COMP3577: Parallel Scientific Computing I

Introduction

Anne Reinarz

anne.k.reinarz@durham.ac.uk

Organisation

Lecturers

- Anne Reinarz (anne.k.reinarz@durham.ac.uk),
    - numerics
- Max Fasi (massimiliano.fasi@durham.ac.uk)
    - parallelism
- Office hours: email to set something up
    - always happy to answer any questions/concerns

Organisation

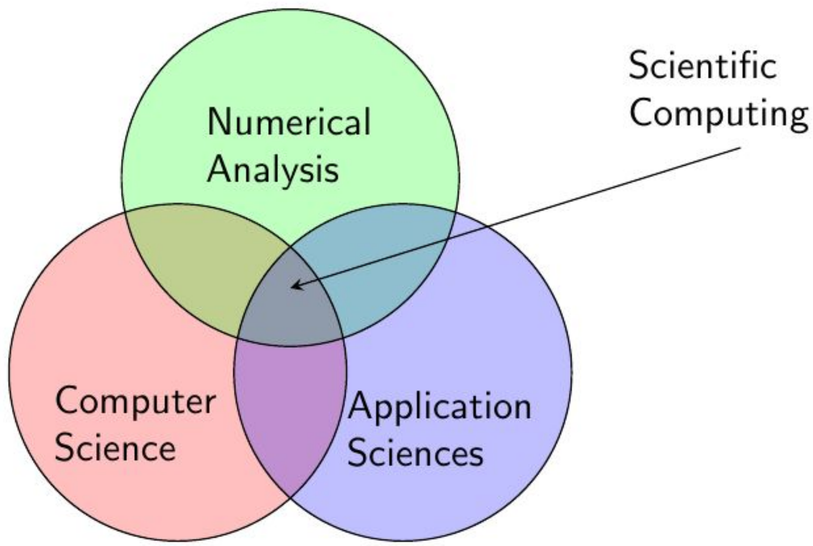
- Module is assessed purely by coursework
  - Available in teacing week 4
  - Submission via LearnUltra

Supercomputer access

- I recommend that you sign up for Hamilton
- If you request access through the web form, please include
  - the course name (Parallel Scientific Computing I) plus the code (COMP3577)
  - Provide Max Fasi or Anne Reinarz as staff contact
- The other available computer for teaching is Durham's NCC. This is a GPU-base machine and thus particular interesting for students who want to do OpenMP+GPU development. To get access to NCC, please consult our departmental DUO pages on IT equipment.

Today: Introduction

What is Scientific Computing?



Your thoughts

https://padlet.com/annekreinarz/scicomp (https://padlet.com/annekreinarz/scicomp)

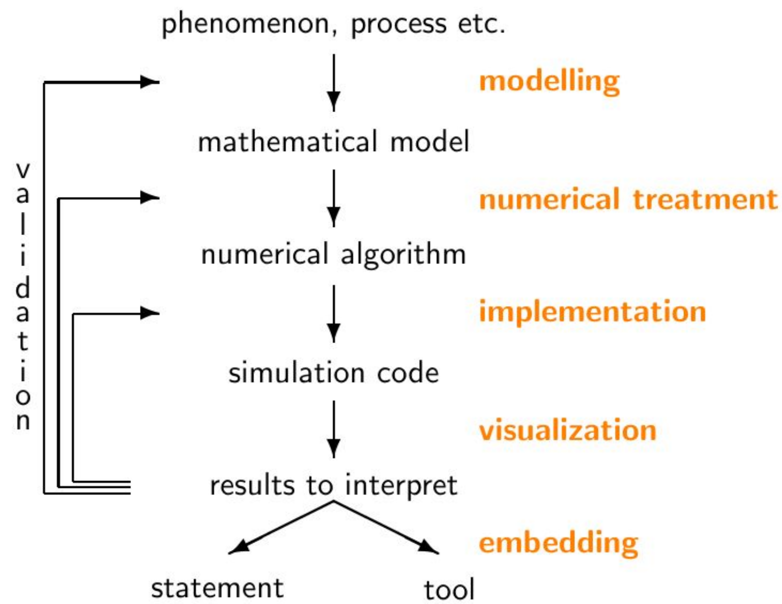
Content

- Introduction, Model Problems
- Floating Point numbers, Error propagation, Arithmetic Stability
- Ordinary Differential Equations (ODEs)
- Von Neumann Architecture, SIMD
- Conditioning and well-posedness, Stability of ODEs
- Vectorisation, Flynn's Taxonomy

Content

- Taylor expansions, Convergence order of ODEs
- Upscaling and scaling measurement
- OpenMP
- High-order Methods, Runge-Kutta Methods
- Debugging and Performance Analysis Tools

