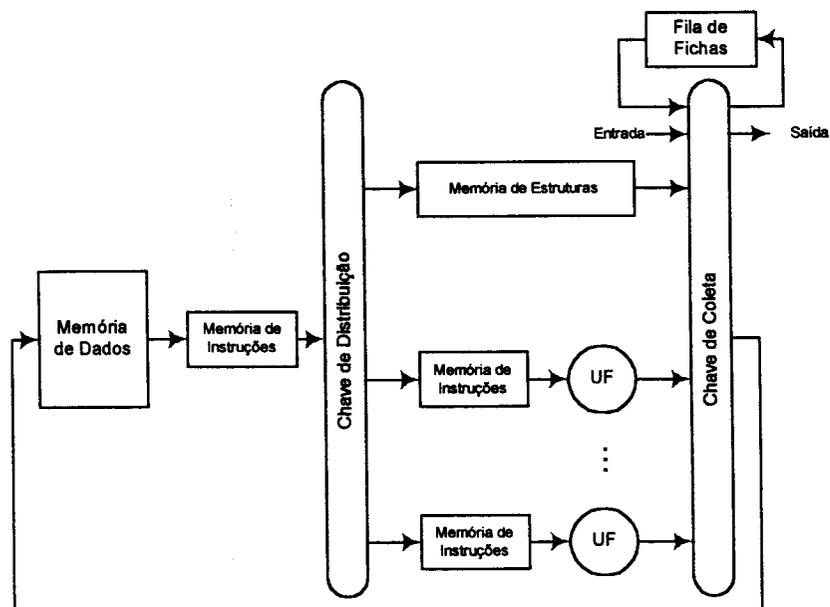


FIGURA 1.2 - Arquitetura Wolf

A arquitetura Wolf emprega duas chaves de interconexão (Chave de Coleta e Chave de Distribuição) para as quais foram implementadas diversas topologias. Para a Chave de Distribuição foram implementadas duas chaves denominadas Cross1 e Demux, para a Chave de Coleta foram implementadas quatro chaves denominadas: Cross2, Crossbar, GSN e Delta. Foram analisadas todas as combinações desses pares de chaves para os programas de teste<sup>2</sup>: Ackermann, Incremento, Integração Binária e Multiplicação de Matrizes. Esse trabalho mostrou que não há uma única combinação de topologias para as chaves da arquitetura que seja eficiente para todas as aplicações.

Por exemplo, os programas de teste utilizados na investigação da arquitetura Wolf [5,9], mostram o seguinte quadro:

- Para o programa Incremento a combinação de chaves denominada Cross1-Cross2 proporciona o menor tempo de execução, enquanto que, as combinações Demux-Delta e Demux-GSN apresentam os maiores tempos de execução, com uma diferença de aproximadamente 64% entre o menor e o maior tempo de execução.
- Para o programa Ackermann a diferença entre os tempos de execução para as combinações de chaves com o menor e o maior tempo de execução foi de 107,3%.
- O programa Integração Binária mostra que a pior combinação de chaves Demux-GSN para esse programa de teste, aumenta em até 229% o tempo de execução em relação à melhor combinação Demux-Cross2.

---

<sup>2</sup> Programas de teste são também denominados de *benchmarks*. São programas de comportamento conhecido que procuram explorar os diferentes recursos computacionais disponíveis numa máquina, permitindo exercícios de avaliação das características da mesma. Os programas utilizados neste trabalho são descritos na seção 3.2.

- Para o programa Multiplicação de Matrizes a diferença entre o menor e o maior tempo de execução foi de 34,8%.

Essa investigação mostra que as diversas combinações de chaves da arquitetura apresentam uma grande variação dos tempos de execução para cada programa de teste. Com diferenças de cerca de 35% entre a melhor e a pior combinação de chaves para um determinado programa de teste, e em outro caso com diferenças de cerca de 229% para outro programa de teste. Chega-se rapidamente à conclusão de que a implantação de chaves eletrônicas é inviável pois não existe uma combinação adequada para todos os casos. Esse é o principal resultado do trabalho de mestrado de Martini [5] e coloca a fotônica numa perspectiva de ser o fator que pode viabilizar a arquitetura a fluxo de dados.

O desenvolvimento de uma nova área tecnológica, chamada fotônica, propiciou a emergência de uma nova tecnologia denominada WDM (*Wavelength Division Multiplexing*) [10], que tem sido objeto de intensa pesquisa porque possui características como: alta largura de banda, baixa taxa de erros, baixo nível de perdas, transparência ao protocolo, independência entre os canais de transmissão e escalabilidade. Nesse contexto se insere o outro trabalho, desenvolvido por Martins [11] que mostra a viabilidade e o potencial de exploração e investigação dessa nova tecnologia na área de redes de interconexão para arquiteturas paralelas em geral.

O presente trabalho se insere num esforço continuado de 20 anos, resumido na tabela 1.1 [12].

TABELA 1.1 - Arquiteturas a Fluxo de Dados

NOME	PERÍODO	LOCAL	CARACTERÍSTICA
Static Dataflow Architecture [13]	1974-1983	MIT - EUA	Modelo estático. Foi construído um protótipo com 8 elementos de processamento.
LAU System [14]	1975-1980	CERT - França	Modelo estático. Máquina estava operacional em 1980.
DDP [15]	1976-1978	Texas Instruments - EUA	Modelo estático, estava operacional em 1978.
DDM1 [16]	1972-1976	Burroughs - EUA	Modelo estático, estava operacional em 1976.
NEC ImPP [17, 18]	1984	NEC Electronics - Japão	Primeiro chip microprocessador de fluxo de dados, o uPD7281. Operacional em 1984.
HDFM [19]	1982	Hughes Aircraft Co. - EUA	Modelo estático. Rede de chaveamento por pacotes.
Manchester Dataflow Machine [20, 21, 22]	1976	Universidade de Manchester - Inglaterra	Modelo dinâmico.
MTTDA [23]	1975-1990	MIT - EUA	Modelo dinâmico. Máquina com 64 elementos de processamento conectados por rede n-cúbica.
SIGMA-1 [24]	1982-1988	ETL Electrotechnical Laboratory - Japão	Operacional desde 1988.
Dataflow Processor Array System [25]	1983	NTT - Nippon Telephone and Telegraph - Japão	Modelo dinâmico. Protótipo EDDY operacional em 1983.
Q-p [26]	1987-1989	Universidade de Osaka, Sharp, Matsushita Electric Ind., Sanyo Electric e Mitsubishi Electric - Japão	Modelo dinâmico.
DDDP [27]	1980-1982	OKI Electric Ind. - Japão	Modelo dinâmico. Protótipo com 4 elementos de proc. operacional em 1982.
PIM-D [28]	1984-1986	OKI Electric Ind - Japão	Protótipo com 16 processadores operacional em 1986.
PATTSY [29]	1989	Universidade de Queensland - Austrália	Modelo dinâmico. Protótipo com 18 processadores está operacional.
Monsoon [30, 31]	1988-1996	MIT - EUA	Evolução da MTTDA - rede de chaveamento por pacotes multi-estágio. Protótipo operacional em 1988.
JUMBO [32]	1982	Universidade de Newcastle	Integração dos modelos fluxo de dados e fluxo de controle.
MADAME [33]	1988-1991	Instituto Jozef Stefan - Eslovenia	Modelo de fluxo de dados híbrido.
MUSE [34]	1985	Universidade de Nottingham - Inglaterra	Arquitetura híbrida.
RAMPS [35]	1987	EIP Inc. - EUA	Arquitetura híbrida.
DTN [36]	1990	Dataflow Technology Netherland - Holanda	Contém oito módulos, cada um com quatro processadores NEC ImPP.
ETCA [37]	1993	ETCA - França	Arquitetura híbrida.
Epsilon-2 [38]	1990	Sandia National Laboratories - EUA	Modelo dinâmico. Sistema construído usa crossbar 4x4.
EM-4 [39]	1986-1994	ETL Electrotechnical Laboratory - Japão	Protótipo operacional desde 1990 com 80 nós. Cada nó usa o chip dataflow EMC-R.

TABELA 1.1 - Arquiteturas a Fluxo de Dados (continuação)

TAM [40]	1991-1993	University of California, Berkeley - EUA	Arquitetura Multithreaded
*T - Star T [41]	1992	MIT - EUA	Descendente da Monsoon.
ADARC [42]	1993- em andamento	J. W. Goethe University - Alemanha	Protótipo com 12 processadores está operacional.
Pebbles [43]	1995	Colorado State University - EUA	Arquitetura Multithreaded
EARTH [44]	1993-1996	McGill University e Concordia University, no Canadá.	Arquitetura Multithreaded
ASTOR [45]	1987-1992	University of Augsburg - Alemanha	Modelo dinâmico.
HDCA [46]	1997	University of Kentucky - EUA	Arquitetura híbrida.
SEED [47]	1997	IFSC - USP	Modelo dinâmico.
WOLF [6, 7, 8]	1991-1998	IFSC - USP	Modelo dinâmico.

O objetivo do presente trabalho consiste na proposta e simulação de redes de interconexão para a arquitetura APO<sup>3</sup>. Um diferencial deste trabalho em relação a grande parte dos trabalhos desenvolvidos na área de redes de interconexão consiste no fato que muitos desses trabalhos, em geral, abordam somente aspectos teórico-conceituais [48], que também são relevantes, mas muitas vezes utilizam hipóteses ideais para desenvolverem suas análises com suposições que por vezes estão um pouco distante do comportamento dos sistemas reais. Este trabalho aborda um aspecto mais prático da área de redes de interconexão, considerando um sistema "real" (arquitetura APO). Para tal, são analisados os requisitos de comunicação da arquitetura bem como os fundamentos de redes de interconexão.

#### 1.4 Formalismo de Interconexão

Como o foco deste trabalho é conectividade faz-se necessário definir formalmente seus conceitos básicos. As redes de interconexão têm sido estudadas há

<sup>3</sup> A arquitetura APO é descrita no capítulo 2 do presente trabalho.

vários anos e seus aspectos teóricos fundamentais já estão bem quantificados. O presente trabalho está embasado nos principais avanços da teoria de redes de interconexão.

### 1.4.1 Redes

As arquiteturas de fluxo de dados, como todas as arquiteturas MIMD são fortemente dependentes de um sistema de interconexão que possibilite a comunicação entre seus elementos [49, 50, 51, 52, 53]. Genericamente denomina-se de rede de interconexão o sistema que viabiliza essa comunicação. A figura 1.3 mostra a utilização de redes de interconexão na interligação entre processadores e entre processadores e módulos de memória em dois sistemas paralelos genéricos.

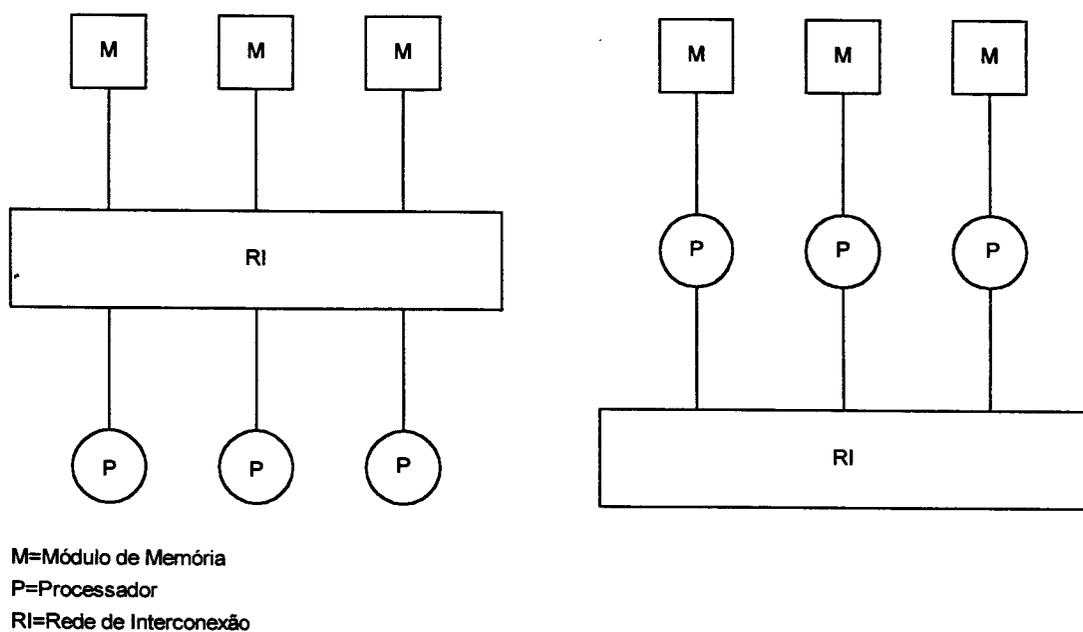


FIGURA 1.3 - Redes de Interconexão em Sistemas Paralelos

Bhuyan [49] define uma rede de interconexão como um arranjo de chaves e conexões que permitem a comunicação de dados entre os elementos que constituem um sistema de computação paralela.

#### 1.4.2 Topologia

Os elementos que constituem um sistema de processamento paralelo podem ser interligados de diversas maneiras. A estrutura de interligação varia conforme a topologia escolhida para a rede de interconexão. A decisão da escolha da topologia tem papel fundamental sobre os custos de comunicação e o desempenho de um sistema. Como consequência de uma escolha pouco criteriosa o custo dos elementos do sistema paralelo pode ser superado pelo custo da rede de interconexão. Portanto, um dos requisitos básicos de projeto de redes de interconexão é que o custo da rede corresponda a uma pequena parcela do custo total do sistema [49, 50].

Uma estrutura de interconexão ideal é mostrada na figura 1.4. Essa rede apresenta a melhor estrutura de comunicação, visto que ela permite que um módulo se comunique com qualquer outro, mesmo que estejam ocorrendo outras comunicações simultâneas. Portanto, ela possibilita que se efetue o número máximo de transações (comunicações) [51, 52].

Na estrutura de uma rede ideal constituída de  $N^*$  módulos existe uma conexão direta entre cada um dos  $N$  módulos com os demais. Isso resulta em  $N*(N-1)/2$  conexões, pois cada módulo possui ligações com os  $(N-1)$  módulos restantes. Portanto, o custo de interconexão para  $N$  módulos cresce com o fator de  $N^2$ . Esse

---

\* No estudo de Redes de Interconexão o número de nós (módulos) é um parâmetro muito importante, e é freqüentemente representado pela variável  $N$ .

custo quadrático torna-se um fator agravante à medida que  $N$  cresce, pois o custo da rede extrapola a expectativa de ser apenas uma fração minoritária do custo total do projeto. Além disso, o aumento do sistema torna cada módulo mais complexo. Há a tendência de que o custo de comunicação torne-se maior que o custo do restante do hardware.

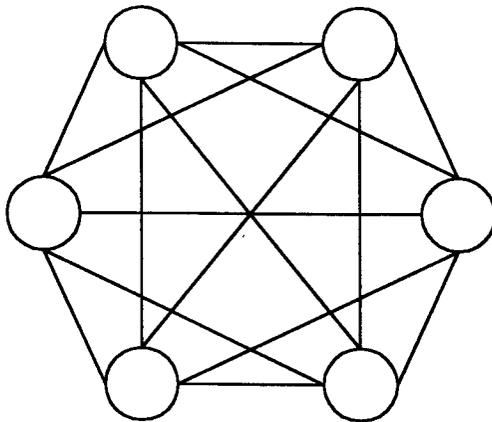


FIGURA 1.4 - Rede de Interconexão Ideal

A complexidade e custo apresentados pelo modelo de rede ideal tornam seu uso proibitivo para sistemas com grande número de módulos (para  $N$  grande). Mesmo assim ela ainda possui características bastante desejáveis para uma rede de interconexão [51]. Como essa estrutura interliga cada módulo do sistema a todos os outros restantes ela possibilita uma *Conectividade* máxima, ou seja, permite o máximo grau de conexão do sistema. A conectividade de um sistema de processamento paralelo corresponde ao número de vizinhos imediatos que cada módulo (também chamado de nó) de processamento possui. Para uma arquitetura MIMD genérica a conectividade de uma rede de interconexão deve ser a maior possível.

É de se esperar também que a rede de conexão ideal realize o maior número de transações possíveis, já que existe a possibilidade de todos os módulos se comunicarem simultaneamente. A taxa de transações efetuadas pela rede é denominada de *Largura de Banda* ou *Throughput*. A largura de banda, portanto, corresponde ao volume de tráfego de mensagens que a rede de interconexão pode transmitir e conclui-se que ela deva ser a mais alta possível.

Cada transação consome um determinado tempo para ser realizada. Este tempo é denominado de *Latência de Comunicação* e corresponde ao tempo de trânsito de uma mensagem da fonte até atingir o módulo de destino final. A latência de comunicação tende a crescer com N, pois as distâncias geralmente crescem conforme aumenta-se o número de módulos. A latência de comunicação deve ser mínima. A tabela 1.2 a seguir resume as características ideais de uma rede de interconexão.

TABELA 1.2 - Características Ideais de uma Rede de Interconexão

Característica	Comportamento
Largura de Banda	Máxima
Conectividade	Máxima
Latência	Mínima
Custo	Mínimo

### 1.4.3 Elementos de Chaveamento

As transações realizadas pela rede de interconexão são feitas através de elementos que roteiam a informação por meio de links desde o nó fonte até o nó de destino. Os elementos que efetuam o roteamento podem ser constituídos desde simples dispositivos capazes de apenas identificar o endereço da mensagem recebida e transmiti-la para o próximo elemento até dispositivos mais complexos como

processadores e memórias. Genericamente esses elementos de roteamento que formam a rede são denominados de Elementos de Chaveamento (EC) [50, 54].

#### 1.4.4 Características Operacionais

As redes de interconexão podem ser classificadas por suas características operacionais [49, 50, 52]: modo de operação, estratégia de controle e método de chaveamento.

- **Modo de Operação:** Uma rede pode transferir dados de modo síncrono e modo assíncrono. O modo síncrono é caracterizado pela existência de um relógio global que provê um referencial de temporização. A estratégia de operação assíncrona trabalha sem a presença de um relógio global.
- **Controle:** A estratégia de controle corresponde à maneira como são gerados os sinais de controle que coordenam as funções de roteamento da rede. O controle pode ser centralizado ou distribuído (descentralizado). No tipo centralizado, uma unidade de controle central faz o gerenciamento da transmissão. Por outro lado, os sinais de controle podem ser originados localmente aos componentes do sistema sem depender de um controlador global, esse esquema é denominado de controle distribuído.
- **Chaveamento:** Há duas metodologias para chaveamento: por circuito e por pacotes. No chaveamento por circuito, um caminho entre fonte e destino é estabelecido antes de se iniciar a comunicação. Esse caminho é mantido durante toda a transmissão. Já no método de chaveamento por pacotes a mensagem é dividida em pacotes para ser emitida por meio da rede através

dos elementos de chaveamento que os recebem e os enviam a um novo elemento num estágio subsequente.

#### **1.4.5 Exemplos de Redes de Interconexão**

A seguir serão apresentadas algumas das estruturas de interconexão empregadas para efetuar a comunicação entre os elementos de uma arquitetura paralela. Nos exemplos a seguir considera-se  $N$  como sendo o número de nós da rede de interconexão.

##### **Barramento Compartilhado**

O barramento [49, 50] mostrado na figura 1.5, conecta  $N$  módulos através de um único canal compartilhado por todos os nós da rede. O barramento é uma das estruturas de interconexão menos complexas, porém é pouco eficiente. Ele pode realizar apenas uma transação por vez. Quando vários pares de módulos necessitam se comunicar simultaneamente deve haver um mecanismo de controle que resolva as disputas pelo barramento.

Como consequência, arquiteturas com um grande número de nós apresentam um longo tempo de espera pela posse do barramento. A latência de comunicação tende a aumentar proporcionalmente a  $N$  e a capacidade total de comunicação do barramento decresce conforme aumenta-se o número de módulos. Esses fatores tornam o barramento compartilhado um gargalo para sistemas com grande número de módulos.

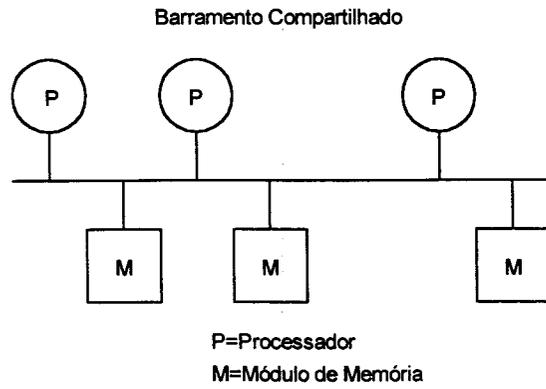


FIGURA 1.5 - Barramento Compartilhado

### Crossbar

A estrutura lógica da rede de interconexão *crossbar* [49, 50, 52, 54, 55, 56, 57, 58] é mostrada na figura 1.6. Esse tipo de estrutura aproxima-se bastante do modelo ideal que é capaz de realizar conexão total entre todos os módulos simultaneamente. Uma rede *crossbar*, permite todas as possíveis conexões simultâneas desde que não haja pedidos conflitantes, isto é, duas ou mais fontes requisitem o mesmo destino. Isso resulta em conflito quando dois ou mais módulos necessitam comunicar-se com o mesmo módulo de destino. O custo de uma rede *crossbar* que conecta N módulos de entrada a N módulos de saída é  $N^2$ .

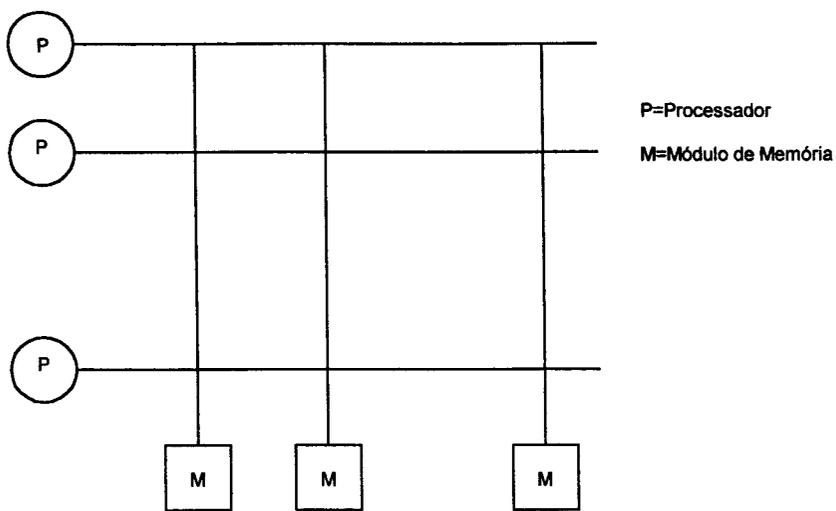


FIGURA 1.6 - Rede Crossbar

## Redes de Interconexão Multi-Estágio

A topologia de barramento compartilhado é atrativa pelo custo, mas pouco eficiente. No entanto, apesar de apresentar características próximas daquelas da rede ideal, a rede *crossbar* possui custo elevado para grandes sistemas, tornando-a inviável em termos de escalabilidade. Essas duas topologias situam-se em extremos opostos do espectro de custo-desempenho. Entre esses dois extremos encontram-se as redes multi-estágio [49, 50, 51, 52, 59] que apresentam um razoável equilíbrio entre custo e desempenho. Essa é a principal vantagem dessas redes. Elas consistem de dois ou mais estágios de elementos de chaveamento que definem caminhos de comunicação entre pares de módulos de entrada e saída.

Uma rede multi-estágio  $N \times N$  conecta  $N$  módulos de entrada a outros  $N$  módulos de saída. Em geral, essas topologias possibilitam uma ampla variedade de mapeamentos simultâneos de módulos de entrada para módulos de saída utilizando  $\log_2 N$  estágios, sendo cada estágio constituído de  $N/2$  chaves *crossbar*  $2 \times 2$ , considerando  $N$  uma potência de 2. Os elementos de chaveamento da rede, mostrados na figura 1.7, são formados por essas chaves *crossbar* com duas entradas e duas saídas. Por meio de um dígito (bit) de controle  $C$  que faz parte do endereço do nó de destino é possível fazer a conexão de uma entrada a uma saída.

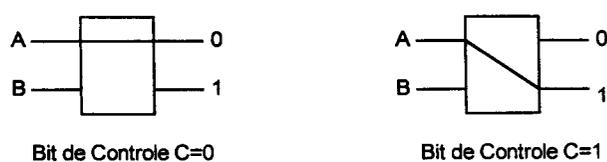


FIGURA 1.7 - Controle do Elemento de Chaveamento

O roteamento depende do bit de controle. Quando  $C=0$ , a entrada é conectada à saída superior e quando  $C=1$  a entrada é conectada à saída inferior. Dessa maneira, o

módulo (origem) que deseja efetuar comunicação com outro (destino) deve especificar o endereço de destino o qual contém um dígito de controle correspondente a cada estágio de chaves. Assim, a chave do  $i$ -ésimo estágio verifica o  $i$ -ésimo bit do endereço de destino contando a partir do dígito mais significativo para determinar se a entrada será conectada à saída superior ou inferior. Um exemplo de rede de interconexão multi-estágio é apresentado na figura 1.8, que mostra uma rede Delta [54]. Algumas das limitações das redes multi-estágio são [60]:

- Redes multi-estágio como a rede Delta, não podem realizar permutações arbitrárias num único passo, ou seja, para transferir informação é necessário passar por vários estágios, além disso, algumas combinações de pares de nós entrada-saída podem ocasionar conflitos, mesmo que os destinos requeridos pelos nós fontes sejam distintos. Por exemplo, para a rede Delta da figura 1.8 os pares de nós entrada-saída 4-1 e 2-0 vão precisar utilizar o mesmo canal de saída no  $EC_4$  (segundo estágio).
- Um outro aspecto a se considerar trata da complexidade do mecanismo de controle da rede, que tem de resolver conflitos e decidir a rota a cada estágio, em cada EC da rede.

## **1.5 Tecnologia WDM (Wavelength Division Multiplexing)**

As interconexões eletrônicas tradicionais como barramento compartilhado ou redes de interconexão multi-estágio apresentam limitações como: velocidade, complexidade de controle e escalabilidade [61, 62, 63]. Essas limitações conduziram a um esforço para o desenvolvimento de redes que utilizam tecnologia óptica.

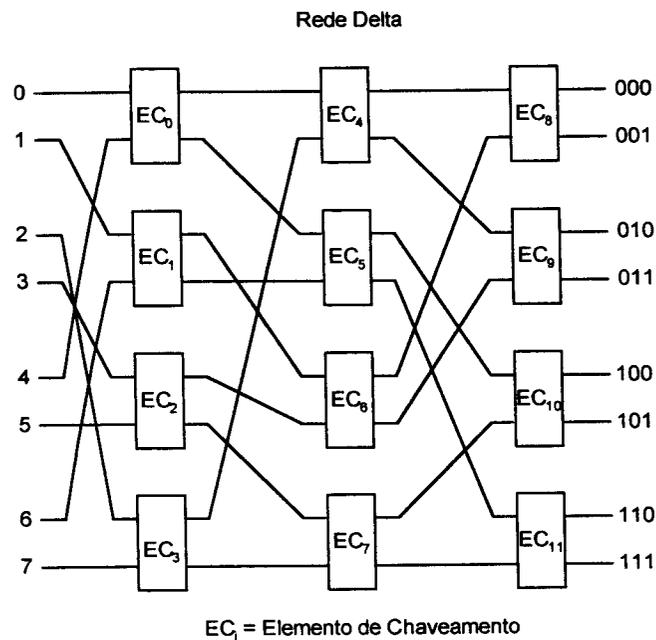


FIGURA 1.8 - Rede de Interconexão multi-estágio Delta

Nesse contexto cabe ressaltar a tecnologia WDM [10, 64, 65], pois as redes baseadas na técnica WDM utilizam com melhor eficiência a banda de passagem disponível nas fibras ópticas multiplexando sinais e utilizando transmissores e receptores sintonizados em frequências diferentes para estabelecer canais ponto-a-ponto independentes [66]. As redes baseadas em WDM têm algumas vantagens sobre as redes eletrônicas convencionais [63]:

- **Largura de Banda:** A largura de banda de uma fibra é de 25THz [10, 63] e há sistemas comerciais operando em Gb/s [67, 68] e sistemas experimentais em Tb/s [10, 67, 68, 70, 71, 72, 73, 74].
- **Baixo Nível de Perdas:** Em seu comprimento de onda de atenuação mínima, 1,55  $\mu\text{m}$ , a perda é da ordem de 0,2 dB/Km, enquanto que num cabo coaxial de cobre a perda é de 2,5 dB/Km em um sinal de 1 MHz e mais de 50 dB/Km em 1 GHz [10].

- **Escalabilidade:** A perda de potência em um barramento comum cresce linearmente com o número de processadores e num acoplador estrela passiva (PSC - *Passive Star Couple*) cresce logaritmicamente [63, 65].
- **Modularidade:** Na maioria dos sistemas *single-hop*<sup>4</sup> um nó pode ser acrescentado à rede sem necessidade de se reprojeter o sistema todo [64, 75].
- **Transparência de Protocolo:** Cada canal pode ser operado independentemente e pode usar um protocolo de comunicação diferente [68].
- **Multicasting:** *Broadcasting* e *multicasting* são viabilizados em uma rede WDM devido a natureza *broadcast* do acoplador PSC.

A técnica WDM é usada para dividir a largura de banda de uma fibra óptica em muitos comprimentos de onda, os quais são chamados de canais [63], e podem operar assincronamente e em diferentes velocidades. Esse mecanismo pode disponibilizar uma capacidade agregada de vários Tb/s.

### 1.5.1 Arquitetura WDM

As arquiteturas paralelas exigem algum tipo de rede de interconexão para conectar todos os elementos de processamento. Isso torna a rede de interconexão um dos principais desafios de projeto. Do ponto de vista de desempenho a chave *crossbar* acena como a opção mais adequada para redes de alta velocidade. Ela contém uma conexão de cada elemento de processamento para todos os outros e portanto, o diâmetro da chave é sempre um. No entanto, a desvantagem fundamental

---

<sup>4</sup> Sistemas *single-hop* são definidos na seção 1.5.1

da chave *crossbar* é a complexidade e custo de implementação. A tecnologia WDM viabiliza a implementação de chaves *crossbar* visto que os sinais ópticos podem ser distribuídos sem interferências por meio de um dispositivo passivo como o acoplador estrela passiva (PSC) [66, 76]. O acoplador estrela passiva é um dispositivo óptico *broadcast*, isto é, tem a capacidade de distribuir uma entrada recebida para todas as suas saídas. Essas características tornam a tecnologia WDM uma promissora candidata a desempenhar o papel de rede de interconexão de arquiteturas paralelas.

A figura 1.9 mostra um modelo genérico de um sistema utilizando uma rede de interconexão baseada em WDM. Cada nó nesse sistema tem transmissores e receptores e está conectado ao acoplador estrela passiva (PSC) através de fibras. Assim cada nó pode transmitir para o acoplador e receber do mesmo. Para transmitir o nó sintoniza um transmissor para um determinado comprimento de onda e envia a informação para o acoplador. Este recebe a informação e a distribui naquele comprimento de onda para todas as saídas, isto é, para cada nó da rede. Para receber a informação um nó deve sintonizar um receptor naquele comprimento de onda. As redes de interconexão devem empregar uma forma de controle para evitar que dois ou mais nós transmitam simultaneamente no mesmo comprimento de onda, o que ocasiona uma colisão. Esse tipo de controle que reduz ou elimina colisões é denominado de protocolo de controle de acesso ao meio.

Um sistema baseado em WDM pode ser classificado quanto ao modo de comunicação entre os nós em: *single-hop* e *multihop*. No *single-hop* qualquer nó pode transmitir para seu destino sem a utilização de nós intermediários; e no *multihop* a comunicação entre dois nós é realizada através de nós intermediários [64, 75, 77].

