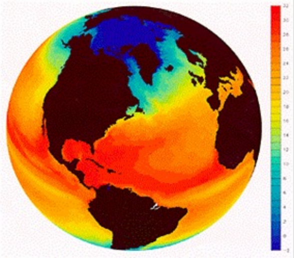
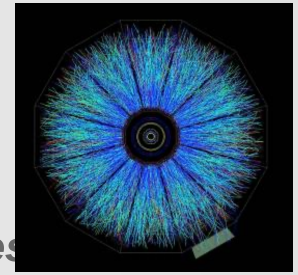
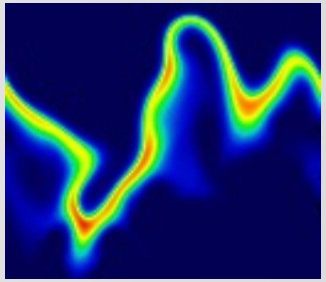
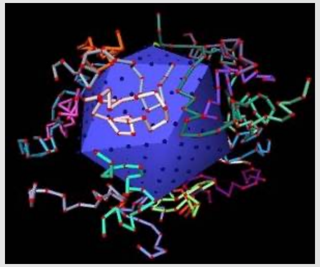
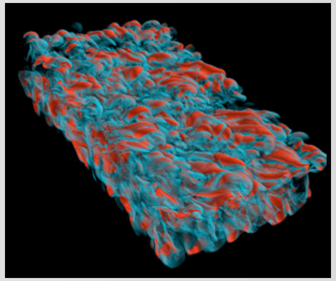
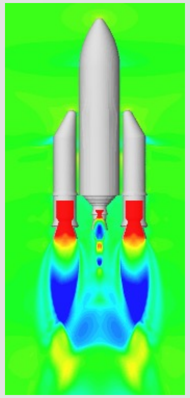
Introduction to HPC

Leon Kos

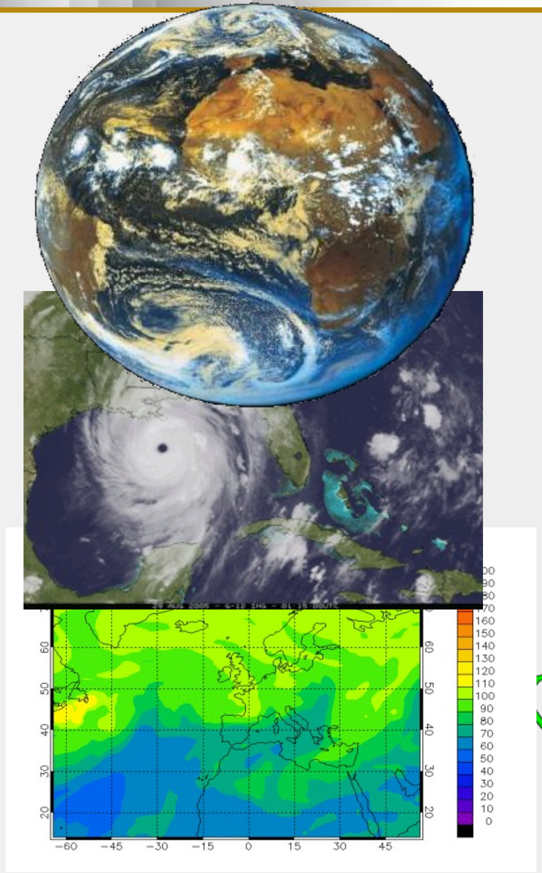
University of Ljubljana, FME, LECAD lab

Why supercomputing?

