Erlangen Regional Computing Center

UNIVERSITÄT GREIFSWALD
Wissen lockt. Seit 1456

FRIEDRICH-ALEXANDER UNIVERSITÄT ERLANGEN-NÜRNBERG

Winter term 2020/2021

Parallel Programming with OpenMP and MPI

Dr. Georg Hager
Erlangen Regional Computing Center (RRZE) at Friedrich-Alexander-Universität Erlangen-Nürnberg
Institute of Physics, Universität Greifswald

Lecture 1: Preliminaries (kick-off meeting)

High Performance Computing

Audience and contact

- Audience
  - Physics, theoretical chemistry, computer science, applied math, materials science, "computational XYZ"
  - Everyone who
    - Needs more computing power than what a laptop/PC can provide
    - Wants to learn about parallel programming from desktop to supercomputers
- Lecturer
  - Georg Hager georg.hager@uni-greifswald.de
  - Associate lecturer at University of Greifswald, Institute of Physics
  - PhD 2005, Habilitation 2014 (both in Greifswald)

- Contact: Preferably use the Moodle forum
  - Moodle course: http://tiny.cc/ParProg20

Course format

- Online lecture
  - 2 hours (90 minutes) per week
  - Lecture video published every Monday in moodle

- Exercises
  - One exercise sheet every week
  - Solutions will be discussed in Q&A (no submits necessary)

- Online Q&A session (via BBB) with discussion of exercises
  - Tuesday 3 p.m.

- All material (slides, videos, exercises) available at http://tiny.cc/ParProg20

Course prerequisites

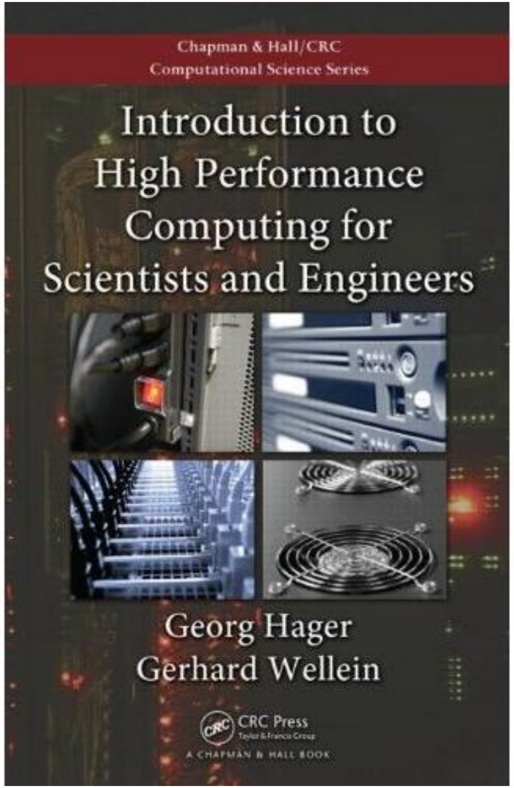
- Lecture:
  - Some C, C++, or Fortran programming
  - Examples are in (simple) C or Fortran

- Exercises:
  - Linux command line (including remote access via SSH)
    - Recommended Windows tool: MobaXTerm (https://mobaxterm.mobatek.net/)
  - Handling a compiler on the command line
  - You will get accounts for accessing the HPC clusters at RRZE (FAU Erlangen-Nürnberg)

- Linux tutorial for n00bs: https://ryanstutorials.net/linuxtutorial/

Supporting material

- G. Hager and G. Wellein:
  Introduction to High Performance Computing for Scientists and Engineers.
  CRC Computational Science Series, 2010.
  ISBN 978-1439811924
- Documentation:
  - https://www.openmp.org
  - https://www.mpi-forum.org
- The big ones and more useful HPC-related information:
  - https://www.top500.org/



Outline of lecture

- Basics of parallel computer architecture
- Basics of parallel computing
- Introduction to shared-memory programming with OpenMP
- OpenMP performance issues
- Introduction to the Message Passing Interface (MPI)
- Advanced MPI
- MPI performance issues
- Hybrid MPI+OpenMP programming

