Imperial College London

Pathways Towards Unstructured Mesh Adaptivity for Reservoir Modelling

James Percival, Pablo Salinas, Dimitrios Pavlidis, Zhihua Xie, Chris Pain, Matthew Jackson

Applied Modelling and Computation Group

Novel Reservoir Modelling and Simulation Group

Presentation to NORMS

12th November 2014

DEPARTMENT OF
EARTH SCIENCE AND ENGINEERING

Outline

- Background (What is Mesh Adaptivity?)
- Motivation (Why Adapt Meshes?)
- Current Methodology (How does IC-FERST adapt meshes?)
- Discussion (How to be better?)

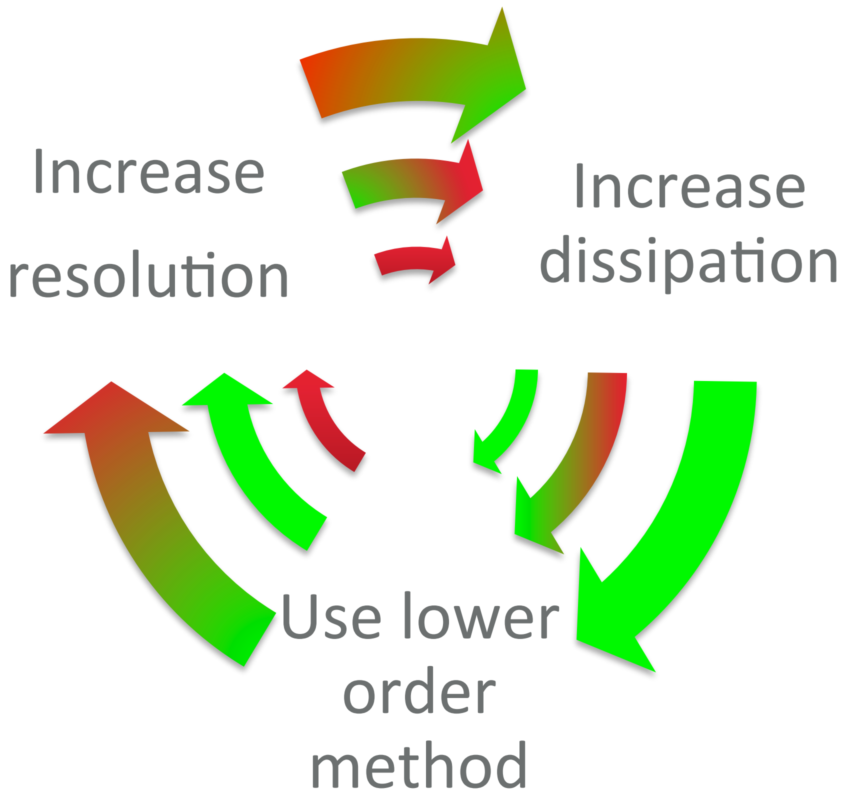
Imperial College London

Background – The goals of numerical simulation

In numerical simulation there are 3 key concerns:

- Accuracy – simulation is a good representation of real behaviour of the system
- Stability – answers remain physically relevant.
- Cost – Time/expense of generating solution

Often imposssible to achieve all of fast, accurate & robust.



Background – Why discretize?

Computers (& humans) have finite memories

PDEs contain information at infinite no. of scales

$$\frac{\partial \phi S_k}{\partial t} + \nabla \cdot \boldsymbol{u}^{\text{Darcy}} = 0$$

$$S := S(\boldsymbol{x}, t)$$

Physics in PDEs converted into linear algebra via a choice of discretization

1. Finite difference
2. Finite Volume
3. Spectral Methods
4. Finite element
5. Mesh Free methods

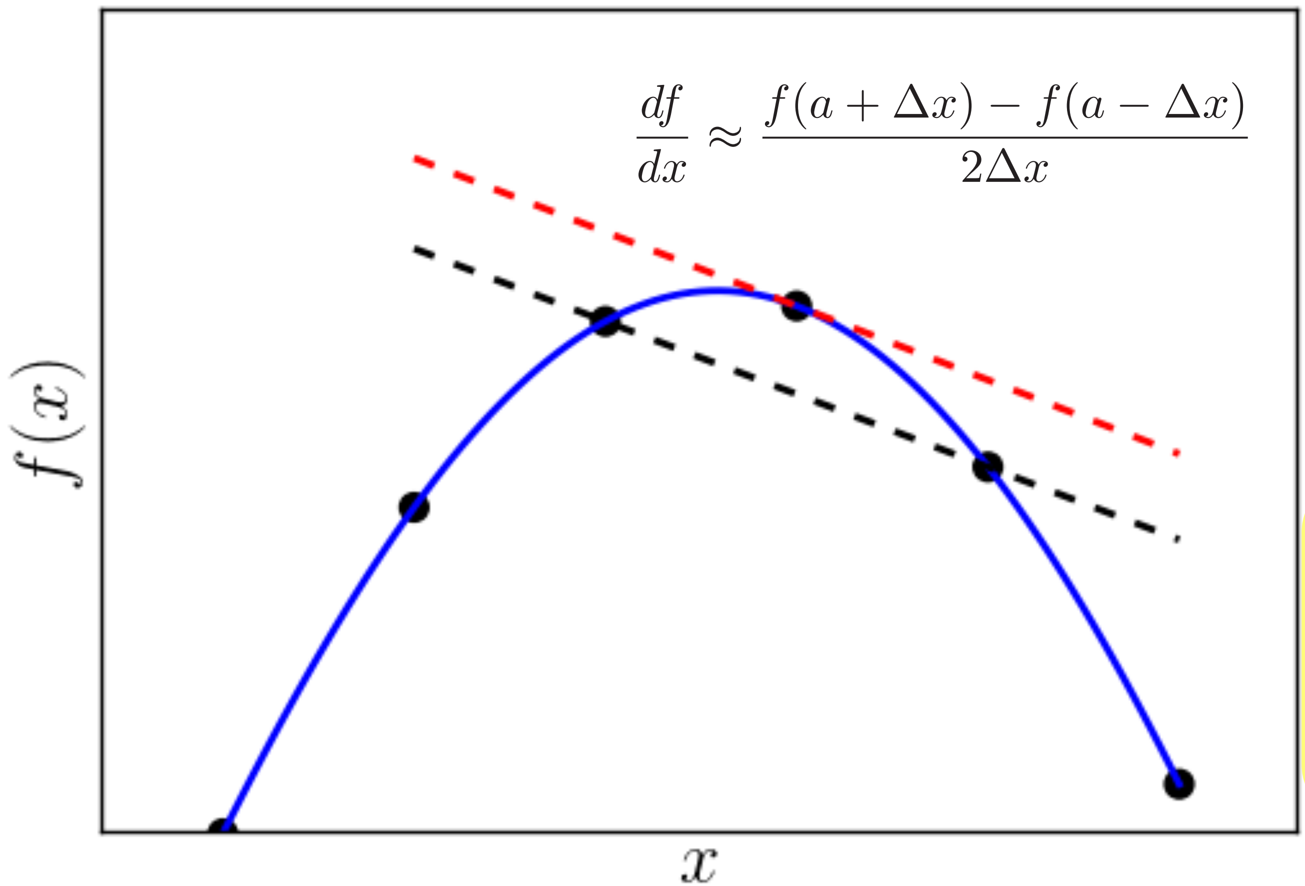
Notice! Different discretization methods often combined.

	Integer	Floating point
16 bit	-32,768	
32 bit	−2,147,4	±1.4 x10-45 to 3.4 x10^{38} (~7.2 sig. fig)
64 bit	−9,223,372,036,854,775,808 to +9,223,372,036,854,775,807	10^{-308} to 10^{308} (16 sig. fig)

Background – Finite Differences

Finite differences – Schoolboy calculus

$$f\,(a+\Delta x) = f(a) + \Delta x\,\frac{df}{dx}\bigg|_{x=a} + \frac{\Delta x}{2}\,\frac{d^2 f}{dx^2}\bigg|_{x=a} + \frac{\Delta x}{6}\,\frac{d^3 f}{dx^3}\bigg|_{x=a} + \ldots$$



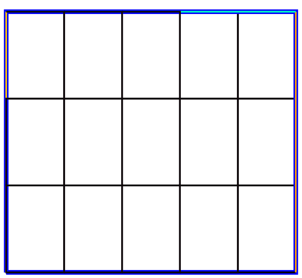
Notice! Implementation is usually simpler (& more accurate) on structured uniform meshes

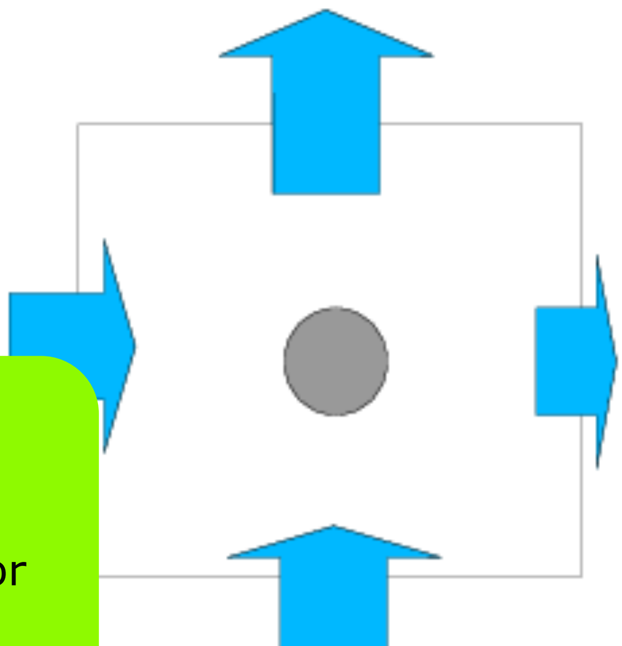
Imperial College London

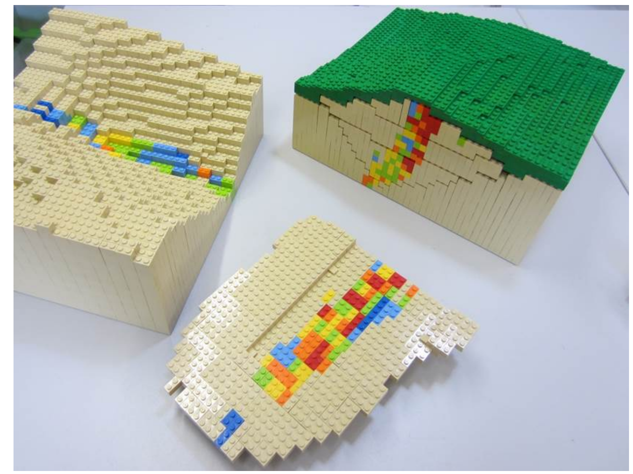
Background – Finite Volumes

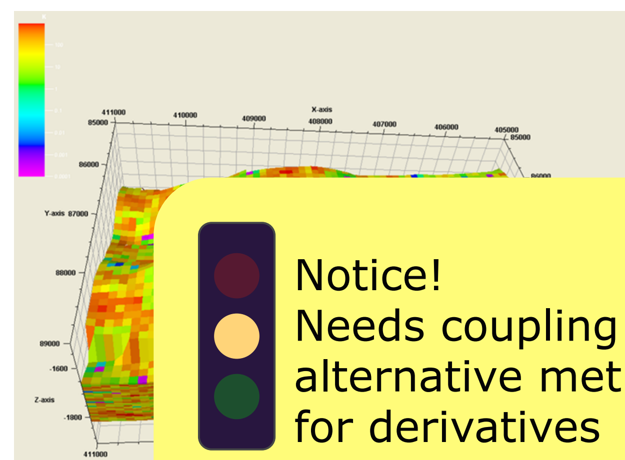
"Lego science"

$$\frac{d}{dt}\int_{\Omega_i} \rho dV = \sum_{\text{faces}} \int_{\delta\Omega_i^{(j)}} \rho \boldsymbol{u} \cdot \boldsymbol{n} dS$$









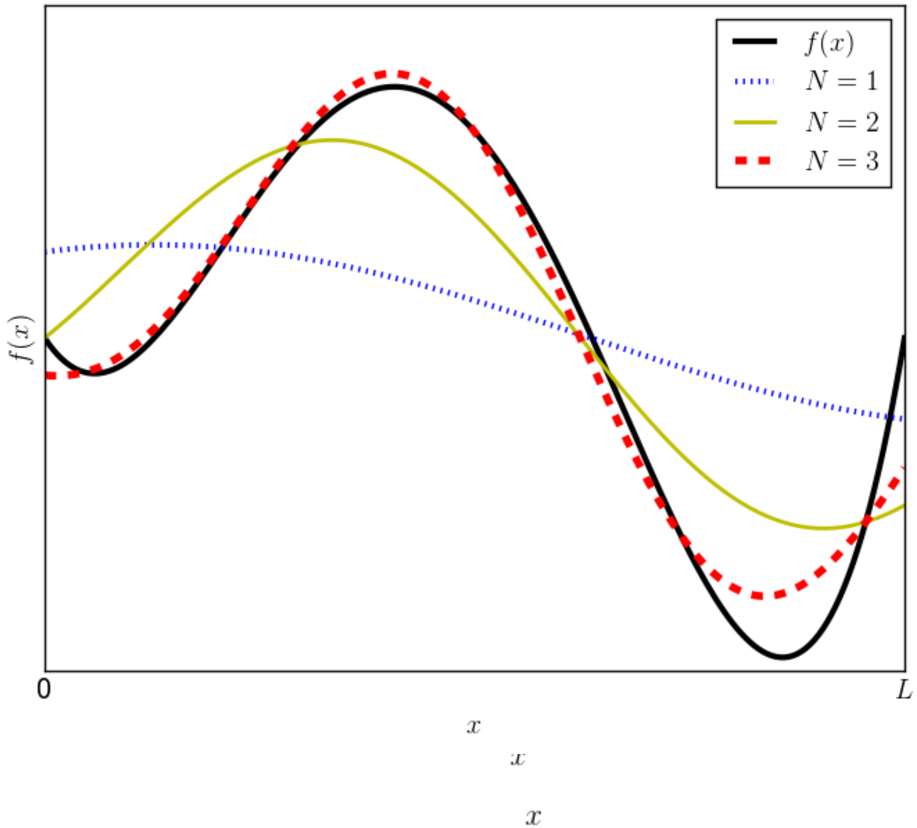
Bonus!
Ensures local conservation for (almost) free

