

Terminologias em Programação Paralela e Distribuída

Jones Albuquerque
DFM-UFRPE

2004, Recife - PE.

Sumário

História

Definições

Paralelismo de dados vs. paralelismo de controle

Pipeline

Escalabilidade

História

Concorrência e Paralelismo

Concorrente significa ao mesmo tempo

Sistemas concorrentes originaram-se com os sistemas operacionais em 1962

Surge a necessidade de mecanismos para

- controle de **seção crítica**, sincronização de processos, semáforos e monitores
- programação concorrente

Redes de computadores são introduzidas entre 1975-1985, em 1990 a W3, internet...

Surge a programação distribuída

Programação paralela: distribuída ou concorrente?

Definições

Programa: código fonte, caráter estático

Processo é o programa em execução, caráter dinâmico
(fluxo de controle e dados)

Processamento paralelo é o processamento da informação que enfatiza a manipulação concorrente de dados pertencentes a um ou mais processos de um mesmo problema

Computador paralelo é um computador com múltiplos processadores capazes de realizar processamento paralelo

Definições

Supercomputador é um computador de propósito geral capaz de resolver problemas em alta velocidade quando comparado com outros computadores. Geralmente são computadores paralelos formados por uma pequena quantidade de processadores extremamente “poderosos”

Throughput (vazão) de um dispositivo é o número de resultados produzidos por ele por unidade de tempo

Latência é o tempo absoluto que um dispositivo leva para processar um resultado

Pipeline, técnica que visa melhorar a vazão e não a latência dos resultados

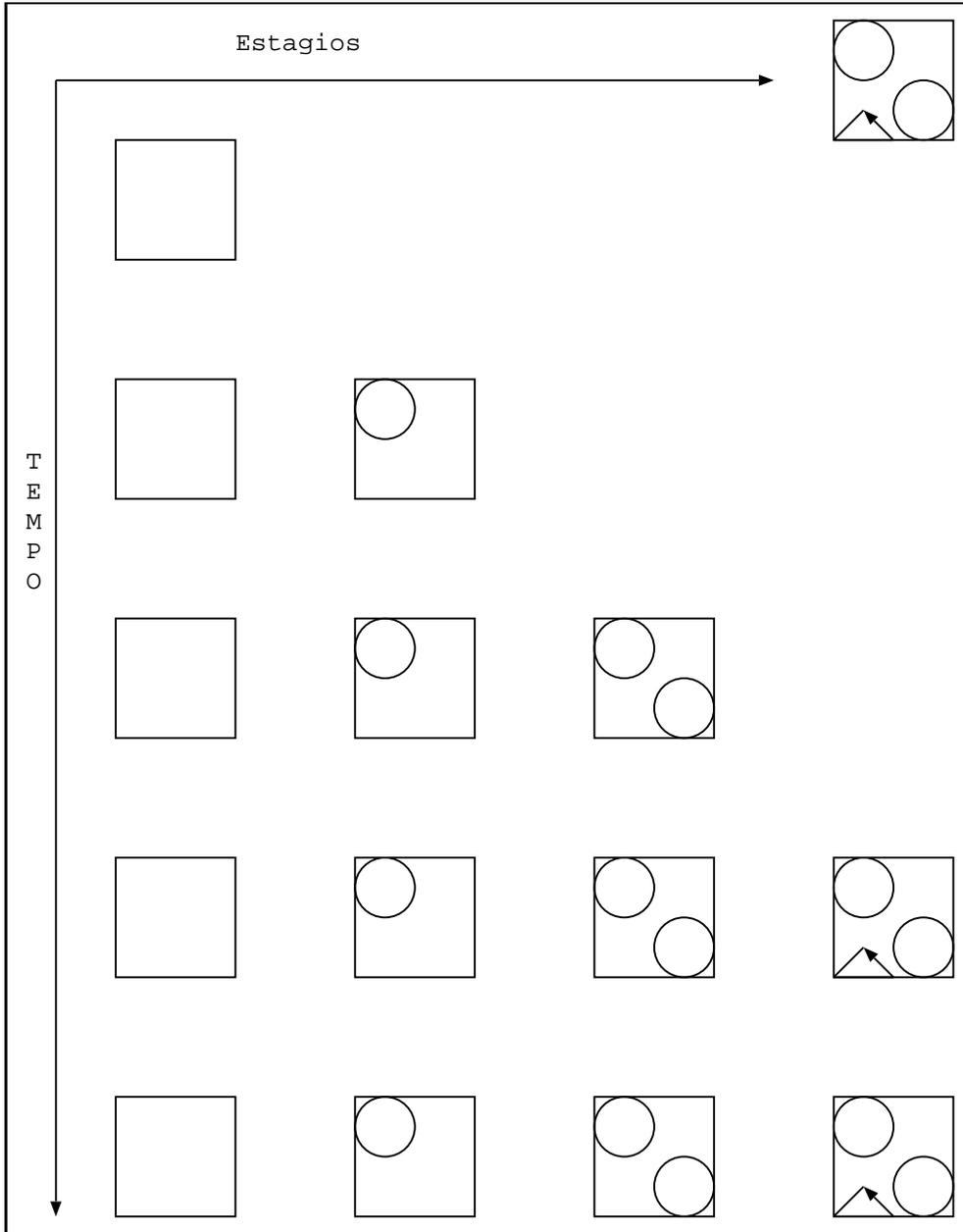
Pipeline

A computação é dividida em passos chamados *segmentos* ou *estágios*

Cada segmento trabalha numa parte da computação

Princípio utilizado em linhas de produção: a saída de um segmento é a entrada do próximo segmento da linha de produção

