Introduction

What is Parallel Architecture?

Why Parallel Architecture?

Evolution of Parallel Architectures

What is Parallel Architecture?

A parallel computer is a collection of processing elements that cooperate to solve large problems fast

Design Questions:

- Resource Allocation:
  - how large a collection?
  - how powerful are the elements?
  - how much memory?
- Data access, Communication and Synchronization
  - how do the elements cooperate and communicate?
  - how are data transmitted between processors?
  - what are the abstractions and primitives for cooperation?
- Performance and Scalability
  - how does it all translate into performance?
  - how does it scale?

Why Parallel?

Application demands: Our insatiable need for computing cycles

- *Scientific computing*: CFD, Biology, Chemistry, Physics, ...
- *General-purpose computing*: Video, Graphics, CAD, Databases, TPC…

Technology Trends

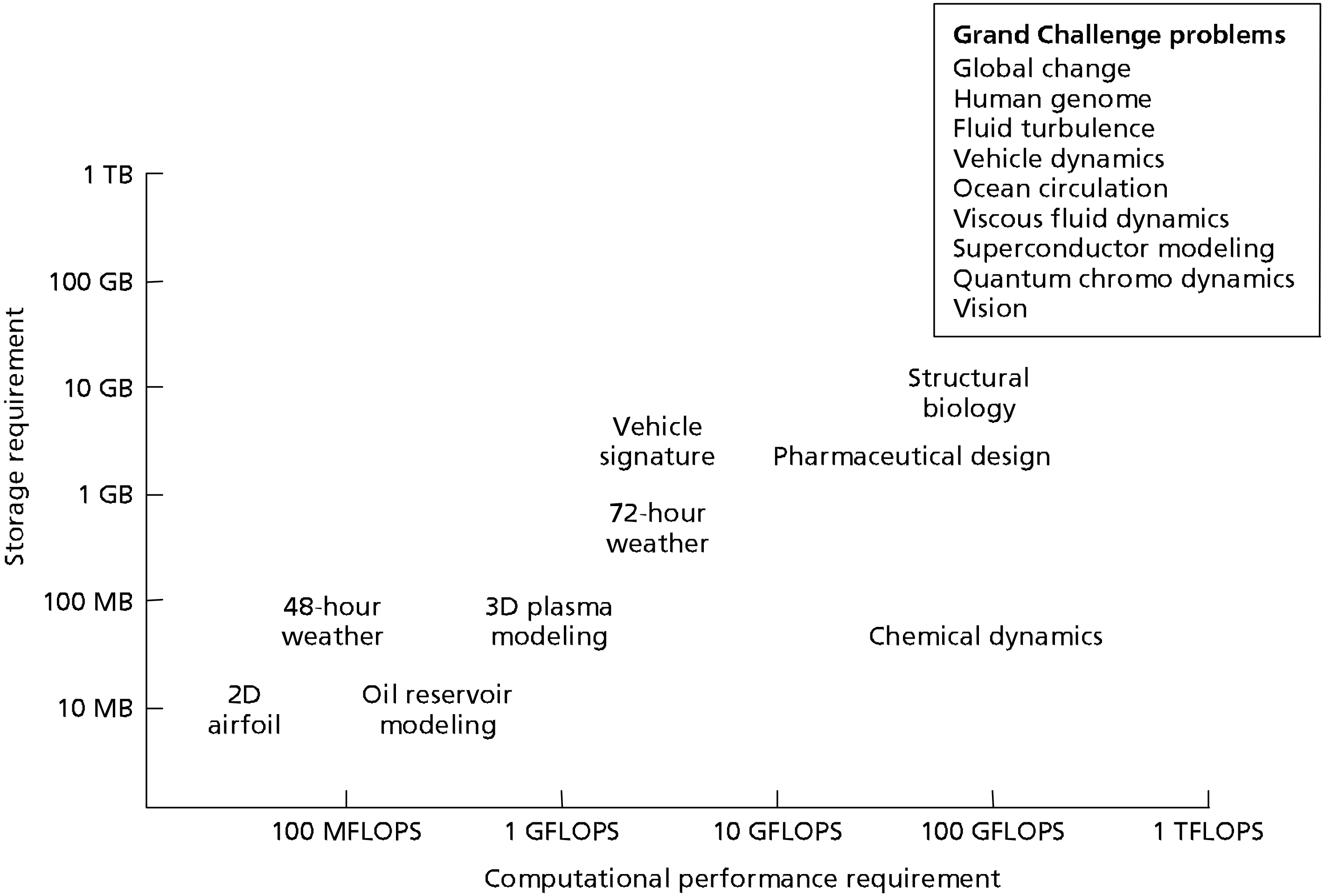
- Denser Integration
- Clock rates are flattening