Notice!
Needs coupling to alternative method for derivatives

Imperial College London

Background – Spectral Methods

Spectral methods – basis functions



$$f(x) = \sum_{n=1}^{\infty} a_n \cos\left(n\frac{\pi x}{L}\right) + b_n \sin\left(n\frac{\pi x}{L}\right)$$

$$\approx \sum_{n=1}^{N} a_n \cos\left(n\frac{\pi x}{L}\right) + b_n \sin\left(n\frac{\pi x}{L}\right)$$

$$\int_\Omega \cos(ax)\cos(bx)dx = C\delta_{ab}$$

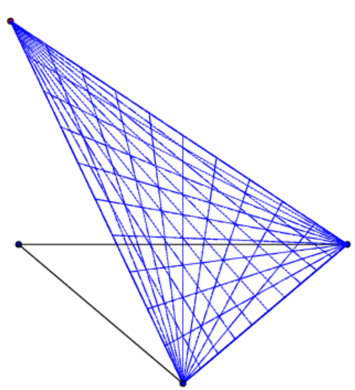
Notice!
Doesn't much like boundary conditions or complex geometries

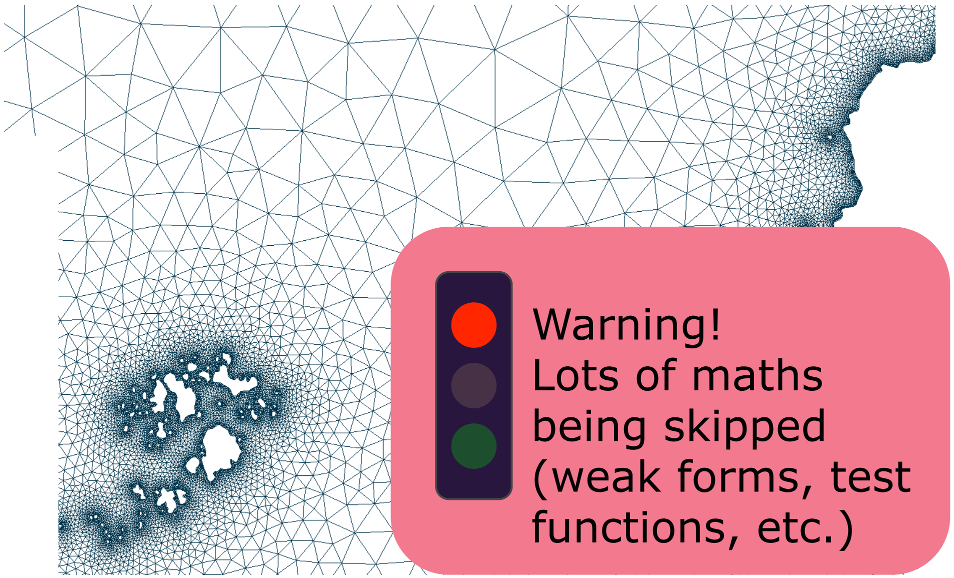
Background – Finite Elements

Couple basis functions

+ partition into simple shapes

$$\sum_j \int_\Omega N_i N_j \psi_j \, dV = \int_\Omega N_i f(\boldsymbol{x}) dV$$



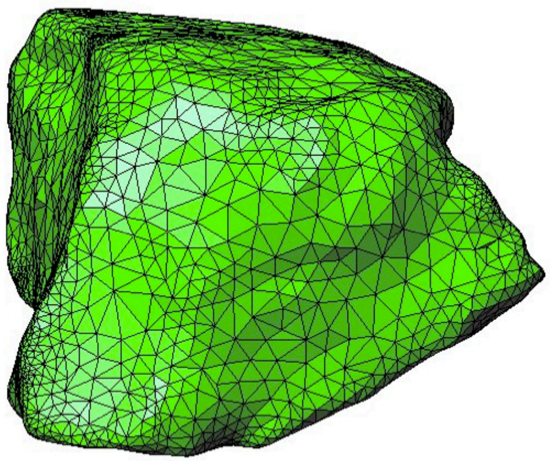


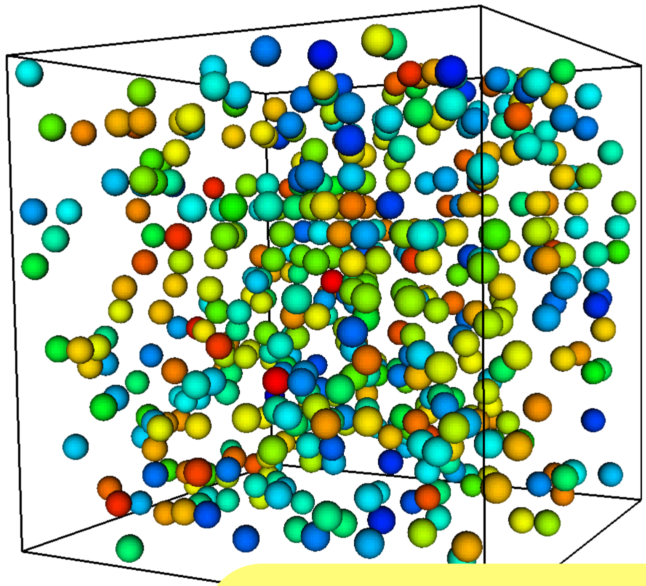
Warning!
Lots of maths being skipped (weak forms, test functions, etc.)

Imperial College London

Background - Mesh Free Methods - DEM

FEM + DEM

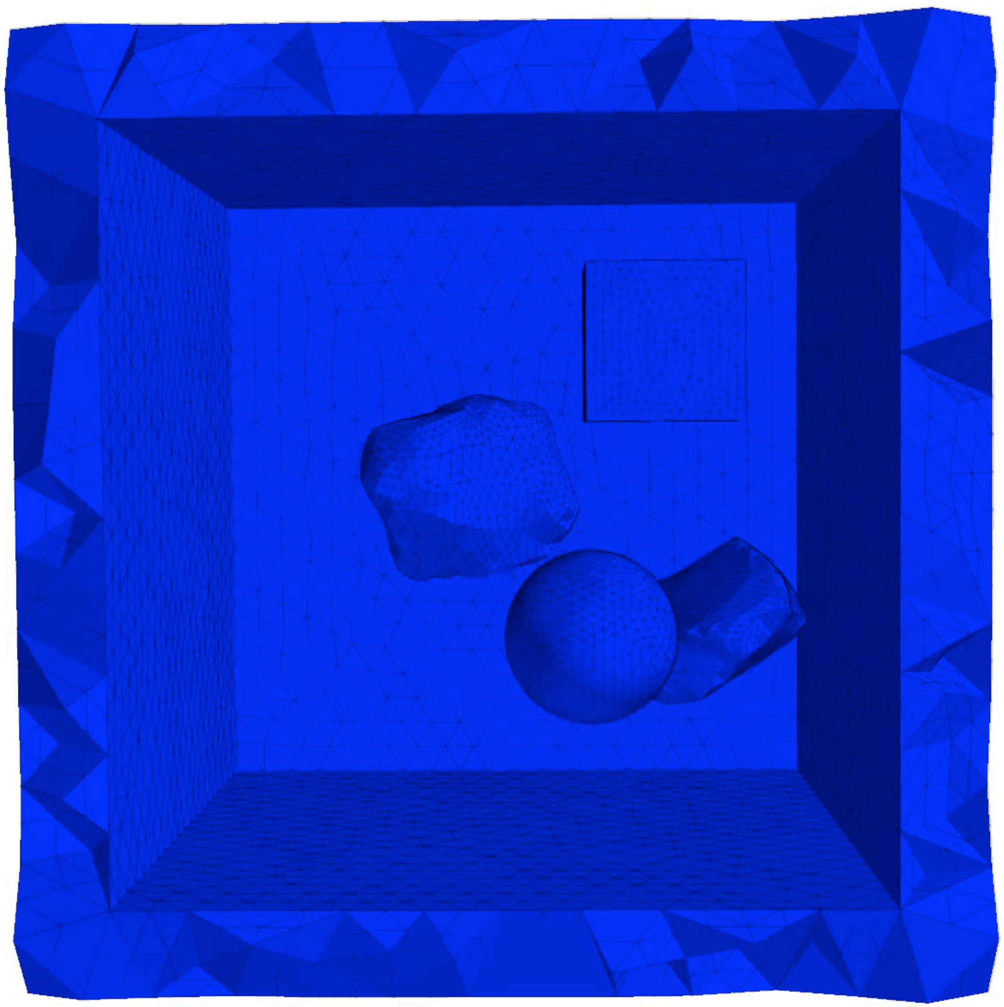




Notice!
Slides from (and questions to) Jiansheng Xiang

Imperial College London

What is FEMDEM?

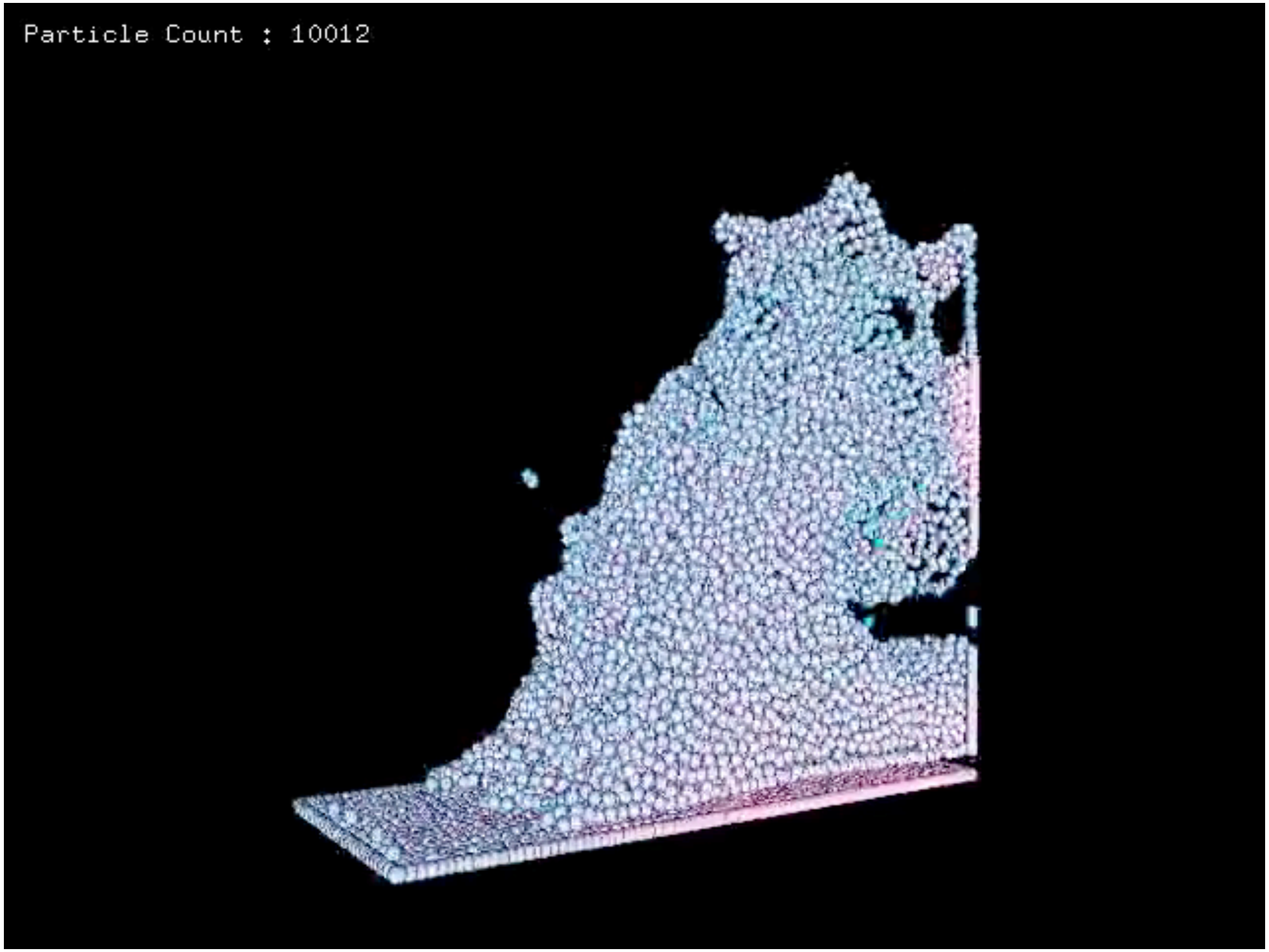


Website:
http://www.vgest.net

Xiang, J. et al. 2009. Finite strain, finite rotation quadratic tetrahedral element for the combined finite-discrete element method. *Int Journal for Numerical Methods in Engineering DOI: 10.1002/nme.2599*

Mesh Free Methods

Smoothed Particle Hydrodynamics (SPH)



Slide from Youtube

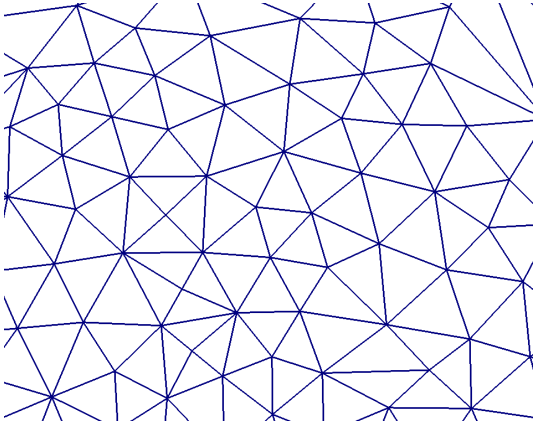
M. Müller

Background – Hybrid Methods

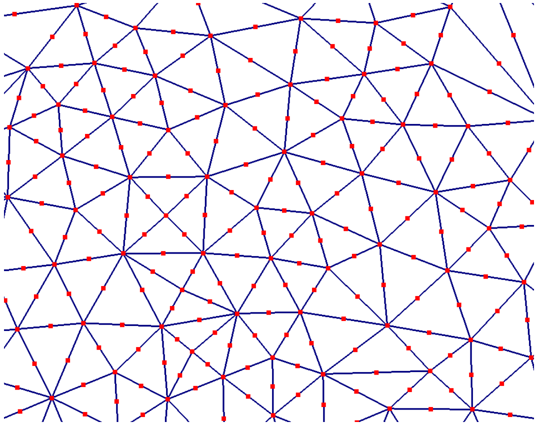
Methods combined.

