SIGGRAPH2013

A New Grid Structure for Domain Extension

Bo Zhu[1] Wenlong Lu[1], Matthew Cong[1,2], Byungmoon Kim[3], Ronald Fedkiw[1,2]

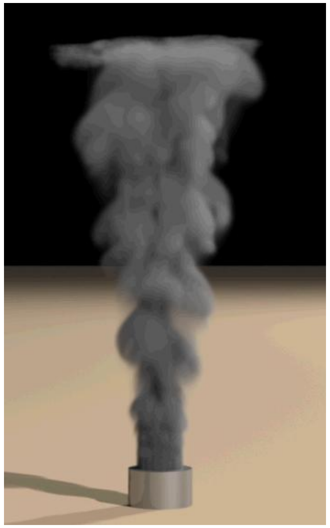
[1]Stanford University
[2]Industrial Light+Magic
[3]Adobe Systems Inc.

Grid-based Fluid Simulation

- **Advantages:**
  - Cache coherent memory layout
  - Regular domain subdivision
  - Fast iterative solvers
  - Axis-aligned voxel data for ray tracing
- **Disadvantages:**
  - The fluid is restricted inside the bounding box of the grid

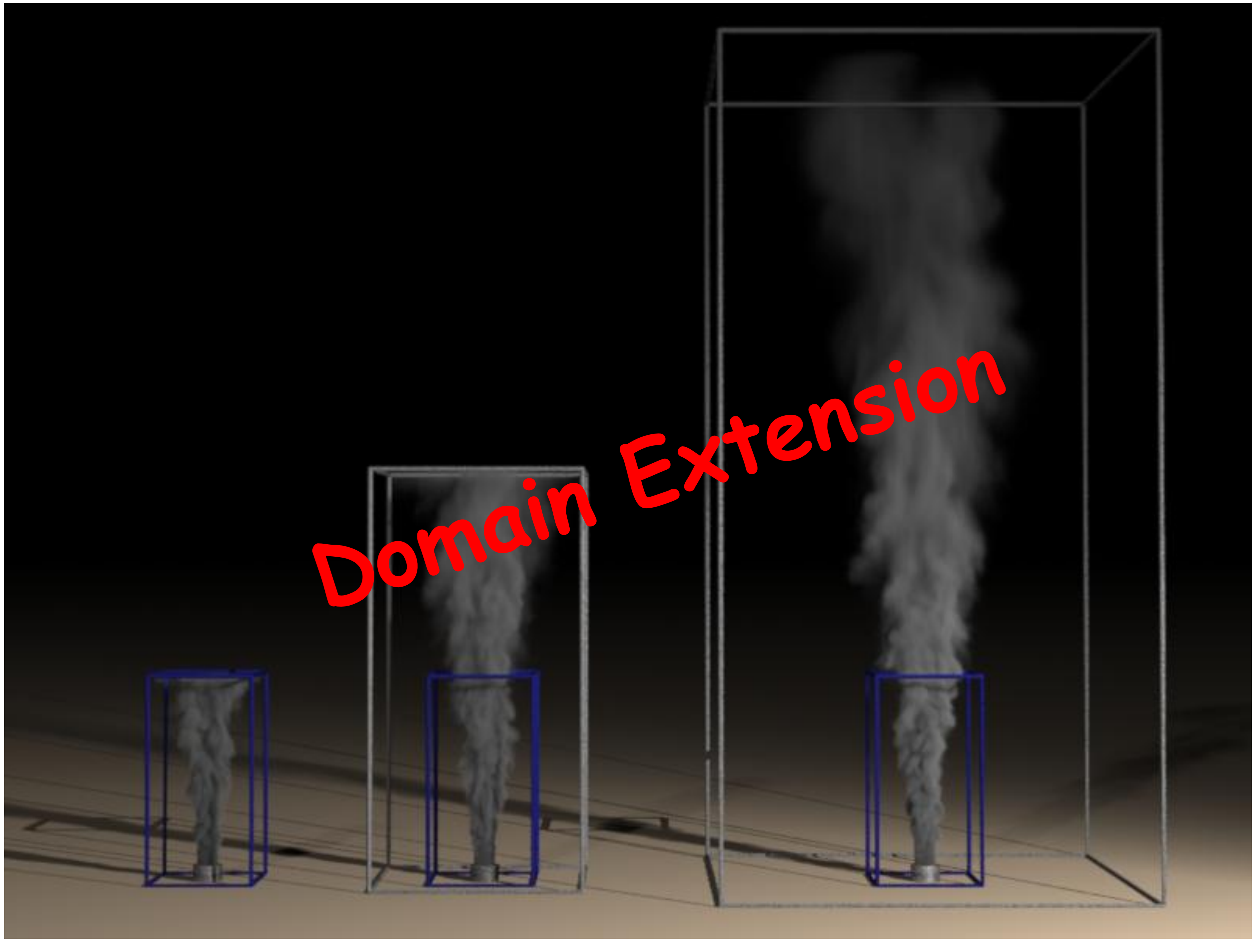
Simulating Fluids in Large Domains

- **In reality the fluid motion is unpredictable**
  - tends to reach the boundary of a wide space
- **Artifacts due to grid boundaries:**
  - Incorrect density distribution
  - Reflective waves



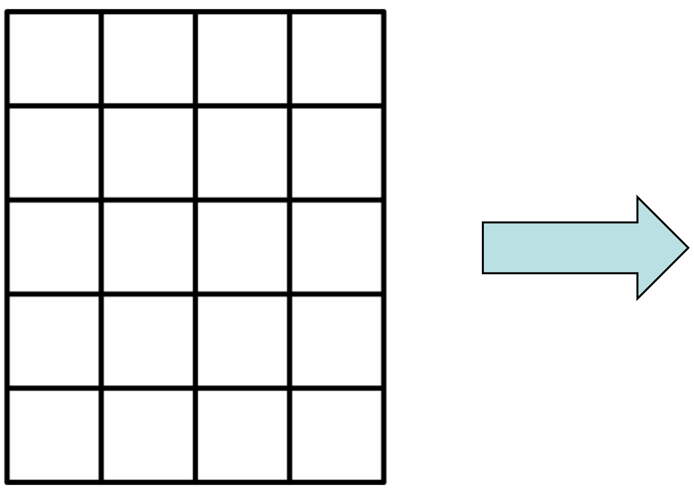
SIGGRAPH2013

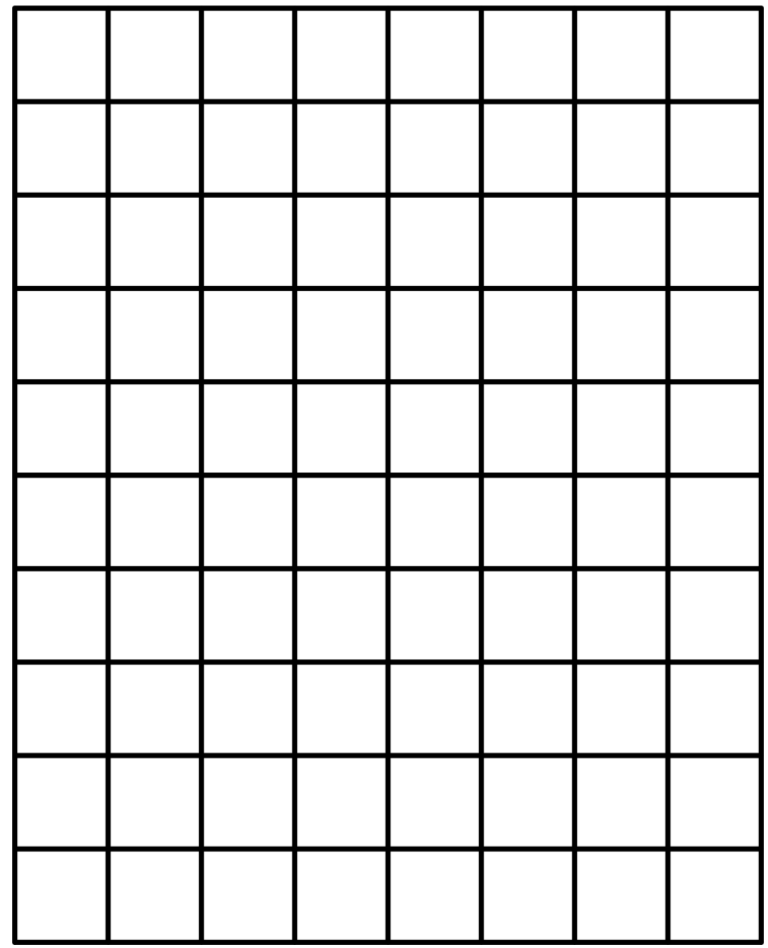
Use Larger Grids



Difficulty in Domain Extension

If we fill the entire domain with fine cells…

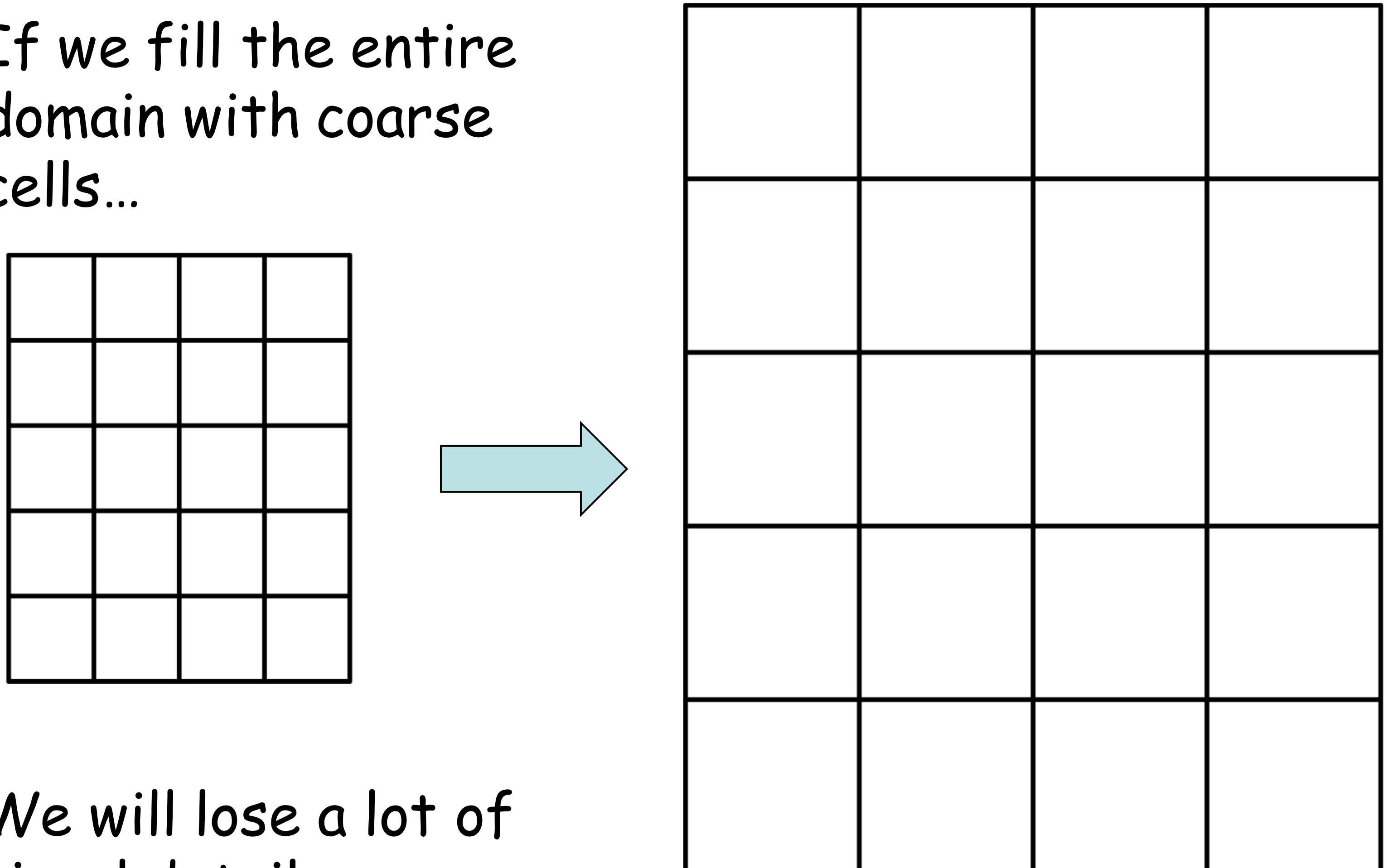




The number of cells increases by $O(n^2)$ in 2d and $O(n^3)$ in 3d

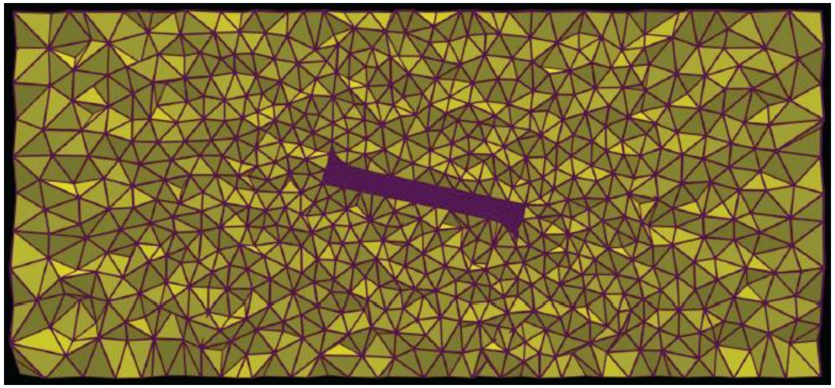
Difficulty in Domain Extension

If we fill the entire domain with coarse cells…

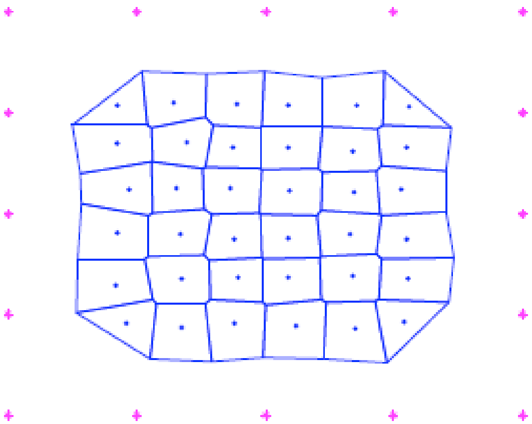


We will lose a lot of visual details

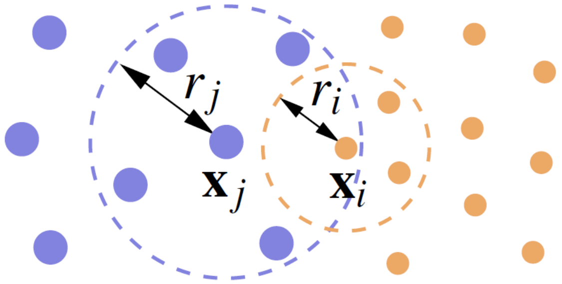
Related Work:
Data Structures in Fluid Simulation