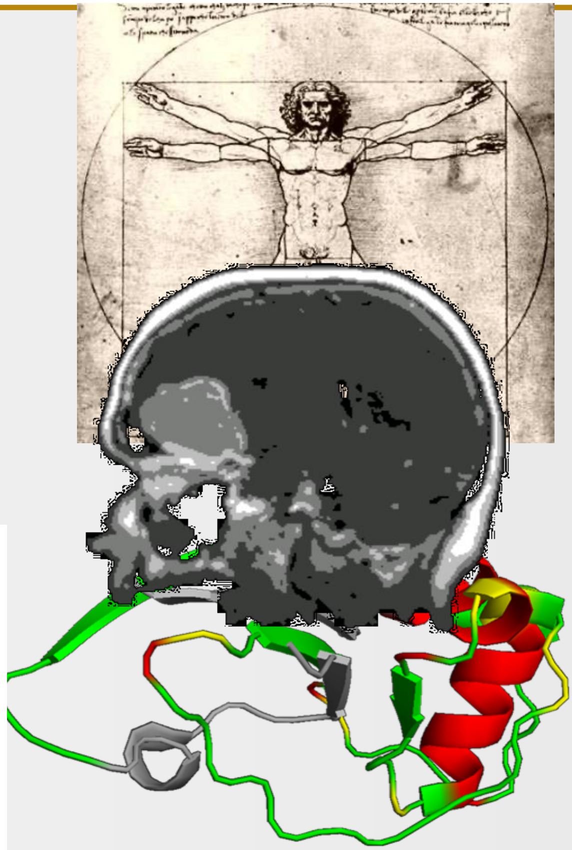
- **Weather, Climatology, Earth Science**
  - degree of warming, scenarios for our future climate.
  - understand and predict ocean properties and variations
  - weather and flood events
- ***Astrophysics, Elementary particle physics, Plasma physics***
  - ***systems, structures which span a large range of different length and time scales***
  - ***quantum field theories like QCD, ITER***
- ***Material Science, Chemistry, Nanoscience***
  - ***understanding complex materials, complex chemistry, nanoscience***
  - ***the determination of electronic and transport properties***
- ***Life Science***
  - ***system biology, chromatin dynamics, large scale protein dynamics, protein association and aggregation, supramolecular systems, medicine***
- ***Engineering***
  - ***complex helicopter simulation, biomedical flows, gas turbines and internal combustion engines, forest fires, green aircraft,***
  - ***virtual power plant***

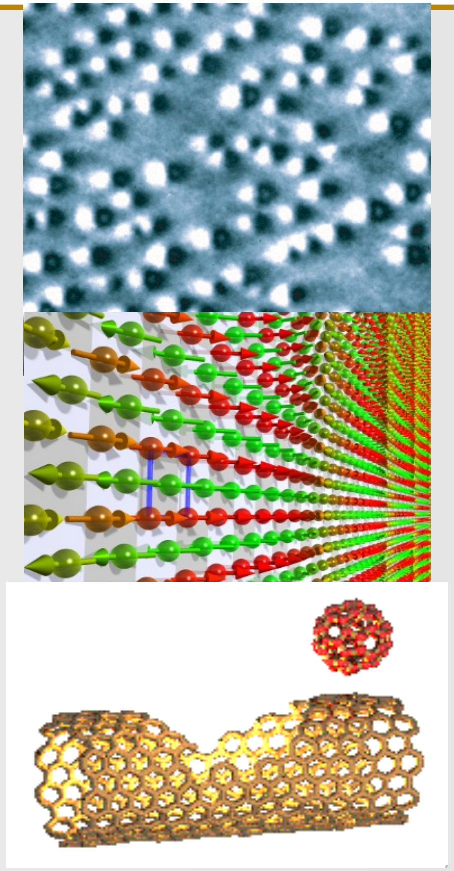
Supercomputing drives science with simulations



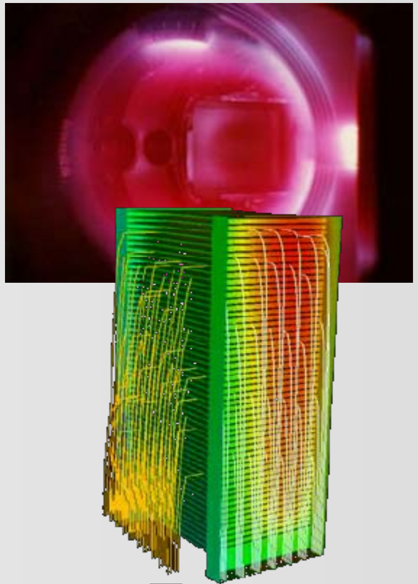
Environment
Weather/ Climatology
Pollution / Ozone Hole



