Computational geometry of soft matter

UMass Summer School on Soft Solids and Complex Fluids 2024

Lecture 4 (Thursday June 6)

Chris H. Rycroft, University of Wisconsin–Madison
(chr@math.wisc.edu)

Outline

Monday

- A model of dense granular drainage
- Voronoi analysis of granular flow
- Neighbor relations

Tuesday

- Development of the Voro++ library
- Network analysis for CO_2 capture
- Alternative models and methods

Wednesday

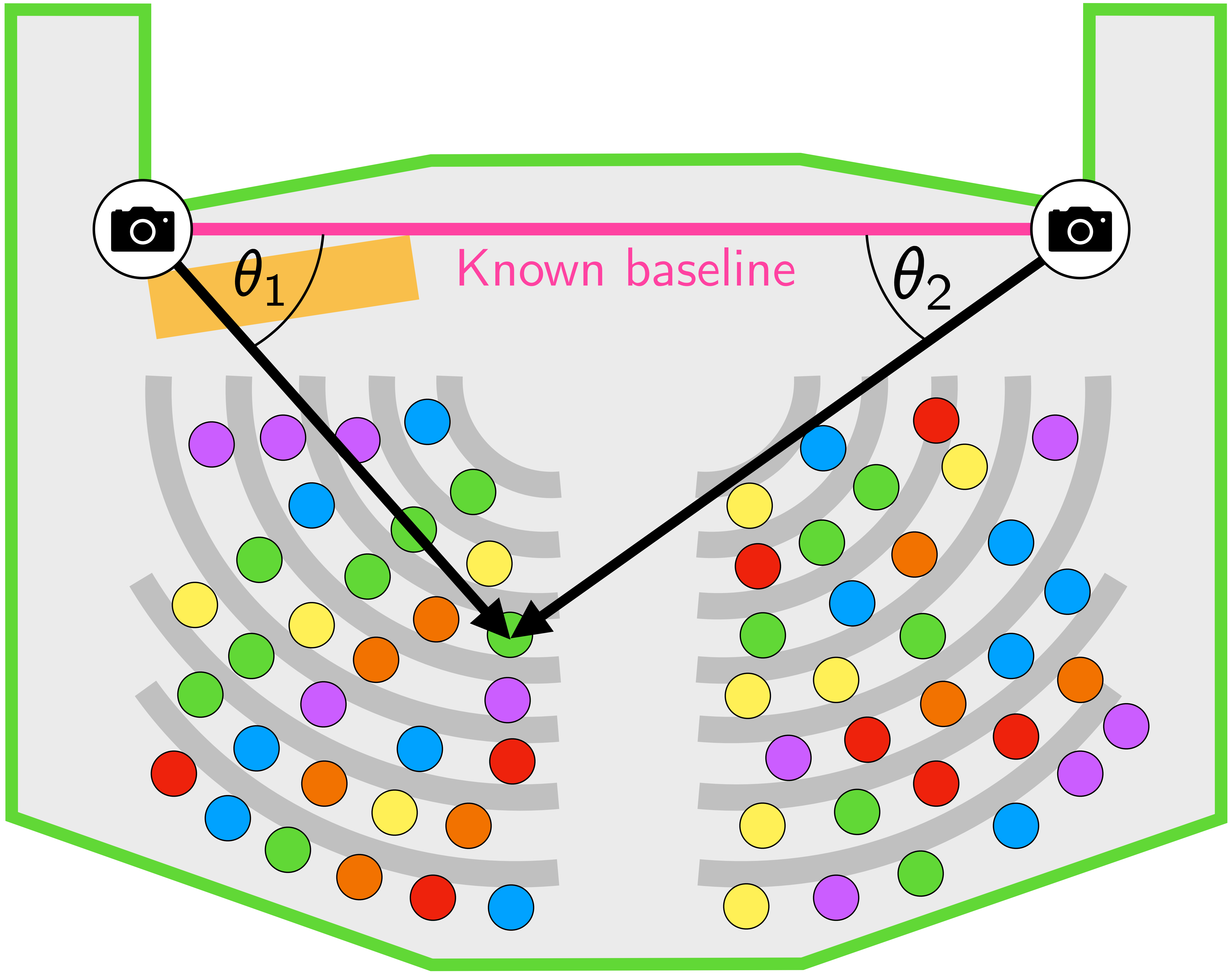
- Topological Voronoi analysis
- Lloyd's algorithm and meshing
- Insect wing structure

Thursday

- Continuum representations of deformation
- The reference map technique
- Fluid–structure interaction

Neighbor relations and triangulation

- In lecture 1 we did an exercise to determine neighbor relationships using two approaches:
  - an intuitive understanding of neighbors
  - neighbors defined via Voronoi cells

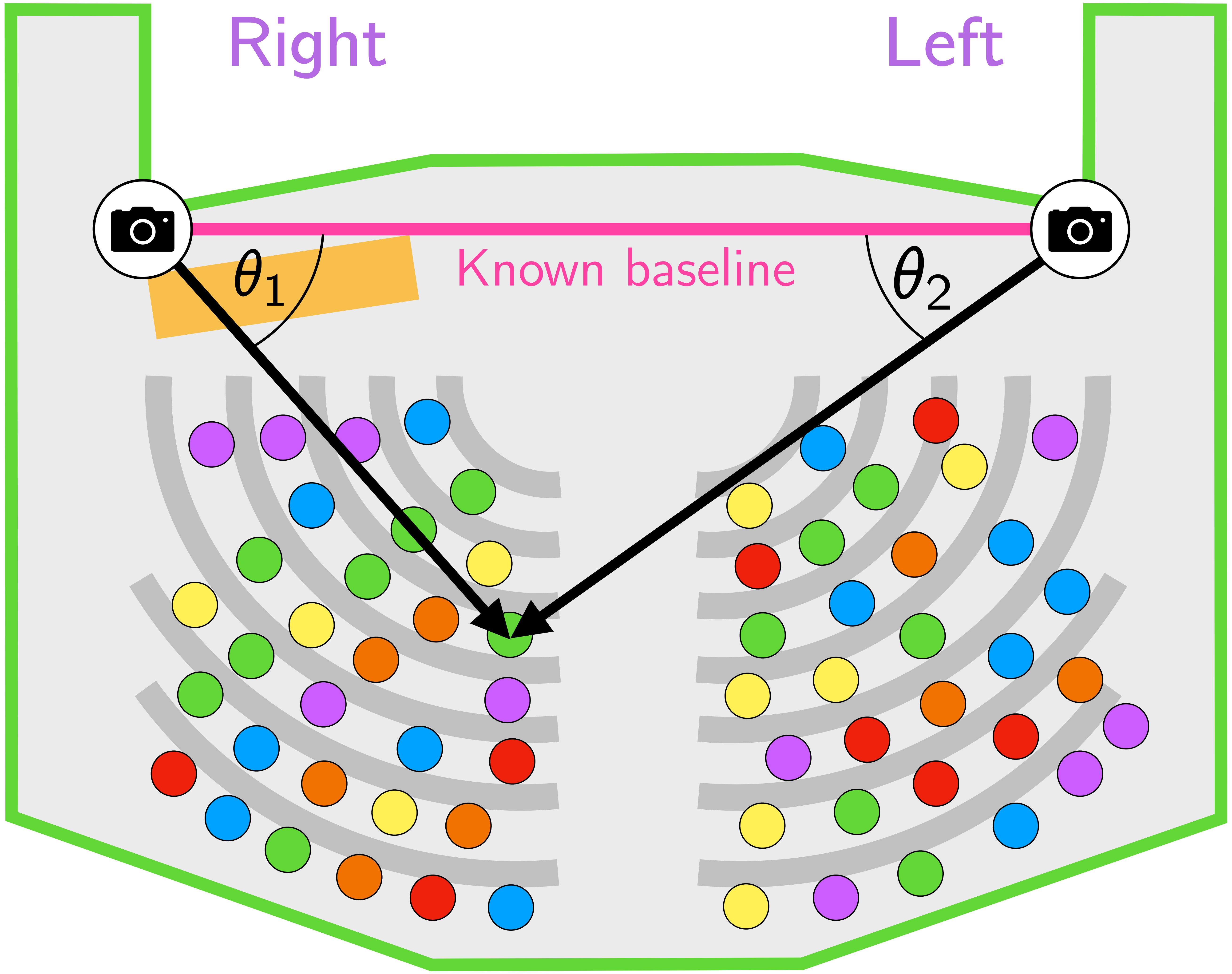


The lecture room

(Left/right defined from speaker's perspective)

Neighbor relations and triangulation

- I took pictures of everyone from either side of the lecture room
- Knowing the room geometry, this is enough to reconstruct everyone position
- Four pictures taken:
  - Camera on left/right of lecture room
  - Camera pointing toward left/ right seating



The lecture room

Left seating from left side

https://flic.kr/p/2q36tyK

Right seating from left side

https://flic.kr/p/2q35A8M

Right seating from right side

Left seating from right side

https://flic.kr/p/2q36ttu

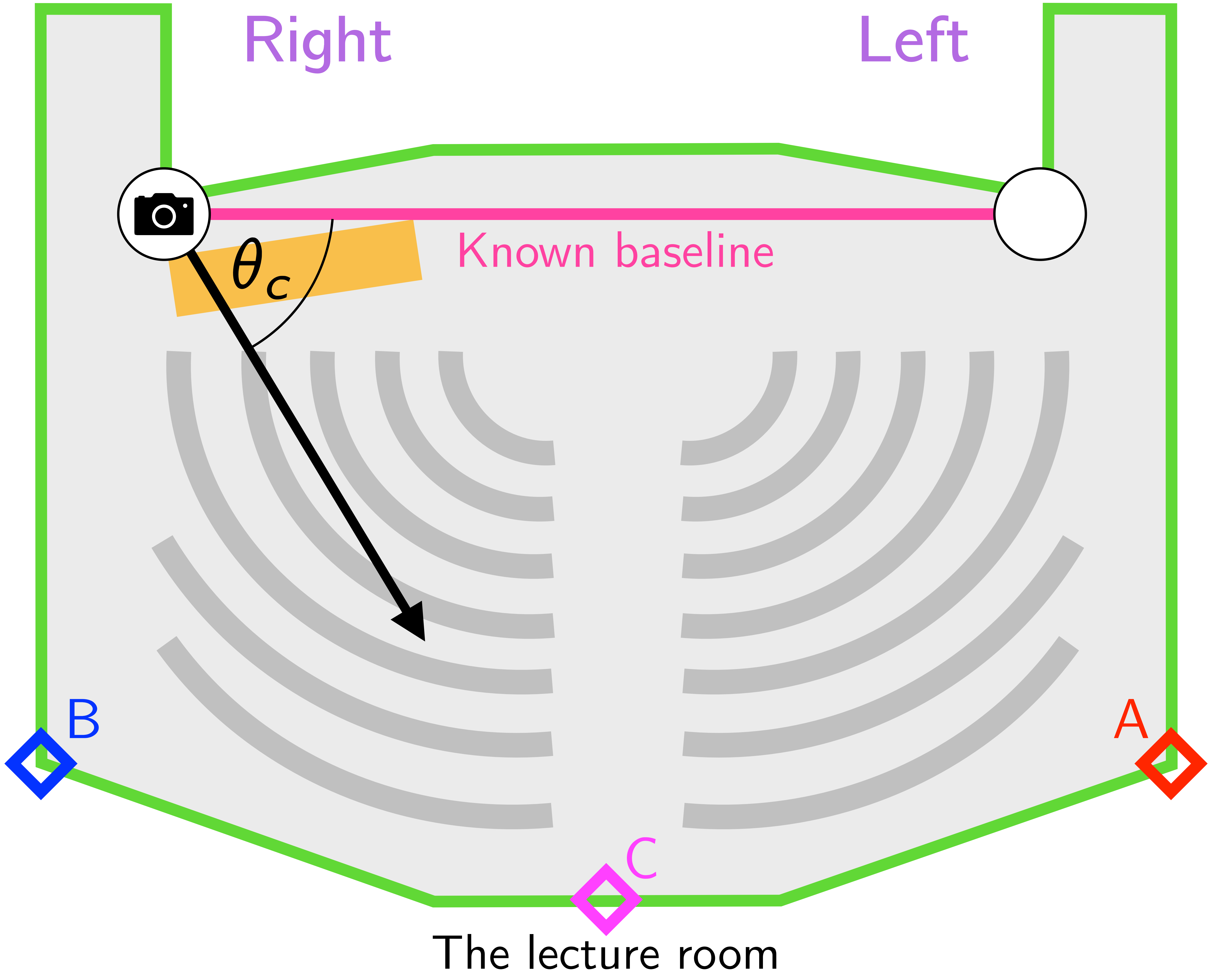
Left seating from right side

Go through each image and measure the horizontal position of each circle in pixels

(Left/right defined from speaker's perspective)

Determining camera orientation

- Three points are used to find the precise camera angle θ_c and field of view
- They are A and B (room corners) and C (clock)
- At least two target points are visible in each photo, which is sufficient



Data collection

- Program files for performing the triangulation are in the **triangulate** directory of the Git repository
- The file **ang.dat** contains the horizontal pixel positions of each circle in each image
- The file has four sections labeled with headers IM0, ..., IM3 for each image
- The header line contains the pixel positions of the three calibration targets

```
IM2 4944 7628 -
6 2277
69 2675
68 2935
65 3238
1 3287
70 3550
67 3741
2 3752
64 3920
60 3950
66 4067
3 4258
59 4407
58 4465
57 4600
4 4724
49 4747
63 4881
56 4937
61 4937
...
```

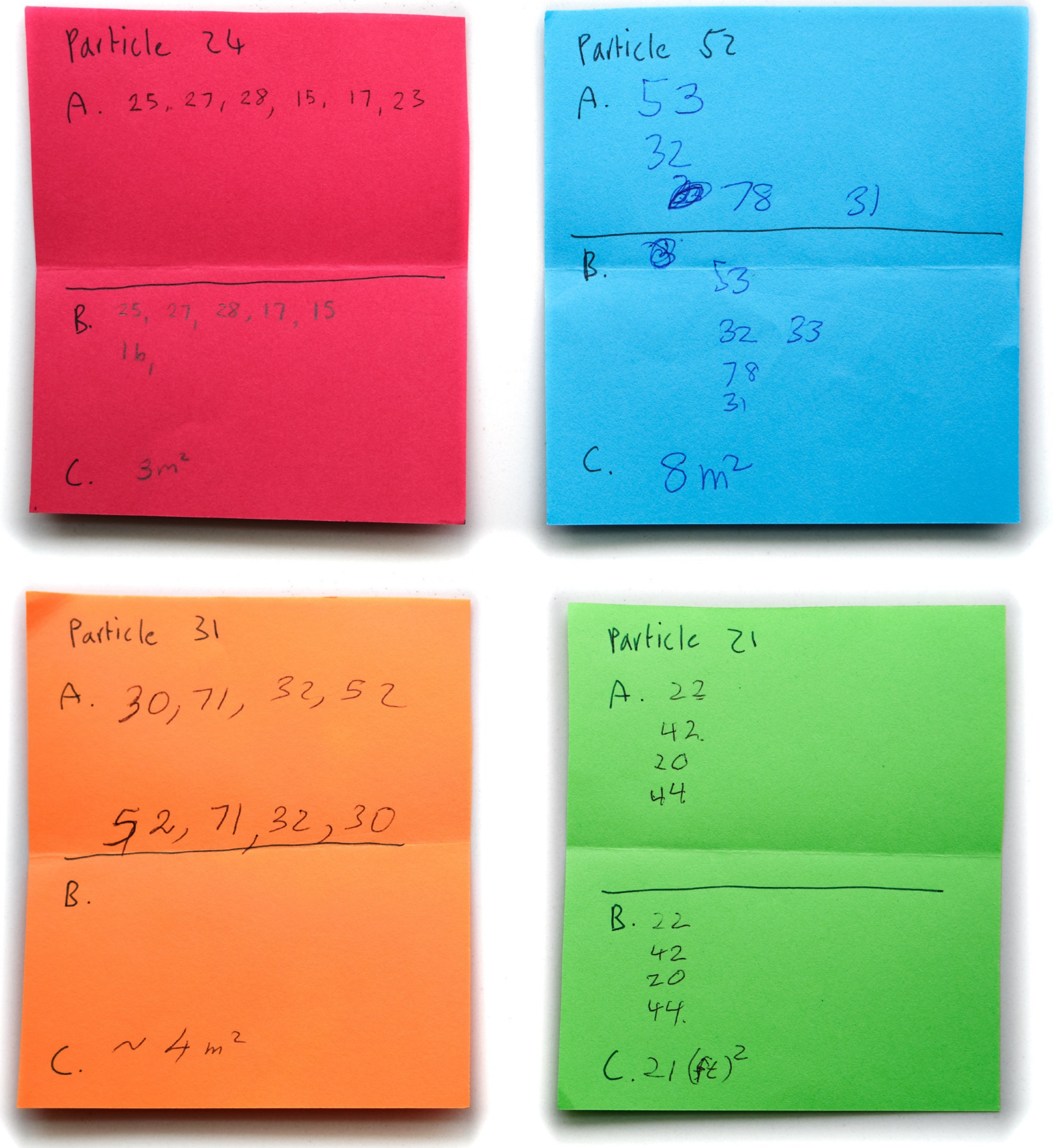
Targets A and C located at 4944 and 7628, respectively. Target B is not visible.

Circle 57 located at pixel position 4600

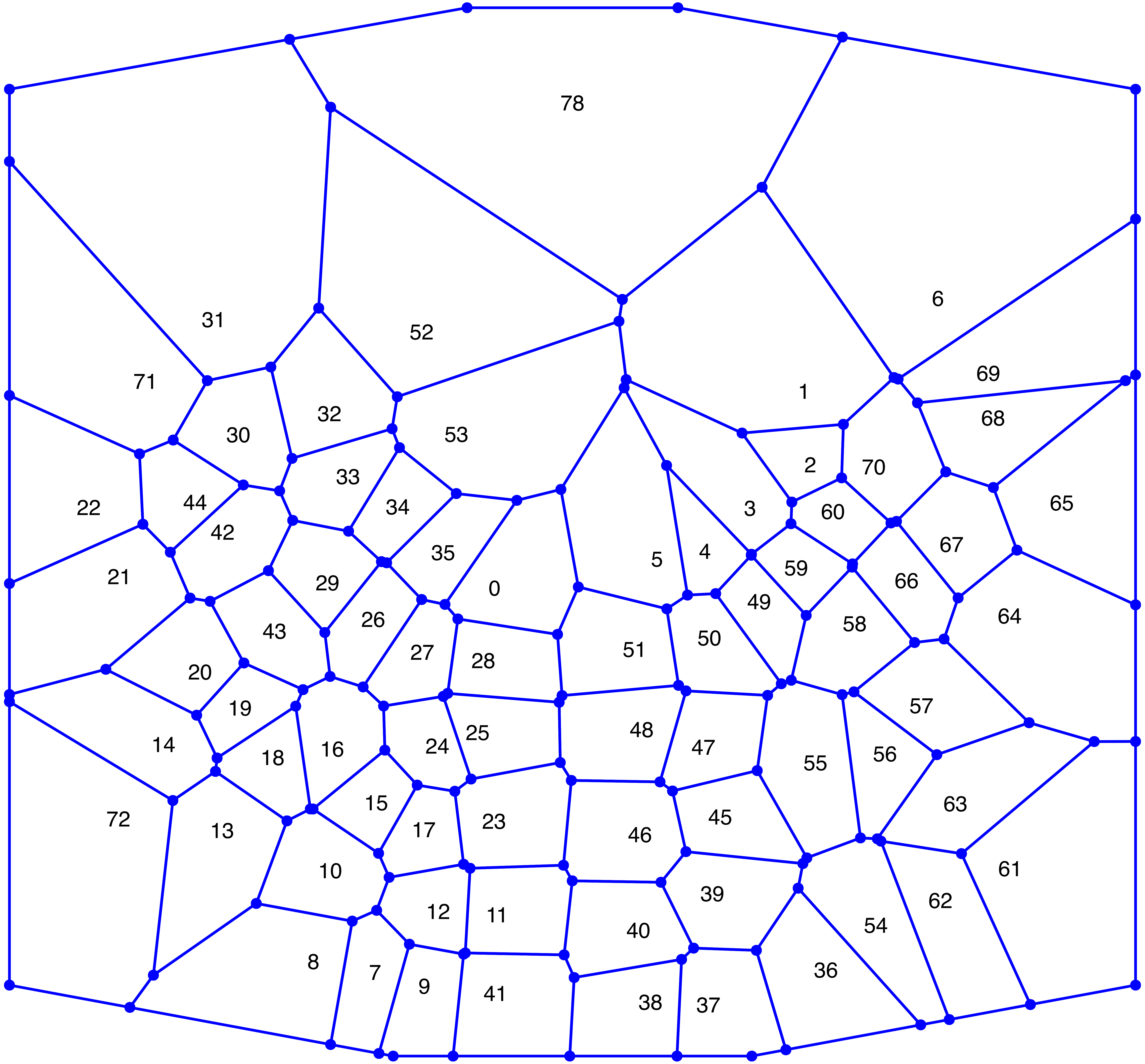
Excerpt of **ang.dat**

Neighbor data

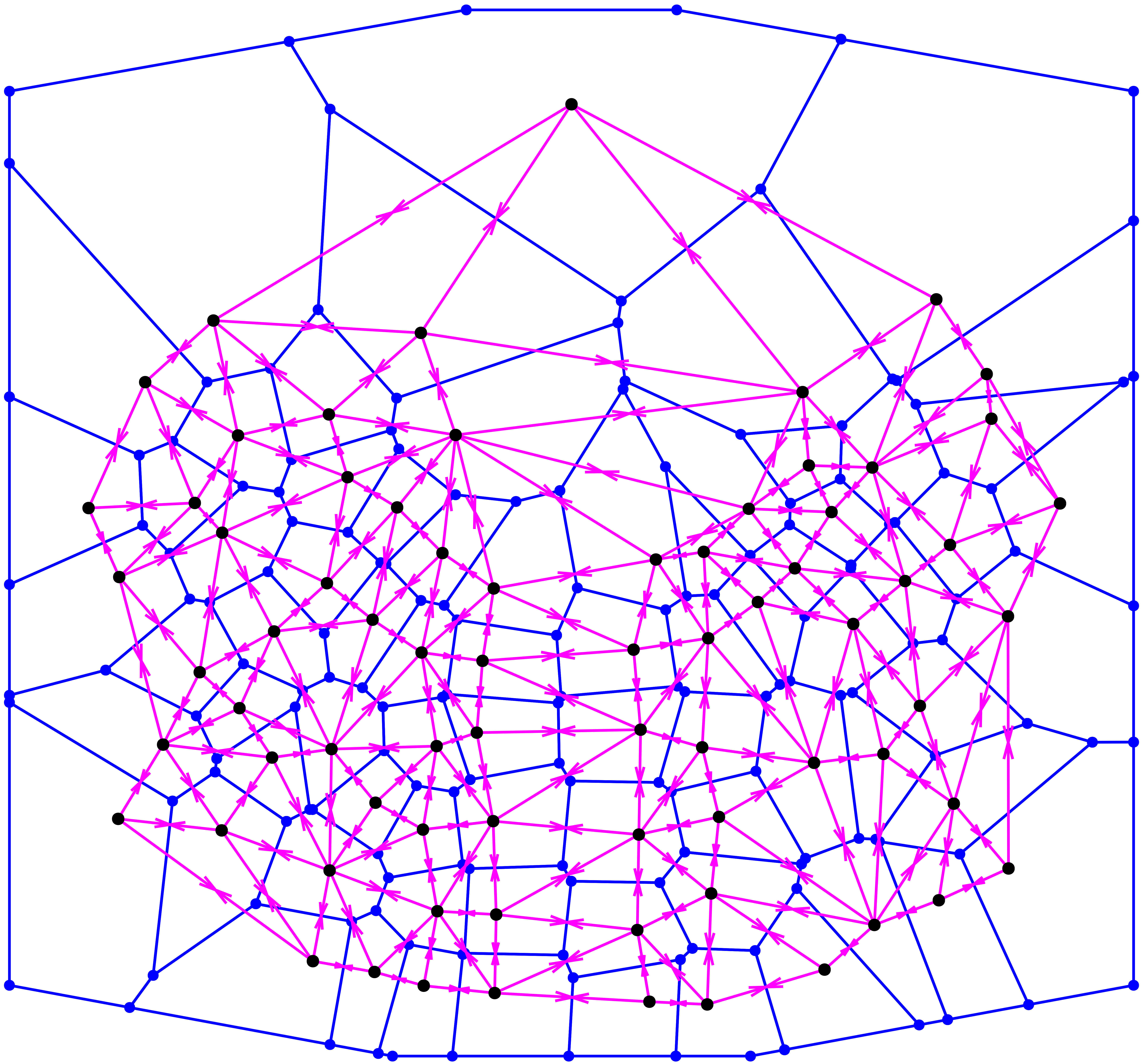
- Each participant submitted a note with
  A. An intuitive list of neighbors
  B. A list of neighbors based on Voronoi cell adjacency
  C. An estimate of the Voronoi cell area
- This data was entered into a file called **nei.dat**
- See the **triangulate** directory in the GitHub repository for more information on analysis