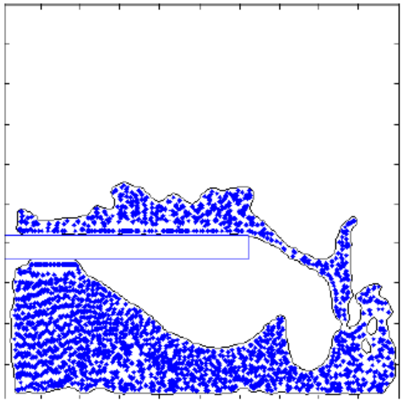
Tetrahedral Mesh
[Klingner et.al. 2006]



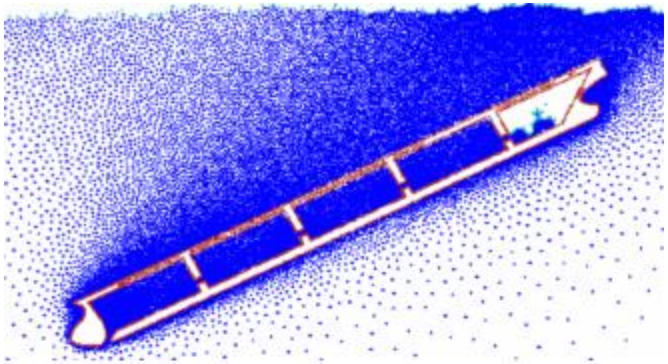
Voronoi Diagram
[Sin et.al. 2009]



Adaptive Particles
[Adams et.al. 2007]

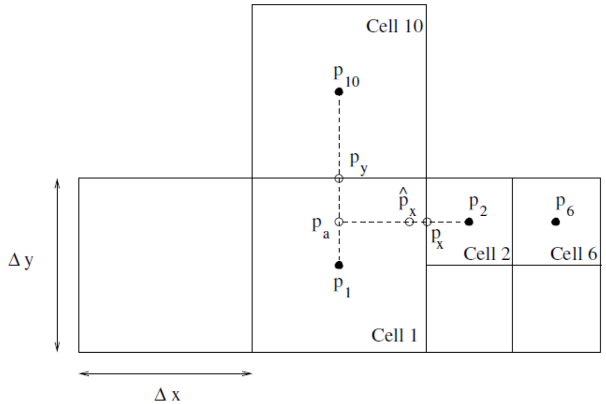


FLIP/PIC
[Zhu et.al. 2005]

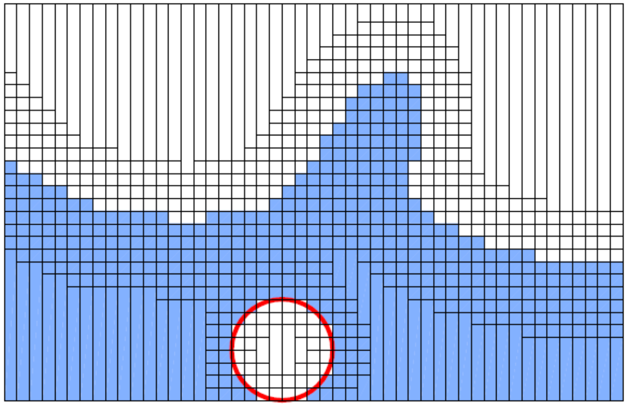


Particle FEM
[Idelsohn et.al. 1996]

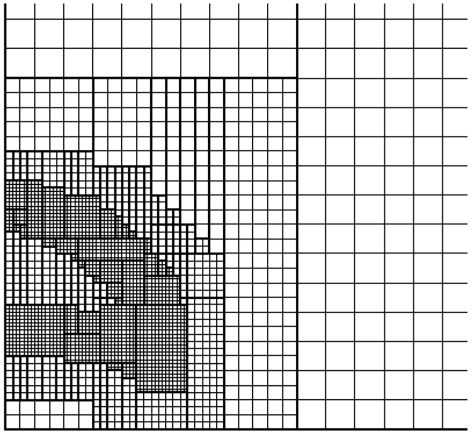
Related Work: Data Structures for Grids



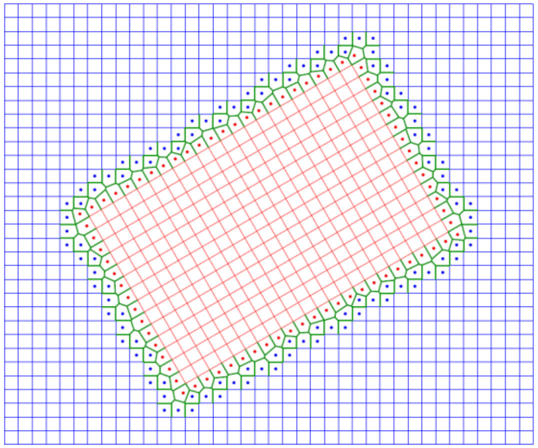
Octree
[Losasso et.al. 2004]



RLE Grid
[Irving et.al. 2006]



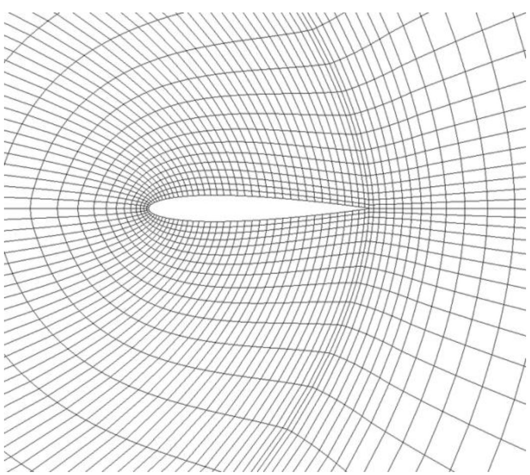
Adaptive Mesh Refinement
[Deffy et.al. 2011]

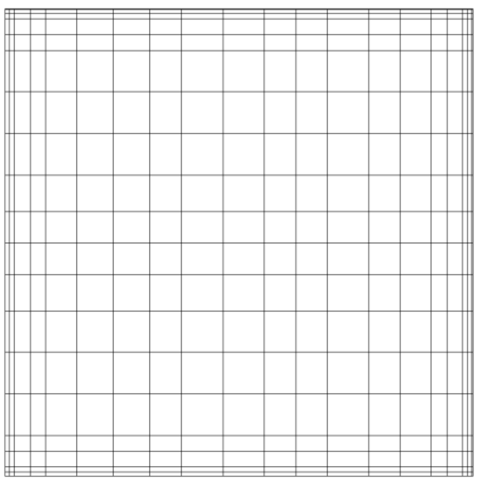


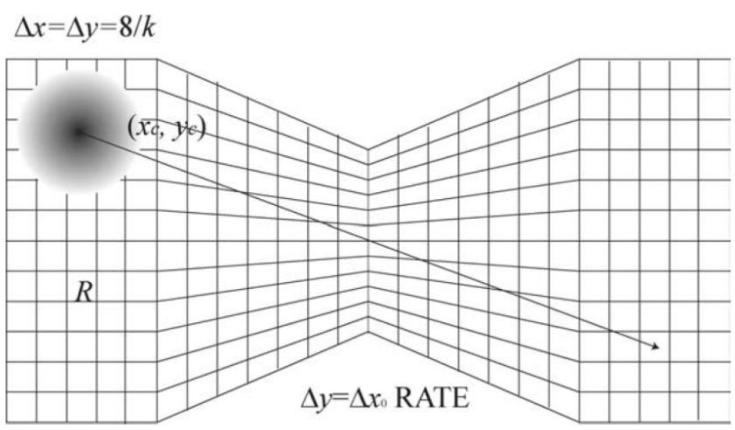
Chimera Grid
[English et.al. 2013]

Related Work: Curvilinear\Rectilinear\Soroban Grids

- Mapping a boundary fitted, stretched, or compressed physical domain into a uniform computational domain
- Jacobian terms are added in the equations for the mapping