Aging Society
Medicine
Biology



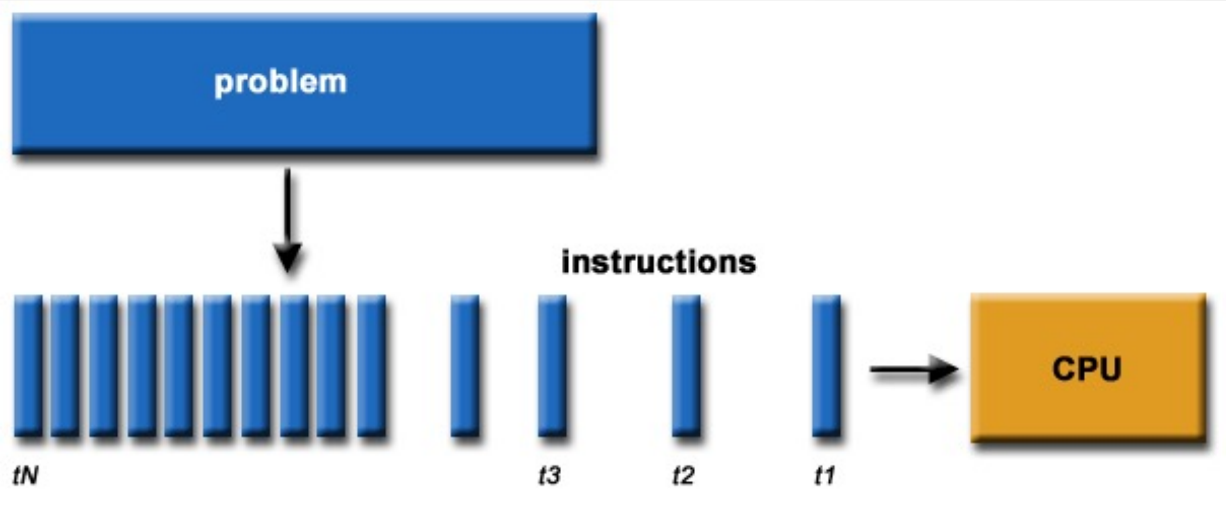
Materials/ Inf. Tech
Spintronics
Nano-science



Energy
Plasma Physics
Fuel Cells

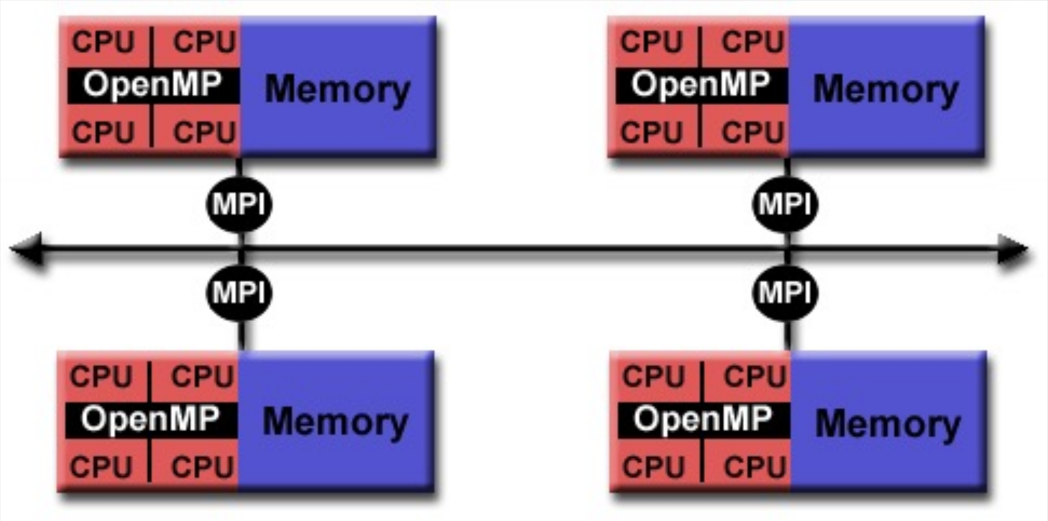
Introduction to parallel computing

- Usually is the program written for serial execution on one processor
- We divide the problem into series of commands that can be executed in parallel
- Only one command at a time can be executed on one CPU



