Capítulo 6

Processadores Paralelos do Cliente à Nuvem

Introdução

- Objetivo: conectar vários computadores para obter melhor desempenho
  - Multiprocessadores
  - Escalabilidade, disponibilidade, eficiência energética
- Paralelismo nível de tarefa (nível de processo)
  - Utilizar vários processadores executando programas independentes simultaneamente
- Programa de processamento paralelo
  - Um único programa que é executado em vários processadores simultaneamente

Introdução

- Cluster
  - Um conjunto de computadores conectados por uma rede local (LAN) que funciona como um único e grande multiprocessador
- Microprocessador multicore
  - Chips com múltiplos processadores (cores)
- Processador de memória compartilhada (SMP)
  - SMP - Shared Memory Processors
  - Um processador paralelo com um único espaço de endereços

Hardware e Software

- Hardware
  - Serial: ex., Pentium 4
  - Paralelo: ex., Intel Core 7
- Software
  - Sequencial: ex., multiplicação de matriz
  - Concorrente: ex., Sistema operacional
- Software sequencial/concorrente pode ser executado no hardware serial/paralelo
  - Desafio: criar programas eficientes para processamento paralelo

Hardware e Software

		Software	
		Sequential	Concurrent
Hardware	Serial	Matrix Multiply written in MatLab running on an Intel Pentium 4	Windows Vista Operating System running on an Intel Pentium 4
	Parallel	Matrix Multiply written in MATLAB running on an Intel Core i7	Windows Vista Operating System running on an Intel Core i7

FIGURE 6.1 Hardware/software categorization and examples of application perspective on concurrency versus hardware perspective on parallelism.

Hardware e Software

- Exemplo de Software
  - Os programadores de compiladores pensam neles como programas sequenciais: as etapas são análise léxica, geração de código, otimização e assim por diante
  - Ao contrário, os programadores de sistemas operacionais normalmente pensam neles como programas concorrentes: processos em cooperação tratando de eventos de E/S devido as tarefas independentes executando em um computador

Hardware e Software

		Software	
		Sequential	Concurrent
Hardware	Serial	Matrix Multiply written in MatLab running on an Intel Pentium 4	Windows Vista Operating System running on an Intel Pentium 4
	Parallel	Matrix Multiply written in MATLAB running on an Intel Core i7	Windows Vista Operating System running on an Intel Core i7

FIGURE 6.1 Hardware/software categorization and examples of application perspective on concurrency versus hardware perspective on parallelism.

Programação Paralela

- O software paralelo é o problema
  - É difícil escrever software que usa processadores múltiplos para completar uma tarefa mais rápido
  - O problema fica pior à medida que o número de processadores aumenta
- Dificuldades
  - Particionamento
  - Coordenação
  - Sobrecarga de comunicações

Programação Paralela

- Como uma analogia, suponha que a tarefa fosse escrever um artigo de jornal
  - Oito repórteres trabalhando no mesmo artigo poderiam potencialmente escrever um artigo oito vezes mais rápido
  - Para conseguir essa velocidade aumentada, seria preciso desmembrar a tarefa de modo que cada repórter tivesse algo para fazer ao mesmo tempo
  - Assim, temos de *escalonar* as subtarefas
    - Se algo saísse errado e apenas um repórter levasse mais tempo do que os sete outros levaram, então o benefício de ter oito escritores seria diminuído

Programação Paralela

- Outro perigo seria se os repórteres tivessem de gastar muito tempo falando uns com os outros para escrever suas seções
- Você também se atrasaria se uma parte do artigo, como a conclusão, não pudesse ser escrita até que todas as outras partes fossem concluídas
  - Assim, deve-se ter o cuidado para *reduzir o overhead de comunicação e sincronização*

- Para essa analogia e para a programação paralela, os desafios incluem
  - Escalonamento
  - Balanceamento de carga
  - Tempo para sincronismo e overhead para comunicação entre as partes

Fluxos de Instrução e Dados

- Uma categorização do hardware paralelo proposta na década de 1960 ainda está em uso atualmente
  - Ela foi baseada no número de fluxos de instruções e no número de fluxos de dados
  - SISD: Um processador converncional tem um único fluxo de instruções e um único fluxo de dados
  - MIMD: Um multiprocessador possui fluxos de instruções e dados múltiplos

Fluxos de Instrução e Dados

- Uma classificação alternativa

		Fluxo de Dados	
		Único	Multiplo
Fluxo de instruções	Único	**SISD**: Intel Pentium 4	**SIMD**: SSE instructions of x86
	Múltiplo	**MISD**: Não há exemplos atuais	**MIMD**: Intel Core i7