Curvilinear


Rectilinear



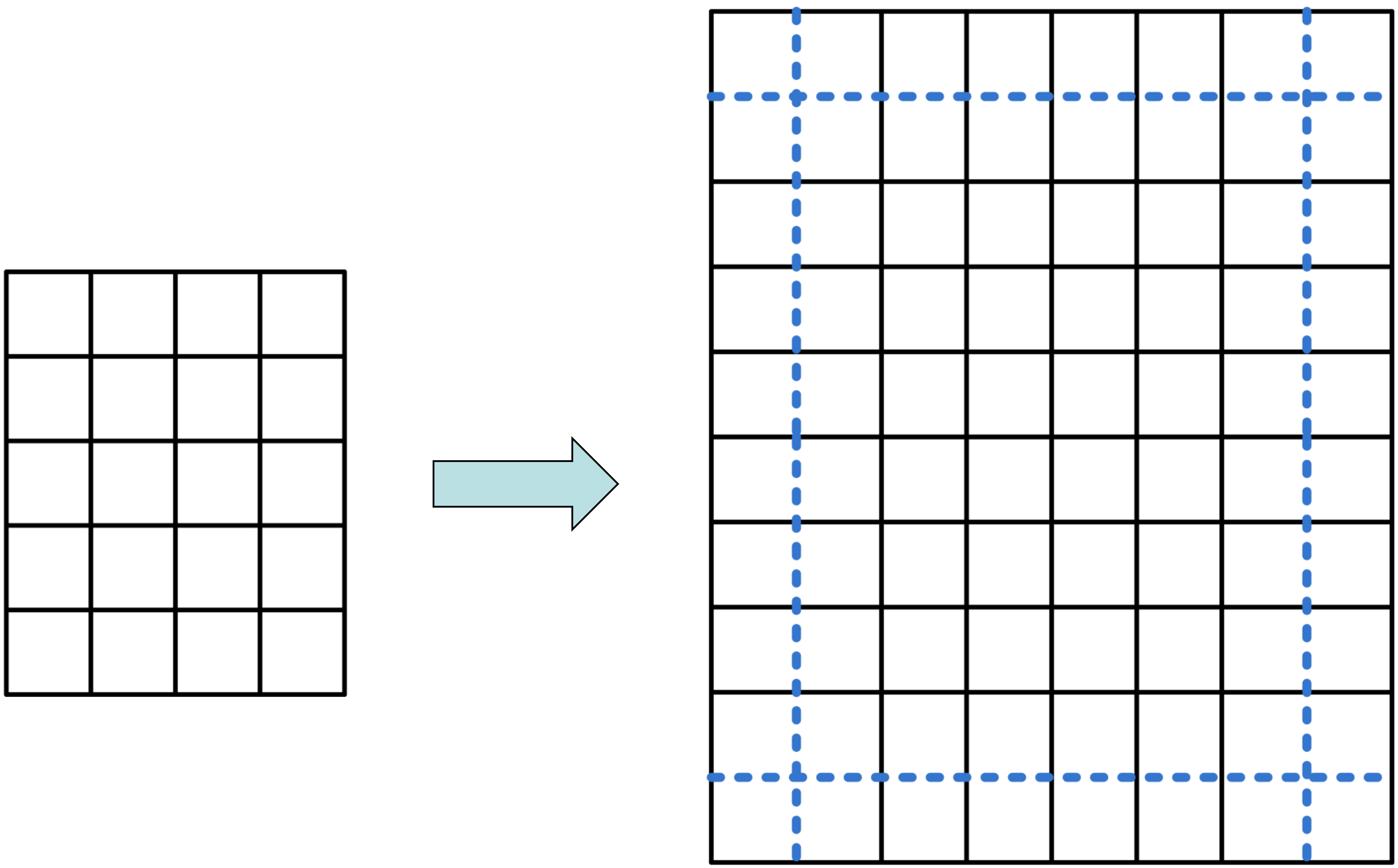
Soroban

Our Goal

- Simplicity of uniform grid
- Still adaptive

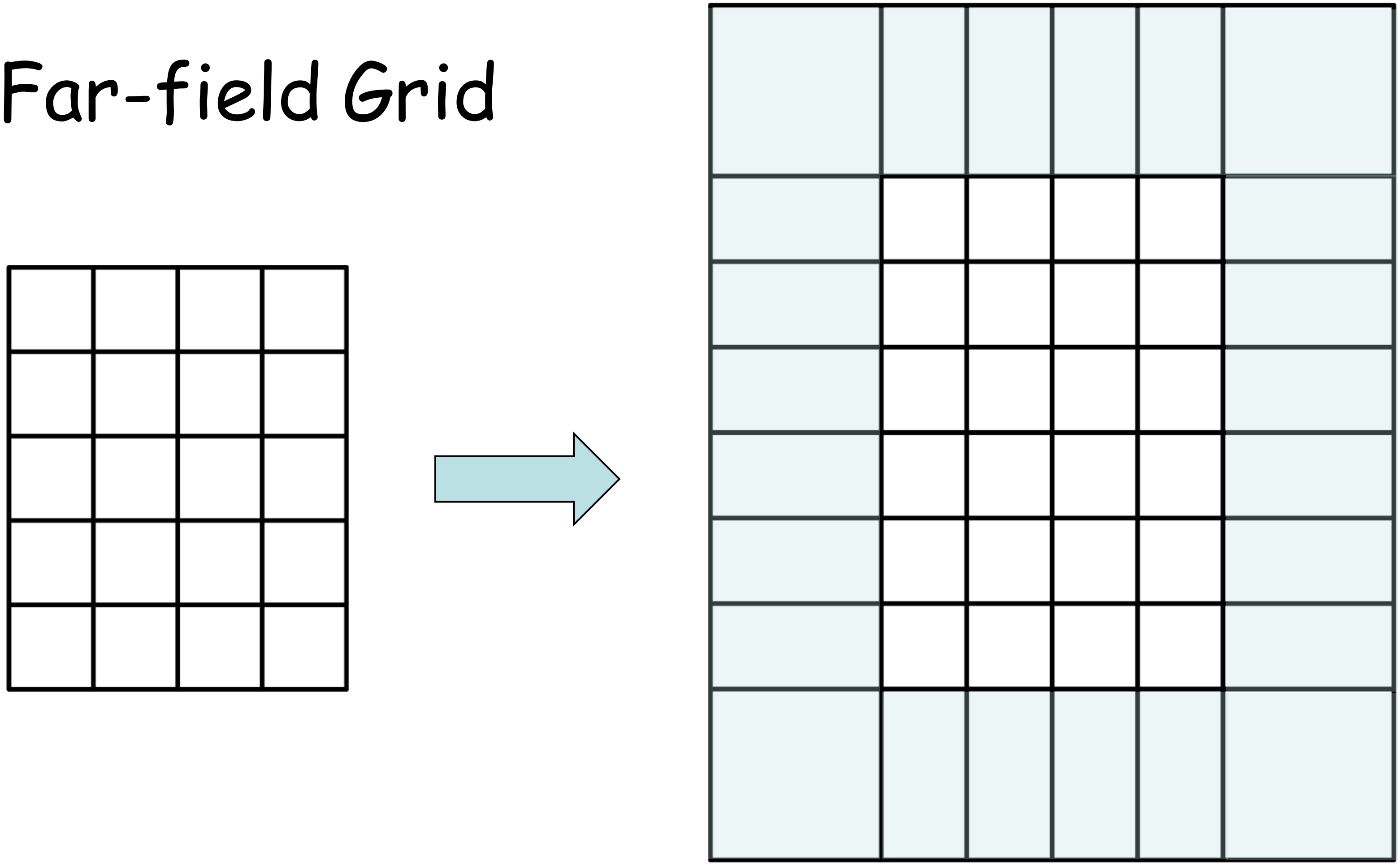
- No Octree/tetmesh/particles/Jacobians

Our Solution

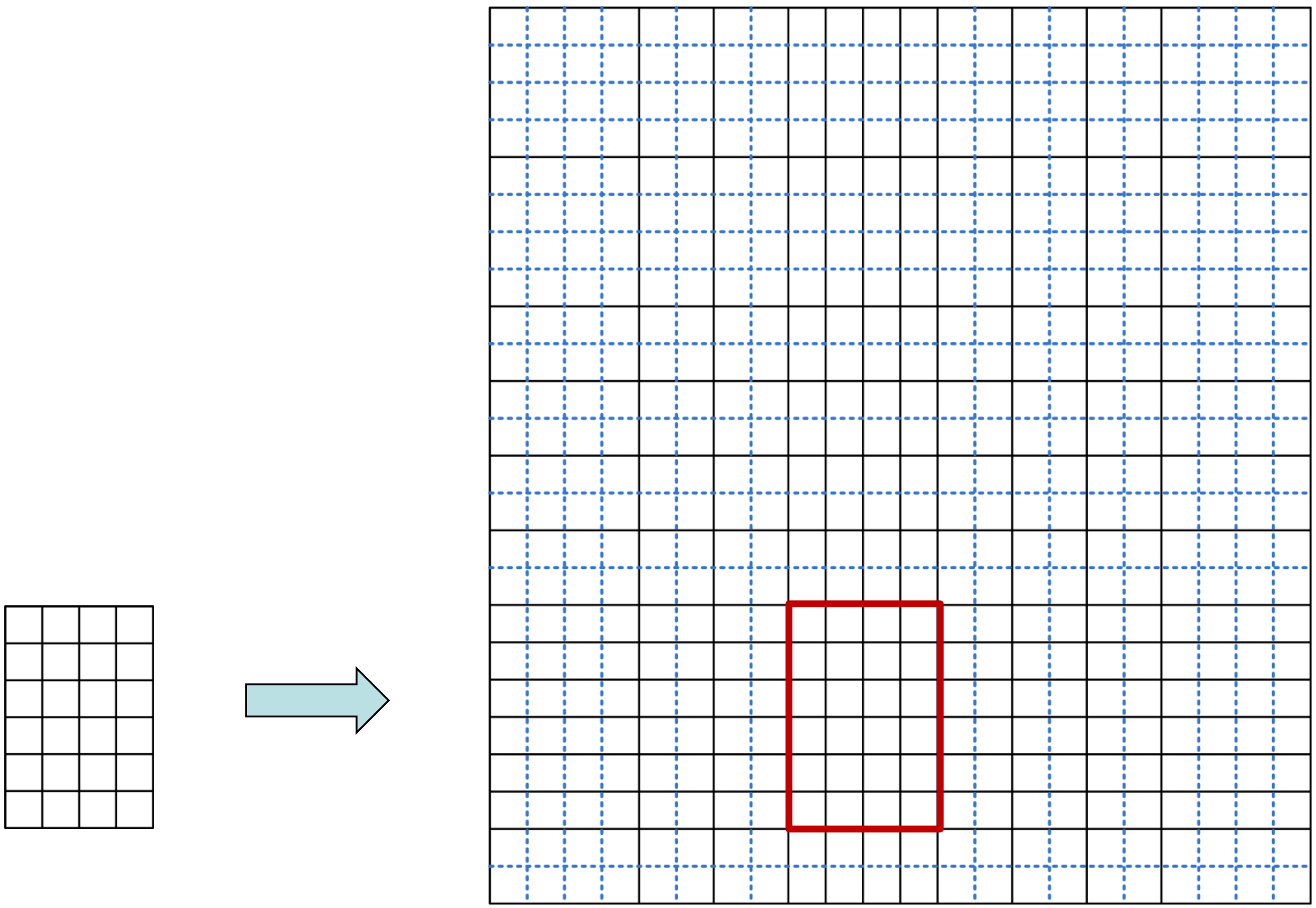


Our Solution

Far-field Grid

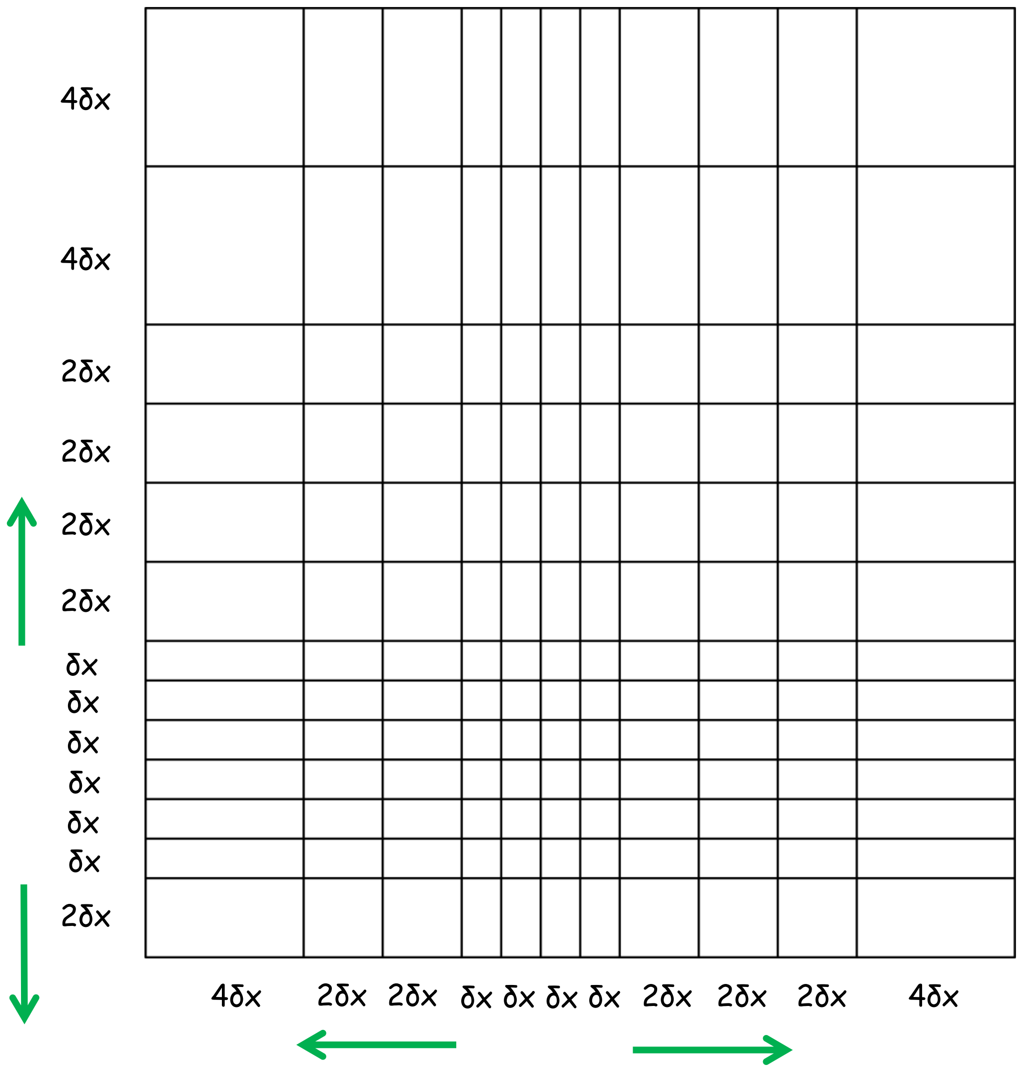


Adaptive Extension



Adaptive Extension

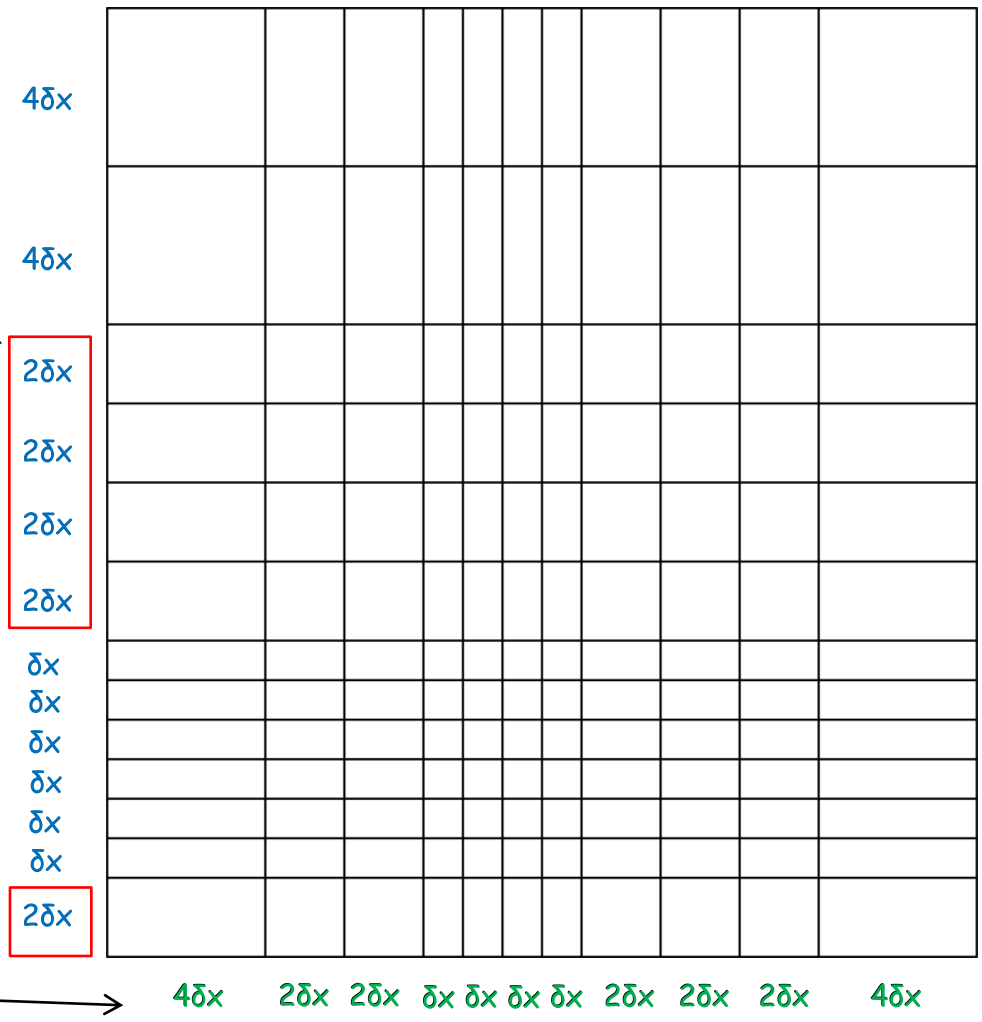
- Cell length $2^{i-1}\delta x$.
- Each dimension is independent
- Extended in both "+" and "-"directions