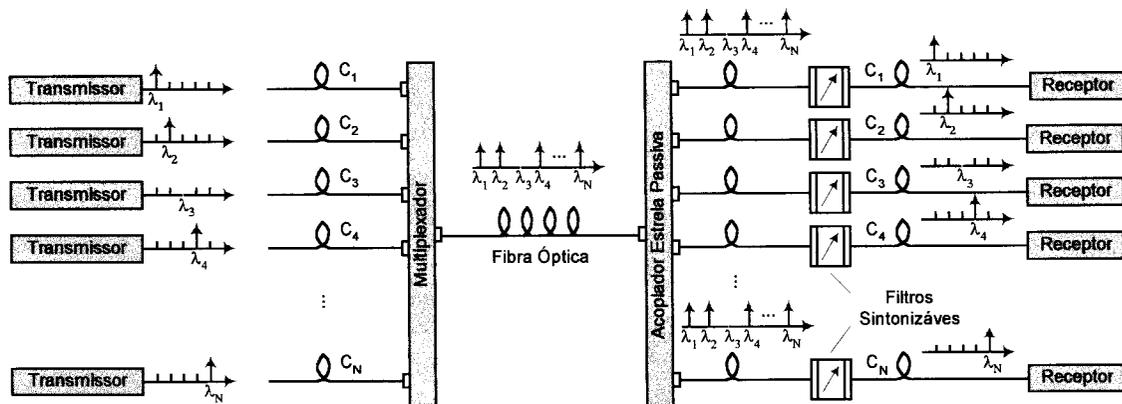


FIGURA 1.9 - Rede de Interconexão baseada em WDM

### 1.5.2 Taxonomia de Mukherjee

Mukherjee [75, 77] propôs uma classificação para sistemas WDM em função das características dos transmissores e receptores empregados. Essa classificação estabelece quatro tipos distintos de redes descritos a seguir:

- **FT-FR:** *Fixed Transmitter - Fixed Receiver* são redes com transmissores e receptores fixos, geralmente utilizadas em sistemas *multihop*.
- **TT-FR:** *Tunable Transmitter - Fixed Receiver* são redes com transmissores que podem ser sintonizados em diferentes comprimentos de onda, mas com receptores fixos em determinado comprimento de onda. Dessa forma, qualquer transmissor pode transmitir para qualquer receptor.
- **FT-TR:** *Fixed Transmitter - Tunable Receiver* são redes com transmissores fixos em determinado comprimento de onda, mas com receptores sintonizáveis. Os receptores controlam a comunicação sintonizando para o comprimento de onda do transmissor do qual receberão informação. Como as transmissões são realizadas em canais fixos as colisões podem ser evitadas e o *broadcast* pode ser facilmente efetuado.

- **TT-TR:** *Tunable Transmitter - Tunable Receiver* são redes com transmissores e receptores sintonizáveis. Permitem maior flexibilidade de configurações na estrutura de comunicação mas também podem exigir maior grau de complexidade de controle.

## 1.6 Conclusões

Arquiteturas paralelas constituem uma alternativa para a obtenção de sistemas computacionais de alta performance. No entanto, um dos principais desafios de projeto e frequentemente o gargalo de tais sistemas são as redes de interconexão. Diante das limitações das redes de interconexão eletrônicas tradicionais e do desenvolvimento e amadurecimento da tecnologia de comunicação óptica, o interesse de pesquisa na área de redes de interconexão migrou para a tecnologia WDM.

No espectro de modelos de arquiteturas paralelas propostas o modelo de fluxo de dados tem sido questionado por exigir grande capacidade de chaveamento. Essa característica parece complementar uma das principais características dos sistemas de comunicação óptica que é disponibilizar uma alta capacidade de comunicação. Surge então uma promissora possibilidade que é a de se investigar a implementação de comunicação óptica em arquiteturas a fluxo de dados. Isso tem o potencial de viabilizar arquiteturas baseadas no paradigma de fluxo de dados e outras arquiteturas paralelas.

## **Capítulo 2**

### **Propostas de redes de interconexão para a arquitetura APO**

#### **2.1 Introdução**

O capítulo 1 mostra que existe espaço para se propor uma chave de interconexão com tecnologia óptica capaz de viabilizar a arquitetura a fluxo de dados. Neste capítulo são apresentadas duas propostas de chaves de interconexão para a arquitetura APO.

#### **2.2 Arquitetura APO**

A arquitetura APO (Arquitetura Paralela com rede de interconexão Óptica) [78] é mostrada na figura 2.1. As unidades que compõem a arquitetura são: unidade de Entrada e Saída (E/S), unidade de Controle de Fluxo (CF), Memória de Estruturas (ME), Memórias de Dados (MDs) e os Elementos de Processamento (EPs) formados pelas Memórias de Instruções (MI) e Unidades Funcionais (UF). A comunicação entre as unidades é realizada através da troca de fichas que representam dados. Essas fichas também contêm informações de controle sobre o dado que elas representam.

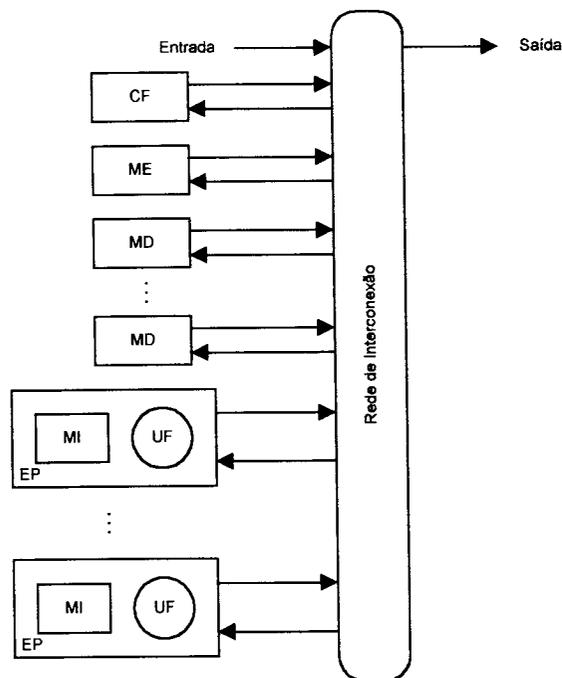


FIGURA 2.1 - Arquitetura APO

### 2.2.1 Descrição Funcional

A rede de interconexão envia as fichas de dados provenientes da Entrada para as Memórias de Dados, que são responsáveis pelo emparelhamento de operandos. No caso da MD de destino estar ocupada, a ficha é armazenada na unidade de Controle de Fluxo para posterior distribuição. As MDs geram fichas do tipo par (PA), constituídas por um par de fichas dirigidas ao mesmo nó no grafo de dependência de dados e com o mesmo rótulo. A ficha PA é distribuída para um dos Elementos de Processamento.

A Memória de Instruções armazena as informações do nó para o qual o par de operandos é dirigido. Ela produz uma ficha executável (EX), contendo todas as informações requeridas para executar o nó e a envia para sua correspondente Unidade Funcional que executa a ficha EX, produzindo resultados. A Memória de

Estruturas (ME) é usada para armazenar estruturas de dados como vetores e matrizes, explorando o paralelismo nessas estruturas de dados.

As principais características evolutivas da arquitetura APO em relação a arquitetura Wolf são:

- Redução do número de redes de interconexão: a arquitetura APO utiliza apenas uma rede de interconexão para conectar suas unidades, enquanto que a arquitetura Wolf emprega duas redes de interconexão. Isso tem implicações sobre a latência do sistema de comunicação, que é maior para a arquitetura Wolf, visto que as fichas devem trafegar através de duas redes. Outro aspecto negativo da utilização dessas duas redes se traduz em custo e complexidade de controle maiores.
- Múltiplas Memórias de Dados: a arquitetura APO emprega várias unidades de Memórias de Dados, o que reduz o gargalo de comunicação que era ocasionado pela utilização de uma única Memória de Dados na arquitetura Wolf.
- Eliminação da Memória de Instruções Externa: na arquitetura APO eliminou-se a Memória de Instruções Externa que gerava fichas para a Memória de Estruturas. A funcionalidade dessa memória externa passou a ser desempenhada pela Memória de Instruções Interna. Isso também reduz o gargalo de comunicação que era provocado pela utilização da Memória de Instruções Externa.

### **2.2.2 Simulador da Arquitetura APO**

Para investigar as redes de interconexão propostas para a arquitetura APO utilizou-se um simulador dirigido a tempo desenvolvido em GNU C++, com aproximadamente 8.000 linhas de código. O simulador implementa as características da arquitetura descrita na subseção 2.2.1 e apresentada na figura 2.1. Cada unidade da arquitetura é definida como uma classe, as sete classes são: E/S para entrada e saída, MI para memória de instruções, P para as unidades funcionais, CHAVE para rede de interconexão, CF para unidade de controle de fluxo, ME para memória de estruturas e MD para a memória de dados. As vantagens desse tipo de abordagem (orientada a objetos) são: encapsulamento, portabilidade de código, exploração de herança dentre outras. O simulador implementa um mecanismo de temporização denominado de relógio global o qual é incrementado a cada vez que todas as unidades da arquitetura são chamadas no simulador. No final da execução de um programa o valor final apresentado pelo relógio global representa o número total de ciclos consumidos para a simulação do referido programa. A menor unidade de tempo gerenciada pelo simulador é denominada de *time step* ou de ciclo.

### **2.3 Requisitos de Comunicação**

Os requisitos de comunicação de cada unidade da arquitetura são descritos a seguir, apresentando-se os destinos almejados por cada uma delas. Isso permite que se conheça o fluxo das fichas na arquitetura. As unidades também serão referenciadas como *nós* da arquitetura.

### 2.3.1 Destinos para E/S

As fichas provenientes da Entrada podem ser endereçadas às MDs, à CF e aos EPs.

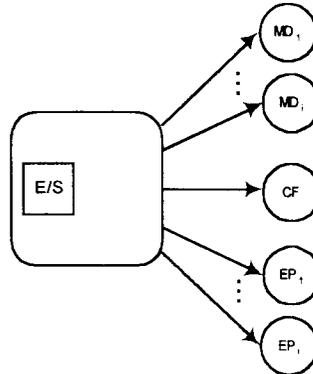


FIGURA 2.2 - Destinos para as fichas do nó E/S

### 2.3.2 Destinos para CF

As fichas que são armazenadas na unidade CF são endereçadas somente às MDs.

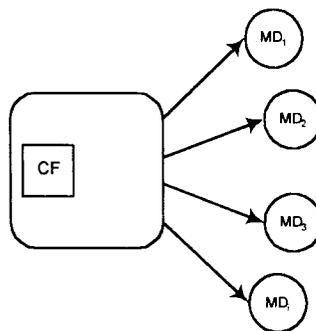


FIGURA 2.3 - Destino para as fichas do nó CF

### 2.3.3 Destinos para ME

As fichas armazenadas na unidade ME são endereçadas às MDs e à CF.

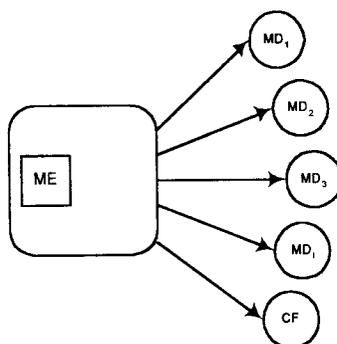


FIGURA 2.4 - Destinos para as fichas do nó ME

### 2.3.4 Destinos para MDs

As fichas armazenadas nas MDs podem ser endereçadas somente para os EPs.

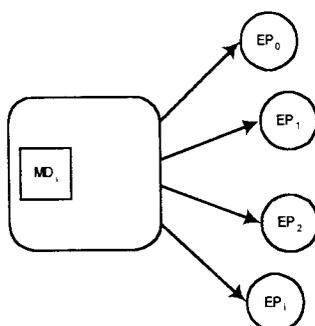


FIGURA 2.5 - Destinos para as fichas dos nós MDs

### 2.3.5 Destinos para EPs

As fichas provenientes dos EPs podem ser endereçadas às seguintes unidades:

MDs, CF, ME e Saída.

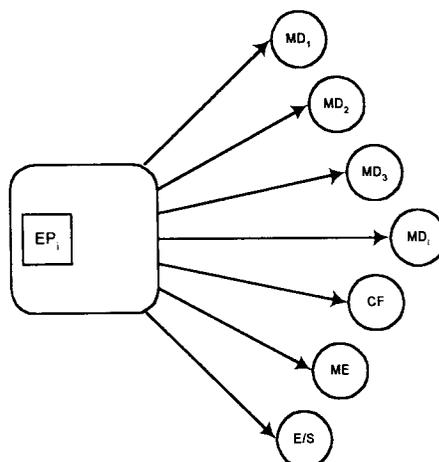


FIGURA 2.6 - Destinos para as fichas dos nós EPs

## 2.4 Rede de Interconexão

A rede de interconexão deve interligar as unidades da arquitetura obedecendo os requisitos de comunicação apresentados por elas. Deve haver algum tipo de controle que faça o gerenciamento quando dois ou mais nós tentam transmitir simultaneamente para o mesmo destino, evitando a ocorrência de colisões. Portanto, é necessário um protocolo de controle que arbitre o acesso ao meio de transmissão para permitir que as unidades efetuem suas transmissões.

Neste trabalho foram empregados dois tipos de protocolos de controle para a rede de interconexão da arquitetura: um utiliza controle descentralizado e o outro controle centralizado. O primeiro é baseado na técnica TDM (*Time Division Multiplexing*) [75] e o segundo utiliza um nó controlador central. A arquitetura da rede é baseada no sistema WDM *single-hop*. Cada nó da rede recebe dados através de um canal (comprimento de onda) específico. Dessa forma, um nó transmissor que queira transmitir para um determinado destino deve ter seu transmissor sintonizado no comprimento de onda específico do receptor de destino desejado.

O controle para transmissão é realizado durante o período de tempo denominado de ciclo de controle. Após o ciclo de controle é realizada a transmissão do pacote de dado, e esse período é então chamado de ciclo de transmissão. A seguir serão apresentadas as duas técnicas de controle usadas.

### 2.4.1 Protocolo de Controle baseado na técnica TDM

No sistema de controle de acesso baseado na técnica TDM, cada nó tem um período de tempo para realizar o controle de transmissão. Cada nó tem um receptor denominado de receptor de verificação que é usado para verificar se o nó de destino está livre para receber o dado.

O acesso de cada nó ao canal de transmissão de dados é realizado através de um transmissor sintonizável e um receptor fixo. Para transmitir, o nó fonte deve sintonizar seu transmissor no comprimento de onda do nó destino. A recepção de cada nó é realizada através de uma onda portadora independente. Isso evita a dependência do sistema em relação à sintonização dos receptores e também evita colisões de receptores em que o receptor não está sintonizado no comprimento de onda de um determinado nó transmissor.

Esse controle evita colisões de transmissores, pois os nós só transmitem para os destinos que estão livres e nunca iniciam um ciclo de transmissão para o mesmo destino concorrentemente. O ciclo de controle de cada nó é seguido pelo seu correspondente ciclo de transmissão, como mostra a figura 2.7. Para otimizar a comunicação, os ciclos de controle e de transmissão dos nós podem ser sobrepostos. Enquanto um nó está realizando seu ciclo de transmissão o outro nó pode realizar seu ciclo de controle. Isso evita os atrasos que ocorreriam no caso de se esperar que cada nó efetuasse seu ciclo de controle e de transmissão para depois iniciar o mesmo processo para o nó subsequente.

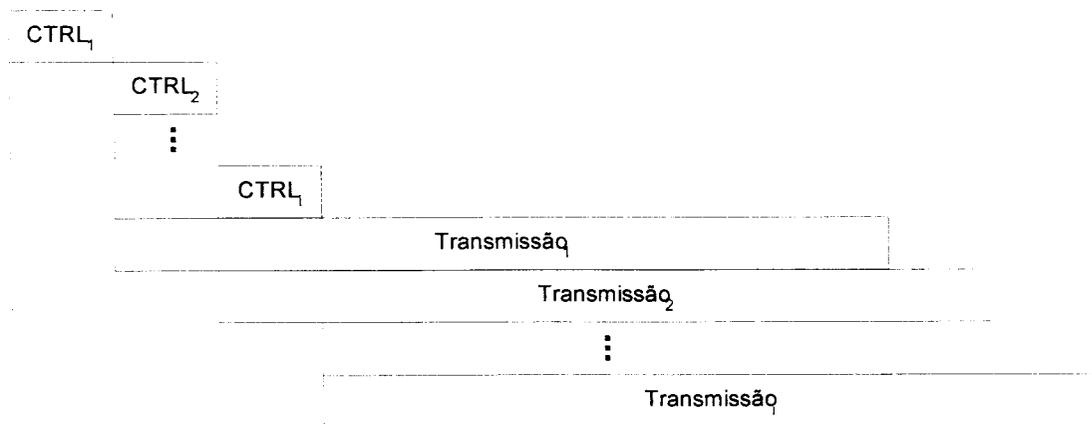


FIGURA 2.7 - Ciclos de Controle e Transmissão para o Protocolo de Controle TDM

Cada nó tem um período de tempo para realizar seu controle de transmissão. Nesse período de controle o nó sintoniza seu receptor de verificação no comprimento de onda do destino e verifica se este está ocupado ou não. A duração desse ciclo de controle deve ter um tamanho que seja suficiente para que o dado que porventura esteja sendo transmitido por algum outro nó atinja o destino. Assim, o nó que está fazendo a verificação de seu destino poderá detectar se ele está livre ou não. A figura 2.8 mostra a transmissão do nó A para o nó C, enquanto o nó B está iniciando seu ciclo de controle. O ciclo de controle do nó B deve durar um período de tempo que seja suficiente para que o dado enviado pelo nó A atinja o destino, no caso o nó C. Se a duração do ciclo de controle for menor que o tempo necessário para que o dado do nó A atinja o nó C, então quando o nó B for verificar seu destino ele não vai detectar a presença do dado no nó C e vai iniciar seu ciclo de transmissão provocando uma colisão com a transmissão do nó A.

A duração do ciclo de controle apresenta então uma relação com a distância entre os nós. Como a velocidade da luz na fibra é aproximadamente  $0,7 * c$ , ou seja, cerca de  $2,1 * 10^8$  m/s [79], sendo  $c$  a velocidade da luz no vácuo, para um determinado tempo de controle  $t_c$  a distância percorrida pelo sinal é  $d = t_c * 2,1 * 10^8$  m. Neste

trabalho adotou-se o tempo de controle de 1ns, visto que há dispositivos com tempo de sintonia com esse desempenho [65], o que implica que a distância máxima entre os nós seja de 21 cm.

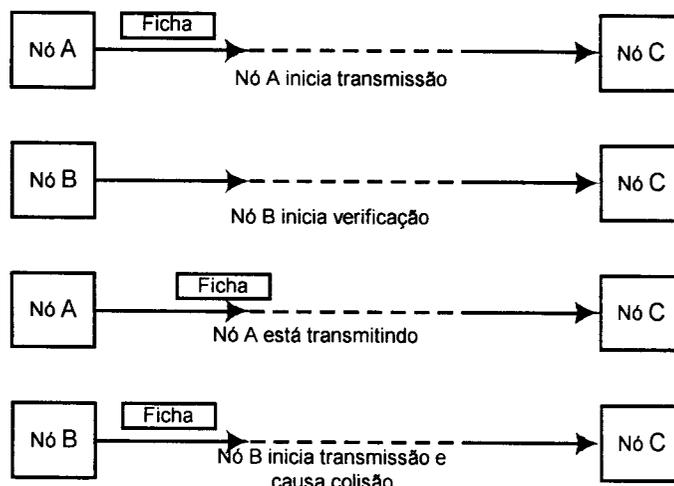


FIGURA 2.8 - Tempo de controle

#### 2.4.1.1 Descrição dos Nós para o Protocolo de Controle TDM

A seguir são apresentadas as estruturas de cada nó em termos de transmissores e receptores para o protocolo de controle TDM. Como mostram as figuras de 2.2 a 2.6, com exceção das MDs e da própria CF, todos os outros nós podem endereçar suas fichas para a unidade CF. Com base nisso, optou-se por utilizar um transmissor fixo no comprimento de onda endereçado à CF. Assim, tais nós quando tentam transmitir para as MDs e não conseguem, podem então tentar transmitir para a CF, sem exigir nova sintonização do transmissor.

Cada MD utiliza um transmissor sintonizável, já que há vários destinos (EPs) endereçados por esse nó. Caso se optasse por transmissores fixos, para cada novo EP adicionado à arquitetura, seria necessário mais um transmissor em cada MD, sintonizado nesse novo EP. Além disso, cada novo EP também precisaria de mais

transmissores: um para a ME, um para a E/S e um para cada MD. Por isso, os nós E/S, CF e ME que transmitem para as MDs também utilizam transmissores sintonizáveis. Eles empregam apenas um transmissor fixo no comprimento de onda da CF para evitar atrasos quando não conseguem transmitir para a MD desejada.

#### Nó E/S

O nó E/S (figura 2.9) insere as fichas na máquina, transmitindo-as ou para os EPs, ou para as MDs ou para a unidade CF. As fichas de programa são enviadas para os EPs, o nó fonte verifica se os destinos estão livres através do receptor de verificação  $RX_{VMDi-EPi}$  e transmite as fichas através do transmissor  $TX_{MDi-EPi}$  sintonizado no comprimento de onda do receptor presente nos nós EPs. Para receber as fichas do tipo programa cada EP possui um receptor fixo sintonizado num determinado comprimento de onda comum a todos os EPs. Dessa forma o nó E/S apenas faz um *multicast* dessas fichas.

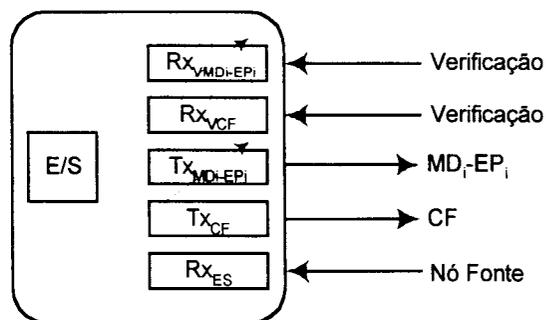


FIGURA 2.9 - Estrutura do nó E/S para controle TDM

As fichas de dados são endereçadas às MDs ou à unidade CF. A prioridade de transmissão é para as MDs e caso não seja possível, então a ficha é enviada para a CF. O nó fonte verifica através do receptor de verificação ( $RX_{VMDi-EPi}$  para as MDs e  $RX_{VCF}$  para a CF) se o nó de destino está livre ou não, e então pode realizar a

transmissão. A lógica de controle do nó E/S faz a transmissão para a MD desejada quando os dois destinos estão disponíveis. Os transmissores  $T_{X_{MD_i-EP_i}}$  e  $T_{X_{CF}}$  transmitem a ficha para seus destinos, e o receptor  $R_{X_{ES}}$  recebe as informações provenientes dos demais nós da arquitetura.

### Nó CF

O nó CF tem o papel de regular o fluxo de fichas que circulam pela máquina. As fichas endereçadas às MDs que não puderam ser enviadas a estas unidades são transmitidas para a unidade de controle de fluxo que as armazena para distribuí-las posteriormente. A unidade CF endereça suas fichas para as MDs e sua estrutura de transmissão é mostrada na figura 2.10. O nó CF possui receptor de verificação, transmissor sintonizável nos comprimentos de onda das MDs e o receptor  $R_{X_{CF}}$  para receber fichas dos outros nós.

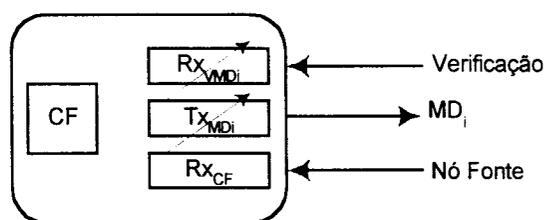


FIGURA 2.10 - Estrutura do nó CF para controle TDM

### Nó ME

Esse nó transmite fichas para as MDs e para a unidade CF. Sua estrutura é mostrada na figura 2.11. Ele usa receptores de verificação para efetuar o ciclo de controle, um receptor  $R_{X_{ME}}$  para receber fichas provenientes dos demais nós e os transmissores para enviar fichas para as MDs e para a unidade CF.

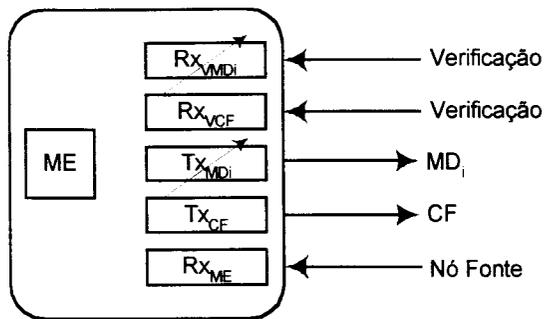


FIGURA 2.11 - Estrutura do nó ME para controle TDM

### Nós MDs

Os nós MDs, cuja estrutura é mostrada na figura 2.12, enviam suas fichas para os nós EPs. Cada nó MD possui um receptor de verificação sintonizável para identificar qual dos EPs se encontra disponível para receber a ficha, um receptor  $RX_{MDi}$  para receber fichas e um transmissor sintonizável para enviar fichas. A MD verifica a cada ciclo de controle se um EP está livre.

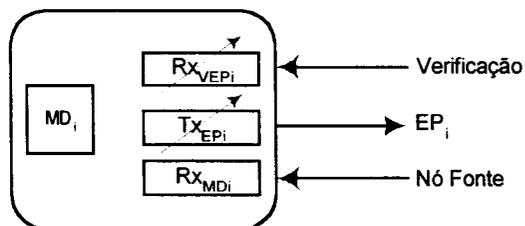


FIGURA 2.12 - Estrutura dos nós MDs para controle TDM

### Nós EPs

Os EPs (figura 2.13) transmitem fichas para as MDs, para a unidade CF, para a ME e para o nó E/S. Possuem receptor sintonizável para a verificação dos destinos ( $RX_{vi}$ ) e o correspondente transmissor sintonizável ( $Tx_i$ ). Possuem também um receptor verificador ( $RX_{VCF}$ ) e um transmissor ( $TX_{CF}$ ) para a CF. O mecanismo de transmissão de fichas para as MDs e CF é idêntico ao de outros nós. A ficha do tipo EX é enviada à ME, e a do tipo SA é transmitida para o nó E/S.

Como a E/S emite fichas do tipo PR, seu controle faz *multicast* dessas fichas para todos os EPs. Assim, cada EP tem um receptor fixo num determinado comprimento de onda comum a todos os outros EPs para receber essas fichas.

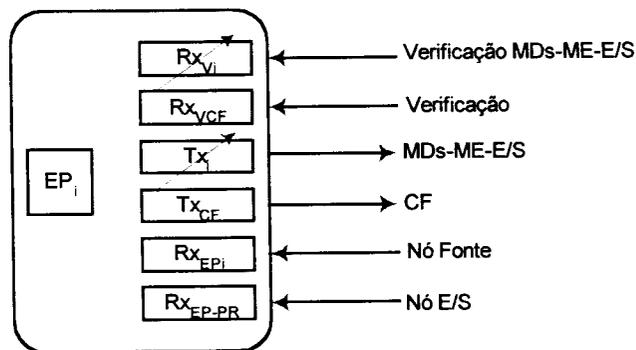


FIGURA 2.13 - Estrutura dos nós EPs para controle TDM

#### 2.4.2 Protocolo de Controle com Nó Central

Nesse protocolo cada nó que deseja transmitir seu dado faz um pedido de transmissão para o nó central (figura 2.14). O nó controlador central analisa os pedidos e envia as respostas para os nós que requisitaram permissão para transmitir. Os pedidos e respostas são formados por um byte de informação. Uma resposta com o valor 1 (um) indica para o nó requisitante que ele está habilitado a efetuar sua transmissão, e uma resposta com o valor 0 (zero) indica que seu pedido foi indeferido. No caso dos nós que endereçam a unidade CF, eles têm permissão para transmitir para essa unidade quando o controlador retorna um valor de 255, com todos os bits em 1.

Os pedidos podem ser feitos em paralelo durante o ciclo de controle. Cada nó envia o endereço do destino para o qual deseja transmitir, o controlador tem um algoritmo de controle que verifica os pedidos e evita que ocorram colisões no caso de mais de um nó requisitar o mesmo destino. O controlador utiliza a técnica

*round-robin* para gerenciamento dos pedidos. Assim, a cada ciclo de controle alterna-se a ordem de transmissão e o nó de maior prioridade num ciclo será o de menor prioridade no ciclo subsequente.

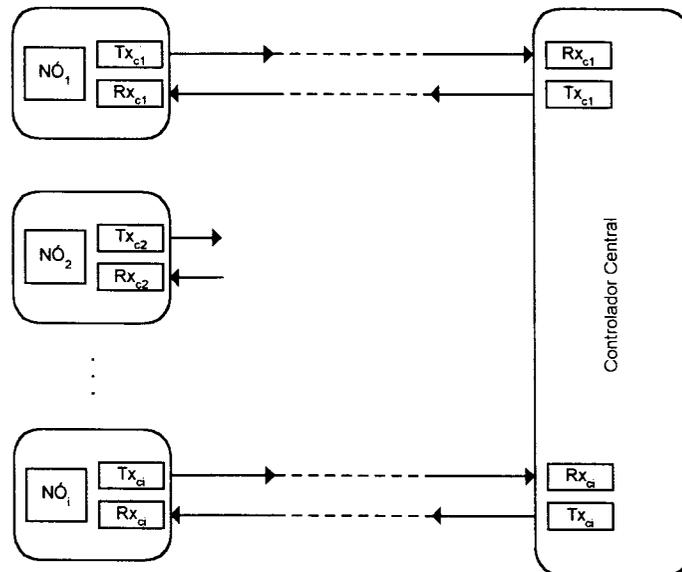


FIGURA 2.14 - Rede com Controlador Central

Os ciclos de controle e transmissão podem ser sobrepostos como mostra a figura 2.15. Somente no primeiro ciclo de controle há um atraso para se iniciar a transmissão, pois nesse caso, é necessário processar o primeiro ciclo de controle antes de se iniciar o primeiro ciclo de transmissão. A partir daí, pode-se sobrepor esses ciclos, visto que, enquanto está ocorrendo o ciclo de transmissão anterior, pode-se iniciar o ciclo de controle subsequente em paralelo, otimizando a comunicação da rede de interconexão.

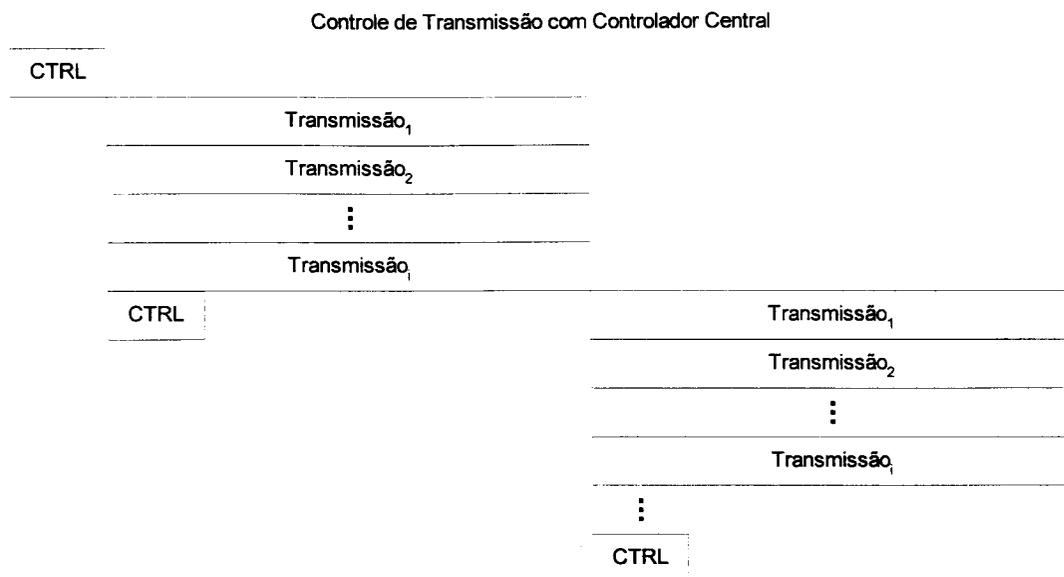


FIGURA 2.15 - Ciclos de Controle e Transmissão para o Protocolo de Controle com Controlador Central

#### 2.4.2.1 Descrição dos Nós para o Protocolo com Controlador Central

A seguir são apresentadas as estruturas de cada nó em termos de transmissores e receptores para o protocolo com controlador central. Cada nó deve ter um transmissor e um receptor para efetuar o ciclo de controle, além dos transmissores e receptores para realizarem a transmissão e recepção das fichas. Esse protocolo não utiliza receptores de verificação como no caso do protocolo baseado na técnica TDM.

