

Performance evaluation of existing and new VoF simulation techniques: solving, interface treatment, and dynamic meshes.

**9th OpenFOAM Workshop
23 June 2014 – Zagreb, Croatia**

Kyle Mooney^{*1} , Jacques Papper² and Tyler Voskuilen³

1 - *Icon Technology & Process Consulting LLC. 4555 Lake Forest Drive, Suite 650 Cincinnati, OH USA, k.mooney@iconcfd.com*

2 - *Icon Technology & Process Consulting Ltd., 7 Rue Auber 31000 Toulouse France, j.papper@iconcfd.com*

3 - *Maurice J. Zucrow Laboratory, Purdue University, West Lafayette, IN 47907 USA, tvoskuil@purdue.edu*



Contact Us

ICON HQ, Berkshire House, Windsor SL41QN, Windsor, UK
P. +44 (0)1753 751400 / contact@iconCFD.com

© 2013 ICON Technology & Process Consulting Ltd.

INTRODUCTION

AGENDA

- Investigate stability limits of explicit and semi-implicit MULES solvers.
- Evaluate curvature estimates of different discretization schemes
- Automatic mesh refinement (AMR)
 - Background
 - Distributed processor load balancing
 - Demo cases
- Next steps

STABILITY AND BOUNDEDNESS

STABILITY IMPLICIT VS EXPLICIT

- Investigate stability limits of explicit and semi-implicit MULES
- Compare run times and changes in solution vs. time step size.
- Compare alpha field boundedness

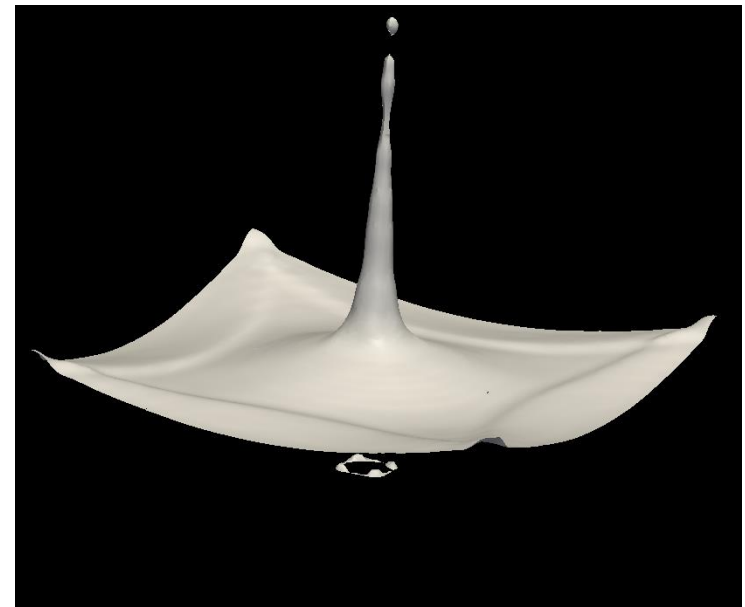
Case:

100x100x100 hex mesh

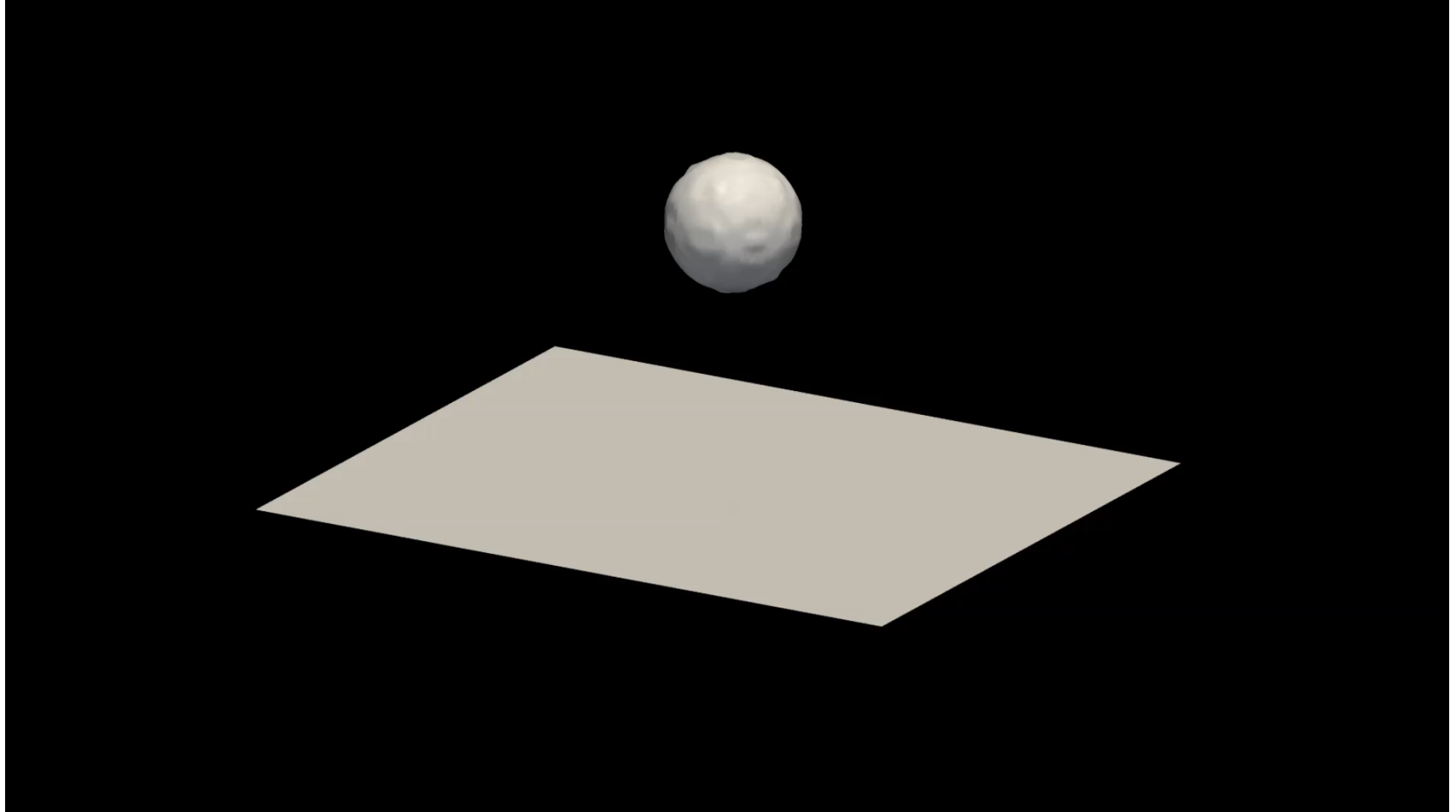
Material: air & water

20 cells / drop diameter

gravity driven



STABILITY IMPLICIT VS EXPLICIT



STABILITY IMPLICIT VS EXPLICIT

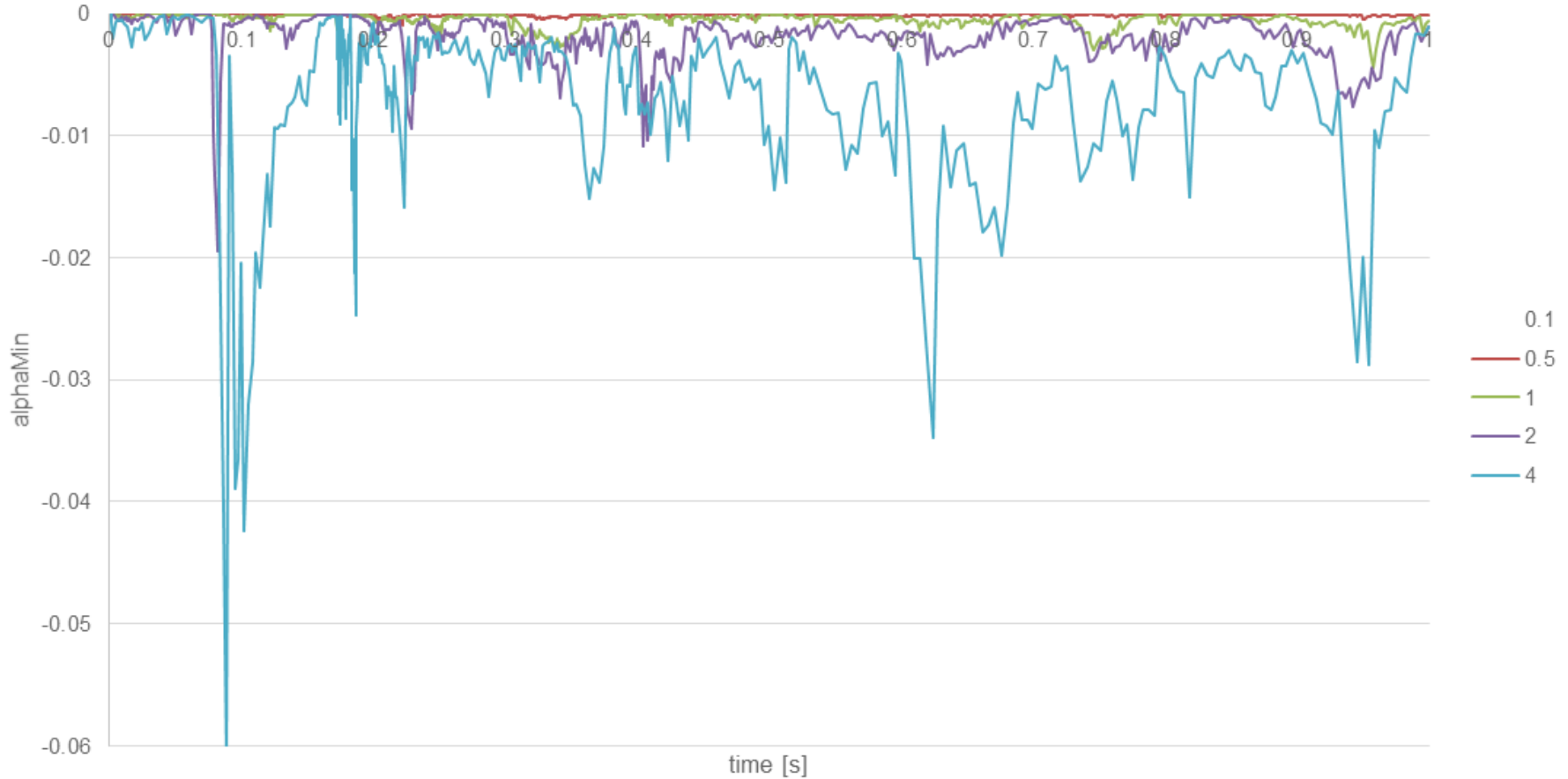
	Implicit	Explicit
maxCo = maxAlphaCo	runTime [s]	
0.1	20579	17614
0.5	3627	3474
1	1995	Unstable
2	1175	Unstable
4	791	Unstable