Adaptive Extension

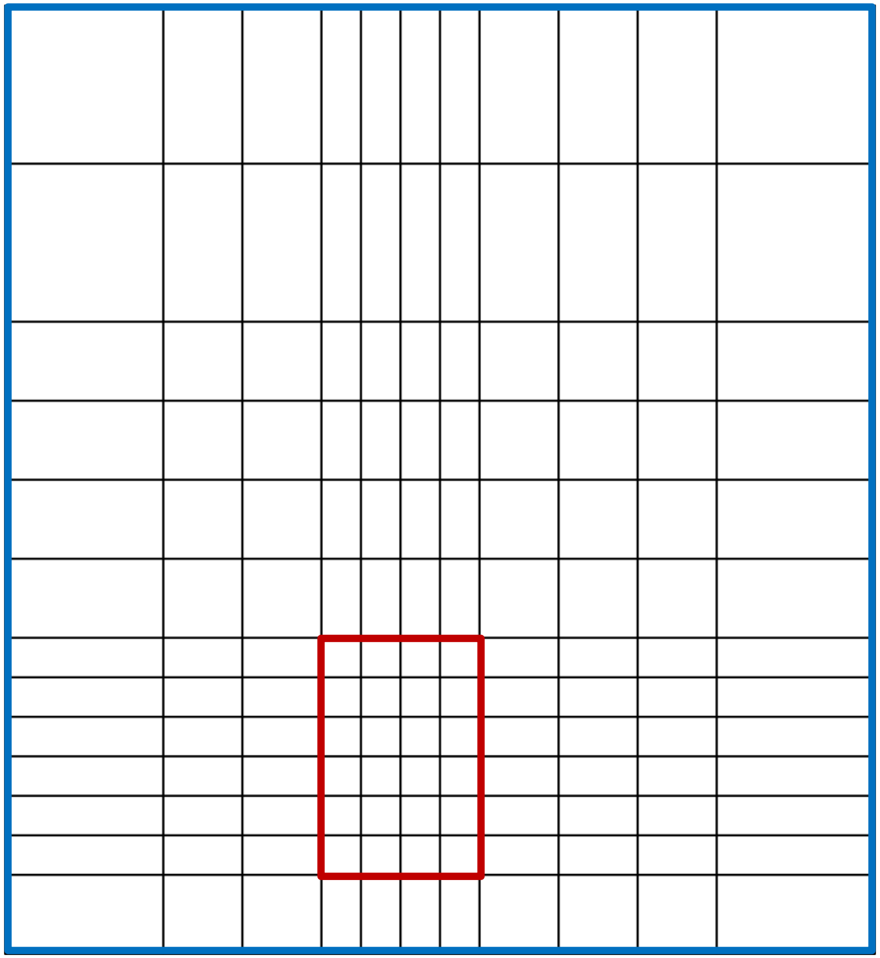
Y-Axis:
Layer 1: 6
Layer 2: (1, 4)
Layer 3: (0, 2)

X-Axis:
Layer 1: 4
Layer 2: (2, 3)
Layer 3: (1, 1)



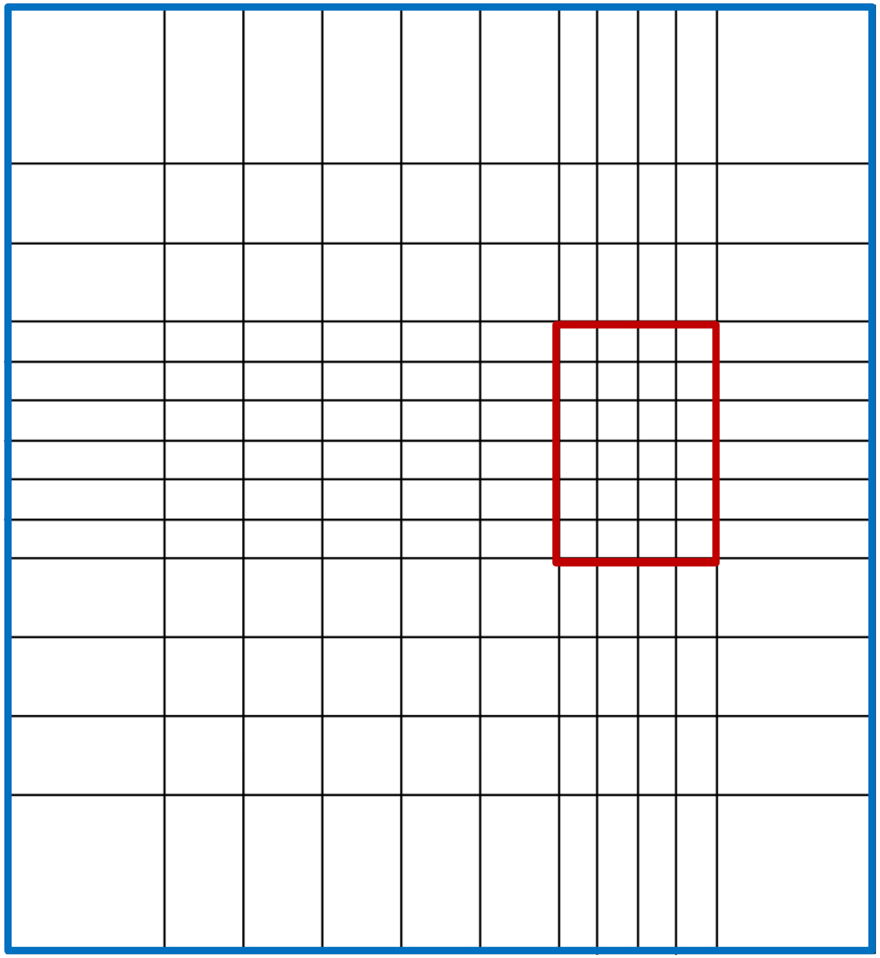
Two Grid Boxes

- The interior box with the finest resolution to resolve fine details
- The exterior box with gradually coarsened resolutions to enclose the entire fluid

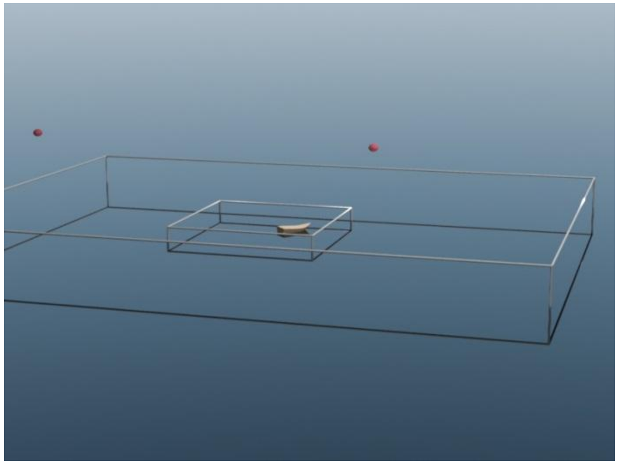


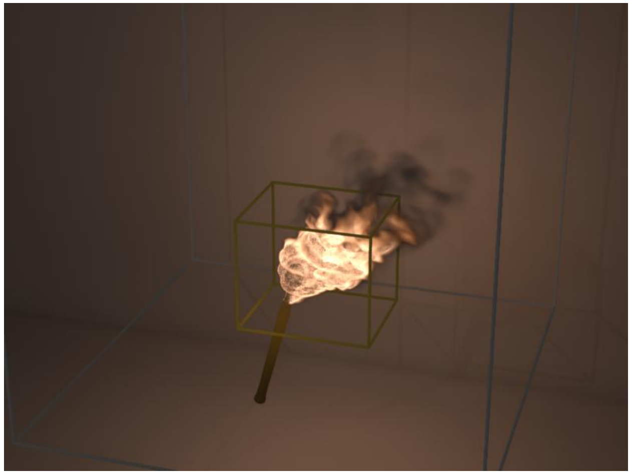
Dynamic Tracking

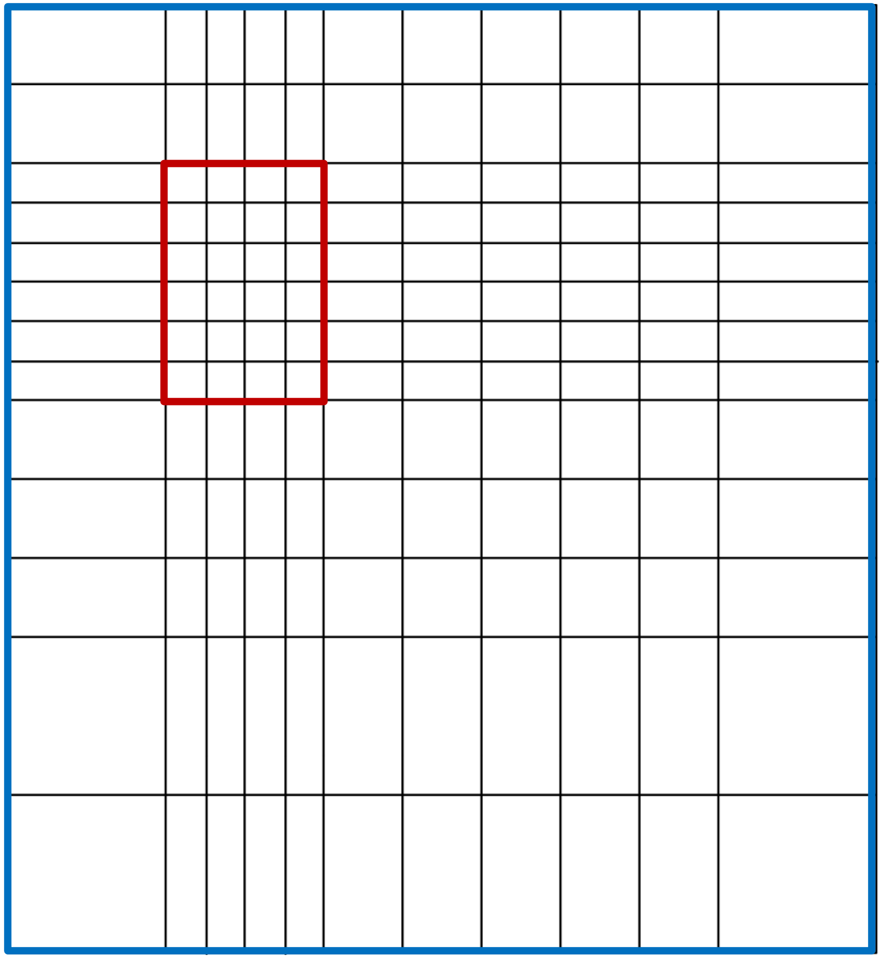
- Change the relative positions of the two grid boxes