Eg. Finite Element Method+ Discrete Element Method = FEMDEM

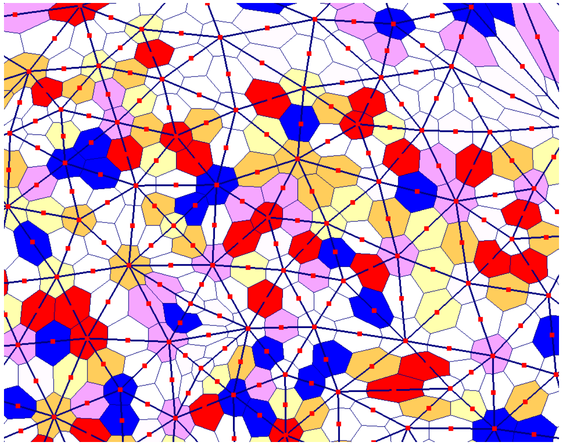
Control volumes: - Just finite volumes piggybacking on finite elements



Finite Element Mesh



Finite element "nodes"



Voronoi dual mesh
Control volumes

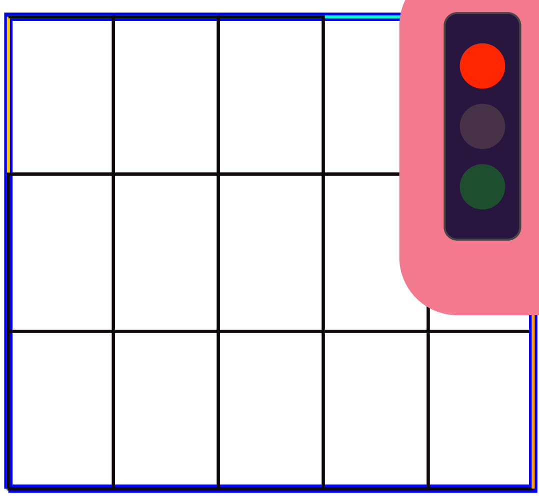
Imperial College London

Motivation – Meshes & Grids

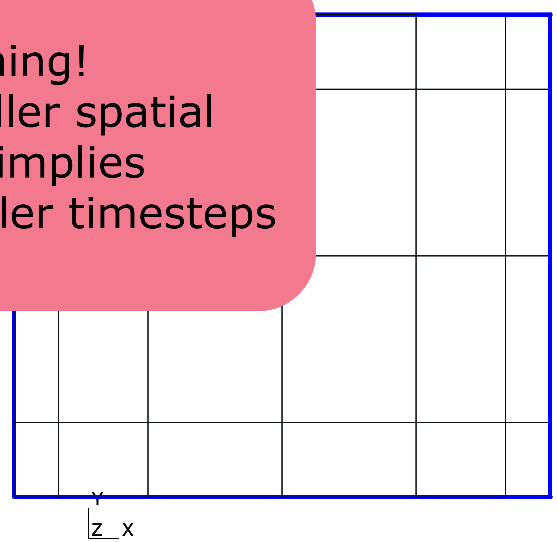
Meshs/grids are ubiquitous in computational science

Discretization imposes length scale on the problem

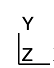
Trade-off: high resolution discretizations often more accurate, but take more time to simulate



Warning! Smaller spatial grid implies smaller timesteps



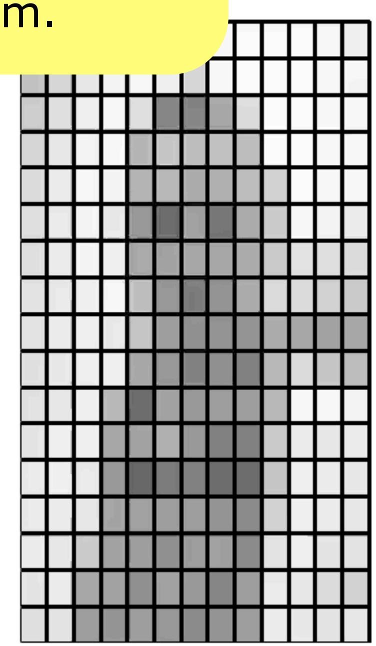
Idea: Only put high resolution "where it needs to be" dynamically.

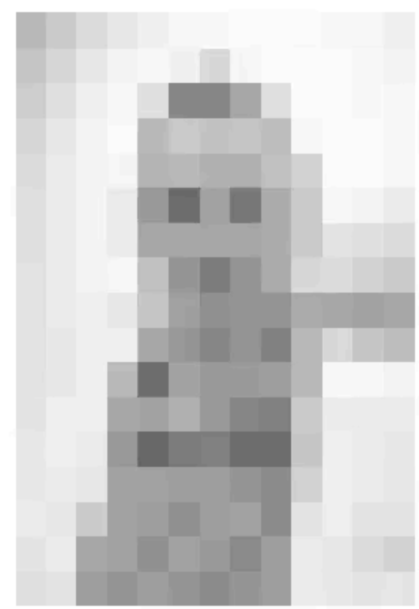


Motivation - AMR

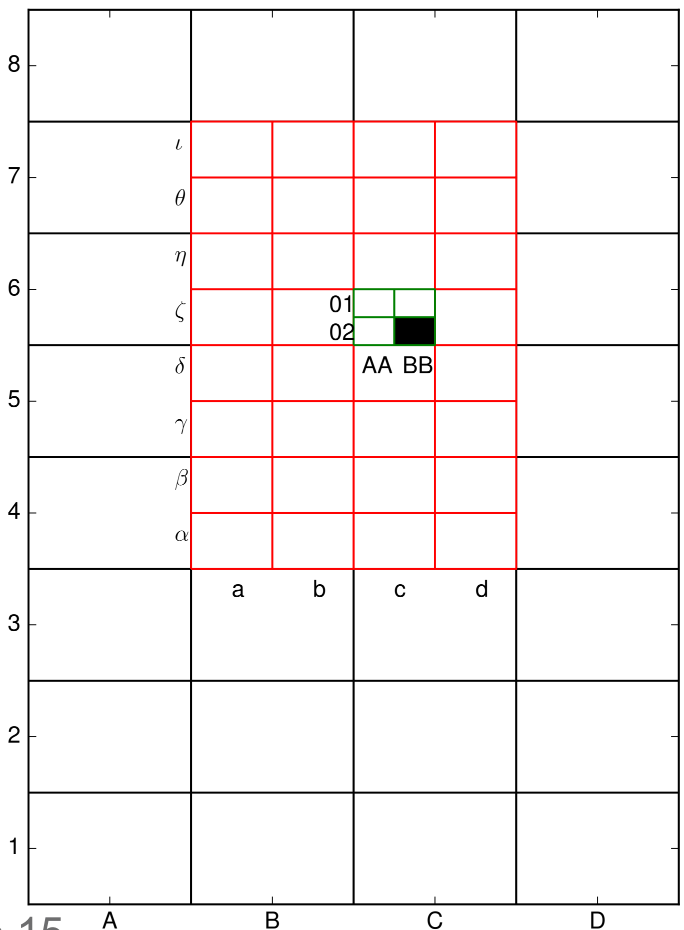
Idea has been applied on structured meshes:
...ive Mesh Refinement (AMR)

Notice!
The video is created by hand, coarsening the high res photo.
No algorithm.





Motivation - AMR



Adaptive Mesh Refinement:

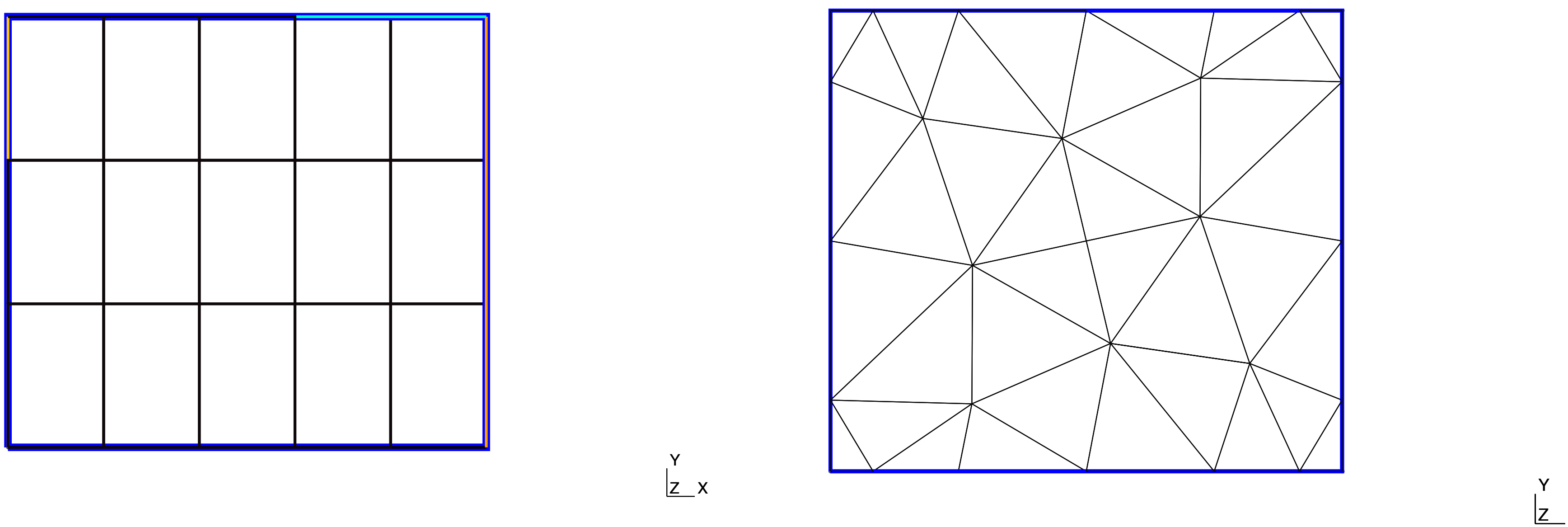
Refinement forms a tree:
Cell BB02 is in cell cγ is in Cell C6

Convenient for implementation

Less good for physics
- Lots of wasted resolution (especially in 3D)
- Can only coarsen on predefined scales
- Factors of 2 everywhere.

Motivation/Methodology

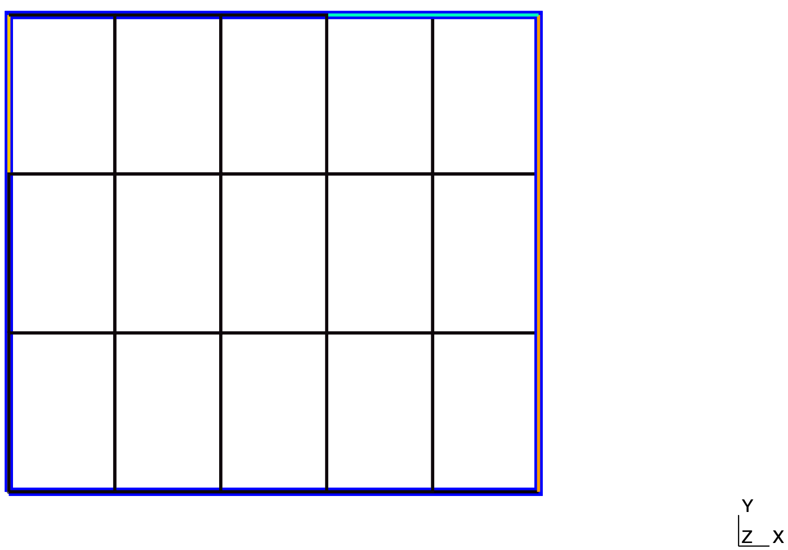
Fluidity uses unstructured meshes. Can we do better?

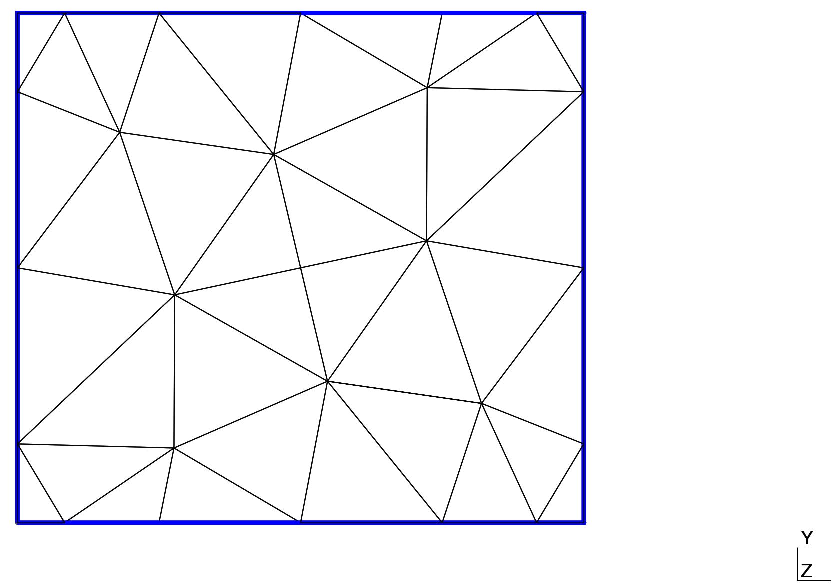


Motivation/Methodology

Yes:

- Fewer wasted regions of high resolution
- Not forced to preserve initial orientation or grid lengths
- Length scales can differ along/across structures (anisotropy)
- Fewer factors of 2.



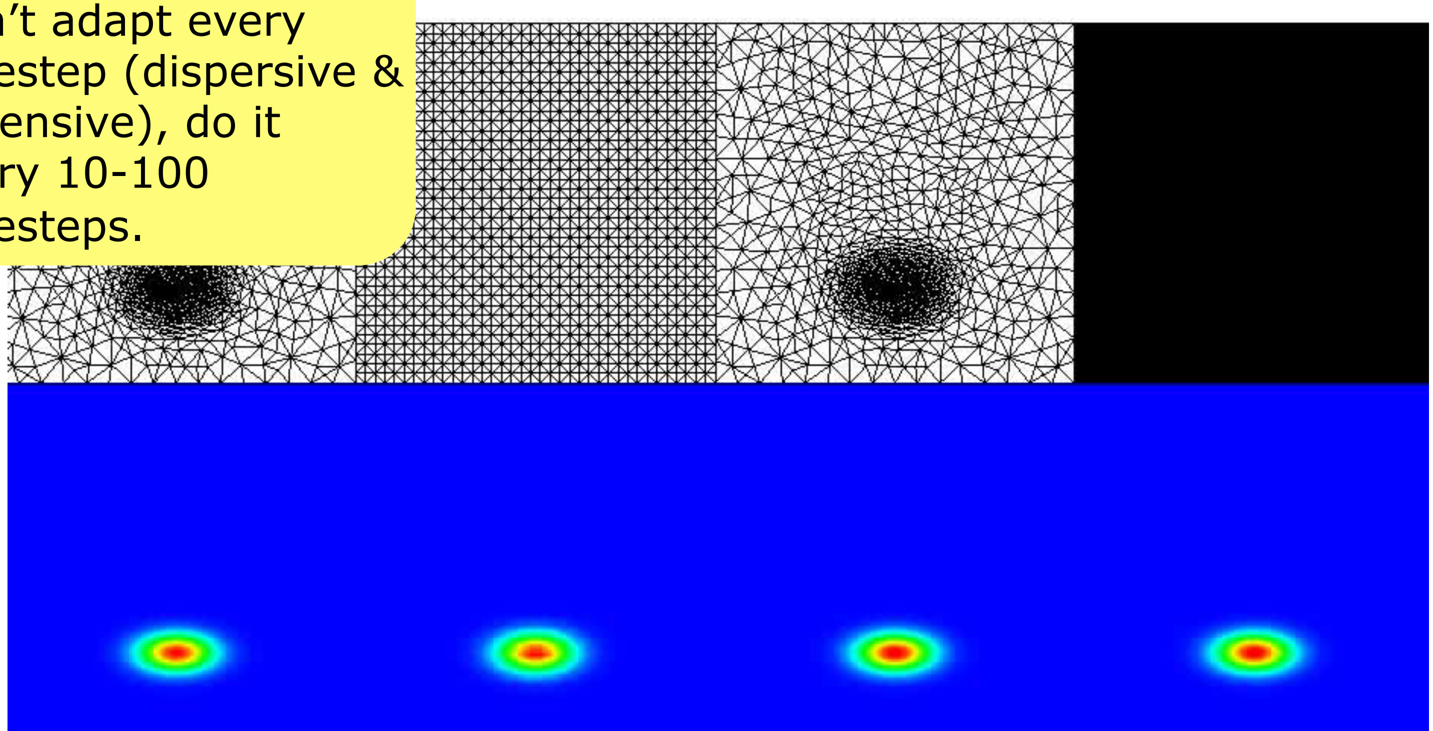


Methodology – An example of Mesh Adaptivity

Advection of a Gaussian bubble

Notice!
Don’t adapt every timestep (dispersive & expensive), do it every 10-100 timesteps.