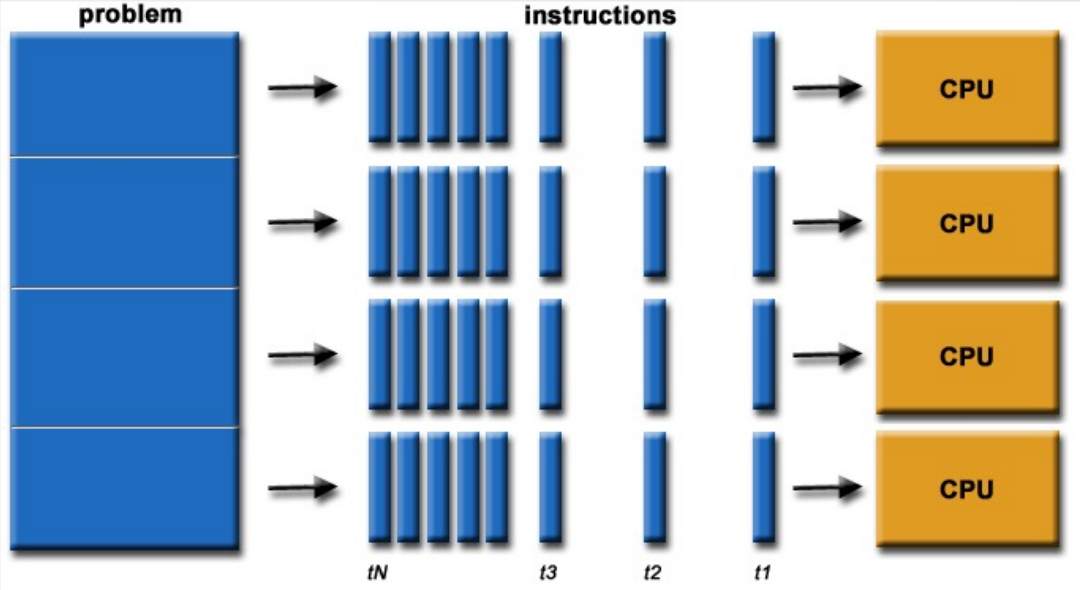
Parallel programming models

- Threading
- ***OpenMP* – *automatic parallelization***
- Distributed memory model = ***Message Passing Interface (MPI)*** – ***manual parallelization needed***
- ***Hybrid model OpenMP/MPI***



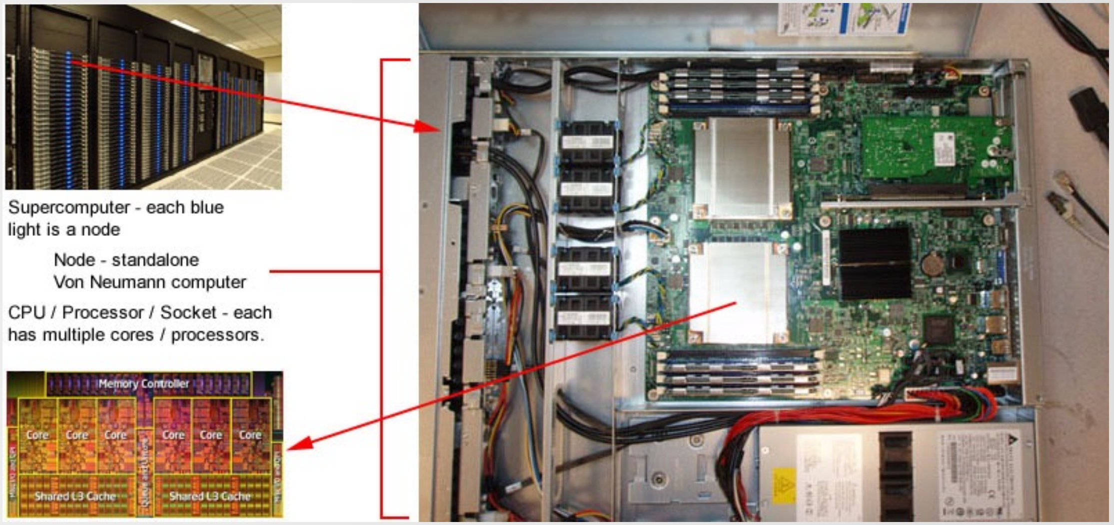
Embarrasingly simple parallel processing