Dynamic Tracking

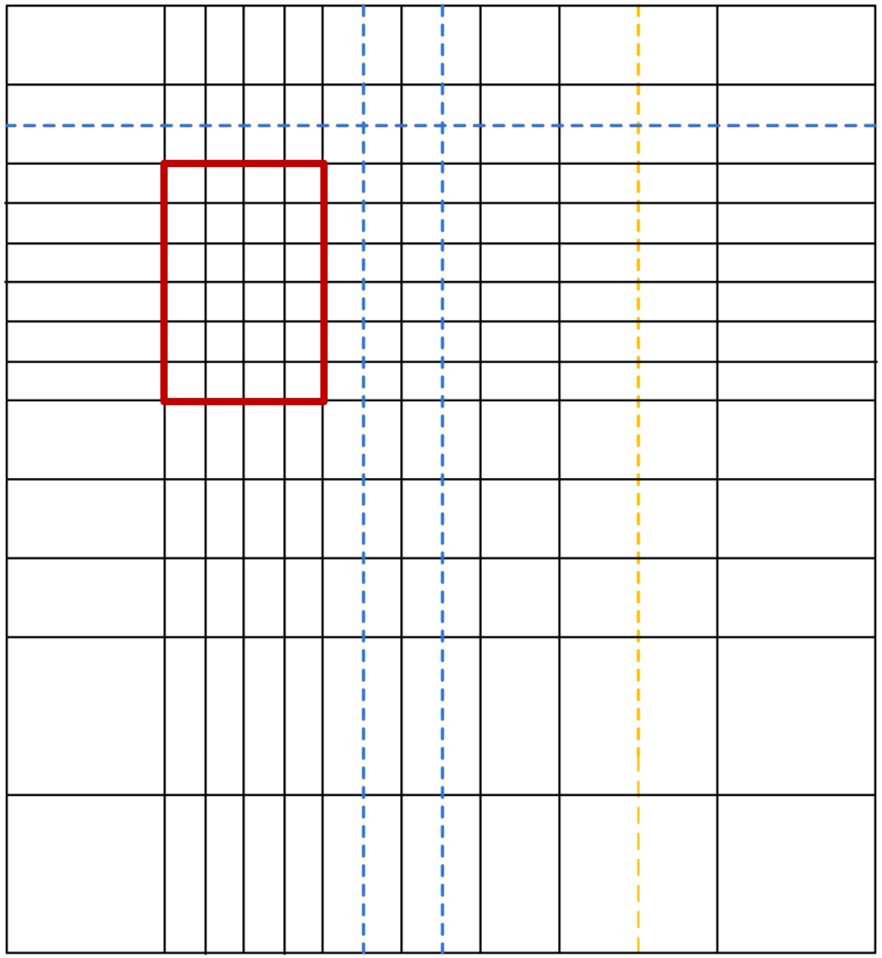




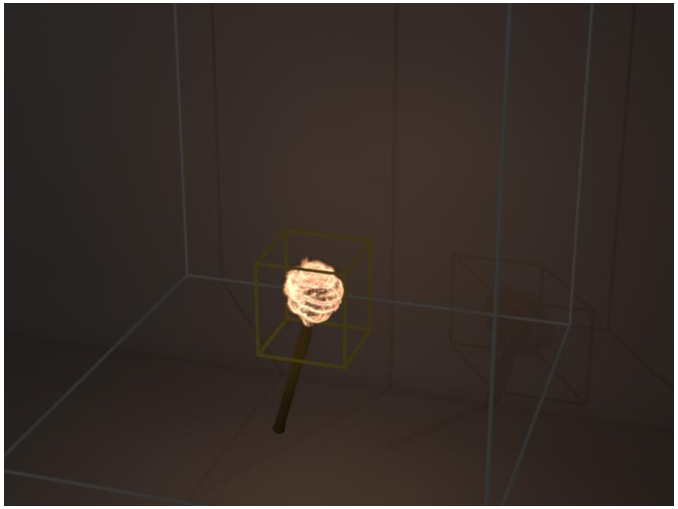


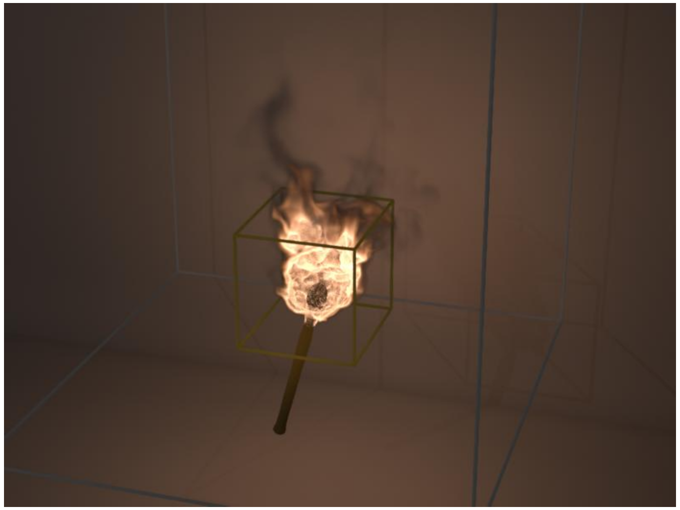
Dynamic Tracking

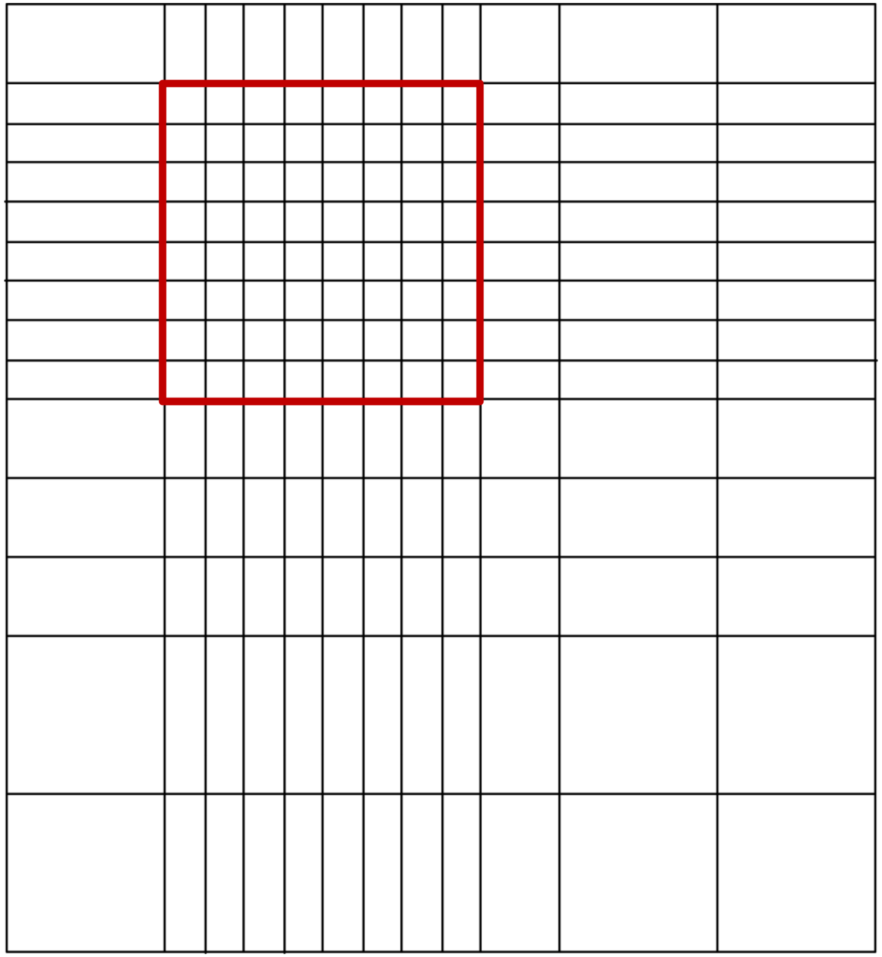
- Translate, add, or delete grid lines



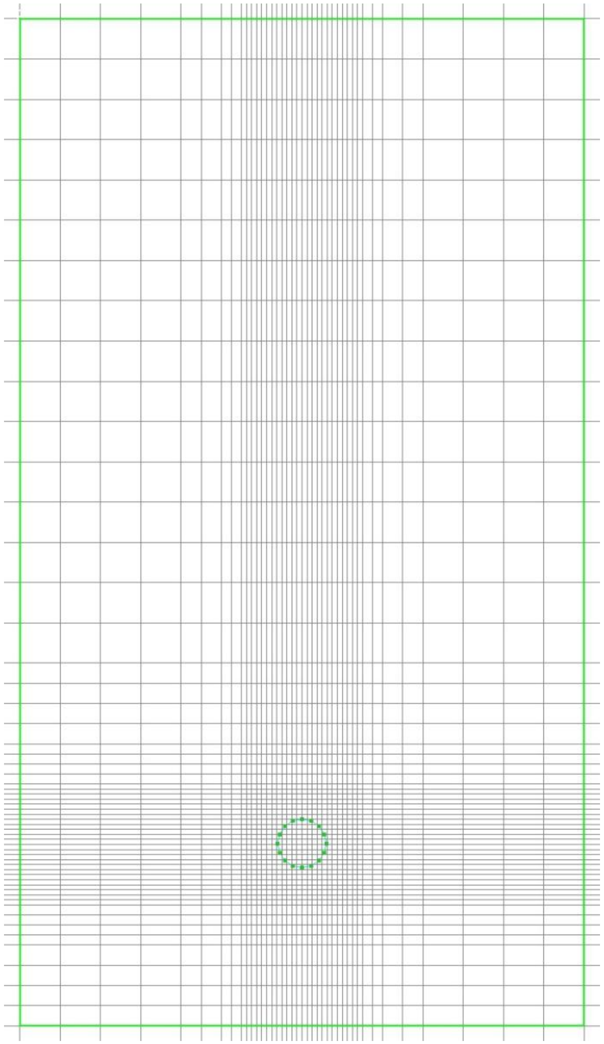
Dynamic Tracking

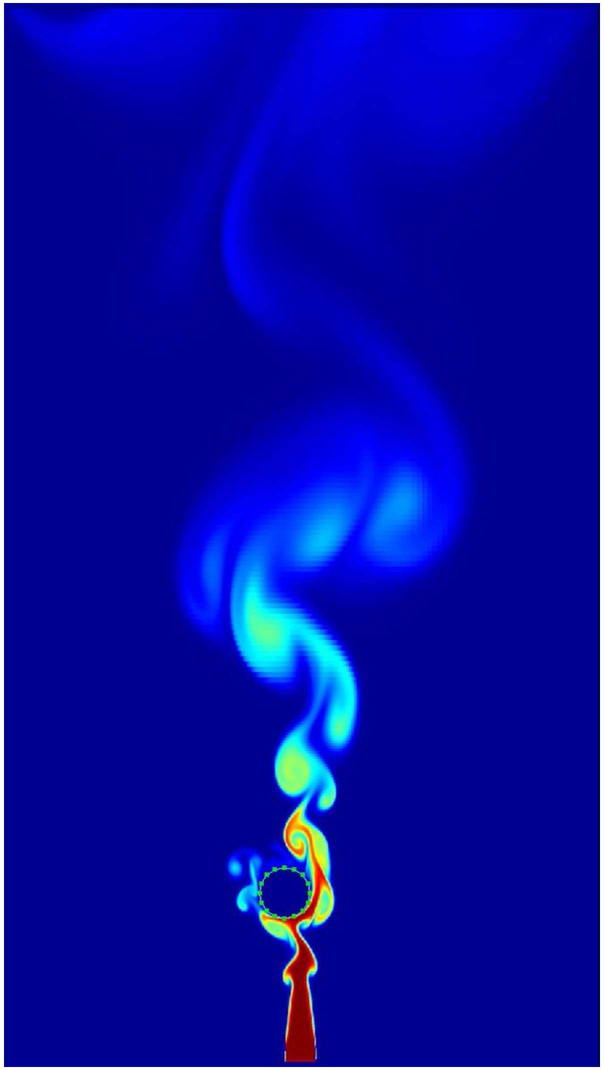






Dynamically Allocating Grid Lines

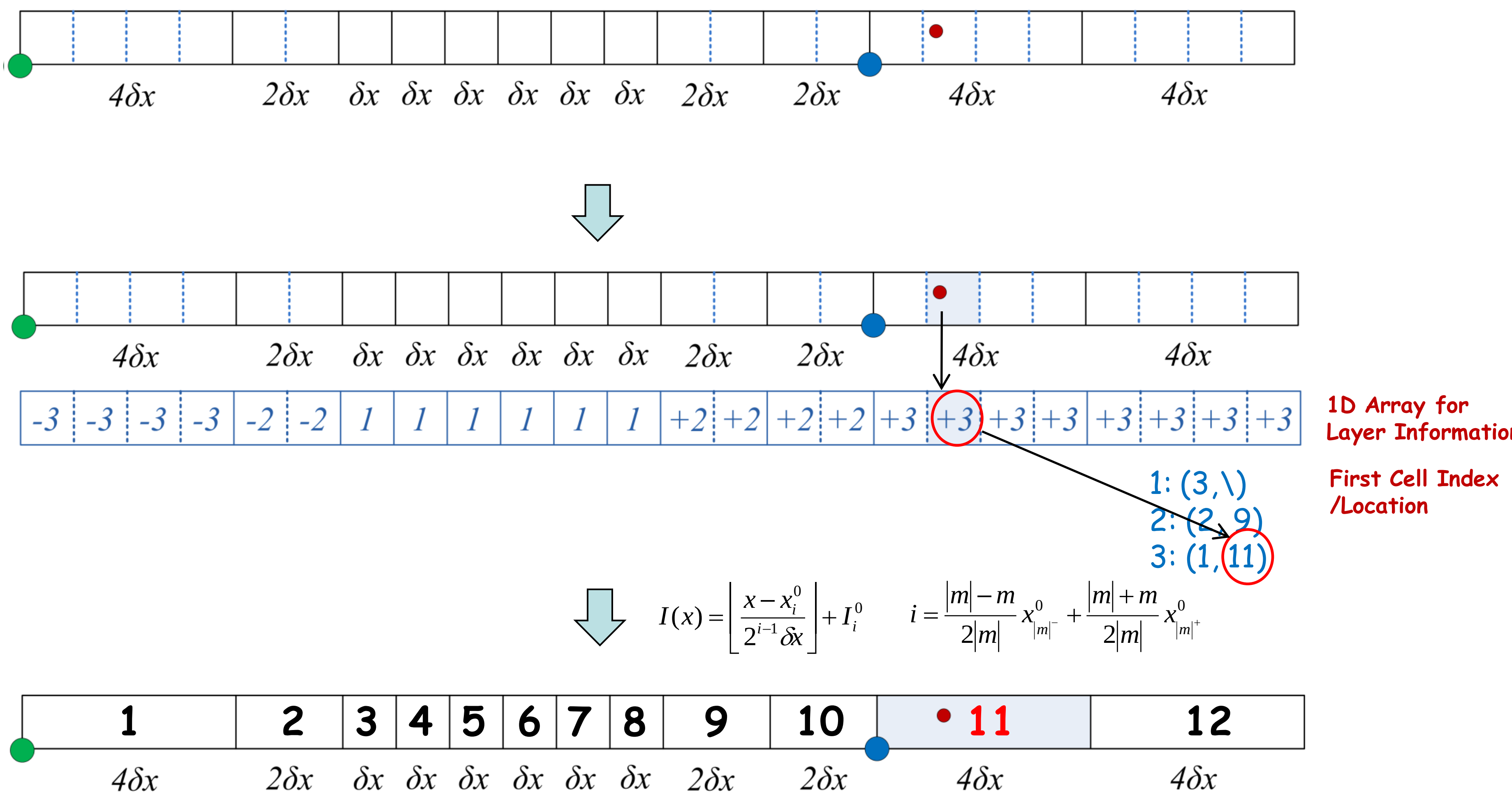




Uniform Grid Plus …

- **Extra Information Per Axis**
  - 1D array of layer index per every grid cell
  - Each layer's first cell position & index (for both "+" and "-" sides)
- **Data Access in O(1) time**
  - Location-to-index
  - Index-to-location

Fast Index Access

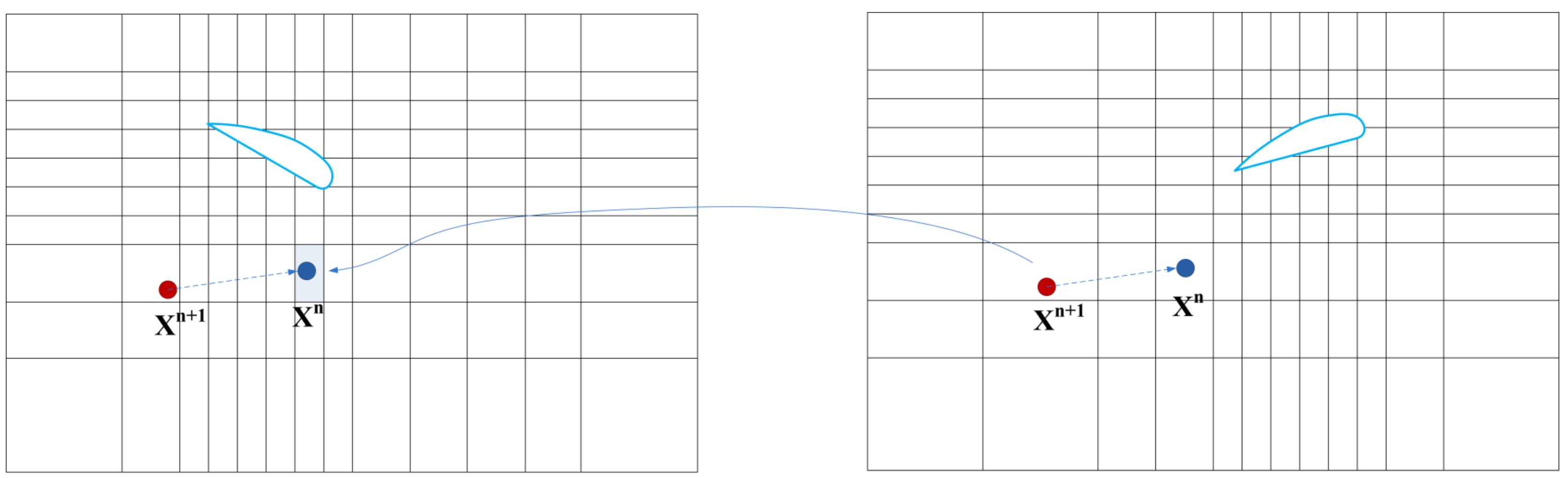


$$I(x) = \left\lfloor \frac{x - x_i^0}{2^{i-1}\delta x} \right\rfloor + I_i^0 \qquad i = \frac{|m| - m}{2|m|} x^0_{|m|^-} + \frac{|m| + m}{2|m|} x^0_{|m|^+}$$