Fixed mesh 1 | Adaptive mesh | Fixed mesh 2



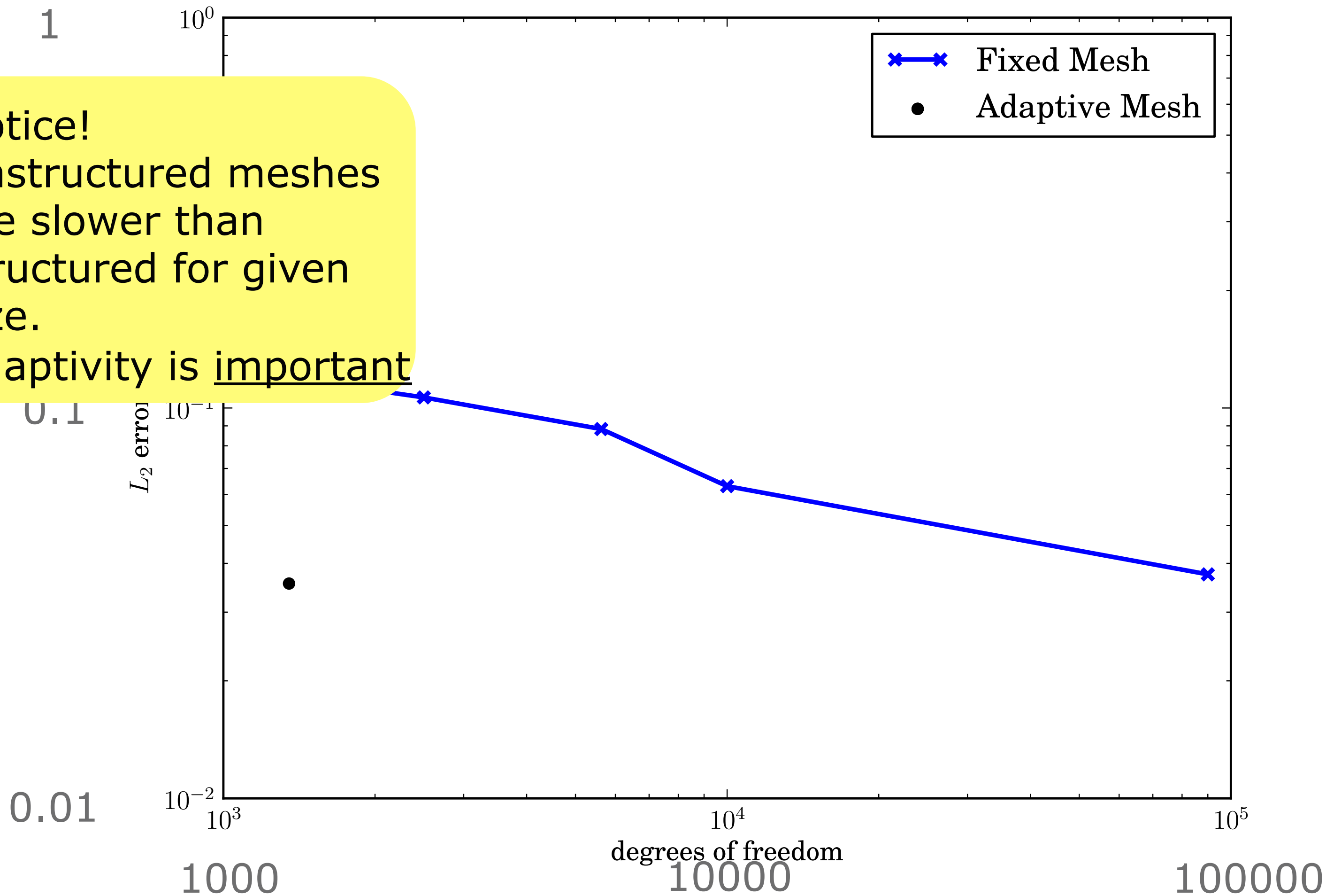
Analytic solution | Fixed mesh 1 | Adaptive mesh | Fixed mesh 2

Mesh Adaptivity: Faster & more accurate solutions

2 orders of magnitude smaller problem/half error

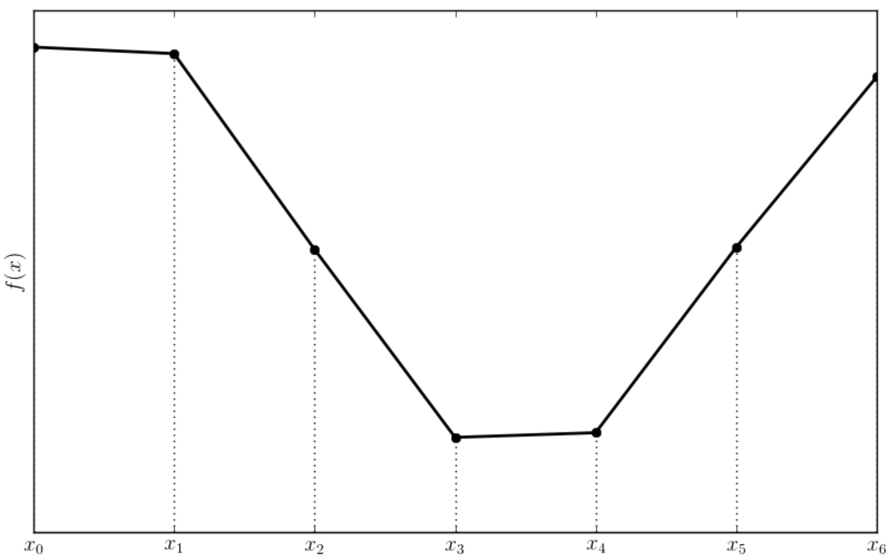
30x speed up

Notice!
Unstructured meshes are slower than structured for given size.
Adaptivity is <u>important</u>



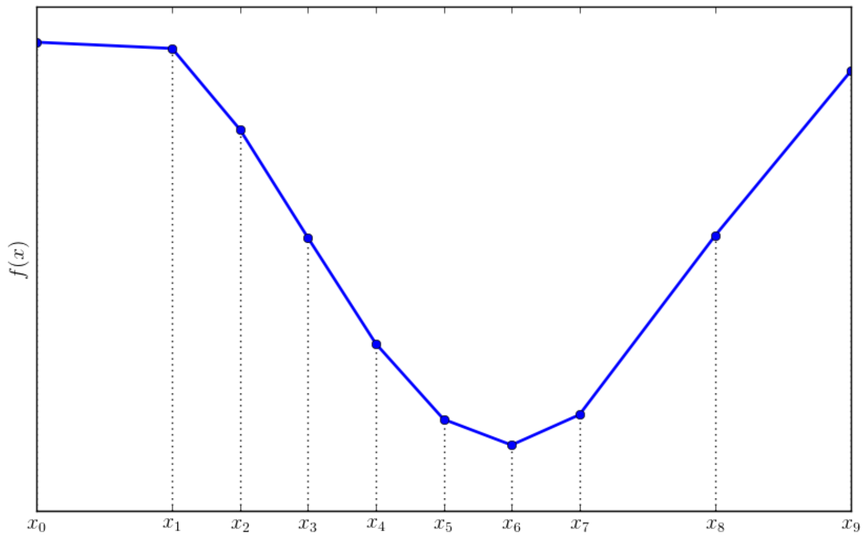
Imperial College London

Current Methodology – *h* adaptivity

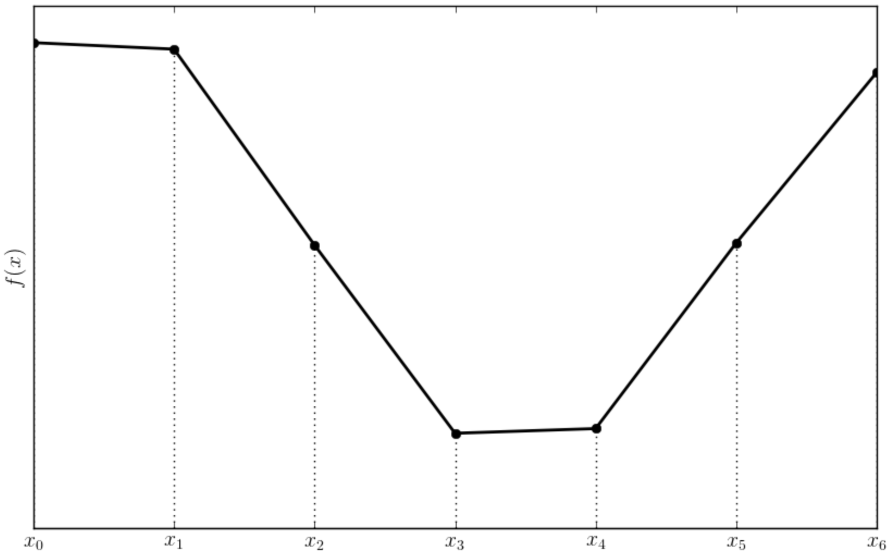


h – for step length
like delta x in
finite difference notation

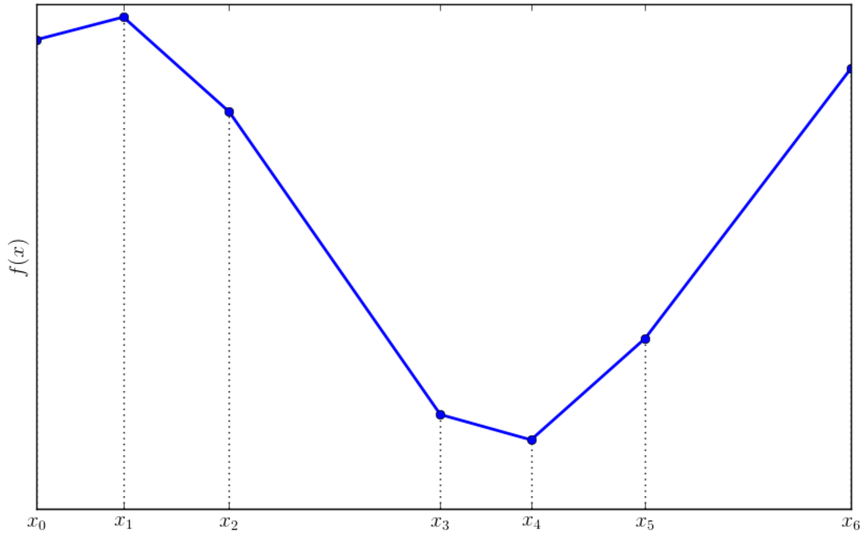
Just like AMR for elements



Current Methodology – *r* adaptivity



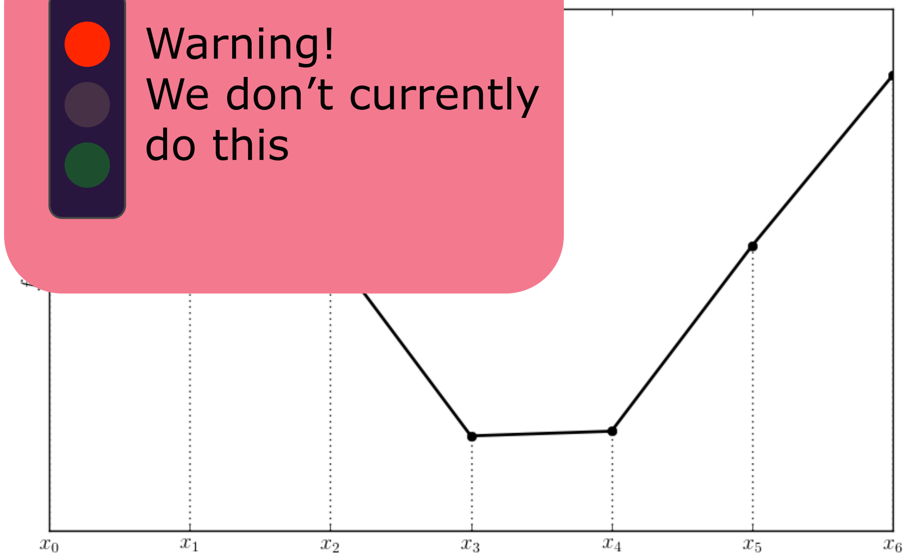
r for
redistrication/
repositioning



Imperial College London

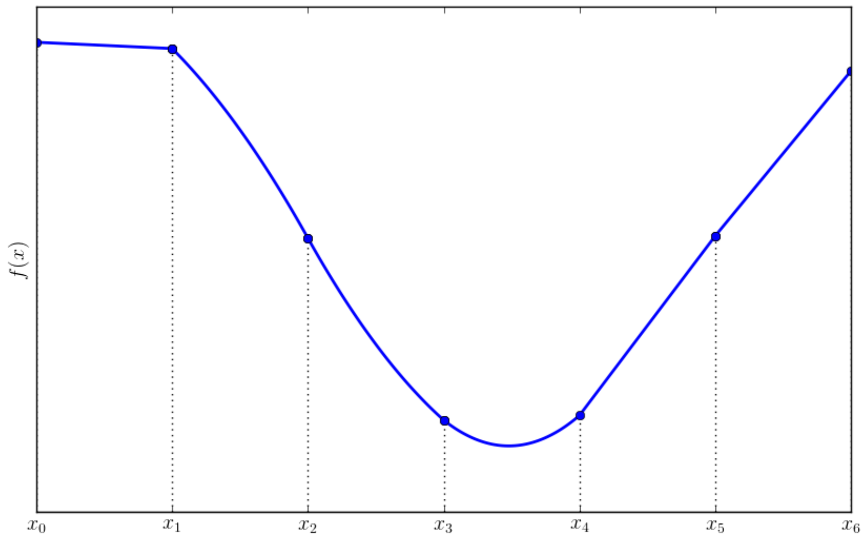
Current Methodology – *p* adaptivity

Warning!
We don’t currently do this



p - adaptivity

P for polynomial order



Current Methodology – Error estimates

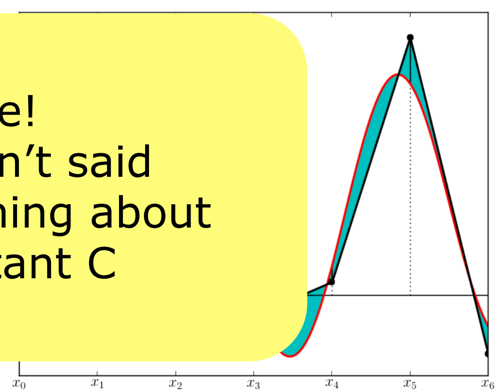
Motivation: Céa’s Lemma

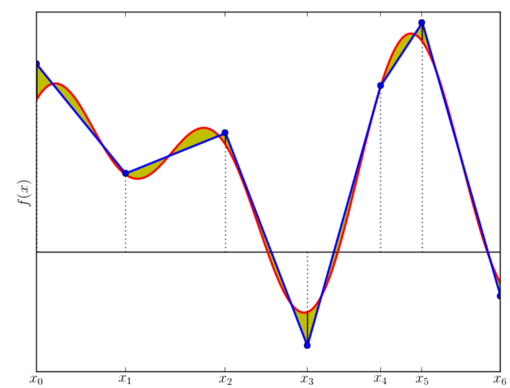
For sufficiently nice PDEs and linear (or better) elements

$$\text{error} := \left\| \psi^{\text{exact}} - \psi^{\delta} \right\| \leq C_1 \left\| \psi^{\text{exact}} - \psi^{\text{proj}} \right\|$$

$$\leq C_2 \sum_i h_i^2 \max_{x \in \Omega} \left| \frac{\partial^2 \psi}{\partial x^2} \right|$$