STABILITY IMPLICIT RUNTIME

	Implicit
maxCo / maxAlphaCo	runTime [-]
0.1	26.01
0.5	4.59
1	2.52
2	1.49
4	1

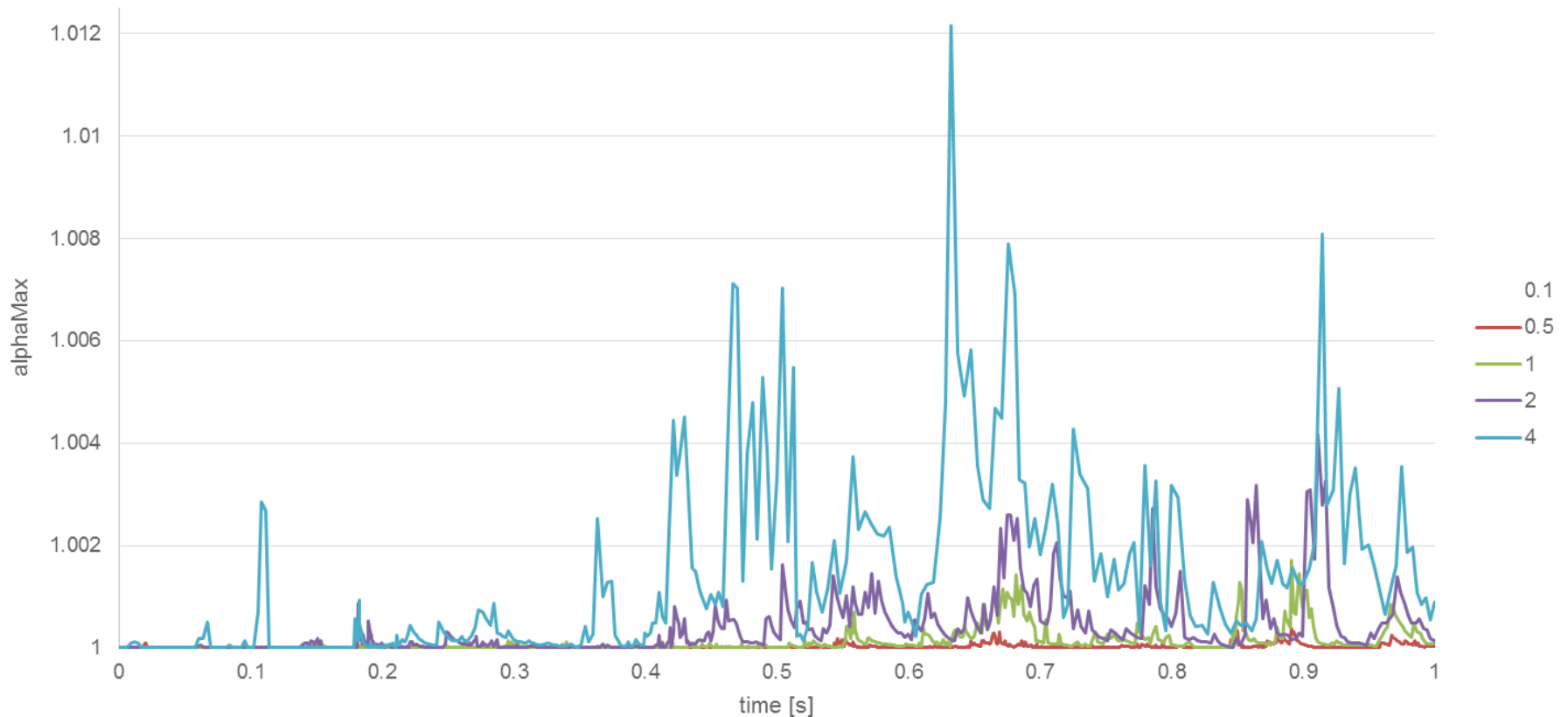
BOUNDEDNESS IMPLICIT-MULES

alphaMin Boundedness vs time by maxCo



BOUNDEDNESS IMPLICIT MULES

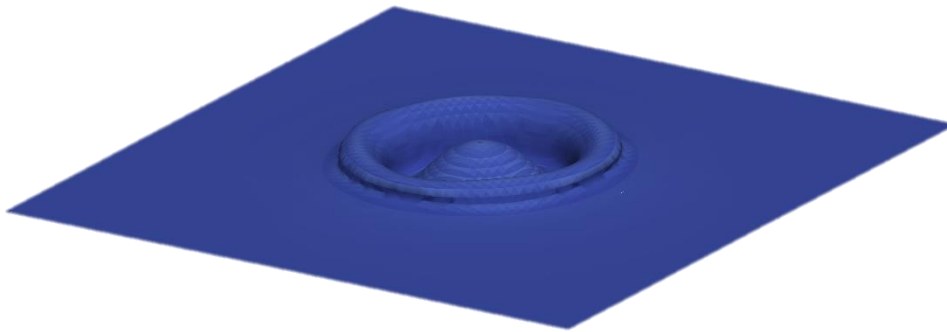
alphaMax Boundedness vs time by maxCo



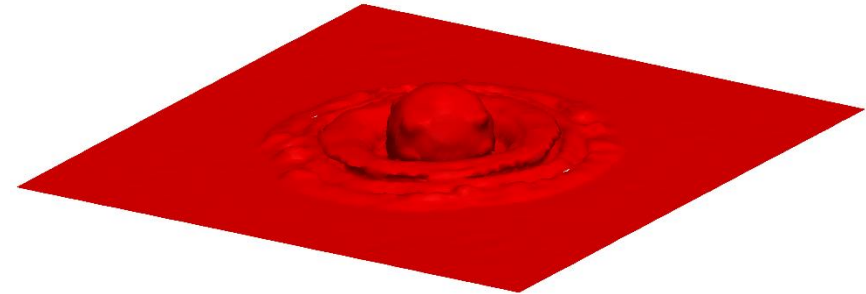
SOLUTION BY MAX COURANT NO.

www.iconCFD.com

$t=0.1$ [s]



maxCo=0.5

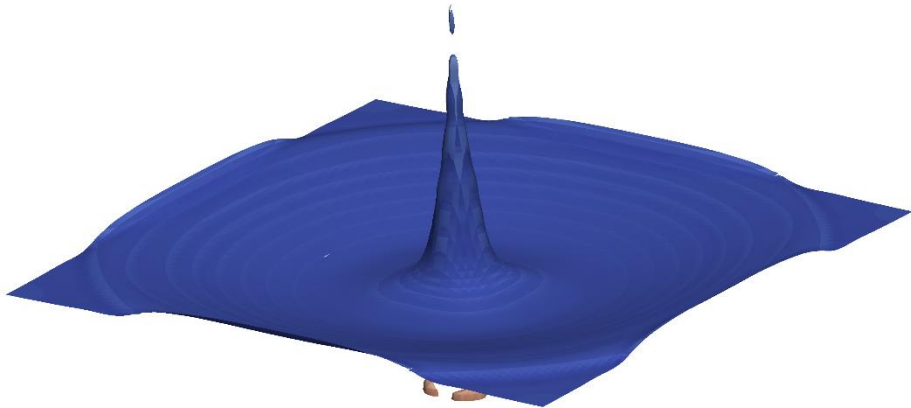


maxCo=4

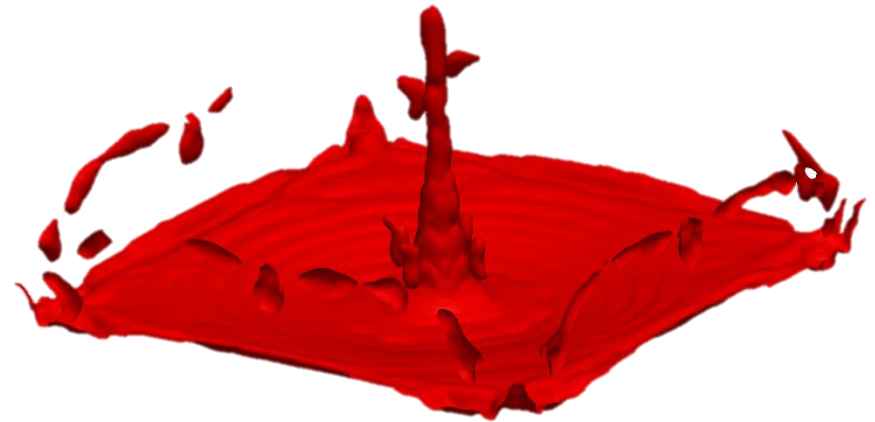
SOLUTION BY MAX COURANT NO.

www.iconCFD.com

$t=0.2$ [s]



maxCo=0.5

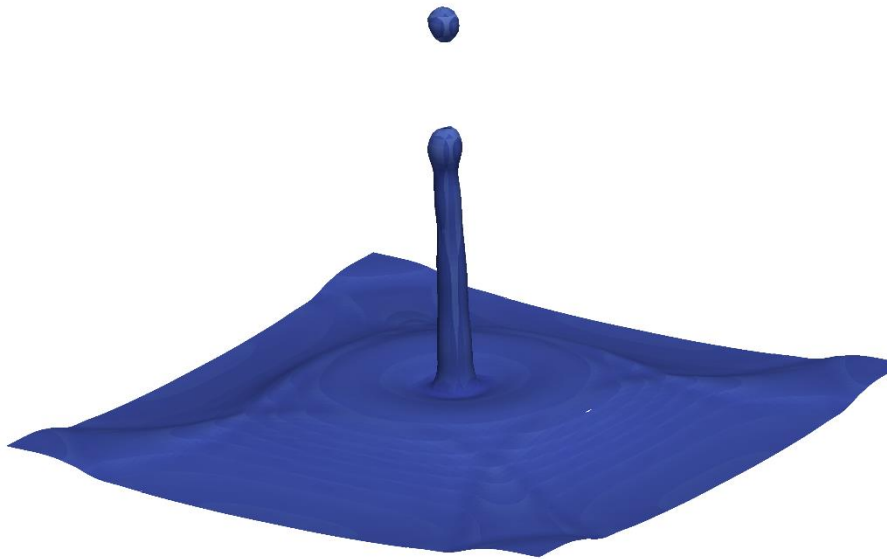


maxCo=4

SOLUTION BY MAX COURANT NO.