Simulation Pipeline



Further Reading

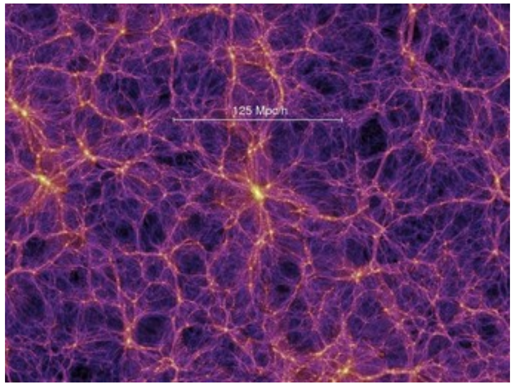
- Moshe Y. Vardi: Science Has Only Two Legs. Communications of the ACM. 53(9 page 5 (2010) http://doi.acm.org/10.1145/1810891.1810892 (http://doi.acm.c /10.1145/1810891.1810892)
- Jaenette M. Wing: Computational Thinking. Communications of the ACM. 49 (3 pp. 33–35 (2006) https://doi.org/10.1145/1118178.1118215 (https://doi.org /10.1145/1118178.1118215)
- Pdfs can be found under lecture material

What is a model?

- A model is an abstraction of a physical system into a mathematical framework
  - often with simplifying assumptions, e.g. no friction
- Models should be verifiable experiments
- Often: Ordinary or Partial Differential Equations
- Example: $F = ma = m\frac{d^2x}{dt^2}$

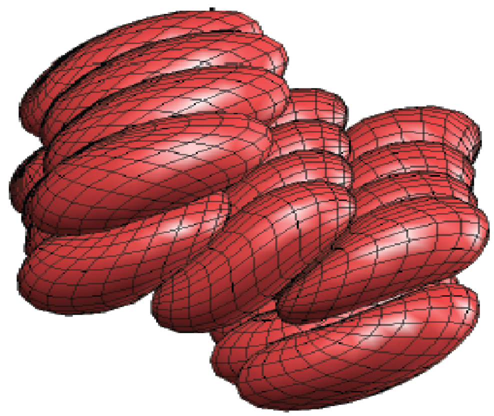
Example: Molecular Dynamics

N-Body Methods: Millennium-XXL Project



- Springel, Angulo, et al., 2010
- N-body simulation with $N = 3 \cdot 10^{11}$ ``particles''
- compute gravitational forces and effects (every ``particle'' correspond to ~ 10 suns)
- simulation of the generation of galaxy clusters plausibility of the ``cold dark matter'' model

N-Body Methods: Particulate Flow Simulation



- Rahimian, ..., Biros, 2010
- direct simulation of blood flow
- particulate flow simulation (coupled problem)
- Stokes flow for blood plasma
- red blood cells as immersed, deformable particles

Simulation - HPC-Related Data:

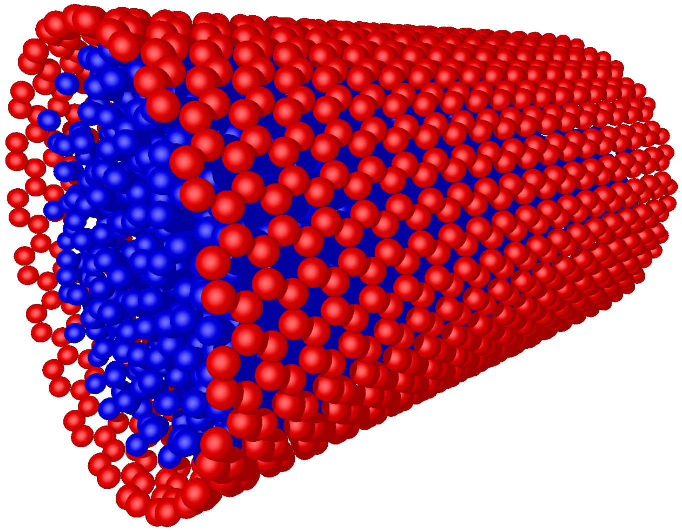
- up to 260 Mio blood cells, up to $9 \cdot 10^{10}$ unknowns
- fast multipole method to compute Stokes flow (octree-based; octree-level 4--2
- scalability: 327 CPU-GPU nodes on Keeneland cluster, 200,000 AMD cores on Jaguar (ORNL)
- 0.7 Petaflops/s sustained performance on Jaguar
- extensive use of GEMM routine (matrix multiplication)