- Goal: A good grasp of the potentials and performance issues of parallel computing in computational science

Erlangen Regional Computing Center

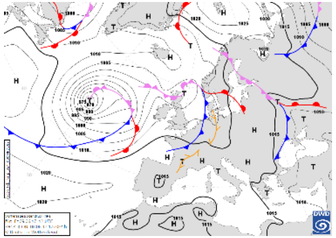
FRIEDRICH-ALEXANDER UNIVERSITÄT ERLANGEN-NÜRNBERG

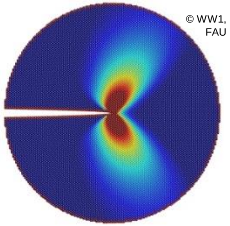
Supercomputing

High Performance Computing

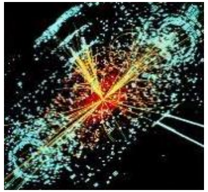
HPC applications

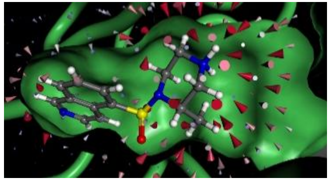
- What are supercomputers good for?
  - Weather and climate prediction
  - Drug design
  - Simulation of biochemical reactions
  - Processing and analysis of measurement data
  - Properties of condensed matter
  - Fundamental interactions and structure of matter
  - Fluid simulations, structural analysis, fluid-structure interaction
  - Mechanical properties of materials
  - Rendering of 3D images and movies
  - Simulation of nuclear explosions
  - Medical image reconstruction
  - …



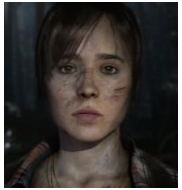


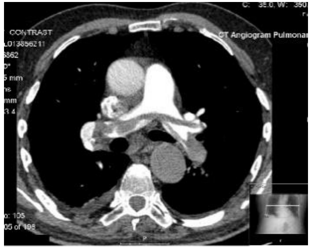
© WW1, FAU





© T. Exner, Molcad GmbH





HPC algorithms

- Whatever the application, there's usually a numerical algorithm behind it
- Computational science → many standard algorithms
- "Seven dwarfs"
  1. Dense linear algebra
  2. Sparse linear algebra
  3. Spectral methods
  4. N-body methods
  5. Structured grids
  6. Unstructured grids
  7. Monte Carlo methods

See also:
[The Landscape of Parallel Computing Research: A View from Berkeley, Chapter 3]