www.iconCFD.com

$t=0.3$ [s]



maxCo=0.5

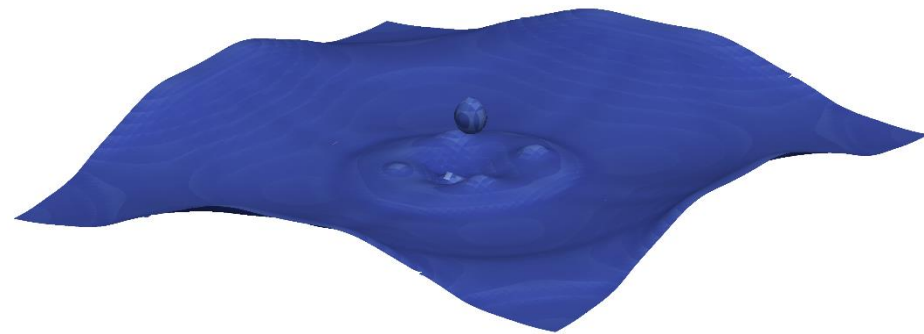


maxCo=4

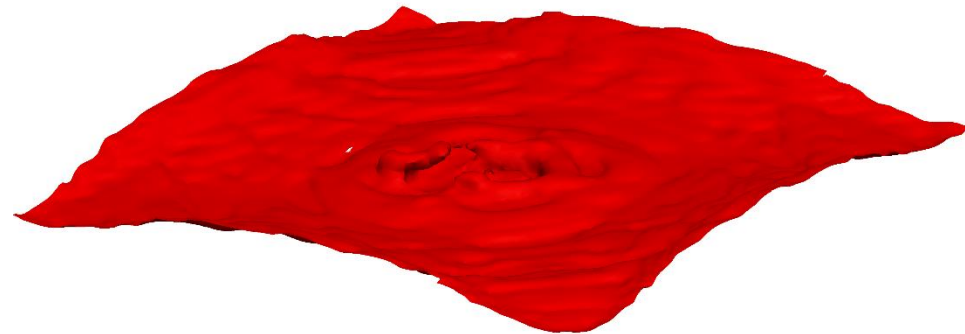
SOLUTION BY MAX COURANT NO.

www.iconCFD.com

$t=0.4$ [s]



maxCo=0.5



maxCo=4

SOLUTION BY MAX COURANT NO.

www.iconCFD.com

$t=0.5$ [s]



maxCo=0.5

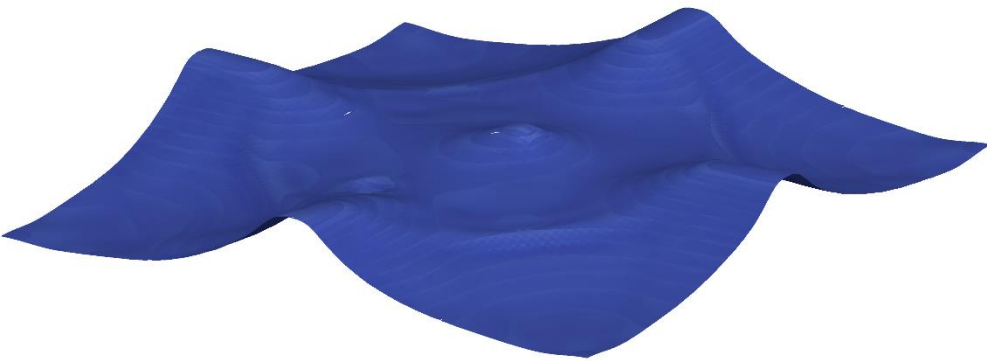


maxCo=4

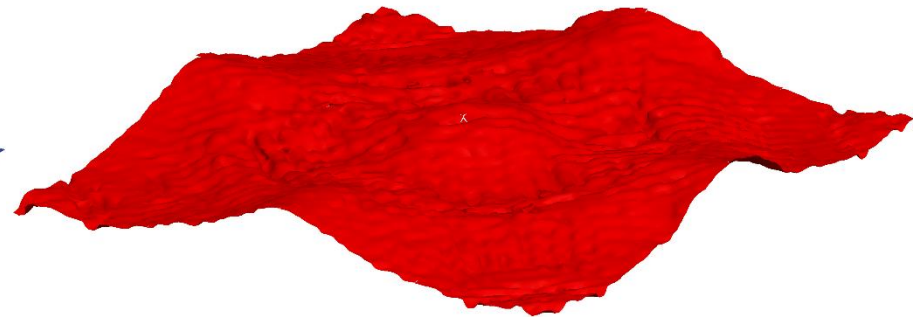
SOLUTION BY MAX COURANT NO.

www.iconCFD.com

t=0.6 [s]



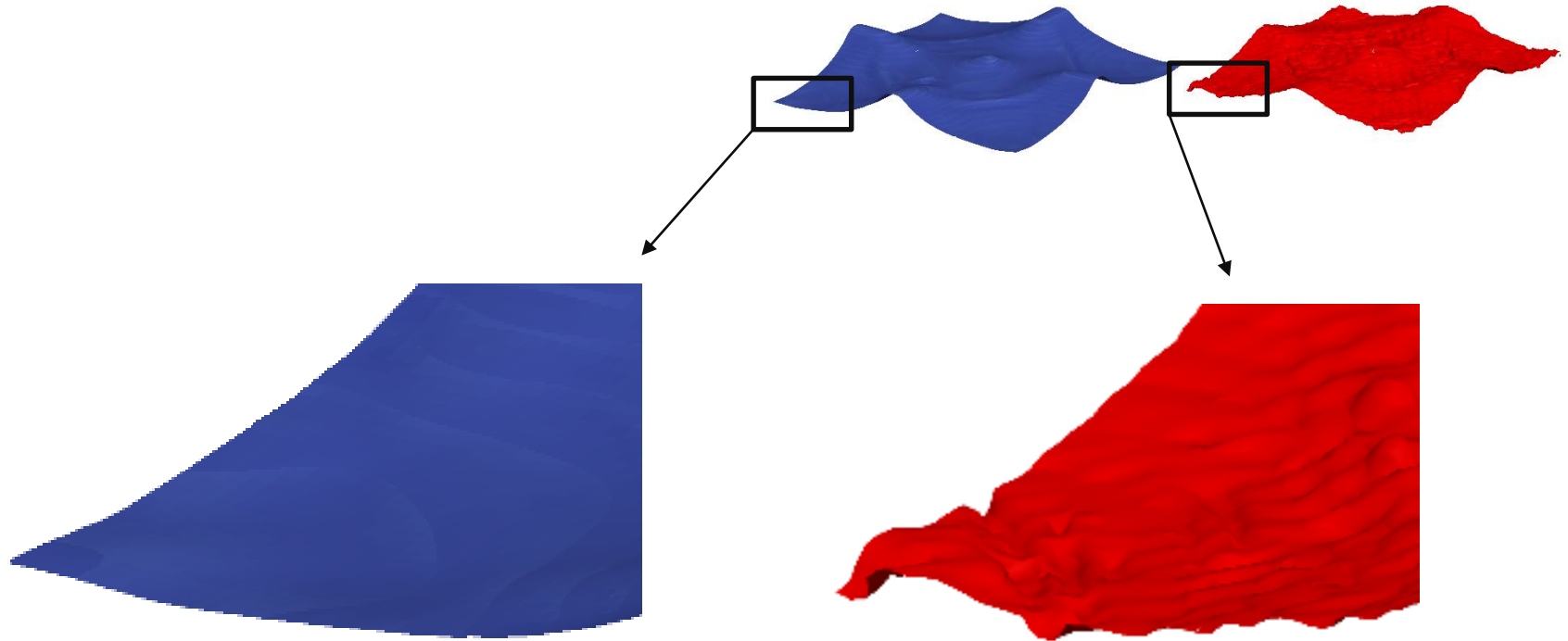
maxCo=0.5



maxCo=4

SOLUTION BY MAX COURANT NO.

t=0.6 [s]