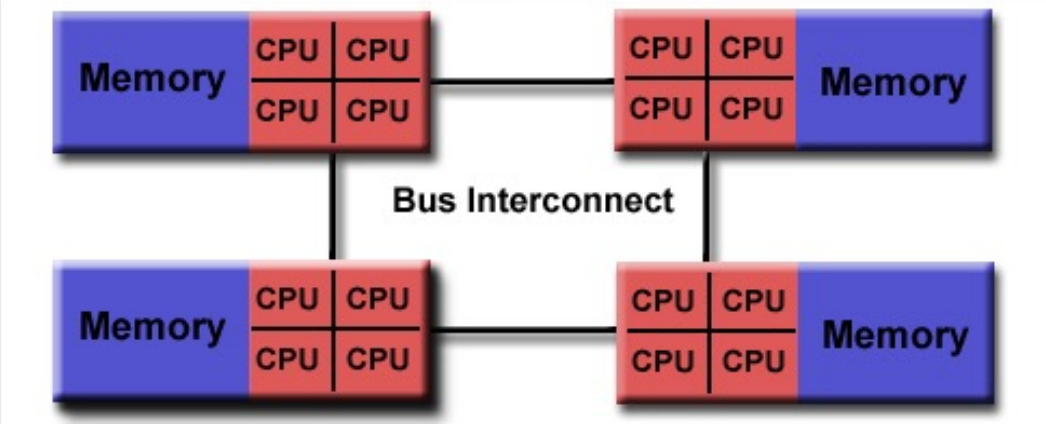
- Parallel processing of the same subproblems on multiple processors
- No communication is needed between processes



Logical view of a computing node

- Need to know computer architecture
- ***Interconnect bus for sharing memory between processors (NUMA interconnect)***

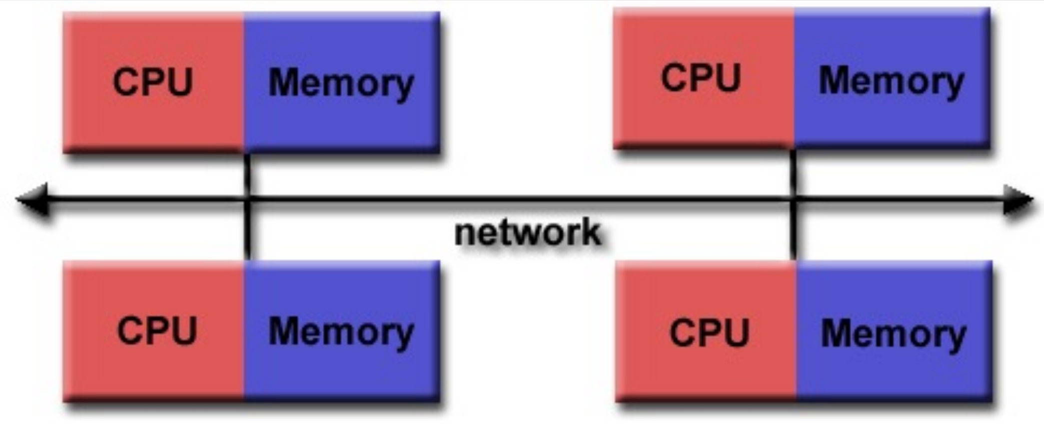


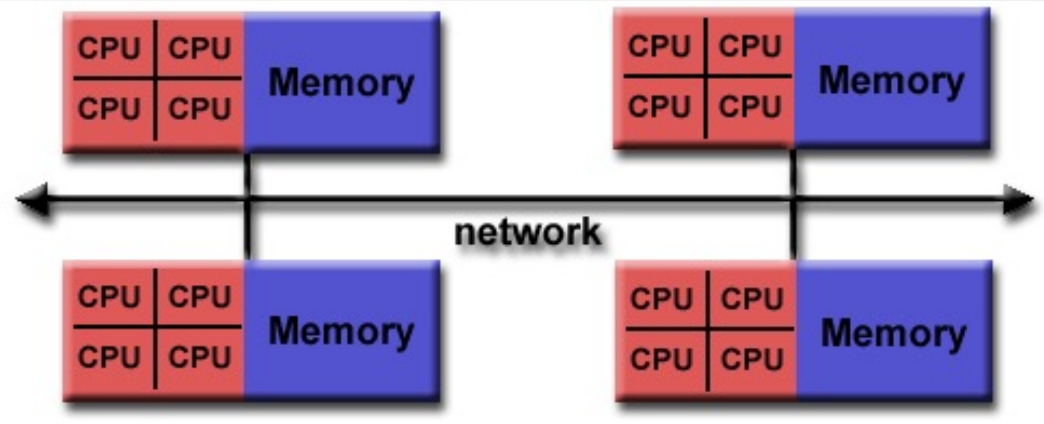


Nodes interconnect

- Distributed computing
- Many nodes exchange messages on
  - high speed,
  - low latency interconnect such as