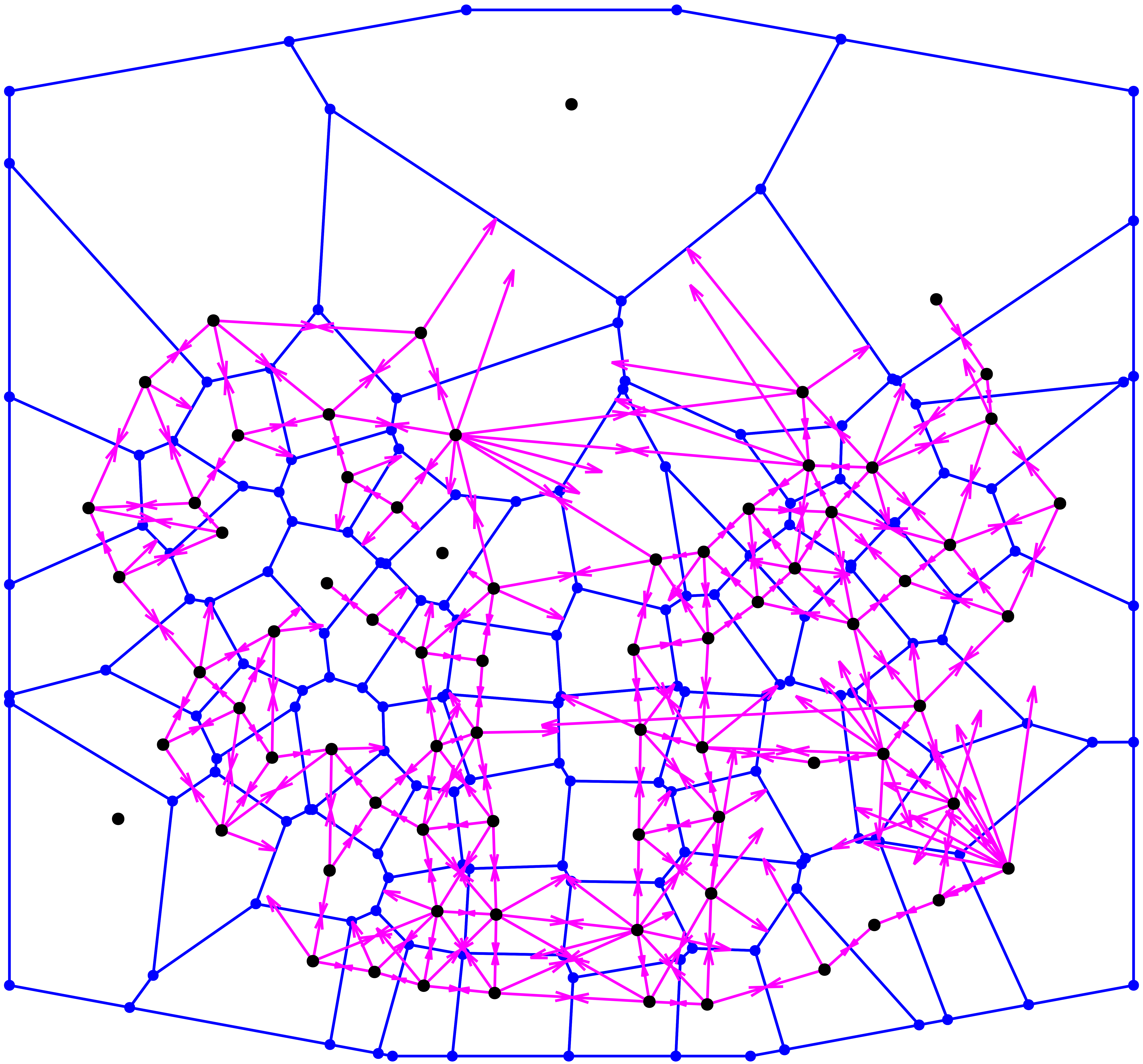
Voronoi tessellation



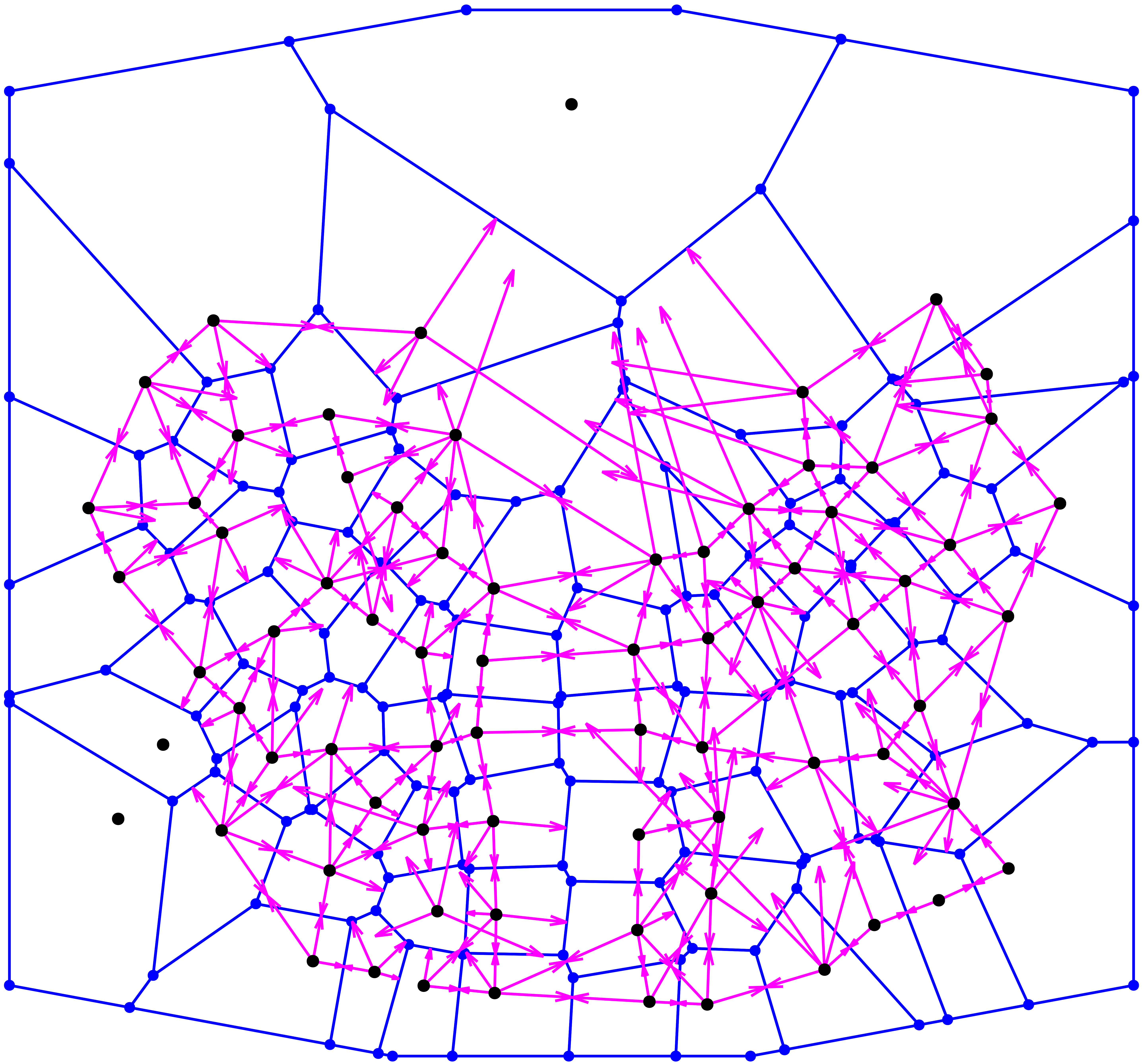
Delaunay triangulation



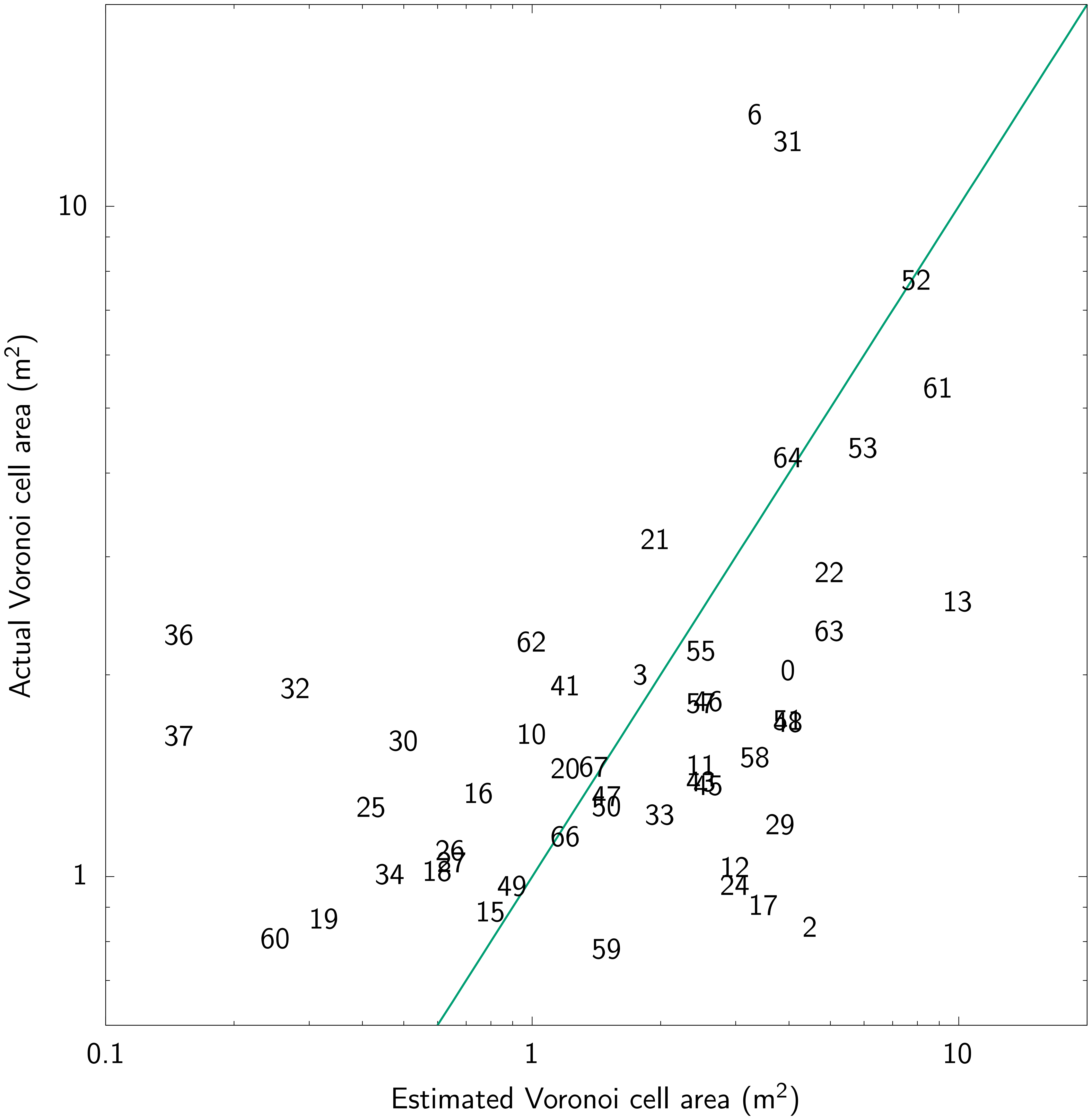
Initial neighbor relation data (A)



Voronoi neighbor relation data (B)



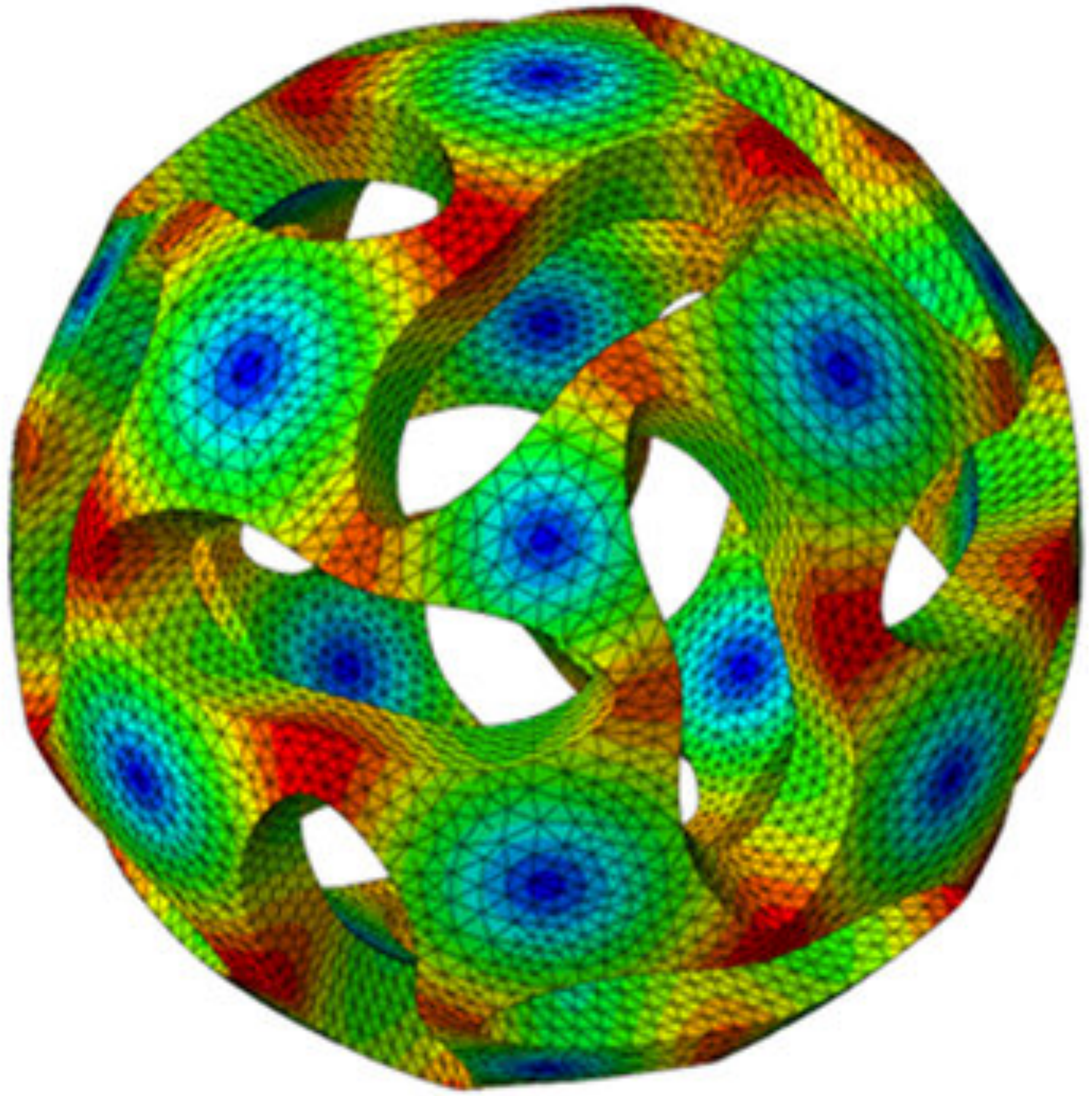
Voronoi area estimation vs. data



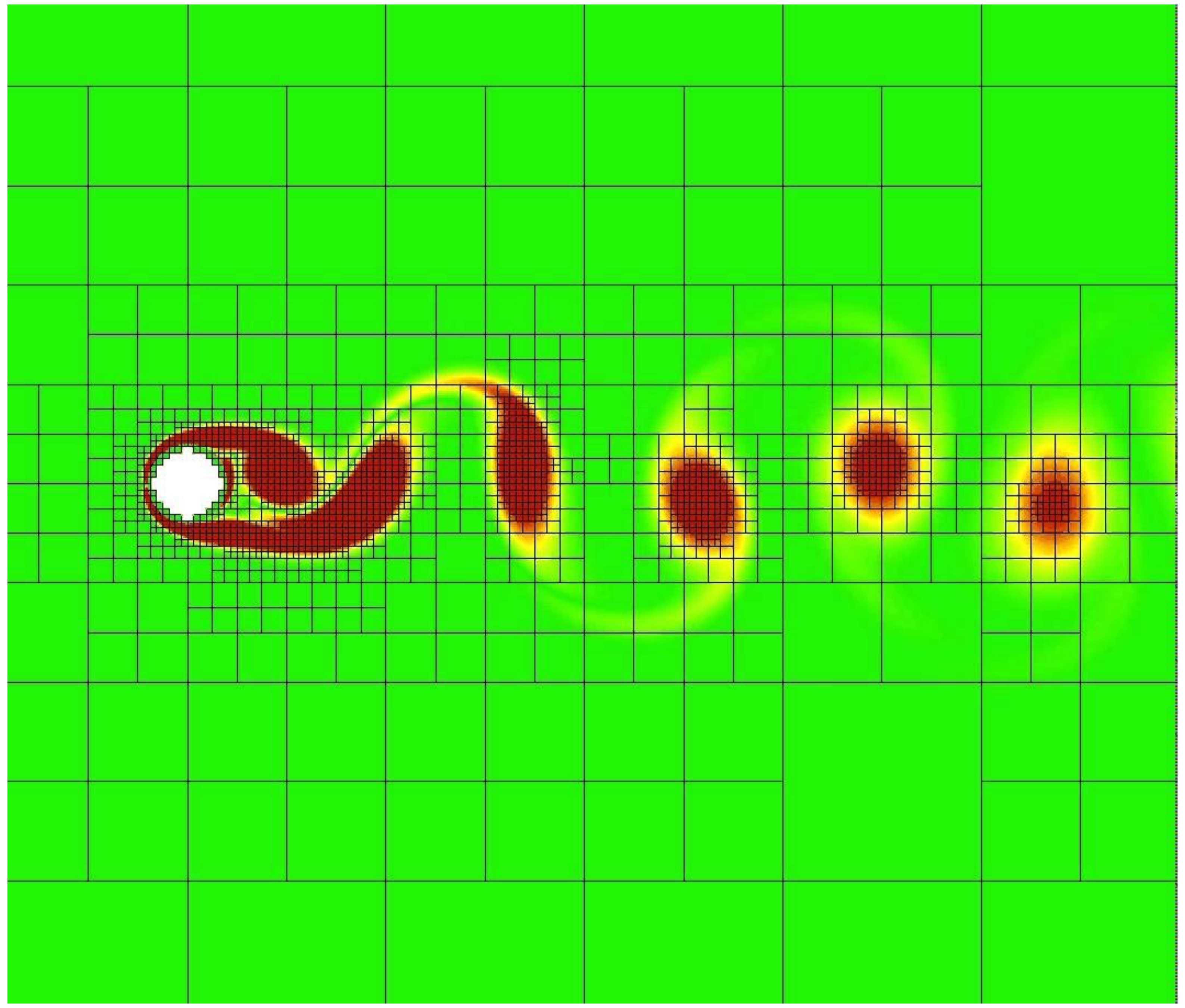
Fluid–structure interaction and the reference map technique

Joint work with Ken Kamrin, Boris Valkov, Chen-Hung Wu, Yue Yu,
Luna Lin, Nicholas Derr, Dan Fortunato, Xiaolin Wang, and Yue Sun

Two different representations



Coupling?