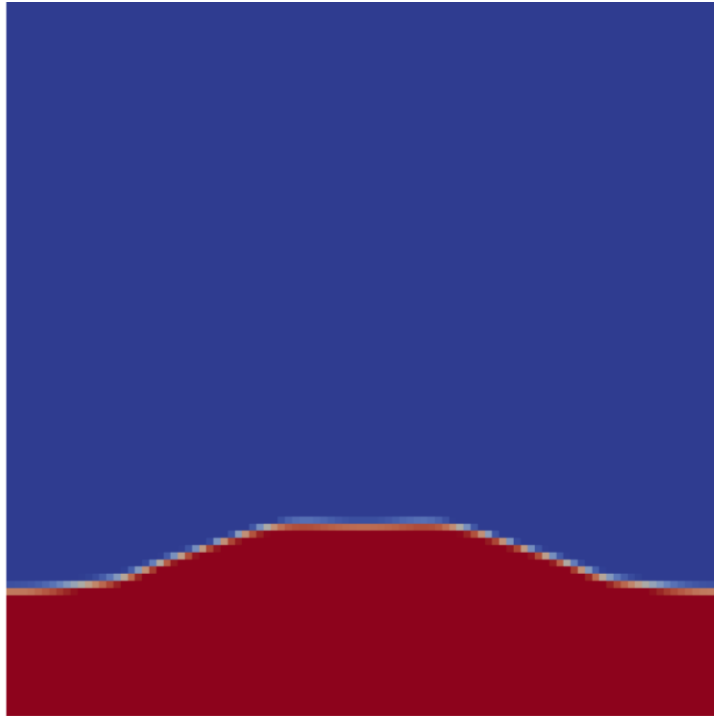
maxCo=0.5

maxCo=4

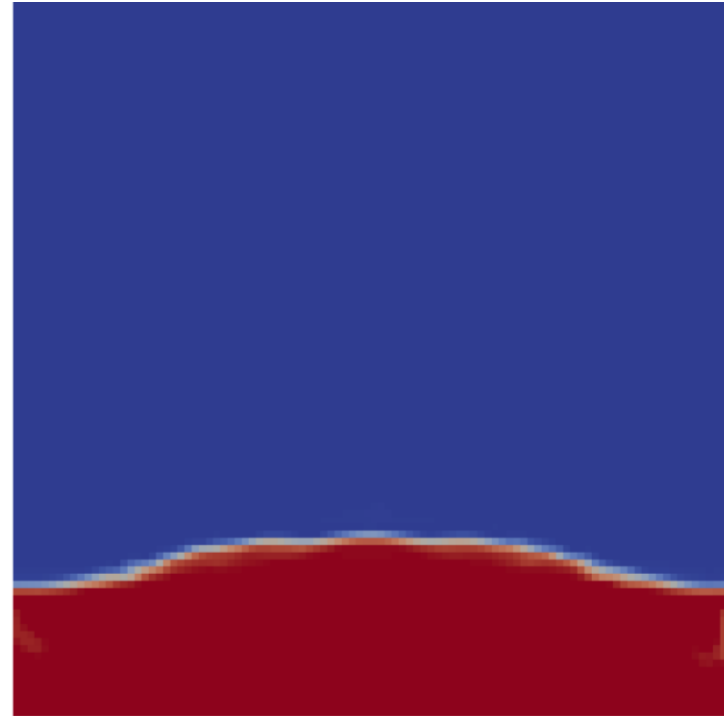
SOLUTION BY MAX COURANT NO.

t=0.2 [s]

Interface diffusion



maxCo=0.5



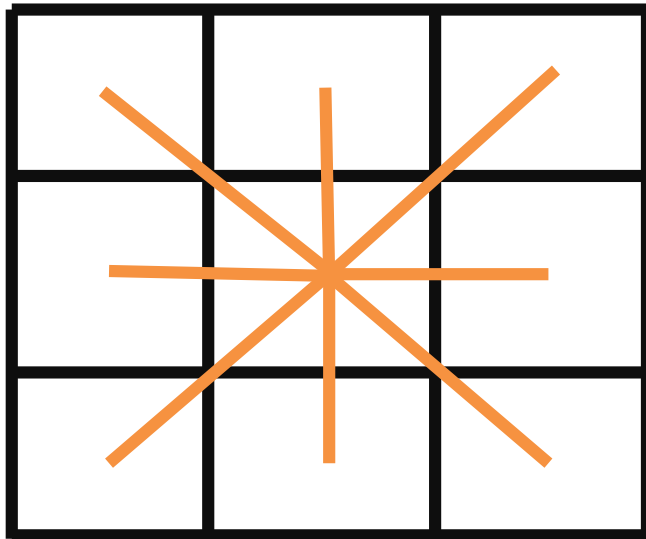
maxCo=4

CURVATURE ESTIMATION

INTERPOLATION CURVATURE ACCURACY

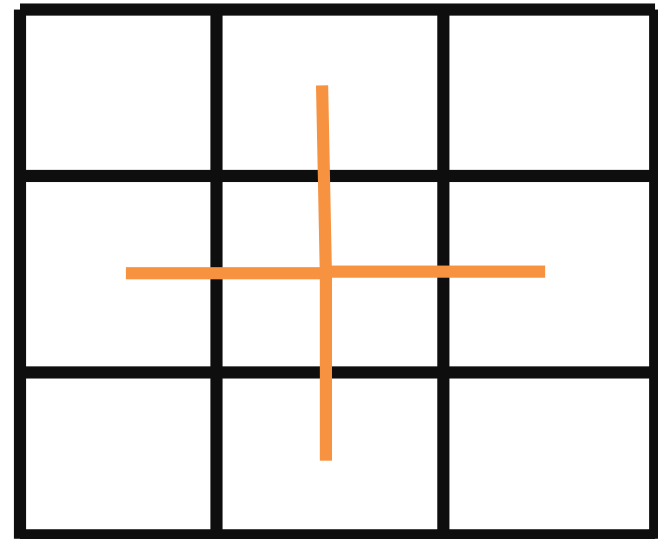
pointCellsLeastSquares

Constructs larger interpolation
stencil by including point
connected cell neighbours



$$\hat{n} = \frac{\nabla \gamma}{|\nabla \gamma|}$$

Gauss linear



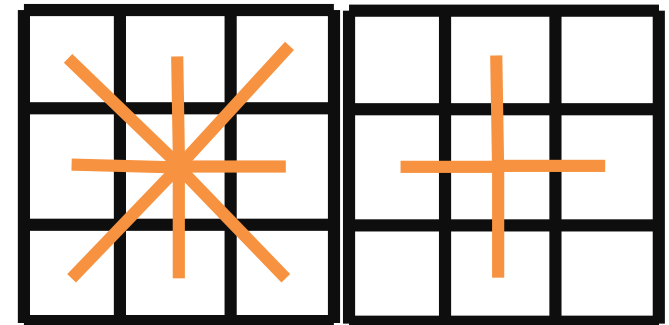
INTERPOLATION CURVATURE ACCURACY

Test procedure:

1. Initialize droplet
2. Wait until droplet stabilizes
3. Write cell-centered K field
4. Create $\alpha=0.5$ isoSurface
5. Interpolate K onto surface
6. Compute curvature error

Case:

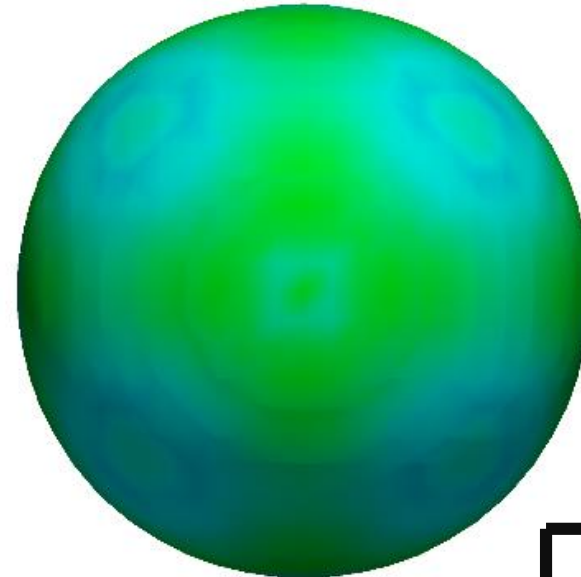
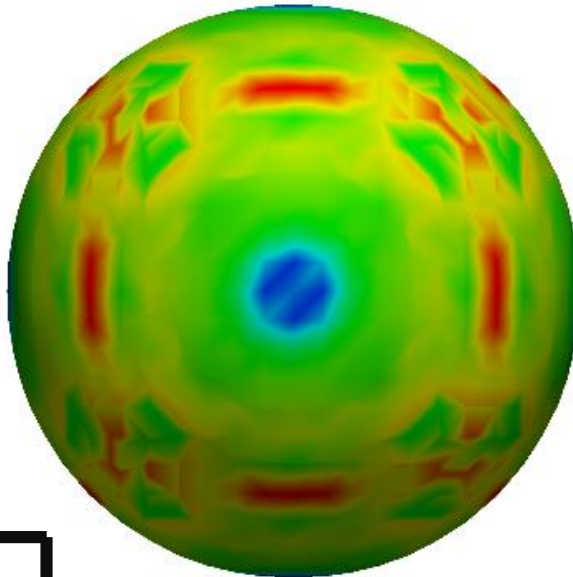
- 100x100x100 uniform block mesh
- 20 cells / drop diameter
- Initial spherical $\alpha=1$ field



INTERPOLATION CURVATURE % ERROR

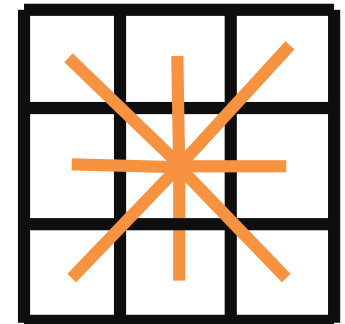
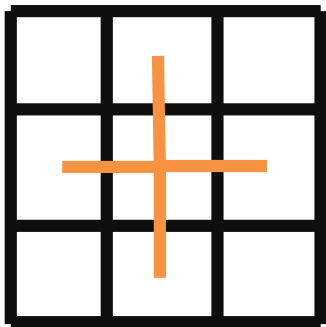
Gauss linear

