Design of Parallel and High-Performance Computing

Fall 2019

Lecture: Introduction

Instructor: Tal Ben-Nun & Markus Püschel

TA: Timo Schneider

ETH

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

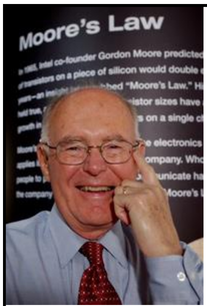
Goals of this lecture

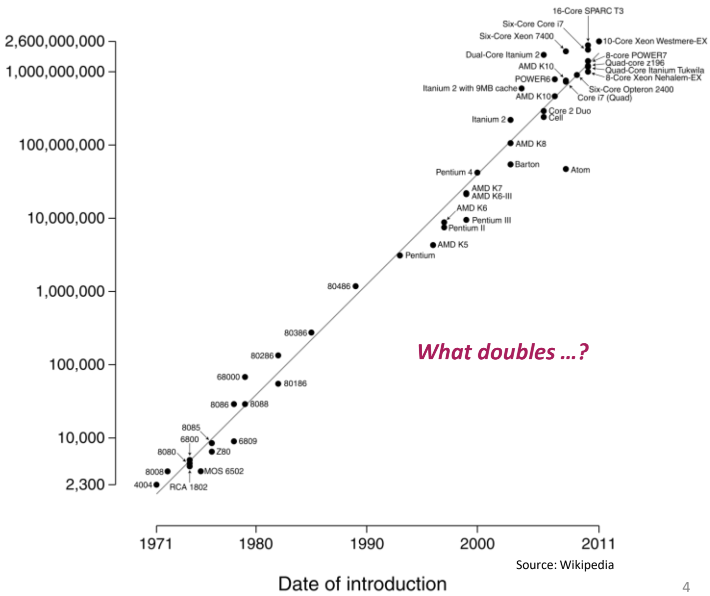
- **Motivate you!**
- **Trends**
- **High performance computing**
- **Programming models**
- **Course overview**

2

Trends

3

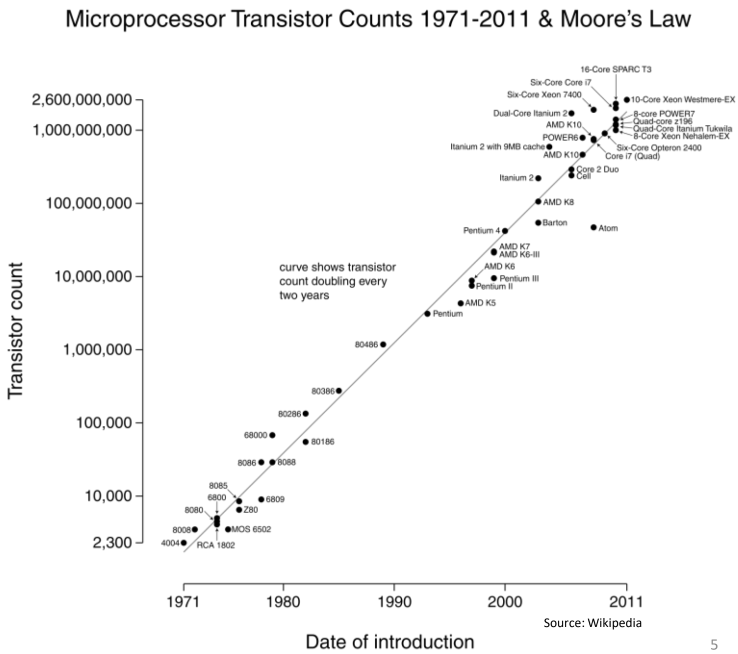




What doubles …?

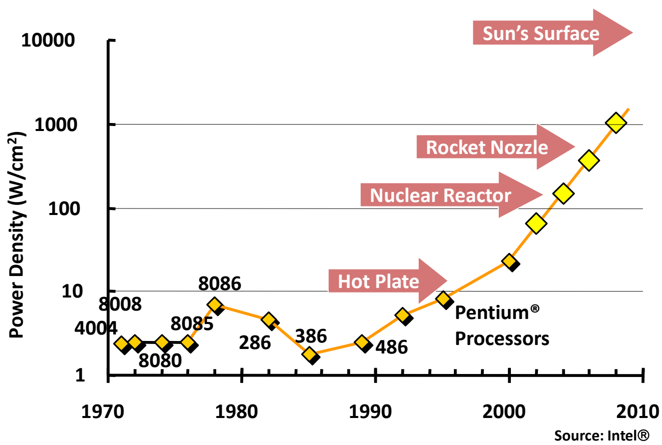
Source: Wikipedia

4



How to increase the compute power?