Solid simulation

Lagrangian approach: grid moves with the material

(Natural for finite-strain elasticity)

Fluid simulation

Eulerian approach: fluid velocity represented on a fixed grid

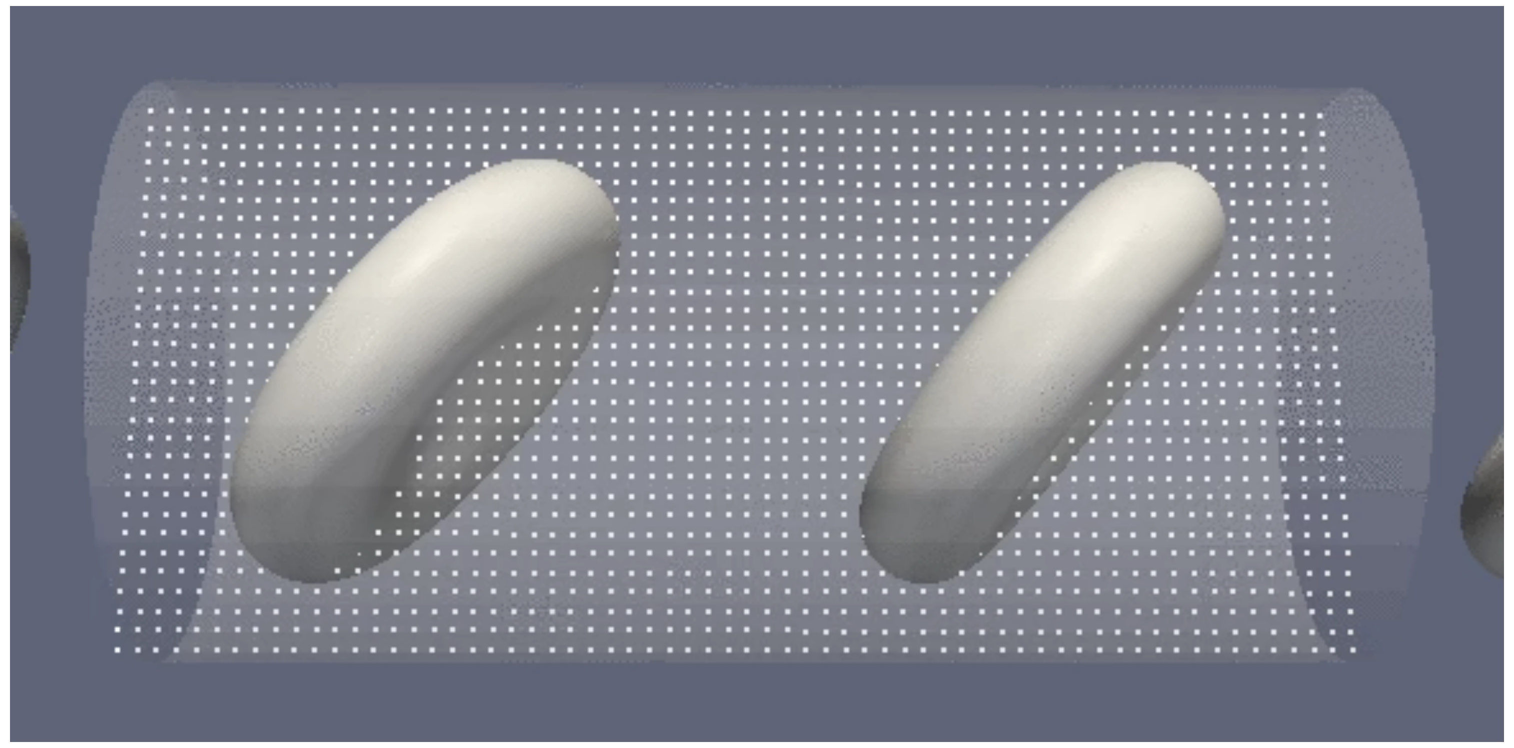
(Natural for the Navier–Stokes equations)

Lin *et al.*, Scientific Reports **5**, 11309 (2015)

Fluid–structure coupling approaches

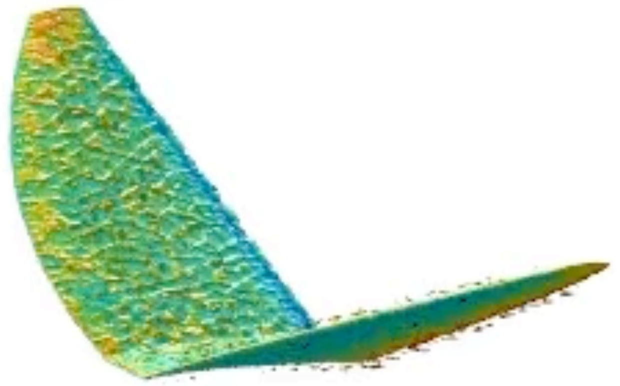
1. Immersed boundary method

Lagrangian solid and Eulerian fluid are coupled using interpolation techniques



2. Conforming meshes

Fluid grid constantly deforms to match the moving solid surface



3. Fully Lagrangian approach

Fluid represented via particles that represent small volume elements

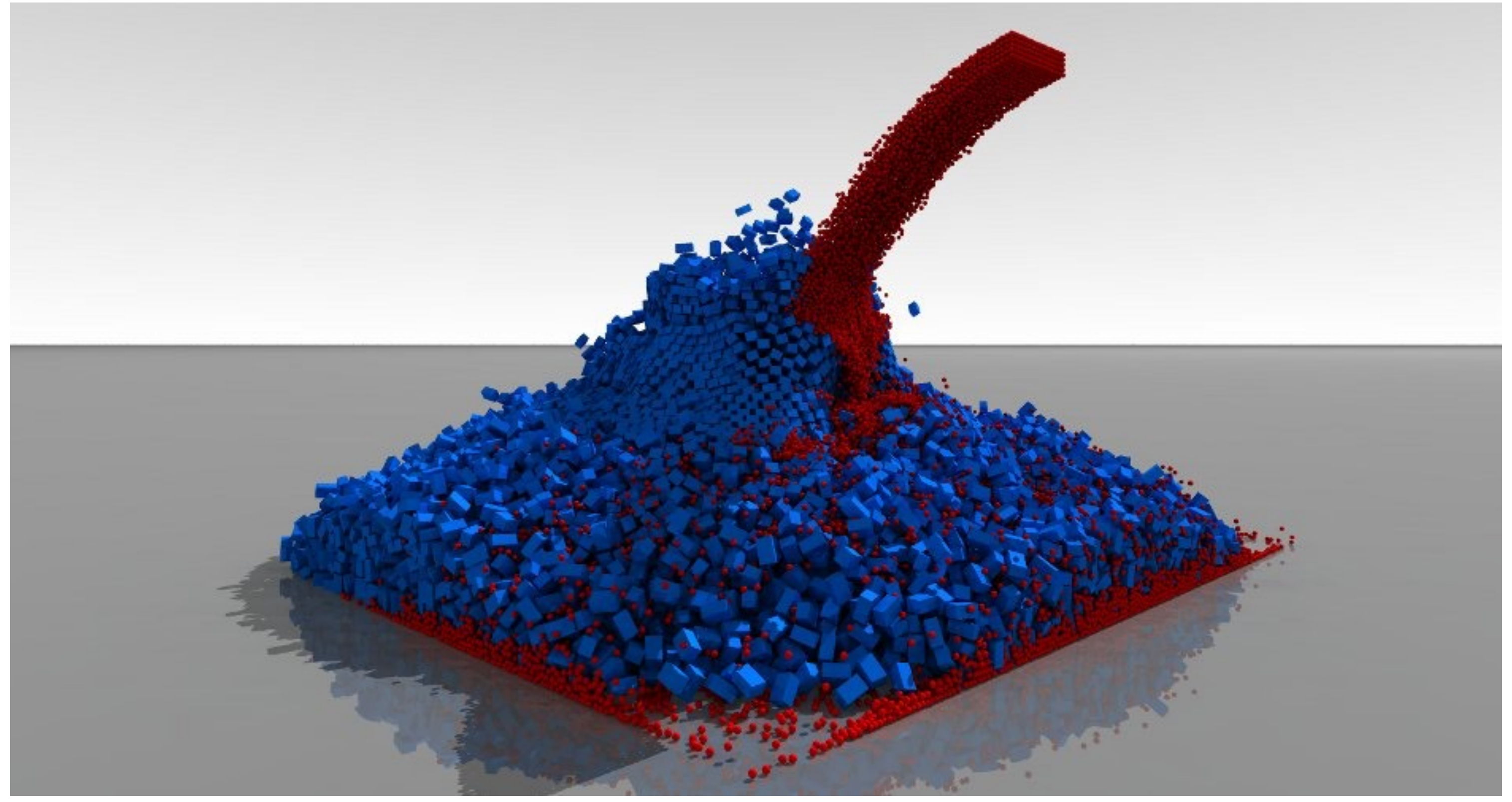
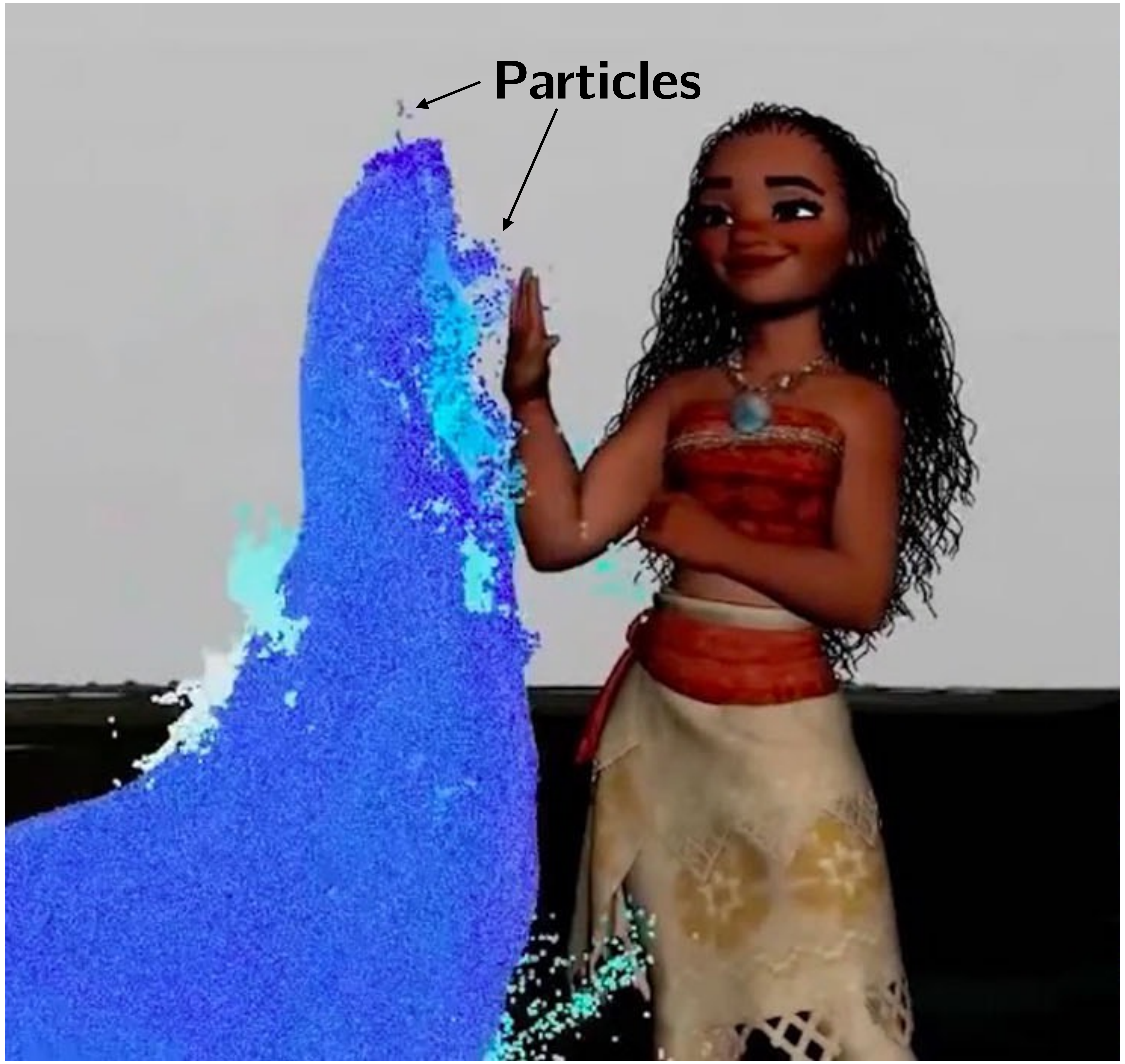


Image credits: (1) Thomas Fai and Charles Peskin, (2) Per-Olof Persson, (3) Dan Negrut

Particle-based methods: well-suited for computer graphics and rendering

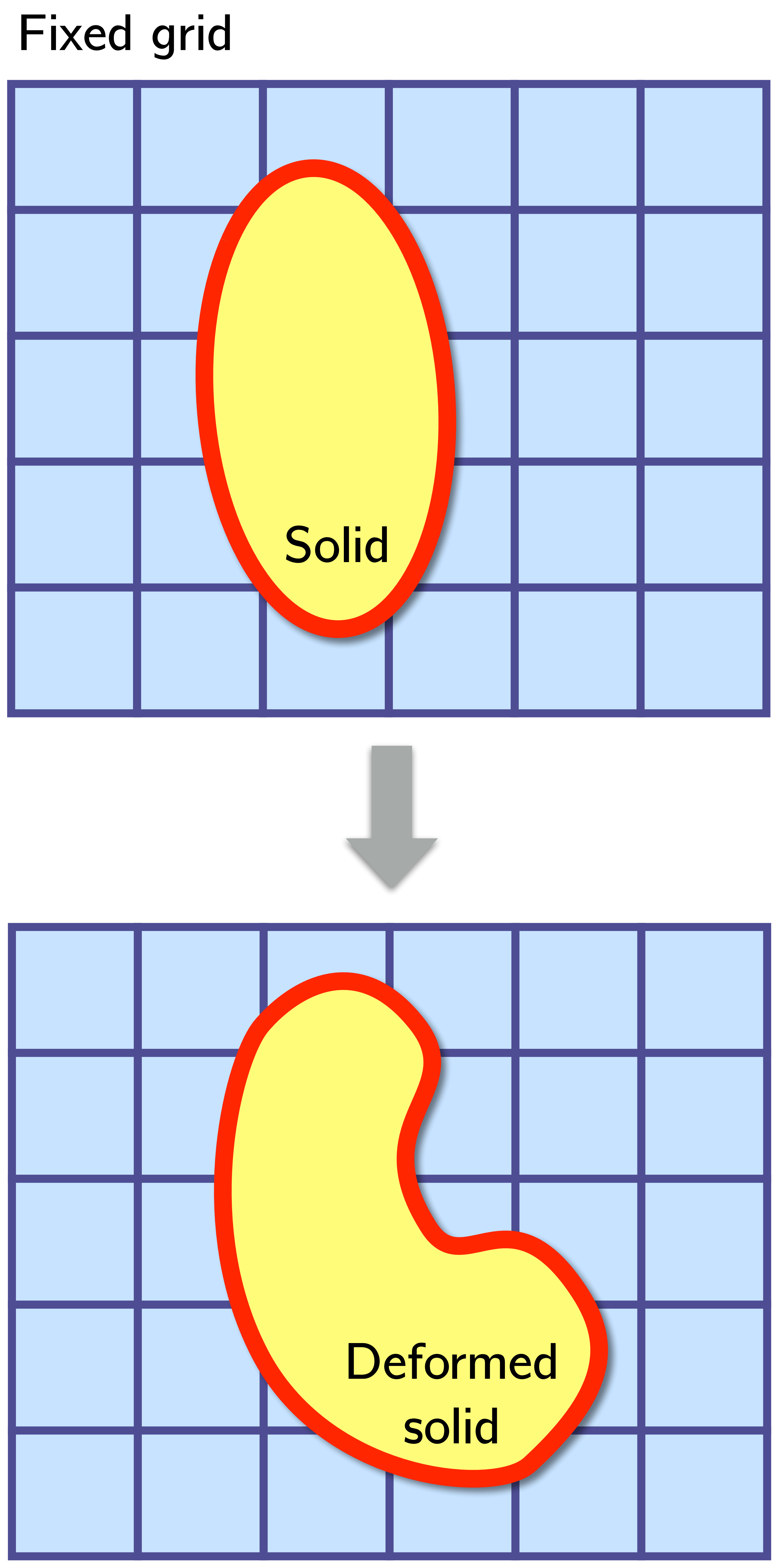


From *Moana*, Walt Disney Animation Studios (2016)

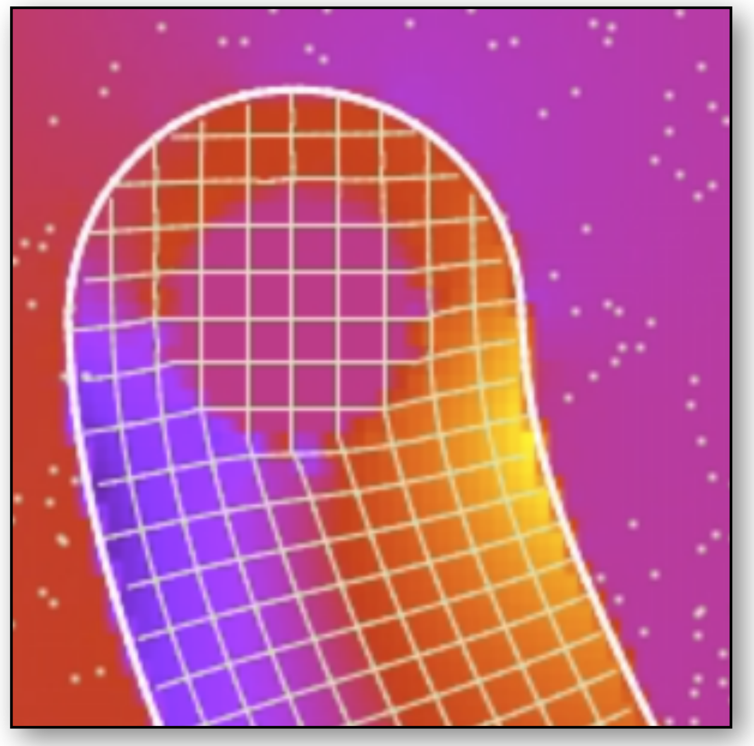
Aim of the talk

- What about a fully Eulerian method? Possible advantages:
  - Eulerian grids are simple and efficient to compute on
  - Easier coupling to other physical processes
  - Avoid meshing difficulties for complex geometrical interactions
  - Many theoretical results for convergence & stability

Key challenge: how can large solid deformations be represented?

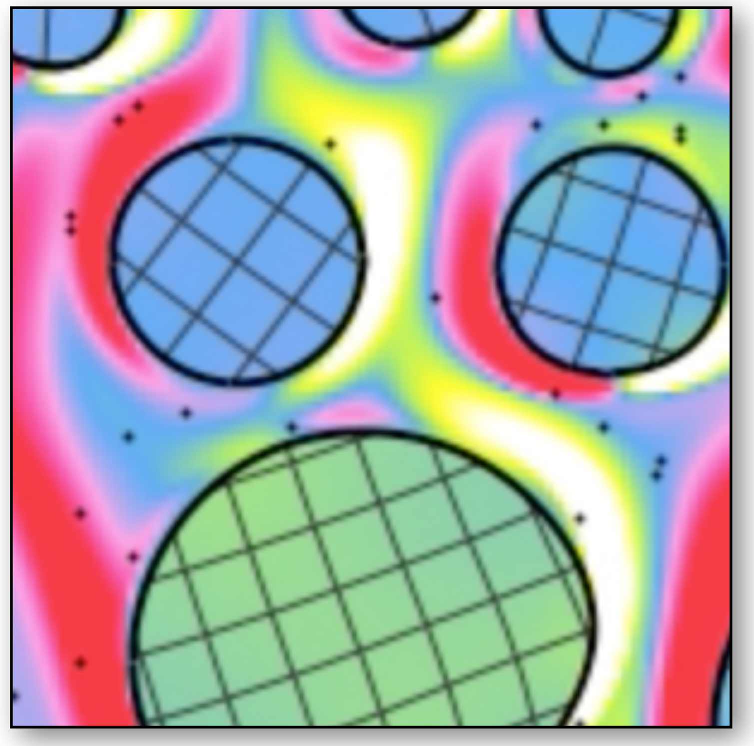


Talk outline



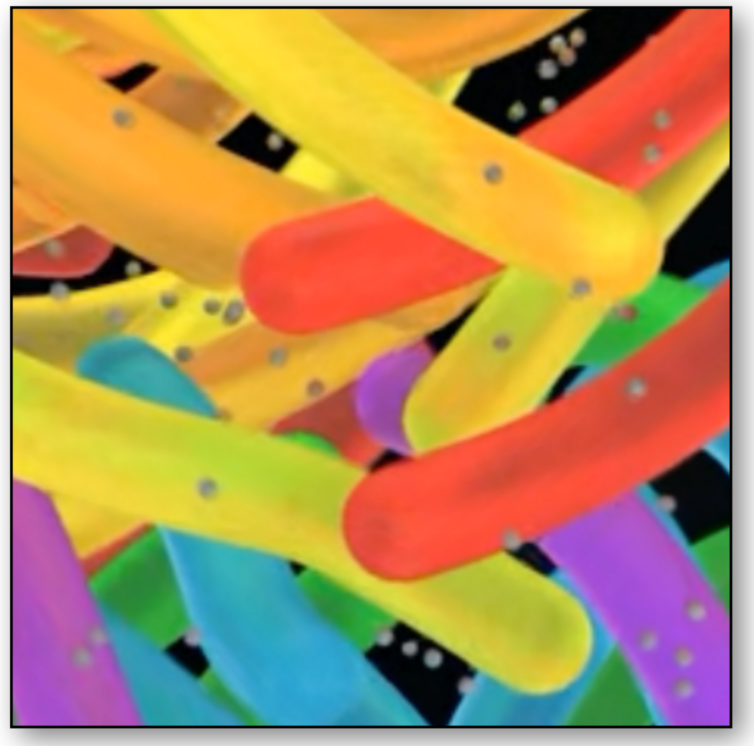
1. The reference map technique

A conceptual overview of the method



2. Reference map simulation for incompressible fluids

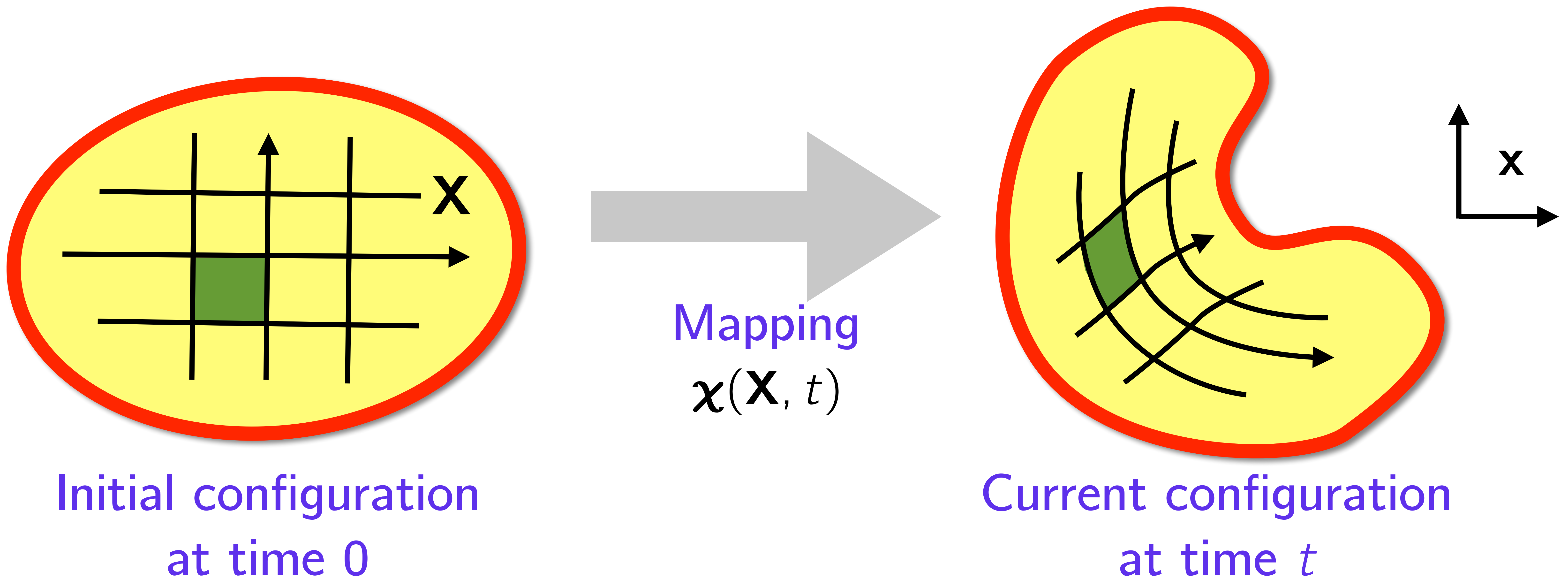
A numerical implementation for simulating flag-flapping and multi-body contact