pointCellsLeastSquares



Run time:
2036

Run time:
2261



K % Err
10



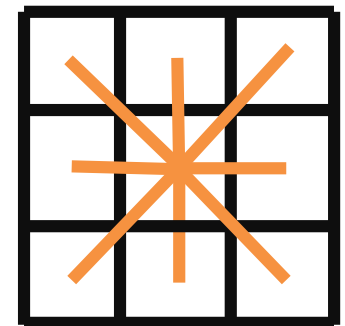
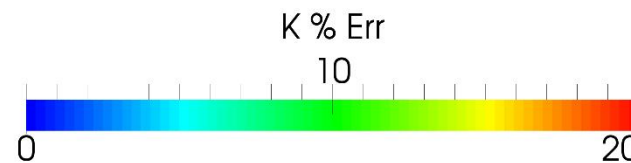
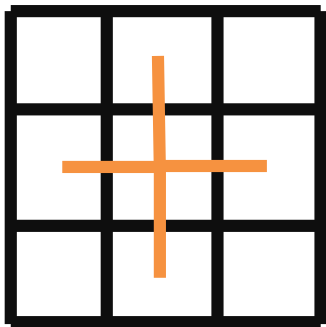
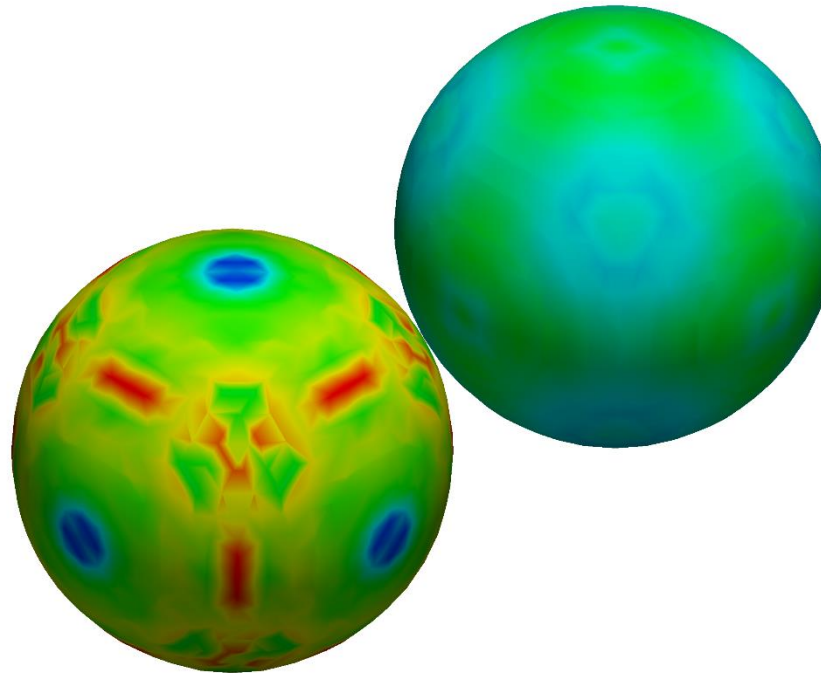
INTERPOLATION CURVATURE % ERROR

Gauss linear

Normalized
run time: 1

pointCellsLeastSquares

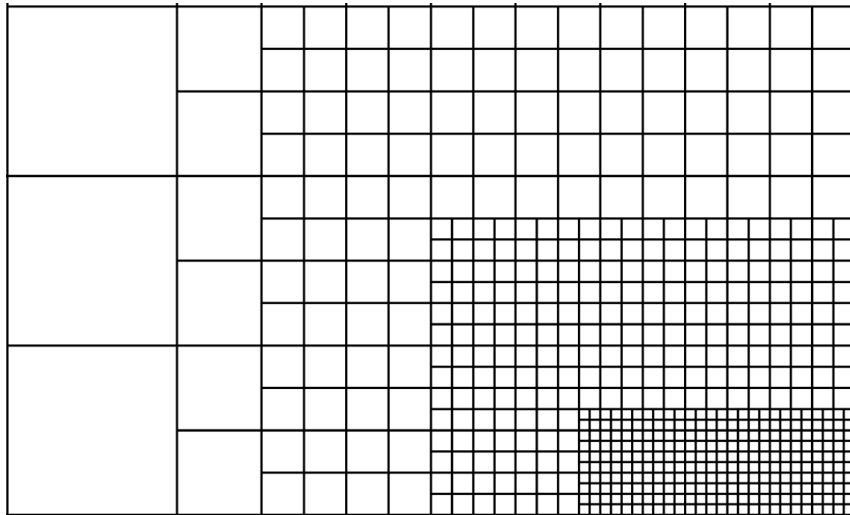
Normalized
run time: 1.11



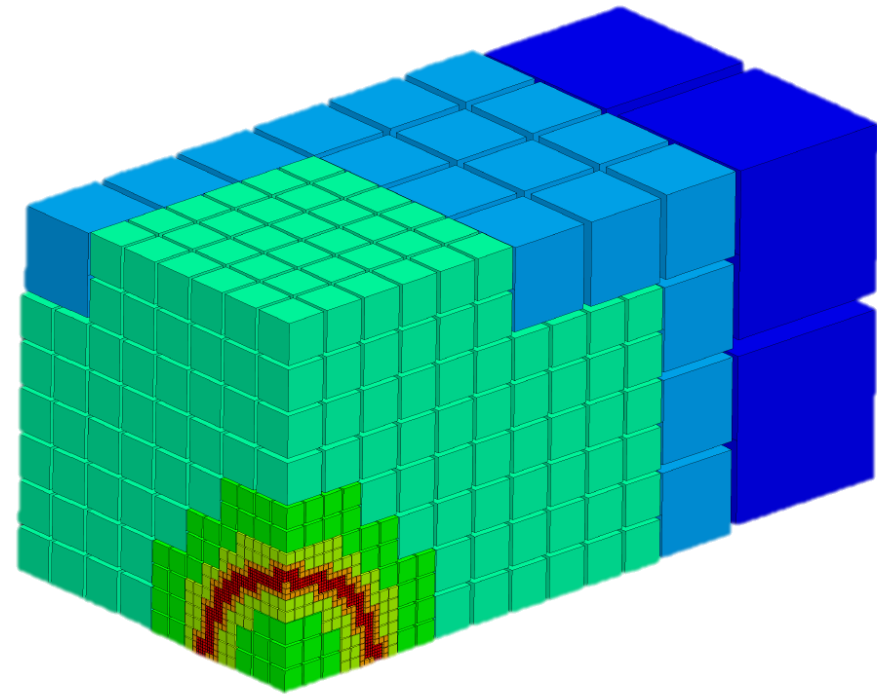
AUTOMATIC REFINEMENT AND PARALLEL LOAD BALANCING

DYNAMIC REFINEMENT

QUAD – HEX BASED



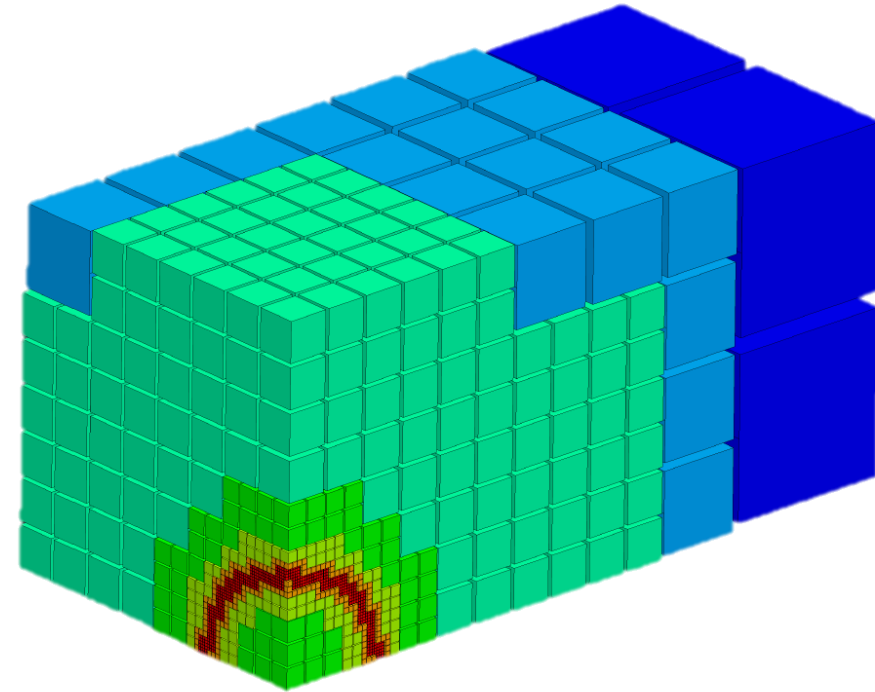
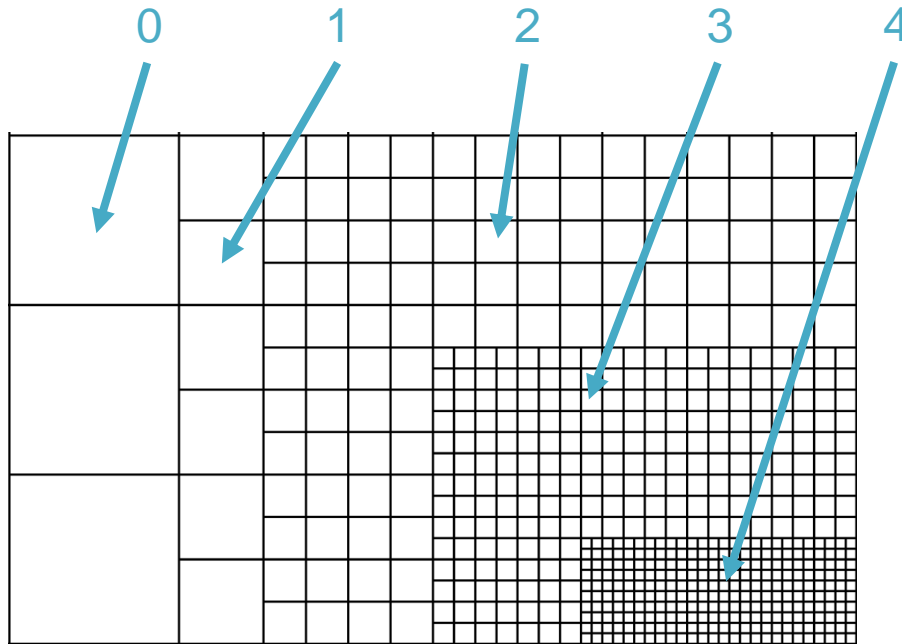
Quadtree – 2D



Octree – 3D

DYNAMIC REFINEMENT QUAD – HEX BASED

Refinement level n :