Parallel computing

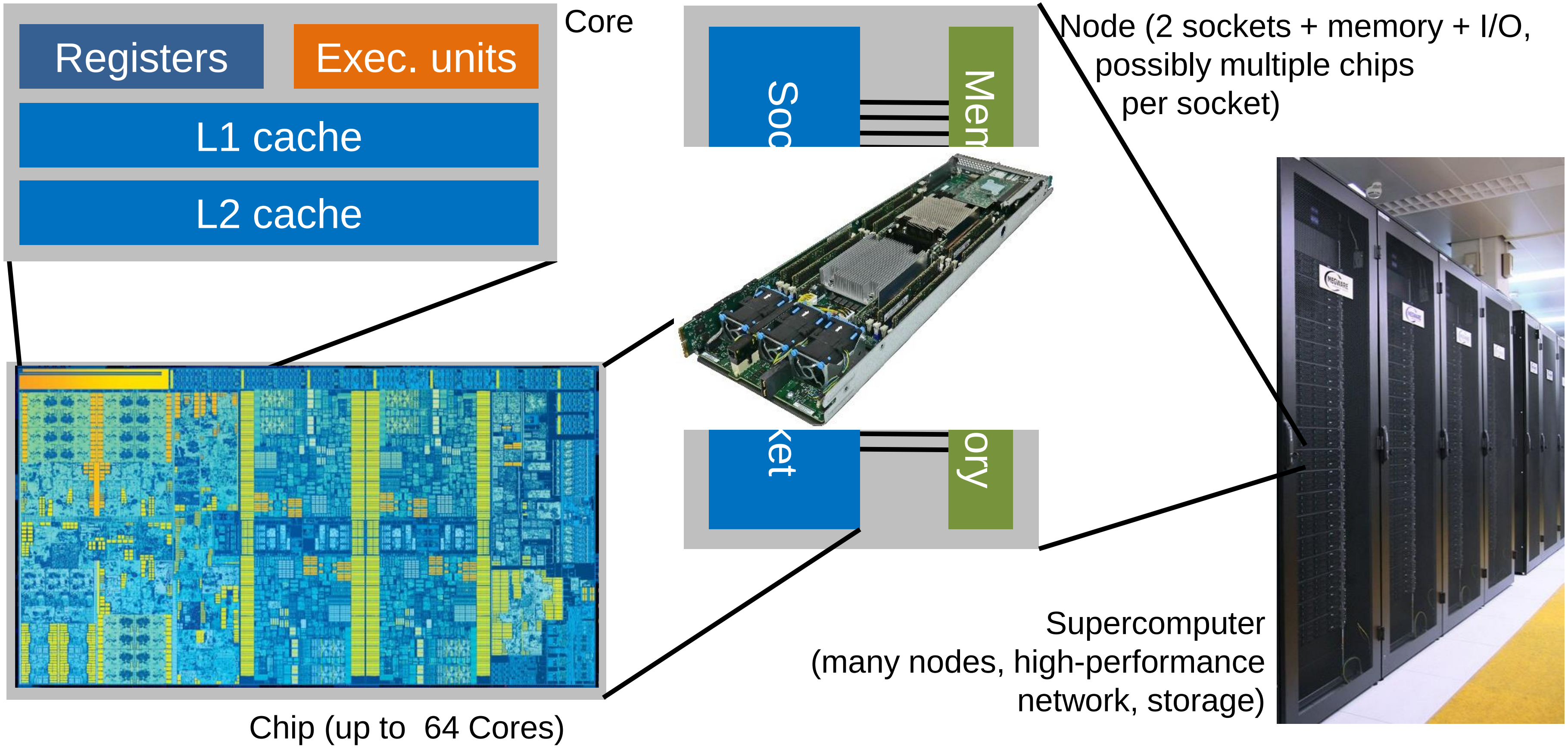
Task: Map a numerical algorithm to the hardware of a parallel computer

$$v_i = \sum_{j=1}^{n} A_{ij} b_j$$

???

Goal: Execute the task as fast and effective as possible

Parallelism in modern computers



The Top500 list

- Survey of the 500 most powerful supercomputers
  - http://www.top500.org
- Performance ranking?
  - Solve large dense system of equations: $Ax = b$ (“LINPACK”)

- Max. performance achieved with 64-Bit floating-point numbers: R_{max}
- Published twice a year (ISC in Germany, SC in USA)

  - First: 1993 (#1: CM5 / 1,024 procs.): **60 Gflop/s**
  - June 2020 (#1: Fugaku / 7.3 mio procs): **415.5 Pflop/s**

- Performance increase: 79% p.a. from 1993 – 2020

What is “performance”?

Performance metric:

$$P = \frac{\text{Work}}{\text{Time}}$$

“**Flops**” (+ - * /)
Lattice site updates
Iterations
“Solving the problem”...

“Wall-clock time”