##### Nó E/S

A estrutura do nó E/S para o protocolo de controle com controlador central é mostrada na figura 2.16. O nó faz o pedido de transmissão ao controlador através do transmissor de controle Tx<sub>CES</sub> e recebe a resposta do seu pedido através do receptor de controle Rx<sub>CES</sub>. Os outros transmissores enviam fichas para os EPs, para as MDs e para a unidade CF. O esquema de transmissão para a unidade CF é semelhante

àquele utilizado no protocolo TDM. Ao invés de se utilizar um receptor verificador para esta unidade o nó simplesmente recebe a resposta de permissão de transmissão através do receptor de controle, assim ele saberá se pode transmitir para a MD ou para a CF. O transmissor fixo destinado à CF permanece, já que o nó sintoniza primeiro tentando enviar fichas para a MD e caso não receba a permissão para transmitir para a MD e sim para a CF, então não precisa sintonizar novamente seu transmissor.

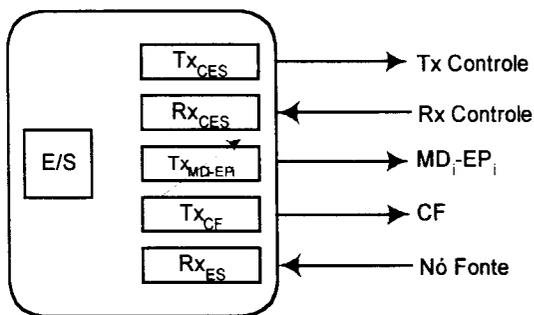


FIGURA 2.16 - Estrutura do nó E/S para controle central

#### Nó CF

O nó CF (figura 2.17) apresenta estrutura semelhante àquela utilizada para o protocolo TDM. Em vez do receptor verificador, para se efetuar o controle emprega-se um transmissor e um receptor de controle.

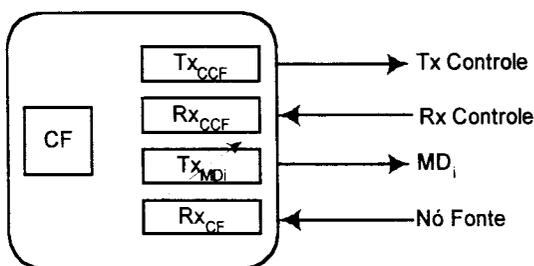


FIGURA 2.17 - Estrutura do nó CF para controle central

Portanto, essa configuração possui um dispositivo a mais ( $T_{X_{CCF}}$ ) que a sua correspondente para o protocolo TDM. O transmissor  $T_{X_{MDi}}$  envia fichas para as MDs e o receptor  $R_{X_{CF}}$  é usado para receber fichas provenientes de outros nós.

### Nó ME

Esse nó (figura 2.18) apresenta o mesmo número de dispositivos que o seu equivalente para o controle TDM. Mas utiliza três transmissores e dois receptores, enquanto que o controle TDM utiliza dois transmissores e três receptores.

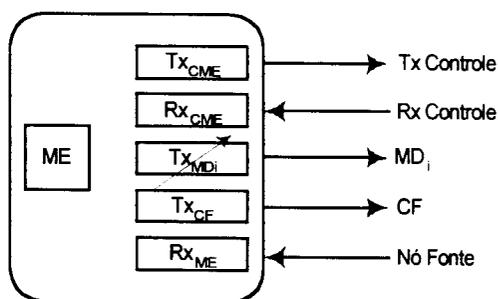


FIGURA 2.18 - Estrutura do nó ME para controle central

### Nós MDs

Os nós MDs, cuja estrutura é mostrada na figura 2.19, possuem um dispositivo a mais ( $T_{X_{CMDi}}$ ) que a estrutura correspondente para o controle TDM. Neste caso emprega-se um transmissor e um receptor para controle. O transmissor sintonizável  $T_{X_{EPi}}$  envia fichas para os EPs e o receptor  $R_{X_{MDi}}$  é usado para receber as fichas enviadas pelos outros nós.

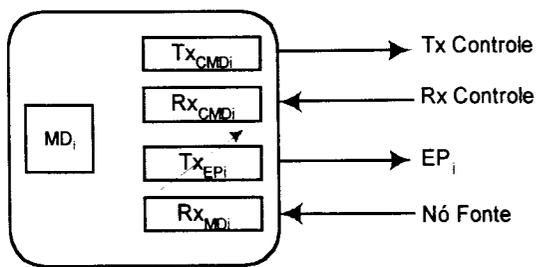


FIGURA 2.19 - Estrutura dos nós MDs para controle central

### Nós EPs

Os nós EPs para o controle com nó central (figura 2.20) mantêm o mesmo número de dispositivos que a configuração para o controle TDM. O transmissor sintonizável Tx<sub>i</sub> envia fichas destinadas às MDs, à ME e à E/S. O mecanismo de transmissão para a CF é o mesmo que para os outros nós que se comunicam com essa unidade. A recepção das fichas de programa é semelhante àquela do protocolo TDM, utilizando-se o receptor Rx<sub>EP-PR</sub> com uma portadora comum a todos os EPs.

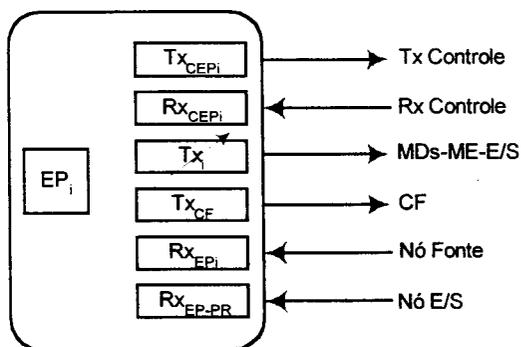


FIGURA 2.20 - Estrutura dos nós EPs para controle central

## 2.6 Conclusões

Este capítulo apresentou a arquitetura APO e suas características evolutivas em relação à arquitetura Wolf que a originou. O capítulo também analisou os requisitos de comunicação apresentados pela arquitetura APO e a partir dessa investigação

foram definidas duas arquiteturas de redes de interconexão para essa arquitetura. Foram propostos dois tipos de protocolos de controle, o primeiro baseado na técnica TDM com controle descentralizado e o segundo que utiliza um controlador central.

As duas redes propostas são otimizadas em termos de ciclos de controle e de transmissão, pois viabilizam a sobreposição dos mesmos, proporcionando maior eficiência do sistema de comunicação. Outra vantagem apresentada por essas redes é a inexistência de colisões, o que evita o *overhead* de comunicação que seria gerado caso houvesse necessidade de se tratar colisões.

Ainda, os dois modelos apresentam independência de receptores sintonizáveis, isto é, evitam as colisões de receptores que ocorrem quando o receptor não está sintonizado no comprimento de onda do transmissor que está lhe enviando o dado.

## Capítulo 3

### Comportamento da Chave: Distribuição de Carga

#### 3.1 Introdução

Este capítulo apresenta os resultados das distribuições de carga para as chaves de interconexão da arquitetura APO. A simulação das chaves de interconexão da arquitetura é complexa e gera resultados volumosos e de difícil interpretação. Para facilitar a análise, a apresentação desses resultados é segmentada.

Os parâmetros analisados são:

- **Distribuição de carga:** Investiga a eficiência do subsistema de chaveamento na transferência de atividade computacional entre os módulos do sistema. Esse aspecto é importante porque arquiteturas a fluxo de dados são de granulosidade fina, gerando um fluxo intenso de fichas em todos os canais.
- **Tempo de execução:** Investiga o impacto da eficiência da chave na execução global do algoritmo (discutido no capítulo 4). Esse aspecto é importante porque sabe-se que há grande variabilidade da carga computacional em arquiteturas a fluxo de dados, e as chaves apresentam comportamentos distintos sob diferentes cargas.

- **Índice de atividade:** Investiga a eficiência do controle das chaves sob diferentes cargas e tráfegos (discutido no capítulo 5).

Este capítulo apresenta os programas de teste utilizados para se investigar as chaves de interconexão da arquitetura APO, também apresenta a distribuição de carga da arquitetura Wolf e compara as metodologias de distribuição das duas arquiteturas (Wolf e APO).

### 3.2 Programas de Teste

Esta seção apresenta os programas de teste utilizados na investigação das chaves de interconexão da arquitetura APO. Foram adotados como programas de teste alguns algoritmos já extensivamente usados na pesquisa de arquiteturas a fluxo de dados por outros grupos de pesquisa [66, 80]. Todos os programas foram escritos em SISAL [81] que é a linguagem usada em outros trabalhos [5, 6, 8, 47, 66, 80]. Isso possibilita uma base uniforme para comparação dos resultados.

Os programas utilizados são os seguintes:

- **Função de Ackermann:** A função de Ackermann explora características de recursividade e o número de chamadas ao algoritmo para o cálculo de  $Ack(3,n)$  é dado por  $(128*4^n - 120*2^n + 9*n + 37)/3$ . O resultado pode ser obtido por  $Ack(3,n) - 1 = 2^{n+3} - 4$ .
- **Somatorial:** Esse programa calcula o somatório de N pela fórmula  $Soma(N, (N+1)/2+1) + Soma((N+1)/2, 1)$ . É um algoritmo duplamente

recursivo que gera uma árvore binária de processos em que folhas de um mesmo nível podem ser executadas em paralelo.

- **Integração Binária:** Esse programa integra a função  $y = x^2 - 6x - 10$ , num intervalo  $[x_1, x_2]$  através do método dos trapézios. É um programa com alto grau de recursividade que cria uma árvore de processos com  $2^{n-1}$  processos, em que  $n$  é o número de divisões do intervalo.
- **Eliminação de Gauss:** Esse algoritmo é extensivamente usado em soluções de sistemas de equações lineares e usa o método de triangularização com pivotamento parcial.
- **Interpolação:** Esse programa usa o método de interpolação polinomial para efetuar aproximações.
- **Laplace:** Esse algoritmo resolve uma equação simples de Laplace através do método de relaxação.
- **Multiplicação de Matrizes:** Esse programa cria e multiplica duas matrizes quadradas.
- **N-Queens:** Esse programa consiste na solução recursiva do problema de colocar  $N$  rainhas em um tabuleiro  $N \times N$  de forma que nenhuma ataque a outra. É um algoritmo típico de estratégia dividir-para-conquistar.

- **Quick Sort:** Esse algoritmo é utilizado para ordenar um vetor de números.
- **Simple:** É um programa de simulação hidrodinâmica simples codificado pelo Lawrence Livermore National Laboratory que tem sido extensivamente usado por grupos de pesquisa em arquiteturas a fluxo de dados.

### **3.3 Chaves de Interconexão**

As simulações foram realizadas para várias taxas de transmissão. Tanto para o protocolo TDM quanto para o protocolo com Nó Central, as chaves são denominadas de Chave 1, Chave 2 e assim por diante até Chave 6, e cada uma corresponde a uma taxa de transmissão diferente, começando em 2,5 Gb/s para a Chave 1, 5 Gb/s, 10 Gb/s, 20 Gb/s, 40 Gb/s e 400 Gb/s para as demais chaves respectivamente.

### **3.4 Distribuição de carga dos processadores**

Esta seção apresenta as distribuições de carga para as diferentes estratégias de controle implementadas para a chave de interconexão da arquitetura APO. A distribuição de carga obtida pela chave da arquitetura Wolf [80] também é apresentada a fim de se comparar os modelos de chaves implementados pelas duas arquiteturas.

As distribuições comparativas em relação à arquitetura Wolf são analisadas em função do programa de Eliminação de Gauss, visto que esse foi o algoritmo utilizado na investigação da arquitetura Wolf. As três subseções seguintes apresentam os resultados relativos ao algoritmo de Gauss e as subseções posteriores apresentam os

resultados para os demais programas de teste, mostrando os resultados obtidos apenas para a arquitetura APO.

#### **3.4.1 Chave de Distribuição da Arquitetura Wolf**

A arquitetura Wolf possui uma chave de interconexão denominada Chave de Distribuição (CD) que é responsável pela distribuição de fichas aos processadores [80].

O protocolo de controle da Chave de Distribuição trabalha da seguinte maneira: a chave ao receber uma ficha identifica, sempre a partir do primeiro, qual dos processadores está livre para recebê-la, aí então, faz a transmissão da ficha.

#### **3.4.2 Distribuição de Carga para a Arquitetura Wolf para Gauss**

A figura 3.1 mostra a distribuição de carga da arquitetura Wolf para o programa de Gauss. O eixo das ordenadas representa o número de instruções, enquanto que no eixo das abcissas estão representados os processadores.

A distribuição de carga para a arquitetura Wolf não é uniforme, os primeiros processadores são sobrecarregados. O segundo processador é utilizado 67,43% em relação ao primeiro processador. A porcentagem de utilização do último processador em relação ao primeiro é de apenas 2,8%.

Essa característica de distribuição de carga não é desejável porque ocasiona um desequilíbrio, isto é, provoca sobrecarga em alguns processadores, enquanto outros processadores trabalham muito pouco, tornando o sistema desbalanceado. A característica desejada para um sistema é que ele seja homogêneo, que seus processadores tenham carga de trabalho uniforme, sem haver sobrecarga de uns e

ociosidade de outros. A arquitetura Wolf apresentou comportamento semelhante de distribuição de carga para todos os outros programas utilizados [80].

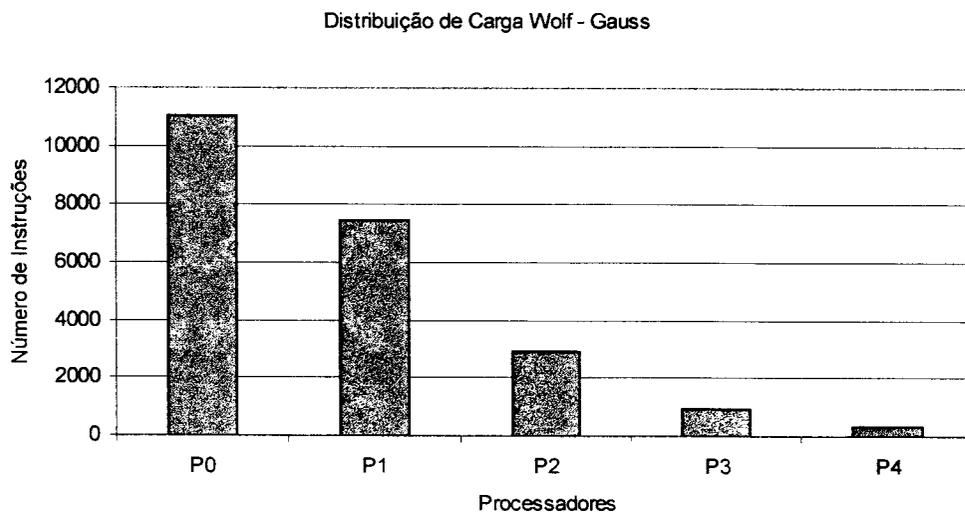


FIGURA 3.1 - Distribuição de Carga da arquitetura Wolf para Gauss

### 3.4.3 Distribuição de Carga para a Arquitetura APO com protocolo TDM para Gauss

Os resultados da distribuição de carga da arquitetura APO para o protocolo TDM usando o programa de Gauss são mostrados na figura 3.2. A figura 3.2 apresenta os resultados das distribuições de carga para cada processador em função da variação da velocidade da chave. A chave com controle TDM proporcionou uma distribuição de carga muito mais uniforme que a chave da arquitetura Wolf (figura 3.1). Neste caso, os processadores trabalharam de forma mais homogênea, os primeiros processadores não são sobrecarregados, nem os últimos ficam ociosos, como acontece para a chave da arquitetura Wolf. Para a arquitetura APO com chave TDM as diferenças entre as cargas dos processadores são menores.

Distribuição de Carga TDM - todas as chaves - Gauss

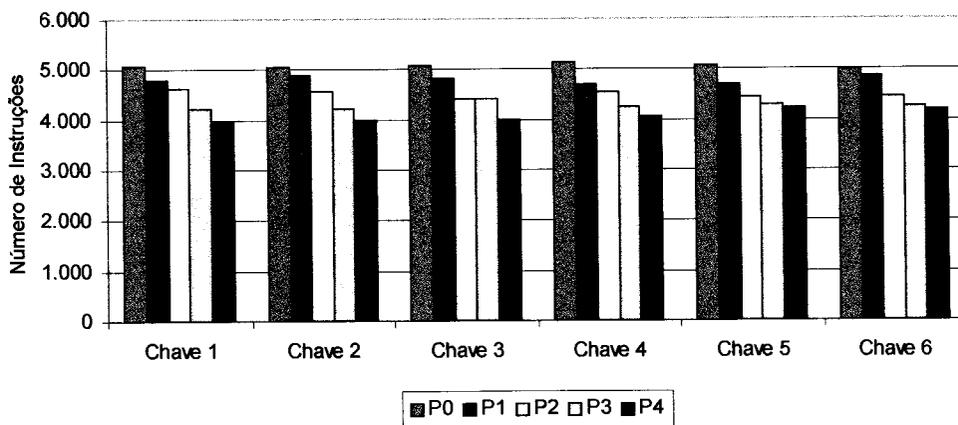


FIGURA 3.2 - Distribuição de Carga da arquitetura APO com o controle TDM para todas as chaves para Gauss

A apresentação consolidada como a da figura 3.2 dificulta a identificação dos resultados individuais, mas permite analisar a tendência global da computação. O Anexo A mostra os resultados individuais. A análise visual dos resultados mostrados na figura 3.2 já indica melhoras. A seção 3.4.5 quantifica analiticamente essas melhoras.

#### 3.4.4 Distribuição de Carga para a Arquitetura APO com protocolo com Nó Central para Gauss

Os resultados da distribuição de carga da arquitetura APO para o protocolo com Nó Central usando o programa de Gauss são mostrados na figura 3.3. Aqui também utilizou-se um gráfico que permite melhor visualização da tendência da distribuição de carga apresentada pela chave com controlador central. Os resultados individuais são mostrados no Anexo B.

A chave com esse modelo de controle também apresentou resultados bem melhores de distribuição que a chave da arquitetura Wolf. A distribuição resultante foi mais uniforme, sem sobrecargas ou ociosidade de processadores.

No entanto, comparando-se com a distribuição obtida pelo controle TDM, a chave com Nó Central apresentou distribuição um pouco mais dispersa, sendo que o processador P1 teve uma carga maior de trabalho em relação aos demais, e o processador P4 foi o menos utilizado.

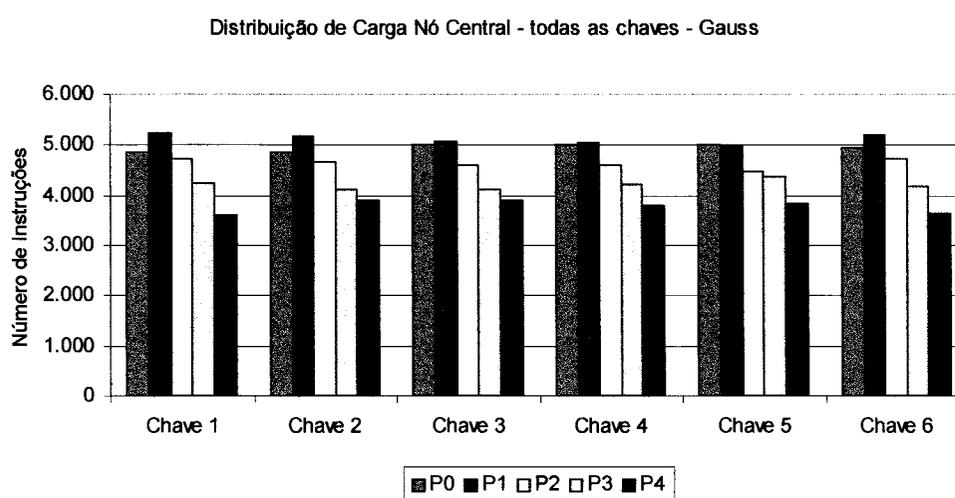


FIGURA 3.3 - Distribuição de Carga da arquitetura APO com o controle de Nó Central para todas as chaves para Gauss

### 3.4.5 Comparação entre as distribuições das duas arquiteturas para Gauss

As tabelas de 3.1 a 3.3 mostram o desvio padrão (D.P.) e o coeficiente de variação (C.V.) para as distribuições de carga apresentadas pelas chaves das duas arquiteturas para Gauss. O desvio padrão representa a variação dos valores em torno da média. Geralmente o desvio padrão é apresentado de forma relativa ao valor da média, essa relação é denominada de coeficiente de variação. Valores menores do coeficiente representam um sistema mais homogêneo.

TABELA 3.1 - Distribuição de carga para a arquitetura Wolf

Média	4.539,60
D.P.	4.601,84
C.V.	101,37

TABELA 3.2 - Distribuição de carga para a arquitetura APO para TDM

	Chave 1	Chave 2	Chave 3	Chave 4	Chave 5	Chave 6
Média	4.539,60	4.539,60	4.539,60	4.539,60	4.539,60	4.539,60
D.P.	447,72	446,36	413,20	417,20	357,99	349,49
C.V.	9,86	9,83	9,10	9,19	7,89	7,70

TABELA 3.3 - Distribuição de carga para a arquitetura APO para Nó Central

	Chave 1	Chave 2	Chave 3	Chave 4	Chave 5	Chave 6
Média	4.539,60	4.539,60	4.539,60	4.539,60	4.539,60	4.539,60
D.P.	628,19	523,00	524,38	525,25	489,34	625,74
C.V.	13,84	11,52	11,55	11,57	10,78	13,78

O resultado apresentado pela chave da arquitetura Wolf apresenta um coeficiente de variação muito alto em relação aos da arquitetura APO. Há uma grande dispersão da distribuição de carga do Wolf. Isso se deve à política de distribuição de fichas adotada pela chave de distribuição (CD) da arquitetura Wolf, que dá prioridade maior para os primeiros processadores, provocando uma sobrecarga dos mesmos em relação aos últimos processadores.

Entre as duas metodologias de distribuição da arquitetura APO, o protocolo com Nó Central apresentou maior dispersão que o TDM para todas as velocidades de transmissão. Os dois métodos apresentaram valores bem melhores que o da arquitetura Wolf, tendo uma distribuição mais homogênea.

Para o protocolo TDM no pior caso o processador com menor carga de trabalho tem uma porcentagem de utilização de 77,96% em relação ao processador com maior carga. E para o protocolo com Nó Central a porcentagem de utilização do processador com menor carga em relação ao processador com maior carga é de 68,91%, para o pior caso.

### **3.4.6 Distribuição de Carga para a Arquitetura APO**

Esta seção apresenta os resultados para os demais programas de teste utilizados na investigação das chaves da arquitetura APO. As tabelas de resultados apresentam na primeira coluna os nomes das unidades (P0, P1, ..., P4) e o nome dado aos parâmetros de comparação (Média, D.P. e C.V.). Na segunda coluna observa-se o número de instruções de cada unidade e os valores obtidos para os parâmetros para a velocidade da Chave 1 (conforme descrito na seção 3.3). As colunas subsequentes também apresentam o mesmo tipo de informações, mas referentes às demais velocidades, isto é, para a Chave 2 até a Chave 6.

#### **3.4.6.1 Distribuição de Carga para a Arquitetura APO para o protocolo TDM**

As tabelas de 3.4 a 3.12 mostram os resultados para o protocolo de controle TDM.

A tabela 3.4 mostra os resultados para Ackermann. A distribuição mais uniforme ocorreu para a Chave 4 com coeficiente de variação de 1,82 e o pior caso de distribuição ocorreu para a Chave 6. Para a Chave 4 a porcentagem de utilização do processador com menor carga em relação ao processador com maior carga é

de 95,9% e para a Chave 6 é de 74,2%. O processador que mais trabalhou utilizando a Chave 4 foi o P2 e o que menos trabalhou foi o P1. Para a Chave 6, a maior carga ficou no processador P4 e a menor carga no processador P1.

TABELA 3.4 - Distribuição de Carga para Ackermann - TDM

Distribuição de Carga						
	Chave 1	Chave 2	Chave 3	Chave 4	Chave 5	Chave 6
P0	8.303	8.303	8.137	8.259	8.377	8.167
P1	8.414	8.414	8.140	8.146	8.058	7.154
P2	8.702	8.688	8.612	8.498	8.588	8.286
P3	7.915	8.097	8.479	8.447	8.251	8.324
P4	8.236	8.068	8.202	8.220	8.296	9.639
Média	8.314,00	8.314,00	8.314,00	8.314,00	8.314,00	8.314,00
D.P.	285,46	253,77	218,05	151,35	192,92	883,70
C.V.	3,43	3,05	2,62	1,82	2,32	10,63

O programa Somatorial, tabela 3.5, apresentou boa distribuição de carga para todas as velocidades, com valores próximos, sendo que para a Chave 2 obteve-se o melhor resultado, com coeficiente de variação igual a 0,32. Para essa chave o processador com menor carga trabalhou 99,3% em relação ao que teve maior carga.

A Chave 4 apresentou a pior distribuição com coeficiente de variação de 0,59 e a porcentagem de utilização do processador com menor carga foi de 98,6% em relação àquele com maior carga. Para a Chave 2 o processador mais utilizado foi o P3 e o que menos trabalhou foi o P1. Para a Chave 4 o processador P4 foi o mais utilizado e o P3 o mais ocioso. Esse programa de teste foi o que apresentou os melhores resultados, tendo as distribuições mais uniformes, com os menores coeficientes de variação para quase todas as velocidades.

TABELA 3.5 - Distribuição de Carga para Somatorial - TDM

Distribuição de Carga						
	Chave 1	Chave 2	Chave 3	Chave 4	Chave 5	Chave 6
P0	57.488	57.979	58.243	58.100	58.101	58.261
P1	58.013	57.808	58.087	57.805	58.350	58.312
P2	58.170	58.152	57.927	57.800	58.142	57.880
P3	58.001	58.223	57.537	57.727	57.478	57.626
P4	58.313	57.823	58.191	58.553	57.914	57.906
Média	57.997,00	57.997,00	57.997,00	57.997,00	57.997,00	57.997,00
D.P.	311,90	188,03	284,06	342,17	328,90	286,59
C.V.	0,54	0,32	0,49	0,59	0,57	0,49

A tabela 3.6 mostra os resultados para o programa Integração Binária. Para esse programa a Chave 2 apresentou a melhor distribuição com coeficiente de variação de 0,41 enquanto a Chave 5 teve a distribuição mais dispersa. A Chave 2 propiciou uma pequena diferença de utilização entre os processadores, sendo que a porcentagem de utilização do processador com menor carga foi de 99,0% em relação ao processador que mais trabalhou. Já para a Chave 5 essa porcentagem de utilização foi de 97,1%.

O processador P3 foi o mais utilizado e o processador P2 foi o menos utilizado para a Chave 2. Para a Chave 5 o processador que mais trabalhou foi o P2 e o de menor utilização foi o P4.

TABELA 3.6 - Distribuição de Carga para Integração Binária - TDM

Distribuição de Carga						
	Chave 1	Chave 2	Chave 3	Chave 4	Chave 5	Chave 6
P0	4.729	4.696	4.724	4.680	4.679	4.648
P1	4.713	4.696	4.754	4.714	4.731	4.749
P2	4.710	4.688	4.682	4.774	4.788	4.681
P3	4.658	4.733	4.699	4.713	4.687	4.696
P4	4.725	4.722	4.676	4.654	4.650	4.761
Média	4.707,00	4.707,00	4.707,00	4.707,00	4.707,00	4.707,00
D.P.	28,52	19,39	32,20	45,03	53,78	47,32
C.V.	0,61	0,41	0,68	0,96	1,14	1,01

A distribuição mais uniforme para o programa de Interpolação, tabela 3.7, foi obtida para a Chave 5, com coeficiente de variação de 1,41 e com porcentagem de utilização do processador com menor carga de 96,7% em relação ao de maior carga. A Chave 2 apresentou a pior distribuição de carga com coeficiente de variação de 2,35 e com porcentagem de utilização do processador menos utilizado em 94,2% em relação ao mais utilizado.

P0 e P4 foram os processadores com maior e menor carga, respectivamente, para a Chave 5. Para a Chave 2 os processadores P0 e P4 também apresentaram a maior e menor carga.

TABELA 3.7 - Distribuição de Carga para Interpolação - TDM

Distribuição de Carga						
	Chave 1	Chave 2	Chave 3	Chave 4	Chave 5	Chave 6
P0	24.382	24.605	24.529	24.519	24.314	24.119
P1	24.376	24.378	24.441	24.374	24.216	24.466
P2	24.185	24.075	24.040	24.115	24.136	23.980
P3	23.730	23.700	23.738	23.620	23.756	23.903
P4	23.266	23.181	23.191	23.311	23.517	23.471
Média	23.987,80	23.987,80	23.987,80	23.987,80	23.987,80	23.987,80
D.P.	482,81	564,48	547,34	510,11	337,58	360,63
C.V.	2,01	2,35	2,28	2,13	1,41	1,50

A tabela 3.8 mostra os resultados da distribuição de carga para o programa Laplace. Para esse algoritmo as distribuições foram bastante dispersas em relação aos resultados anteriores. Para a Chave 6 obteve-se o menor coeficiente de variação com valor de 9,44 e o valor maior foi obtido para a Chave 5 com coeficiente de variação de 15,6.

A porcentagem de utilização do processador com menor carga em relação ao processador que mais trabalhou foi de 79,4% para a Chave 6 e 70,0% para a Chave 5. O par de processadores com maior e menor carga para a Chave 6 foi P0 e P4, respectivamente. Para a Chave 5 foi o par P0 e P3.

Outro programa que apresentou resultados com distribuição bastante dispersa, com altos valores de coeficiente de variação foi Multiplicação de Matrizes, cujos resultados são mostrados na tabela 3.9. Para todas as chaves a distribuição resultante é bastante dispersa, com maior carga para os primeiros processadores.

A Chave 6 apresentou menor coeficiente de variação para esse programa (23,01) e a taxa de utilização do processador menos utilizado em relação ao mais utilizado foi de 59,2%, enquanto que a Chave 5 resultou o maior coeficiente de variação, igual a 29,22 e taxa de utilização do processador menos utilizado de 53,3%. Tanto para a Chave 6 quanto para a Chave 5 o processador que mais trabalhou foi o P0 e o que menos trabalhou foi o P4. Esse foi o pior resultado, apresentando os maiores coeficientes de variação.