- runtime: ≈ 1 minute per time step

- Article for Supercomputing conference: http://www.cc.gatech.edu/~gbiros/papers/sc10.pdf (http://www.cc.gatech.edu/~gbiros/papers/sc10.pdf)

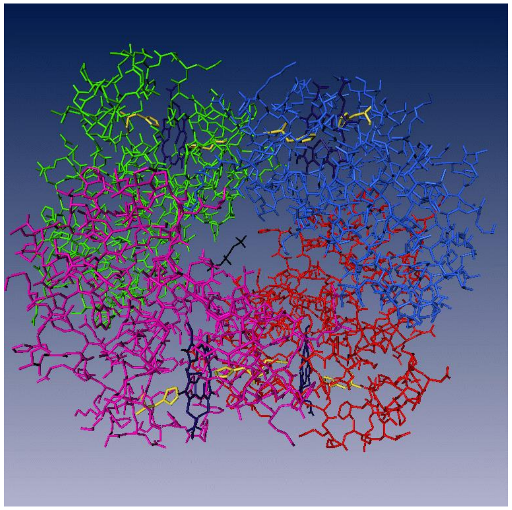
Example: Molecular Dynamics

Applications for Micro and Nano Simulations



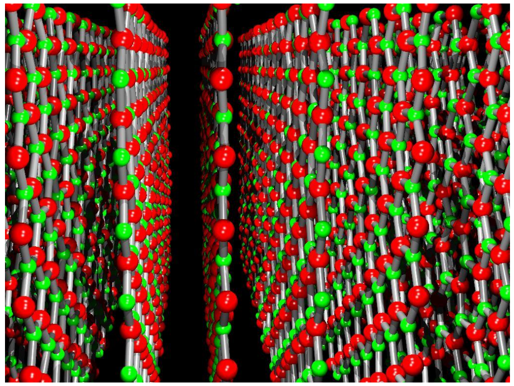
Flow through a nanotube (where the assumptions of continuum mechanics are no long
valid)

Applications for Micro and Nano Simulations



- Protein simulation: human haemoglobin (light blue and purple: alpha chains; re and green: beta chains; yellow, black, and dark blue: docked stabilizers or poter docking positions for oxygen)

Applications for Micro and Nano Simulations



- Material science: hexagonal crystal grid of Bornitrid

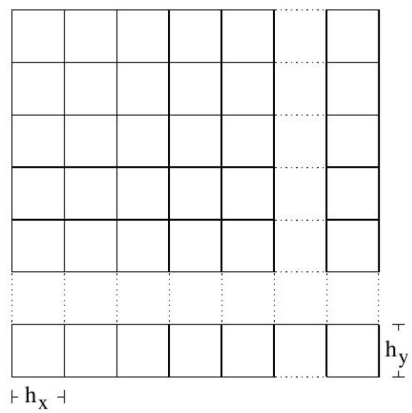
Example: Discretisation Methods

Discretisation

- Many methods for discretising differential operators exist:
    - Finite Differences
    - Finite Volume Methods
    - Finite Element Methods
    - Disconnuous Galerkin
- All transform the operator into a linear system to be solved

Finite Volume Model for Heat Equation

- object: a rectangular metal plate
- model as a collection of small connected rectangular cells



- compute the temperature distribution on this plate!

Finite Volume Model for Heat Equation

- model assumption: temperatures in equilibrium in every grid cell
- heat flow across a given edge is proportional to
  - temperature difference $(T_1 - T_0)$ between the adjacent cells
  - length h of the edge
- e.g.: heat flow across the left edge: ???

Finite Volume Model for Heat Equation

- e.g.: heat flow across the left edge:

$$q_{i,j}^{(\mathrm{left})} = k_x \left(T_{i,j} - T_{i-1,j}\right) h_y$$

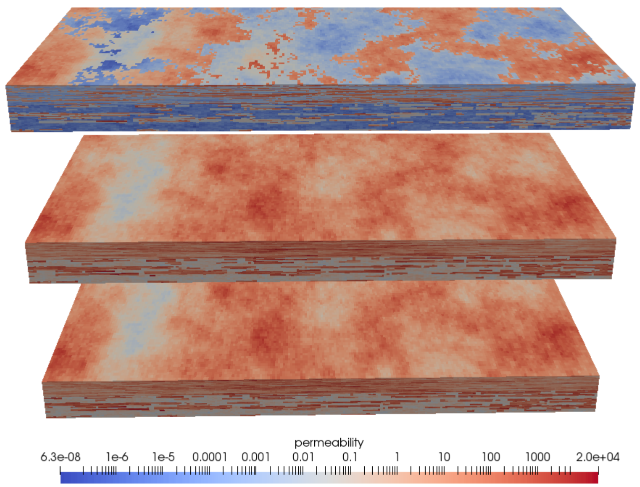
  - note: heat flow **out of** the cell (and $k_x > 0$)