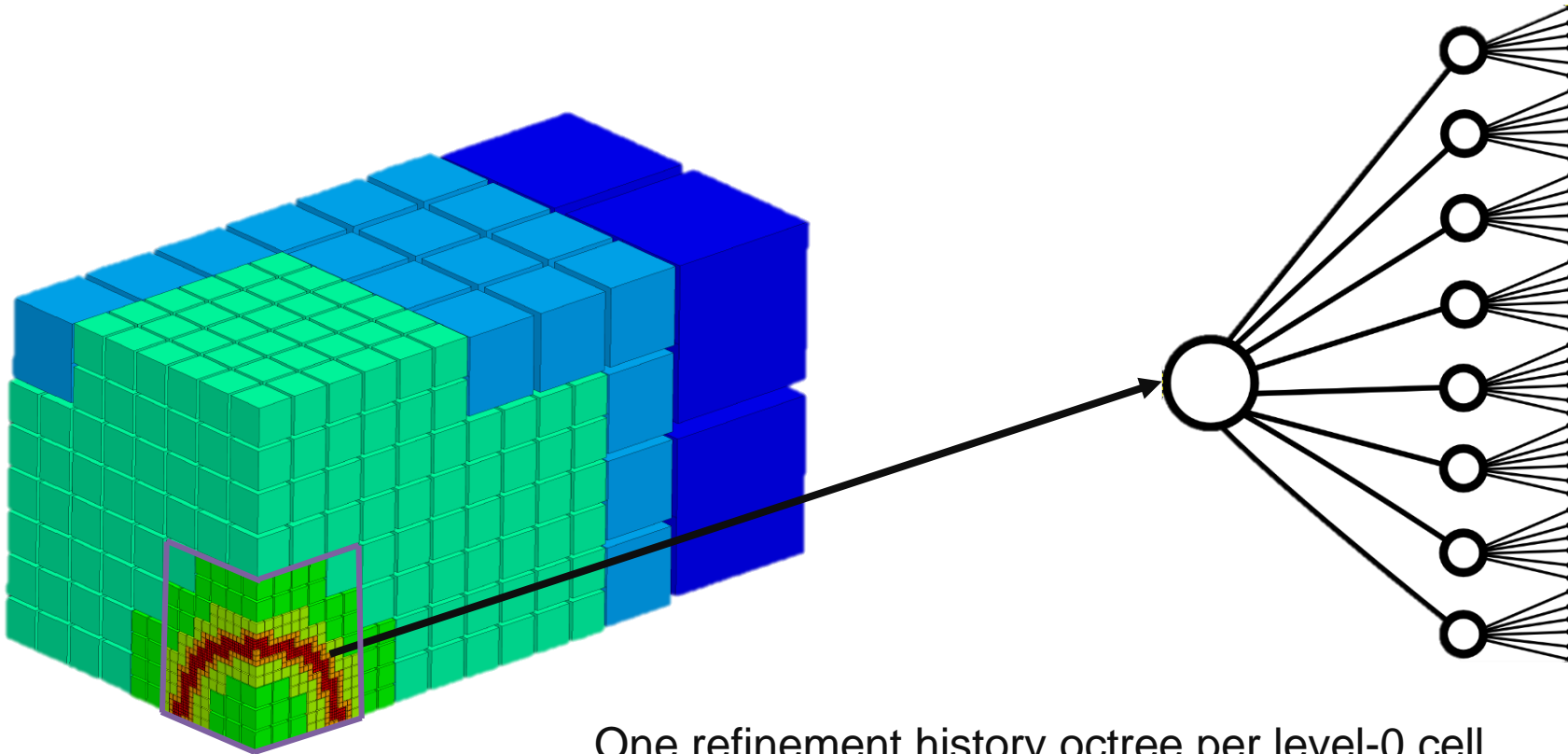
Quadtree – 2D

$$\Delta x = \frac{\text{baseSize}}{2^n}$$

Octree – 3D

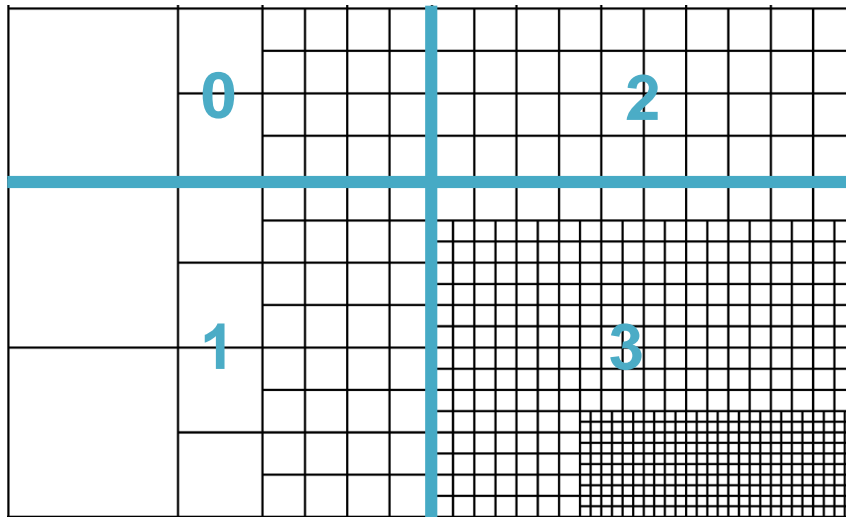
DYNAMIC REFINEMENT

OCTREE REFINEMENT HISTORY

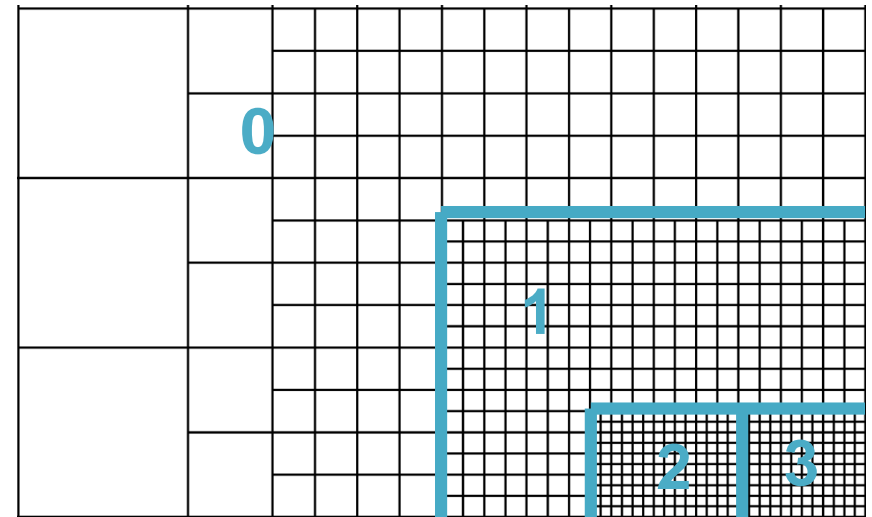


One refinement history octree per level-0 cell
Contains data required for de-refinement
Cannot split octree during load balancing

DYNAMIC REFINEMENT PTSCOTCH - GRAPH PARTITIONER



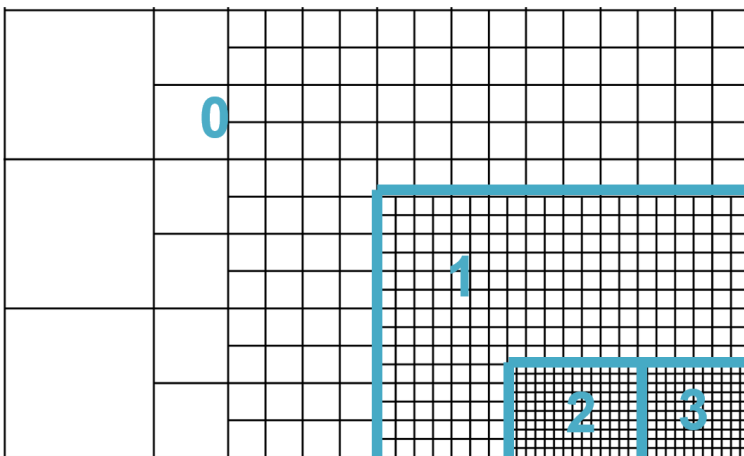
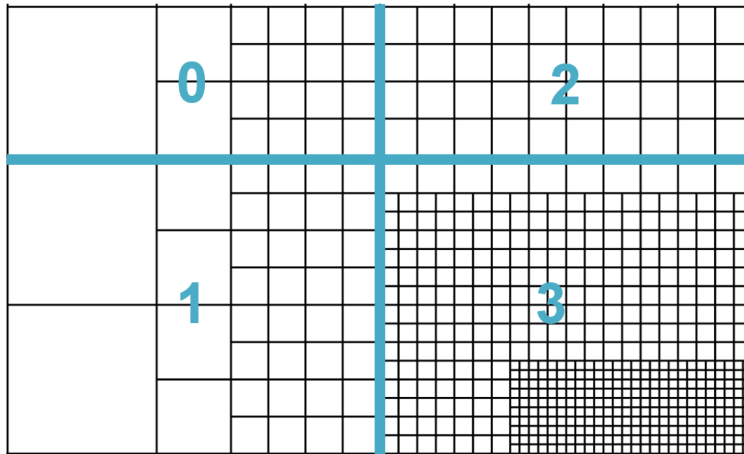
Poor processor balance