Solving Incompressible Flow: Advection

- Store the grid structures at both time n and n+1
- Trace the semi-Lagrangian rays backward in time and interpolate from the time n grid

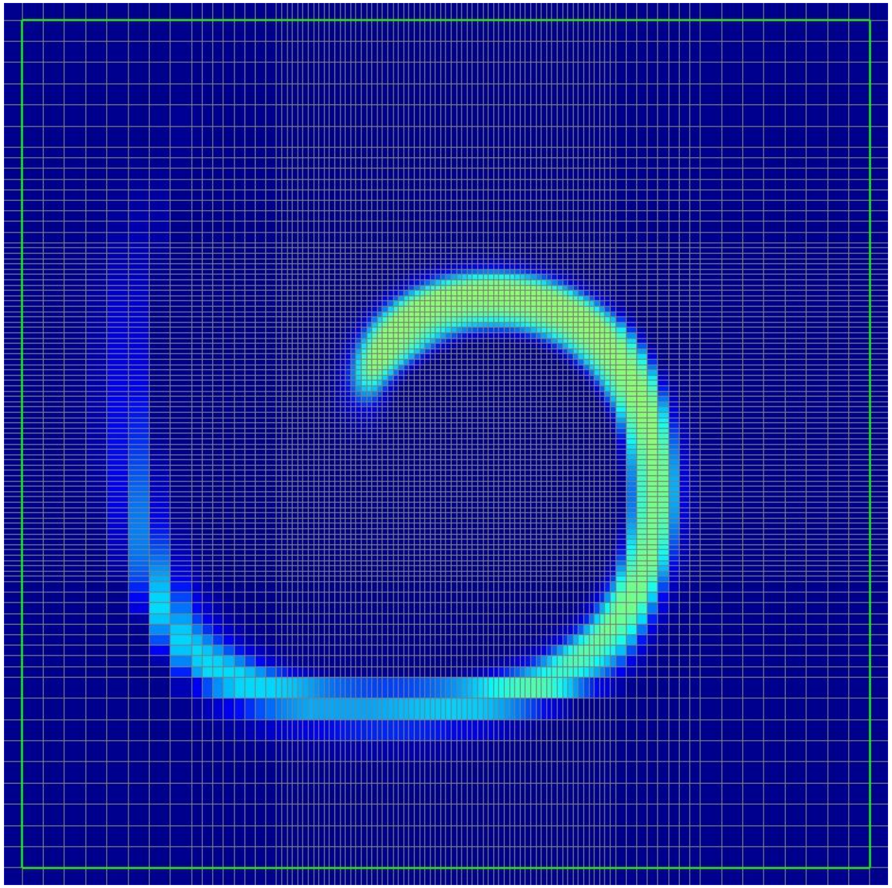
$$\frac{\mathbf{u}^* - \mathbf{u}^n}{\Delta t} + (\mathbf{u}^n \cdot \nabla)\mathbf{u}^n = \mathbf{f}$$

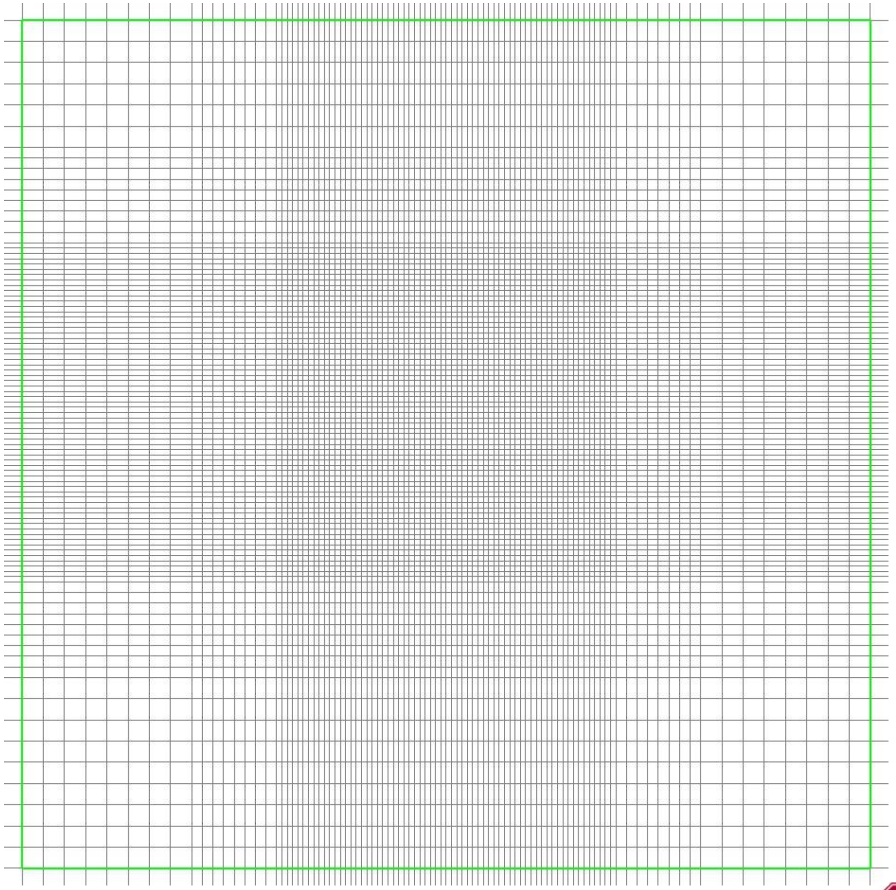


Grid in Time n

Grid in Time n+1

Advection on the Dynamic Grid



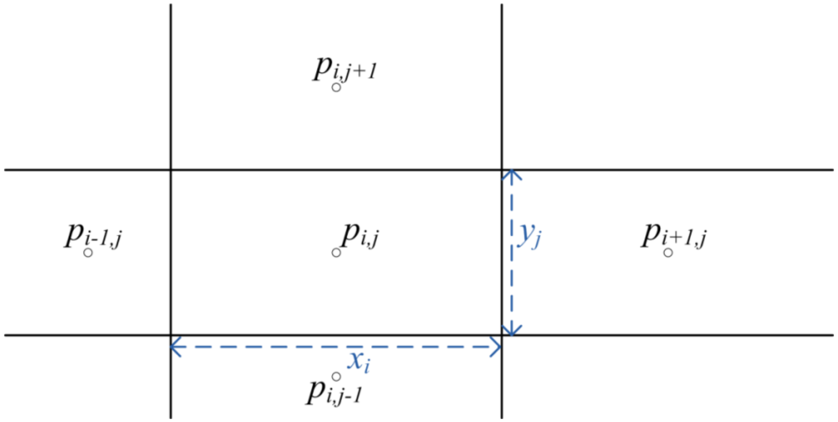


Solving Incompressible Flow: Projection

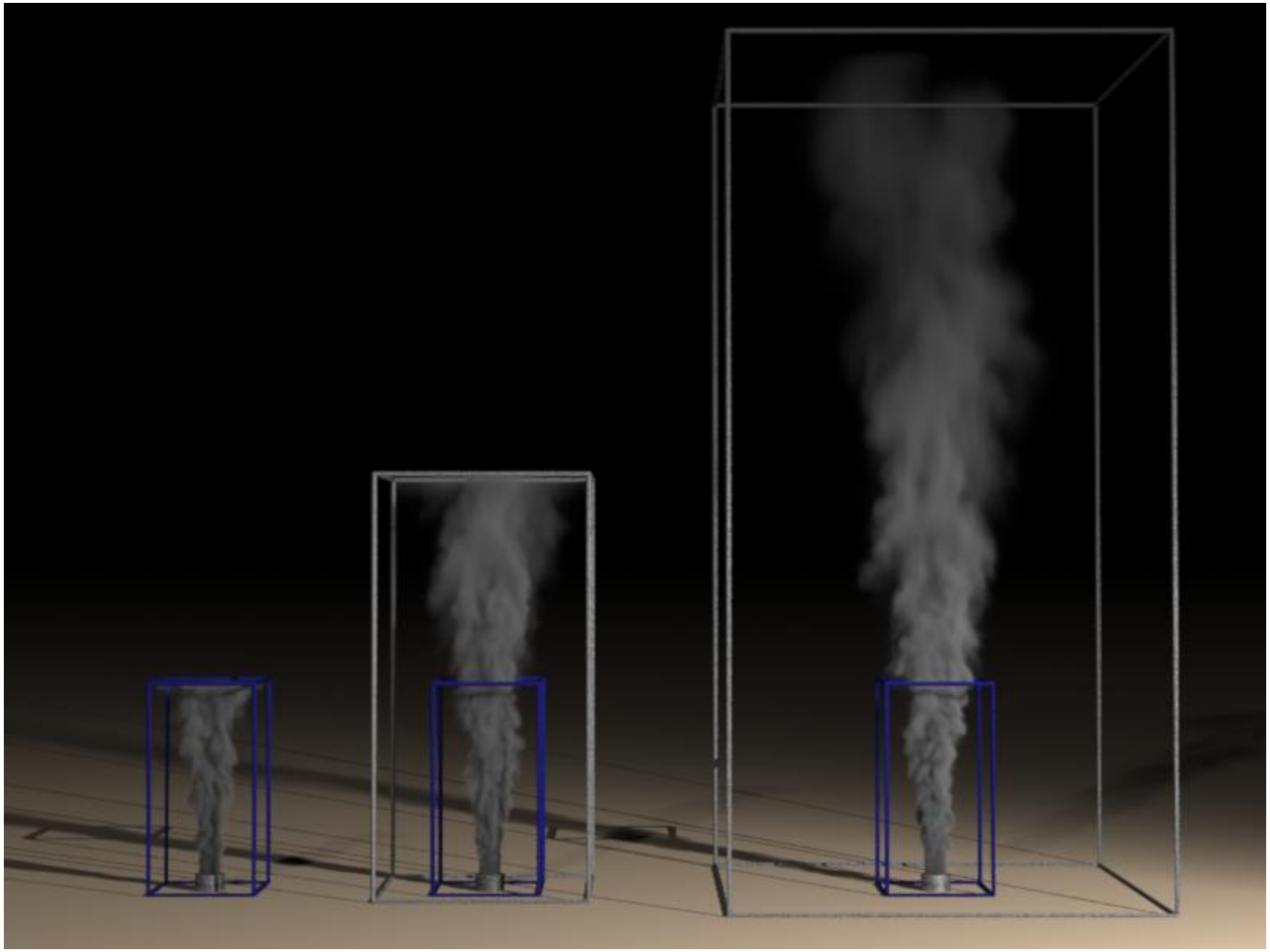
- Use the volume weighted divergence [Losasso et al. 2004] to solve the Poisson equation for pressure on stretched cells in order to obtain a SPD system

$$V_{\text{cell}} \nabla \cdot \left(\frac{\nabla \hat{p}}{\rho} \right) = V_{\text{cell}} \nabla \cdot \mathbf{u}^*$$

$$\sum_{\text{faces}} \frac{\nabla \hat{p}}{\rho} \cdot \mathbf{dA}_{\text{face}} = \sum_{\text{faces}} \mathbf{u}^*_{\text{face}} \cdot \mathbf{dA}_{\text{face}}$$