TABELA 3.8 - Distribuição de Carga para Laplace - TDM

Distribuição de Carga						
	Chave 1	Chave 2	Chave 3	Chave 4	Chave 5	Chave 6
P0	2.226	2.127	2.123	2.231	2.344	2.117
P1	2.039	1.958	2.092	1.980	1.899	1.936
P2	1.825	1.818	1.808	1.841	1.784	1.855
P3	1.702	1.755	1.671	1.673	1.640	1.720
P4	1.517	1.651	1.615	1.584	1.642	1.681
Média	1.861,80	1.861,80	1.861,80	1.861,80	1.861,80	1.861,80
D.P.	278,31	185,30	235,28	256,68	290,46	175,69
C.V.	14,95	9,95	12,64	13,79	15,60	9,44

TABELA 3.9 - Distribuição de Carga para Multiplicação de Matrizes - TDM

Distribuição de Carga						
	Chave 1	Chave 2	Chave 3	Chave 4	Chave 5	Chave 6
P0	18.204	18.042	19.039	19.238	19.358	18.092
P1	15.280	14.216	14.190	13.622	13.548	13.116
P2	11.586	11.615	11.198	11.512	11.295	11.565
P3	10.451	10.925	10.625	10.555	10.406	11.438
P4	9.399	10.122	9.868	9.993	10.313	10.709
Média	12.984,00	12.984,00	12.984,00	12.984,00	12.984,00	12.984,00
D.P.	3.665,47	3.217,37	3.761,04	3.759,40	3.794,04	2.986,99
C.V.	28,23	24,78	28,97	28,95	29,22	23,01

Para o programa N-Queens, tabela 3.10, a melhor distribuição foi obtida com a Chave 6: coeficiente de variação de 0,36. A pior distribuição com a Chave 1: coeficiente de variação de 1,79. A utilização do processador com menor carga foi de 99,1% em relação ao de maior carga para a Chave 6. Para a Chave 1 foi de 95,7%. Para a Chave 6 o processador mais utilizado foi o P2 e o menos utilizado foi o P4, para a Chave 1 o processador P1 foi o mais utilizado e o processador P4 o menos utilizado.

TABELA 3.10 - Distribuição de Carga para N-Queens - TDM

Distribuição de Carga						
	Chave 1	Chave 2	Chave 3	Chave 4	Chave 5	Chave 6
P0	39.838	40.028	40.032	39.250	39.657	39.328
P1	39.933	39.672	39.631	39.833	39.732	39.498
P2	39.705	39.530	39.553	39.802	39.396	39.540
P3	39.197	39.194	39.022	39.232	39.183	39.352
P4	38.229	38.478	38.664	38.785	38.934	39.184
Média	39.380,40	39.380,40	39.380,40	39.380,40	39.380,40	39.380,40
D.P.	703,47	586,49	538,25	440,49	331,14	142,68
C.V.	1,79	1,49	1,37	1,12	0,84	0,36

A tabela 3.11 mostra os resultados da distribuição para o programa Quick Sort. A distribuição mais uniforme foi obtida pela Chave 5 com coeficiente de variação de 0,68. A execução com a Chave 1 resultou na distribuição mais dispersa com coeficiente de variação de 2,29. A taxa de utilização do processador de menor carga para o de maior carga foi de 98,3% para a Chave 5 e de 94,6% para a Chave 1. Para a Chave 5 o processador P2 foi o que mais trabalhou, enquanto o processador P4 foi o menos utilizado. Também para a Chave 1, o processador mais utilizado foi o P2, já o que teve menor carga foi o P3.

Assim como Laplace e Multiplicação de Matrizes, o programa Simple também apresentou resultados altos para o coeficiente de variação em relação aos demais programas. Para esse programa a Chave 6 propiciou o menor coeficiente de variação

com valor igual a 6,79 e a Chave 1 o maior coeficiente com valor igual a 8,61. Para a Chave 6 o processador com menor carga trabalhou 84,7% em relação ao processador com maior carga. Para a Chave 1 essa taxa de utilização foi de 80,9%. Para a Chave 6 o processador P0 foi o que trabalhou mais e o processador P4 foi o que trabalhou menos. Para a Chave 1 o mesmo par de processadores correspondeu à maior e à menor utilização.

TABELA 3.11 - Distribuição de Carga para Quick Sort - TDM

Distribuição de Carga						
	Chave 1	Chave 2	Chave 3	Chave 4	Chave 5	Chave 6
P0	4.834	4.830	4.873	4.786	4.831	4.916
P1	4.875	4.929	4.835	4.810	4.810	4.797
P2	4.978	4.779	4.895	4.852	4.860	4.852
P3	4.711	4.897	4.780	4.894	4.846	4.831
P4	4.725	4.688	4.740	4.781	4.776	4.727
Média	4.824,60	4.824,60	4.824,60	4.824,60	4.824,60	4.824,60
D.P.	110,66	96,06	64,33	47,88	32,89	69,70
C.V.	2,29	1,99	1,33	0,99	0,68	1,44

TABELA 3.12 - Distribuição de Carga para Simple - TDM

Distribuição de Carga						
	Chave 1	Chave 2	Chave 3	Chave 4	Chave 5	Chave 6
P0	63.122	62.182	62.916	63.314	63.875	62.998
P1	59.896	59.392	58.964	58.315	57.956	57.316
P2	55.880	56.082	55.547	55.491	55.024	55.460
P3	53.334	53.686	53.939	53.588	53.714	54.145
P4	51.065	51.955	51.931	52.589	52.728	53.378
Média	56.659,40	56.659,40	56.659,40	56.659,40	56.659,40	56.659,40
D.P.	4.878,62	4.161,49	4.343,16	4.312,53	4.487,53	3.845,04
C.V.	8,61	7,34	7,67	7,61	7,92	6,79

### 3.4.6.2 Distribuição de Carga para a Arquitetura APO para o protocolo com Nó Central

Os resultados das tabelas anteriores são referentes ao protocolo de controle TDM. As tabelas de 3.13 a 3.21 mostram os resultados dos mesmos experimentos empregando o protocolo com Nó Central.

O programa Ackermann, tabela 3.13, apresentou distribuição mais uniforme para a Chave 6. As Chaves 1, 3 e 4 apresentaram o mesmo resultado com coeficiente de variação 2,93 sendo este o pior caso de distribuição para este programa. Para a Chave 6, a taxa de utilização do processador de menor carga em relação ao de maior carga foi de 97,1% e para as chaves com a pior distribuição essa taxa de utilização foi de 92,3%.

Para a distribuição mais uniforme, o processador com maior utilização foi o P3 e o de menor utilização foi o P4. Para o caso da distribuição mais dispersa o processador P2 teve a maior carga de trabalho e o processador P1 foi o que trabalhou menos.

TABELA 3.13 - Distribuição de Carga para Ackermann - Nó Central

Distribuição de Carga						
	Chave 1	Chave 2	Chave 3	Chave 4	Chave 5	Chave 6
P0	8.285	8.285	8.285	8.285	8.295	8.391
P1	8.030	8.116	8.030	8.030	8.266	8.338
P2	8.702	8.664	8.702	8.702	8.486	8.290
P3	8.315	8.387	8.315	8.315	8.323	8.399
P4	8.238	8.118	8.238	8.238	8.200	8.152
Média	8.314,00	8.314,00	8.314,00	8.314,00	8.314,00	8.314,00
D.P.	243,83	227,10	243,83	243,83	106,45	100,69
C.V.	2,93	2,73	2,93	2,93	1,28	1,21

A tabela 3.14 mostra os resultados de distribuição de carga para o programa Somatorial. Esse programa apresenta a melhor distribuição para a Chave 1, com coeficiente de variação de 0,39. A Chave 6 apresenta o pior resultado com uma distribuição bastante dispersa em relação às demais chaves. A porcentagem de utilização do processador com menor carga foi de 99,1% em relação ao processador com maior carga para a Chave 1. Para a Chave 6 foi de 86,4%. Para a Chave 1 o processador mais utilizado foi o P2 e o menos utilizado foi o P4. Para a Chave 6, o mais utilizado foi o P1 e o P4 o menos utilizado. Assim como para o protocolo TDM,

esse programa também foi o que apresentou as distribuições mais uniformes com os menores coeficientes de variação para quase todas as velocidades.

TABELA 3.14 - Distribuição de Carga para Somatorial - Nó Central

Distribuição de Carga						
	Chave 1	Chave 2	Chave 3	Chave 4	Chave 5	Chave 6
P0	57.824	57.631	58.122	58.390	57.905	58.069
P1	58.181	58.016	58.312	58.231	58.843	61.302
P2	58.285	58.064	58.175	57.337	57.965	59.943
P3	57.916	58.341	57.935	58.188	57.568	57.691
P4	57.779	57.933	57.441	57.839	57.704	52.980
Média	57.997,00	57.997,00	57.997,00	57.997,00	57.997,00	57.997,00
D.P.	224,06	255,56	338,94	420,19	498,77	3.161,45
C.V.	0,39	0,44	0,58	0,72	0,86	5,45

Os resultados para o programa Integração Binária apresentados na tabela 3.15, mostram que a distribuição mais uniforme ocorre para a Chave 2. A Chave 6 apresentou o pior resultado para a distribuição com coeficiente de variação de 8,65. Para a Chave 2, a taxa de utilização do processador de menor carga em relação ao de maior carga foi de 97,3%. Para a Chave 6 essa taxa foi de 79,7%. Os processadores P2 e P4 apresentaram a maior e a menor carga de trabalho para a Chave 2 respectivamente. Para a Chave 6 foram os processadores P1 e P4 respectivamente.

TABELA 3.15 - Distribuição de Carga para Integração Binária - Nó Central

Distribuição de Carga						
	Chave 1	Chave 2	Chave 3	Chave 4	Chave 5	Chave 6
P0	4.754	4.747	4.738	4.725	4.706	4.863
P1	4.718	4.690	4.706	4.789	4.753	5.076
P2	4.780	4.760	4.738	4.741	4.756	4.940
P3	4.699	4.706	4.758	4.679	4.715	4.612
P4	4.584	4.632	4.595	4.601	4.605	4.044
Média	4.707,00	4.707,00	4.707,00	4.707,00	4.707,00	4.707,00
D.P.	75,58	50,80	65,32	71,11	61,21	407,24
C.V.	1,61	1,08	1,39	1,51	1,30	8,65

A tabela 3.16 mostra que a Chave 2 proporcionou a melhor distribuição para o programa Interpolação com coeficiente de variação de 3,37. A Chave 6 apresentou a pior distribuição. O processador com menor taxa de utilização trabalhou 91,9% em relação ao processador com maior taxa, para a Chave 2. Para a Chave 6 essa taxa de utilização foi de 76,7%. Para essas duas chaves, os processadores P1 e P4 apresentaram a maior e a menor utilização respectivamente.

TABELA 3.16 - Distribuição de Carga para Interpolação - Nó Central

Distribuição de Carga						
	Chave 1	Chave 2	Chave 3	Chave 4	Chave 5	Chave 6
P0	24.261	24.236	24.509	24.444	24.381	25.732
P1	25.033	24.672	25.156	24.625	25.290	26.635
P2	24.448	24.554	24.108	24.534	23.995	24.512
P3	23.657	23.802	23.676	23.632	23.698	22.640
P4	22.540	22.675	22.490	22.704	22.575	20.420
Média	23.987,80	23.987,80	23.987,80	23.987,80	23.987,80	23.987,80
D.P.	946,63	807,51	998,93	819,64	991,26	2.493,51
C.V.	3,95	3,37	4,16	3,42	4,13	10,39

Os resultados da distribuição de carga para o programa Laplace são mostrados na tabela 3.17. Para esse algoritmo as distribuições foram bastante dispersas. Para a Chave 2 obteve-se o menor coeficiente de variação com valor de 12,4. O maior valor foi obtido para a Chave 6 com coeficiente de variação de 20,11. A porcentagem de utilização do processador com menor carga em relação ao processador que mais trabalhou foi de 76,8% para a Chave 2 e 61,7% para a Chave 6. O processador P0 foi o mais utilizado e o P4 o menos utilizado, tanto para a Chave 2 quanto para a Chave 6.

O programa Multiplicação de Matrizes apresentou uma distribuição bastante dispersa para todas as chaves com altos valores de coeficiente de variação. Esses valores são mostrados na tabela 3.18. A Chave 4 apresentou o menor coeficiente de variação (27,54) e a taxa de utilização do processador menos utilizado em relação ao

mais utilizado foi de 51,5%. Enquanto que a Chave 3 apresentou o maior coeficiente de variação (34,19) e taxa de utilização do processador menos utilizado de 42,9% em relação ao processador mais utilizado. Tanto para a Chave 4 quanto para a Chave 3 o processador P0 foi o mais utilizado e o menos utilizado foi o P4. Esse programa de teste foi o que apresentou as distribuições mais dispersas, com os maiores coeficientes de variação, sendo o pior caso para o protocolo de controle central.

TABELA 3.17 - Distribuição de Carga para Laplace - Nó Central

Distribuição de Carga						
	Chave 1	Chave 2	Chave 3	Chave 4	Chave 5	Chave 6
P0	2.080	2.116	2.249	2.248	2.171	2.343
P1	2.182	2.091	2.196	2.165	2.148	2.149
P2	1.926	1.808	1.821	1.798	1.810	1.762
P3	1.638	1.669	1.523	1.606	1.630	1.609
P4	1.483	1.625	1.520	1.492	1.550	1.446
Média	1.861,80	1.861,80	1.861,80	1.861,80	1.861,80	1.861,80
D.P.	294,89	230,92	351,74	334,42	287,73	374,43
C.V.	15,84	12,40	18,89	17,96	15,45	20,11

TABELA 3.18 - Distribuição de Carga para Multiplicação de Matrizes - Nó Central

Distribuição de Carga						
	Chave 1	Chave 2	Chave 3	Chave 4	Chave 5	Chave 6
P0	17.503	18.290	19.122	17.717	18.222	19.548
P1	15.978	16.143	15.706	15.574	15.569	14.904
P2	12.154	11.998	12.083	12.004	11.786	11.333
P3	10.156	9.759	9.803	10.509	10.246	10.177
P4	9.129	8.730	8.206	9.116	9.097	8.958
Média	12.984,00	12.984,00	12.984,00	12.984,00	12.984,00	12.984,00
D.P.	3.637,75	4.111,03	4.439,71	3.576,20	3.813,12	4.289,12
C.V.	28,02	31,66	34,19	27,54	29,37	33,03

A tabela 3.19 mostra os resultados para o programa N-Queens. A melhor distribuição foi obtida com a Chave 4, com coeficiente de variação de 2,29 e a pior distribuição com a Chave 6 com coeficiente de variação de 7,89. A utilização do processador com menor carga foi de 94,9% em relação ao de maior carga para a

Chave 4 e para a Chave 6 foi de 81,7%. P1 é o processador mais utilizado e P4 é o menos utilizado para essas duas chaves.

TABELA 3.19 - Distribuição de Carga para N-Queens - Nó Central

Distribuição de Carga						
	Chave 1	Chave 2	Chave 3	Chave 4	Chave 5	Chave 6
P0	39.809	40.033	39.915	40.166	40.073	40.924
P1	40.672	40.260	40.687	40.167	40.223	42.484
P2	39.882	40.003	39.864	39.651	39.828	40.924
P3	38.788	38.901	39.042	38.804	39.067	37.874
P4	37.751	37.705	37.394	38.114	37.711	34.696
Média	39.380,40	39.380,40	39.380,40	39.380,40	39.380,40	39.380,40
D.P.	1.130,20	1.075,17	1.253,71	900,71	1.033,82	3.106,87
C.V.	2,87	2,73	3,18	2,29	2,63	7,89

Os resultados da distribuição de carga para o programa Quick Sort são mostrados na tabela 3.20. A distribuição mais uniforme foi obtida utilizando-se a Chave 4 com coeficiente de variação de 1,70. A execução com a Chave 6 resultou na distribuição mais dispersa com coeficiente de variação de 5,71. A taxa de utilização do processador de menor carga com relação ao de maior carga foi de 95,9% para a Chave 4 e de 86,3% para a Chave 6. O processador mais utilizado para a Chave 4 foi o P0 e para a Chave 6 foi o P1. Para as duas chaves o processador que menos trabalhou foi o P4.

TABELA 3.20 - Distribuição de Carga para Quick Sort - Nó Central

Distribuição de Carga						
	Chave 1	Chave 2	Chave 3	Chave 4	Chave 5	Chave 6
P0	4.879	4.947	4.910	4.892	4.879	4.893
P1	4.985	4.927	4.954	4.885	4.901	5.072
P2	4.905	4.905	4.889	4.818	4.915	5.013
P3	4.716	4.724	4.706	4.839	4.711	4.766
P4	4.638	4.620	4.664	4.689	4.717	4.379
Média	4.824,60	4.824,60	4.824,60	4.824,60	4.824,60	4.824,60
D.P.	142,97	144,84	130,43	81,90	101,80	275,40
C.V.	2,96	3,00	2,70	1,70	2,11	5,71

A tabela 3.21 mostra os resultados para o programa Simple, que também apresentou resultados altos para o coeficiente de variação. Para esse programa a Chave 2 apresentou o menor coeficiente de variação com valor igual a 8,44. A Chave 6 apresentou o maior coeficiente com valor igual a 12,9. O processador com menor carga trabalhou 82,0% em relação ao de maior carga, para a Chave 2. Para a Chave 6 essa taxa de utilização foi de 73,6%. Para a Chave 2 o processador P0 teve maior utilização e para a Chave 6 o mais utilizado foi o P1. O processador P4 foi que menos trabalhou para as duas chaves.

TABELA 3.21 - Distribuição de Carga para Simple - Nó Central

Distribuição de Carga						
	Chave 1	Chave 2	Chave 3	Chave 4	Chave 5	Chave 6
P0	61.499	61.349	61.842	61.872	61.983	63.573
P1	62.179	61.144	61.679	61.041	61.239	63.746
P2	57.411	56.803	56.820	56.783	56.932	56.962
P3	53.131	53.716	53.102	53.221	52.884	52.077
P4	49.077	50.285	49.854	50.380	50.259	46.939
Média	56.659,40	56.659,40	56.659,40	56.659,40	56.659,40	56.659,40
D.P.	5.576,63	4.780,72	5.269,02	4.940,61	5.113,87	7.307,43
C.V.	9,84	8,44	9,30	8,72	9,03	12,90

### 3.5 Conclusões

A investigação das distribuições de carga mostra que os modelos propostos para a chave de interconexão da arquitetura APO proporcionam uma distribuição muito mais homogênea que o modelo adotado para a chave de distribuição da arquitetura Wolf. As chaves da arquitetura APO não apresentaram sobrecarga de processadores como no caso da arquitetura Wolf, em que, os primeiros processadores são muito mais utilizados que os últimos. Por exemplo, para o melhor caso de distribuição da arquitetura APO, o coeficiente de variação é 0,32 e com uma taxa de utilização

de 99,3% para o processador que foi menos utilizado em relação ao que trabalhou mais.

Entre as metodologias adotadas para a arquitetura APO, o modelo TDM apresentou melhor distribuição que o modelo com controlador central. Por exemplo, para os programas de teste Integração Binária, Gauss, Interpolação, N-Queens, Quick Sort e Simple o modelo TDM apresentou coeficientes de variação menores que os obtidos com o modelo de Nó Central para todas as velocidades. Para os programas Somatorial e Laplace o modelo TDM proporcionou melhores resultados para cinco velocidades e para os programas Ackermann e Multiplicação de Matrizes foi melhor para quatro velocidades. Essa melhor distribuição é devido à maneira como as fichas são distribuídas para os EPs: as MDs tentam enviar fichas para os EPs e caso estes não estejam disponíveis, elas tentam enviar para o próximo EP no ciclo de controle seguinte.

No caso do Nó Central, esse resultado provavelmente decorre da forma como as MDs enviam suas fichas para os EPs. Nesse modelo, as MDs quando tentam transmitir para os EPs e não recebem permissão para transmissão, elas esperam o próximo ciclo de controle do Nó Central para tentarem transmitir para o mesmo EP requisitado no ciclo anterior.

A variação de velocidade da chave não influi de maneira uniforme sobre a distribuição de carga. Não se obteve uma relação entre a distribuição de carga e a variação da velocidade das chaves. Por exemplo, para o algoritmo de Laplace o coeficiente de variação para a Chave 2 é menor que o coeficiente de variação para a Chave 3, tanto para o protocolo de controle TDM quanto para o controle com Nó Central. Para a Chave 4 o coeficiente é menor que para a Chave 5 no protocolo

TDM, enquanto que o coeficiente para a Chave 4 é maior que o da Chave 5 para o controle com Nó Central.

Da mesma forma como identificado no trabalho de mestrado [5], **não há, portanto, uma regra de comportamento para as distribuições em função da variação das velocidades de transmissão.** Esse é um problema sem solução única, existem soluções melhores e piores. Para alguns algoritmos a distribuição é mais sensível às variações da velocidade que para outros. Por exemplo, para o modelo de controle TDM, o programa Simple apresentou o menor coeficiente de variação para a Chave 6 e o maior para a Chave 1. O coeficiente de variação da Chave 1 é aproximadamente 26,8% maior que o da Chave 6. Para as demais chaves as diferenças são menores. Já para o programa Quick Sort, o coeficiente de variação para a pior distribuição é aproximadamente 236,8% maior que o coeficiente de variação para a melhor distribuição.

## **Capítulo 4**

### **Comportamento da Chave: Tempos de Execução**

#### **4.1 Introdução**

Este capítulo apresenta os resultados dos tempos de execuções dos programas de teste em função das chaves de interconexão da arquitetura APO. Esse estudo é importante para permitir o balanceamento da velocidade da chave em relação à velocidade das demais unidades. Essa investigação possibilita detectar se há limitantes inferiores e superiores para a velocidade da chave em função dos tempos de execução.

A segunda seção apresenta uma proposta para a classificação do desempenho do sistema, a terceira seção mostra os resultados para o protocolo de controle TDM e a quarta seção mostra os resultados para o protocolo de controle com Nó Central. Esses resultados mostram que a metodologia de controle com Nó Central é mais adequada, por ser menos sensível à variação de desempenho em função da velocidade.

## **4.2 Classificação do desempenho do sistema**

Esta seção propõe uma metodologia para classificação do desempenho da arquitetura em função do tempo de execução. Esta classificação facilita a identificação dos limites de velocidade da chave, isto é, até que ponto pode ser vantajoso aumentar a velocidade da chave em função do comportamento do sistema. A classificação é baseada na porcentagem de redução do tempo de execução em função do aumento da velocidade da chave. Por exemplo, na mudança da Chave 1 para a Chave 2, aumenta-se a velocidade de transmissão em 100% (de 2,5 Gb/s para 5,0 Gb/s), que no caso ideal, proporcionaria uma redução linear correspondente sobre o tempo de execução. Esta classificação define três categorias: desempenho inferior, desempenho linear e desempenho ótimo.

Os critérios de classificação são os seguintes:

- Desempenho inferior: quando a taxa de redução do tempo de execução é inferior a 45% para um aumento de 100% na velocidade da chave.
- Desempenho linear: quando a taxa de redução do tempo de execução está na faixa entre 45% e 50%, para um aumento de 100% na velocidade da chave.
- Desempenho ótimo: quando a taxa de redução do tempo de execução é superior a 50% para um aumento de 100% na velocidade da chave.

## **4.3 Tempos de execução para a chave com controle TDM**

As figuras de 4.1 a 4.10 mostram os tempos de execução para a chave com protocolo de controle TDM em função das variações de velocidade da chave.

O eixo das ordenadas representa as velocidades denominadas de Chave 1 até Chave 6 e o eixo das abscissas representa o tempo de execução<sup>1</sup> simulado em função da velocidade da chave, usando a mesma metodologia descrita na seção 3.3.

A figura 4.1 mostra os tempos de execução para o programa Ackermann. A redução do tempo de execução foi mais significativa para as Chaves 2 e 3. Apenas para a mudança da Chave 1 para a Chave 2 o desempenho foi próximo ao desempenho do tipo linear. Nessa mudança houve uma redução de 44,9% no tempo de execução e na mudança da Chave 2 para a Chave 3 a redução foi de 40,5%. Para as velocidades maiores o desempenho foi menos significativo. Para todas as velocidades o desempenho foi categorizado como desempenho do tipo inferior. O programa de Ackermann apresentou os melhores resultados de redução de tempo de execução entre todos os programas de teste executados para o modelo de controle TDM.

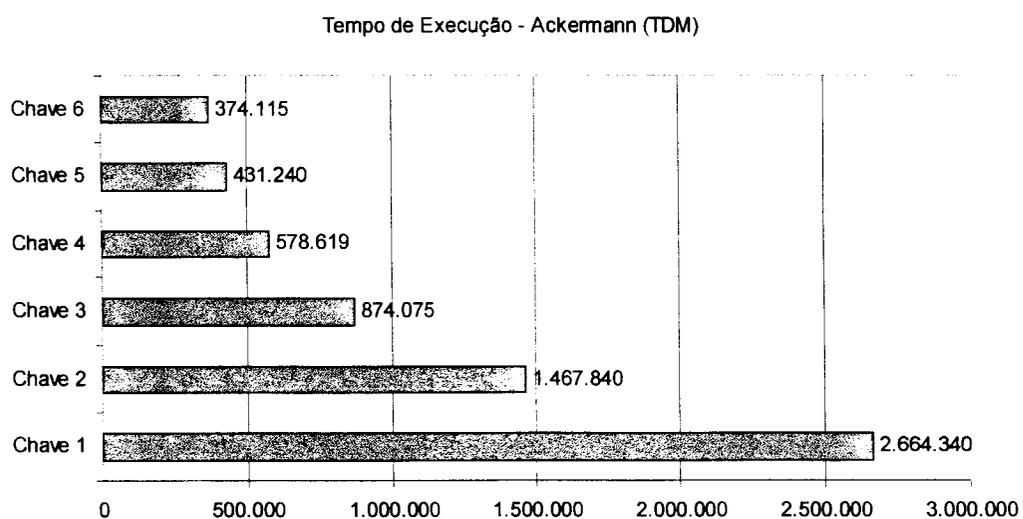


FIGURA 4.1 - Tempo de execução para chave com controle TDM - Ackermann

<sup>1</sup> A unidade de tempo de execução é denominada de *time step* conforme apresentado na seção 2.2.2, e é utilizada como unidade do eixo das abscissas para todos os gráficos deste capítulo.

Os resultados dos tempos de execução para o programa Somatorial são mostrados na figura 4.2. Para esse programa a maior redução no tempo de execução ocorre na mudança da Chave 1 para a Chave 2 com melhoria de 41,9% no tempo de execução. As outras velocidades apresentaram desempenho menor. Para esse programa o desempenho também foi do tipo inferior para todas as velocidades. A execução com a Chave 6 apresentou um comportamento anômalo, com tempo de execução maior que o da Chave 5. Isso mostra que o sistema a fluxo de dados estudado não é linear, com várias realimentações complexas. Nesse tipo de sistema o fluxo de execução do programa não é linear, havendo muitas combinações de situações que tornam o comportamento do sistema bastante complexo. Por exemplo, há situações em que uma única ficha pode gerar muitas outras fichas, ou ainda duas fichas geram apenas uma ficha de saída; há também trechos de execução estritamente seqüenciais em que a execução de uma instrução depende única e exclusivamente da execução da instrução anterior. Mudando-se a velocidade da chave, combinações diferentes dessas características podem ocorrer, modificando muito o comportamento do sistema.

Esse resultado mostra também que a velocidade da chave não pode ser arbitrariamente aumentada, pois o desbalanceamento grande do sistema termina por comprometer qualquer ganho que a chave mais veloz pudesse trazer.

A figura 4.3 mostra os resultados dos tempos de execução para o programa Integração Binária. Esse algoritmo apresentou maior redução do tempos de execução para a Chave 2 com 42,0% e também mostrou um comportamento anômalo em relação à Chave 6 com tempo de execução maior do que com a Chave 5. Para todas as velocidades o sistema teve desempenho do tipo inferior.

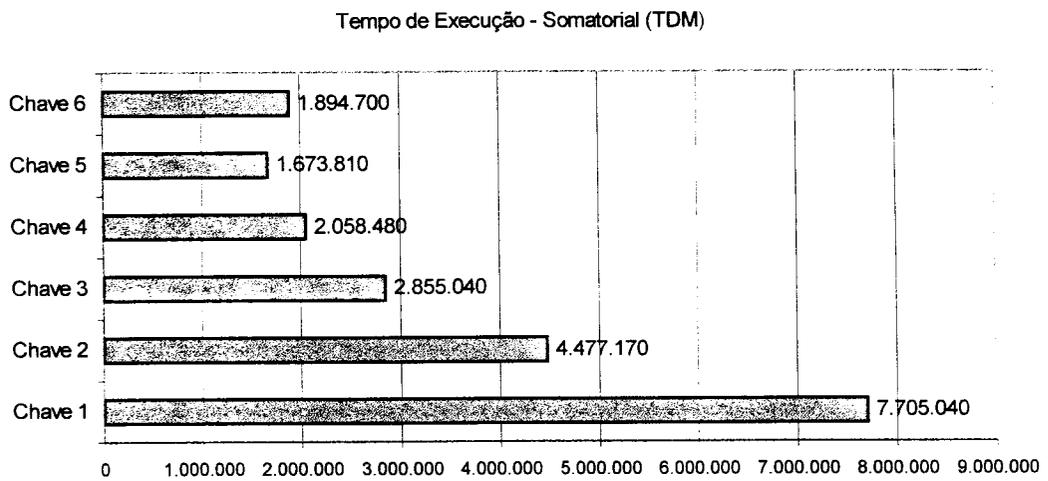


FIGURA 4.2 - Tempo de execução para chave com controle TDM - Somatorial

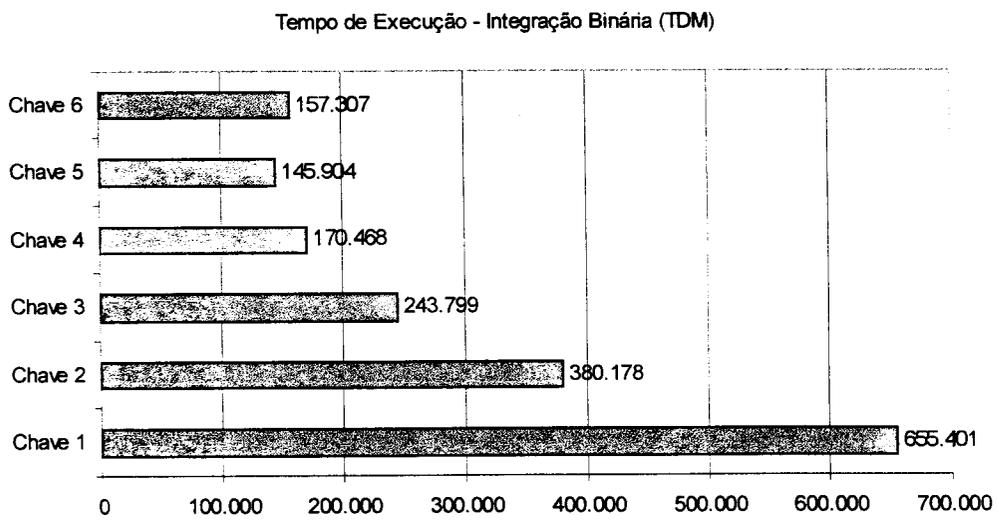


FIGURA 4.3 - Tempo de execução para chave com controle TDM - Integração Binária

A figura 4.4 mostra os resultados dos tempos de execução para o programa de Gauss. A mudança da Chave 1 para a Chave 2 resultou na maior redução para o tempo de execução com 42,9%. A mudança de velocidade da Chave 5 para a Chave 6 resultou numa melhoria de apenas 9,1% no tempo de execução. Esses resultados indicam que o desempenho foi do tipo inferior para esse programa.