Architecture Trends

- Instruction-level parallelism valuable but limited
- Coarser-level parallelism (thread/process level) the most viable approach

Economics

Scientific Computing Demand



Grand Challenge problems
Global change
Human genome
Fluid turbulence
Vehicle dynamics
Ocean circulation
Viscous fluid dynamics
Superconductor modeling
Quantum chromo dynamics
Vision

Industries

Large parallel machines a mainstay in many industries

- Petroleum (reservoir analysis)
- Automotive (crash simulation, drag analysis, combustion efficiency),
- Aeronautics (airflow analysis, engine efficiency, structural mechanics, electromagnetism),
- Computer-aided design
- Pharmaceuticals (molecular modeling)
- Visualization
  - in all of the above
  - entertainment (films like Toy Story)
  - architecture (walk-throughs and rendering)
- Financial modeling (yield and derivative analysis)
- etc.

Commercial Computing

Also relies on parallelism for high end

- Scale not so large, but use much more wide-spread
- Computational power determines scale of business that can be handled

Databases, online-transaction processing, decision support, data mining, data warehousing ...

TPC benchmarks (TPC-C order entry, TPC-D decision support)

- Explicit scaling criteria provided
- Size of enterprise scales with size of system
- Problem size no longer fixed as *p* increases, so throughput is used as a performance measure (transactions per minute or *tpm*)