- SISD: Single Instruction stream, Single Data stream
  - Um processador único
- MIMD: Multiple Instruction streams, Multiple Data streams
  - Um multiprocessador

Fluxos de Instrução e Dados

- Uma classificação alternativa

		Fluxo de Dados	
		Único	Multiplo
Fluxo de instruções	Único	**SISD**: Intel Pentium 4	**SIMD**: SSE instructions of x86
	Múltiplo	**MISD**: Não há exemplos atuais	**MIMD**: Intel Core i7

- SPMD: Single Program Multiple Data
  - Modelo de programação MIMD convencional, um único programa é executado em todos os processadores
- SIMD: Single Instruction stream, Multiple Data streams
  - A mesma instrução é aplicada a muitos fluxos de dados, assim como em um processador de vetor

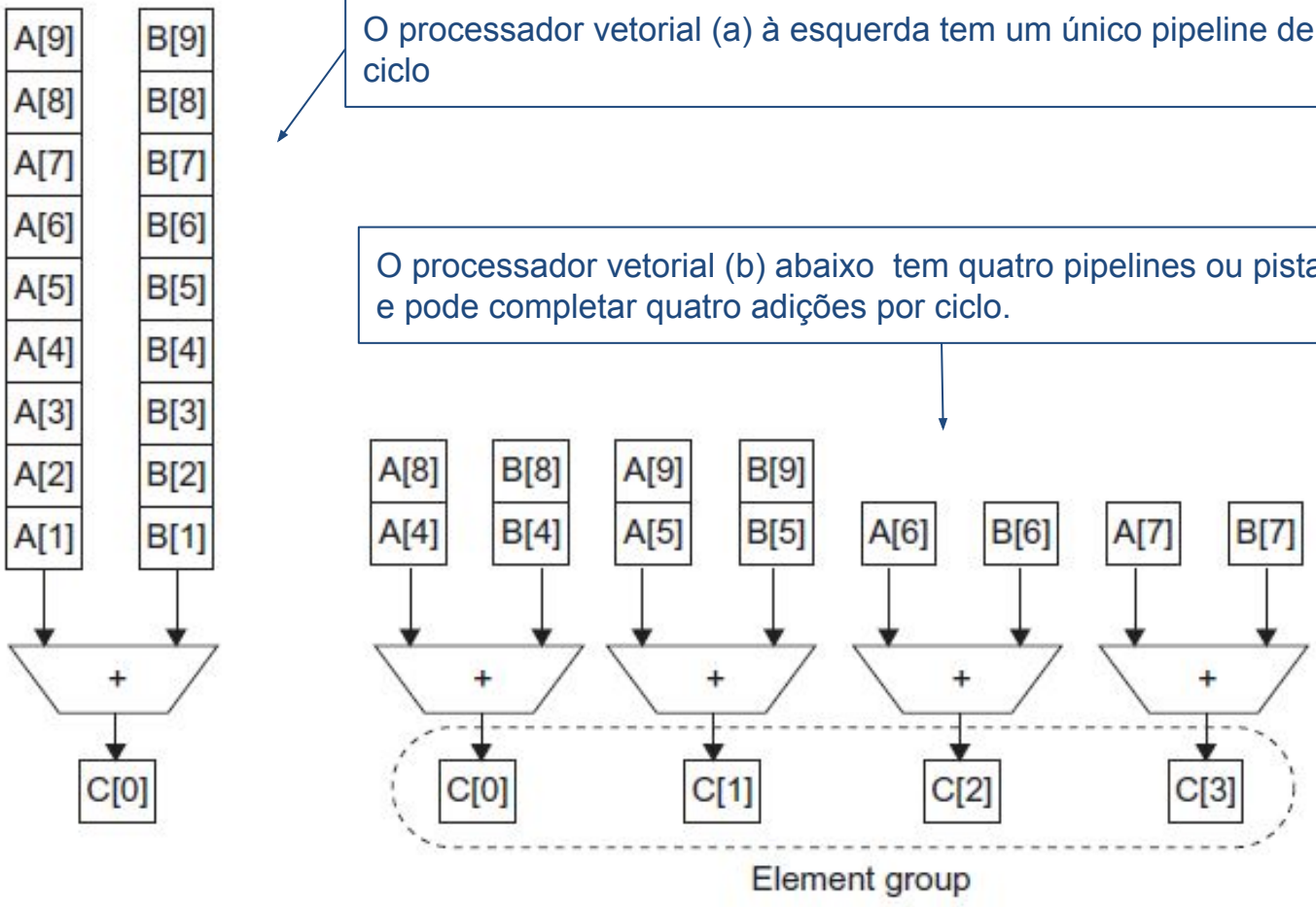
Processadores Vetoriais

- Mais antiga e mais elegante do SIMD é a chamada *arquitetura de vetor*
- Também é uma grande combinação de problemas com muito paralelismo em nível de dados
- Unidades de função altamente pipeline
- Transmita dados de/para registradores de vetor para unidades
  - Dados coletados da memória em registradores
  - Resultados armazenados dos registradores na memória

Precessadores Vetoriais

O processador vetorial (a) à esquerda tem um único pipeline de adição por ciclo

O processador vetorial (b) abaixo tem quatro pipelines ou pistas de adição e pode completar quatro adições por ciclo.