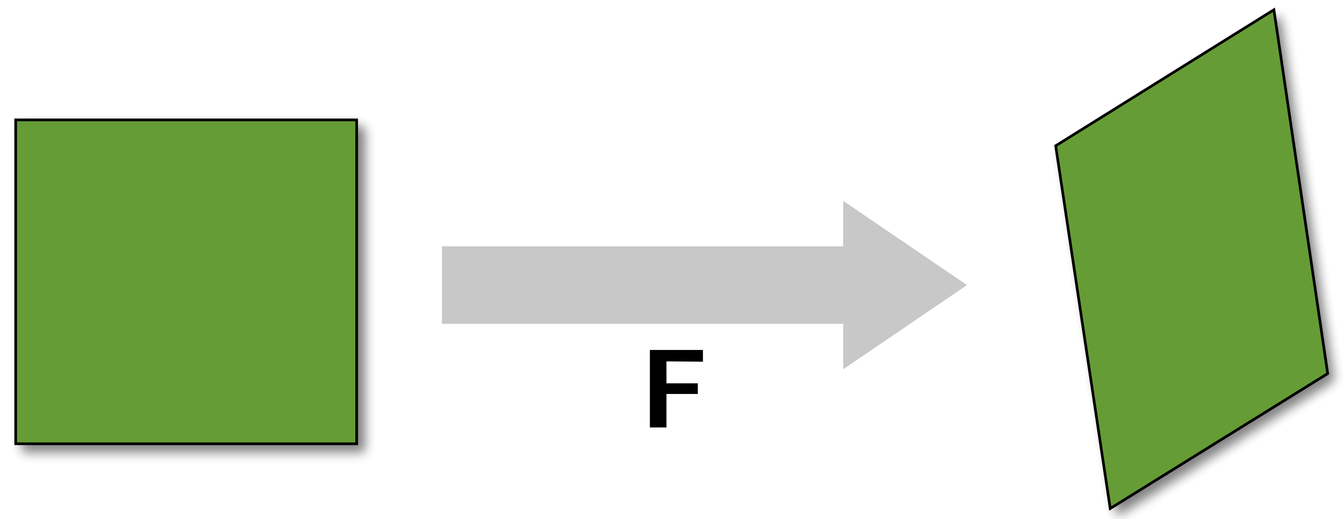
3. Applications and extensions

Three-dimensional simulation, higher-order discretizations

Finite strain elasticity theory

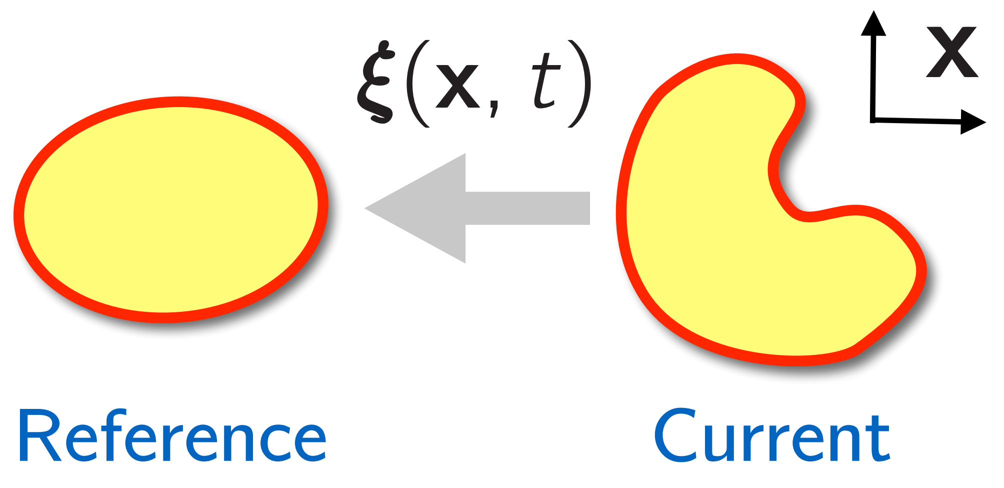


- Consider a mapping from the initial state to the current state, and examine a small region:



- Transformation described by the *deformation gradient* $\mathbf{F} = \dfrac{\partial \mathbf{x}}{\partial \mathbf{X}}$
- Specify arbitrary constitutive law to obtain stress $\boldsymbol{\sigma} = f(\mathbf{F})$

Reference map simulation



- Introduce reference map field $\boldsymbol{\xi}(\mathbf{x}, t)$
- Initial condition is $\boldsymbol{\xi}(\mathbf{x}, 0) = \mathbf{x}$
- Evolve ξ according to

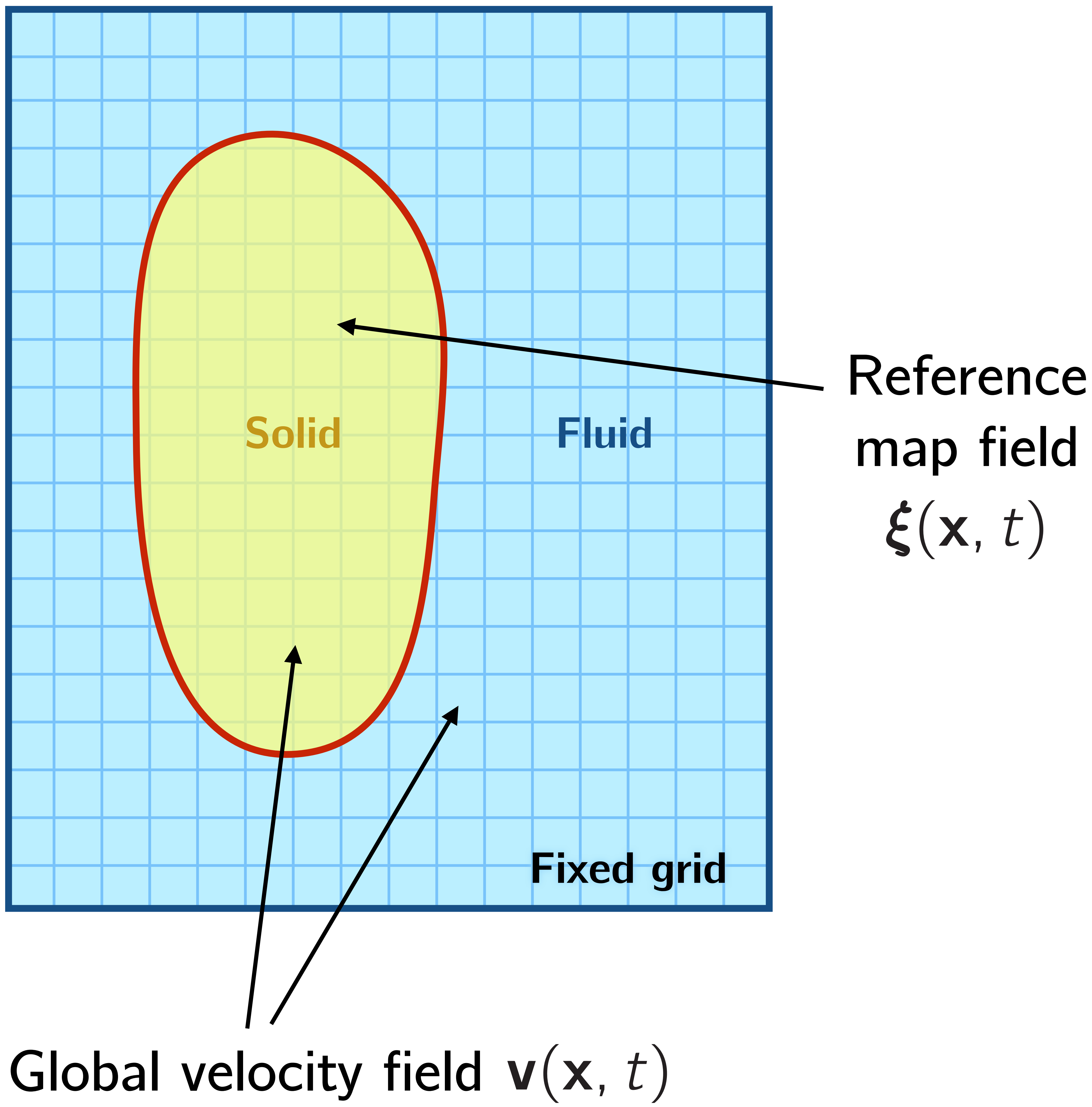
$$\frac{\partial \boldsymbol{\xi}}{\partial t} + (\mathbf{v} \cdot \nabla)\boldsymbol{\xi} = \mathbf{0}$$

- Deformation map computed as

$$\mathbf{F} = \frac{\partial \mathbf{x}}{\partial \boldsymbol{\xi}} = \left(\frac{\partial \boldsymbol{\xi}}{\partial \mathbf{x}}\right)^{-1}$$

- Use constitutive law $\boldsymbol{\sigma} = f(\mathbf{F})$ and Newton's second law

$$\rho \frac{d\mathbf{v}}{dt} = \nabla \cdot \boldsymbol{\sigma}$$



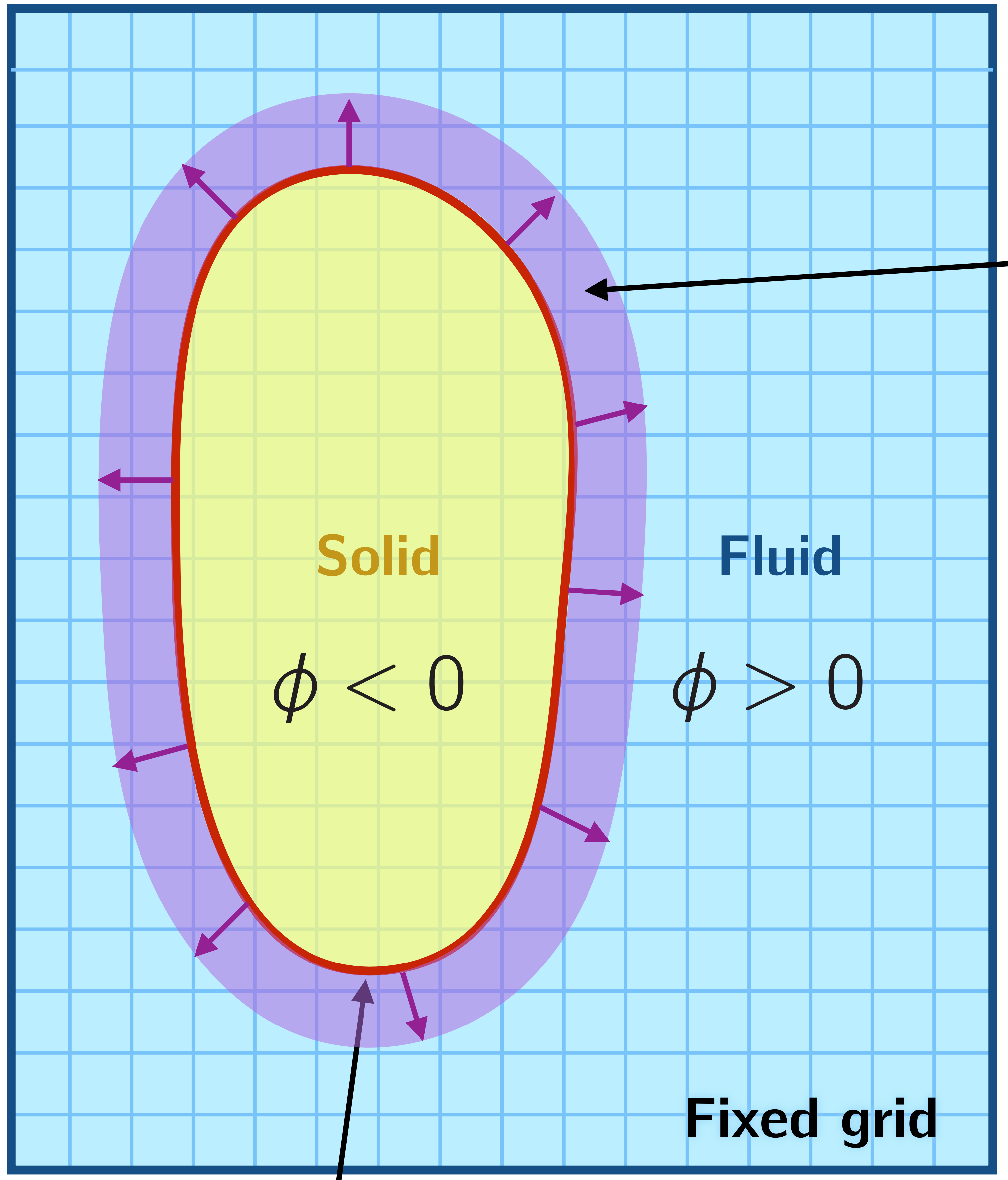
K. Kamrin, C. H. Rycroft, and J.-C. Nave, J. Mech. Phys. Solids **60**, 1952–1969 (2012).

Representing the boundary

- Introduce a function $\phi(\boldsymbol{\xi})$ describing the boundary
- Define $\phi < 0$ in the solid and $\phi > 0$ in the fluid
- For example for a circle of radius R at the origin

$$\phi(\boldsymbol{\xi}) = |\boldsymbol{\xi}| - R$$

- Requires extrapolating $\boldsymbol{\xi}$ into a layer of grid points in the fluid



K. Kamrin, C. H. Rycroft, and J.-C. Nave, J. Mech. Phys. Solids **60**, 1952–1969 (2012).

Stress computations in fluid and solid

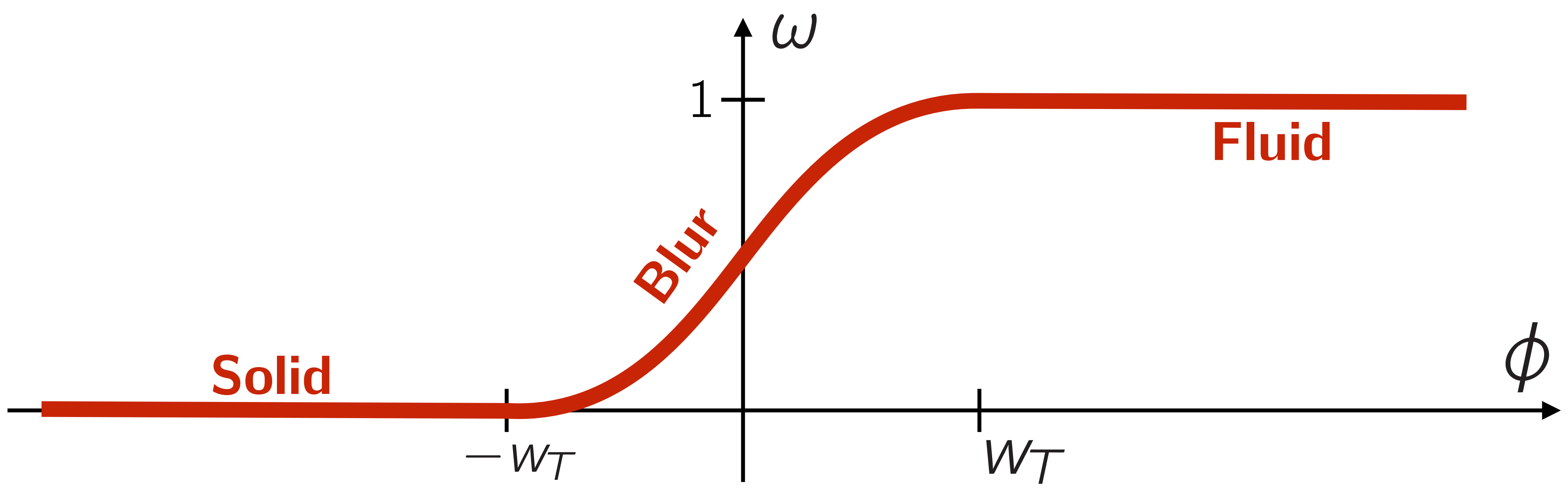
- The stress tensor:

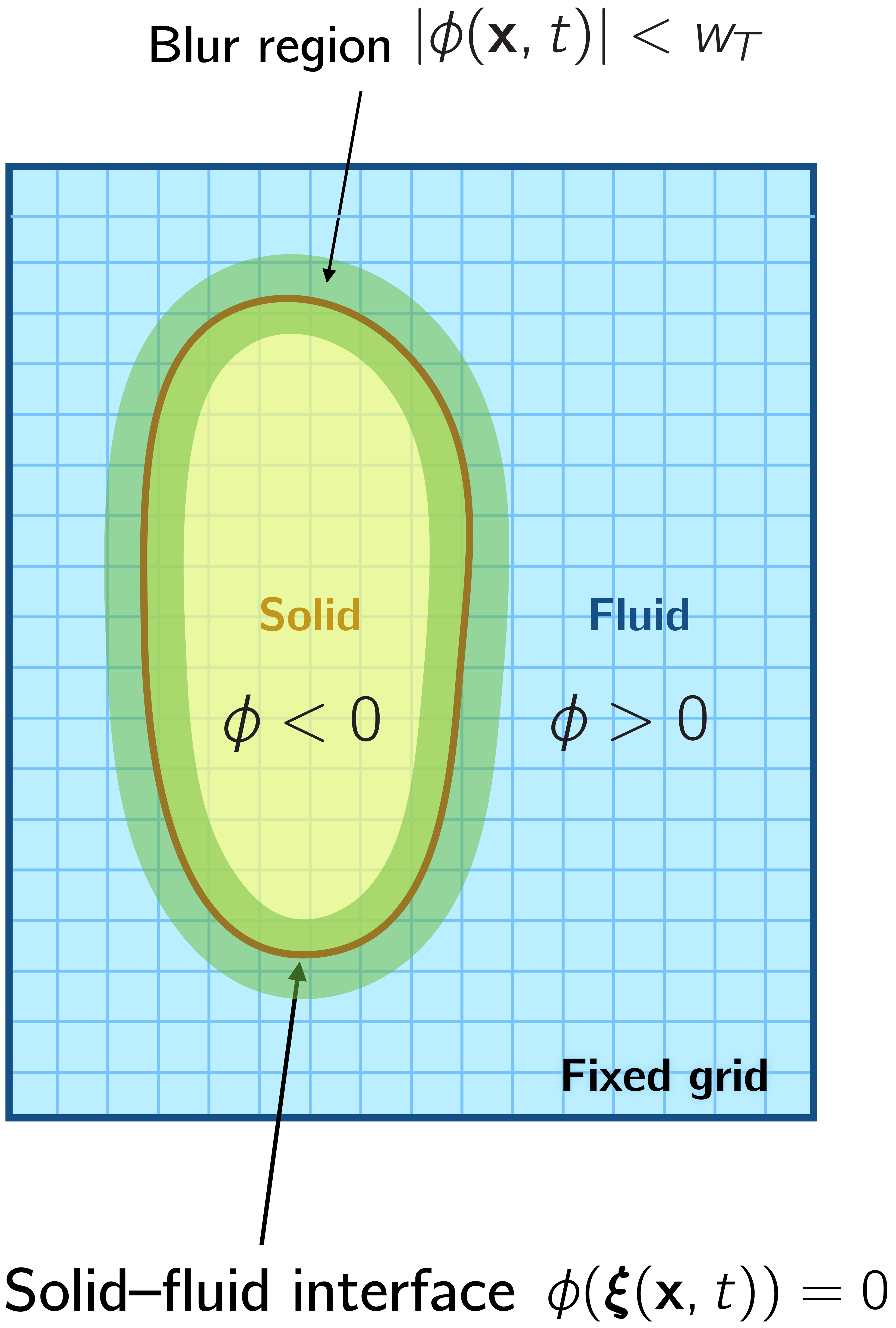
$$\boldsymbol{\sigma} = \omega \boldsymbol{\sigma}_{\text{fluid}} + (1-\omega)\boldsymbol{\sigma}_{\text{solid}}$$