- heat flow across all edges determines change of heat energy:

$$\begin{aligned} q_{ij} = k_x \left(T_{ij} - T_{i-1,j}\right) h_y + k_x \left(T_{ij} - T_{i+1,j}\right) h_y + k_y \left(T_{ij} - T_{i,j-1}\right) h_x \\ + k_y \left(T_{ij} - T_{i,j+1}\right) h_x \end{aligned}$$

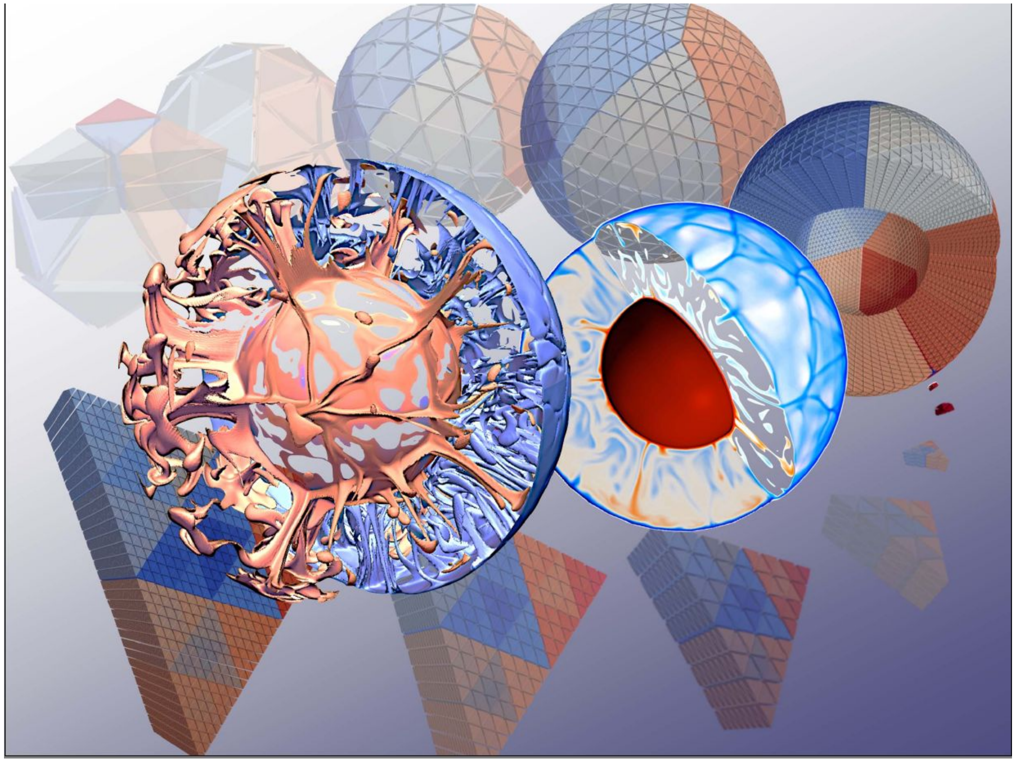
Darcy's law

- Describes flow in porous media, e.g. groundwater flow
- Poisson problem with variable coefficients



Multigrid: HHG for Mantle Convection}

- (Ruede et al., 2013; project: TERRA NEO)



Multigrid: HHG for Mantle Convection

Mantle Convection on PetaScale Supercomputers:

- mantle convection modeled via Stokes equation (``creeping flow'')
- linear Finite Element method on an hierarchically structured tetrahedral mesh
- requires solution of global pressure equation in each time step

Multigrid: HHG for Mantle Convection

Weak Scaling of HHG Multigrid Solver on JuQueen:

- geometric multigrid for Stokes flow via pressure-correction
- pressure residual reduced by 10^{-3} (A) or 10^{-8} (B)

Nodes	Threads	Grid points	Resolution	Time: (A)	(B)
1	30	$2.1 \cdot 10^{07}$	32 km	30 s	89 s
4	240	$1.6 \cdot 10^{08}$	16 km	38 s	114 s
30	1 920	$1.3 \cdot 10^{09}$	8 km	40 s	121 s
240	15 360	$1.1 \cdot 10^{10}$	4 km	44 s	133 s
1 920	122 880	$8.5 \cdot 10^{10}$	2 km	48 s	153 s
15 360	983 040	$6.9 \cdot 10^{11}$	1 km	54 s	170 s

In [ ]:

```

```

In [ ]:

```

```

In [ ]:

```

```