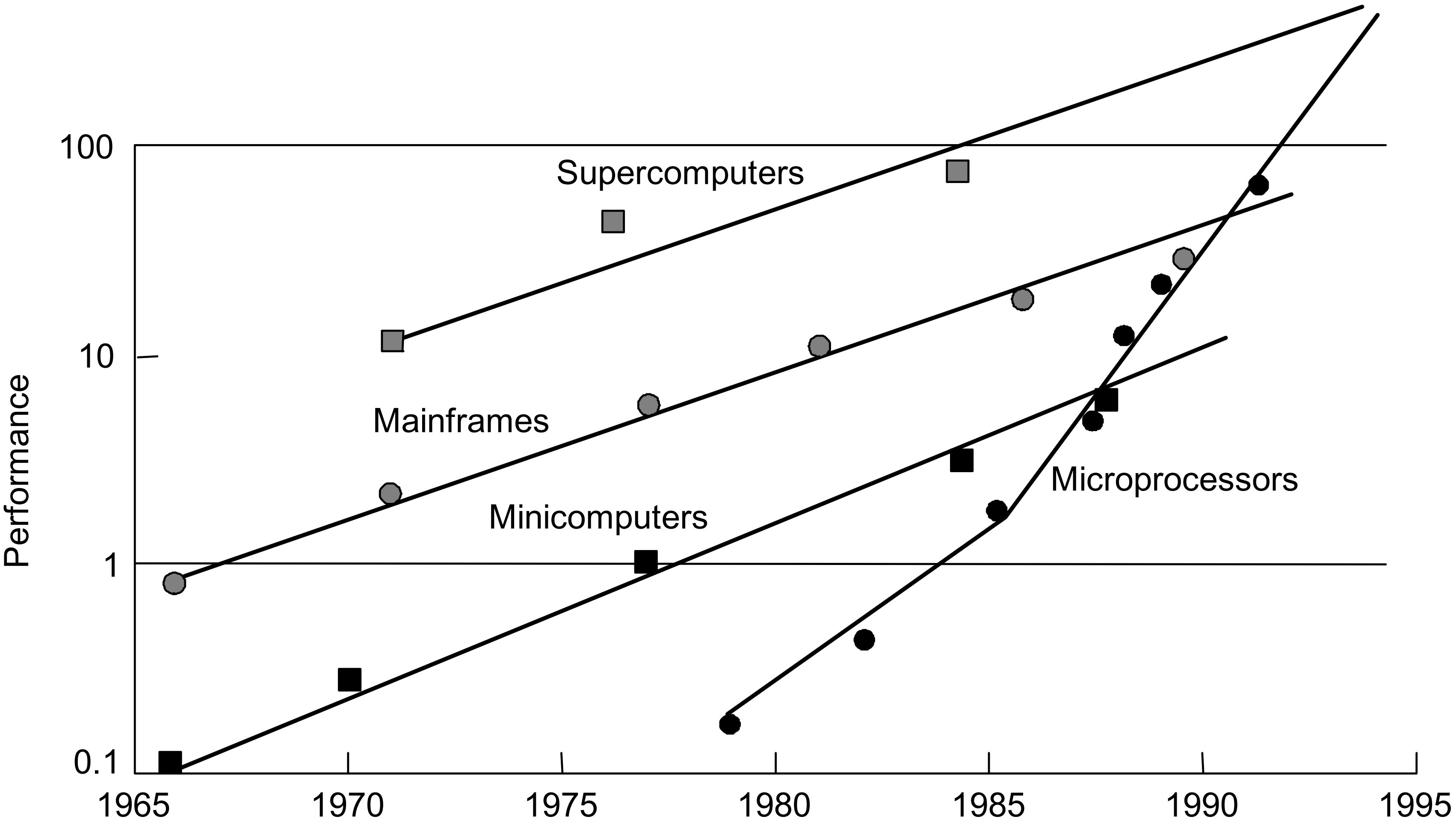
Cloud Computing

Amazon EC2, Microsoft Azure, ….

Both commercial and scientific applications

Memcached, Map-Reduce, …. (are all parallel!)