The flop is quite popular…

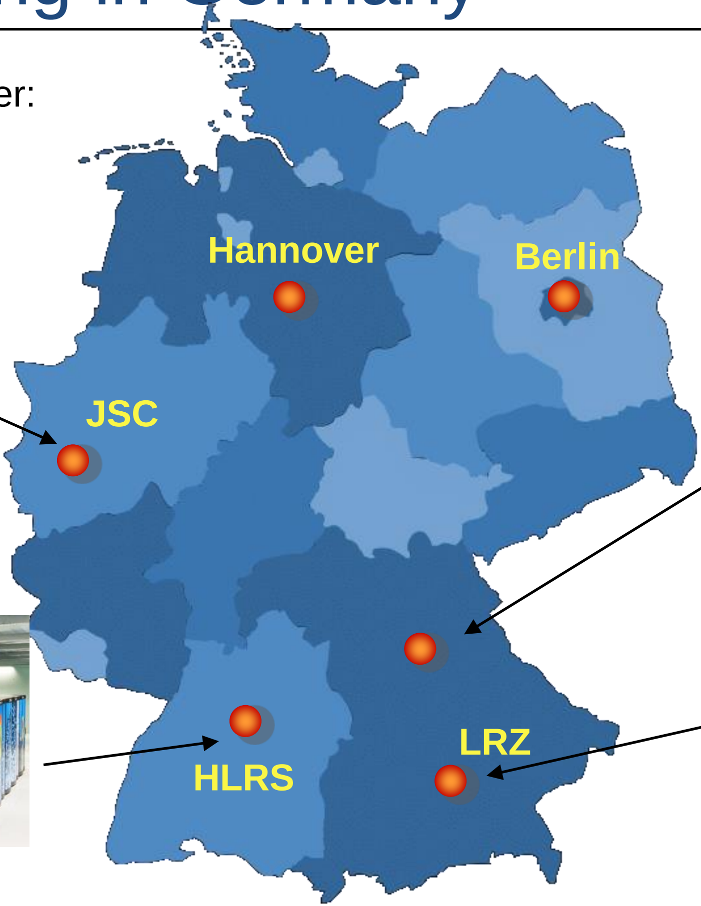
- Flop == Floating-point operation (add, subtract, multiply, divide)
- Flop/s == “how many flops can be done per second?”

- How many flops can be done by a machine at most (“peak performance”)?
  - Depends on accuracy of input operands (double, float, half-precision)
  - Divides are slow and thus usually neglected

- Some double-precision peak numbers to get you orientated…
  - Top500 range (June 2020): **2.6 Pflop/s** … **514 Pflop/s**
  - Modern multicore server CPU (AMD Rome 7742): **2.3 Tflop/s**
  - Your PC: **100 … 500 Gflop/s** (+ GPU 0.5 … 10 Tflop/s)
  - Your cellphone: **5 … 50 Gflop/s**

Supercomputing in Germany

Jülich Supercomputing Center:
JUWELS (9.9 PF/s)



RRZE (0.5 PF/s)

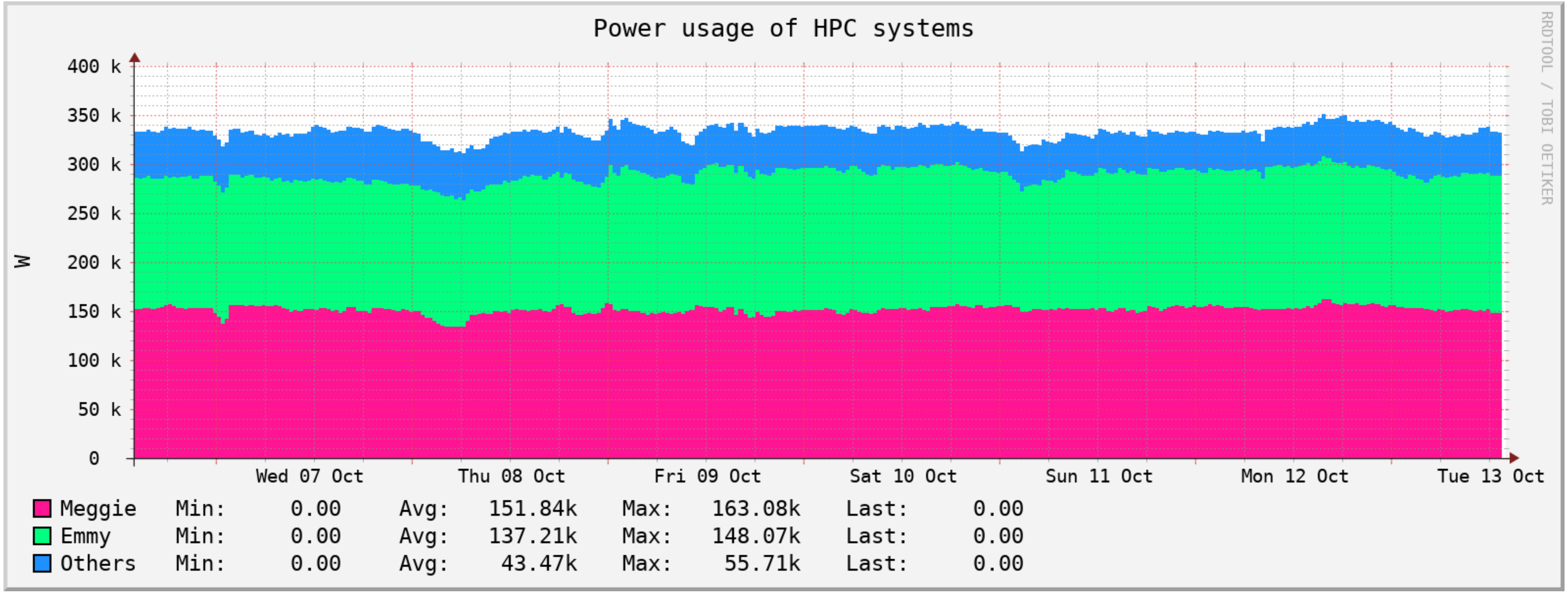
Leibniz Supercomputing Center: SuperMUC-NG (26.8 PF/s)