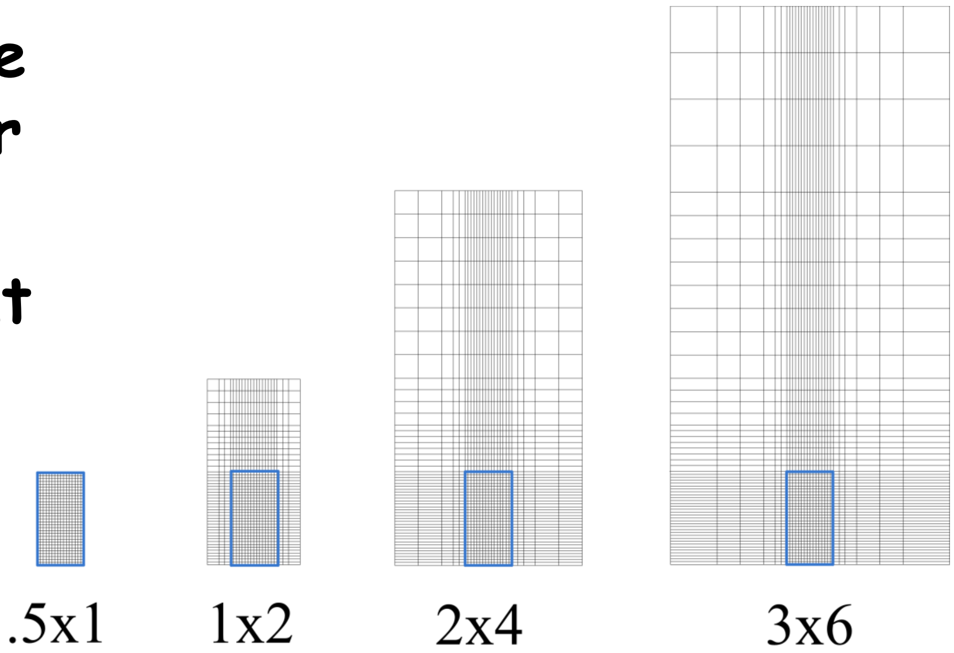


Domain: Domain Extension



Performance

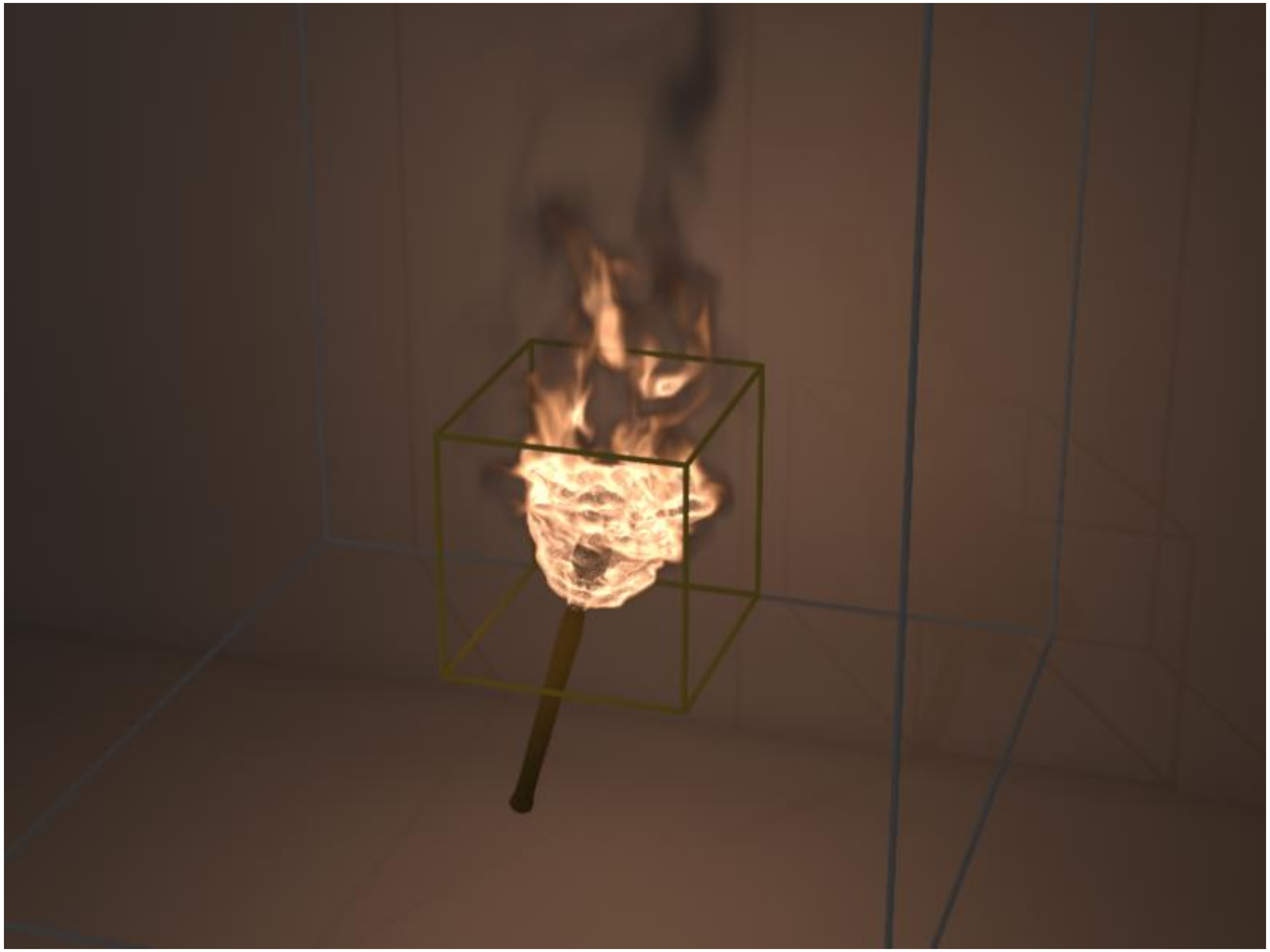
- Expand the domain to be 2, 4, and 6 times bigger
- Compare the DOF increase with the amount of time to run the simulation



Grid	Domain	Resolution	DOF Increase	Advection	Projection	Total
Uniform	$.5 \times 1 \times .5$	$128 \times 256 \times 128$	1.0	1.0	1.0	1.0
Far-field	$1 \times 2 \times 1$	$176 \times 352 \times 176$	2.6	2.4	3.2	3.1
Uniform	$1 \times 2 \times 1$	$256 \times 512 \times 256$	8.0	5.6	13	12
Far-field	$2 \times 4 \times 2$	$208 \times 416 \times 208$	4.3	3.8	6.4	6.2
Uniform	$2 \times 4 \times 2$	$512 \times 1024 \times 512$	64	37	173	160
Far-field	$3 \times 6 \times 3$	$224 \times 448 \times 224$	5.4	4.5	11	10
Uniform	$3 \times 6 \times 3$	$768 \times 1536 \times 768$	216	-	-	-