Better processor balance

LOAD BALANCING

PTSCOTCH - GRAPH PARTITIONER



PT-Scotch
+
Constrain to keep octrees

DYNAMIC REFINEMENT

CONFIG: GRADIENT / CURL / REGION DRIVEN

```
refinementControls
{
    enableRefinementControl true;

    gradients
    (
        | alpha.water (1000.0 10)
    );

    curls
    (
        | U (0.5 1)
    );

    regions
    (
        | boxToCell
        | {
        | minLevel 1;
        | box (-1 -1 -1) (1 1 1);
        | }
    );
}

dynamicRefineFvMeshCoeffs
{
    enableBalancing true;
    allowableImbalance 0.15;

    // How often to refine
    refineInterval 2;

    // Field to be refinement on
    field internalRefinementField;

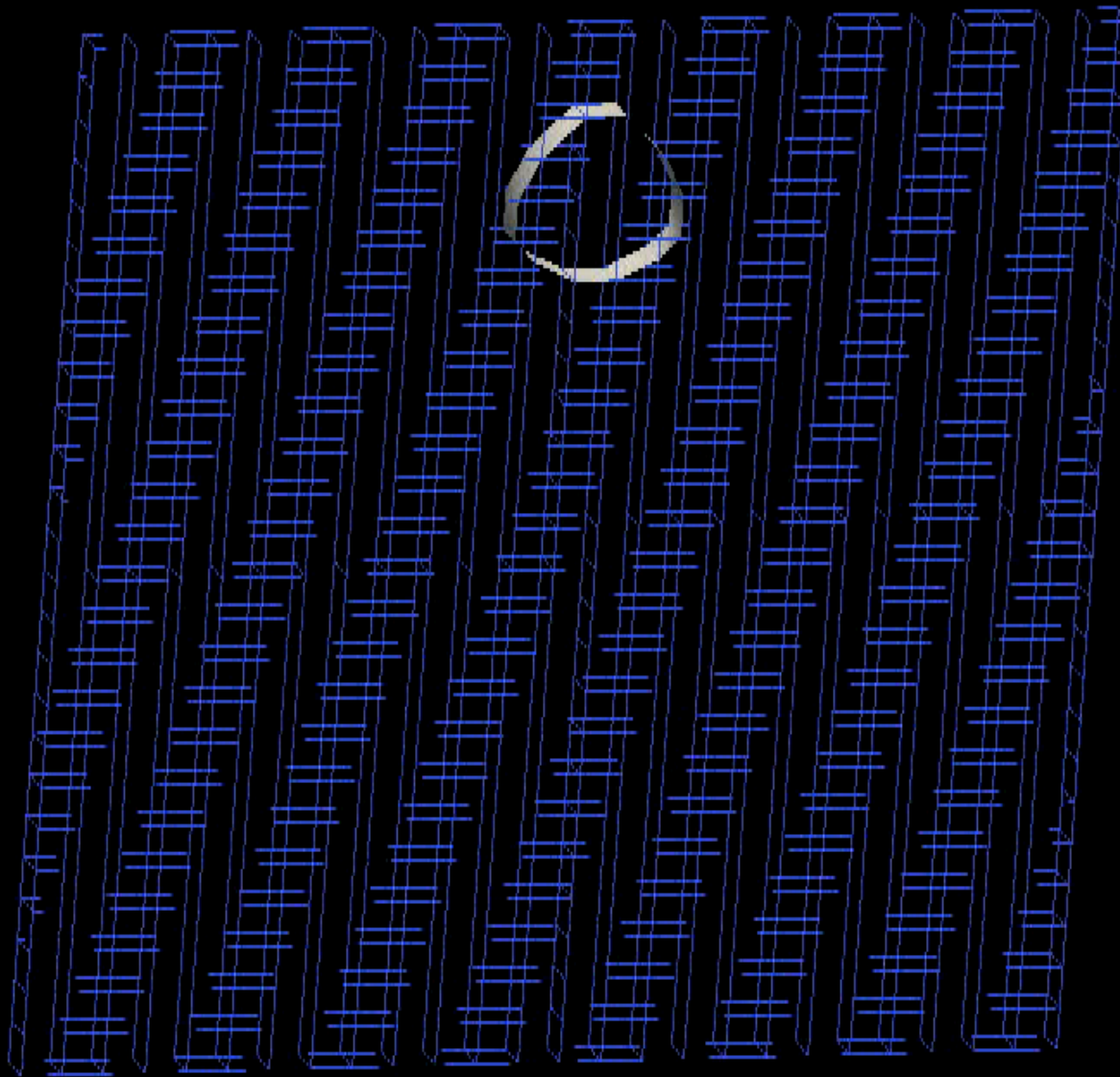
    // Refine field inbetween lower..upper
    lowerRefineLevel 0.3;
    upperRefineLevel 1e200;

    // If value < unrefineLevel unrefine
    unrefineLevel 0.05;

    // Have slower than 2:1 refinement
    nBufferLayers 1;

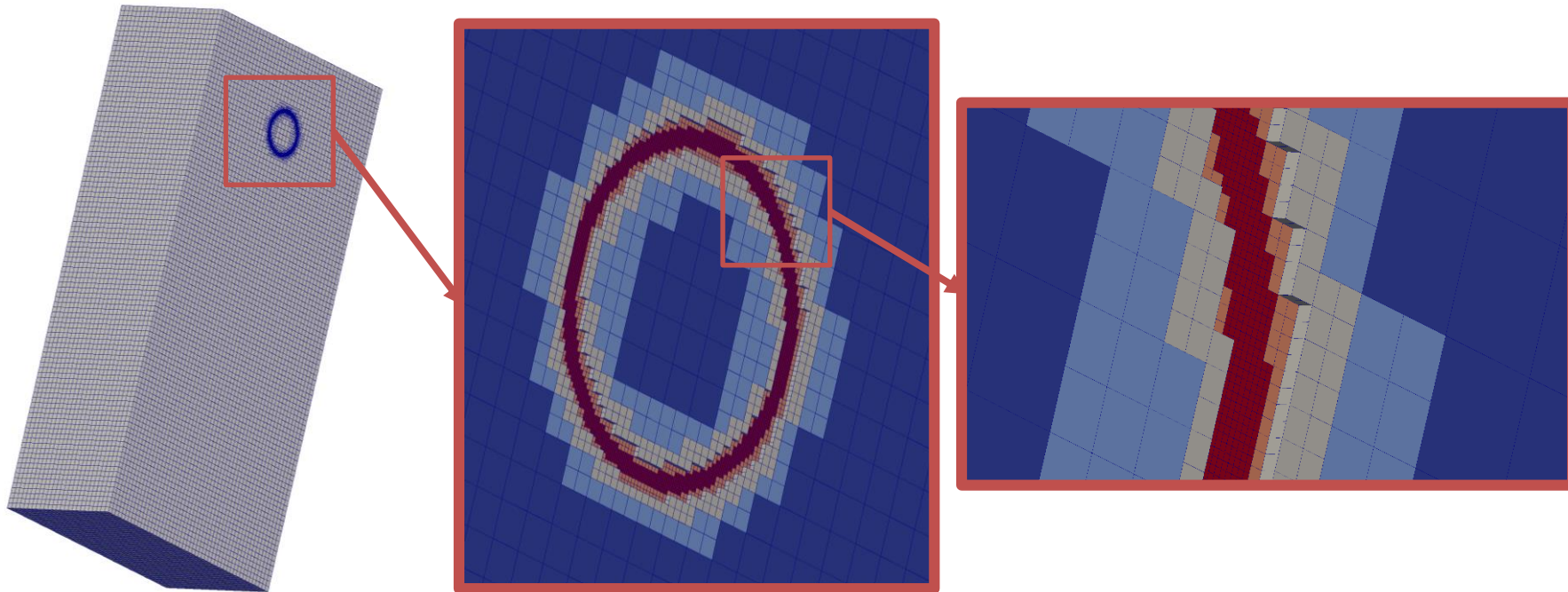
    // Refine cells only up to maxRefinement levels
    maxRefinement 5;

    // Stop refinement if maxCells reached
    maxCells 10000000;
```



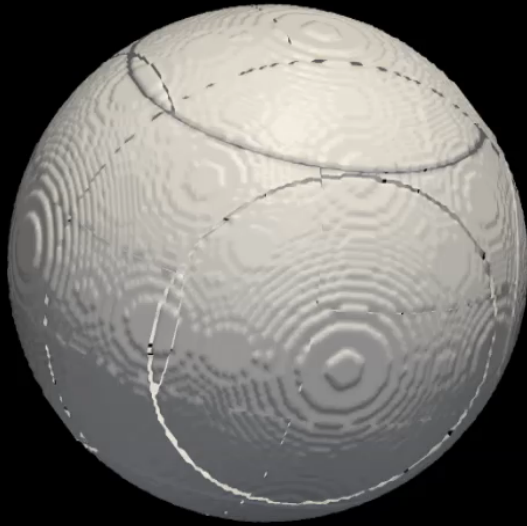
DYNAMIC REFINEMENT

LARGE SCALE CASE



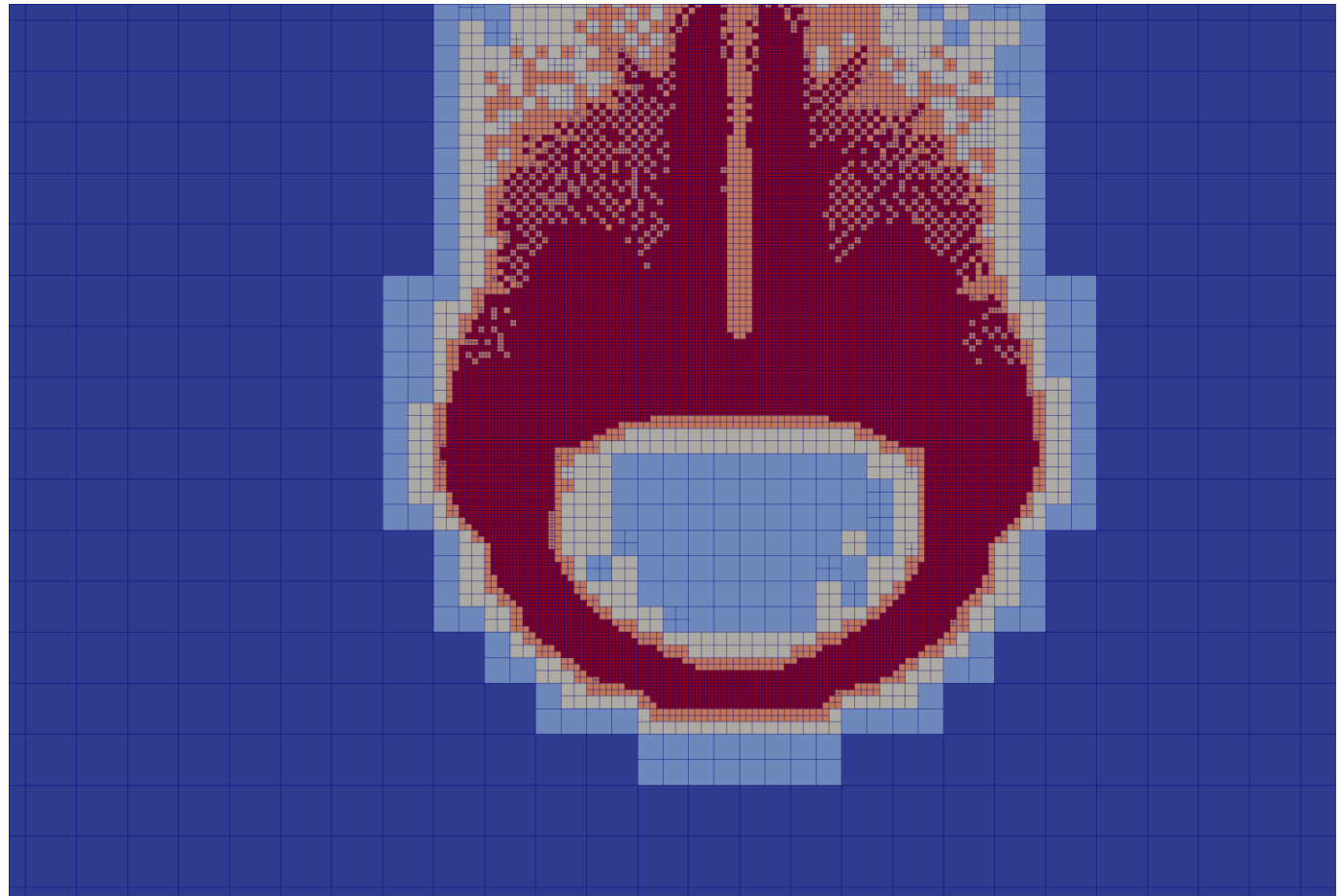
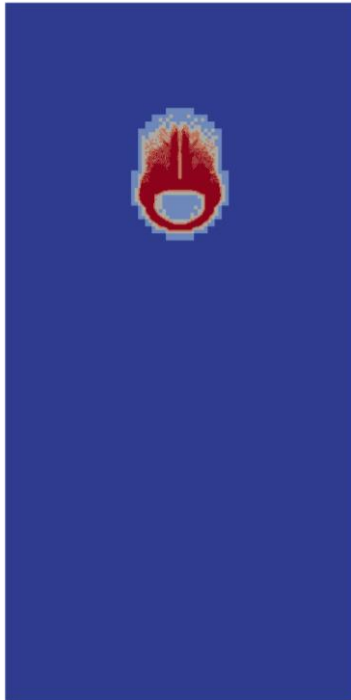
Eo	196.875
Oh_d	6.32E-01
Oh_c	2.00E-01
Vis ratio	10
Density ratio	10