    Infiniband





Development of parallel codes

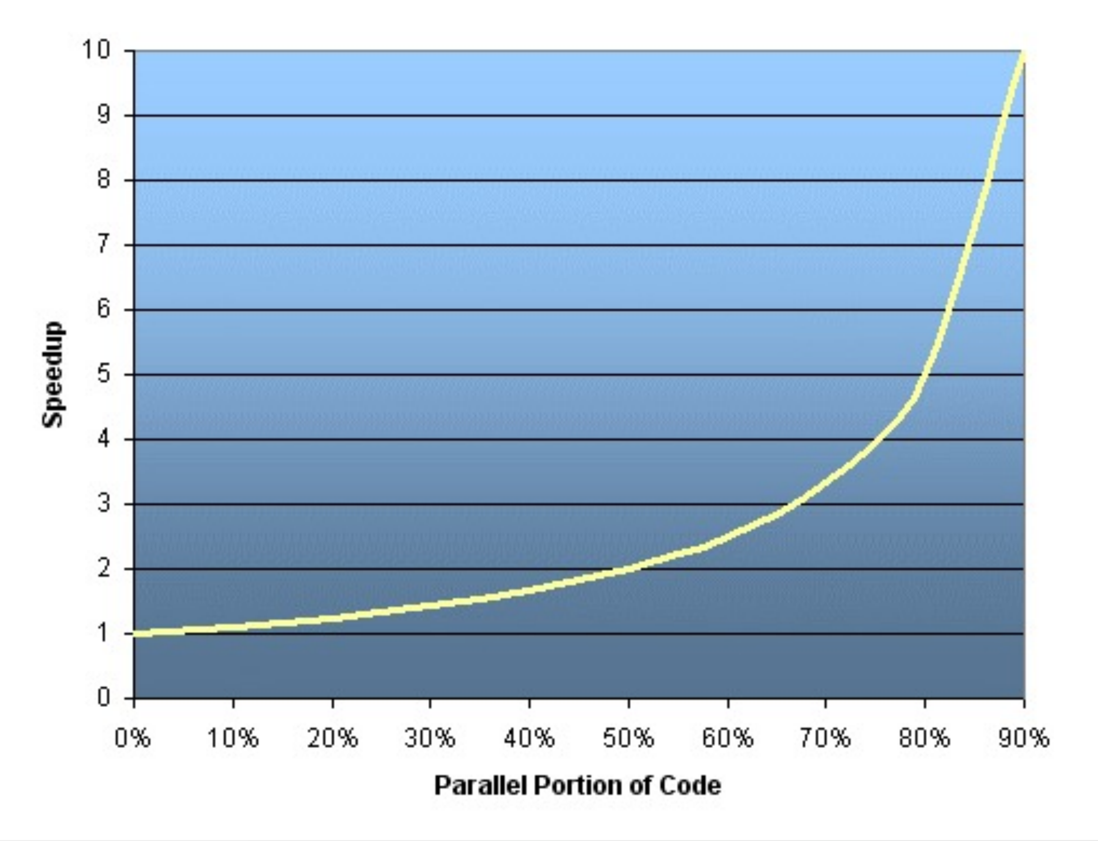
- Good understanding of the problem being solved in parallel
- How much of the problem can be run in parallel
- Bottleneck analysis and profiling gives good picture on scalability of the problem
- We optimize and parallelize parts that consume most of the computing time
- Problem needs to be dissected into parts functionally and logically