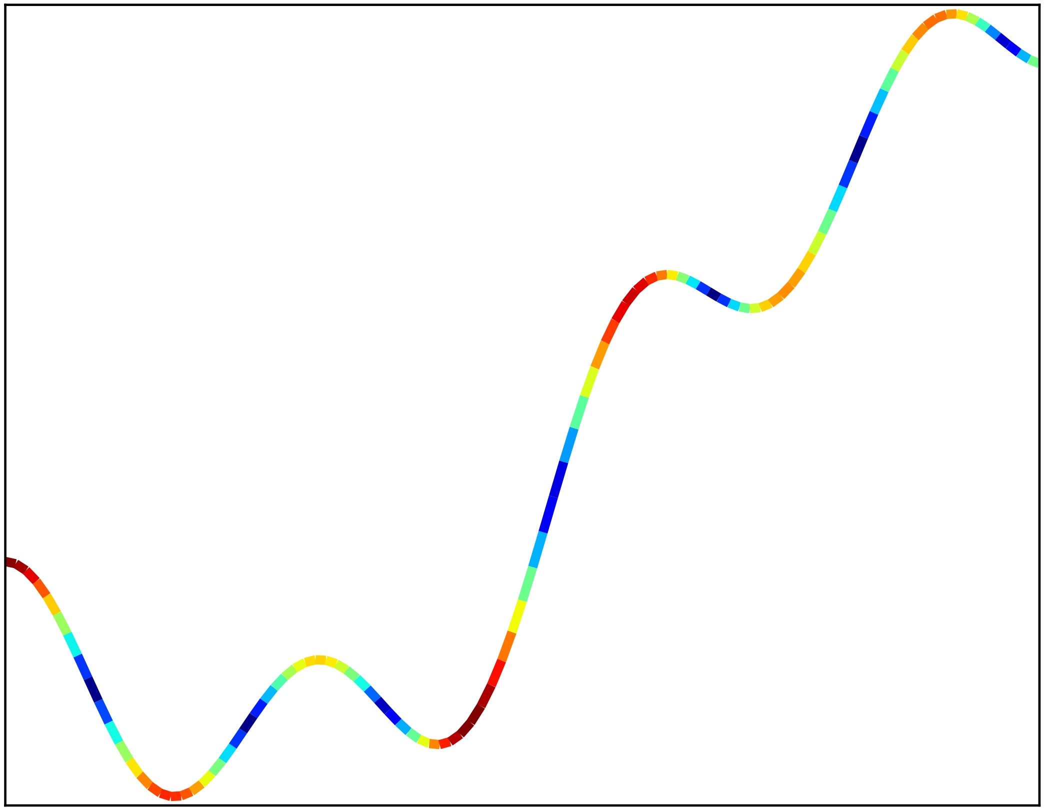
Notice!
Haven’t said anything about constant C





Imperial College London

Current Methodology – Interpolation error estimates



Plotting a linear function with areas of high curvature coloured red

Places where a linear approximation is bad.

Imperial College London

Current Methodology – Interpolation error estimates

In higher dimensions, error estimate is a function of the Hessian and the edge vectors.

$$\mathcal{H}(\psi) = \begin{pmatrix} \frac{\partial^2 \psi}{\partial x^2} & \frac{\partial^2 \psi}{\partial x \partial y} & \frac{\partial^2 \psi}{\partial x \partial z} \\ \frac{\partial^2 \psi}{\partial x \partial y} & \frac{\partial^2 \psi}{\partial y^2} & \frac{\partial^2 \psi}{\partial y \partial z} \\ \frac{\partial^2 \psi}{\partial x \partial z} & \frac{\partial^2 \psi}{\partial y \partial z} & \frac{\partial^2 \psi}{\partial z^2} \end{pmatrix}$$

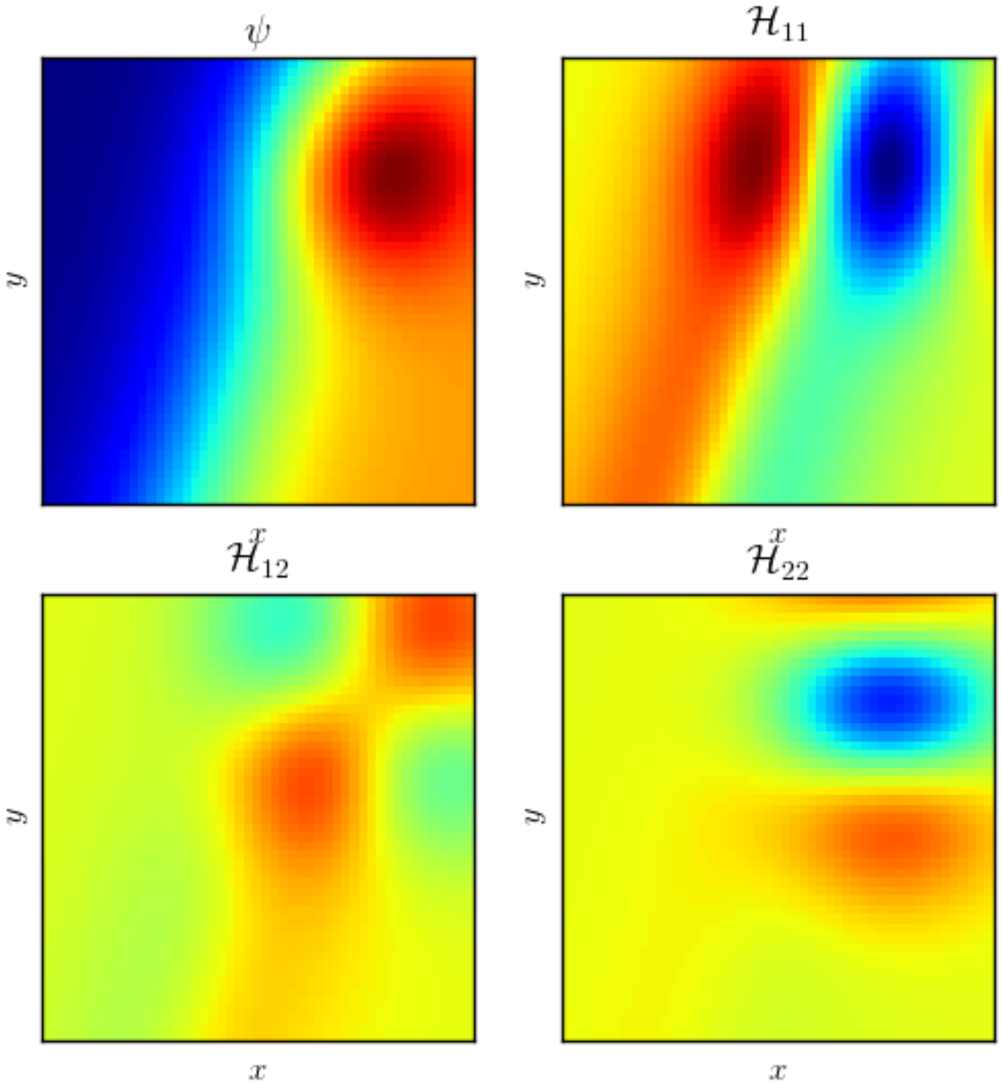
$$\text{error} \sim \sum_k \boldsymbol{v}_k^T \mathcal{M} \boldsymbol{v}_k$$

$$\mathcal{M} = \mathcal{M}(\mathcal{H}(\psi))$$

The **v**s are the edges of the mesh

"Mesh metric"

Current Methodology – Interpolation error estimates

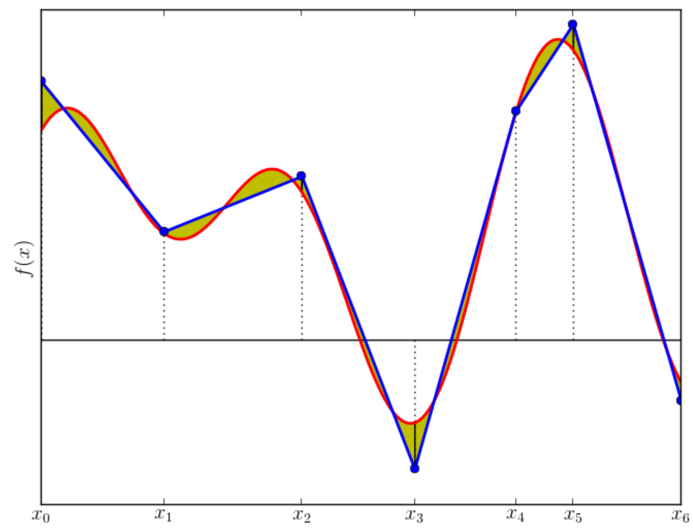


Pick out areas of curvature in physical space or in value.

Anistropic: Allowable scales along a front larger than acrros it.

Current Methodology – Interpolation Error Estimates

Problem: we don't know ψ^{exact}
Answer: Use ψ^{δ} instead.



Estimate second derivatives from the finite element data

Need a signal present in the data before we can adapt to it

Try to extremize

$$I(\boldsymbol{v}) = \sum_k \frac{1}{\epsilon} \boldsymbol{v}_k^T \mathcal{M} \boldsymbol{v}_k - 1$$

The **v**s are the edges of the mesh

$$\mathcal{M} = \mathcal{M}\left(\mathcal{H}(\psi^{\delta})\right)$$

Current Methodology – Optimization

Two possibilies once metric is calculated:

1. Global remeshing: “Slash and burn” Delete old mesh structure and create new mesh from blank sheet of paper
2. Local remeshing: “Patch and fix” Preserve the old mesh in regions where it is adequate

IC-FERST impliments the second option.

- Advatanges in speed, data retention and parallelism

Apply *hr* adaptivity in the regions where fixing is necessary: Add, remove nodes, reconnect nodes and move nodes.

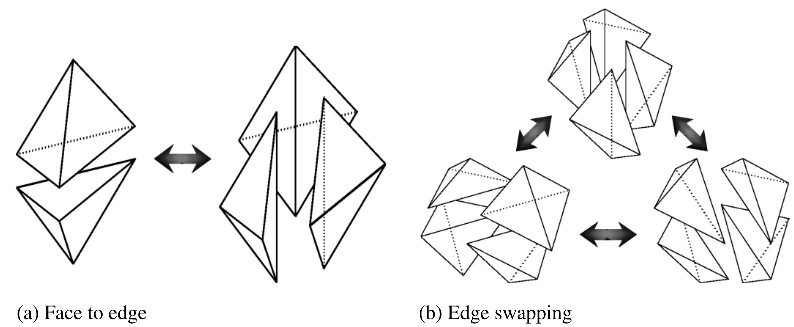


Fig. 1. Digram showing: (a) edge to face and face to edge swapping; (b) edge to edge swapping with four elements.

Pain et al. 2001

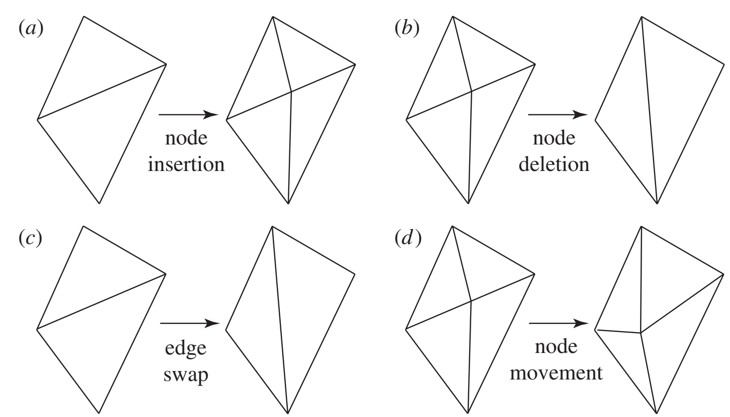


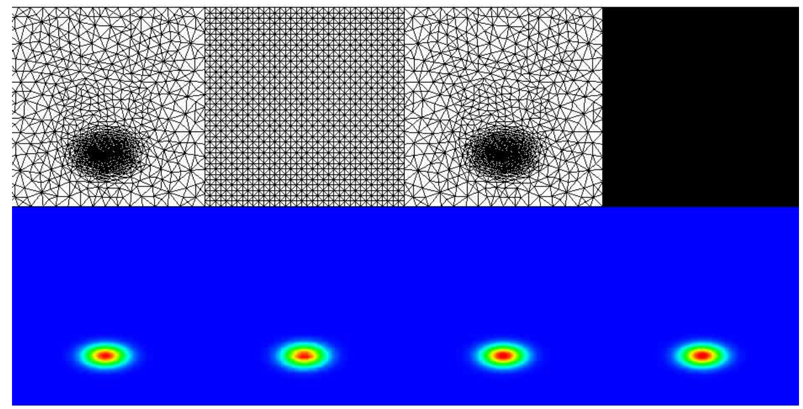
Figure 1. Local element operations used to optimize the mesh in two dimensions. (*a*) Node insertion or edge split. (*b*) Node deletion or edge collapse. (*c*) Edge swap. (*d*) Node movement.