Gravity driven droplet acceleration / breakup
Max 5 levels of refinement
Equivalent to ~1.2 billion cells of uniform mesh size
Actual max mesh size = ~3.5 million



DYNAMIC REFINEMENT

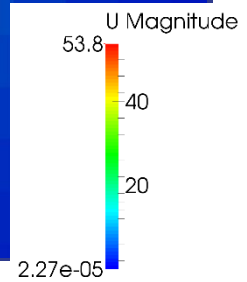
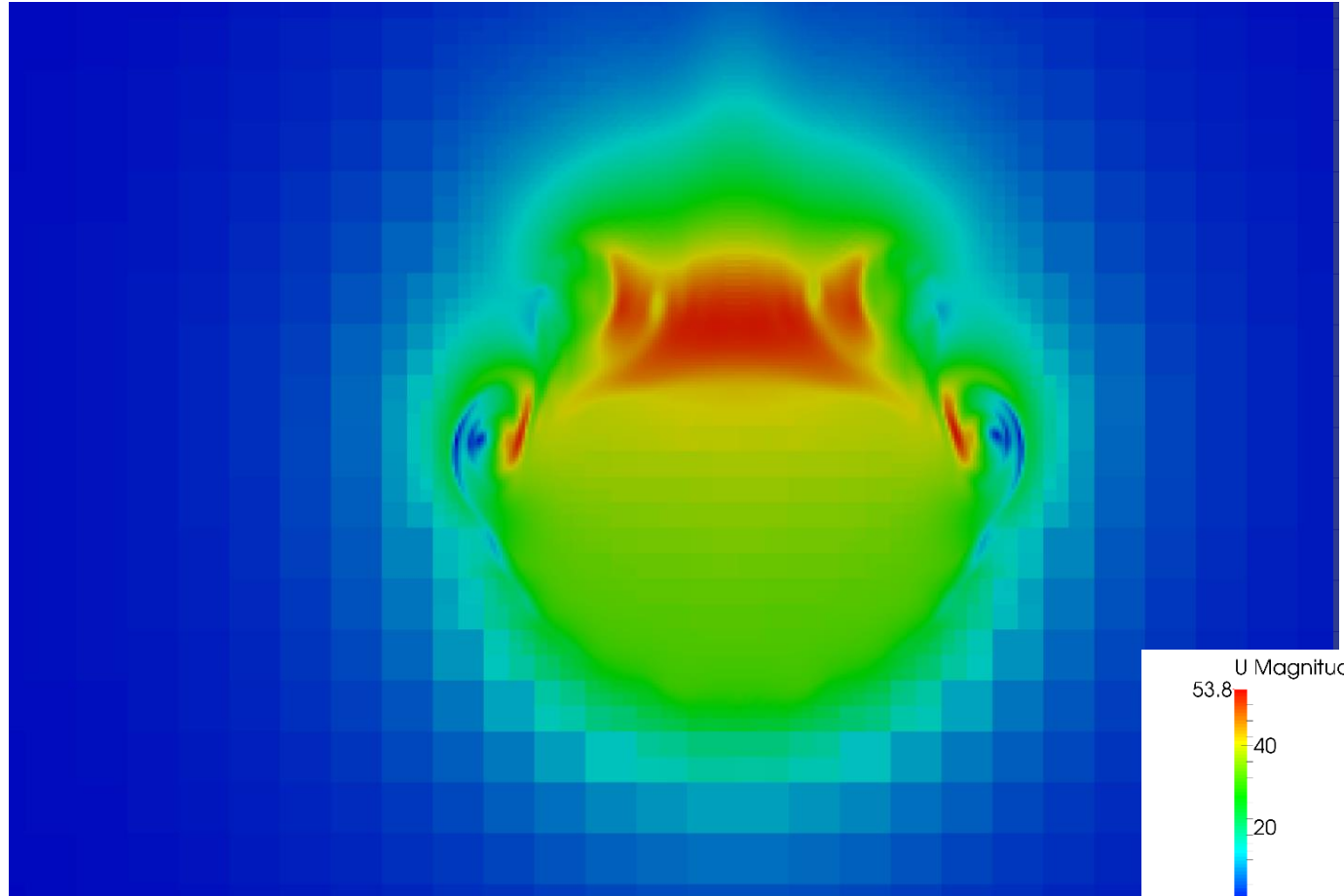
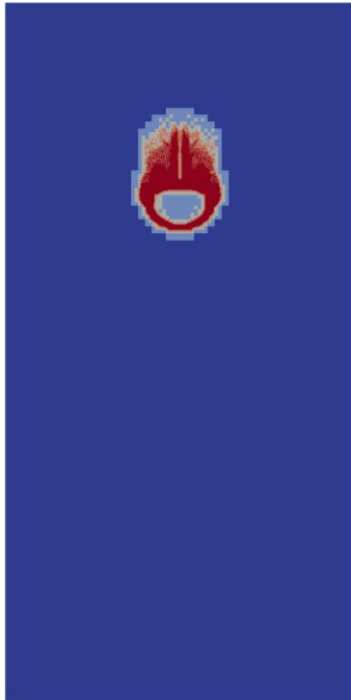
LARGE SCALE CASE – CELL LEVEL



Eo	196.875
Oh_d	6.32E-01
Oh_c	2.00E-01
Vis ratio	10
Density ratio	10

DYNAMIC REFINEMENT

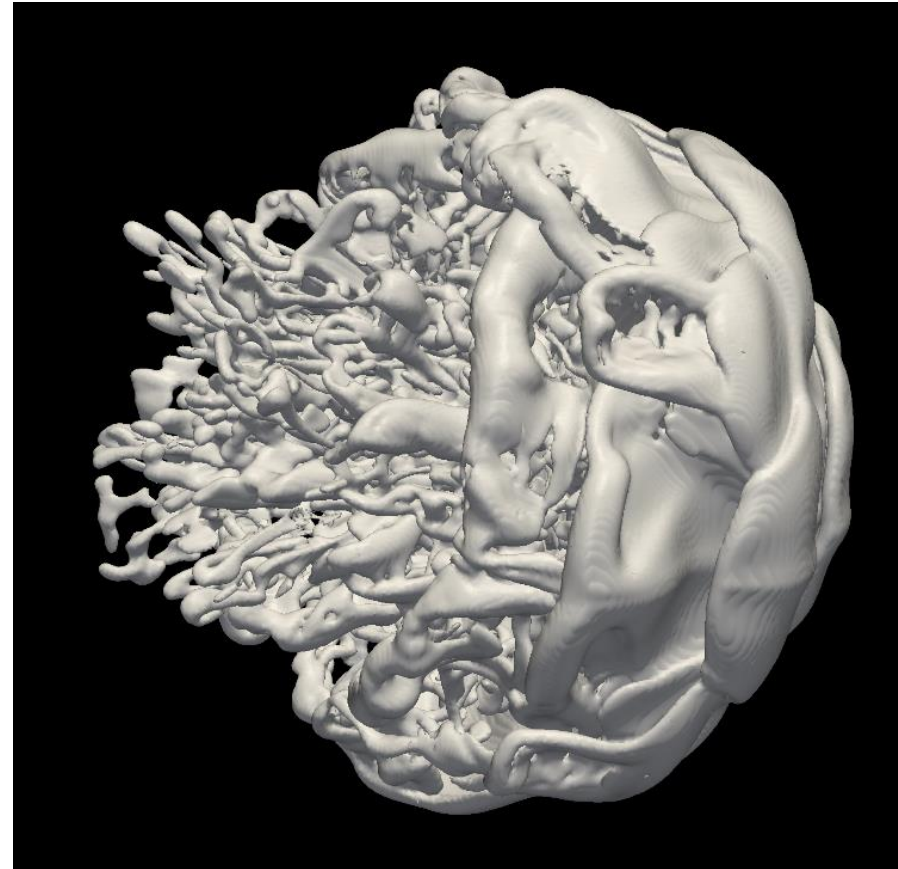
LARGE SCALE CASE – MAG(U)



Eo	196.875
Oh_d	6.32E-01
Oh_c	2.00E-01
Vis ratio	10
Density ratio	10

NEXT STEPS

- Incorporate error estimation
- Fix additional memory issues with PT-Scotch
- Extend to operate within non-hexahedral meshes (ex: hex-core within polyhedral)
- Apply immersed boundary / moving immersed boundary methods
- Optimize for run time: re-meshing frequency, allowable imbalance.



THANK YOU COLLABORATORS

Collaborators / Advisors:

- Prof. David Schmidt
- Prof. Timothee Pourpoint
- Prof. Stephen Heister



PURDUE
UNIVERSITY

THANK YOU