Clock Speed!

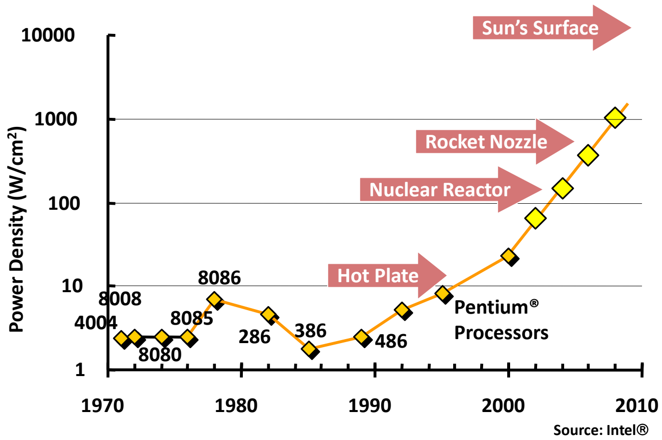


© Torsten Hoefler and Markus Püschel
Computer Science

How to increase the compute power?

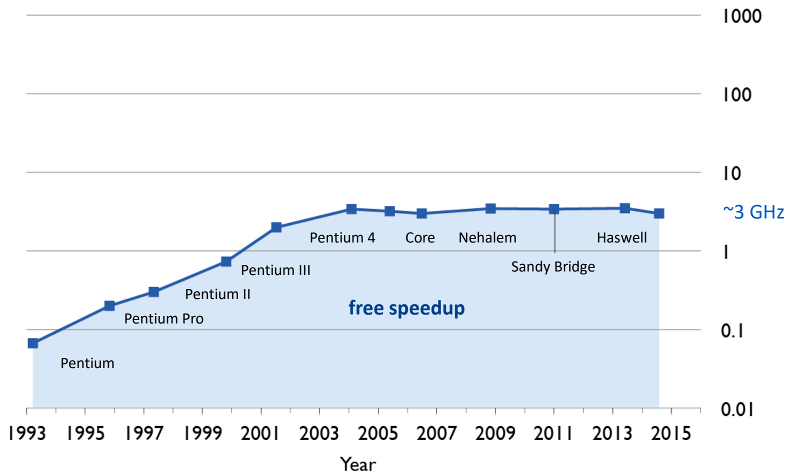
Not an option anymore!

~~Clock Speed!~~



Evolutions of Processors (Intel)

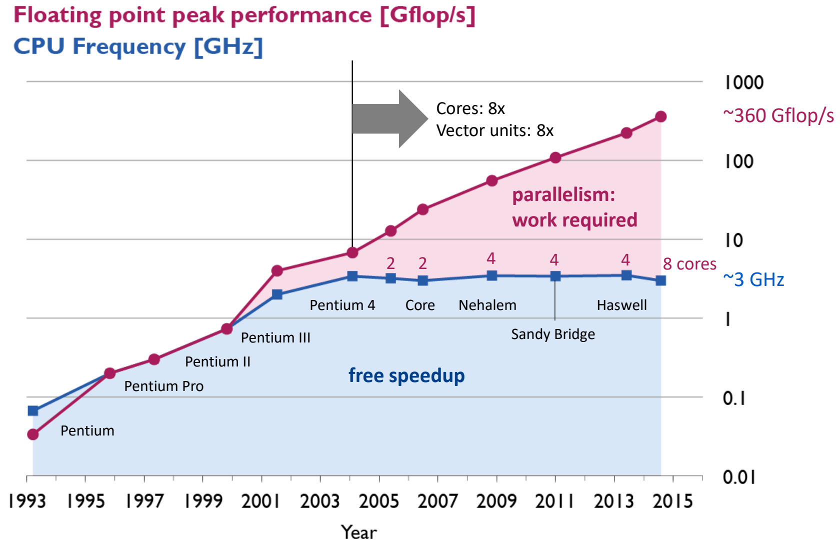
CPU Frequency [GHz]