- The transition function:

$$\omega(\mathbf{x}) = \begin{cases} 0 & \text{for } \mathbf{x} \in \text{solid} \\ H_s(\phi(\mathbf{x})) & \text{for } \mathbf{x} \in \text{blur region} \\ 1 & \text{for } \mathbf{x} \in \text{fluid} \end{cases}$$

$$H_s(\phi) = \frac{1}{2}\left(1 + \frac{\phi}{w_T} + \frac{1}{\pi}\sin\frac{\pi\phi}{w_T}\right)$$

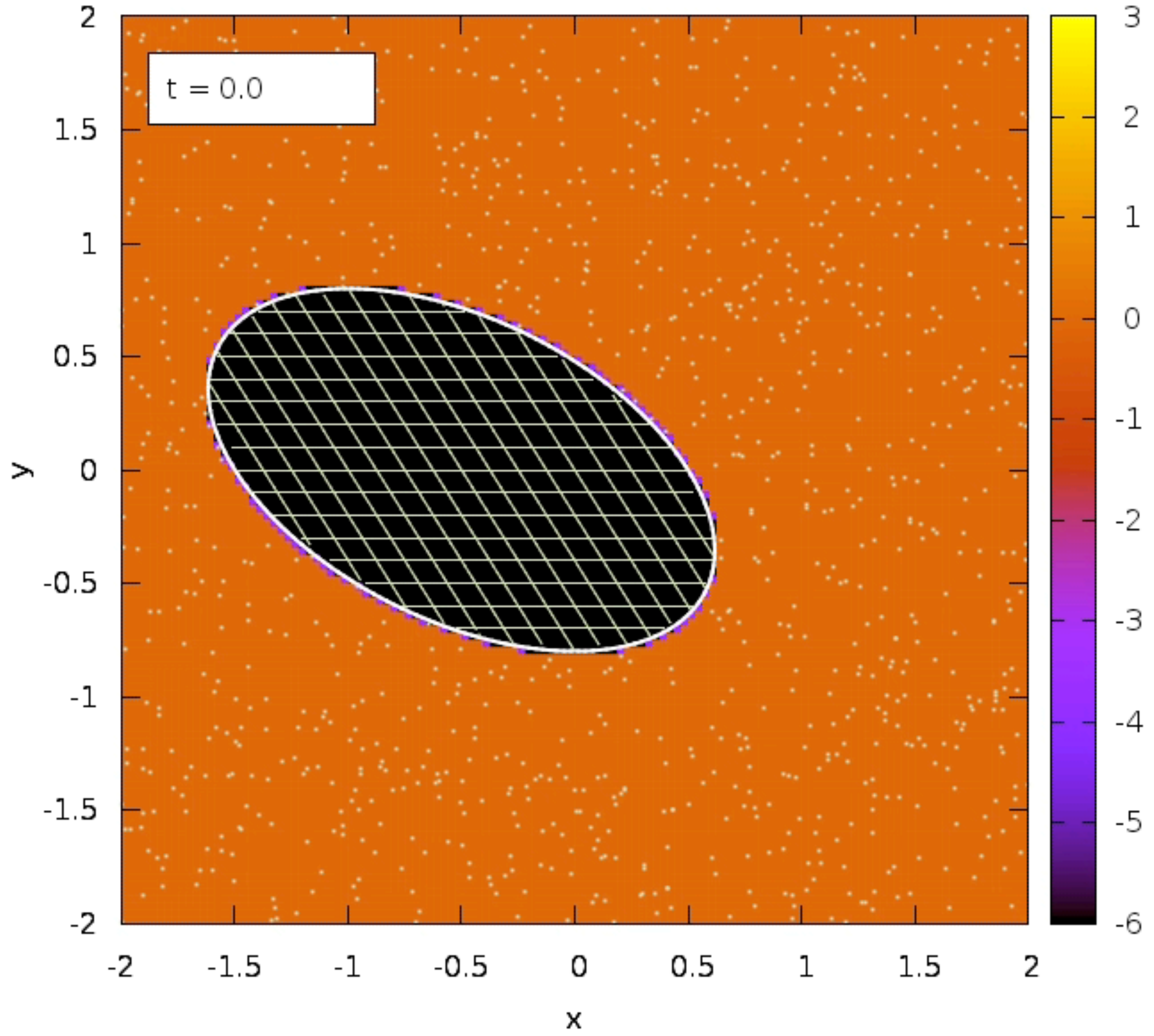


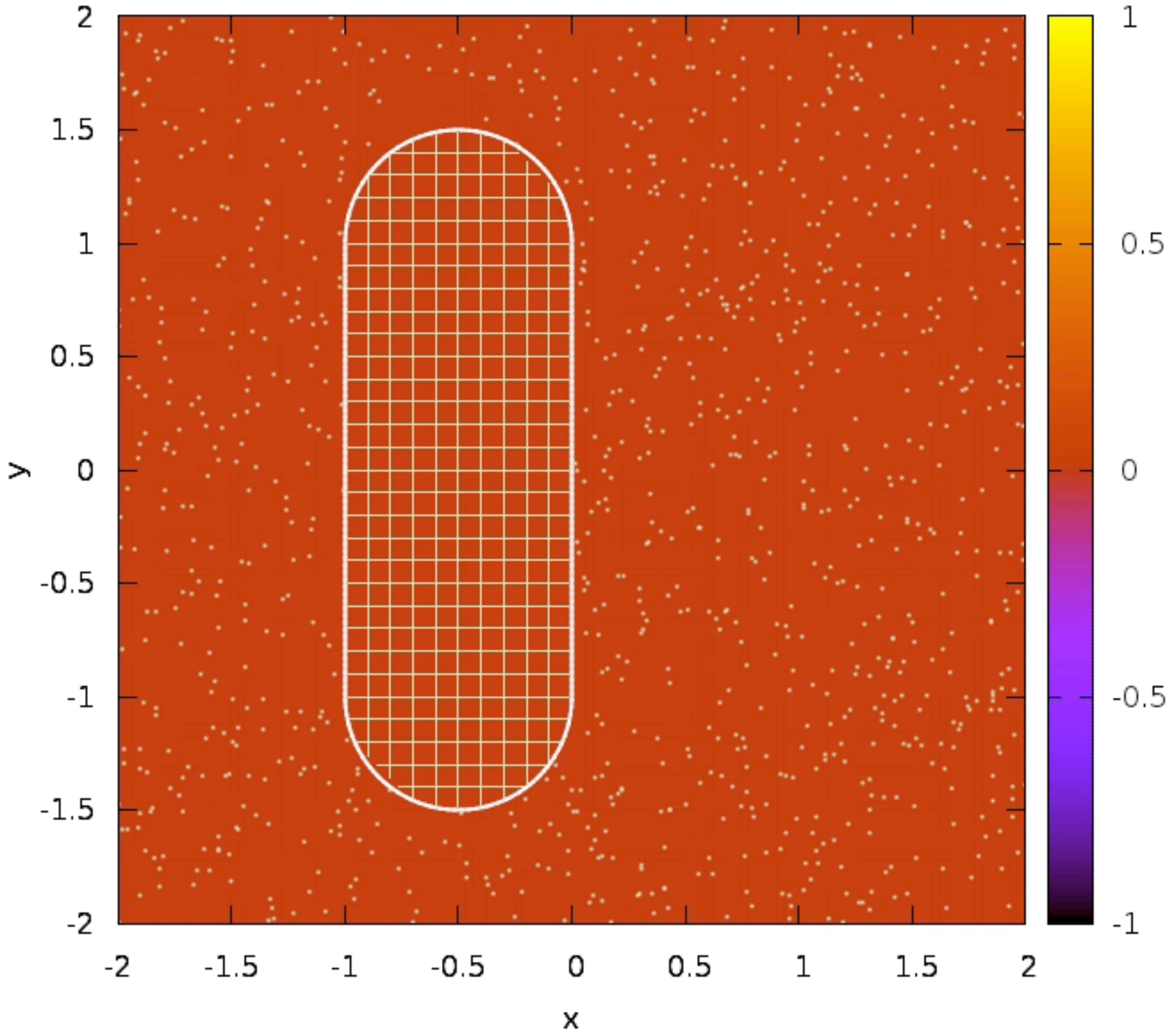


B. Valkov, C. H. Rycroft, and K. Kamrin, J. Appl. Mech. **82**, 041011 (2015).

Reference map simulations

Neo-Hookean solid and compressible fluid simulated with explicit timestepping procedure





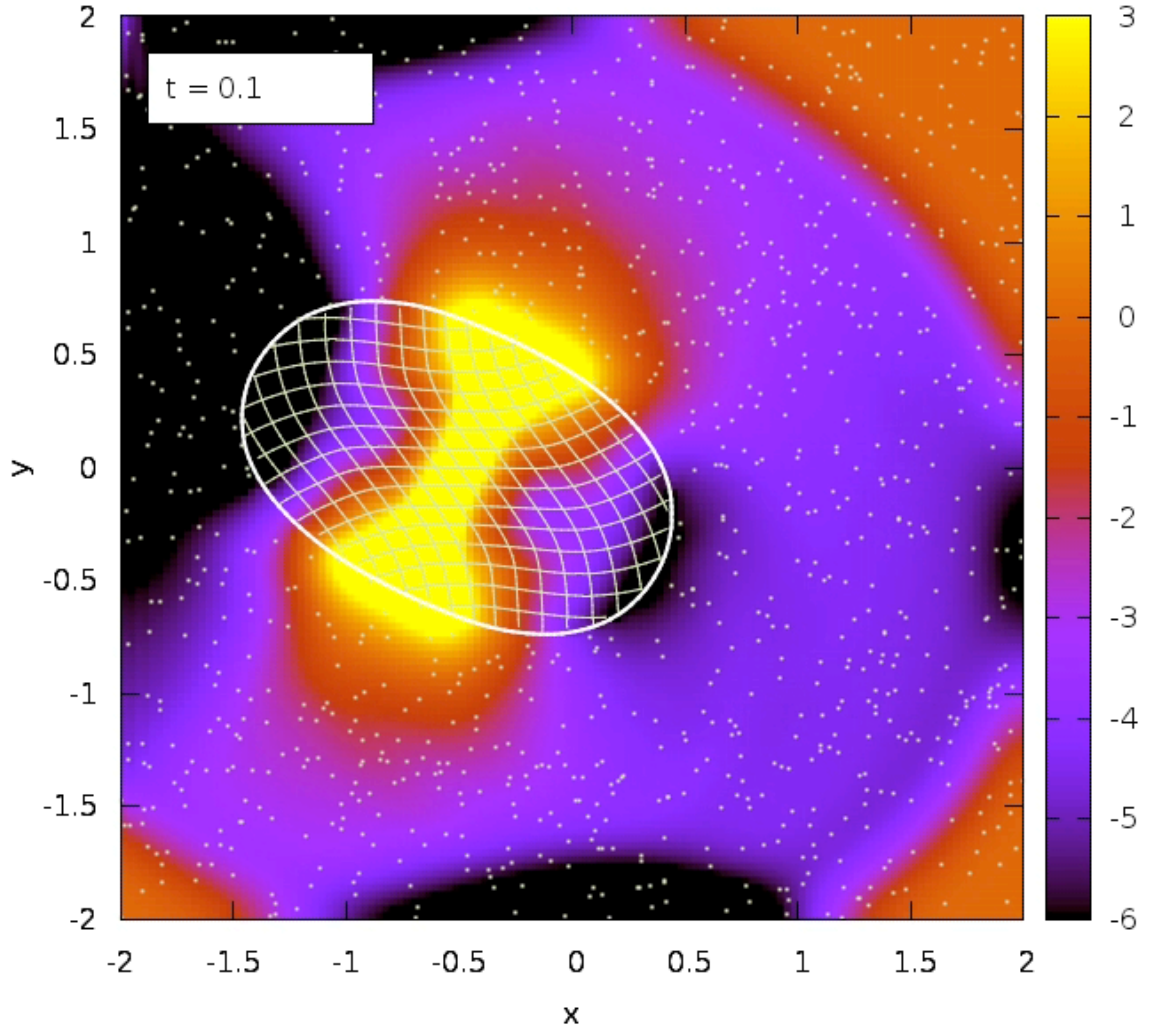
Pre-strained circular disk

(Colors show pressure field; non-dimensionalized units)

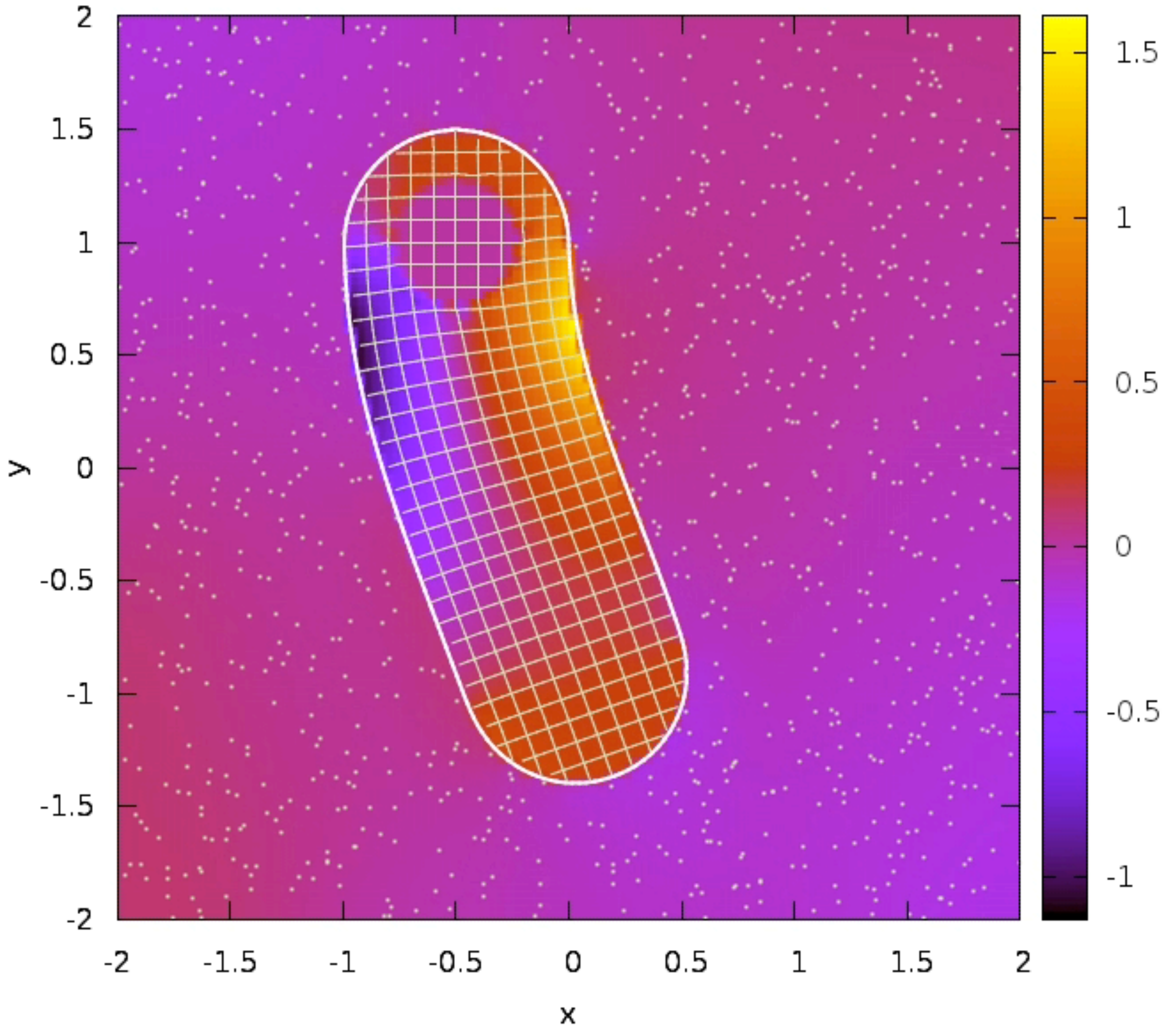
(129 by 129 numerical grid; 2 min computation time)

Reference map simulations

Neo-Hookean solid and compressible fluid simulated with explicit timestepping procedure



Pre-strained circular disk

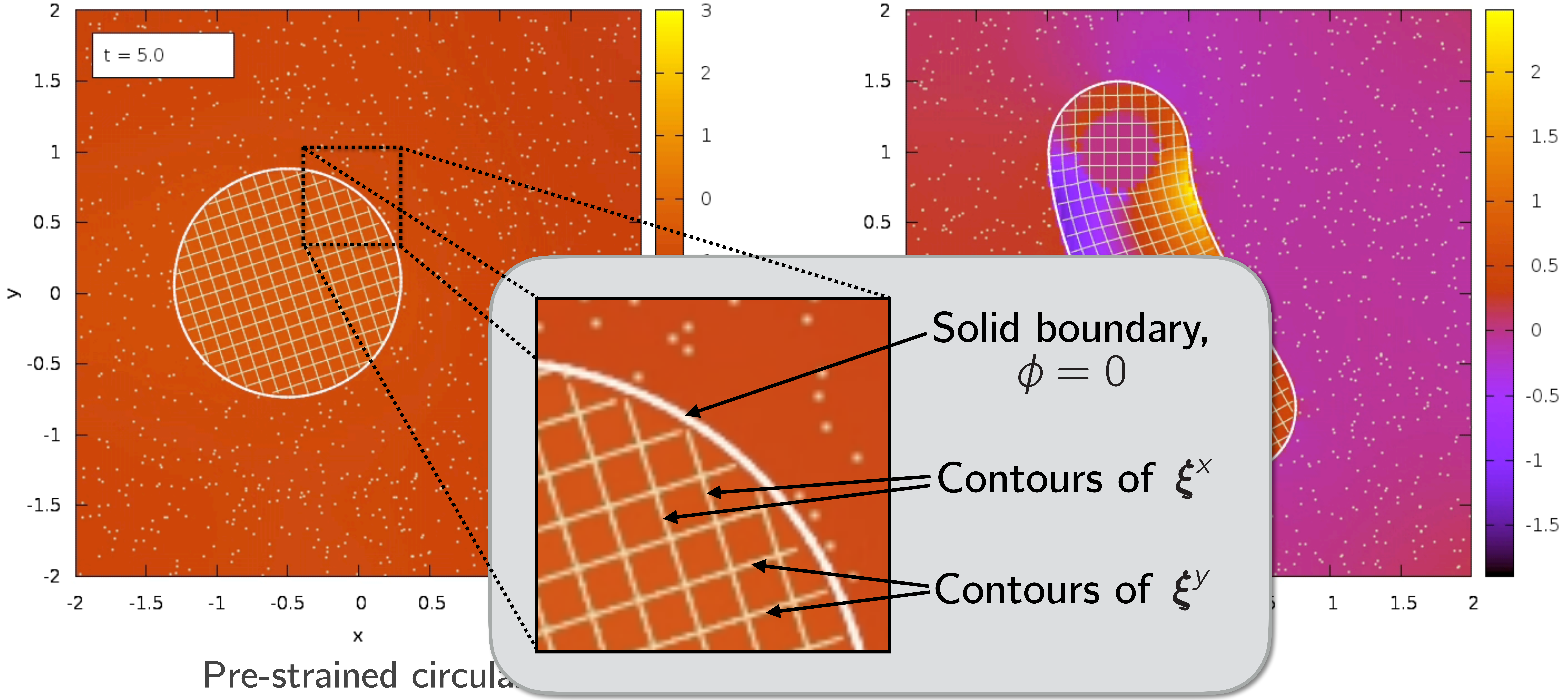


(Colors show pressure field; non-dimensionalized units)

(129 by 129 numerical grid; 2 min computation time)

Reference map simulations

Neo-Hookean solid and compressible fluid simulated with explicit timestepping procedure



Pre-strained circula

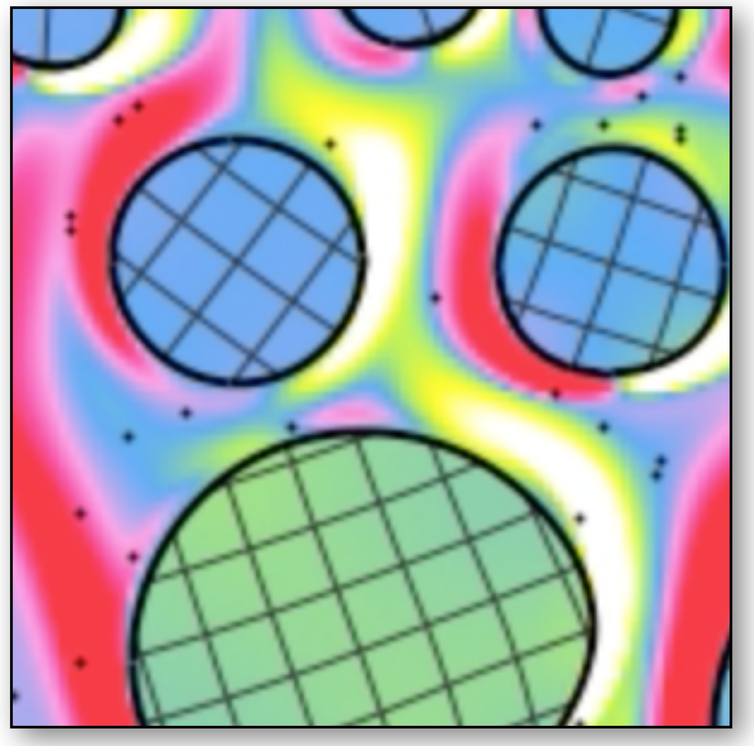
(Colors show pressure field; non-dimensionalized units)

(129 by 129 numerical grid; 2 min computation time)

Talk outline

1. The reference map technique

A conceptual overview of the method



2. Reference map simulation for incompressible fluids

A numerical implementation for simulating flag-flapping and multi-body contact

3. Applications and extensions

Three-dimensional simulation, higher-order discretizations

Toward incompressibility

- Typical fluids of interest (*e.g.* water) have very low compressibility
- Pressure waves travel at very fast speeds c on the order of km/s
- Courant–Friedrichs–Lewy (CFL) condition states the simulation timestep Δt must satisfy $\Delta t \leq \Delta x / c$
- Impose incompressibility: a good model, and also removes the CFL condition

Incompressible Navier–Stokes equations:

$$(\partial_t + (\mathbf{v} \cdot \nabla))\mathbf{v} = -(\nabla p)/\rho + \nu \nabla^2 \mathbf{v}$$

