Parallel Processing

Why Distributed Computing

- Memory Wall + ILP Wall + Power Wall for single processor
  - Memory wall
    - The increasing gap between the processor and memory performance
  - Instruction Level Parallel (ILP) Wall
    - Diminishing returns of more advanced uniprocessor architecture
  - Power Wall

Why Distributed Computing

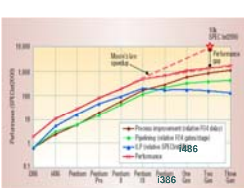


Figure 1. Performance trends for desktop processors. The star indicates the mobile supercomputer's requirement.

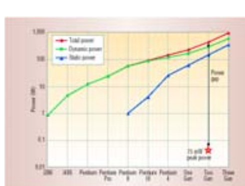
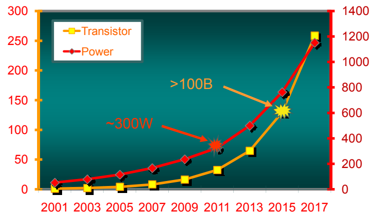


Figure 2. Power trends for desktop processors. The star indicates the mobile supercomputer's requirement.

T. Austin et al., “Mobile Supercomputers”, IEEE Computers, vol 37, No 5 , pp. 81-83, May 2004

The Power/Thermal Challenges



- ***A 300mm² die is capable of integrating over 100 billion transistors in the middle of the next decade***
- ***The power consumption will reach 300 W in early next decade***

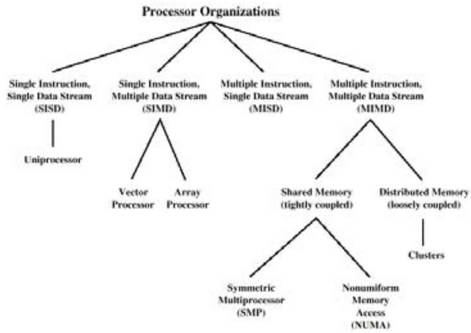
Flynn's Computer Models

- Single instruction, single data stream - SISD
- Single instruction, multiple data stream - SIMD
- Multiple instruction, single data stream - MISD
- Multiple instruction, multiple data stream- MIMD

MIMD

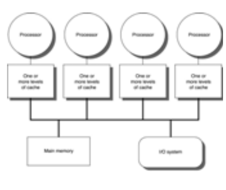
- Flexible
  - One-user high performance platform
  - Run multiple tasks simultaneously
- Cost-effective
  - Using COTS (commercial off-the-shelf) processors
- Exploiting thread-level parallelism
  - Thread

Taxonomy of Parallel Architectures



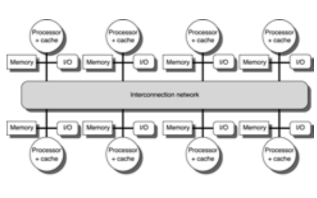
Two Types of MIMDs

- Shared memory architecture
  - Centralized (tightly coupled) shared-memory multiprocessor
    - Single memory
    - Symmetric to all processor (SMP)
  - Distributed shared memory
    - Multiple memory
    - Same memory space

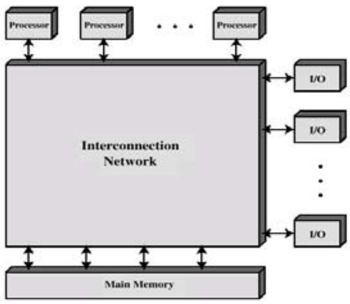


Two Types of MIMD (cont'd)

- Multiprocessor with private distributed memory
  - Multiple (distributed) memory (high memory bandwidth)
  - Memory space is private
  - Communication between processors more costly
    - High bandwidth interconnection network



Symmetric Multiprocessor (SMP)

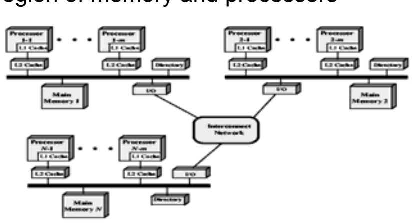


Symmetric Multiprocessor (SMP)

- Similar processors of comparable capacity
- Share same memory and I/O
- Communicate via that shared memory
- Connected by a bus or other internal connection
- Approximately same memory access time
- All processors can perform the same functions (hence symmetric)
- System controlled by integrated operating system
  - providing interaction between processors
  - Interaction at job, task, file and data element levels

Nonuniform Memory Access (NUMA)

- Accessible to all parts of memory
- Access time of memory differs depending on region of memory and processors

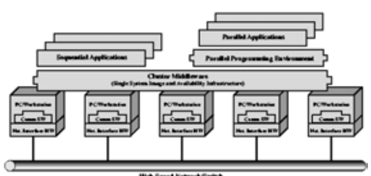


Nonuniform Memory Access (NUMA)

- Effective performance at higher levels of parallelism than SMP
- No major software changes
- Performance can breakdown if too much access to remote memory
- Not as transparent as SMP
- Page allocation, process allocation and load balancing changes needed

Clusters

- Collection of independent computers or SMPs
- Interconnected to work together as unified resource (as if one computer)
- Communication via fixed path or network connections
- Each computer is called a node



SMP, NUMA, and Cluster

- Multiprocessor support to high demand applications.
- SMP is easier to manage and control, but has practical limit to number of processors
- Clustering
  - superior incremental & absolute scalability
  - each node has its own memory
  - apps do not see large global memory
  - Coherence maintained by software not hardware
- NUMA retains SMP flavour while giving large scale multiprocessing

Performance issues of MIMD

- Parallelism of the application
- Communication cost

Example I

- Using 100 processors to achieve a speedup of 80. What fraction of the original computation can be sequential?

Amdahl's law: $\text{Speedup}_{overall} = \frac{\text{execution time}_{old}}{\text{execution time}_{new}} = \frac{1}{(1-\alpha)+\frac{\alpha}{n}}$

$$80 = \frac{1}{(1-\alpha)+\frac{\alpha}{100}} ===> 1-\alpha = 0.25\%$$

The parallelism available in programs limits the performance of distributed computing !

Example II

- Assume:
  - 32-processor MIMD
  - 400ns cycle to access remote memory
  - Processor clock rate 1Ghz.
  - IPC is 2 w/o communication
- What is the IPC if 0.2% of instructions involve communication?

$$CPI = CPI_{ideal} + Avg.\ Comm.\ stalls$$

$$CPI = 1/IPC$$

$$Avg.\ Comm.\ stalls = Comm\ \ rate \times Comm.\ Cost = 0.2\% \times \frac{400ns}{1ns} = 0.8$$

$$IPC = 1/CPI = \frac{1}{0.5+0.8} = \frac{1}{1.3} \qquad speedup = \frac{1.3}{0.5} = 2.6$$

The communication cost limits the efficiency of the performance for distributed computing !

Summary

- Motivations for distributed computing
- Flynn’s computer models
- Two types of MIMDs
  - Shared memory
    - Centralized shared memory, e.g. SMP
    - Distributed shared memory, e.g. NUMA
  - Private memory, e.g. cluster
- Performance of MIMD
  - Parallelism
  - Communication cost