Source: Wikipedia/Intel/PCGuide

© Torsten Hoefler and Markus Püschel
Computer Science
ETH Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

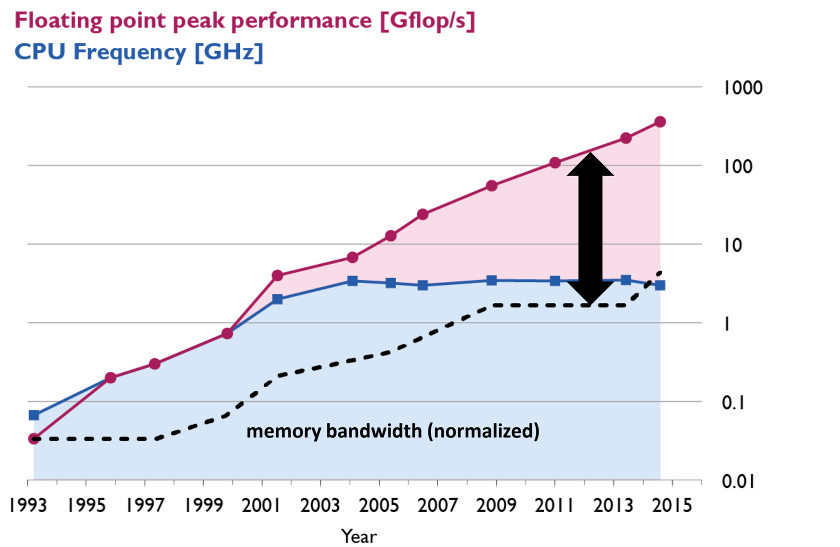
Evolutions of Processors (Intel)



Source: Wikipedia/Intel/PCGuide

9

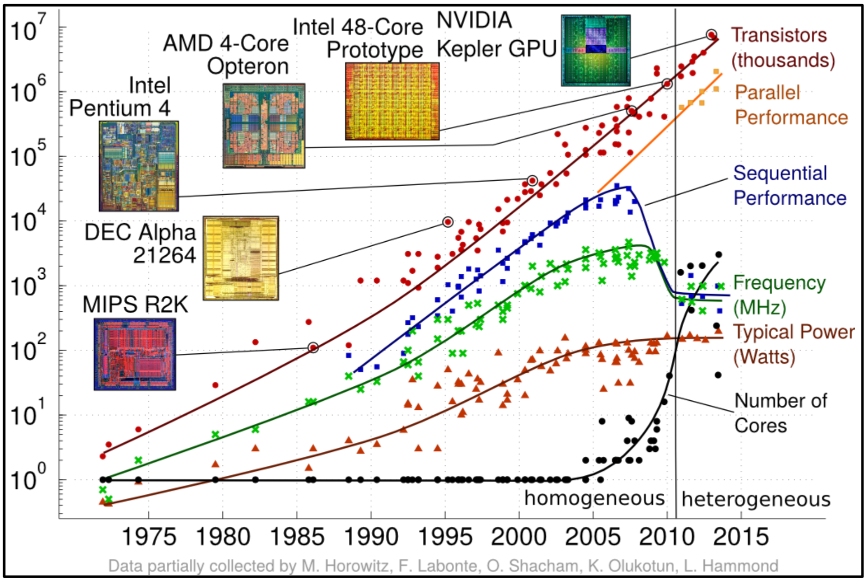
Evolutions of Processors (Intel)