SIMD

- Opera elemento a elemento em vetores de dados
  - Ex., instruções MMX e SSE em x86
    - Vários elementos de dados em registros de 128 bits
- Todos os processadores executam a mesma instrução ao mesmo tempo
  - Cada um com diferentes endereços de dados, etc.
- Simplifica a sincronização
- Hardware de controle de instrução reduzido
- unciona melhor para aplicativos altamente paralelos de dados

Multithreading

- Multithreading do hardware
  - Aumentar a utilização de um processador trocando para outra thread quando uma thread é suspensa
- Threading
  - Uma thread inclui o contador de programa, o estado do registrador e a pilha
  - Ela é um processo simplificado
    - Enquanto as threads normalmente compartilham um único espaço de endereços
    - O processo não faz isso
- Processo
  - Um processo inclui uma ou mais threads, o espaço de endereços e o estado do SO

Multithreading

- Executando vários threads de execução em paralelo
  - Replicar registradores, PC, etc.
  - Troca rápida entre trheads
- Multithreading fine-grained
  - Trocar tópicos após cada ciclo
  - Execução da instrução intercalar
  - Se um thread travar, outros serão executados
- Multithreading coarse-grained
  - Apenas comuta threads em stall onerosos (por exemplo, falha cache L2)
  - Simplifica o hardware, mas não esconde stall mais curtos (por exemplo, hazards de dados)

Simultaneous Multithreading (SMT)

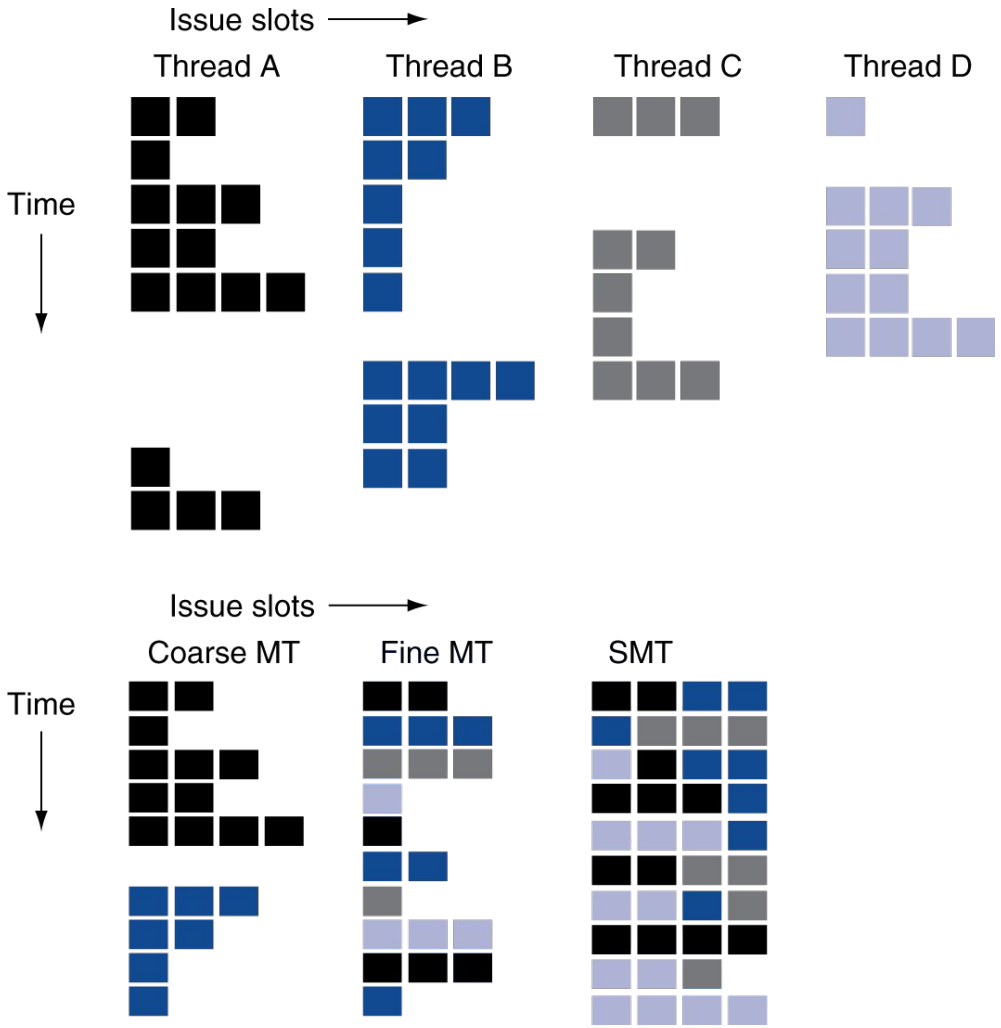
- Em processador de despacho múltiplo, escalonado dinamicamente
  - Despacha instruções de várias threads
  - As instruções de threads independentes são executadas quando as unidades de função estão disponíveis
  - Dentro de threads, dependências tratadas pelo escalonamento dinâmico e renomeação de registradores
- Exemplo: Intel Pentium-4 HT
  - Duas threads: registradores duplicados, unidades funcionais compartilhadas e caches