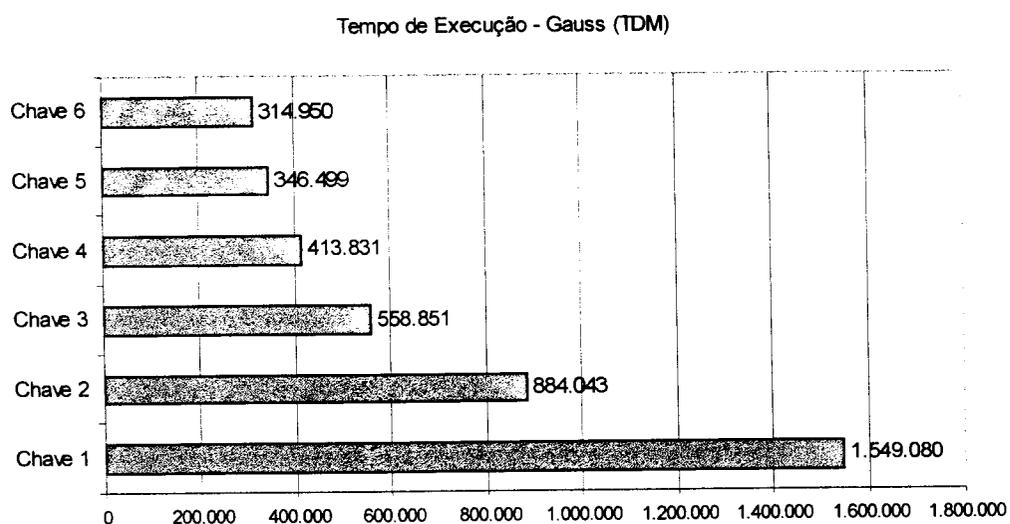


FIGURA 4.4 - Tempo de execução para chave com controle TDM - Gauss

Para o programa Interpolação (figura 4.5) a Chave 2 apresentou uma redução de 41,8% em relação ao tempo obtido com a Chave 1 e a Chave 6 reduziu em 11,1% o tempo em relação à Chave 5. Desempenho do tipo inferior também para esse programa.

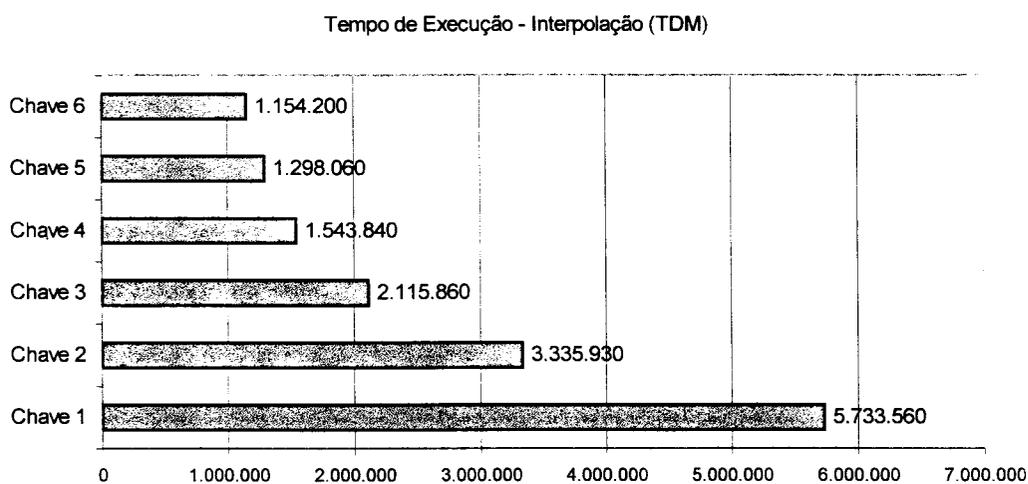


FIGURA 4.5 - Tempo de execução para chave com controle TDM - Interpolação

Os resultados para o programa Laplace são mostrados na figura 4.6. A redução mais significativa no tempo de execução ocorre na mudança da Chave 1 para a Chave 2 com diminuição de 44,2% do tempo. Para as outras velocidades o desempenho é menor, sendo que a Chave 6 proporciona uma redução de 9,7% no tempo em relação à Chave 5. Com essas taxas de redução o desempenho para Laplace é categorizado como sendo do tipo inferior.

A figura 4.7 apresenta os resultados para o programa Multiplicação de Matrizes. Para esse programa o desempenho também é menor para as velocidades mais altas. E a redução maior no tempo de execução ocorre com a Chave 2 com 42,7%, implicando em desempenho do tipo inferior.

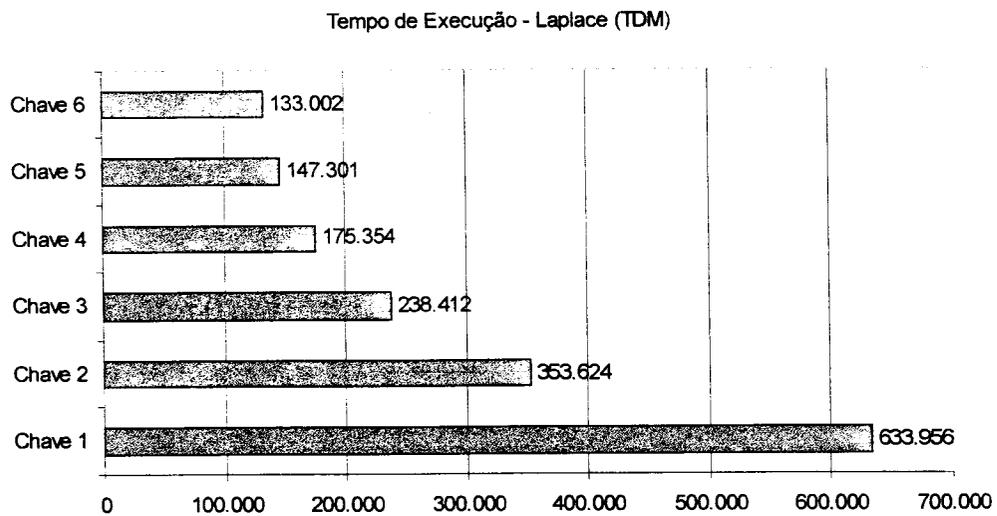


FIGURA 4.6 - Tempo de execução para chave com controle TDM - Laplace

O programa N-Queens (figura 4.8) assim como os programas Somatorial e Integração Binária também apresenta um resultado anômalo para a Chave 6 com um pequeno aumento no tempo de execução em relação à Chave 5. A maior redução no tempo de execução foi obtida na mudança da Chave 1 para a Chave 2 com 41,9% e o desempenho propiciado para todas as velocidades é classificado como desempenho do tipo inferior. É interessante notar que chaves com velocidade acima de 40 Gb/s não trazem benefícios em alguns programas como Somatorial, Integração Binária e N-Queens. Enquanto para outros há um pequeno benefício, como por exemplo para Gauss, Laplace e Simple.

Tempo de Execução - Multiplicação de Matrizes (TDM)

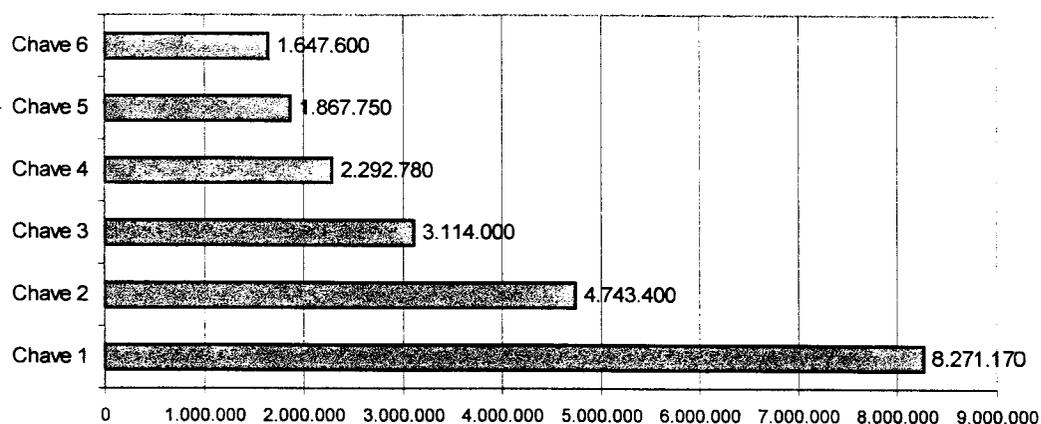


FIGURA 4.7 - Tempo de execução para chave com controle TDM - Multiplicação de Matrizes

Tempo de Execução - N-Queens (TDM)

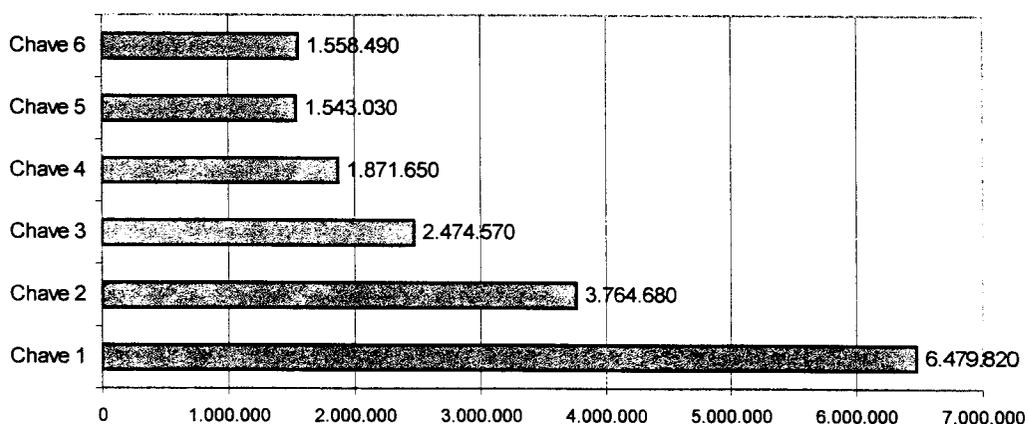


FIGURA 4.8 - Tempo de execução para chave com controle TDM - N-Queens

Os resultados para o programa Quick Sort são mostrados na figura 4.9. A redução maior no tempo de execução foi obtida com a Chave 2. As demais velocidades resultaram em porcentagens menores de redução, com a Chave 6 apresentando 12,0% de redução em relação à Chave 5. Para esse programa o desempenho também foi do tipo inferior.

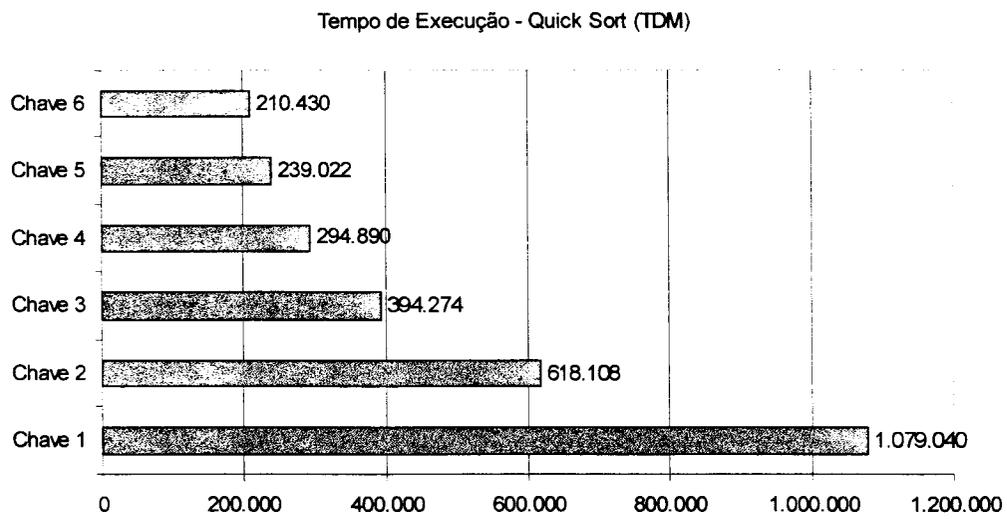


FIGURA 4.9 - Tempo de execução para chave com controle TDM - Quick Sort

A figura 4.10 mostra os resultados para o programa Simple. Com a Chave 2 obteve-se a maior porcentagem de redução no tempo de execução com 41,0%. A Chave 6 apresentou a menor redução no tempo com 8,3% em relação à Chave 5. As taxas de redução do tempo de execução para esse programa dão a classificação de desempenho do tipo inferior.

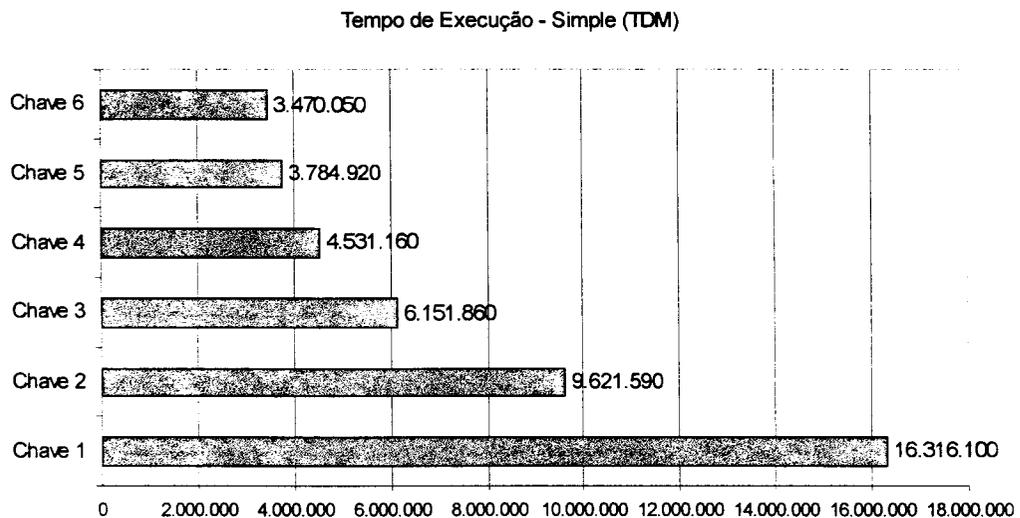


FIGURA 4.10 - Tempo de execução para chave com controle TDM - Simple

#### **4.4 Tempos de execução para a chave com Nó Central**

Assim como foram apresentados na seção 4.3 os resultados dos tempos de execução dos programas de teste para o protocolo de controle TDM, nesta seção as figuras de 4.11 a 4.20 mostram os tempos de execução para a chave com controlador central em função das variações de velocidade da chave. Aqui também o eixo das ordenadas representa as velocidades denominadas de Chave 1 até Chave 6, e o eixo das abcissas representa o tempo de execução simulado em função da velocidade da chave.

Os tempos de execução para o programa Ackermann são mostrados na figura 4.11. Para todas as velocidades houve uma redução significativa no tempo de execução em torno de 48%, sendo que a Chave 6 apresentou a maior porcentagem de redução com 50,9% em relação à Chave 5. Da Chave 1 até a Chave 4 o desempenho do sistema é classificado como linear. Na mudança para a Chave 5 o desempenho cai para o tipo inferior, sendo este o pior caso entre todos os programas para o protocolo de controle com Nó Central. Para a Chave 6 obteve-se desempenho do tipo ótimo.

A figura 4.12 mostra os tempos obtidos para o programa Somatorial. A maior redução no tempo de execução ocorreu para a Chave 6 com 52,1%, sendo este o melhor resultado entre todos os programas para o controle com Nó Central. A menor redução foi obtida com a Chave 5 com 46,1%. O desempenho é classificado como sendo do tipo linear da Chave 1 até a Chave 5 e tipo ótimo para a Chave 6.

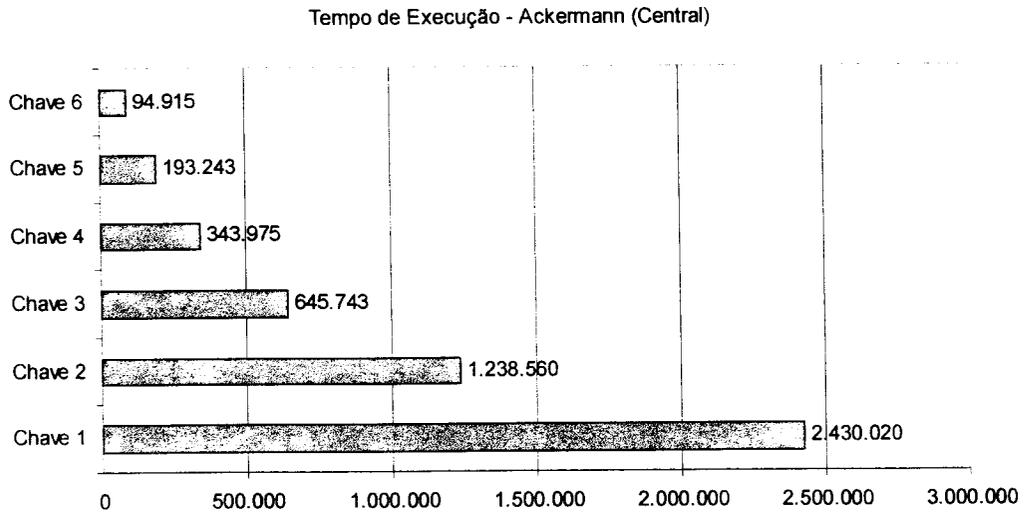


FIGURA 4.11 - Tempo de execução para chave com Nó Central - Ackermann

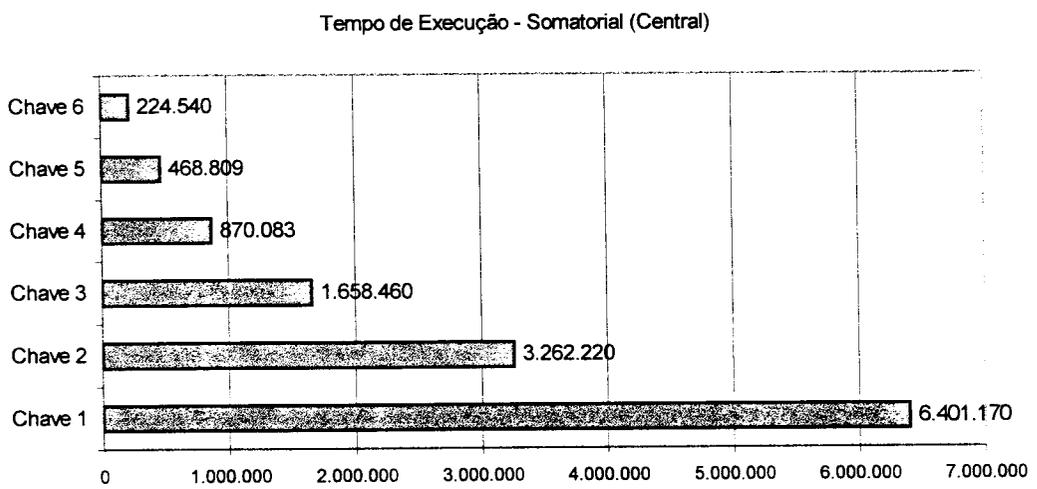


FIGURA 4.12 - Tempo de execução para chave com Nó Central - Somatorial

Para o programa Integração Binária (figura 4.13) o melhor desempenho foi obtido com a Chave 3 que reduziu em 50,5% o tempo de execução em relação à Chave 2. A menor redução do tempo de execução ocorreu para a Chave 5 com 44,5% em relação à Chave 4. As Chaves 2, 4 e 6 proporcionaram um desempenho do tipo linear, enquanto que para a Chave 3 o desempenho foi do tipo ótimo e para a Chave 5 o desempenho foi do tipo inferior.

Para o programa Gauss, os resultados da figura 4.14 mostram uma redução média do tempo de execução em torno de 47,8% para as velocidades da chave. Sendo que a Chave 3 apresentou a maior diminuição do tempo com percentagem de 49,7% em relação à Chave 2. Para todas as velocidades o desempenho obtido foi do tipo linear.

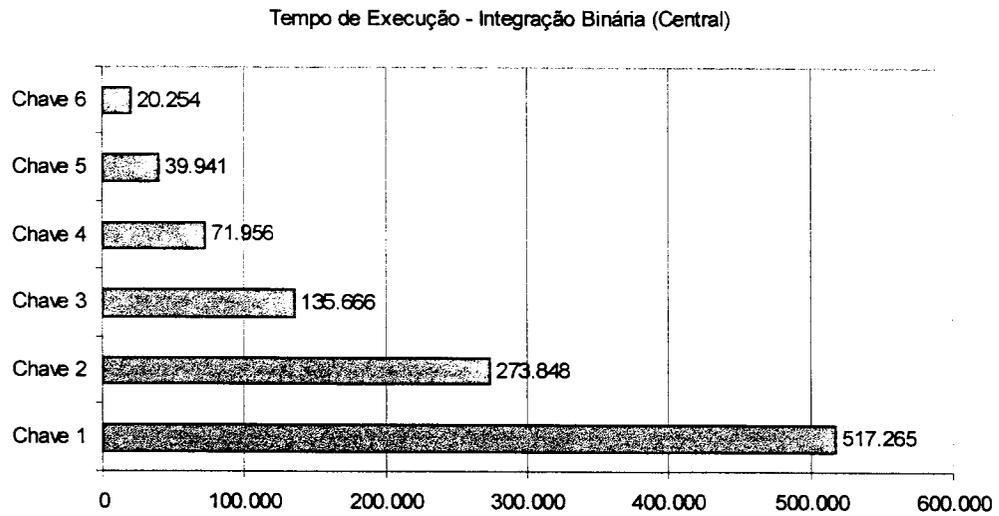


FIGURA 4.13 - Tempo de execução para chave com Nó Central - Integração Binária

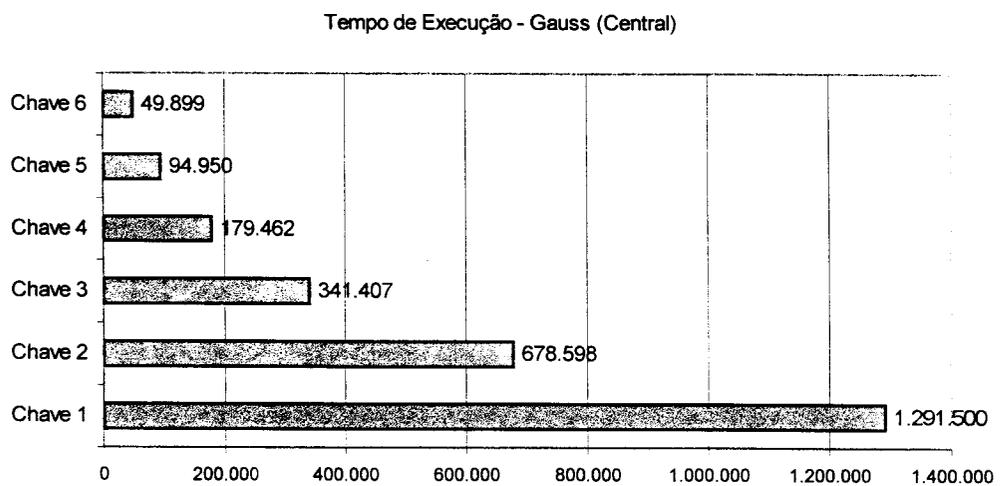


FIGURA 4.14 - Tempo de execução para chave com Nó Central - Gauss

O programa Interpolação com resultados apresentados pela figura 4.15, obteve a melhor redução do tempo de execução de 50,2% com a Chave 6 e a porcentagem menor ficou para a Chave 5 com 45,2%. Para esse programa, da Chave 1 até a Chave 5 o desempenho foi do tipo linear e para a Chave 6 o desempenho foi do tipo ótimo.

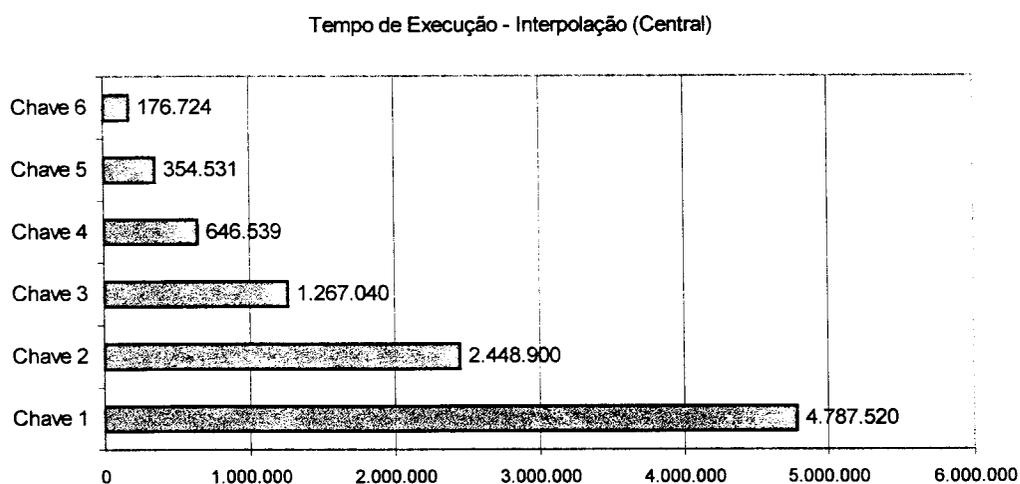


FIGURA 4.15 - Tempo de Execução para chave com Nó Central - Interpolação

Para o programa Laplace, a Chave 5 apresentou a maior redução do tempo de execução com 49,7% e em segundo lugar ficou a Chave 2 com 48,6% de redução do tempo. A Chave 4 foi a que resultou na menor redução com 46,2%. Para todas as chaves o desempenho obtido foi do tipo linear.

O programa Multiplicação de Matrizes obteve melhor desempenho para a Chave 6 que resultou numa diminuição de 49,9% do tempo de execução em relação à Chave 5. Outra diminuição significativa do tempo ocorreu para a Chave 4 com redução de 49,2% do tempo. A Chave 5 apresentou um desempenho do tipo inferior, com a menor redução do tempo entre todas as chaves. As demais chaves apresentaram desempenho do tipo linear.

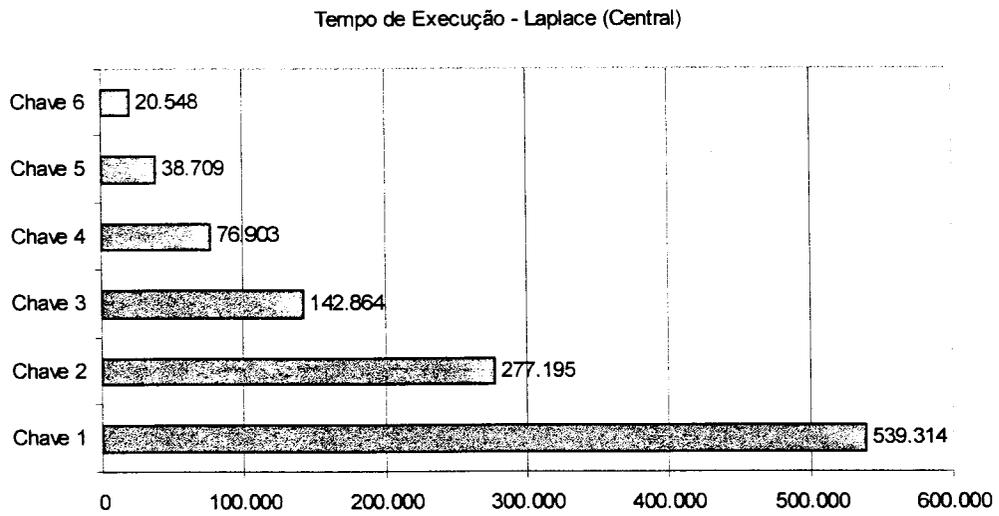


FIGURA 4.16 - Tempo de Execução para chave com Nó Central - Laplace

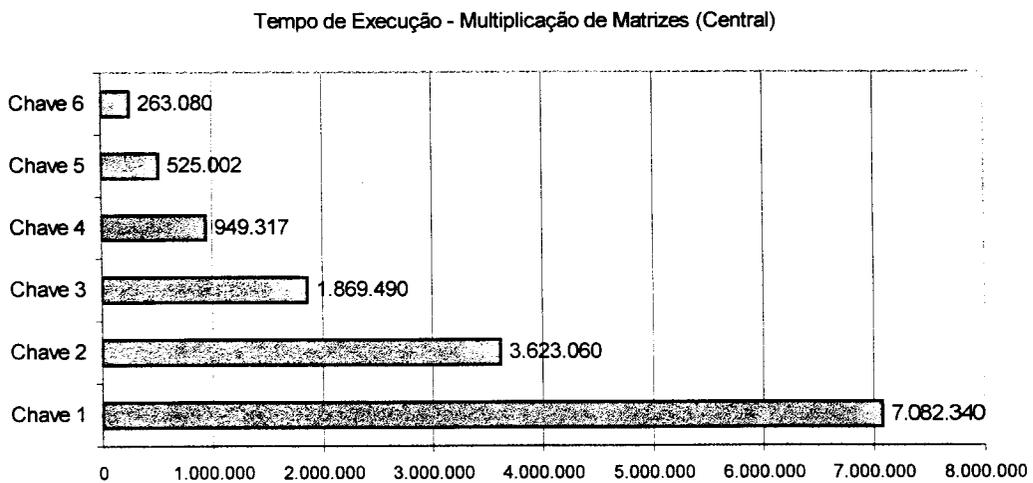


FIGURA 4.17 - Tempo de Execução para chave com Nó Central - Multiplicação de Matrizes

Para o programa N-Queens a Chave 2 proporcionou a maior diminuição do tempo de execução com redução de 49,7%. A menor redução do tempo foi obtida com a Chave 5 com 44,9%. Assim como para o programa Multiplicação de Matrizes, o desempenho foi do tipo linear para todas as chaves, exceto para a Chave 5 que proporcionou desempenho do tipo inferior.

A figura 4.19 mostra os tempos de execução para o programa Quick Sort. A Chave 2 foi a que apresentou a maior redução do tempo de execução com uma percentagem de diminuição de 51,7%. A Chave 5 apresentou a menor taxa de redução com 46,3%. Desempenho do tipo linear foi obtido com as Chaves 3, 4 e 5 e desempenho do tipo ótimo foi obtido com as Chaves 2 e 6.

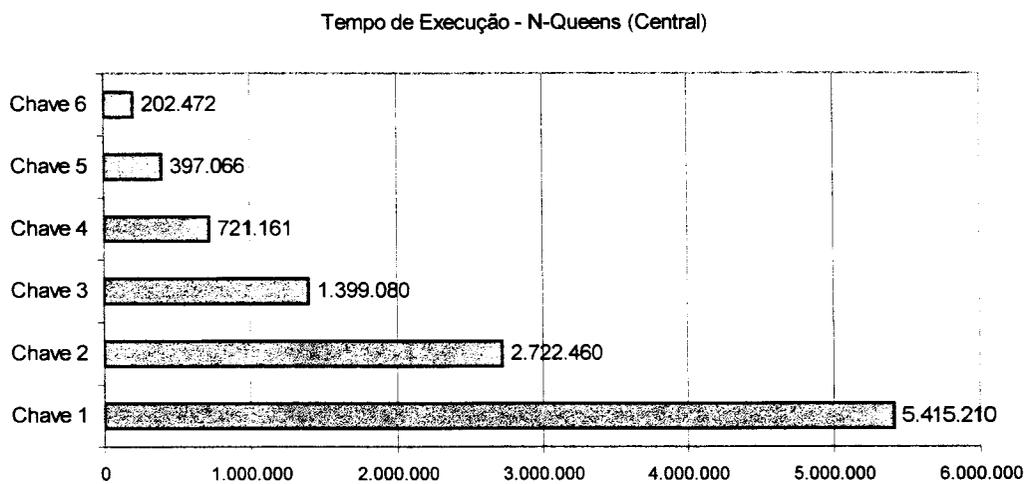


FIGURA 4.18 - Tempo de Execução para chave com Nó Central - N-Queens

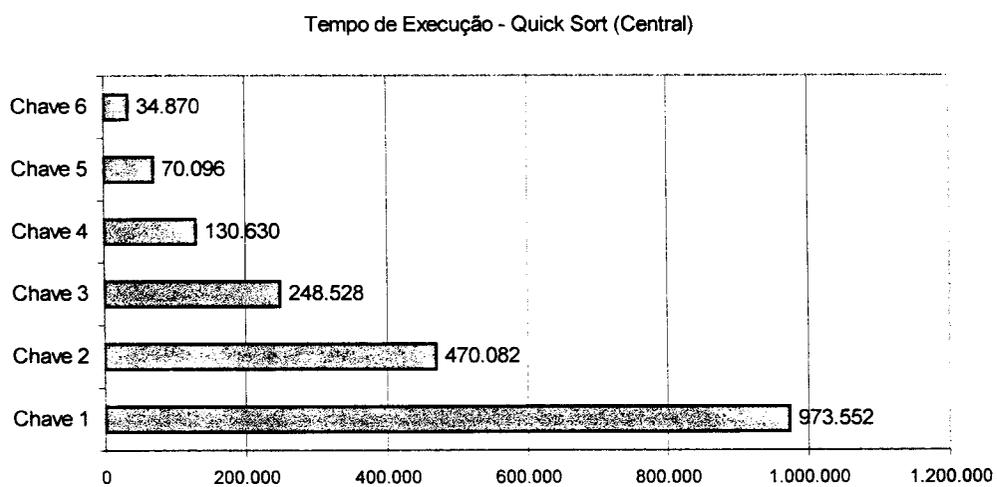


FIGURA 4.19 - Tempo de Execução para chave com Nó Central - Quick Sort

O programa Simple obteve a maior redução do tempo de execução com a Chave 6 com taxa de redução de 49,4%. E a menor redução do tempo ocorreu para a Chave 5: 46,3%. Para todas as chaves obteve-se desempenho do tipo linear.