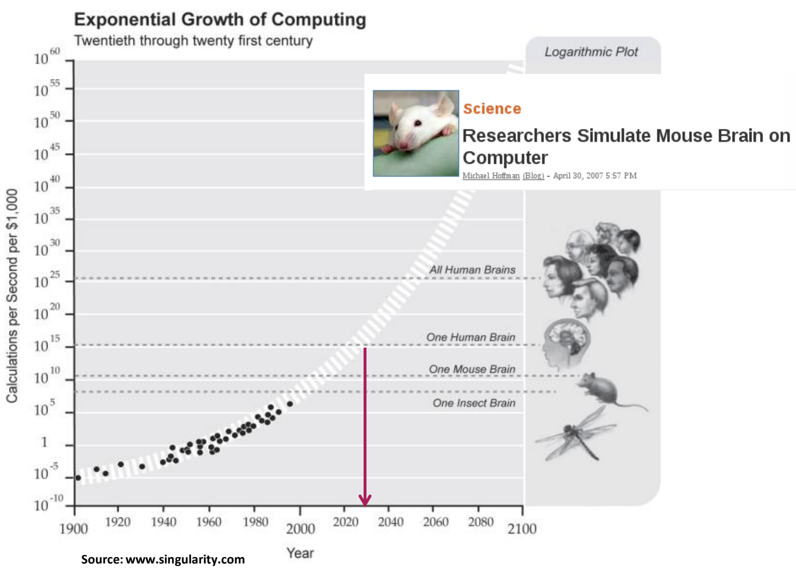
Source: Wikipedia/Intel/PCGuide

10

A more complete view



11



Source: www.singularity.com

12

High-Performance Computing

High-Performance Computing (HPC)

- a.k.a. "Supercomputing"
- Question: define "Supercomputer"!

© Torsten Hoefler and Markus Püschel
Computer Science
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

High-Performance Computing (HPC)

- **a.k.a. "Supercomputing"**
- **Question: define "Supercomputer"!**
  - "A supercomputer is a computer at the frontline of contemporary processing capacity—particularly speed of calculation." (Wikipedia)
  - Usually quite expensive ($s and kWh) and big (space)
- **HPC is a quickly growing niche market**
  - Not all "supercomputers", wide base
  - Important enough for vendors to specialize
  - Very important in research settings (up to 40% of university spending)

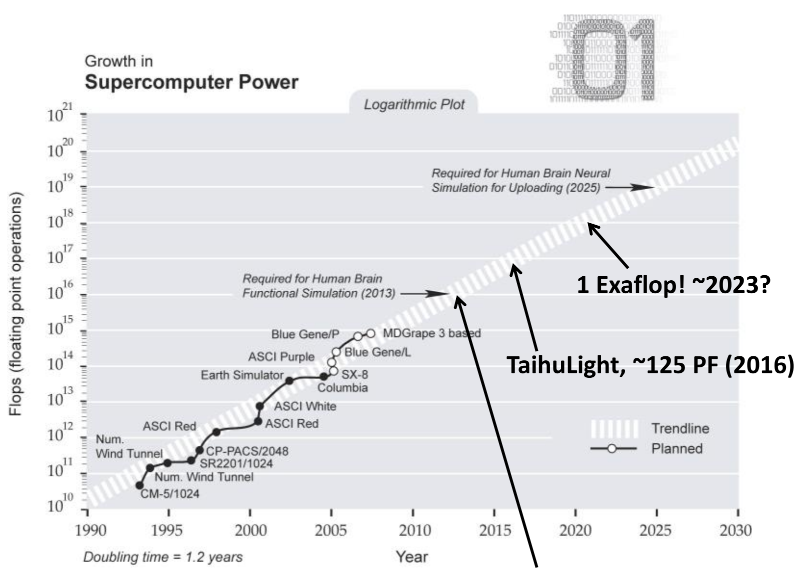
    "Goodyear Puts the Rubber to the Road with High Performance Computing"

    "High Performance Computing Helps Create New Treatment For Stroke Victims"

    "Procter & Gamble: Supercomputers and the Secret Life of Coffee"

    "Motorola: Driving the Cellular Revolution With the Help of High Performance Computing"

    "Microsoft: Delivering High Performance Computing to the Masses"



Source: www.singularity.com

Blue Waters in 2012

Simulating 1 second of human brain activity takes 82,944 processors

By Ryan Whitwam on August 5, 2013 at 1:34 pm | 21 Comments

Share This Article

436 Like | 123 Tweet | 228 | 108 +1 | 24 Share

The brain is a deviously complex biological computing device that even the fastest supercomputers in the world fail to emulate. Well, that's not entirely true anymore. Researchers at the Okinawa Institute of Technology Graduate University in Japan and Forschungszentrum Jülich in Germany have managed to simulate a single second of human brain activity in a very, very powerful computer.