Piggott et al 2009

Mesh metrics - Metric Advection

Adaptivity algorithm attemps to minimize error for given data at given time level.

$$\frac{\partial \mathcal{M}}{\partial t} + \boldsymbol{u}\left(t_0\right) \cdot \nabla \mathcal{M} = 0, \quad t \in [t_0, t_f]$$



Want a mesh that remains good in the future too.

For advection dominated problems, treat metric as another advected quantity.

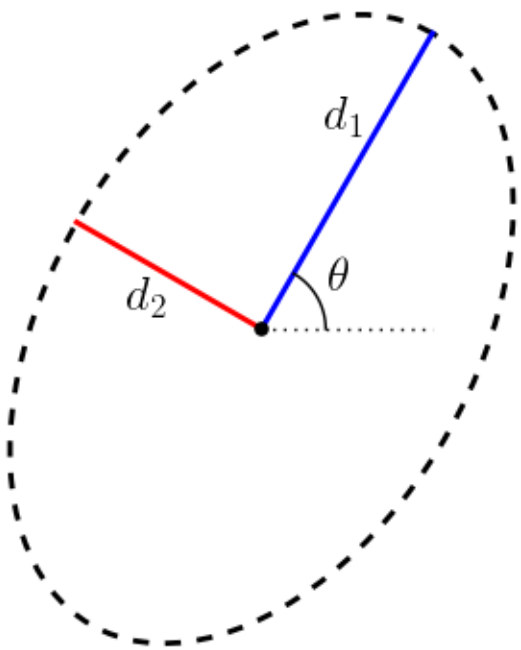
Extremise for

$$\sum_k \frac{1}{\epsilon} \mathrm{max}_{\tau \in [t_0, t_f]} \left(\boldsymbol{v}_k^T \mathcal{M}\left(\tau\right) \boldsymbol{v}_k \right) - 1$$

Mesh metrics – bounding edge length

Usually have constraints on:

- Minimum edge length [cost]
- Maximum edge length [accuracy]



Simple – do surgery on specified metric after it's calculated

$$\Phi = \begin{pmatrix} a_{11} & a_{21} \\ a_{21} & a_{22} \end{pmatrix}$$

$$= R \begin{pmatrix} d_1 & 0 \\ 0 & d_2 \end{pmatrix} R^T$$

$$R = \begin{pmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{pmatrix}$$

$$\boldsymbol{v}^T \Phi^{-1} \Phi^{-1} \boldsymbol{v}$$

Mesh metrics - Gradation

Often a bad idea to have length scales changing very rapidly between neighbouring elements. (conditioning)

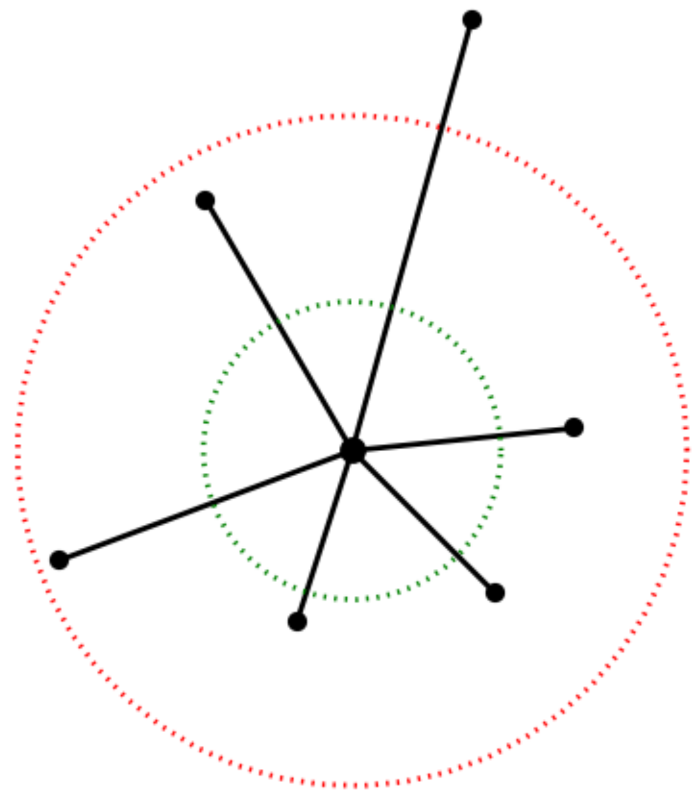
Post-processing of M can enforce smooth increases in the metric away from minima:

Isotropic

$$\|\boldsymbol{v}_i\|/\|\boldsymbol{v}_j\| \leq 1.5$$

Anisotropic

$$\boldsymbol{v}_i \cdot \underline{\Gamma} \cdot \boldsymbol{v}_j \leq \|\boldsymbol{v}_j\|^2$$



Imperial College London

Mesh Metrics - Other knobs and levers

Aspect ratio –like gradation, but edges of a single element

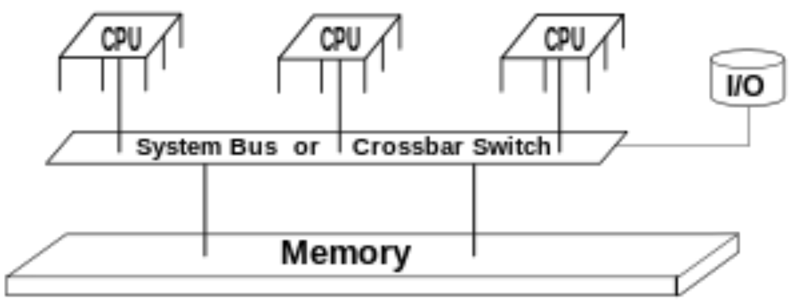
“Fixed” surfaces

Mesh – Metrics – Aside on parallelism

Shared Memory

Multiple processors shared memory (RAM/Hard disk etc)
Threading

“Painting a wall/Piano duet”



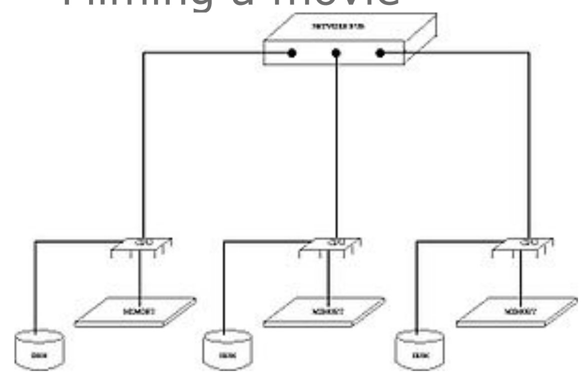
Communication fast, but hard to scale

Not currently implemented in IC-FERST but in use in single-phase Fluidity

Distributed Memory

Multiple systems each have their own RAM/hard disk and communicate over a LAN.

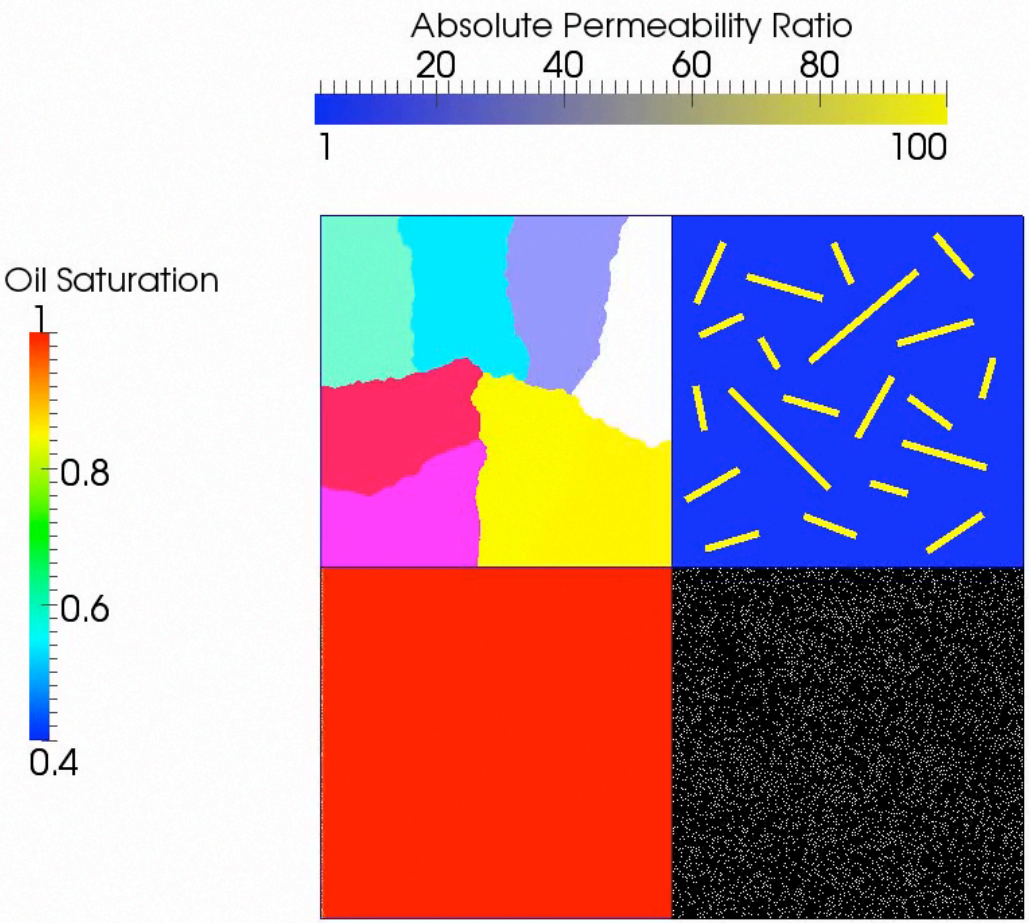
“Filming a movie



Scales well, but communication is slow.

Implimented and under testing in IC-FERST

Mesh Metrics - Parallel



IC FERST uses distributed memory paradigm for parallelism.

Each process acts independently and communicates information as needed.

For adaptivity this means locking boundary nodes to other processes, then adapting the rest of the domain as in serial.

Once new mesh is found, nodes are redistributed (hopefully shifting the ownership boundaries)

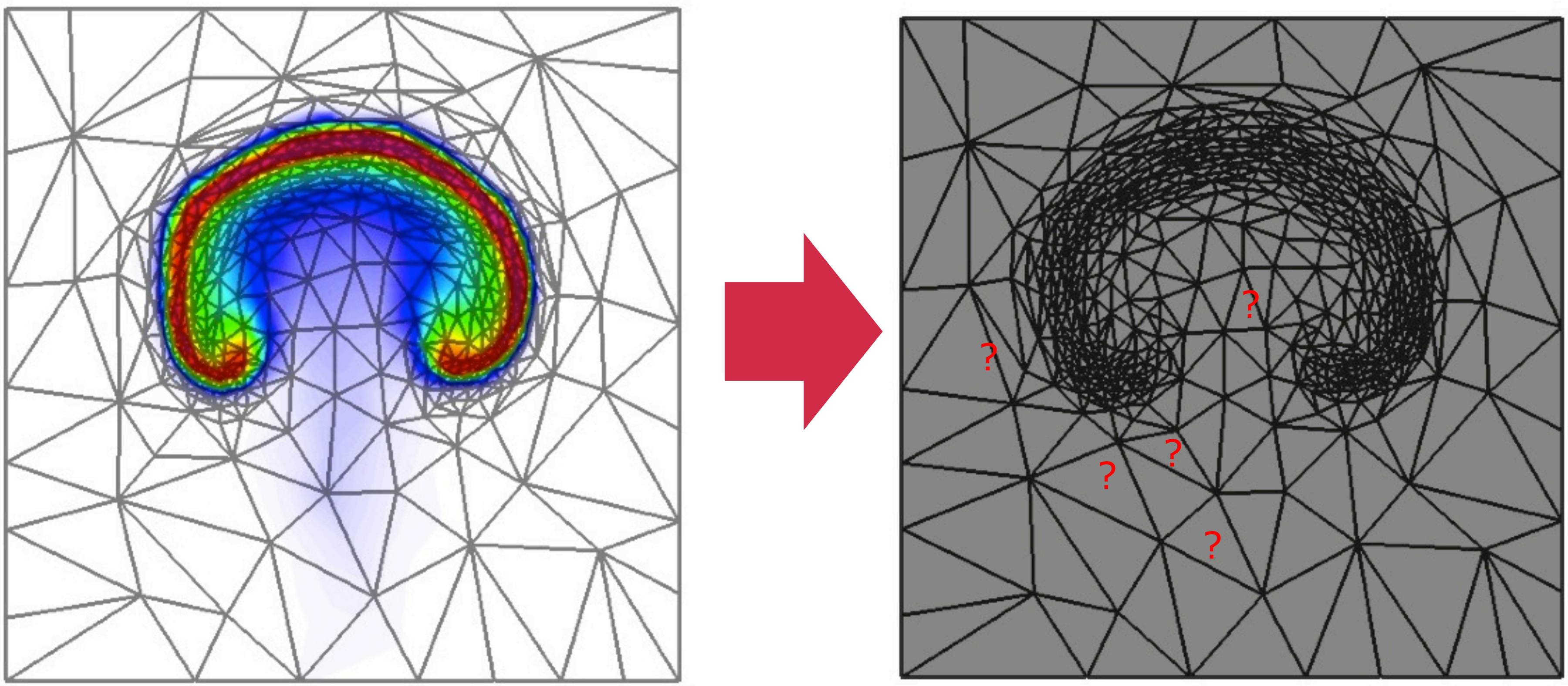
This loop is then repeated several times.

Imperial College London

Current Methodology – mesh to mesh interpolation

Data on old mesh

Data on new mesh?