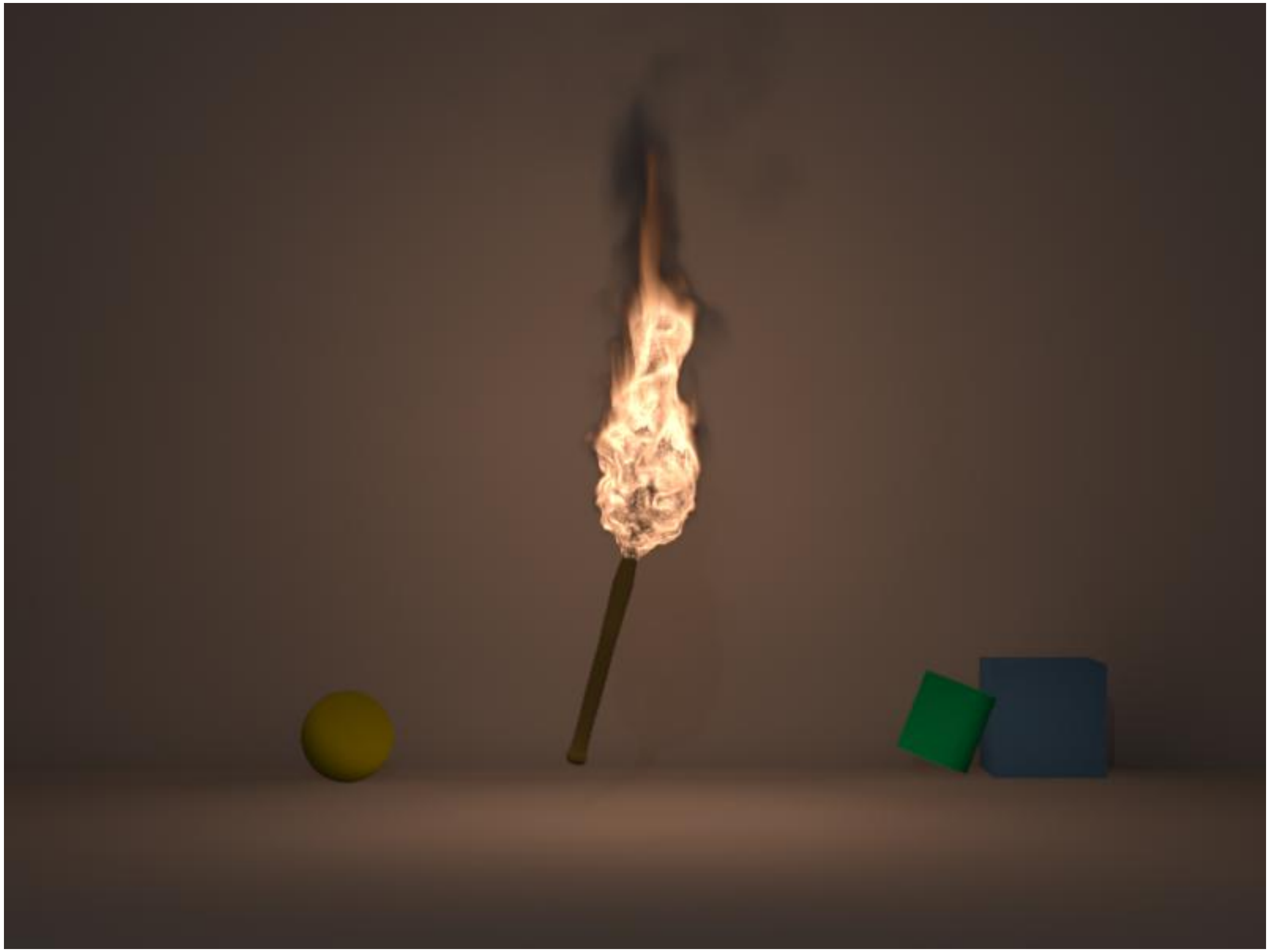
Example: Tracking the Flame Interface

Example: Tracking the Flame Interface

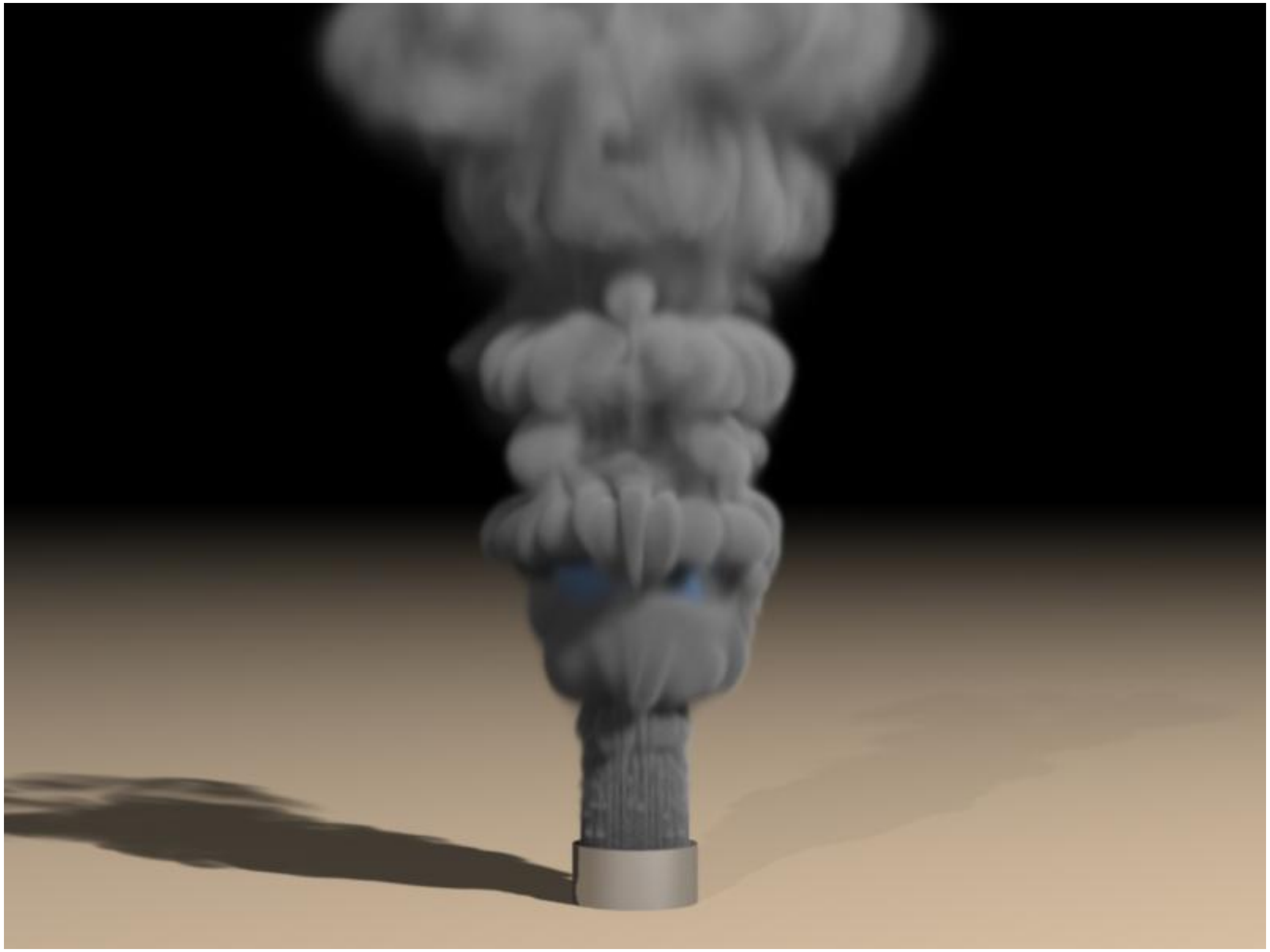


SIGGRAPH2013

Example: Tracking the Flame Interface

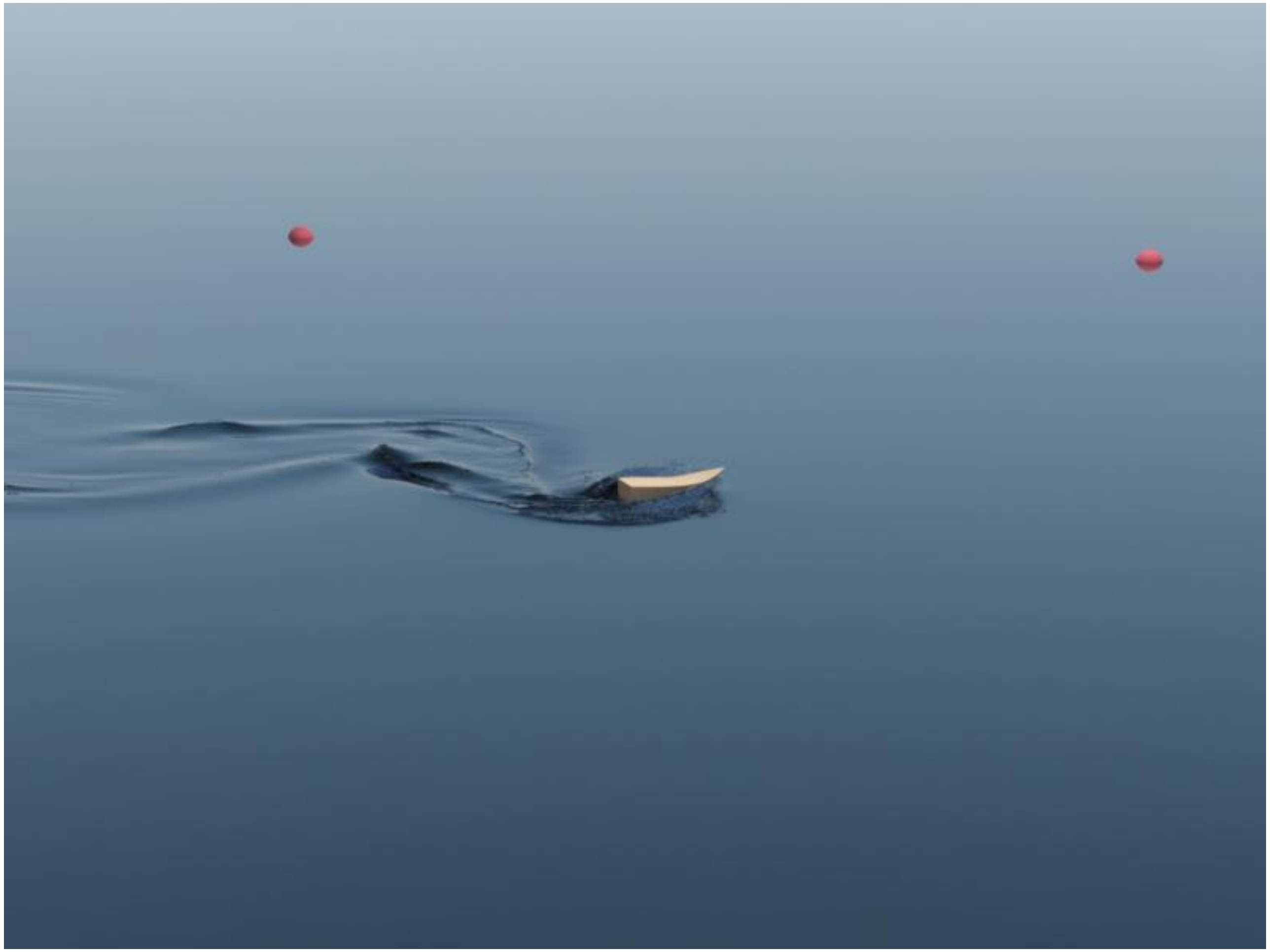


Example: Tracking the Kinematic Object



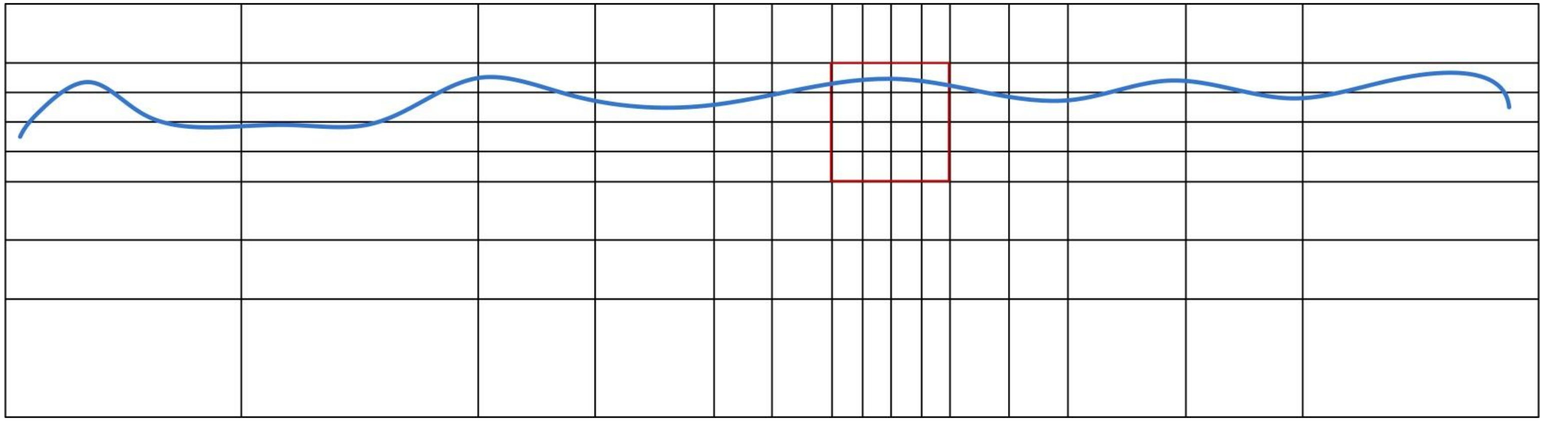
Example: Tracking the Kinematic Object

Example: Tracking the Kinematic Object

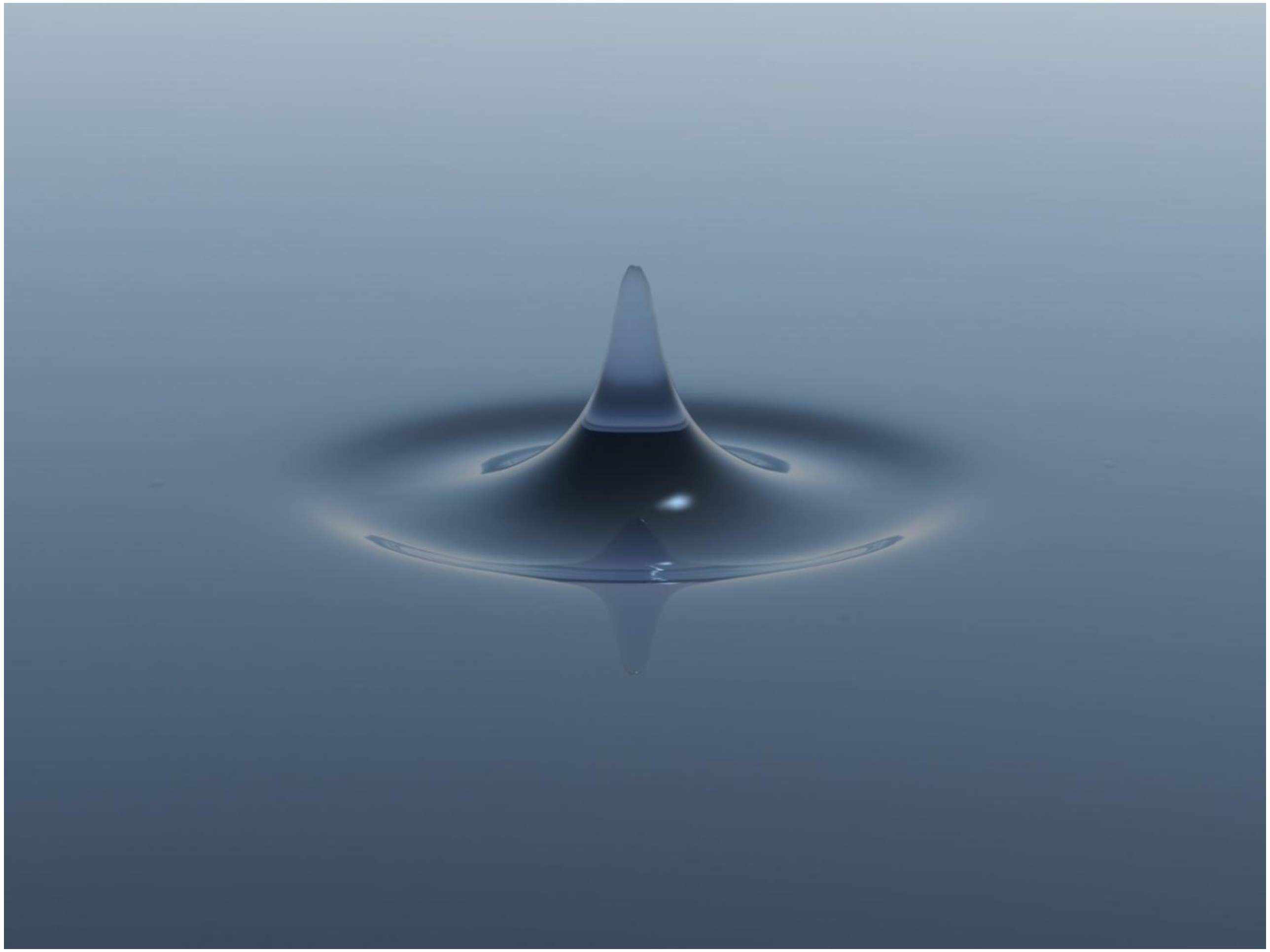


Non-Reflecting Waves

- **Use the far-field grid structure to cover a large domain**
  - Keep enough number of fine cells in the region of interest
  - Use coarse cells for waves moving far away

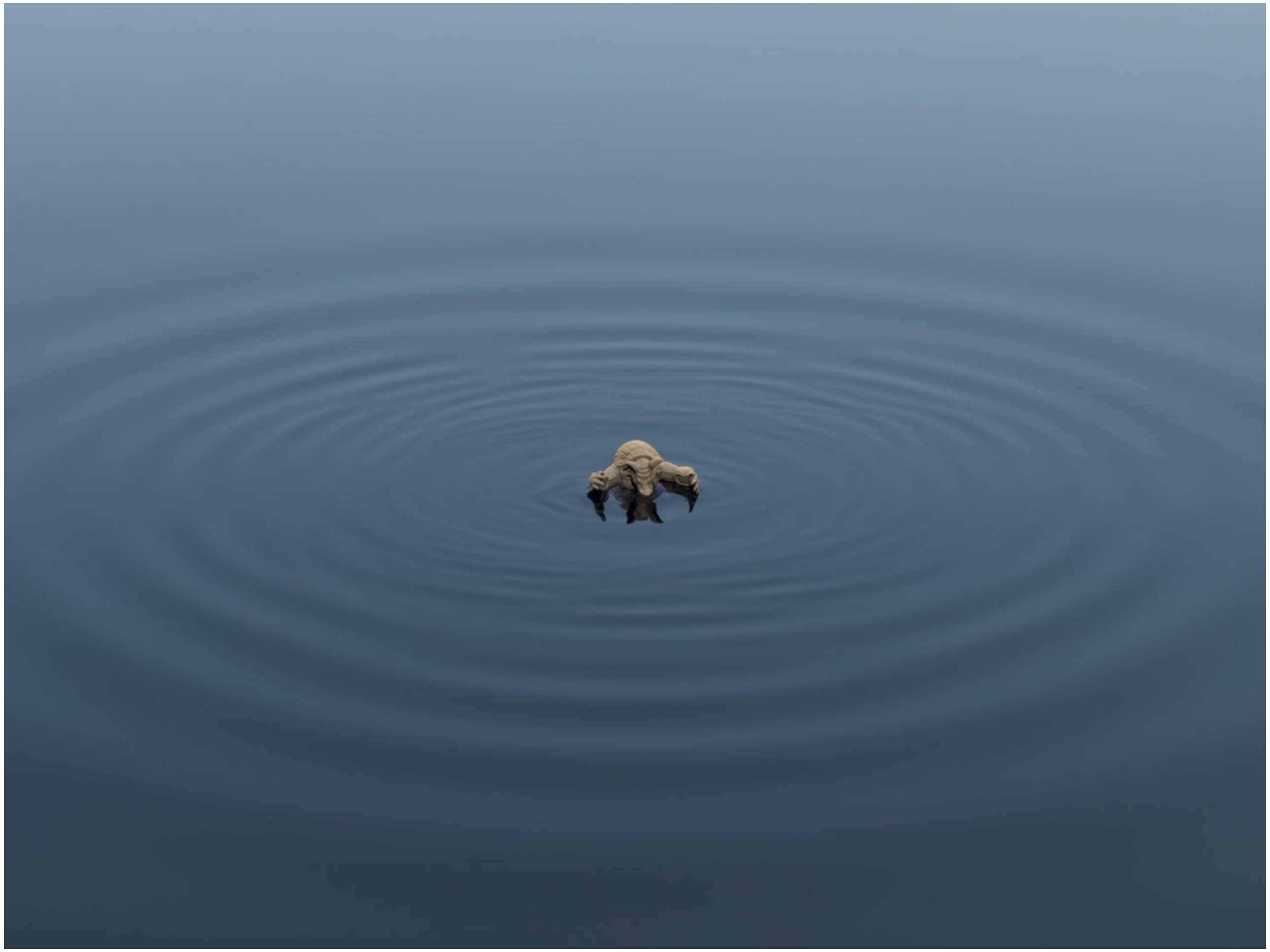


Example: Non-reflective Waves



Example: Non-reflective Waves

Example: Non-reflective Waves



Limitations

- **Simultaneously tracking multiple regions on a single far-field grid**
  - The number of fine regions is determined by the Cartesian product of the fine regions along each axis.
- **Visual artifacts due to the stretched cells.**
  - More suitable to modeling localized fluid phenomena
  - Such as fire and water as opposed to smoke

Future Work

- **Real-time applications**
  - The structured data storage and the one dimensional arrays for fast cell accessing can be easily adapted to the GPU.

Conclusion

- An efficient grid structure for domain extension
- Adaptively allocating grid cells to important regions
- Preserves almost every computational advantage of uniform grids
- Easy to implement

Thank you!