Source: extremetech.com

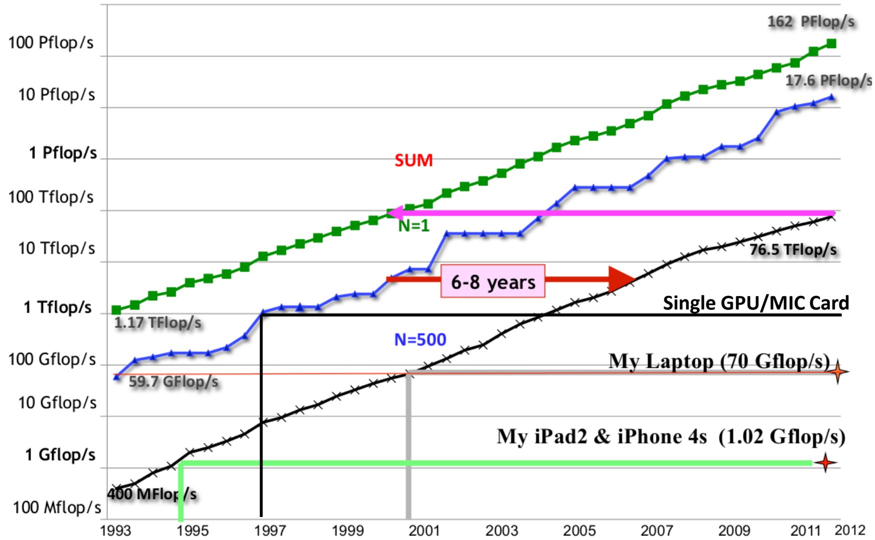
The Top500 List

- **A benchmark, solve Ax=b**
  - As fast as possible! → as big as possible ☺
  - Reflects **some** applications, not all, not even many
  - Very good historic data!
- **Speed comparison for computing centers, states, countries, nations, continents ☹**
  - Politicized (sometimes good, sometimes bad)
  - Yet, fun to watch

The Top500 List (June 2019)

Rank	Site	System	Cores	Rmax (TFlop/s)	Rpeak (TFlop/s)	Power (kW)
1	DOE/SC/Oak Ridge National Laboratory United States	**Summit** - IBM Power System AC922, IBM POWER9 22C 3.07GHz, NVIDIA Volta GV100, Dual-rail Mellanox EDR Infiniband IBM	2,414,592	148,600.0	200,794.9	10,096
2	DOE/NNSA/LLNL United States	**Sierra** - IBM Power System S922LC, IBM POWER9 22C 3.1GHz, NVIDIA Volta GV100, Dual-rail Mellanox EDR Infiniband IBM / NVIDIA / Mellanox	1,572,480	94,640.0	125,712.0	7,438
3	National Supercomputing Center in Wuxi China	**Sunway TaihuLight** - Sunway MPP, Sunway SW26010 260C 1.45GHz, Sunway NRCPC	10,649,600	93,014.6	125,435.9	15,371
4	National Super Computer Center in Guangzhou China	**Tianhe-2A** - TH-IVB-FEP Cluster, Intel Xeon E5-2692v2 12C 2.2GHz, TH Express-2, Matrix-2000 NUDT	4,981,760	61,444.5	100,678.7	18,482
5	Texas Advanced Computing Center/Univ. of Texas United States	**Frontera** - Dell C6420, Xeon Platinum 8280 28C 2.7GHz, Mellanox InfiniBand HDR Dell EMC	448,448	23,516.4	38,745.9	
6	Swiss National Supercomputing Centre (CSCS) Switzerland	**Piz Daint** - Cray XC50, Xeon E5-2690v3 12C 2.6GHz, Aries interconnect , NVIDIA Tesla P100 Cray Inc.	387,872	21,230.0	27,154.3	2,384
7	DOE/NNSA/LANL/SNL United States	**Trinity** - Cray XC40, Xeon E5-2698v3 16C 2.3GHz, Intel Xeon Phi 7250 68C 1.4GHz, Aries interconnect Cray Inc.	979,072	20,158.7	41,461.2	7,578

Top500: Trends



Source: Jack Dongarra

21

Piz Daint @ CSCS

More pictures at: http://spcl.inf.ethz.ch/Teaching/2015-dphpc/

HPC Applications: Scientific Computing

- **Most natural sciences are simulation driven or are moving towards simulation**
  - Theoretical physics (solving the Schrödinger equation, QCD)
  - Biology (Gene sequencing)
  - Chemistry (Material science)
  - Astronomy (Colliding black holes)
  - Medicine (Protein folding for drug discovery)
  - Meteorology (Storm/Tornado prediction)
  - Geology (Oil reservoir management, oil exploration)
  - and many more … (even Pringles uses HPC)