Constraint: $\nabla \cdot \mathbf{v} = 0$

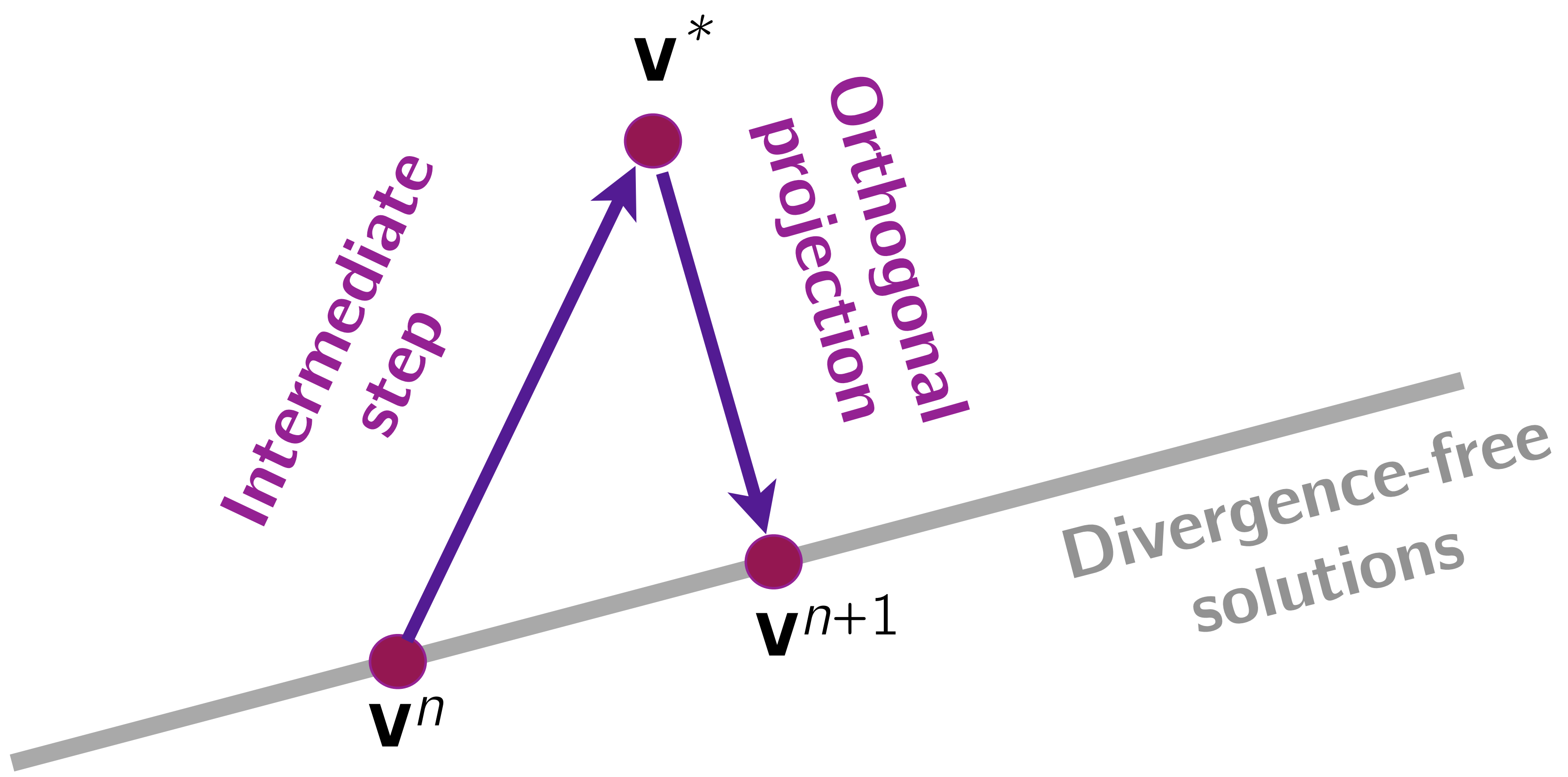
Δx: grid spacing

Chorin's projection method for incompressible fluid mechanics (1968)

Incompressible Navier–Stokes equations:

$$(\partial_t + (\mathbf{v}\cdot\nabla))\mathbf{v} = -(\nabla p)/\rho + \nu\nabla^2\mathbf{v}$$

Constraint: $\nabla\cdot\mathbf{v} = 0$



- Let $\mathbf{v}_n$ be the velocity at timestep n. First take an intermediate step

$$\frac{\mathbf{v}_* - \mathbf{v}_n}{\Delta t} = -(\mathbf{v}_n\cdot\nabla)\mathbf{v}_n + \nu\nabla^2\mathbf{v}_n$$

- Then

$$\frac{\mathbf{v}_{n+1} - \mathbf{v}_*}{\Delta t} = -\frac{\nabla p_{n+1}}{\rho} \qquad (*)$$

- Taking divergence and enforcing $\nabla\cdot\mathbf{v}_{n+1} = 0$ gives Poisson problem for pressure

$$\nabla^2 p_{n+1} = \frac{\rho\nabla\cdot\mathbf{v}_*}{\Delta t}$$

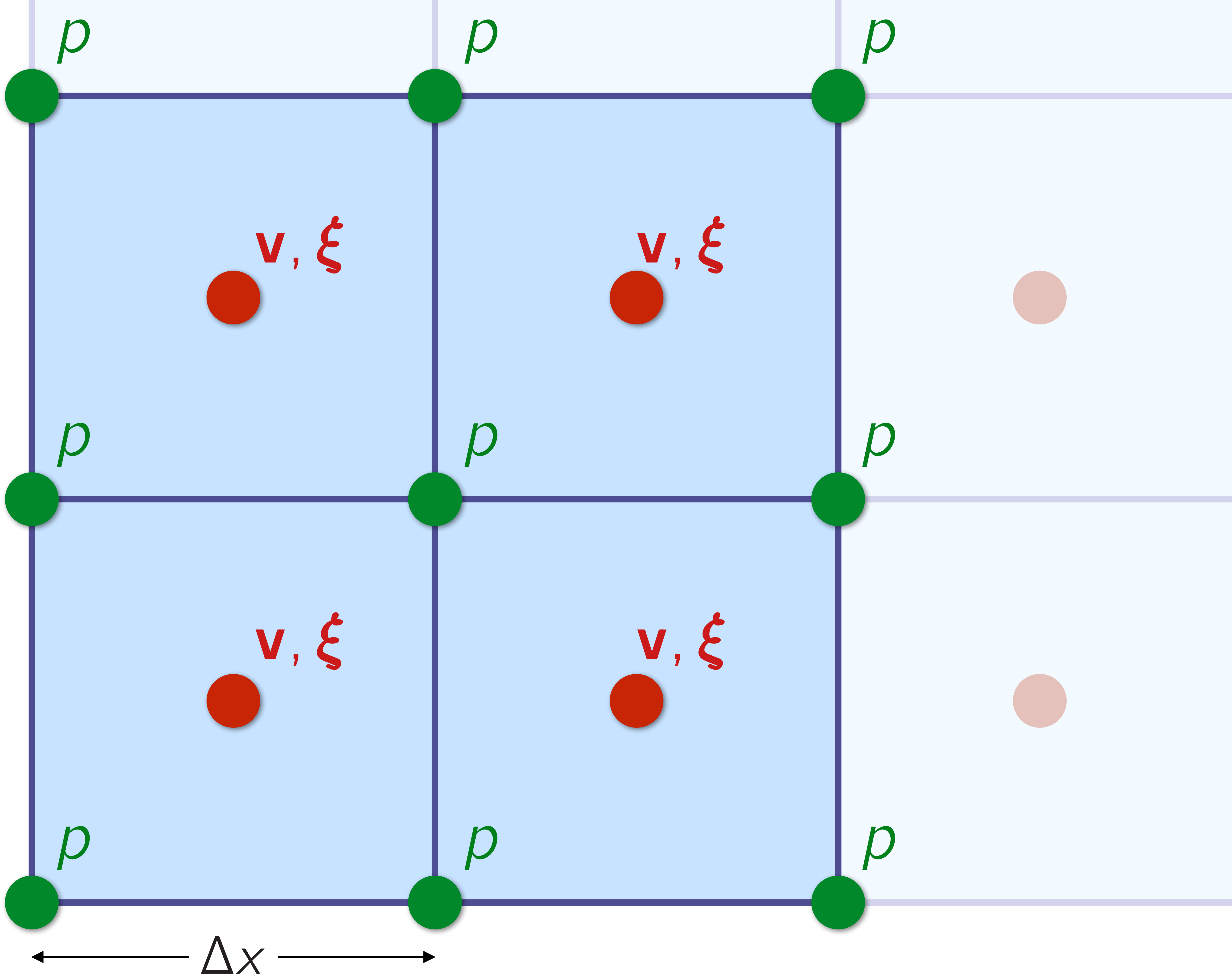
- Calculate $\mathbf{v}_{n+1}$ using (*)

A. J. Chorin, *Numerical solution of the Navier–Stokes equations*, Math. Comp. **22**, 745–762 (1968).

ρ : density, ν : viscosity

A modern implementation of Chorin's method

- Since original paper, many improvements to Chorin's method have been introduced
- Reference map technique is built upon a modern implementation of Chorin's method



Intermediate step: $\mathbf{v}_* = \mathbf{v}_n + \Delta t \left(-[(\mathbf{v}\cdot\nabla)\mathbf{v}]_{n+1/2} + \frac{1}{\rho}\nabla\cdot\boldsymbol{\sigma}_n \right)$

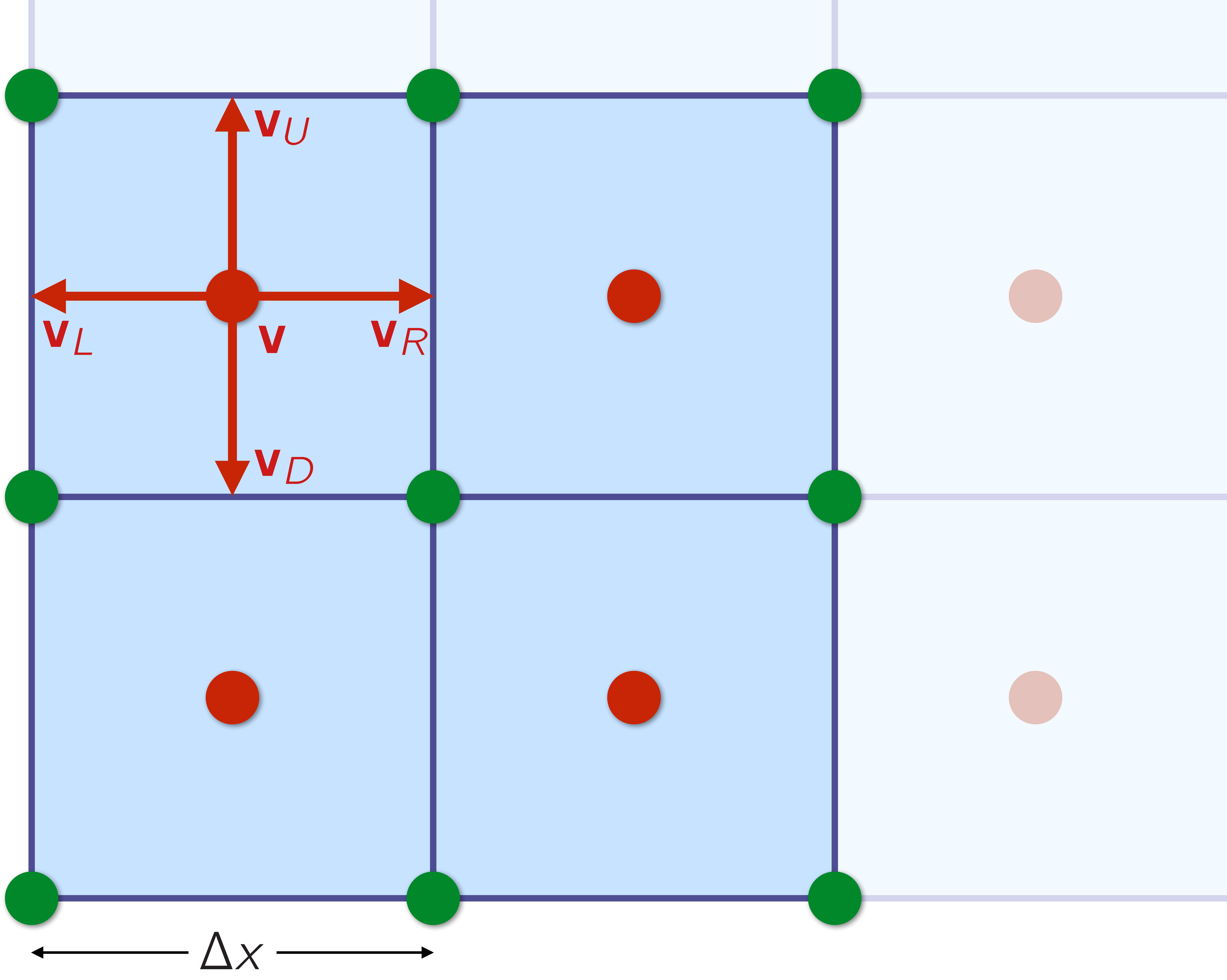
Projection step: $\nabla\cdot\mathbf{v}_* = \frac{\Delta t}{\rho}\nabla^2 p_{n+1}$ $\quad \mathbf{v}_{n+1} = \mathbf{v}_* - \frac{\Delta t}{\rho}\nabla p_{n+1}$

Reference map step: $\boldsymbol{\xi}_{n+1} = \boldsymbol{\xi}_n - \Delta t\, [(\mathbf{v}\cdot\nabla)\boldsymbol{\xi}]_{n+1/2}$

Fluid advection term

- To handle advective term, first construct edge velocities at the half-timestep $\Delta t/2$ using Taylor expansions
- For example

$$\mathbf{v}_R = \mathbf{v} + \frac{\Delta t}{2}\frac{\partial \mathbf{v}}{\partial t} + \frac{\Delta x}{2}\frac{\partial \mathbf{v}}{\partial x}$$



Intermediate step: $\mathbf{v}_* = \mathbf{v}_n + \Delta t \left(-[(\mathbf{v}\cdot\nabla)\mathbf{v}]_{n+1/2} + \frac{1}{\rho}\nabla\cdot\boldsymbol{\sigma}_n \right)$

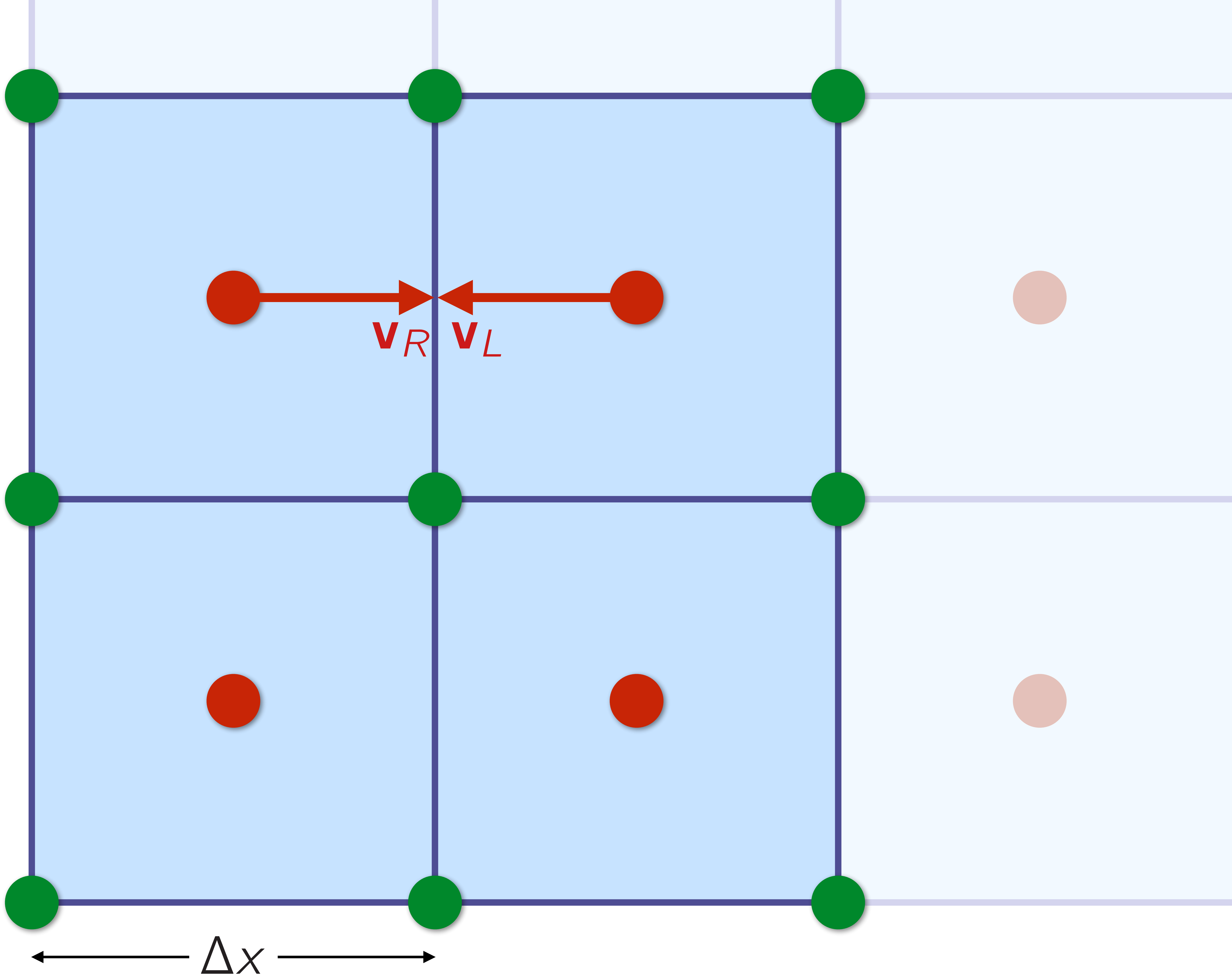
Projection step: $\nabla\cdot\mathbf{v}_* = \frac{\Delta t}{\rho}\nabla^2 p_{n+1}$ $\quad \mathbf{v}_{n+1} = \mathbf{v}_* - \frac{\Delta t}{\rho}\nabla p_{n+1}$

Reference map step: $\boldsymbol{\xi}_{n+1} = \boldsymbol{\xi}_n - \Delta t\,[(\mathbf{v}\cdot\nabla)\boldsymbol{\xi}]_{n+1/2}$

Fluid advection term

- Each edge obtains two velocities from its neighboring cells
- To capture flow of information, use Godunov upwinding procedure
- For x component:

$$\mathbf{v}_E^x = \begin{cases} \mathbf{v}_R^x & \text{if } \mathbf{v}_R^x > 0 \text{ and } \mathbf{v}_R^x + \mathbf{v}_L^x > 0 \\ \mathbf{v}_L^x & \text{if } \mathbf{v}_L^x < 0 \text{ and } \mathbf{v}_R^x + \mathbf{v}_L^x < 0 \\ 0 & \text{otherwise} \end{cases}$$



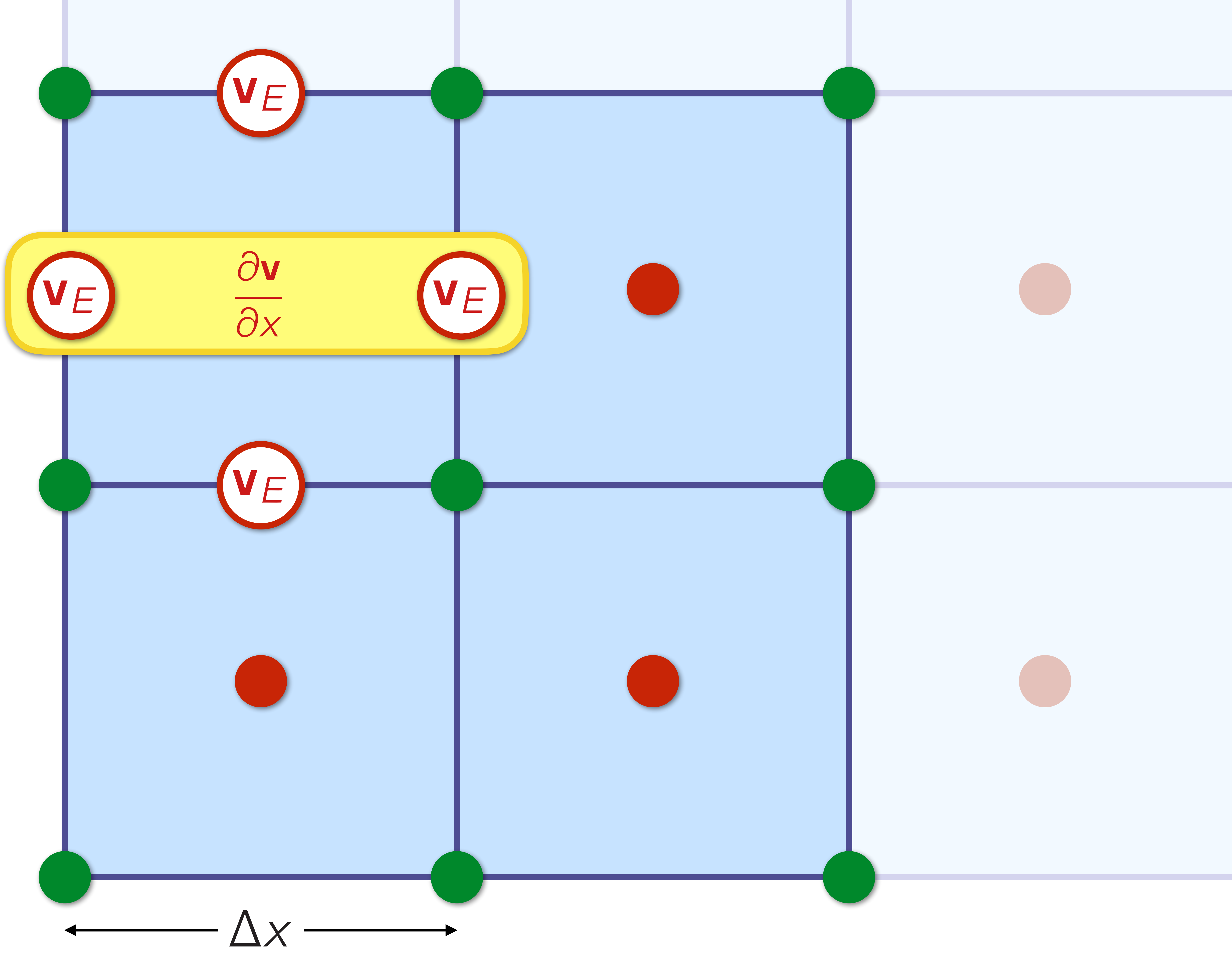
Intermediate step: $\mathbf{v}_* = \mathbf{v}_n + \Delta t \left(-[(\mathbf{v} \cdot \nabla)\mathbf{v}]_{n+1/2} + \frac{1}{\rho} \nabla \cdot \boldsymbol{\sigma}_n \right)$