Exemplo Multithreading

As quatro threads mostram como cada uma seria executada em um processador superescalar padrão sem suporte a multithreading

Os três exemplos mostram como elas seriam executadas juntas em três opções de multithreading

A dimensão vertical representa uma sequência dos ciclos de clock.
Uma caixa vazia (branca) indica que o slot de despacho correspondente está vago nesse ciclo de clock.
Os tons de cinza e preto correspondem a quatro threads diferentes nos processadores multithreading.

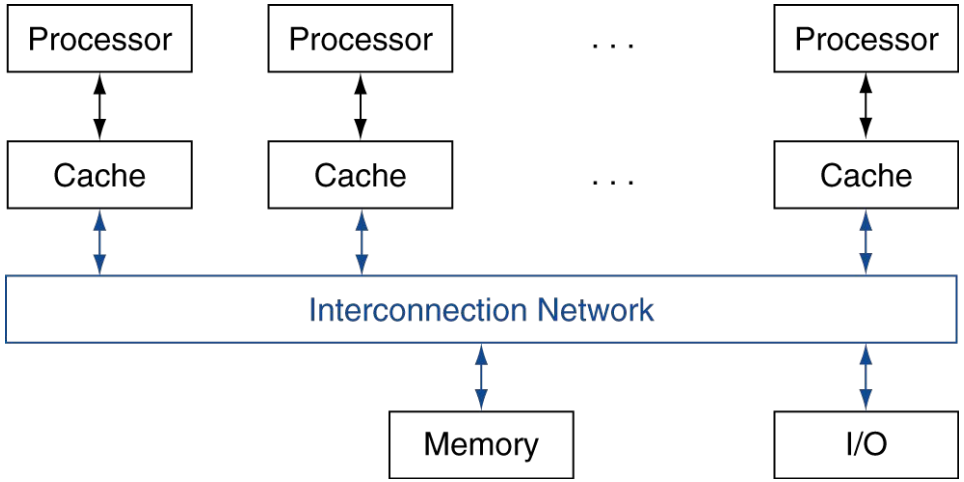


Futuro do Multithreading

- Ele vai sobreviver? De que forma?
- Considerações de energia ⇒ microarquitetura simplificada
  - Formas mais simples de multithreading
- Tolerando a latência de falha de cache
  - A troca de thread pode ser mais eficaz
- Vários núcleos simples podem compartilhar recursos de forma mais eficaz

Memoria Compartilhada

- SMP: multiprocessador de memória compartilhada
  - SMP - Shared Memory Multiprocessor
  - O hardware fornece espaço de endereço físico único para todos os processadores
  - Sincronizar variáveis compartilhadas usando lock
  - É o que quase sempre acontece para os chips multicore



Memoria Compartilhada

- Acesso uniforme à memória (UMA)
  - Um multiprocessador em que a latência a qualquer palavra na memória é aproximadamente a mesma, não importa qual processador solicita o acesso
- Acesso não uniforme à memória (NUMA)
  - Um tipo de multiprocessador com espaço de endereços único em que alguns acessos à memória são muito mais rápidos do que outros, dependendo de qual processador solicita qual palavra

Referências

- Seção 6.1 a 6.5 - Hennessy, J. *Organização e Projeto de Computadores*. [Digite o Local da Editora]: Grupo GEN, 2017. 9788595152908. Disponível em: https://integrada.minhabiblioteca.com.br/#/books/9788595152908/. Acesso em: 22 Oct 2020
- Seção 6.1 a 6.5 - Computer organization and design: the hardware/software interface/David A. Patterson, John L. Hennessy, Elsevier, 5th ed, 2013.