Imperial College London

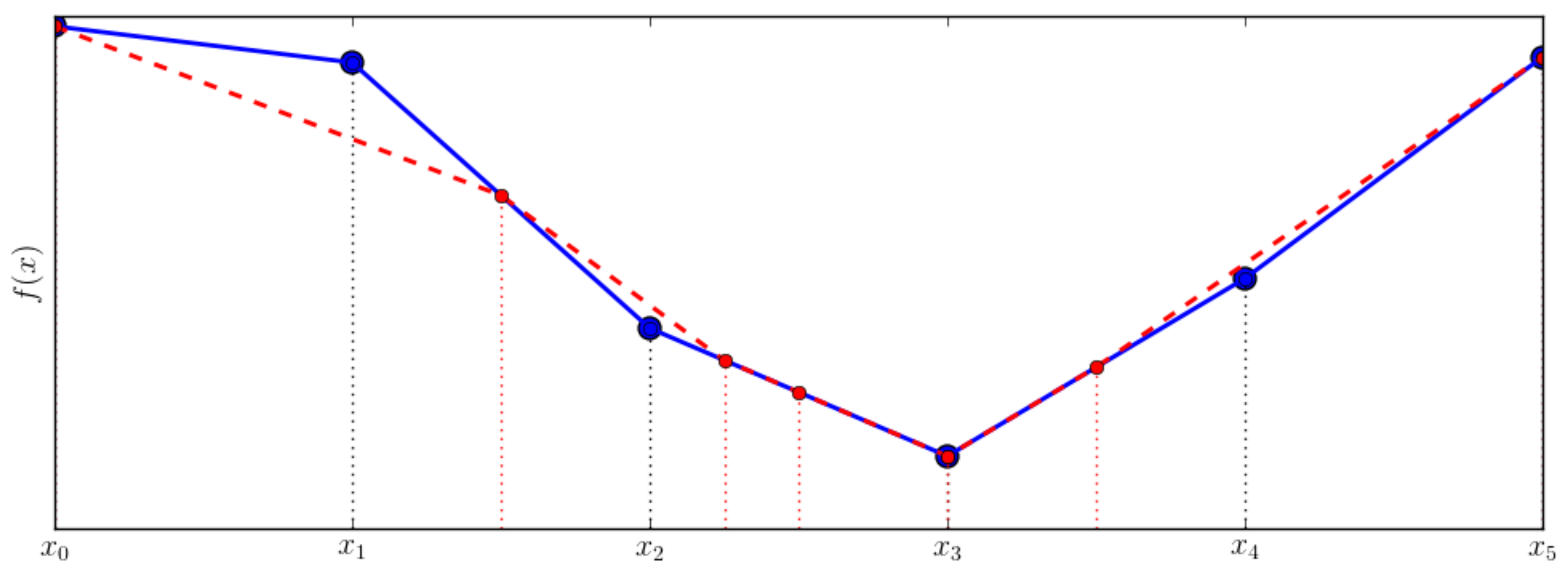
Interpolation – Method 1 "Consistent" Interpolation

Set values at new nodes to be spatial value on old finite element representation

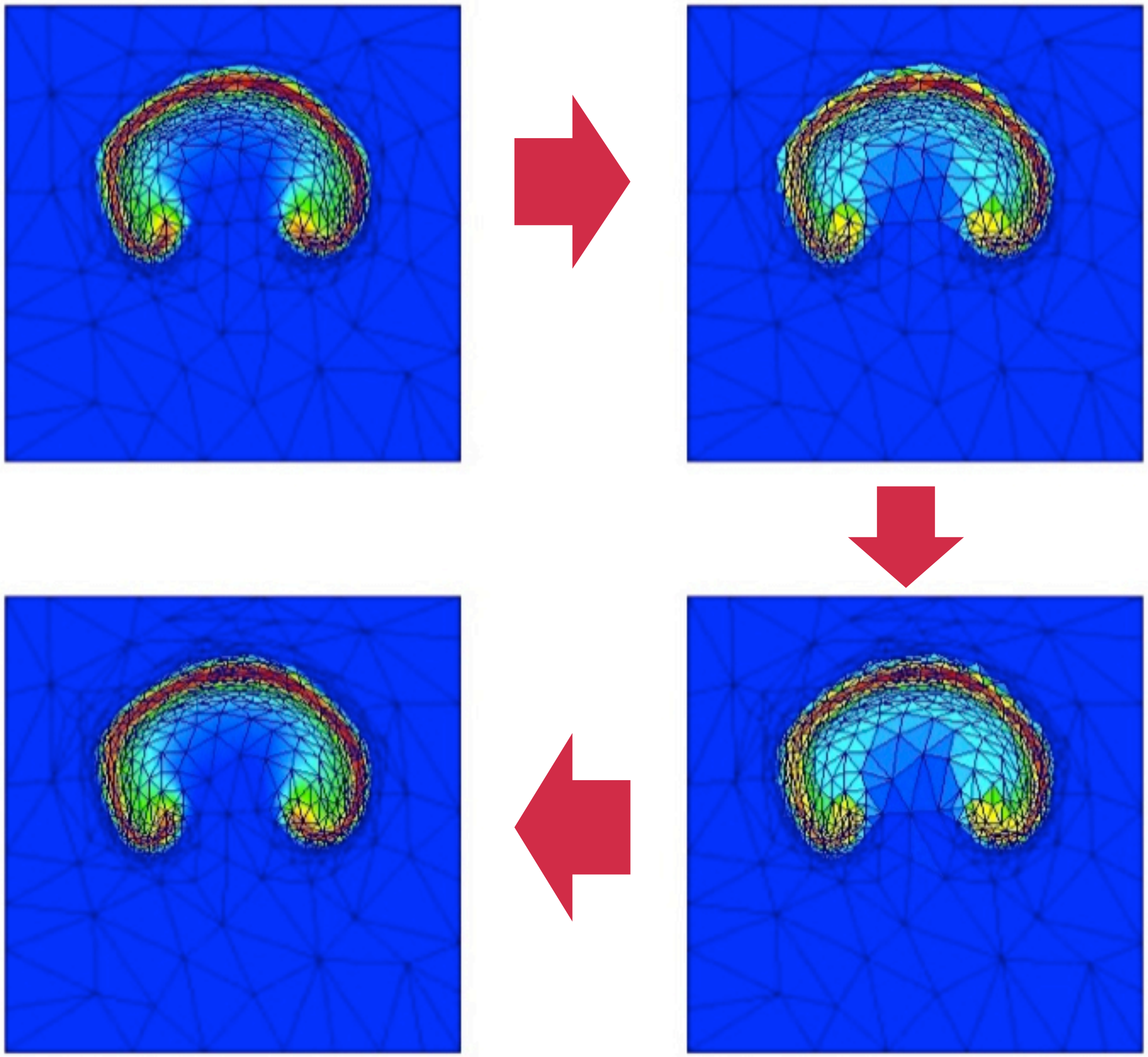
$$\psi_i^{\text{new}} = \psi^{\text{new}}(\boldsymbol{p}_i) = \sum_j N_j^{\text{old}}(\boldsymbol{p}_j)\,\psi_j^{\text{old}}$$

Pros:
Cheap
Bounded
Gives original data back if mesh doesn't change

Cons:
Nonconservative



Interpolation – Method 2 - Grandy Interpolation



1. Average mesh data onto old elements as flat functions.
2. Calculate average on new elements
3. Project back to original shapes.

Pros:
Cheap(ish)
Bounded
Local

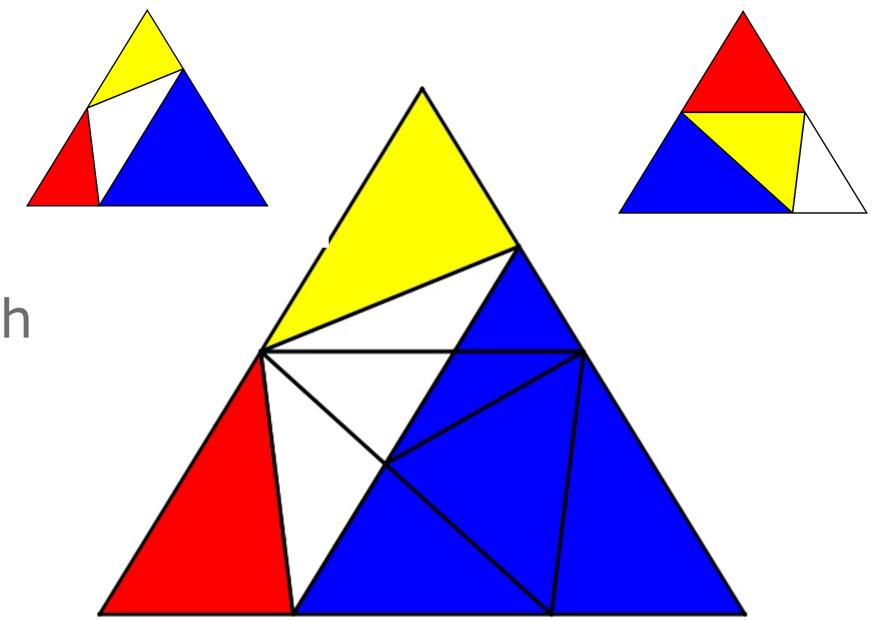
Cons:
Dissipative especially for discontinuous fields

Supermeshing

FE solutions/test functions piecewise smooth over mesh elements

Elements of supermesh: old variables and new test fns both smooth.
No jumps.

Allows efficient conservative mesh to mesh interpolation via projection methods



P. E. Farrell & J. R. Maddison (2011) Computer Methods in Applied Mechanics and Engineering

Imperial College London

Interpolation - Galerkin Projection

Galerkin Finite Element solution to $\psi^{\text{new}} = \psi^{\text{old}}$

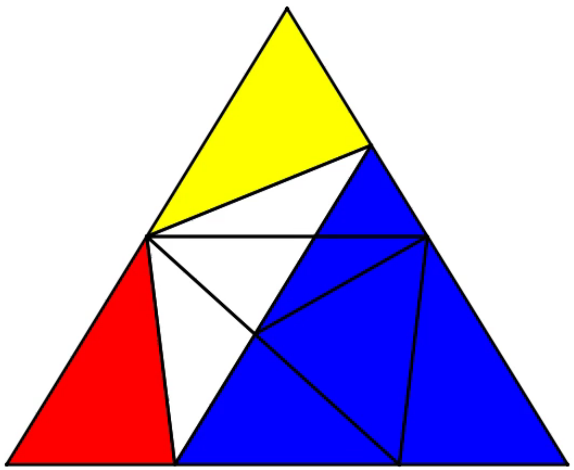
$$\sum_j \int_\Omega N_i^{\text{new}} N_j^{\text{new}} \psi_j^{\text{new}} dV = \sum_k \int_\Omega N_i^{\text{new}} N_k^{\text{old}} \psi_k^{\text{old}} dV$$

Left hand side is usual mass matrix

Right hand side is trickier
Supermeshing to the rescue!

Pros
Conservative
Local

Cons
No bounding
(more) expensive
Questions over control volumes



Imperial College London

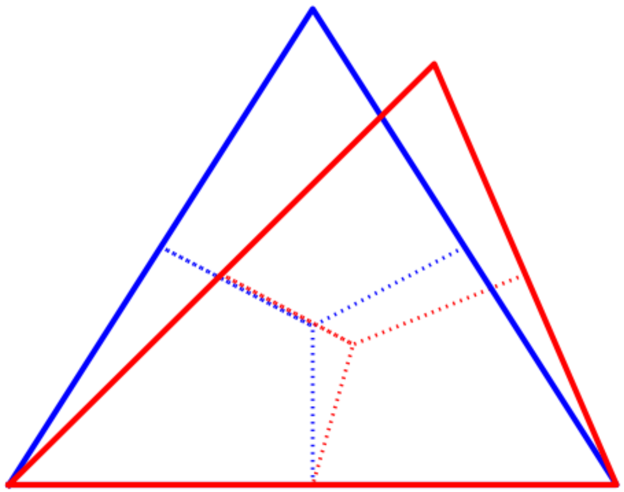
Discussion

Questions to the room:

- What variables are worthwhile to adapt to?
  - Saturation? Pressure?
- Do the length-scales inside the absolute permeability tensor matter?
- Interpolation methods for control volumes in the fully discontinuous formulation

The issue with control volumes

Supermeshing is <u>much</u> easier with triangles/tetrahedra than with control volumes

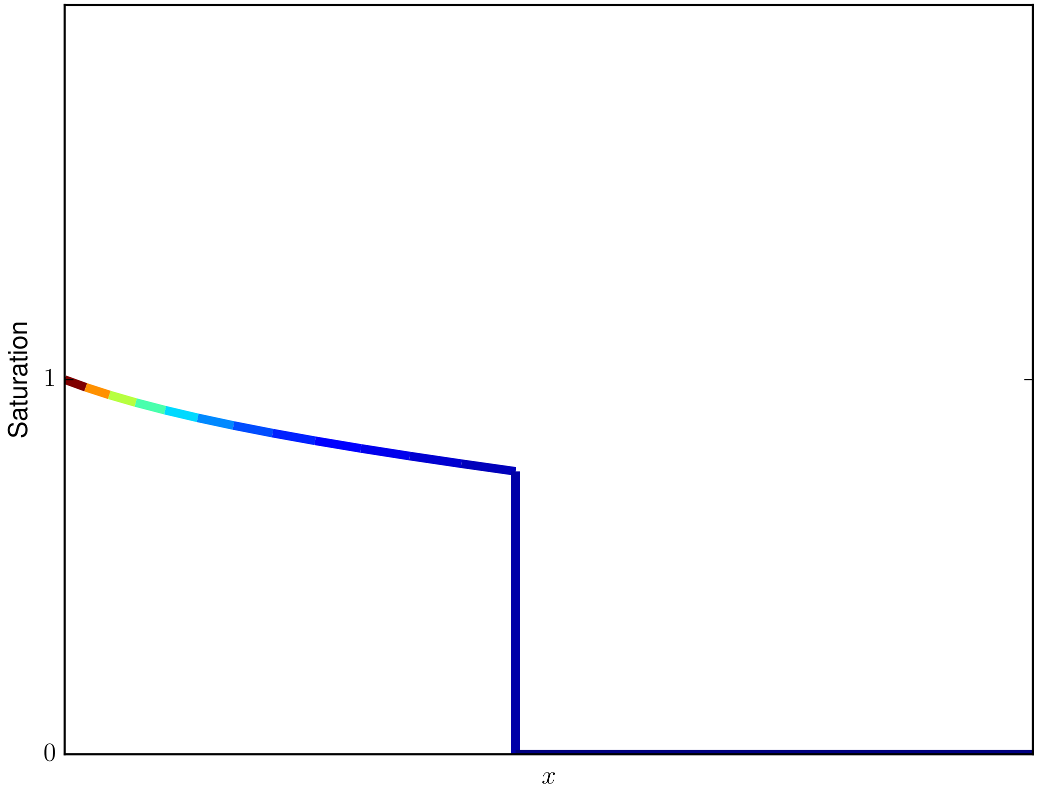


Moving one node creates 3 triangles of overlap for FE

Moving one node creates 7 quadrilaterals and one triangles of overlap for CV

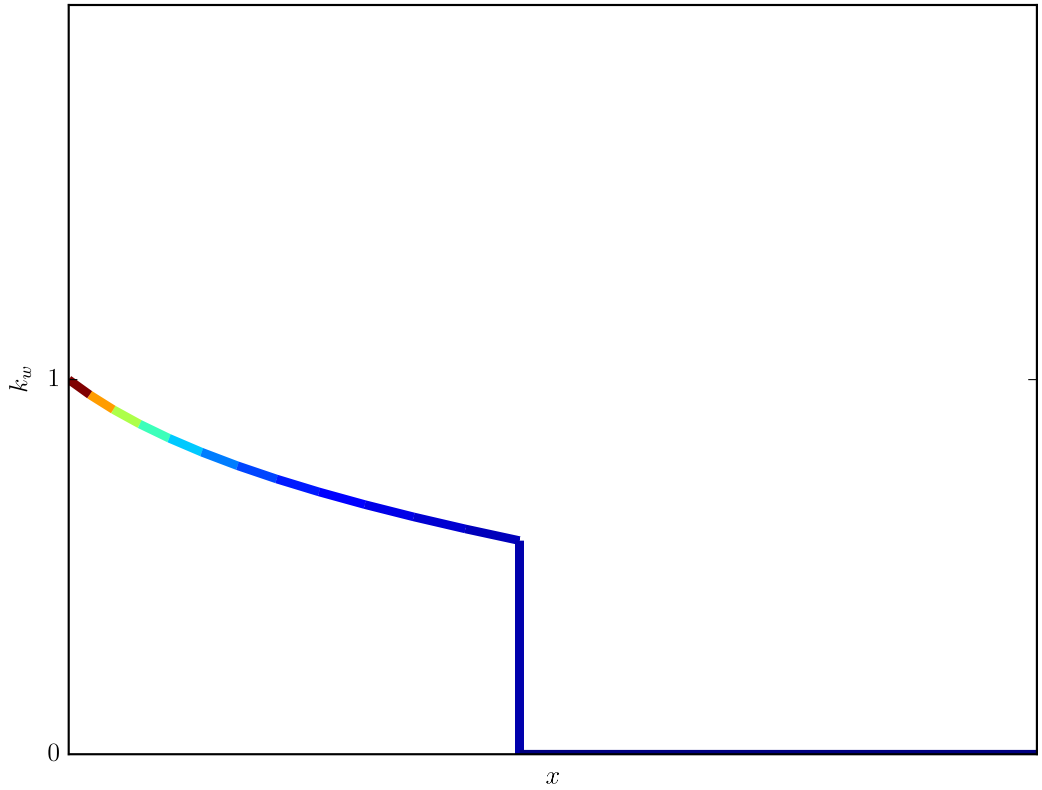
Discussion: Choice of Variables to adapt to