HLRS: Hawk (26 PF/s)

RRZE “Meggie” cluster (you will get access to this!)

- 728 Compute nodes (14.560 cores)
  - 2x Intel Xeon E5-2630 v4 (Broadwell) 2.2 GHz (10 cores)
  - 20 cores/node
  - 64 GB main memory per node
  - No local disks
- Peak Performance: $R_{peak} = 0.5\ \mathrm{Pflop/s}$
- #346 @TOP500 (Nov. 2016)
  - $R_{max} = 0.48\ \mathrm{Pflop/s}$
- Price tag: 2.5 million €
- Power consumption: 120 kW – 210 kW (depending on workload)

Power consumption of RRZE HPC systems (last 7 days)



Power consumption of supercomputers

- Cost of electrical energy (example FAU): 20 ct/kWh
  - 1 MW of power costs 1.8 million € per year
    → cost of electrical power over lifetime ≈ investment sum
  - This does not include the cost for cooling (may be 5% … 150% of electrical power)
  - ≈ 1000 €/a for a typical server

Rank	System	Cores	Rmax (TFlop/s)	Rpeak (TFlop/s)	Power (kW)
1	**Supercomputer Fugaku** - Supercomputer Fugaku, A64FX 48C 2.2GHz, Tofu interconnect D, Fujitsu RIKEN Center for Computational Science Japan	7,299,072	415,530.0	513,854.7	28,335
2	**Summit** - IBM Power System AC922, IBM POWER9 22C 3.07GHz, NVIDIA Volta GV100, Dual-rail Mellanox EDR Infiniband, IBM DOE/SC/Oak Ridge National Laboratory United States	2,414,592	148,600.0	200,794.9	10,096
3	**Sierra** - IBM Power System AC922, IBM POWER9 22C 3.1GHz, NVIDIA Volta GV100, Dual-rail Mellanox EDR Infiniband, IBM / NVIDIA / Mellanox DOE/NNSA/LLNL United States	1,572,480	94,640.0	125,712.0	7,438

- Other countries have different boundary conditions
  - US: 7ct/kWh for industrial customers (2019)

Take-home messages

- Supercomputers are parallel computers
  - No parallelism → no performance
  - It's your task to write parallel code (or use parallel programs that someone else wrote)
  - Even your desktop PC is a parallel computer nowadays

- Supercomputers are expensive
  - … to buy
  - … and to run,

    so their efficient use is paramount
  - → learn how to write efficient parallel programs