HPC Applications: Commercial Computing

- **Databases, data mining, search**
  - Amazon, Facebook, Google
- **Transaction processing**
  - Visa, Mastercard
- **Decision support**
  - Stock markets, Wall Street, Military applications
- **Parallelism in high-end systems and back-ends**
  - Often throughput-oriented
  - Used equipment varies from COTS (Google) to high-end redundant mainframes (banks)

© Torsten Hoefler and Markus Püschel
Computer Science ETH Eidgenössische Technische Hochschule Zürich Swiss Federal Institute of Technology Zurich

HPC Applications: Industrial Computing

- **Aeronautics (airflow, engine, structural mechanics, electromagnetism)**
- **Automotive (crash, combustion, airflow)**
- **Computer-aided design (CAD)**
- **Pharmaceuticals (molecular modeling, protein folding, drug design)**
- **Petroleum (Reservoir analysis)**
- **Visualization (all of the above, movies, 3d)**

What can faster computers do for us?

- **Solving bigger problems than we could solve before!**
  - E.g., Gene sequencing and search, simulation of whole cells, mathematics of the brain, …
  - The size of the problem grows with the machine power
    → *Weak Scaling*
- **Solve today's problems faster!**
  - E.g., large (combinatorial) searches, mechanical simulations (aircrafts, cars, weapons, …)
  - The machine power grows with constant problem size
    → *Strong Scaling*

Towards the age of massive parallelism

- **Everything goes parallel**
  - Desktop computers get more cores
    - *2,4,8, soon dozens, hundreds?*
  - Supercomputers get more PEs (cores, nodes)
    - *> 10 million today*
    - *> 50 million on the horizon*
    - *➢1 billion in a couple of years*
- **Parallel Computing is inevitable!**

> ***Parallel vs. Concurrent computing***
> Concurrent activities ***may*** be executed in parallel
> Example:
> A1 starts at T1, ends at T2; A2 starts at T3, ends at T4
> Intervals (T1,T2) and (T3,T4) may overlap!
> Parallel activities:
> A1 is executed ***while*** A2 is running
> Usually requires separate resources!

Programming Models

© Torsten Hoefler and Markus Püschel
Computer Science
ETH Eidgenössische Technische Hochschule Zürich Swiss Federal Institute of Technology Zurich

Flynn's Taxonomy

Single instruction, multiple data (→ SIMD)

SISD Standard Serial Computer (nearly extinct)	**SIMD** Vector Machines or Extensions (very common)
MISD Redundant Execution (fault tolerance)	**MIMD** Multicore (ubiquituous)

Parallel Resources and Programming

Parallel Resource

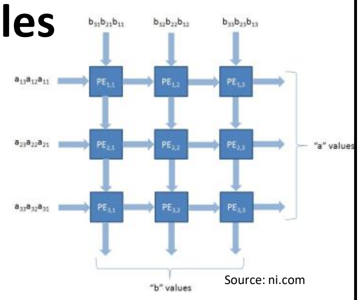
- Instruction-level parallelism
  - Pipelining
  - VLIW
  - Superscalar
- SIMD operations
  - Vector operations
- Instruction sequences
  - Multiprocessors
  - Multicores
  - Multithreading

Programming

- Compiler
  - (inline assembly)
  - Hardware scheduling
- Compiler (inline assembly)
- Intrinsics
- Libraries
- Compilers (very limited)
- Expert programmers
  - Parallel languages
  - Parallel libraries
  - Hints

© Torsten Hoefler and Markus Püschel
Computer Science

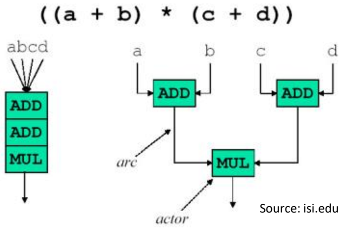
Historic Architecture Examples



Source: ni.com

- **Systolic Array**
  - Data-stream driven (data counters)
  - Multiple streams for parallelism
  - Specialized for applications (reconfigurable)
- **Dataflow Architectures**
  - No program counter, execute instructions when all input arguments are available
  - Fine-grained, high overheads