Buckley Leverett solution : Saturation



Discussion: Choice of Variables to adapt to

Buckley Leverett solution : Relative Permeability



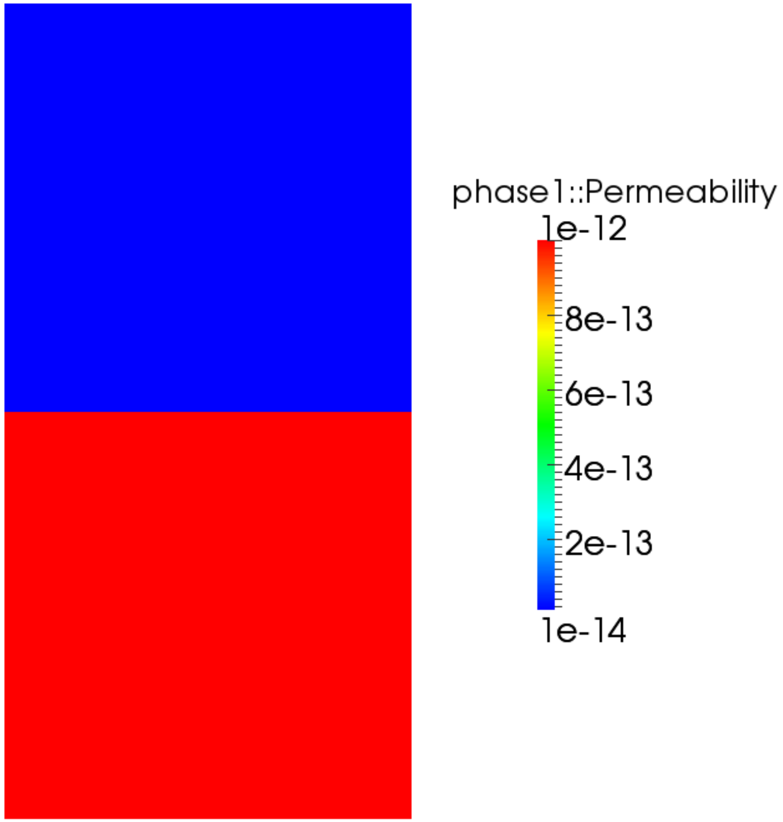
Imperial College London

Discussion: Choice of Variables to adapt to

Velocity or pressure?

Single phase Darcy flow through two porous media

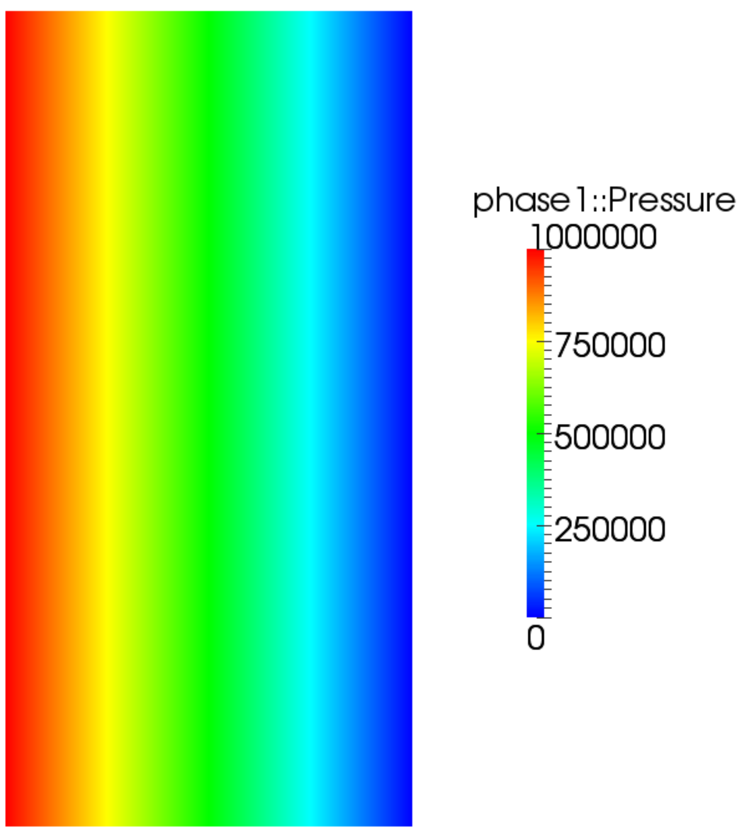
Prescribed pressure difference



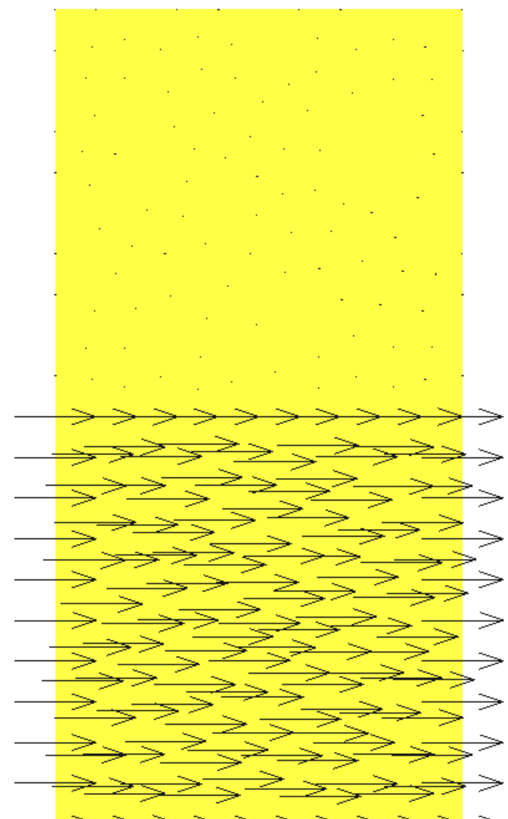
Discussion: Choice of Variables to adapt to

Velocity or pressure?

Pressure



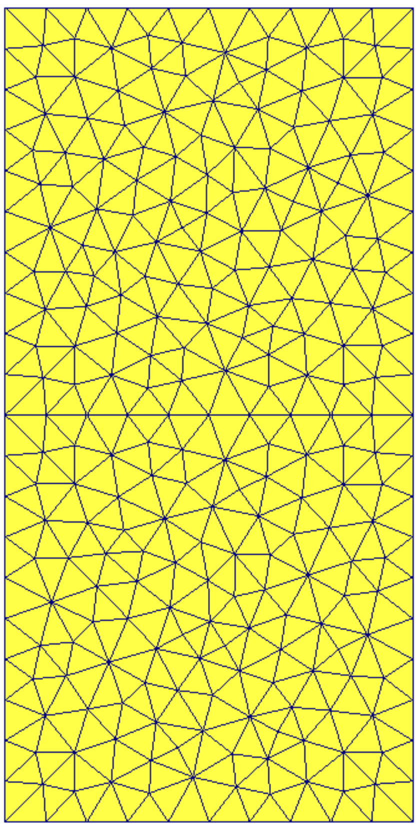
Velocity



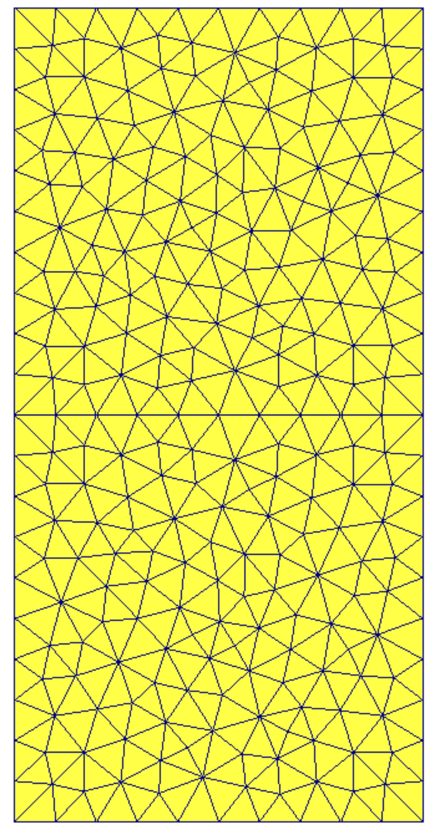
Discussion: Choice of Variables to adapt to

Velocity or pressure?

Pressure



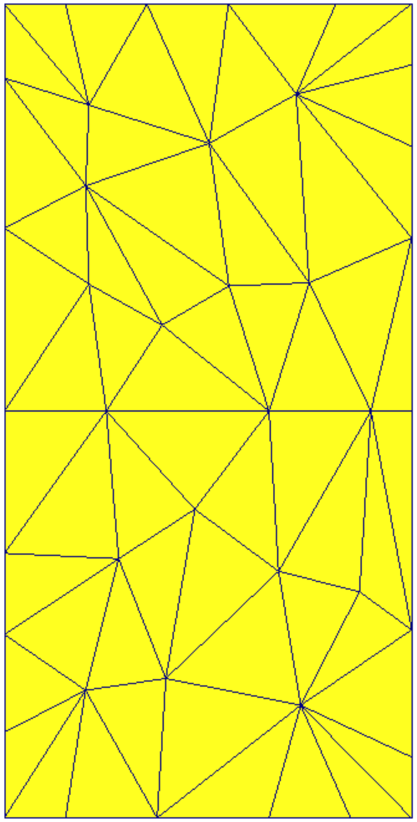
Velocity



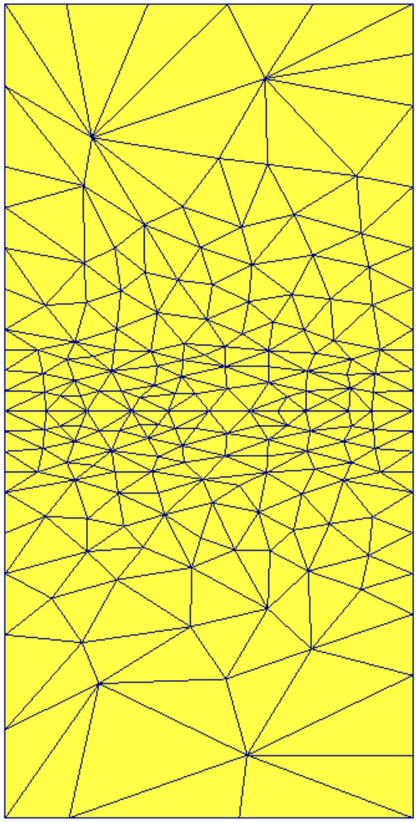
Discussion: Choice of Variables to adapt to

Velocity or pressure?

Pressure



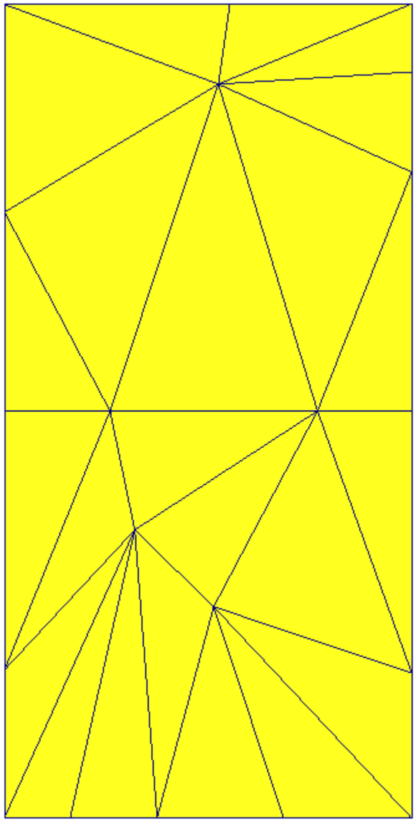
Velocity



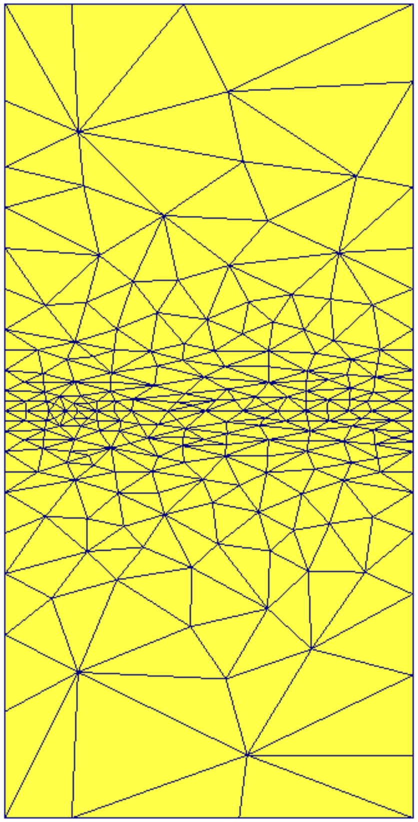
Discussion: Choice of Variables to adapt to

Velocity or pressure?

Pressure



Velocity



Imperial College London

AMCG

Any further points?

Thank you very much for your input

j.percival@imperial.ac.uk
Room 4.85 RSM