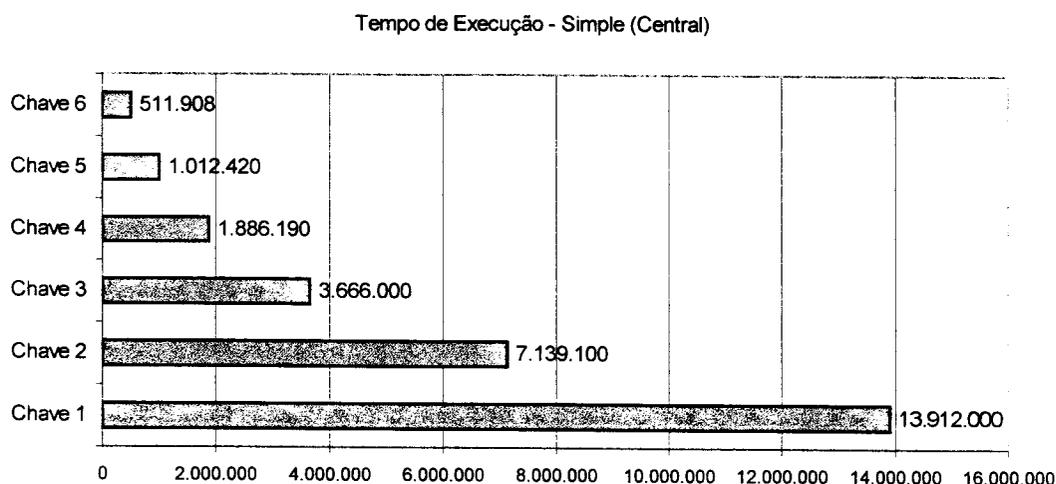


FIGURA 4.20 - Tempo de Execução para chave com Nó Central - Simple

É interessante notar que para o protocolo de controle Nó Central não ocorre o comportamento anômalo detectado para o protocolo de controle TDM para velocidade acima de 40 Gb/s. Isso significa que o protocolo de controle com Nó Central é mais adequado, visto que apresenta um comportamento uniforme para o desempenho em função da velocidade.

#### 4.5 Análise dos tempos de execução

Esta seção discute os resultados obtidos utilizando-se as chaves de interconexão da arquitetura APO. Primeiramente, discute-se os tempos obtidos para a chave com controle TDM e em seguida, os tempos para a chave com controle central.

#### 4.5.1 Resultados para a chave com controle TDM

Para a chave implementada com este modelo de controle obteve-se uma redução significativa do tempo de execução na primeira mudança de velocidade da chave, isto é, da Chave 1 para a Chave 2, em torno de 42% para todos os programas. A partir da Chave 2, a redução no tempo de execução foi cada vez menor, sendo que para a Chave 6 a redução foi em média em torno de 5%. Em alguns casos houve até um aumento no tempo de execução para a chave de maior velocidade.

O desempenho desse modelo não apresentou uma melhora na mesma proporção que o aumento de velocidade da chave. Isso é devido à estratégia de distribuição de fichas que ele implementa: cada nó espera pelo seu ciclo de controle, que se repete após todos os demais nós terem efetuado seus respectivos ciclos de controle.

O ciclo de controle tem um tempo de duração fixo independente da velocidade da chave. Assim, para taxas de transmissão maiores, o nó consome um tempo menor para transmissão do dado, mas continua dependente do tempo de controle que é fixo. Esse tempo de controle tem um peso maior à medida em que a taxa de transmissão é aumentada (velocidades maiores).

Por exemplo, para a chave com velocidade de 2,5 Gb/s o tempo para transmitir uma determinada ficha é de 48 ns. O tempo de controle de 1 ns representa cerca de 2% do tempo de transmissão da ficha. Para a chave com velocidade de 10 Gb/s o tempo para transmitir a mesma ficha é de 12 ns e o tempo de controle representa cerca de 8% do tempo de transmissão. Para a chave de 40 Gb/s o tempo de transmissão da ficha é de 3 ns e o tempo de controle representa cerca de 33% do tempo de transmissão da ficha.

Assim, mesmo com o aumento da velocidade da chave o tempo que cada nó espera para executar seu ciclo de controle é fixo, o que proporciona um desempenho menos satisfatório para velocidades maiores. Isso também gera casos em que ocorre aumento do tempo de execução para a chave mais veloz, como por exemplo para os programas Somatorial, Integração Binária e N-Queens. Nesse caso, o tempo para transmissão da ficha é de 0,3 ns e esse tempo corresponde a 30% do tempo de controle. **Isso implica que, aumentar progressivamente a velocidade da chave para valores bastante elevados, como por exemplo além de 40 Gb/s para esse modelo de controle, não resulta em desempenhos compatíveis com os aumentos de velocidade.**

Para todos os programas de teste, o desempenho obtido com o protocolo de controle TDM foi classificado como sendo do tipo inferior. A partir da Chave 4 os ganhos não foram significativos. As porcentagens de redução dos tempos de execução foram inferiores a 25%, exceto para o programa Ackermann (com 25,5%). Pode-se então delimitar uma fronteira superior para a velocidade da chave para o controle TDM. A Chave 4 pode ser esse limite superior, pois para velocidades maiores o desempenho é muito pobre. Um limitante inferior pode ser considerada a Chave 2 que é a que apresenta o maior ganho.

#### **4.5.2 Resultados para a chave com Nó Central**

A chave de interconexão implementada com o modelo de controlador central apresentou um desempenho uniforme para todas as velocidades de chave. A cada aumento de velocidade da chave o tempo de execução foi reduzido em torno de 48%.

Esse desempenho é devido ao gerenciamento dinâmico dos pedidos de transmissões dos nós. Pois, a cada ciclo de controle o nó central gerencia pedidos de transmissão. Um nó que deseja transmitir não tem de esperar que os demais executem seus ciclos de controle. Ele pode fazer um pedido assim que tiver ficha para transmitir. Assim, é possível que vários nós transmitam ao mesmo tempo. Proporcionando uma distribuição mais eficiente das fichas.

Esse protocolo também sofre influência do tempo de ciclo de controle assim como no protocolo TDM, mas com um peso menor sobre seu desempenho. Pois, para a chave de maior velocidade obteve-se aproximadamente a mesma redução no tempo de execução que para as outras velocidades menores. Assim, a velocidade da chave para esse modelo também não apresenta melhorias significativas a ponto de ser aumentada além de 40 Gb/s. O tempo do ciclo de controle apresenta influência menor sobre o desempenho porque neste modelo os pedidos podem ser feitos em paralelo.

Como no caso do protocolo de controle TDM, para o modelo com controlador central foi possível identificar uma faixa com limitantes inferior e superior para o desempenho. Para quase todos os programas de teste a região abrangendo da Chave 2 até a Chave 4 apresentou desempenho do tipo linear, sendo que para dois programas de teste obteve-se desempenho do tipo ótimo. A Chave 5 apresentou desempenho do tipo inferior para quatro programas de teste e a Chave 6 teve desempenho do tipo linear para a maioria dos programas, sendo que para quatro programas obteve-se desempenho do tipo ótimo.

### 4.5.3 Análise comparativa dos resultados para os dois protocolos de controle

Como os protocolos de controle TDM e Nó Central têm resultados com a mesma ordem de grandeza, uma comparação diferencial é útil. Esta seção apresenta as diferenças entre os tempos de execução para os protocolos de controle TDM e Nó Central. A tabela 4.1 mostra na primeira coluna os nomes dos programas de teste. A segunda coluna mostra a diferença percentual entre os resultados dos tempos de execução obtidos para a Chave 1 utilizando o protocolo TDM e o protocolo com Nó Central. As colunas subsequentes apresentam o mesmo tipo de informação, mas referentes às demais velocidades.

Esses resultados representam a porcentagem de redução do tempo de execução de um protocolo em relação ao outro. Como para todos os programas e para todas as velocidades da chave, o protocolo de controle com Nó Central proporcionou desempenho melhor que o protocolo TDM, a tabela omite os nomes dos protocolos. A diferença entre os resultados dos dois protocolos se torna mais aparente conforme se aumenta a velocidade da chave. Isso é devido à influência do tempo de ciclo de controle, discutida nas seções 4.5.1 e 4.5.2.

TABELA 4.1 - Diferenças entre os tempos de execução dos protocolos de controle TDM e Nó Central

Programa	Chave 1	Chave 2	Chave 3	Chave 4	Chave 5	Chave 6
Ackermann	8,8	15,6	26,1	40,6	55,2	74,6
Add	16,9	27,1	41,9	57,7	72,0	88,1
Bin	21,1	28,0	44,4	57,8	72,6	87,1
Gauss	16,6	23,2	38,9	56,6	72,6	84,2
Interp	16,5	26,6	40,1	58,1	72,7	84,7
Laplace	14,9	21,6	40,1	56,1	73,7	84,6
Mms	14,4	23,6	40,0	58,6	71,9	84,0
Qid	16,4	27,7	43,5	61,5	74,3	87,0
Quick	9,8	23,9	37,0	55,7	70,7	83,4
Simple	14,7	25,8	40,4	58,4	73,3	85,2

## 4.6 Conclusões

O protocolo de controle TDM apresenta melhoras significativas de desempenho apenas para as primeiras velocidades de chave, ou seja, para Chave 2 e Chave 3. A partir daí o desempenho é menor devido à política de distribuição das fichas.

Já o modelo com protocolo de controle com Nó Central apresenta desempenho uniforme para todas as velocidades de chave. Esse modelo é mais eficiente no controle dos pedidos de transmissão de nós devido ao gerenciamento dinâmico dos pedidos.

Para os dois modelos de chave de interconexão o tempo do ciclo de controle constitui um fator de influência sobre o desempenho do sistema de comunicação. Esse fator é mais significativo conforme se aumenta a velocidade de transmissão. No controle TDM esse aspecto é mais marcante, degradando o desempenho da rede, visto que, nesse protocolo cada nó deve esperar pelo término do ciclo de controle para então realizar a transmissão. O método com controlador central é menos sensível a esse efeito, pois o ciclo de controle é mais eficiente e os pedidos dos nós são realizados em uma velocidade maior conforme se aumenta a taxa de transmissão da rede.

## **Capítulo 5**

### **Comportamento da Chave: Índice de Atividade**

#### **5.1 Introdução**

Este capítulo apresenta os resultados dos índices de atividade das unidades da arquitetura para as chaves de interconexão. O índice de atividade de uma unidade é a porcentagem do tempo de atividade da unidade durante a execução de um programa. A segunda seção mostra os resultados para o protocolo de controle TDM e a terceira seção mostra os resultados para o protocolo de controle com Nó Central. A seção subsequente apresenta os índices de atividade da chave para os dois modelos de controle implementados.

As tabelas de resultados das seções 5.2 e 5.3 apresentam na primeira coluna os nomes das unidades (EP0, EP1, ..., EP4, MD0, etc.). Na segunda coluna mostram os índices de atividade das unidades (em porcentagens) obtidos para a velocidade da Chave 1. As colunas subsequentes também apresentam o mesmo tipo de informações, mas referentes às demais velocidades. As tabelas da seção 5.4 mostram os resultados dos índices de atividade para a chave em função da velocidade. Na primeira coluna apresentam os nomes dos programas e a partir da segunda coluna mostram os índices de atividade da chave para cada velocidade.

## 5.2 Índices de atividade das unidades para o protocolo de controle TDM

As tabelas de 5.1 a 5.10 mostram os índices de atividade das unidades para o protocolo de controle TDM em função das variações de velocidade da chave.

A tabela 5.1 mostra os resultados dos índices de atividade para o programa Ackermann. A maioria das unidades apresentou crescimento do índice de atividade em função do aumento da velocidade da chave. O crescimento do índice de atividade das unidades não tem um comportamento uniforme: ele é mais significativo para as velocidades mais baixas. Conforme se aumenta a velocidade da chave os índices aumentam numa taxa menor. A maior taxa de crescimento médio, de cerca de 84%, foi obtida na variação de velocidade da Chave 1 para a Chave 2. Esse foi o melhor caso entre todos os programas para o protocolo de controle TDM.

TABELA 5.1 - Índices de Atividade para Ackermann - TDM

Índice de Atividade	Chave 1	Chave 2	Chave 3	Chave 4	Chave 5	Chave 6
EP0	0,52	0,95	1,55	2,39	3,15	3,41
EP1	0,53	0,97	1,57	2,36	3,04	3,00
EP2	0,55	1,00	1,66	2,47	3,36	3,51
EP3	0,49	0,92	1,63	2,45	3,21	3,62
EP4	0,52	0,92	1,57	2,38	3,23	4,22
MD0	0,28	0,51	0,86	1,30	1,75	1,94
MD1	0,27	0,49	0,83	1,26	1,70	1,98
MD2	0,28	0,50	0,85	1,30	1,74	1,92
MD3	0,27	0,49	0,83	1,26	1,70	1,94
MD4	0,28	0,50	0,86	1,30	1,75	2,01
CF	0,32	0,62	1,09	1,72	2,33	7,04

Os resultados dos índices de atividade para o programa Somatorial são mostrados na tabela 5.2. Para esse programa os índices de atividade também têm um crescimento mais acentuado para as velocidades menores. Sendo que a Chave 2 apresentou a maior taxa média de crescimento para o índice de atividade: em torno

de 70%. A Chave 6 não proporcionou crescimento dos índices de atividade das unidades, esse foi o pior caso entre todos os programas de teste para o protocolo TDM. Isso mostra que para esse programa, aumentar indefinidamente a velocidade da chave pode não trazer benefícios em termos de crescimento do índice de atividade. Portanto, nesse caso, há limitantes inferior e superior para a velocidade da chave em função dos índices de atividade. A Chave 2 corresponde ao limite inferior e a Chave 5 ao limite superior.

TABELA 5.2 - Índices de Atividade para Somatorial - TDM

Índice de Atividade						
	Chave 1	Chave 2	Chave 3	Chave 4	Chave 5	Chave 6
EP0	1,38	2,37	3,67	5,02	6,12	5,13
EP1	1,40	2,37	3,68	5,01	6,14	5,16
EP2	1,40	2,39	3,67	5,01	6,14	5,25
EP3	1,41	2,40	3,67	5,02	6,09	5,32
EP4	1,42	2,39	3,72	5,13	6,17	5,36
MD0	0,89	1,50	2,33	3,22	3,96	3,48
MD1	0,89	1,50	2,34	3,23	3,97	3,45
MD2	0,87	1,52	2,35	3,26	4,02	3,53
MD3	0,89	1,53	2,38	3,27	4,07	3,48
MD4	0,89	1,52	2,38	3,36	4,07	3,52
CF	4,06	7,07	10,95	14,99	18,29	23,17

O programa Integração Binária, cujos resultados para os índices de atividade são mostrados na tabela 5.3, apresenta comportamento semelhante ao do programa Somatorial. Com a maior taxa média de crescimento dos índices de atividade proporcionada pela Chave 2 e a Chave 5 sendo o limitante superior para a velocidade da chave.

A tabela 5.4 mostra os resultados dos índices de atividade para o programa Gauss. Para quase todas as velocidades houve crescimento dos índices de atividade, no entanto, para as velocidades maiores a taxa média de crescimento foi menor que para as velocidades menores.

TABELA 5.3 - Índices de Atividade para Integração Binária - TDM

Índice de Atividade						
	Chave 1	Chave 2	Chave 3	Chave 4	Chave 5	Chave 6
EP0	1,37	2,31	3,53	4,94	5,68	4,86
EP1	1,37	2,32	3,60	5,02	5,81	5,10
EP2	1,36	2,30	3,53	5,13	5,88	5,15
EP3	1,33	2,35	3,59	5,05	5,83	5,27
EP4	1,37	2,35	3,57	5,04	5,77	5,34
MD0	0,86	1,49	2,28	3,27	3,78	3,51
MD1	0,87	1,48	2,26	3,33	3,93	3,50
MD2	0,86	1,49	2,29	3,19	3,79	3,45
MD3	0,88	1,45	2,32	3,21	3,85	3,43
MD4	0,89	1,50	2,33	3,38	3,81	3,50
CF	4,01	6,94	10,88	14,88	18,04	22,88

TABELA 5.4 - Índices de Atividade para Gauss - TDM

Índice de Atividade						
	Chave 1	Chave 2	Chave 3	Chave 4	Chave 5	Chave 6
EP0	0,99	1,65	2,53	3,47	3,95	3,86
EP1	0,91	1,56	2,39	3,23	3,71	3,97
EP2	0,84	1,44	2,24	2,99	3,50	3,72
EP3	0,76	1,34	2,11	2,80	3,31	3,73
EP4	0,70	1,27	2,05	2,65	3,32	3,56
MD0	0,95	1,44	2,00	3,04	3,22	3,58
MD1	0,65	1,03	1,59	2,38	1,69	2,42
MD2	0,66	0,92	1,33	2,36	2,92	2,56
MD3	0,47	1,09	1,78	1,74	2,45	2,88
MD4	0,51	1,12	1,86	1,61	2,84	2,68
ME	0,52	0,98	1,65	2,33	2,78	3,05
CF	3,44	6,14	9,82	13,57	16,99	21,80

Os resultados dos índices de atividade para o programa Interpolação são apresentados na tabela 5.5. Para esse programa a Chave 2 proporcionou um aumento médio de 72% para os índices de atividade das unidades em relação à Chave 1, enquanto que a Chave 6 proporcionou um aumento médio de apenas 15% em relação à Chave 5.

Para o programa Laplace os índices de atividade das unidades (tabela 5.6) aumentam em média 77% na mudança da velocidade da Chave 1 para a Chave 2. Com o aumento da velocidade da chave essa taxa de aumento dos índices de

atividade foi menos significativa, sendo que da Chave 5 para a Chave 6 houve um aumento médio de cerca de 15%.

TABELA 5.5 - Índices de Atividade para Interpolação - TDM

Índice de Atividade						
	Chave 1	Chave 2	Chave 3	Chave 4	Chave 5	Chave 6
EP0	1,00	1,70	2,65	3,62	4,18	4,32
EP1	0,99	1,70	2,64	3,57	4,16	4,50
EP2	0,97	1,66	2,58	3,50	4,14	4,49
EP3	0,95	1,61	2,55	3,42	4,07	4,52
EP4	0,92	1,59	2,49	3,42	4,04	4,50
MD0	0,74	1,20	1,79	2,59	2,82	3,08
MD1	0,70	1,19	1,71	2,28	2,70	2,96
MD2	0,73	1,15	1,83	2,29	2,86	3,01
MD3	0,65	1,17	1,92	2,31	2,85	2,91
MD4	0,69	1,24	1,84	2,52	2,68	3,22
ME	0,03	0,05	0,08	0,11	0,14	0,15
CF	3,65	6,44	10,29	14,30	17,72	22,54

TABELA 5.6 - Índices de Atividade para Laplace - TDM

Índice de Atividade						
	Chave 1	Chave 2	Chave 3	Chave 4	Chave 5	Chave 6
EP0	0,94	1,66	2,34	3,46	4,36	4,05
EP1	0,84	1,53	2,19	2,80	3,28	3,44
EP2	0,73	1,31	1,90	2,53	2,91	3,21
EP3	0,68	1,19	1,82	2,44	2,75	3,17
EP4	0,63	1,13	1,69	2,22	2,60	3,00
MD0	0,79	1,34	1,51	2,84	3,04	3,08
MD1	0,67	0,99	1,51	2,22	2,88	2,95
MD2	0,67	0,92	1,56	1,84	1,84	2,42
MD3	0,51	0,91	1,44	1,59	1,89	2,29
MD4	0,59	1,45	2,00	2,14	2,85	2,61
ME	0,51	1,01	1,59	2,23	2,79	2,94
CF	3,58	6,31	9,95	13,52	16,55	21,87

A tabela 5.7 mostra o comportamento dos índices de atividade das unidades em função da variação de velocidade da chave para o programa Multiplicação de Matrizes. Também para esse programa a taxa média de crescimento dos índices de atividade é mais acentuada para as velocidades mais baixas.

TABELA 5.7 - Índices de Atividade para Multiplicação de Matrizes - TDM

Índice de Atividade						
	Chave 1	Chave 2	Chave 3	Chave 4	Chave 5	Chave 6
EP0	0,99	1,69	2,79	3,93	4,99	5,11
EP1	0,76	1,29	1,93	2,40	2,86	2,83
EP2	0,56	0,95	1,39	1,85	2,18	2,47
EP3	0,45	0,81	1,16	1,57	1,94	2,30
EP4	0,40	0,73	1,07	1,50	1,82	2,18
MD0	0,58	0,99	1,45	1,94	2,40	2,76
MD1	0,57	0,97	1,40	1,85	2,31	2,63
MD2	0,56	0,97	1,41	1,86	2,31	2,58
MD3	0,56	0,94	1,45	1,86	2,23	2,54
MD4	0,59	1,01	1,53	2,02	2,35	2,60
ME	0,89	1,71	2,77	3,80	4,74	5,37
CF	3,40	5,94	9,20	12,65	15,52	20,15

Os resultados dos índices de atividade para o programa N-Queens são mostrados na tabela 5.8. Assim como para os programas Somatorial e Integração Binária, para o programa N-Queens a Chave 6 não proporcionou crescimento dos índices de atividade para a maioria das unidades. Nesse caso também nota-se que o limitante superior de velocidade é a Chave 5 e que as taxas maiores de crescimento dos índices de atividade são proporcionadas pelas velocidades menores.

TABELA 5.8 - Índices de Atividade para N-Queens - TDM

Índice de Atividade						
	Chave 1	Chave 2	Chave 3	Chave 4	Chave 5	Chave 6
EP0	1,22	2,09	3,12	4,06	4,87	4,57
EP1	1,23	2,08	3,13	4,08	4,88	4,65
EP2	1,22	2,09	3,11	4,04	4,86	4,69
EP3	1,21	2,06	3,09	4,04	4,86	4,71
EP4	1,18	2,03	3,08	4,02	4,85	4,76
MD0	0,73	1,31	1,79	2,33	2,78	2,86
MD1	0,77	1,26	1,90	2,46	2,99	2,90
MD2	0,76	1,30	1,95	2,58	2,98	2,85
MD3	0,79	1,26	1,95	2,45	2,96	2,71
MD4	0,76	1,29	1,91	2,52	3,23	2,96
ME	0,19	0,32	0,49	0,64	0,77	0,75
CF	3,93	6,87	10,76	14,88	18,06	23,13

O programa Quick Sort (tabela 5.9) apresentou maior taxa de crescimento dos índices de atividade das unidades para as velocidades menores. Na mudança da Chave 1 para a Chave 2 a taxa de crescimento foi em média em torno de 73% e para a Chave 6 foi de cerca de 16%.

A tabela 5.10 mostra os resultados dos índices de atividade das unidades para o programa Simple. Esse programa também apresenta comportamento semelhante aos outros com crescimento dos índices de atividade das unidades em função do aumento da velocidade da chave, mas com taxas de crescimento maiores para as velocidades mais baixas.

TABELA 5.9 - Índices de Atividade para Quick Sort - TDM

Índice de Atividade						
	Chave 1	Chave 2	Chave 3	Chave 4	Chave 5	Chave 6
EP0	0,95	1,62	2,56	3,31	4,12	4,43
EP1	0,96	1,66	2,53	3,34	4,07	4,46
EP2	0,98	1,65	2,56	3,33	4,13	4,54
EP3	0,93	1,65	2,51	3,38	4,01	4,55
EP4	0,92	1,60	2,47	3,36	4,03	4,54
MD0	0,86	0,66	2,59	2,16	3,47	3,69
MD1	0,39	1,50	1,43	1,38	3,57	1,84
MD2	0,74	1,05	1,16	2,32	1,52	2,23
MD3	0,58	0,96	1,93	3,10	2,39	2,72
MD4	0,63	1,34	1,05	1,67	1,66	3,61
ME	0,33	0,57	0,89	1,19	1,48	1,68
CF	3,31	5,94	9,90	14,05	17,49	22,42

TABELA 5.10 - Índices de Atividade para Simple - TDM

Índice de Atividade						
	Chave 1	Chave 2	Chave 3	Chave 4	Chave 5	Chave 6
EP0	1,01	1,69	2,64	3,60	4,36	4,31
EP1	0,93	1,55	2,37	3,12	3,66	3,76
EP2	0,85	1,41	2,17	2,89	3,40	3,62
EP3	0,79	1,33	2,07	2,75	3,25	3,55
EP4	0,75	1,29	1,99	2,70	3,19	3,51
MD0	0,66	1,50	1,92	2,71	3,35	2,83
MD1	0,68	1,22	1,88	2,19	2,53	3,27
MD2	0,68	1,15	1,82	1,98	2,86	3,00
MD3	0,76	0,99	1,58	2,45	2,68	2,85
MD4	0,73	0,98	1,73	2,57	2,74	2,89
ME	0,67	1,20	1,98	2,73	3,30	3,62
CF	3,68	6,44	10,11	13,89	17,10	22,06

Identificou-se algumas características de comportamento para os índices de atividade das unidades da arquitetura em função da variação de velocidade da chave para o protocolo de controle TDM. Essas características de comportamento são:

- Para a maioria dos programas o aumento da velocidade da chave resulta numa melhora dos índices de atividade das unidades.
- Essa melhora é mais significativa para as velocidades menores, à medida em que se aumenta a velocidade da chave os índices de atividade crescem numa taxa menor.
- Para alguns programas o aumento progressivo da velocidade da chave para valores acima de 40 Gb/s não provê benefícios e por vezes, esse aumento de velocidade ocasiona índices de atividade menores que para velocidades mais baixas. Por exemplo, para os programas Somatorial, Integração Binária e N-Queens a Chave 6 proporcionou índices de atividade menores que para a Chave 5.

### **5.3 Índices de atividade das unidades para o protocolo de controle com Nó Central**

As tabelas de 5.11 a 5.20 mostram os índices de atividade das unidades para o protocolo de controle com Nó Central em função das variações de velocidade da chave.

A tabela 5.11 mostra os resultados dos índices de atividade para o programa Ackermann. Para todas as velocidades da chave ocorre aumento dos índices de atividade das unidades. Esse aumento não é uniforme em função da variação da velocidade, mas diferentemente do comportamento registrado para esse programa no

método de controle TDM, a taxa de crescimento dos índices de atividade não diminuiu com o aumento da velocidade da chave. A Chave 6 proporcionou o maior crescimento médio para os índices de atividade, sendo este o melhor caso entre todos os programas para o protocolo de controle com Nó Central com taxa de crescimento em torno de 103%. Enquanto a Chave 5 resultou na menor taxa média de crescimento dos índices de atividade.

TABELA 5.11 - Índices de Atividade para Ackermann - Nó Central

Índice de Atividade						
	Chave 1	Chave 2	Chave 3	Chave 4	Chave 5	Chave 6
EP0	0,57	1,12	2,15	4,03	6,75	13,66
EP1	0,55	1,10	2,07	3,89	6,74	13,57
EP2	0,61	1,19	2,28	4,28	6,93	13,48
EP3	0,58	1,15	2,16	4,05	6,78	13,71
EP4	0,57	1,11	2,14	4,02	6,68	13,26
MD0	0,31	0,61	1,16	2,18	3,87	8,02
MD1	0,30	0,58	1,12	2,10	3,74	7,61
MD2	0,30	0,59	1,14	2,15	3,82	7,70
MD3	0,30	0,59	1,11	2,09	3,72	7,69
MD4	0,30	0,58	1,14	2,14	3,81	7,60
CF	0,37	0,75	1,41	2,48	3,63	8,78

O programa Somatorial (tabela 5.12) obteve as maiores taxas médias de crescimento dos índices de atividade para as Chaves 2 e 3 com uma média de crescimento em torno de 95% e a menor taxa para a Chave 5 em torno de 78%.

Os resultados para o programa Integração Binária são mostrados na tabela 5.13. Assim como para o programa Somatorial também para esse programa a Chave 3 e a Chave 5 apresentaram a maior e a menor taxa de crescimento, respectivamente.

A tabela 5.14 mostra os resultados para o programa Gauss. Os maiores índices de atividade são obtidos para a Chave 6. As taxas de aumento dos índices não são uniformes, sendo mais significativas para a mudança de velocidade da Chave 2 para a Chave 3.

TABELA 5.12 - Índices de Atividade para Somatorial - Nó Central

Índice de Atividade						
	Chave 1	Chave 2	Chave 3	Chave 4	Chave 5	Chave 6
EP0	1,71	3,33	6,59	12,44	22,45	42,76
EP1	1,73	3,37	6,64	12,48	22,87	45,61
EP2	1,73	3,38	6,65	12,36	22,70	45,48
EP3	1,73	3,41	6,66	12,58	22,72	44,69
EP4	1,74	3,40	6,63	12,58	22,97	41,80
MD0	1,07	2,10	4,12	7,84	14,52	29,52
MD1	1,09	2,11	4,18	7,88	14,45	29,42
MD2	1,08	2,14	4,21	7,82	14,48	29,67
MD3	1,09	2,14	4,17	7,85	14,31	29,51
MD4	1,08	2,15	4,18	7,97	14,37	30,21
CF	4,66	9,09	17,50	31,94	52,93	65,26

TABELA 5.13 - Índices de Atividade para Integração Binária - Nó Central

Índice de Atividade						
	Chave 1	Chave 2	Chave 3	Chave 4	Chave 5	Chave 6
EP0	1,81	3,35	6,80	12,56	22,05	39,52
EP1	1,80	3,29	6,70	12,85	22,42	41,63
EP2	1,81	3,42	6,86	12,76	22,21	41,52
EP3	1,79	3,31	6,89	12,54	22,27	39,52
EP4	1,75	3,27	6,61	12,42	21,98	35,08
MD0	1,13	2,11	4,34	7,97	14,52	27,17
MD1	1,15	2,13	4,28	7,91	13,83	27,77
MD2	1,14	2,18	4,24	8,11	14,35	27,01
MD3	1,10	2,12	4,23	7,75	13,92	26,53
MD4	1,14	2,14	4,32	8,07	14,27	27,49
CF	4,64	9,00	17,24	31,41	51,17	62,27

TABELA 5.14 - Índices de Atividade para Gauss - Nó Central

Índice de Atividade						
	Chave 1	Chave 2	Chave 3	Chave 4	Chave 5	Chave 6
EP0	1,14	2,17	4,41	8,14	15,67	23,52
EP1	1,17	2,24	4,34	8,16	15,56	25,37
EP2	1,05	1,96	3,86	7,19	13,74	23,13
EP3	0,95	1,75	3,45	6,75	12,35	21,29
EP4	0,83	1,62	3,17	6,03	10,99	18,84
MD0	1,05	2,01	3,92	6,96	16,35	20,18
MD1	0,56	1,33	2,77	5,04	10,05	19,47
MD2	0,71	1,31	2,98	4,23	8,42	13,55
MD3	0,78	1,44	2,54	5,38	8,29	17,73
MD4	0,88	1,49	2,75	6,31	8,14	15,89
ME	0,53	1,01	2,00	3,71	6,83	13,16
CF	3,90	7,59	15,02	27,59	46,94	64,99

Para o programa Interpolação (tabela 5.15) a Chave 2 proporcionou a maior taxa média de crescimento para os índices de atividade em torno de 95%. Para todas as velocidades ocorre aumento dos índices, sendo que os maiores valores para os índices são obtidos através da Chave 6.