Projection step: $\nabla \cdot \mathbf{v}_* = \frac{\Delta t}{\rho} \nabla^2 p_{n+1}$ $\quad \mathbf{v}_{n+1} = \mathbf{v}_* - \frac{\Delta t}{\rho} \nabla p_{n+1}$

Reference map step: $\boldsymbol{\xi}_{n+1} = \boldsymbol{\xi}_n - \Delta t \, [(\mathbf{v} \cdot \nabla)\boldsymbol{\xi}]_{n+1/2}$

Fluid advection term

- With edge velocities in place, advection term evaluated using finite differences
- Achieves second-order spatial accuracy



Intermediate step: $\mathbf{v}_* = \mathbf{v}_n + \Delta t \left(-[(\mathbf{v}\cdot\nabla)\mathbf{v}]_{n+1/2} + \frac{1}{\rho}\nabla\cdot\boldsymbol{\sigma}_n \right)$

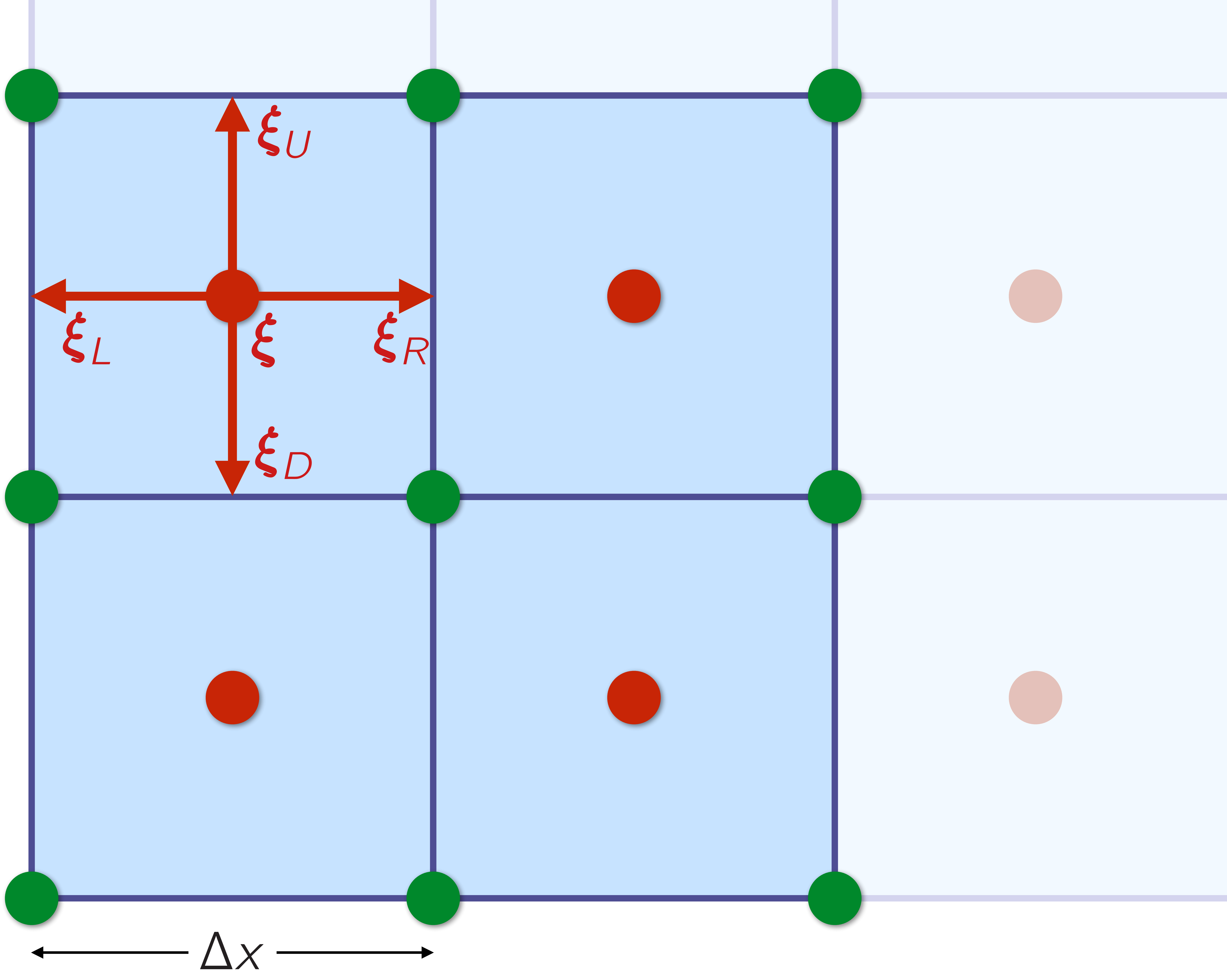
Projection step: $\nabla\cdot\mathbf{v}_* = \frac{\Delta t}{\rho}\nabla^2 p_{n+1}$ $\quad \mathbf{v}_{n+1} = \mathbf{v}_* - \frac{\Delta t}{\rho}\nabla p_{n+1}$

Reference map step: $\boldsymbol{\xi}_{n+1} = \boldsymbol{\xi}_n - \Delta t\,[(\mathbf{v}\cdot\nabla)\boldsymbol{\xi}]_{n+1/2}$

Reference map advection term

- Same procedure for velocity works for the reference map field
- Construct half-timestep edge extrapolations using formulae such as

$$\boldsymbol{\xi}_R = \boldsymbol{\xi} + \frac{\Delta t}{2}\frac{\partial \boldsymbol{\xi}}{\partial t} + \frac{\Delta x}{2}\frac{\partial \boldsymbol{\xi}}{\partial x}$$



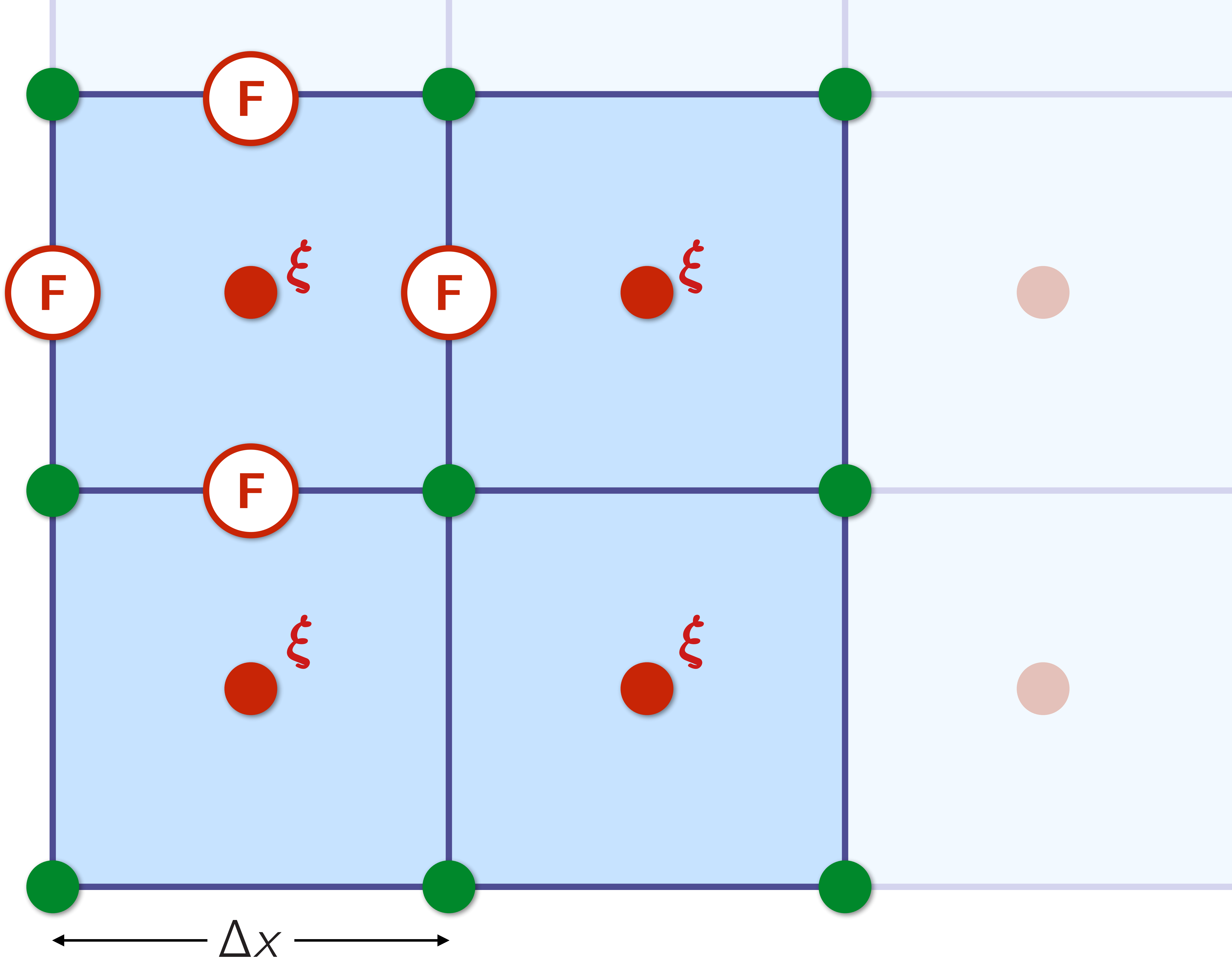
Intermediate step: $\mathbf{v}_* = \mathbf{v}_n + \Delta t \left(-[(\mathbf{v}\cdot\nabla)\mathbf{v}]_{n+1/2} + \frac{1}{\rho}\nabla\cdot\boldsymbol{\sigma}_n \right)$

Projection step: $\nabla\cdot\mathbf{v}_* = \frac{\Delta t}{\rho}\nabla^2 p_{n+1} \qquad \mathbf{v}_{n+1} = \mathbf{v}_* - \frac{\Delta t}{\rho}\nabla p_{n+1}$

Reference map step: $\boldsymbol{\xi}_{n+1} = \boldsymbol{\xi}_n - \Delta t\, [(\mathbf{v}\cdot\nabla)\boldsymbol{\xi}]_{n+1/2}$

Stress calculation

- First, compute the deformation gradient tensor **F** on each edge using finite differences of ξ



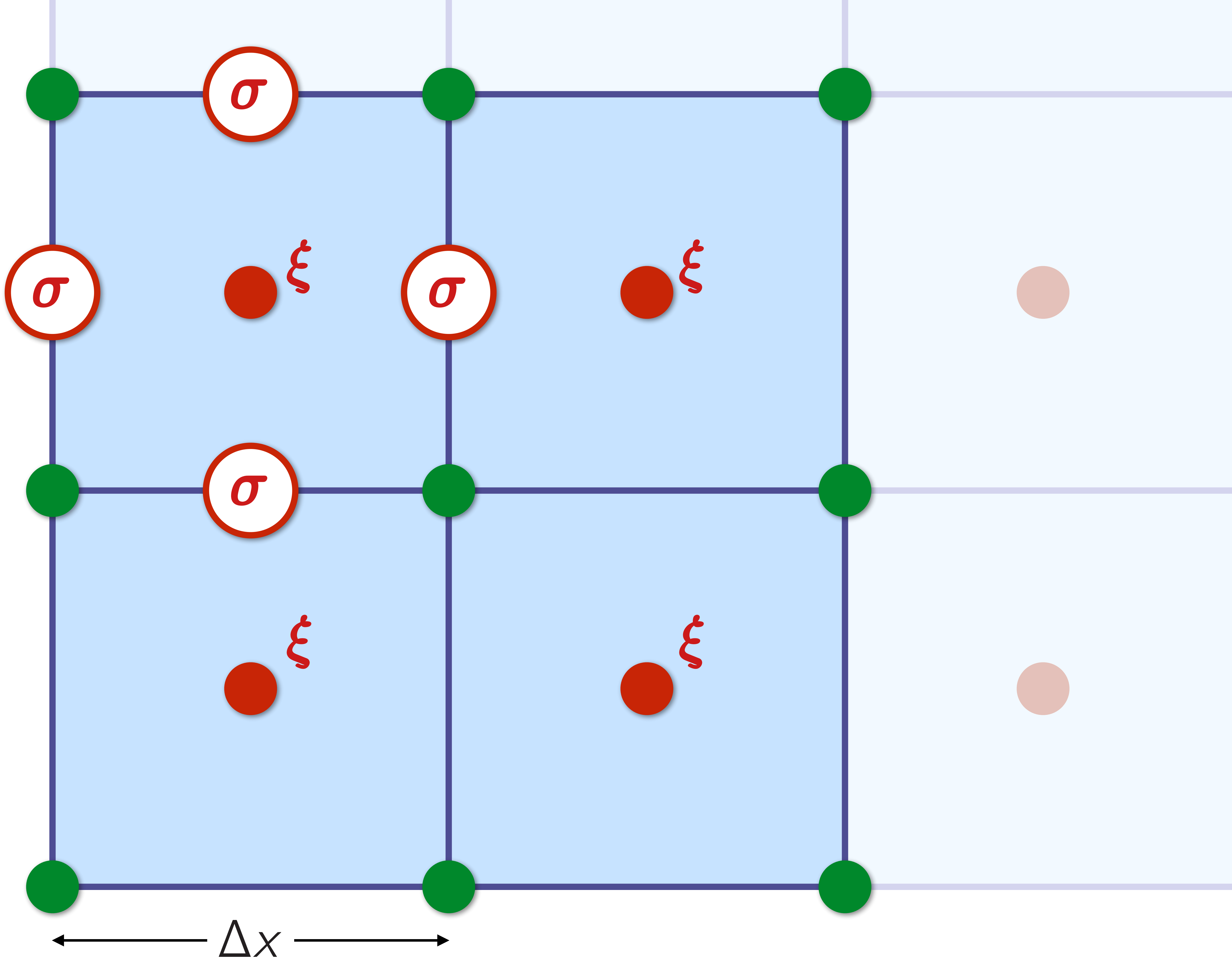
Intermediate step: $\mathbf{v}_* = \mathbf{v}_n + \Delta t \left(-[(\mathbf{v} \cdot \nabla)\mathbf{v}]_{n+1/2} + \frac{1}{\rho} \nabla \cdot \boldsymbol{\sigma}_n \right)$

Projection step: $\nabla \cdot \mathbf{v}_* = \frac{\Delta t}{\rho} \nabla^2 p_{n+1}$ $\quad \mathbf{v}_{n+1} = \mathbf{v}_* - \frac{\Delta t}{\rho} \nabla p_{n+1}$

Reference map step: $\boldsymbol{\xi}_{n+1} = \boldsymbol{\xi}_n - \Delta t \, [(\mathbf{v} \cdot \nabla)\boldsymbol{\xi}]_{n+1/2}$

Stress calculation

- First, compute the deformation gradient tensor $\mathbf{F}$ on each edge using finite differences of ξ
- Use constitutive law $\boldsymbol{\sigma} = f(\mathbf{F})$ to compute the edge stresses (merging fluid/solid stress as needed)



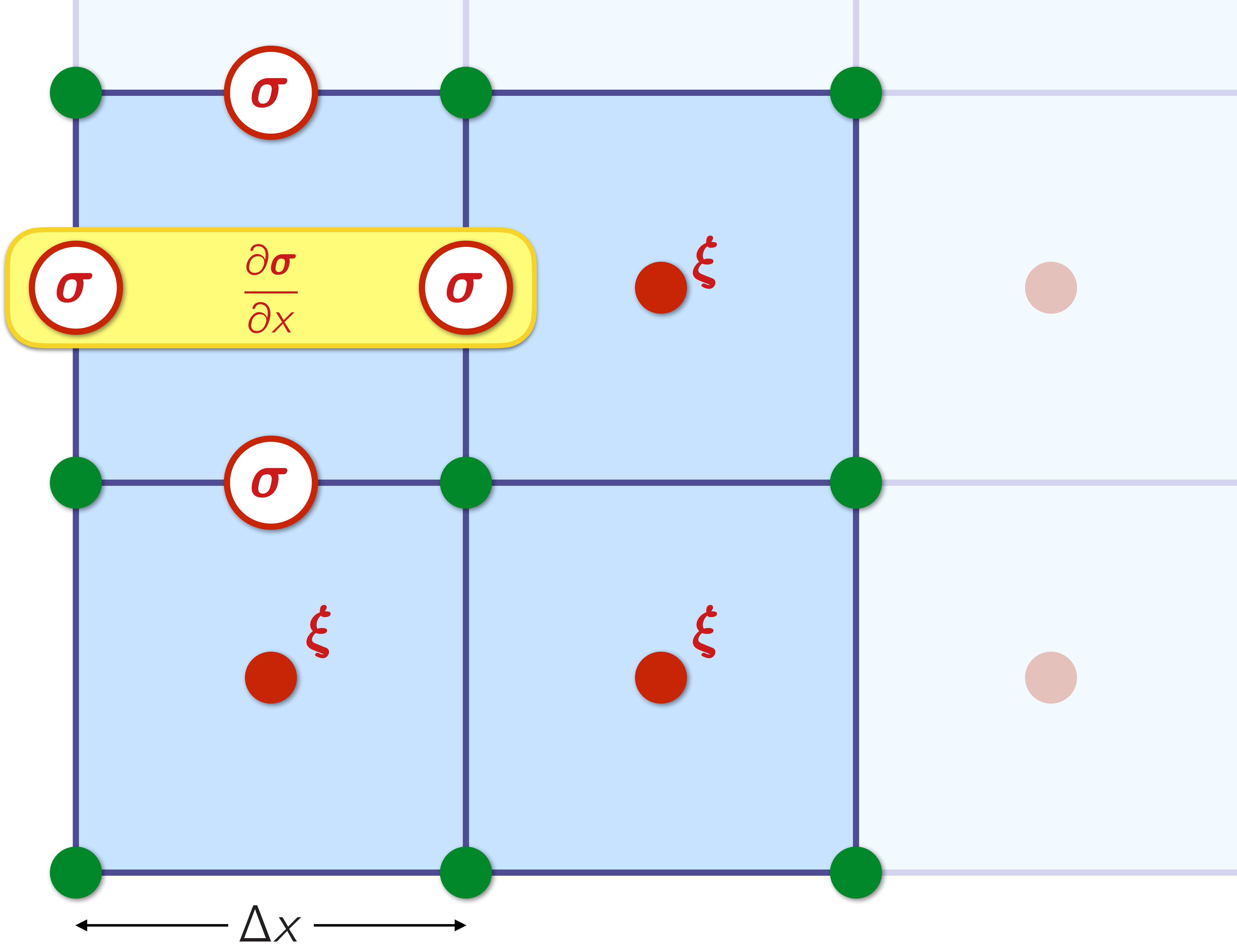
Intermediate step: $\mathbf{v}_* = \mathbf{v}_n + \Delta t \left(-[(\mathbf{v}\cdot\nabla)\mathbf{v}]_{n+1/2} + \frac{1}{\rho}\nabla\cdot\boldsymbol{\sigma}_n \right)$

Projection step: $\nabla\cdot\mathbf{v}_* = \frac{\Delta t}{\rho}\nabla^2 p_{n+1}$ $\quad \mathbf{v}_{n+1} = \mathbf{v}_* - \frac{\Delta t}{\rho}\nabla p_{n+1}$

Reference map step: $\boldsymbol{\xi}_{n+1} = \boldsymbol{\xi}_n - \Delta t\,[(\mathbf{v}\cdot\nabla)\boldsymbol{\xi}]_{n+1/2}$

Stress calculation

- First, compute the deformation gradient tensor **F** on each edge using finite differences of ξ
- Use constitutive law $\boldsymbol{\sigma} = f(\mathbf{F})$ to compute the edge stresses (merging fluid/solid stress as needed)
- Compute $\nabla \cdot \boldsymbol{\sigma}$



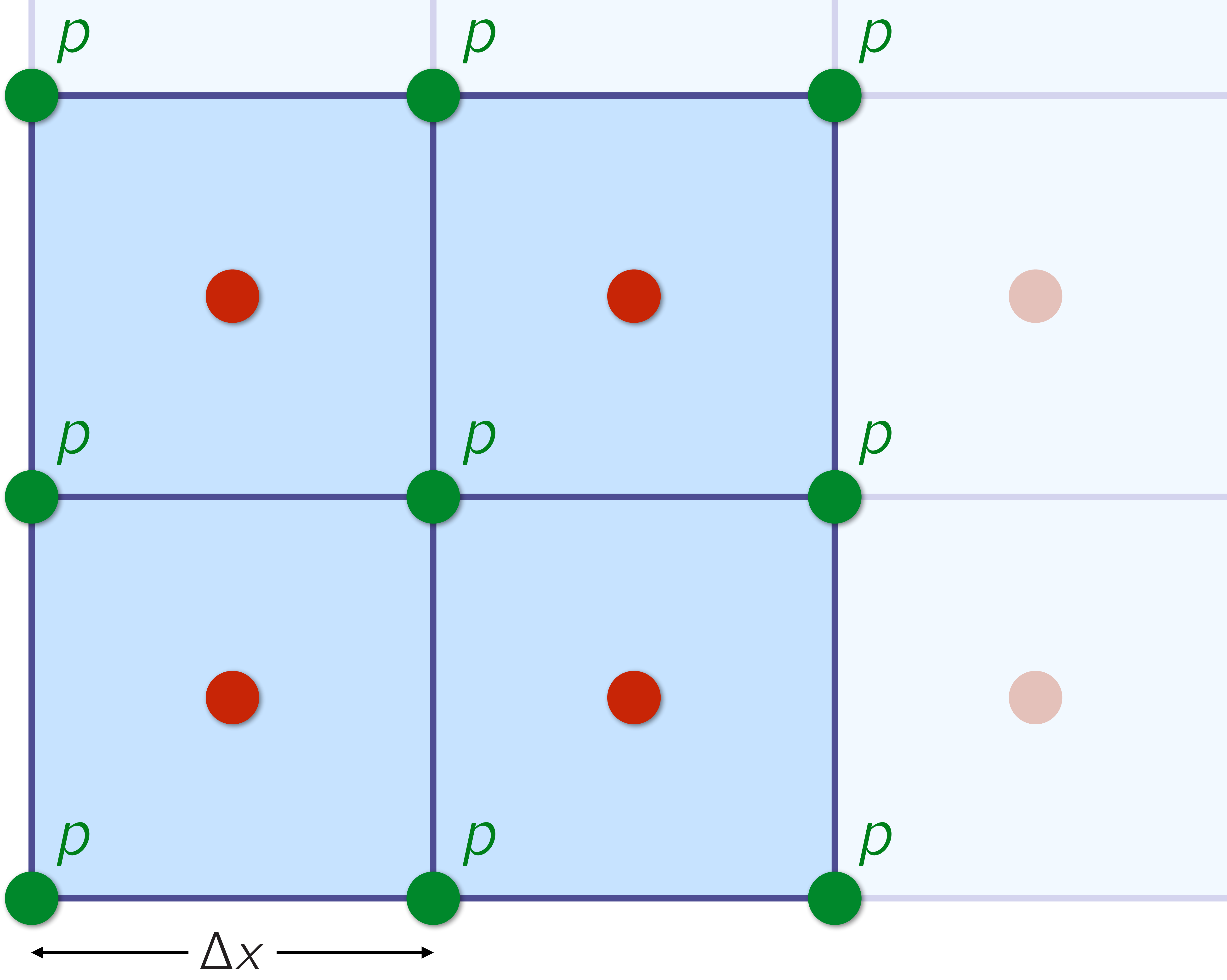
Intermediate step: $\mathbf{v}_* = \mathbf{v}_n + \Delta t \left(-[(\mathbf{v} \cdot \nabla)\mathbf{v}]_{n+1/2} + \frac{1}{\rho} \nabla \cdot \boldsymbol{\sigma}_n \right)$