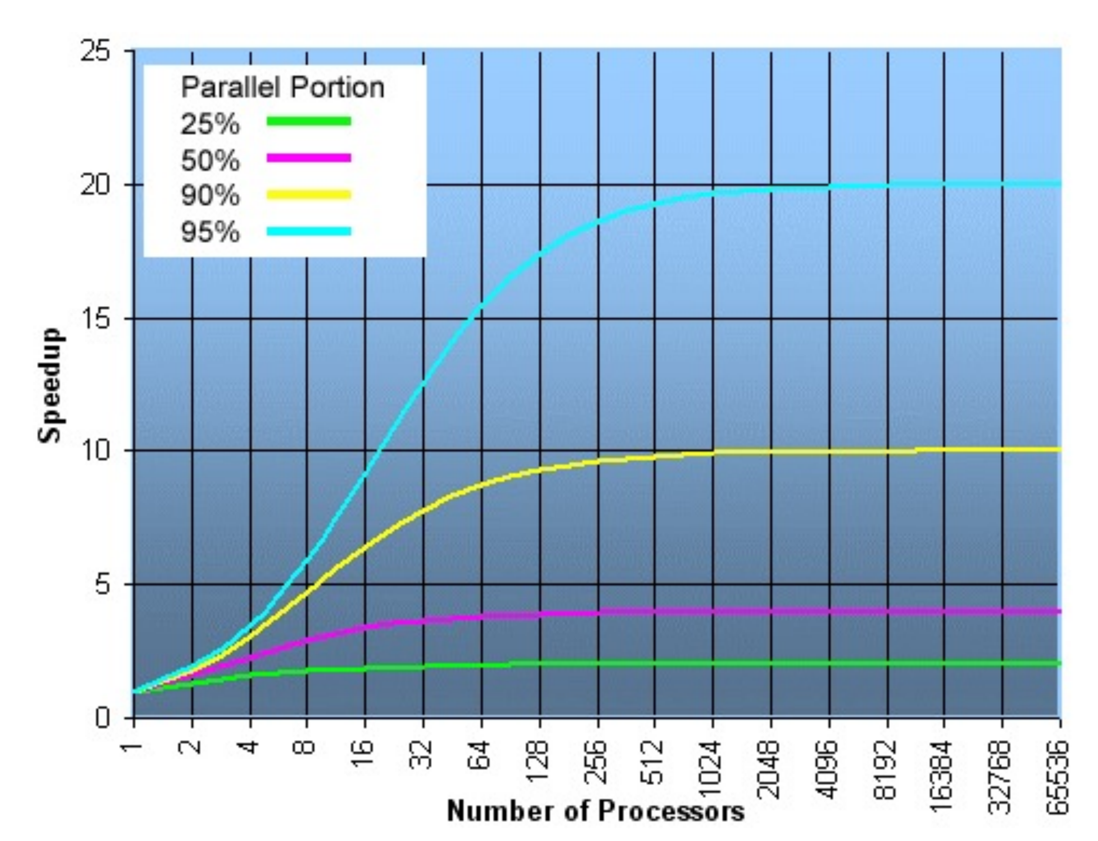
Interprocess communications

- Having little an infrequent communication between processes is the best
- Determining the largest block of code that can run in parallel and still provides scalability
- Basic properties
    - ***response time***
    - ***transfer speed - bandwidth***
    - ***interconnect capabilities***

Parallel portion of the code determines code scalability

- Amdahl's law: ***Speedup = 1/(1-p)***





Direct Solver or Iterative Solver?

EURO

- We are solving a set of matrix equations of the form [K]{u} = {f}. Here [K] is referred to as the stiffness matrix; {f} as the force vector and {u} as the set of unknowns.
  - Several milions of unknowns
  - Lot of zeros in K

- Direct solvers: Multfront, MUMPS, and LDLT, Pardiso, ...
- Iterative solvers: PETSc and GCPC, ...

Computer Aided Engineering open source tools

- CAD/CAM: Salome, Freecad, OpenSCAD, LibreCad, Pycam, Camotics, dxf2gcode & Cura
- FEA, CFD & multiphysic simulation: Salome-Meca / Code-Aster, SalomeCFD/Code-Saturne, HelyxOs/OpenFOAM, Elmer FEM, Calculix with Launcher & CAE GUI, Impact FEM, MBDyn, FreeFEM, MFEM, Sparselizard
- Meshing, pre-post, & visualization: Salome, Paraview, Helyx-OS, Elmer GUI, VoxelMesher, Tetgen, CGX, GMSH

Questions and practicals on the HPCFS cluster

- Demonstration of the work on the cluster by repeating
- Access with NX client
- Learning basic Linux commands
- SLURM scheduler commands
- Modules
- Development with OpenMP and OpenMPI parallel paradigms
- Exercises and extensions of basic ideas
- Instructions available at http://hpc.fs.uni-lj.si/

SLING EURO

Thanks!

EuroHPC Joint Undertaking

This project has received funding from the European High-Performance Computing Joint Undertaking (JU) under grant agreement No 951732. The JU receives support from the European Union’s Horizon 2020 research and innovation programme and Germany, Bulgaria, Austria, Croatia, Cyprus, Czech Republic, Denmark, Estonia, Finland, Greece, Hungary, Ireland, Italy, Lithuania, Latvia, Poland, Portugal, Romania, Slovenia, Spain, Sweden, United Kingdom, France, Netherlands, Belgium, Luxembourg, Slovakia, Norway, Switzerland, Turkey, Republic of North Macedonia, Iceland, Montenegro