    *Example: compute f = (a+b) * (c+d)*

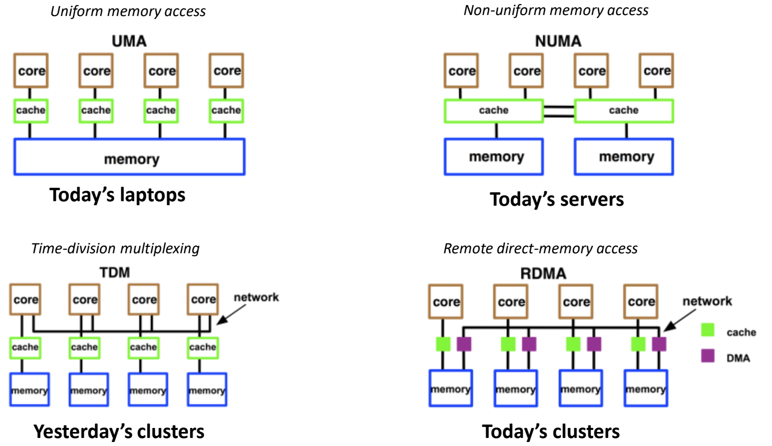


Source: isi.edu

- **Both come-back in FPGA computing**
  - Interesting research opportunities!

31

Parallel Architectures 101

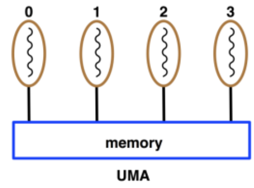


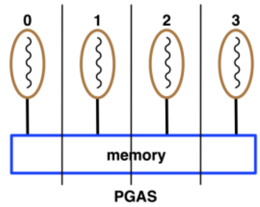
- **… and mixtures of those**

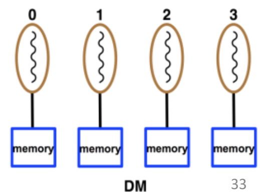
32

Programming Models

- **Shared Memory Programming (SM)**
  - Shared address space
  - Implicit communication
  - Hardware for cache-coherent remote memory access
  - Cache-coherent Non Uniform Memory Access (cc NUMA)
  - Pthreads, OpenMP
- **(Partitioned) Global Address Space (PGAS)**
  - Remote Memory Access
  - Remote vs. local memory (cf. ncc-NUMA)
- **Distributed Memory Programming (DM)**
  - Explicit communication (typically messages)
  - Message Passing