Os resultados dos índices de atividade para o programa Laplace (tabela 5.16) mostram que a mudança de velocidade da Chave 5 para a Chave 6 apresentou a menor taxa média de crescimento do índice de atividade com 59% e a maior taxa foi obtida para a Chave 3 com cerca de 94% de crescimento.

A Chave 2 proporcionou a maior taxa média de crescimento dos índices de atividade para o programa Multiplicação de Matrizes, cujos resultados são mostrados na tabela 5.17. Todas as velocidades de chave resultaram em aumentos dos índices de atividade. A Chave 6 apresentou os maiores índices de atividade para esse programa.

TABELA 5.15 - Índices de Atividade para Interpolação - Nó Central

Índice de Atividade						
	Chave 1	Chave 2	Chave 3	Chave 4	Chave 5	Chave 6
EP0	1,21	2,30	4,57	8,86	15,79	29,04
EP1	1,25	2,40	4,68	8,97	16,19	30,83
EP2	1,19	2,33	4,44	8,75	15,51	28,86
EP3	1,14	2,26	4,30	8,36	14,99	26,64
EP4	1,08	2,15	4,05	8,05	14,31	23,63
MD0	0,85	1,80	3,40	6,70	11,38	19,79
MD1	0,87	1,63	3,31	6,25	10,78	18,97
MD2	0,87	1,66	3,21	5,87	10,24	19,54
MD3	0,86	1,61	3,19	5,73	10,93	18,31
MD4	0,82	1,64	3,08	6,37	11,10	19,34
ME	0,04	0,07	0,14	0,27	0,49	0,98
CF	4,23	8,18	15,99	29,42	50,00	67,24

TABELA 5.16 - Índices de Atividade para Laplace - Nó Central

Índice de Atividade						
	Chave 1	Chave 2	Chave 3	Chave 4	Chave 5	Chave 6
EP0	1,13	2,07	4,36	8,09	15,49	26,58
EP1	1,08	2,01	4,04	7,57	14,27	24,83
EP2	0,92	1,74	3,34	6,18	12,26	19,84
EP3	0,76	1,59	2,91	5,40	10,51	17,95
EP4	0,73	1,45	2,65	4,82	9,92	15,76
MD0	1,01	1,99	3,53	7,81	13,93	17,16
MD1	0,88	1,00	2,86	5,36	9,92	18,19
MD2	0,78	1,54	2,68	4,07	8,26	11,41
MD3	0,59	1,38	1,96	3,67	7,25	16,46
MD4	0,68	1,65	3,58	5,47	10,71	21,17
ME	0,47	0,90	1,80	3,21	6,17	11,86
CF	4,06	7,96	15,46	27,24	46,41	62,68

TABELA 5.17 - Índices de Atividade para Multiplicação de Matrizes - Nó Central

Índice de Atividade						
	Chave 1	Chave 2	Chave 3	Chave 4	Chave 5	Chave 6
EP0	1,13	2,30	4,60	8,33	15,49	28,55
EP1	0,96	1,94	3,63	6,82	12,40	19,72
EP2	0,68	1,31	2,48	5,03	8,81	15,22
EP3	0,54	1,00	1,94	4,10	7,31	11,78
EP4	0,45	0,85	1,64	3,50	6,17	10,31
MD0	0,70	1,35	2,64	5,13	9,03	17,05
MD1	0,67	1,31	2,58	4,89	8,64	16,13
MD2	0,68	1,33	2,53	4,90	8,64	16,00
MD3	0,67	1,32	2,54	4,90	8,61	15,71
MD4	0,71	1,40	2,68	5,12	9,06	16,20
ME	0,77	1,50	2,93	5,57	9,72	20,29
CF	3,95	7,63	14,91	27,96	48,23	66,74

A tabela 5.18 mostra o comportamento dos índices de atividade para o programa N-Queens. Os valores dos índices de atividade crescem com o aumento da velocidade da chave, mas a taxa média de crescimento é mais acentuada para a Chave 2.

O programa Quick Sort (tabela 5.19) apresenta aumentos dos índices de atividade para todas as velocidades. A variação de velocidade da Chave 1 para a Chave 2 proporcionou a maior taxa média de crescimento, em torno de 102%.

Os índices de atividade para o programa Simple (tabela 5.20) têm taxas de crescimento mais significativas para as velocidades menores. Sendo que para as Chaves 2 e 3 a taxa média é de cerca de 94%.

TABELA 5.18 - Índices de Atividade para N-Queens - Nó Central

Índice de Atividade						
	Chave 1	Chave 2	Chave 3	Chave 4	Chave 5	Chave 6
EP0	1,48	2,95	5,73	11,16	19,68	36,01
EP1	1,53	3,02	5,87	11,18	19,97	38,04
EP2	1,50	2,98	5,77	11,03	19,79	37,14
EP3	1,46	2,90	5,65	10,86	19,39	35,11
EP4	1,42	2,83	5,45	10,57	18,75	32,19
MD0	0,96	1,87	3,50	7,14	11,96	21,17
MD1	0,88	1,92	3,47	6,92	11,50	21,47
MD2	0,94	1,79	3,77	6,67	12,27	21,41
MD3	0,95	1,89	3,78	6,62	12,14	22,32
MD4	0,98	1,91	3,61	6,88	12,20	21,46
ME	0,22	0,43	0,83	1,55	2,57	4,93
CF	4,51	8,76	16,83	30,77	51,55	67,09

TABELA 5.19 - Índices de Atividade para Quick Sort - Nó Central

Índice de Atividade						
	Chave 1	Chave 2	Chave 3	Chave 4	Chave 5	Chave 6
EP0	1,06	2,21	4,22	7,91	14,50	26,10
EP1	1,12	2,27	4,27	8,08	14,74	27,82
EP2	1,11	2,27	4,23	7,82	14,52	27,62
EP3	1,05	2,18	4,10	7,84	13,88	26,62
EP4	1,03	2,15	4,04	7,71	13,97	24,96
MD0	0,73	2,04	2,00	7,03	12,44	16,42
MD1	1,08	1,40	2,33	4,90	11,22	14,65
MD2	0,70	0,94	3,64	4,93	7,81	15,13
MD3	0,62	2,19	2,20	4,74	5,86	12,41
MD4	0,56	1,08	4,21	5,08	10,61	24,98
ME	0,36	0,75	1,41	2,67	4,86	9,63
CF	3,76	7,25	14,14	26,05	44,99	63,70

TABELA 5.20 - Índices de Atividade para Simple - Nó Central

Índice de Atividade						
	Chave 1	Chave 2	Chave 3	Chave 4	Chave 5	Chave 6
EP0	1,18	2,27	4,48	8,58	15,86	28,01
EP1	1,17	2,24	4,37	8,31	15,27	27,56
EP2	1,04	2,00	3,88	7,46	13,71	24,54
EP3	0,94	1,84	3,56	6,93	12,69	22,41
EP4	0,87	1,72	3,30	6,48	11,88	20,15
MD0	0,90	1,78	3,11	6,37	11,10	22,32
MD1	0,72	1,46	3,59	7,18	11,40	18,68
MD2	0,81	1,72	2,84	5,30	9,70	18,83
MD3	0,88	1,58	2,80	5,01	10,34	16,13
MD4	0,89	1,66	3,51	6,28	11,70	21,05
ME	0,63	1,23	2,39	4,51	8,07	16,39
CF	4,24	8,29	16,15	29,70	50,48	67,72

Para o protocolo de controle com Nó Central nota-se algumas características de comportamento dos índices de atividade das unidades em função da variação de velocidade da chave.

Essas características de comportamento são:

- Para todos os programas de teste, os aumentos de velocidade da chave propiciaram um crescimento dos índices de atividade das unidades.
- O crescimento dos índices de atividade não é uniforme em função da variação de velocidade da chave. Por exemplo, para o programa Laplace a mudança de velocidade da Chave 2 para Chave 3 resultou numa taxa média de crescimento do índice de cerca de 94%, e a mudança de velocidade da Chave 5 para a Chave 6 proporcionou uma taxa média de crescimento em torno de 59%.
- A Chave 6 apresenta os maiores índices de atividade para todos os programas.
- O protocolo de Nó Central não apresentou limitante superior para o desempenho dos índices de atividade como no caso do protocolo TDM em

que, para alguns programas a chave mais veloz não traz benefícios em termos de crescimento dos índices de atividade.

## **5.4 Índices de atividade da Chave**

Os resultados dos índices de atividade obtidos para a *Chave* são apresentados nas subseções seguintes. A primeira subseção mostra os resultados para o protocolo de controle TDM e a segunda mostra os resultados para o protocolo de controle com Nó Central.

### **5.4.1 Resultados para a chave com protocolo de controle TDM**

A tabela 5.21 mostra os índices de atividade da *Chave* para o protocolo de controle TDM em função das variações de velocidade. Para os programas Somatorial, Interpolação, Multiplicação de Matrizes, N-Queens e Simple a Chave 1 apresentou índices de atividade mais baixos em relação aos resultados obtidos para outros programas de teste.

A Chave 2 apresentou dois resultados com índices abaixo de 90% para os programas Multiplicação de Matrizes e Simple. Enquanto que a Chave 3 apresentou um índice baixo para o programa Simple. Para as velocidades maiores, a partir da Chave 4, os índices de atividade obtidos foram todos maiores que 90%.

Nota-se, portanto, que para o protocolo de controle TDM as chaves mais rápidas (Chave 4, Chave 5 e Chave 6) proporcionaram índices de atividade maiores, com valores próximos.

TABELA 5.21 - Índices de Atividade da Chave para controle TDM

Índice de Atividade da Chave						
Programa	Chave 1	Chave 2	Chave 3	Chave 4	Chave 5	Chave 6
Ackermann	98,98	98,16	96,64	94,82	92,99	92,36
Somatorial	54,44	93,68	99,91	99,72	99,42	99,57
Int. Binária	99,84	99,72	99,54	99,15	98,74	98,95
Gauss	99,84	99,71	99,50	98,88	98,41	98,85
Interpolação	73,15	99,82	99,67	99,11	98,47	99,06
Laplace	99,77	99,58	99,22	98,48	98,03	98,41
Mult. Mat.	50,71	88,42	99,47	98,97	98,55	98,79
N-Queens	64,73	99,92	99,80	99,24	98,59	99,13
Quick Sort	99,84	99,71	99,56	99,06	98,36	98,73
Simple	25,71	43,59	68,18	92,57	98,93	99,29

#### 5.4.2 Resultados para a chave com protocolo de controle com Nó Central

A tabela 5.22 mostra os índices de atividade da chave para o protocolo de controle com Nó Central em função das variações de velocidade. Assim como para o protocolo de controle TDM, para o controle de Nó Central a Chave 1 novamente apresentou os índices mais baixos para os mesmos programas: Somatorial, Interpolação, Multiplicação de Matrizes, N-Queens e Simple. A Chave 2 apresentou um índice baixo para o programa Simple como aconteceu para o protocolo de controle TDM. A Chave 5 apresentou seu menor índice para o programa Ackermann com 84,07%. A média dos índices de atividade para a Chave 6 foi de 79,07%. Os maiores índices de atividade foram obtidos para a Chave 3 e Chave 4 com médias de 99,16% e 98,4% respectivamente.

TABELA 5.22 - Índices de Atividade da Chave para controle com Nó Central

Índice de Atividade da Chave						
Programa	Chave 1	Chave 2	Chave 3	Chave 4	Chave 5	Chave 6
Ackermann	98,94	97,92	95,63	91,77	84,07	53,13
Somatorial	65,52	99,96	99,93	99,86	99,71	91,06
Int. Binária	99,80	99,63	99,24	98,56	97,36	86,18
Gauss	99,83	99,66	99,48	98,91	97,81	79,70
Interpolação	87,61	99,84	99,76	99,47	98,88	81,41
Laplace	99,75	99,51	99,07	98,13	96,37	77,15
Mult. Mat.	59,22	99,76	99,40	99,27	98,24	72,42
N-Queens	77,45	99,93	99,87	99,71	99,42	86,30
Quick Sort	99,82	99,64	99,47	98,90	97,93	81,78
Simple	30,15	58,75	99,72	99,49	99,00	81,52

## 5.6 Conclusões

Este capítulo investigou o comportamento dos índices de atividade das unidades em função das chaves de interconexão da arquitetura. Essa investigação permitiu analisar a eficiência do controle das chaves sob diferentes cargas e tráfegos.

Para a chave com protocolo de controle TDM os índices de atividade crescem numa taxa menor conforme se aumenta a velocidade da chave. Esse tipo de controle apresenta limitantes para a velocidade da chave, pois, para alguns casos o aumento progressivo da velocidade não resultou em melhorias nos índices de atividade.

A chave com protocolo de controle com Nó Central proporcionou maiores índices de atividade das unidades que a chave com controle TDM. Isso mostra que em termos de índices de atividade o modelo de controle com Nó Central é mais eficiente.

Para o protocolo de controle com Nó Central os maiores índices de atividade das unidades foram obtidos para a velocidade da Chave 6. Analisando-se os índices de atividade apresentados pela unidade *Chave*, verifica-se que para a velocidade Chave 6 obteve-se os menores índices de atividade. Isso pode representar a

aproximação de um ponto de equilíbrio, isto é, a velocidade de Chave 6 pode estar próxima de um ponto em que as unidades da arquitetura apresentem os maiores índices de atividade e a própria *Chave* esteja sendo menos sobrecarregada. Pode existir então um limitante superior de velocidade para a *Chave* em termos de índice de atividade.

## Capítulo 6

### Considerações finais e trabalhos futuros

#### 6.1 Introdução

Este capítulo sintetiza as conclusões embasado nas discussões e dados expostos nos capítulos precedentes. Discute também oportunidades de continuidade de investigações futuras para este trabalho.

Trabalhos de investigação sobre redes de interconexão têm sido extensivamente realizados. Entretanto, uma parcela significativa dos modelos de investigação são comumente baseados em hipóteses que não são realistas, como por exemplo, quando um pedido de transmissão é bloqueado, este pedido é então descartado e um pedido independente é gerado para substituir aquele previamente bloqueado e que não foi atendido. Essas hipóteses ajudam a simplificar o modelo teórico, mas a simplificação pode resultar em discrepâncias na predição do desempenho da rede.

Este trabalho é importante porque apresenta medidas quantitativas de cada estratégia de rede de interconexão proposta para a arquitetura. Além disso, este trabalho considera dois modelos de redes de interconexão enquanto que a maioria dos trabalhos considera apenas uma rede de interconexão. Isso permite identificar

características de cada modelo que possam contribuir para melhorar o desempenho do sistema.

## **6.2 Redes de Interconexão para a Arquitetura a Fluxo de Dados APO**

Este trabalho investigou o sistema de comunicação da arquitetura a fluxo de dados APO. Com base nos requisitos de comunicação da arquitetura foram propostos dois modelos de redes de interconexão: o primeiro baseado na técnica TDM com controle descentralizado e o segundo com controle central.

As limitações das redes de interconexão eletrônicas tradicionais e o desenvolvimento da tecnologia de comunicação óptica ocasionaram um crescente interesse de pesquisas na área de redes de interconexão com utilização da tecnologia WDM. As duas redes propostas para a arquitetura APO utilizam a tecnologia WDM e apresentam as seguintes características:

- São otimizadas em termos de ciclos de controle e de transmissão. Permitem a sobreposição dos mesmos, proporcionando maior eficiência do sistema de comunicação.
- São livres de colisões, evitando o *overhead* de comunicação que seria gerado caso houvesse necessidade de se tratar colisões.
- Empregam receptores fixos, evitando colisões de receptores e a necessidade de sintonização.
- Proporcionam facilidade de escalabilidade para a arquitetura.

As redes de interconexão eletrônicas como barramento compartilhado ou redes multi-estágio apresentam limitações como: velocidade, complexidade de controle e escalabilidade. Fato que as torna inviáveis para arquiteturas a fluxo de dados. Uma característica bastante criticada das arquiteturas a fluxo de dados é o chaveamento de fichas com muitos bits em paralelo. Isso implica que a complexidade e tamanho dos subsistemas de chaveamento representam uma parcela significativa da complexidade do sistema. Os sistemas ópticos, em contraste, oferecem altas taxas de transmissão sem a necessidade de se utilizar várias conexões paralelas, evitando as dificuldades de interconexões físicas apresentadas pelas redes eletrônicas. Num sistema WDM vários canais podem ser operados em paralelo numa única fibra, isso reduz o número de conexões dos nós do sistema.

Algumas outras características das redes de interconexão eletrônicas tradicionais são:

- As redes de interconexão multi-estágio aumentam a latência, pois a informação deve trafegar por vários estágios até chegar ao destino final.
- Nas redes multi-estágio podem ocorrer conflitos nos elementos de chaveamento mesmo quando os nós de destino são diferentes.
- Essa possibilidade de ocorrência de conflitos torna o controle da rede mais complexo, por isso essas redes usam técnicas de roteamento para resolver conflitos.
- Para cada estágio da rede, cada elemento de chaveamento deve ter mecanismo de decisão de rota.
- Outra característica dessas redes multi-estágio é que, em geral, elas têm estrutura regular, isto é, têm  $N$  entradas e  $N$  saídas, com  $N$  sendo uma

potência de 2. Essa falta de flexibilidade pode acarretar um desperdício de conexões.

Os dois modelos de redes ópticas propostos para a arquitetura APO apresentam ainda outras vantagens em relação às redes de interconexão eletrônicas:

- Num sistema óptico *single-hop* evita-se a latência de roteamento entre nós intermediários, como ocorre nas redes multi-estágio eletrônicas.
- Nas redes ópticas com conexão direta não ocorrem conflitos quando os nós de destino são diferentes.
- A ausência de conflitos quando os nós fontes endereçam destinos diferentes permite que o controle da rede seja mais simplificado nesse aspecto que o de uma rede multi-estágio.
- Nas redes ópticas da APO não há elementos de chaveamento intermediários que devem tomar decisões de roteamento como nas redes multi-estágio. As decisões de chaveamento cabem aos próprios nós, no caso do modelo TDM e ao Nó Central, no caso do modelo com controle central.
- A estrutura das redes da APO são do tipo ponto-a-ponto, não dependem de uma disposição regular com fator de potência de 2. Portanto, só interligam nós que necessitem se comunicar diretamente.

### **6.3 Comportamento das Redes de Interconexão da arquitetura APO**

O comportamento das chaves de interconexão foi analisado em função do modelo implementado e das velocidades da chave para os seguintes parâmetros: distribuição de carga, tempo de execução e índice de atividade.

Os resultados para as distribuições de carga mostram que os dois modelos de chaves de interconexão da arquitetura APO apresentaram distribuições muito mais homogêneas do que as distribuições obtidas pelo modelo de chave da arquitetura Wolf. As chaves da arquitetura APO não apresentaram sobrecarga de processadores como no caso da arquitetura Wolf em que, os primeiros processadores são muito mais utilizados que os últimos. Entre as chaves da arquitetura APO a melhor distribuição de carga foi obtida pelo modelo TDM. Não se obteve uma relação entre a distribuição de carga e a variação da velocidade das chaves. Para alguns algoritmos a distribuição é mais sensível às variações da velocidade que para outros. Não há, portanto, uma regra de comportamento para as distribuições de carga em função da variação das velocidades de transmissão.

Os resultados da investigação do comportamento da chave em termos dos tempos de execução mostram que a metodologia de controle com Nó Central é mais adequada por ser menos sensível à variação de desempenho em função da velocidade. O protocolo de controle TDM apresenta melhoras significativas de desempenho apenas para as primeiras velocidades de chave, sendo que o desempenho para as velocidades mais altas é menor. O protocolo de Nó Central apresenta desempenho uniforme para todas as velocidades de chave, sendo mais eficiente no controle dos pedidos de transmissão de nós.

A investigação do comportamento dos índices de atividade das unidades em função das chaves de interconexão mostra que para o protocolo de controle TDM os índices de atividade crescem numa taxa menor conforme se aumenta a velocidade da chave. Esse modelo apresenta limitantes para a velocidade da chave, visto que, em alguns casos o aumento progressivo da velocidade não proporcionou melhorias dos

índices de atividade. No entanto, a chave com protocolo de Nó Central proporcionou índices de atividade maiores que a chave com controle TDM sendo mais eficiente.

O problema de investigação de chaves de interconexão para a arquitetura APO é um problema sem solução única: existem soluções melhores e piores. Não há um método de controle e uma única velocidade de chave que sejam melhores para todos os casos. Mas, para dois dos três parâmetros analisados (tempos de execução e índices de atividade) nota-se que o modelo de controle com Nó Central foi mais eficiente.

As redes de interconexão com tecnologia de comunicação óptica constituem uma boa solução, visto que, os resultados obtidos nessa investigação são coerentes, diferentemente do trabalho de mestrado [5] que empregava redes de interconexão eletrônicas, em que havia uma grande dispersão dos resultados.

A arquitetura APO apresenta algumas melhorias em relação às outras arquiteturas (Wolf e SEED) propostas pelo Grupo de Pesquisas em Arquiteturas a Fluxo de Dados do IFSC:

- APO apresenta melhor distribuição de carga que a arquitetura Wolf.
- APO emprega apenas uma chave de interconexão, portanto, reduz a latência, custo e complexidade de controle. Enquanto que as arquiteturas Wolf e SEED utilizam duas chaves de interconexão.
- A arquitetura Wolf usa apenas uma MD que pode ser um gargalo em termos de comunicação, enquanto que a APO possibilita o uso de múltiplas MDs.

## **6.4 Trabalhos Futuros**

Os resultados deste trabalho instigam esforços para novos trabalhos a serem desenvolvidos. Esta seção apresenta propostas de continuidade de investigações futuras para este trabalho.

### **6.4.1 Emprego dos modelos de redes de interconexão da arquitetura APO em outras arquiteturas paralelas**

Este trabalho tem o potencial de disponibilizar a utilização dos dois modelos de redes de interconexão propostos para a arquitetura APO também para outros modelos de arquiteturas.

### **6.4.2 Incorporação dos modelos de redes de interconexão da arquitetura APO na arquitetura SEED**

As redes de interconexão da arquitetura APO podem ser incorporadas também ao simulador SEED, permitindo a investigação de desempenho dessa arquitetura em função das chaves ópticas.

### **6.4.3 Investigação do comportamento dos índices de atividade das unidades da arquitetura APO**

Os valores baixos obtidos para os índices de atividade das unidades da arquitetura APO demandam investigações desse comportamento. É necessário um estudo para se definir uma forma de balanceamento desses índices.

#### **6.4.4 Investigação da distribuição da unidade CF nas entradas das MDs na arquitetura APO**

Do aspecto do subsistema de comunicação é interessante se investigar a implementação de unidades CF distribuídas nas entradas das MDs. Isso tem o potencial de melhorar o tráfego, visto que, a utilização de uma única unidade CF pode sobrecarregar esta unidade em relação às demais. Além disso, essa distribuição da CF pode proporcionar alterações também em relação aos tempos de execução e aos índices de atividade.

#### **6.4.5 Incorporar a metodologia de distribuição de carga do modelo TDM no modelo com Nó Central**

O protocolo de controle TDM apresentou melhores resultados para a distribuição de carga na arquitetura APO. Isso suscita uma investigação sobre as mudanças necessárias para se implementar a metodologia de distribuição de carga do modelo TDM no modelo com Nó Central.

#### **6.4.6 Investigação do balanceamento de velocidade da Chave para o modelo com Nó Central**

Os resultados mostram que a rede de interconexão com Nó Central foi mais eficiente para a maioria dos parâmetros de investigação de desempenho. Isso suscita uma investigação sobre o balanceamento de velocidade da *Chave* para esse modelo de controle. Pois as unidades da arquitetura APO apresentaram os maiores índices de atividade para a velocidade Chave 6 e a unidade *Chave* apresentou valores mais baixos para o índice de atividade para essa velocidade. Isso pode representar a

existência de um ponto de equilíbrio, isto é, é possível que para velocidades próximas da Chave 6 as unidades comecem a trabalhar mais e a própria unidade *Chave* seja menos sobrecarregada.

## 6.5 Contribuições deste Trabalho

As contribuições deste trabalho são apresentadas a seguir:

- O presente trabalho propõe um modelo de rede de interconexão óptica *single-hop* TT-FR baseada na tecnologia WDM com controle TDM para a arquitetura a fluxo de dados APO.
- Propõe também um segundo modelo de rede de interconexão óptica *singe-hop* TT-FR baseada na tecnologia WDM com um controlador centralizado para a arquitetura APO.
- Este trabalho implementa o processo evolutivo da arquitetura Wolf para a arquitetura APO. Esse processo consiste na eliminação do gargalo de comunicação presente na arquitetura Wolf ocasionado pela utilização da Memória de Instruções Externa e de uma única Memória de Dados por onde circula todas as fichas da arquitetura. O trabalho de desenvolvimento da arquitetura APO envolveu a eliminação da Memória de Instruções Externa e a implementação de sua funcionalidade através das Memórias de Instruções Internas, além disso, foram implementadas múltiplas Memórias de Dados aproveitando o campo de Nome de Ativação (NA) presente nas fichas da arquitetura, evitando-se assim a criação de um novo campo para controle que geraria overhead de controle nas fichas.

- Com as modificações implementadas para a arquitetura APO foi possível solucionar o problema de distribuição de carga detectado anteriormente na arquitetura Wolf.
- Este trabalho permite a investigação do comportamento da arquitetura APO em função das redes de interconexão propostas.
- O trabalho também propõe caminhos de investigações futuras para a arquitetura APO baseado nos resultados obtidos na investigação do impacto das redes de interconexão sobre a arquitetura.

## **Anexo A**

### **Comportamento da Chave: Distribuição de Carga para o Protocolo de Controle TDM**

#### **A.1 Introdução**

Este anexo apresenta os resultados das distribuições de carga para a chave de interconexão da arquitetura APO com controle TDM para o programa Gauss.

#### **A.2 Distribuição de carga dos processadores para a arquitetura APO com controle TDM**

Os resultados da distribuição de carga da arquitetura APO para o protocolo de controle TDM usando o programa de Gauss são mostrados nas figuras de A.1 a A.6.

Distribuição de Carga TDM Chave 6 APO - Gauss

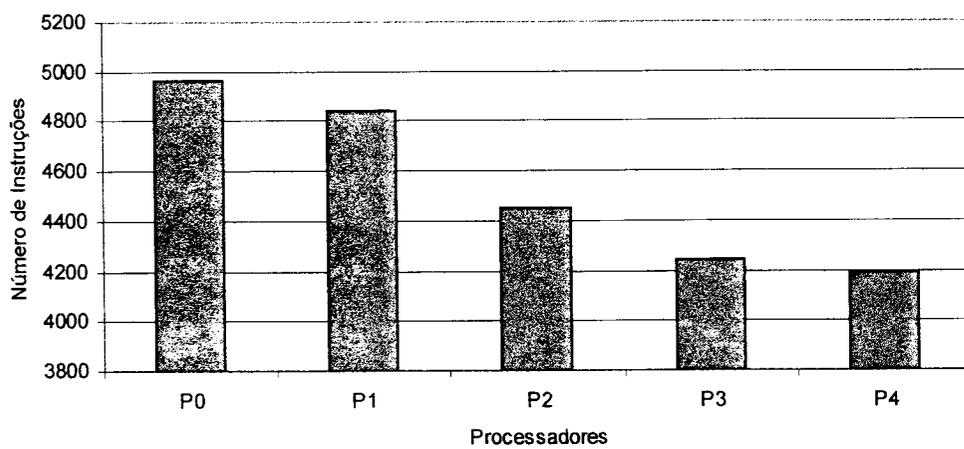


FIGURA A.1 - Distribuição de Carga da arquitetura APO para controle TDM com Chave 6 para Gauss

Distribuição de Carga TDM Chave 5 APO - Gauss

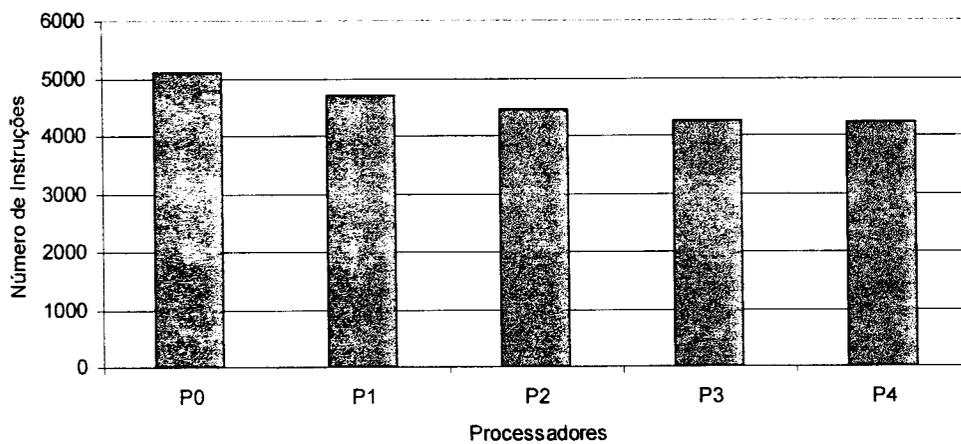


FIGURA A.2 - Distribuição de Carga da arquitetura APO para controle TDM com Chave 5 para Gauss

Distribuição de Carga TDM Chave 4 APO - Gauss

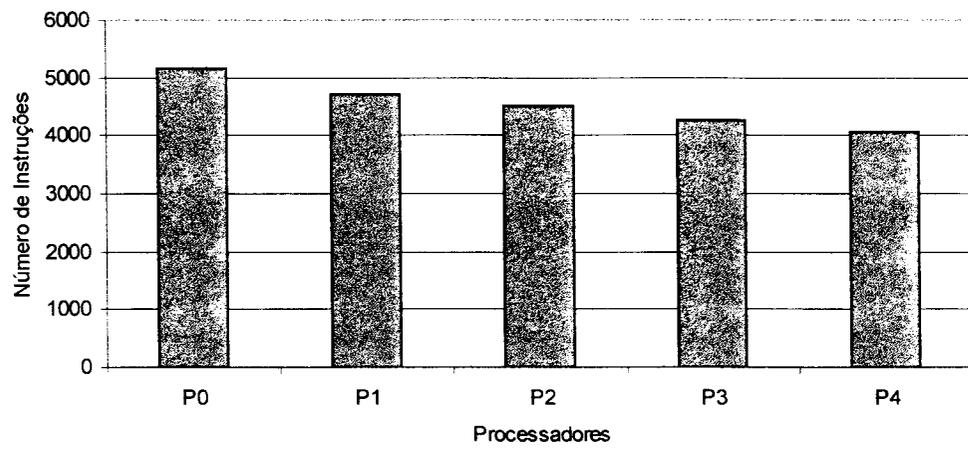


FIGURA A.3 - Distribuição de Carga da arquitetura APO para controle TDM com Chave 4 para Gauss

Distribuição de Carga TDM Chave 3 APO - Gauss

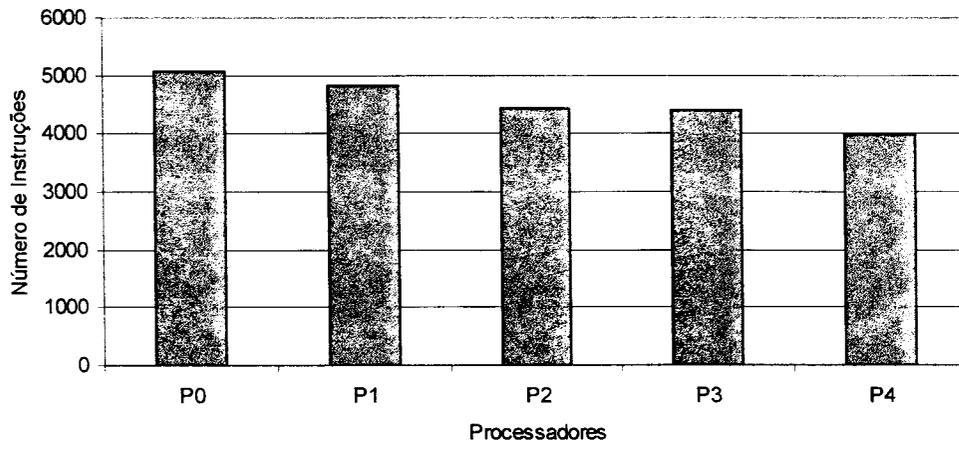


FIGURA A.4 - Distribuição de Carga da arquitetura APO para controle TDM com Chave 3 para Gauss

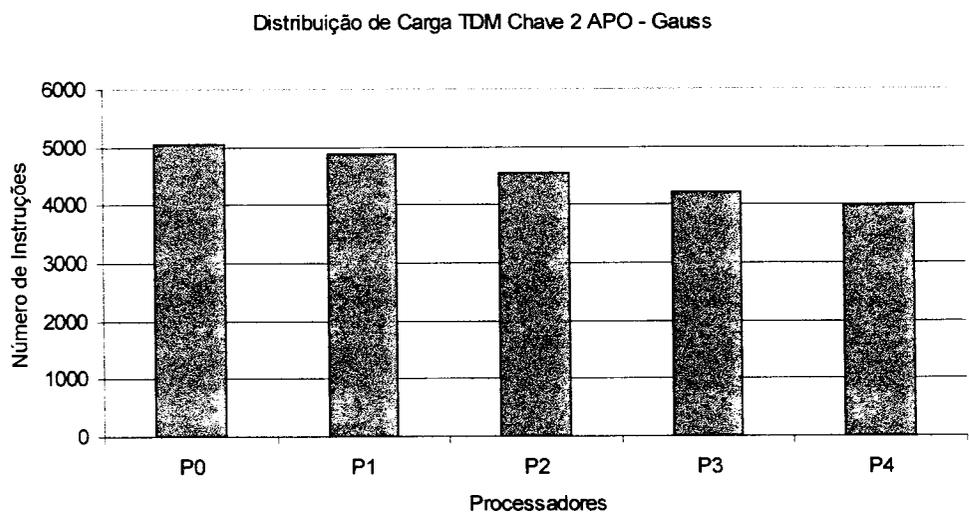


FIGURA A.5 - Distribuição de Carga da arquitetura APO para controle TDM com Chave 2 para Gauss

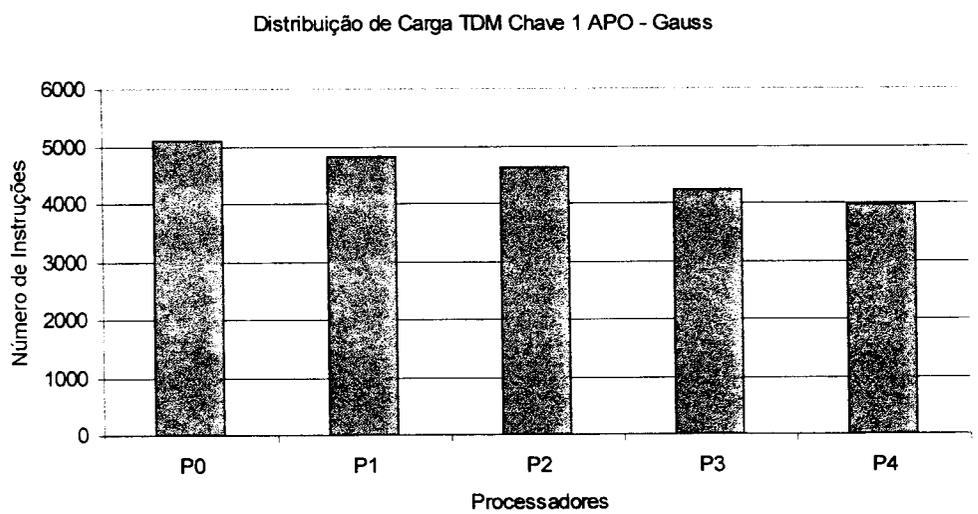


FIGURA A.6 - Distribuição de Carga da arquitetura APO para controle TDM com Chave 1 para Gauss

## **Anexo B**

### **Comportamento da Chave: Distribuição de Carga para o Protocolo de controle com Nó Central**

#### **B.1 Introdução**

Este anexo apresenta os resultados das distribuições de carga para a chave de interconexão da arquitetura APO com o protocolo de controle com Nó Central para o programa Gauss.