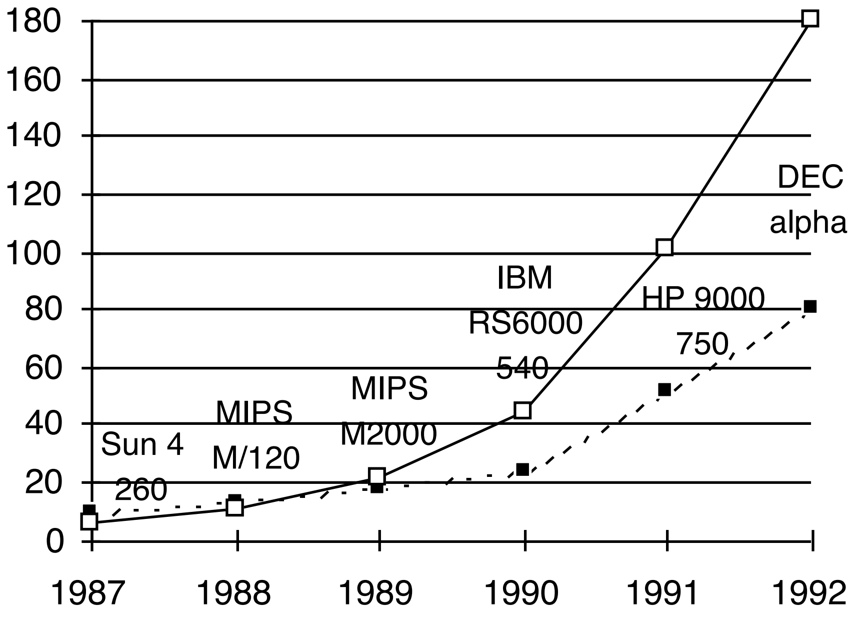
Technology Trends

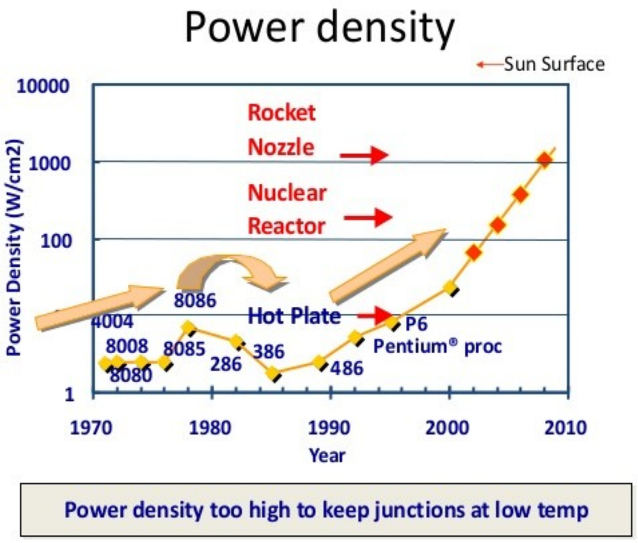


The natural building block for multiprocessors is now also about the fastest!

General Technology Trends

- *Microprocessor performance* increases 50% - 100% per year
- *Transistor count* doubles every 3 years
- *DRAM size* quadruples every 3 years
- Huge investment per generation is carried by huge commodity market





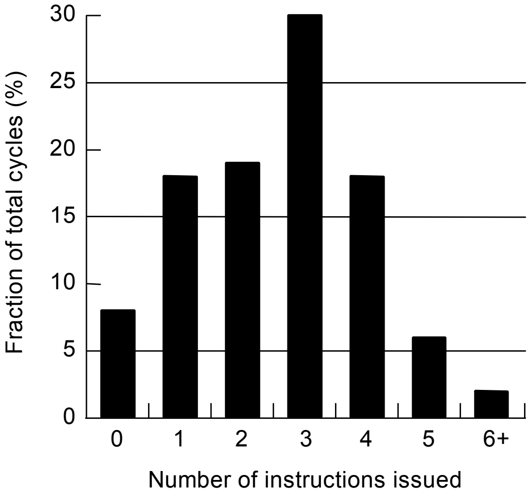
Courtesy, Intel

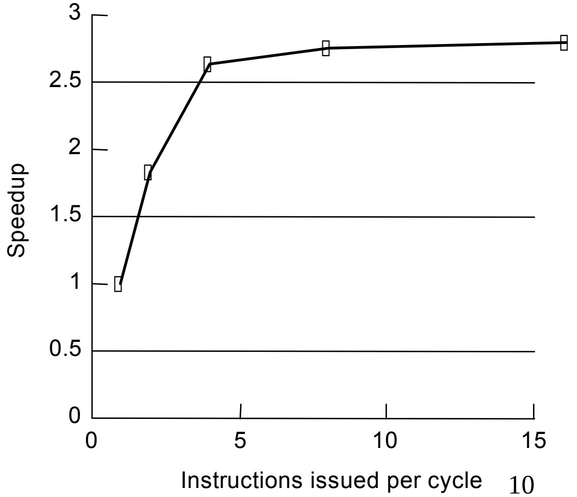
- Not that single-processor performance is plateauing, but that parallelism is a natural way to improve it.

Architectural Trends

Greatest trend is increase in parallelism

- Up to 1985: bit level parallelism: 4-bit -> 8 bit -> 16-bit
  - slows after 32 bit
  - adoption of 64-bits now under way, 128-bit far (not performance issue)
- Mid 80s to mid 90s: instruction level parallelism
  - pipelining and simple instruction sets, + compiler advances (RISC)
  - on-chip caches and functional units => superscalar execution
  - greater sophistication: out of order execution, speculation, prediction
    - to deal with control transfer and latency problems
- Now: thread level parallelism





Economics

Commodity microprocessors not only fast but CHEAP

- Development cost is tens of millions of dollars (5-100 typical)
- BUT, many more are sold compared to supercomputers
- Crucial to take advantage of the investment, and use the commodity building block
- Exotic parallel architectures no more than special-purpose

Multiprocessors being pushed by software vendors (e.g. database) as well as hardware vendors

Standardization by processor vendors makes small, bus-based SMPs commodity

Desktop: few smaller processors versus one larger one?

- Multiprocessor on a chip