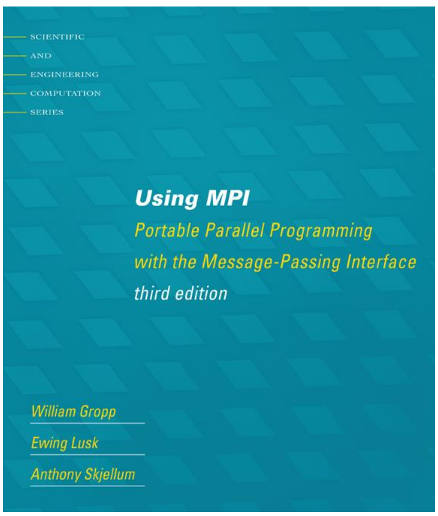




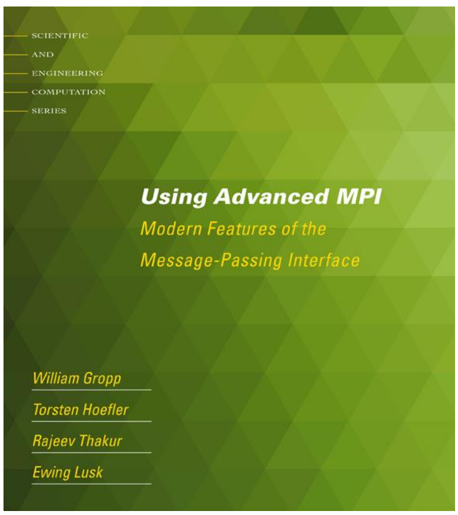


33

MPI: de-facto large-scale prog. standard



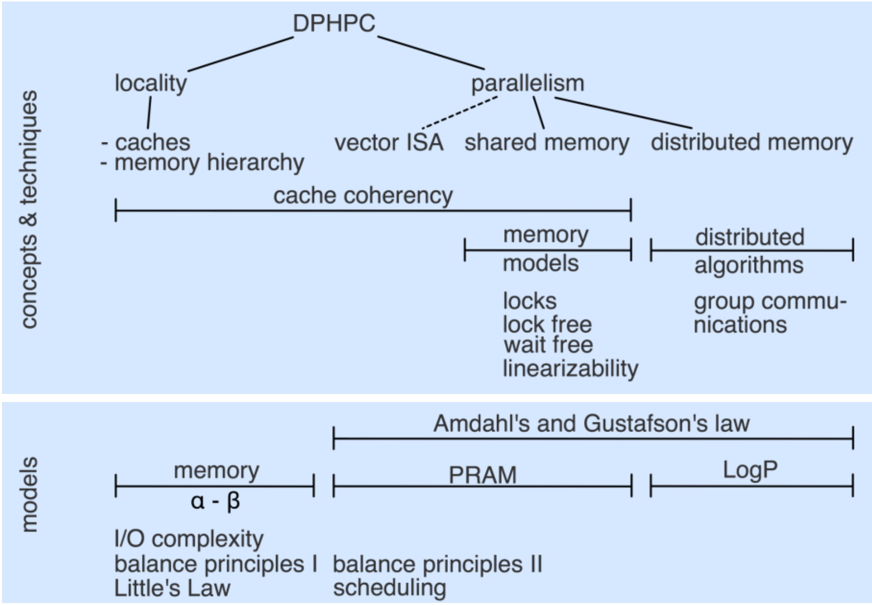
Basic MPI



Advanced MPI, including MPI-3

34

DPHPC Overview



35

Schedule of Last Year

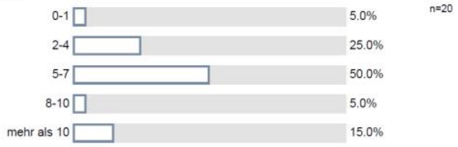
Tentative schedule of lectures

Week	Monday	Thursday
1	09/17: no lecture	09/20: no recitation
2	09/24: Organization - Introduction	09/27: MPI Tutorial - Code
3	10/01: Caches and Cache Coherence (1pp) (6pp) (unrolled animations)	10/04: Caches recap - Euler cluster - Euler sample job script
4	10/08: Memory models (1pp) (6pp) (unrolled animations)	10/11: Teams and initial projects presentations - OpenMP intro
5	10/15: Languages, Fast Locks, and Lock-Free (1pp) (6pp) (unrolled animations)	10/18: Memory Models - MPI Part 2
6	10/22: Fast Locks and Lock-Free (1pp) (6pp) (unrolled animations)	10/25: SPIN Tutorial - Continuing MPI Part 2
7	10/29: Intermediate Project Presentations	11/01: No recitation session
8	11/05: Amdahl's Law, PRAM, Alpha-Beta Model, Little's Law, Operational Intensity, Roofline Model I	11/08: Amdahl's Law, PRAM
9	11/12: Notes - Roofline Model II - Balance Principles - Scheduling	11/15: Roofline Model - Balance Principles
10	11/19: SIMD Vector Extensions	11/22: Balance Principles, SIMD - Vandermonde matrix determinant vectorized
11	11/26: Finishing consensus, scalable lock study, and oblivious algorithms (1pp) (6pp) (unrolled animations)	11/29: Work-Depth Model - MS thesis proposal
12	12/03: I/O complexity, Red-Blue Pebble Game, and Recomputation in Neural Networks	12/06: Red-Blue Pebble Game
13	12/10: Oblivious and Non-Oblivious Algorithms (1pp) (6pp)	12/13: Prefix-Sum, Network Models
14	12/17: Final Presentations	

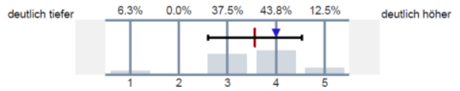
36

From Evaluation 2018

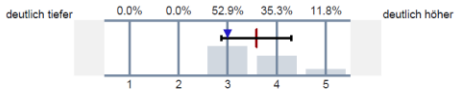
Wie viele Stunden haben Sie während des Semesters durchschnittlich pro Woche für Arbeiten ausserhalb der Kontaktstunden (Lösen der Übungen, Vor- und Nachbereitung der Lerneinheit) aufgewendet?



Im Vergleich zu anderen Lehrveranstaltungen derselben Kurskategorie war der fachliche Anspruch...



Im Vergleich zu anderen Lehrveranstaltungen derselben Kurskategorie war der Aufwand pro Kreditpunkt...



A comment:
The course description should mention that a strong parallel computing background would be appreciated to fully enjoy this course