Exemplo - Pipeline



Parelismo de Controle e de Dados

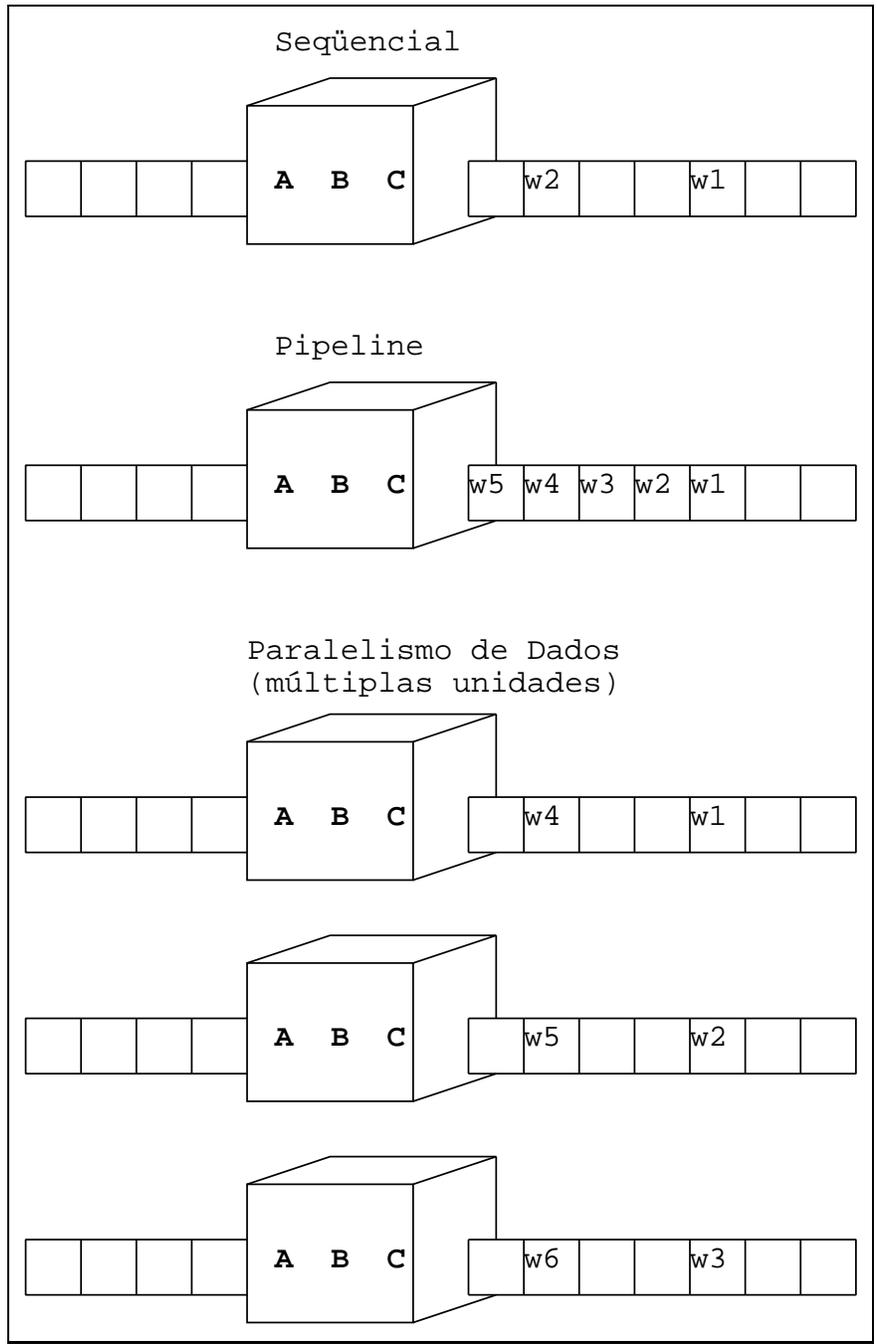
Paralelismo de controle \leftrightarrow pipeline

Paralelismo de dados é o uso de unidades funcionais múltiplas para aplicar a mesma operação simultaneamente aos elementos de uma base de dados. Teoricamente, k -unidades produzem um aumento de vazão de k -vezes no sistema

Speedup é a razão entre o tempo necessário para o algoritmo seqüencial mais eficiente existente realizar uma computação e o tempo necessário para realizar esta mesma computação numa máquina que incorpore pipeline e/ou paralelismo:

$$\text{Speedup} = \frac{\text{tempo sequencial}}{\text{tempo paralelo}}$$

Comparando Paralelismos



Escalabilidade

Um algoritmo é **escalável** se o nível de paralelismo cresce pelo menos linearmente com o tamanho do problema

Um arquitetura é escalável se ela continua rendendo o mesmo desempenho por processador quando o número de processadores aumenta

Algoritmos paralelos em dados são mais escaláveis que algoritmos paralelos em controle: o nível de paralelismo de controle é geralmente uma constante, independente do tamanho do problema!