#### **B.2 Distribuição de carga dos processadores para o protocolo de controle com Nó Central**

Os resultados da distribuição de carga da arquitetura APO para o protocolo de controle com Nó Central usando o programa Gauss são mostrados nas figuras de B.1 a B.6.

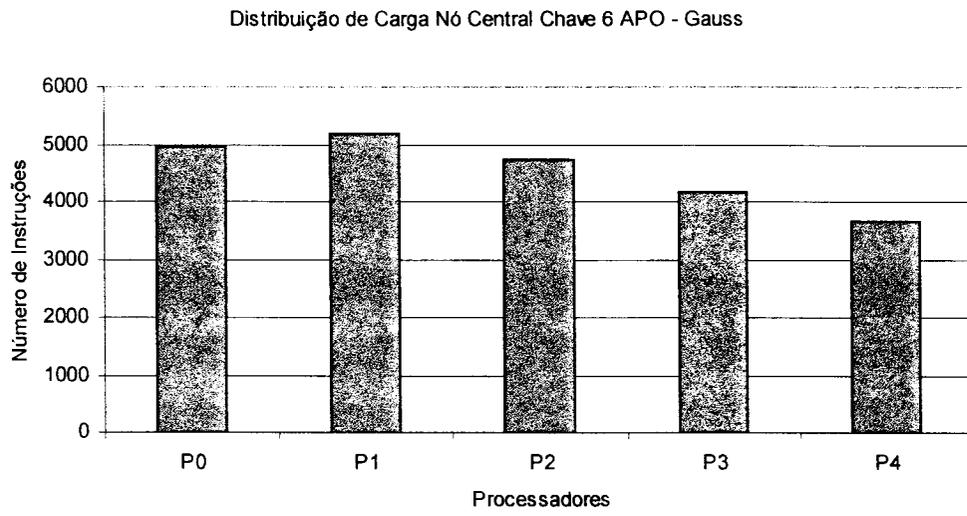


FIGURA B.1 - Distribuição de Carga da arquitetura APO para controle com Nó Central com Chave 6 para Gauss

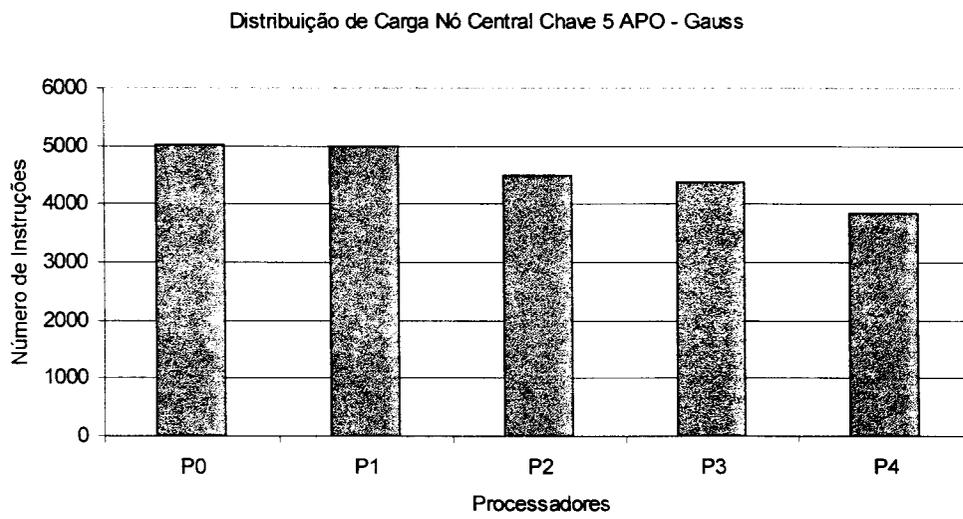


FIGURA B.2 - Distribuição de Carga da arquitetura APO para controle com Nó Central com Chave 5 para Gauss

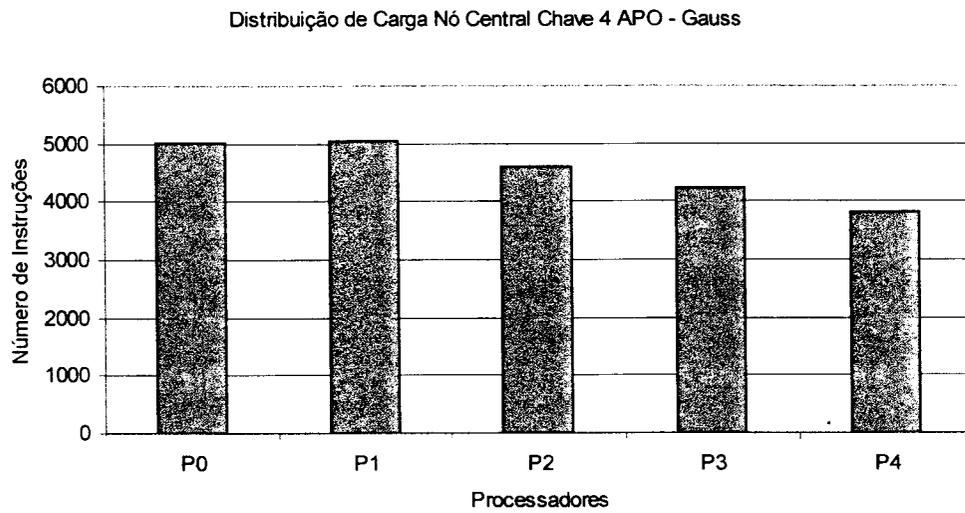


FIGURA B.3 - Distribuição de Carga da arquitetura APO para controle com Nó Central com Chave 4 para Gauss

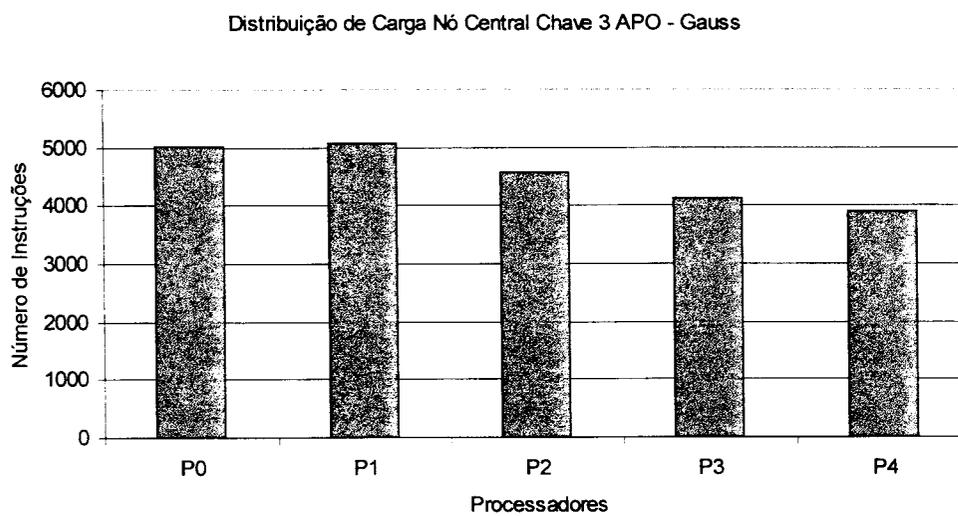


FIGURA B.4 - Distribuição de Carga da arquitetura APO para controle com Nó Central com Chave 3 para Gauss

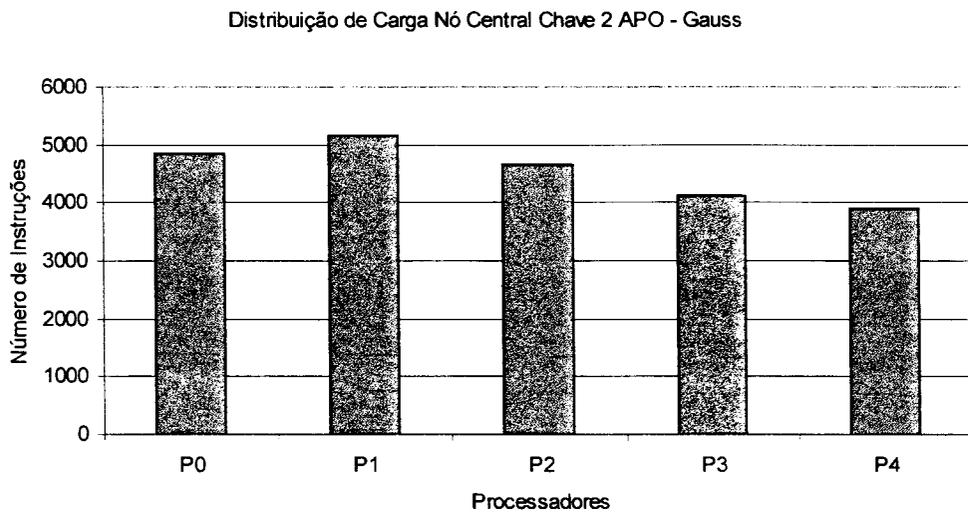


FIGURA B.5 - Distribuição de Carga da arquitetura APO para controle com Nó Central com Chave 2 para Gauss

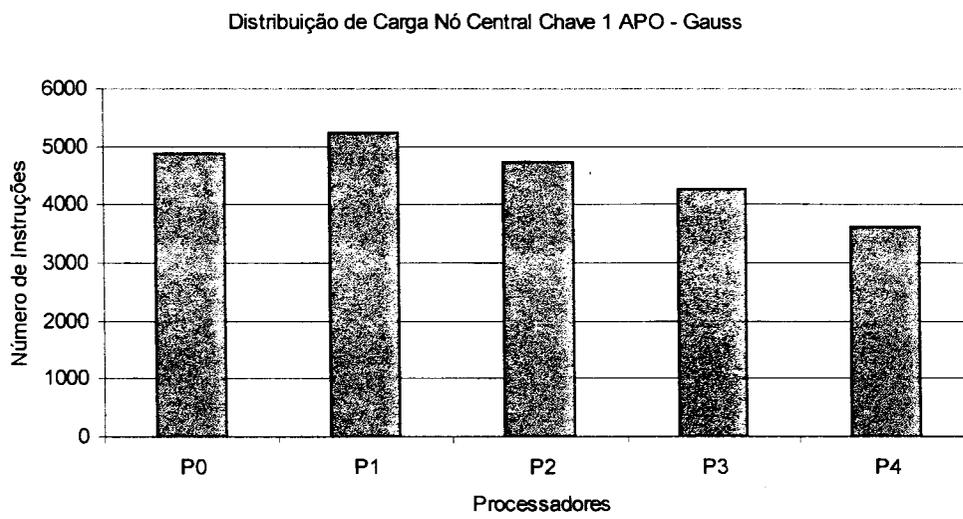


FIGURA B.6 - Distribuição de Carga da arquitetura APO para controle com Nó Central com Chave 1 para Gauss

## Bibliografia

- [1] Dennis, J. B. First Version of a data-flow procedure language. *Lect. Notes Comput. Sc.*, 19, p. 362-376, 1974.
- [2] Veen, Arthur H. Dataflow Machine Architecture. *ACM Computing Surveys*, Vol. 18, N° 4, p. 365-396, December 1986.
- [3] Stone, H. S. e Cocke, J. Computer Architecture in the 1990s. *Computer*, Vol. 24, p. 30-38, September 1991.
- [4] Louri, A. e Sung, H. 3D Optical Interconnects for High-Speed Interchip and Interboard Communications. *IEEE Computer*, Vol. 27, N° 10, , p. 27-37, October 1994.
- [5] Martini, João A.. *Estudo de Desempenho de Redes de Interconexão para a Arquitetura a Fluxo de Dados Proto-Wolf*. São Carlos, 1994. 110 p. Dissertação de Mestrado. Instituto de Física de São Carlos, Universidade de São Paulo.
- [6] Garcia Neto, A. e Ruggiero, C. A. The Proto-Architecture WOLF, a Dataflow Supercomputer. In: XVIII Conferencia Latinoamericana de Informatica, Las Palmas de Gran Canaria. Espanha, 1992, p. 531-539.
- [7] Cavenaghi, M. A.; Spolon, R.; Martins, J. E. M. P.; Domingues, S. G., Garcia Neto, A. Using PVM in the Simulation of a Hybrid Dataflow Architecture. In: Third European PVM Conference. Technische Universitaet Muenchen, Munich, Germany, October 1996. *Proceedings* p. 343-346. Lecture Notes in Computer Science.
- [8] Cavenaghi, M. A.; Travieso, G. e Garcia Neto, A. A Simple Mechanism to Deal with Sequential Code in Dataflow Architectures. In: Proceedings of International Conference on High Performance Computing (HiPC), Madras, Índia, 1998, p. 188-193.
- [9] Martini, João A., Garcia Neto, A. e Cavenaghi, M. A.. Influência das redes de interconexão sobre os tempos de execução na arquitetura Wolf. *Acta Scientiarum*, Vol. 20, N° 4, p. 477-481, Dezembro 1998.
- [10] Willner, A. E. Mining the optical bandwidth for a terabit per second. *IEEE Spectrum*, Vol. 34, N° 4, p. 32-41, April 1997.
- [11] Martins, João E. M. P. *Investigação de Técnicas Fotônicas de Chaveamento Aplicadas em Arquiteturas Paralelas*. São Carlos, 1998. 197 p. Tese de Doutorado. Instituto de Física de São Carlos, Universidade de São Paulo.
- [12] Silc, J.; Robic, B. e Ungerer, T. Asynchrony in Parallel Computing: From Dataflow to Multithreading. *Parallel and Distributed Computing Practices*. Volume 1, N° 1, January 1998.

- [13] Dennis, J. B. Data flow supercomputers. *IEEE Computer*, 13, p. 48-56, November 1980.
- [14] Comte, D., Hifdi, N. e Syre, J. C. The data driven LAU multiprocessor system: Results and perspectives. In: Proceedings World Comp. Congress IFIP'80, October 1980, p. 175-180.
- [15] Cornish, M.; Hogan, D. W. e Jensen, J. C. The Texas Instruments distributed data processor. In: Proceedings Louisiana Computer Exposition, March 1979, p. 189-193.
- [16] Davis, A. L. The architecture and system method of DDM1: A recursively structured data driven machine. In: Proceedings 5<sup>th</sup> ISCA, April 1978, p. 210-215.
- [17] Iwashita, M. e Temma, T. Experiments on a data flow machine. In: Proceedings IFIP Conf., 1987, p. 235-245.
- [18] Jeffery, T. The uPD7281 processor. *Byte*, p. 237-246, November 1985.
- [19] Vedder, R.; Campbell, M. e Tucker, G. The Hughes data flow multiprocessor. In: Proceedings 5<sup>th</sup> International Conference Distributed Computing Systems, May 1985, p. 2-9.
- [20] Böhm, A. P. W. e Teo, Y. M. Resource management in a multi-ring dataflow machine. In: Proceedings CONPAR88, September 1988, p. 191-200.
- [21] Gurd, J. R.; Kirkham, C. C. e Watson, I. The Manchester prototype dataflow computer. *Communication of the ACM*, 28, p. 34-52, 1985.
- [22] Inagami, Y. e Foley, J. F. The specification of a new Manchester dataflow machine. In: Proceedings 1989 International Conference Supercomputing, June 1989, p. 371-380.
- [23] Arvind e Nikhil, R. S. Executing a program on the MIT tagged-token dataflow architecture. *IEEE Transactions on Computers*, C-39, p. 300-318, 1990.
- [24] Hiraki, K.; Sekiguchi, S. e Shimada, T. *Status report of SIGMA-1: A dataflow supercomputer*. (Advanced Topics in Data-Flow Computing), J. -L. Gaudiot e L. Bic. Eds., Prentice-Hall, p. 207-223, 1991.
- [25] Takahashi, N. e Amamiya, M. A data flow processor array system: Design and analysis. In: Proceedings 10<sup>th</sup> ISCA, June 1983, p. 243-250.
- [26] Nishikawa, H. et al. Architecture of a one-chip data-driven processor: Q-p . In: Proceedings 1987 ICPP, August 1987, p. 319-326.
- [27] Kishi, M.; Yasuhara, H. e Kawamura, Y. DDDP: A distributed data driven processor. In: Proceedings 10<sup>th</sup> ISCA, June 1983, p. 236-242.

- [28] Ito, N.; Sato, M.; Kuno, E. e Rokusawa, K. The architecture and preliminary evaluation results of the experimental parallel inference machine PIM-D. In: Proceedings 13<sup>th</sup> ISCA, June 1986, p. 149-156.
- [29] Narasimhan, V. L. e Downs, T. Operating system features of a dynamic dataflow array processing system (PATTSY). In: Proceedings 3<sup>rd</sup> Annual Parallel Processing Symposium, March 1989, p. 722-740.
- [30] Hicks, J. et al. Performance Studies of Id on the Monsoon Dataflow System. *Journal of Parallel and Distributed Computing*, (18): 273-300, 1993.
- [31] Shaw, A.; Arvind e Johnson, R. P. Performance tuning scientific codes for dataflow execution. In: Proceedings PACT'96, October 1996, p. 198-207.
- [32] Treleaven, P. C.; Hopkins, R. P. e Rautenbach, P. W. Combining data flow and control flow computing. *Computer Journal*, 25, 2, p. 207-217, February 1982.
- [33] Silc, J. e Robic, B. MADAME -- Macro-dataflow machine. In: Proceedings MELECON'91, May 1991, p. 985-988.
- [34] Brailsford, D. F. e Duckworth, R. J. The MUSE machine -- an architecture for structured data flow computation. *New Generation Computing*, 3, p.181-195, 1985.
- [35] Barkhordarian, S. RAMPS: A realtime structured small-scale data flow system for parallel processing. In: Proceedings 1987 ICPP, August 1987, p. 610-613.
- [36] Veen, A. H. e van den Born, R. The RC compiler for the DTN dataflow computer. *Journal Parallel and Distributed Computing*, 10, p. 319-322, 1990.
- [37] Sérot, J.; Quénot, G. M. e Zavidovique, B. A functional data-flow architecture dedicated to real-time image processing. *IFIP Trans.*, A-23, p. 129-140, 1993.
- [38] Grafe, V. G. e Hoch, J. E. The Epsilon-2 multiprocessor system. *Journal of Parallel and Distributed Computing*, 10 (4), p. 309-318, 1990.
- [39] Sohn, A.; Kim, C. e Sato, M. Multithreading with the EM-4 distributed-memory multiprocessor. In: Proceedings IFIP WG 10.3 Working Conf. Parallel Arch. and Compilation Techniques, June 1995, p. 27-36.
- [40] Culler, D. E.; Goldstein, S.; Schauser, K. E. e von Eicken, T. TAM: A compiler controlled threaded abstract machine. *Journal of Parallel and Distributed Computing*, 18 (12), p. 347-370, 1993.

- [41] Beckerle, M. J. Overview of the START (\*T) multithreaded computer. In: Proceedings COMPCON'93, 1993, p. 148-156.
- [42] Moore, R.; Klauer, B. e Waldschmidt, K. A Combined Virtual Shared Memory and Network that Schedules. *International Journal of Parallel and Distributed Systems and Networks*, Vol. 1, N<sup>o</sup>. 2, p. 51-56, 1998.
- [43] Roh, L. e Najjar, W. A. Design of a storage hierarchy in multithreaded architectures. In: Proceedings MICRO-28, 1995, p. 271-278.
- [44] Nemawarkar, S. S. e Gao, G. R. Measurement and Modeling of EARTH-MANNA Multithreaded Architecture. In: Proceedings of the 4<sup>th</sup> International Workshop on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS'96), San Jose, California, IEEE Computer Society, February 1-3, 1996, p. 109-114.
- [45] Ungerer, T. e Zehendner, E. Threads and subinstruction level parallelism in a data flow architecture. *Lecture Notes Comput. Sc.*, 634, p. 731-736, 1992.
- [46] Heath, J. R. e Sivanesa, B. Development, Analysis, and Verification of a Parallel Hybrid Dataflow Computer Architectural Framework and Associated Load-Balancing Strategies and Algorithms via Parallel Simulation. *Simulation*, 69:1, p. 007-025, July 1997.
- [47] Magna, P. *Proposta e Simulação de uma Arquitetura a Fluxo de Dados de Segunda Geração*. São Carlos, 1997. 143 p. Tese de Doutorado. Instituto de Física de São Carlos, Universidade de São Paulo.
- [48] Wu, C. e Lee, M. Performance Analysis of Multistage Interconnection Network Configurations and Operations. *IEEE Transactions on Computers*, Vol. 41, N<sup>o</sup> 1, p. 18-27, January 1992.
- [49] Bhuyan, L. N. Interconnection Networks for Parallel and Distributed Processing. *IEEE Computer*, p. 9-12, June 1987.
- [50] Bhuyan, L. N.; Yang, Q. e Agrawal, D. P. Performance of Multiprocessor Interconnection Networks. *IEEE Computer*, Vol. 22, N<sup>o</sup> 2, p. 25-37, February 1989.
- [51] Almasi, G. S. e Gottlieb, A. *Highly Parallel Computing*. 390 Bridge Parkway, Redwood City, CA 94065, The Benjamin Cummings Publishing Company Inc., 1989.
- [52] Ibbet, R. N. e Topham, N. P. *Architecture of High Performance Computers*. 1<sup>a</sup> Edição (Capítulo 3: Interconnection Networks), p. 22-42, Macmillan Education LTD, 1989.
- [53] Lee, B. e Hurson, A. R.. Dataflow Architectures and Multithreading. *IEEE Computer*, Vol. 27, N<sup>o</sup> 8, p. 27-39, August 1994.

- [54] Patel, J. H. Performance of Processor-Memory Interconnections for Multiprocessors. *IEEE Transactions on Computers*, Vol. C-30, N° 10, p. 771-780, October 1981.
- [55] Wilkinson, B. On Crossbar Switch and Multiple Bus Interconnection Networks with Overlapping Connectivity. *IEEE Transactions on Computers*, Vol. 41, N° 6, p. 738-746, June 1992.
- [56] Tabak, D. e Marczynski, J. Distributed Concentrically Connected Network Architecture. *International Journal of Parallel and Distributed Systems and Networks*, Vol. 1, N° 1, p. 26-33, 1998.
- [57] Sawchuk, A. A. et al. Optical Crossbar Networks. *IEEE Computer*, Vol. 20, N° 6, p. 50-60, June 1987.
- [58] Nowatzky, A. G. e Prucnal, P. R. Are Crossbars Really Dead? The Case for Optical Multiprocessor Interconnect Systems. *Computer Architecture News*, Vol. 23, N° 2, p. 106-115, May 1995.
- [59] Bhuyan, L. N. e Agrawal, D. P. Design and Performance of General Interconnection Networks. *IEEE Transactions on Computers*, Vol. C-32, N° 12, p. 1081-1090, December 1983.
- [60] Wu, C. e Feng, T. The Reverse-Exchange Interconnection Network. *IEEE Transactions on Computers*, Vol. C-29, N° 9, p. 801-811, September 1980.
- [61] Stone, H. S. e Cocke, J. Computer Architecture in the 1990s. *IEEE Computer*, Vol. 24, N° 9, p. 30-38, September 1991.
- [62] Louri, A. e Sung, H. 3D Optical Interconnects for High-Speed Interchip and Interboard Communications. *IEEE Computer*, Vol. 27, N° 10, p. 27-37, October 1994.
- [63] Borella, M. et al. Optical Interconnects for Multiprocessor Architectures Using Wavelength-Division Multiplexing. In: Proceedings of the Twenty-Seventh Annual Hawaii International Conference on System Sciences, 1994, p. 499-508.
- [64] Senior, J. M.; Handley, M. R. e Leeson, M. S. Developments in Wavelength Division Multiple Access Networking. *IEEE Communications Magazine*, Vol. 36, N° 12, p. 28-36, December 1998.
- [65] Borella, M. et al. Optical Components for WDM Lightwave Networks. *Proceedings of the IEEE*, Vol. 85, N° 8, p. 1274-1307, August 1997.
- [66] Garcia Neto, A. *Perspectivas em Arquiteturas a Fluxo de Dados: Implementações Fotônicas e Aplicações Multimídia*. São Carlos, 1995. 203p. Tese (Livre Docência) - Instituto de Física de São Carlos, Universidade de São Paulo.

- [67] Ryan, J. P. WDM: North American Deployment Trends. *IEEE Communications Magazine*, Vol. 36, N° 2, p. 40-44, February 1998.
- [68] Ramaswami, R. e Sivarajan, K. N. *Optical Networks: A Practical Perspective*. 1ª Edição (Capítulo 13). San Francisco, California, Morgan Kaufmann Publishers, Inc., 1998.
- [69] Kiniry, J. R. Wavelength Division Multiplexing: Ultra High Speed Fiber Optics. *IEEE Internet Computing*, p. 13-15, March/April 1998
- [70] Shi, H. et al. 20x5 Gbit/s optical WDM transmitter using single-stripe multiwavelength modelocked semiconductor laser. *Electronics Letters*, Vol. 34, N° 2, p. 179-181, January 22, 1998.
- [71] Taga, H. et al. 50 WDM, 10.66 Gbit/s transmission experiment over 1600 Km employing grain-flattened EDFA chain. *Electronics Letters*, Vol. 34, N° 4, p. 382-383, February 19, 1998.
- [72] Terahara, T. et al. 0.7 Tbit/s (66x10.66 Gbit/s) WDM transmission over 2212 Km using broadband, high-power EDFA with pump reflector. *Electronics Letters*, Vol. 34, N° 10, p. 1001-1002, May 14, 1998.
- [73] Aisawa, S. et al. Ultra-wideband, long distance WDM demonstration of 1 Tbit/s (50x20 Gbit/s), 600 Km transmission using 1550 and 1580 nm wavelength bands. *Electronics Letters*, Vol. 34, N° 11, p. 1127-1129, May 28, 1998.
- [74] Chen, C. D. et al. 1.2 Tbit/s (30 ch x 40 Gbit/s) WDM transmission over 85 Km fibre. *Electronics Letters*, Vol. 34, N° 10, p. 1002-1004. May 14, 1998.
- [75] Mukherjee, B. WDM-Based Lightwave Local Networks. Part I: Single-Hop Systems. *IEEE Network*, V. 6, N° 3, p. 12-27, May 1992.
- [76] Webb, B. e Louri, A. A Free Space Optical Crossbar Switch Using Wavelength Division Multiplexing and Vertical-Cavity Surface-Emitting Lasers. In: Proceedings of The Fifth International Conference on Massively Parallel Processing Using Optical Interconnections. June 1998, Las Vegas, p 50-57.
- [77] Mukherjee, B. WDM-Based Lightwave Local Networks. Part II: Multihop Systems. *IEEE Network*, V. 6, N° 4, p 20-32, July 1992.
- [78] **Martini, João A. et al.** Proposal of a Dataflow Architecture with Optical Interconnection Network. In: Proceedings of High Performance Computing Symposium - HPC'99, 1999, San Diego, California. SCS Society for Computer Simulation International. p. 377-380.
- [79] Partridge, C. *Gigabit Networking*. 2ª Edição (Capítulo 2). Massachusetts, USA, Addison-Wesley Publishing Company, Inc., 1993.

- [80] Cavenaghi, M. A. *Implementação e Estudo da Arquitetura a Fluxo de Dados Wolf*. São Carlos, SP, 1997. 136 p. Tese de Doutorado. Instituto de Física de São Carlos, Universidade de São Paulo.
- [81] McGraw, J. R. SISAL: Streams and Iterations in a Single Assignment Language. Language Reference Manual, LLNL, Tech. Report, M-14, 1985.