Projection step: $\nabla \cdot \mathbf{v}_* = \frac{\Delta t}{\rho} \nabla^2 p_{n+1} \qquad \mathbf{v}_{n+1} = \mathbf{v}_* - \frac{\Delta t}{\rho} \nabla p_{n+1}$

Reference map step: $\boldsymbol{\xi}_{n+1} = \boldsymbol{\xi}_n - \Delta t \, [(\mathbf{v} \cdot \nabla)\boldsymbol{\xi}]_{n+1/2}$

The projection step

- Use finite-element method to compute the pressure field p_{n+1}
- Requires solving large linear system of equations: use a custom multigrid method



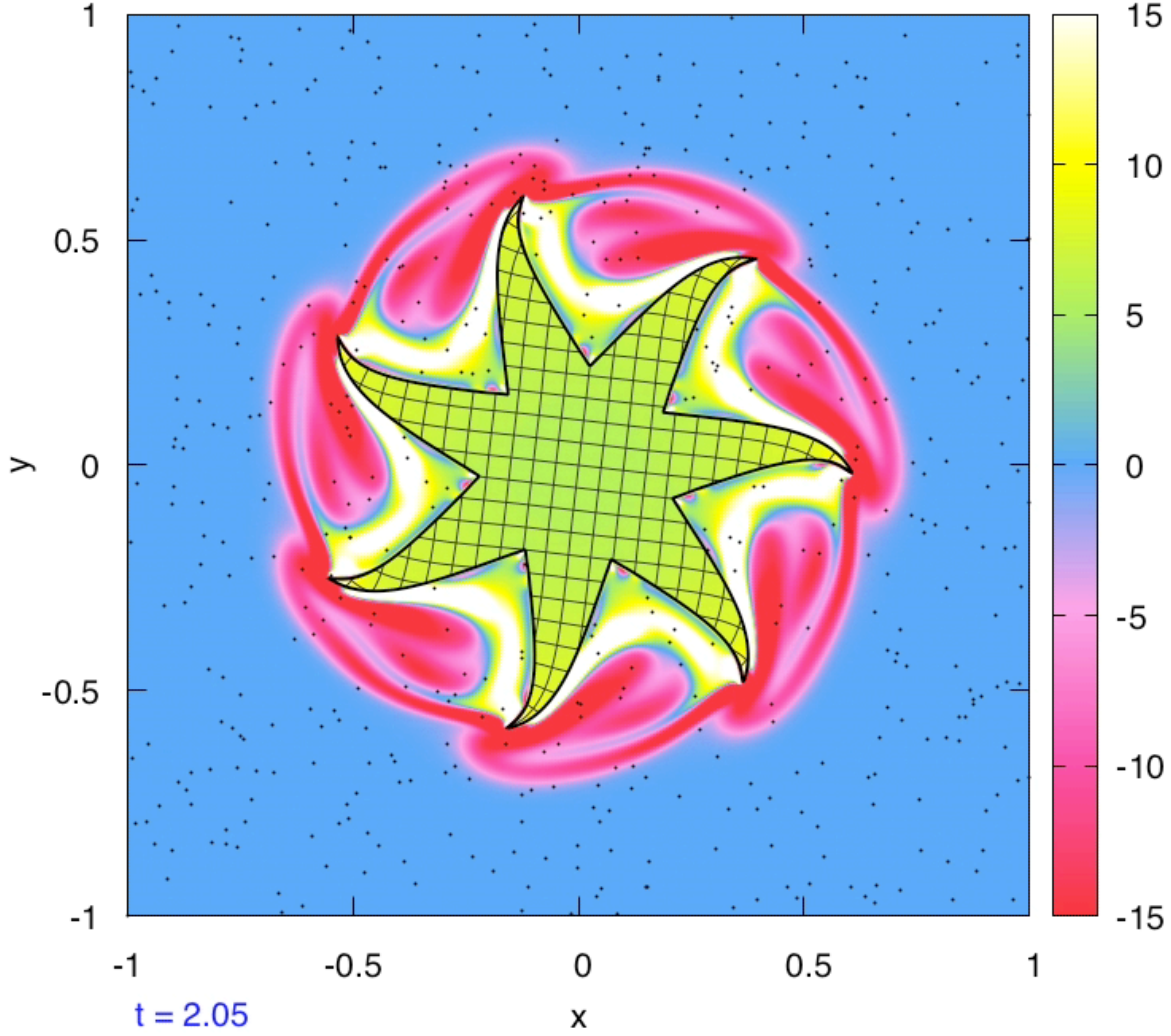
Intermediate step: $\mathbf{v}_* = \mathbf{v}_n + \Delta t \left(-[(\mathbf{v} \cdot \nabla)\mathbf{v}]_{n+1/2} + \frac{1}{\rho} \nabla \cdot \boldsymbol{\sigma}_n \right)$

Projection step: $\nabla \cdot \mathbf{v}_* = \frac{\Delta t}{\rho} \nabla^2 p_{n+1}$ $\quad \mathbf{v}_{n+1} = \mathbf{v}_* - \frac{\Delta t}{\rho} \nabla p_{n+1}$

Reference map step: $\boldsymbol{\xi}_{n+1} = \boldsymbol{\xi}_n - \Delta t \, [(\mathbf{v} \cdot \nabla)\boldsymbol{\xi}]_{n+1/2}$

Example 2D computation

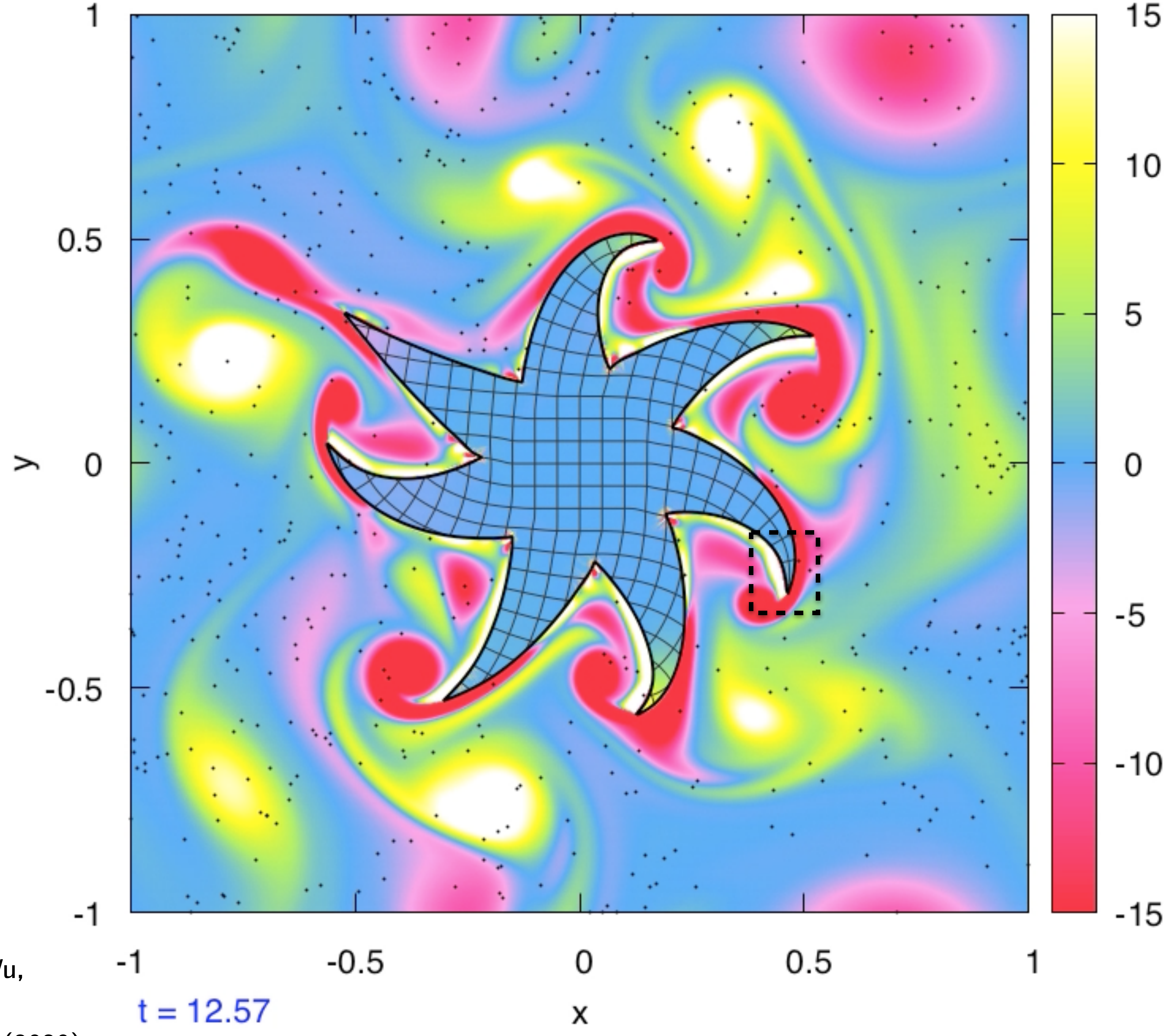
(All simulations use non-dimensionalized units)
(Colors show vorticity)

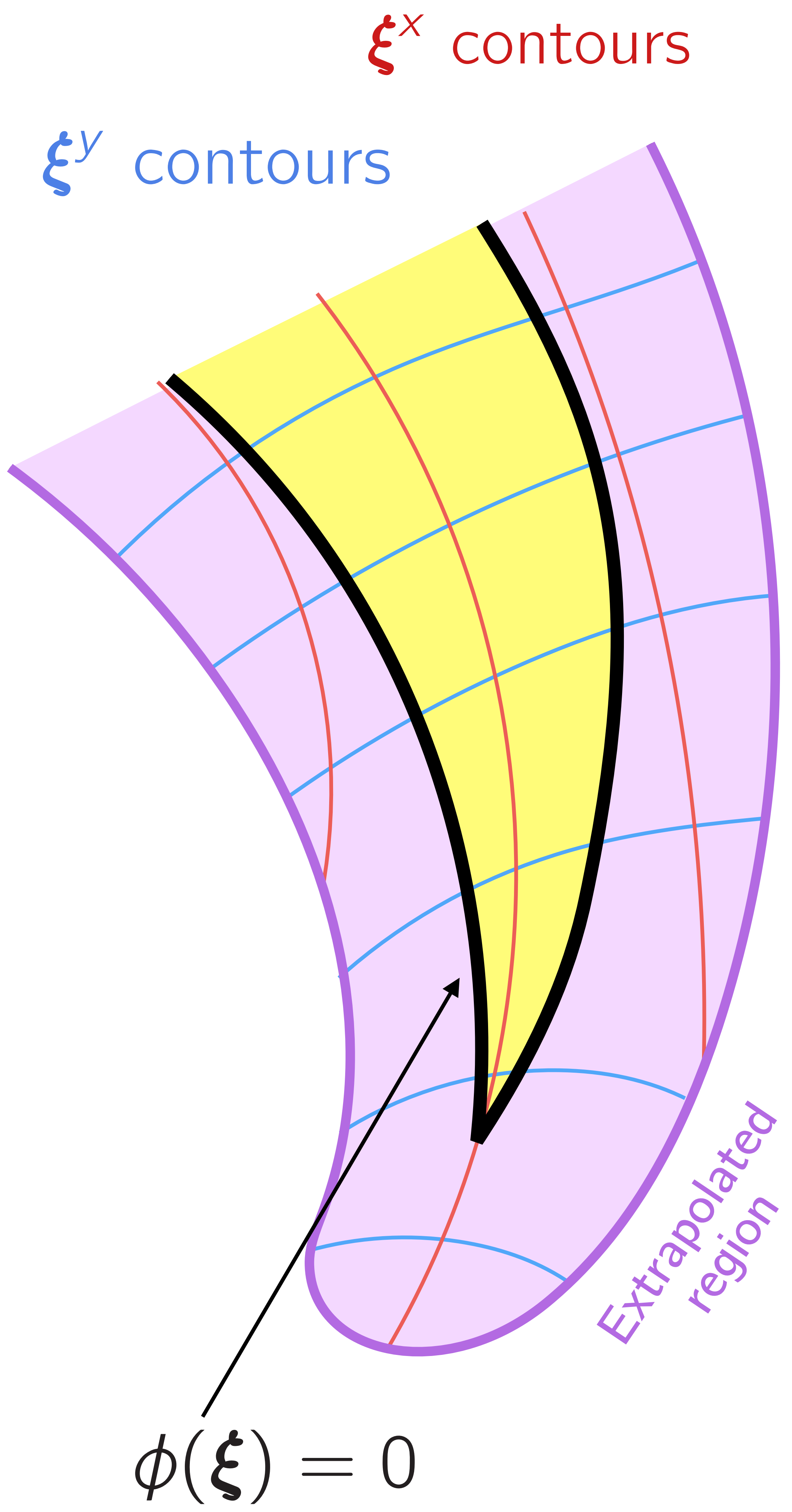


C. H. Rycroft, C.-H. Wu,
Y. Yu, and K. Kamrin,
J. Fluid Mech **898**, A9 (2020).

Example 2D computation

(All simulations use non-dimensionalized units)
(Colors show vorticity)





Has sharp corner even though reference map is smooth

C. H. Rycroft, C.-H. Wu, Y. Yu, and K. Kamrin, J. Fluid Mech **898**, A9 (2020).

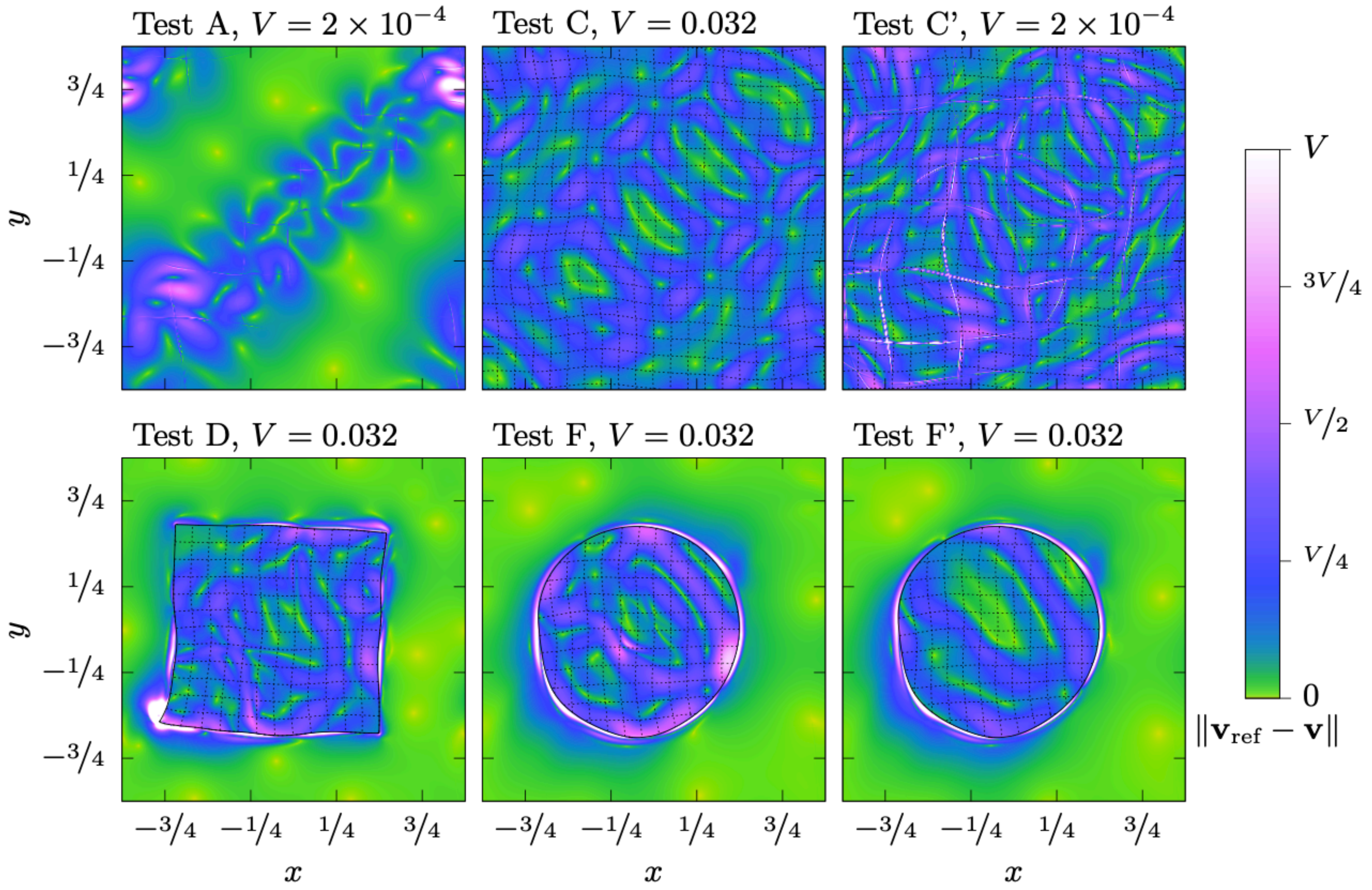
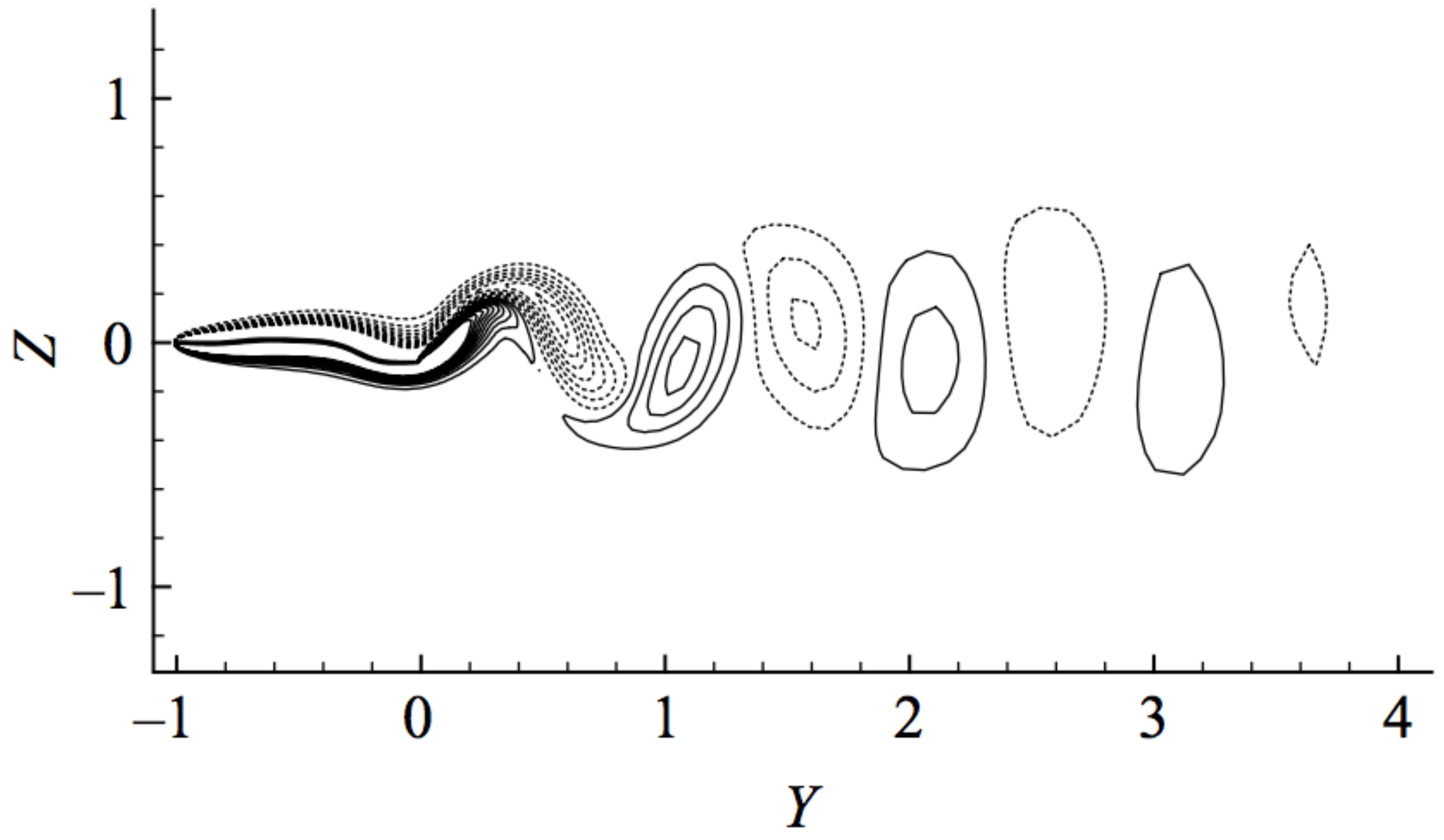
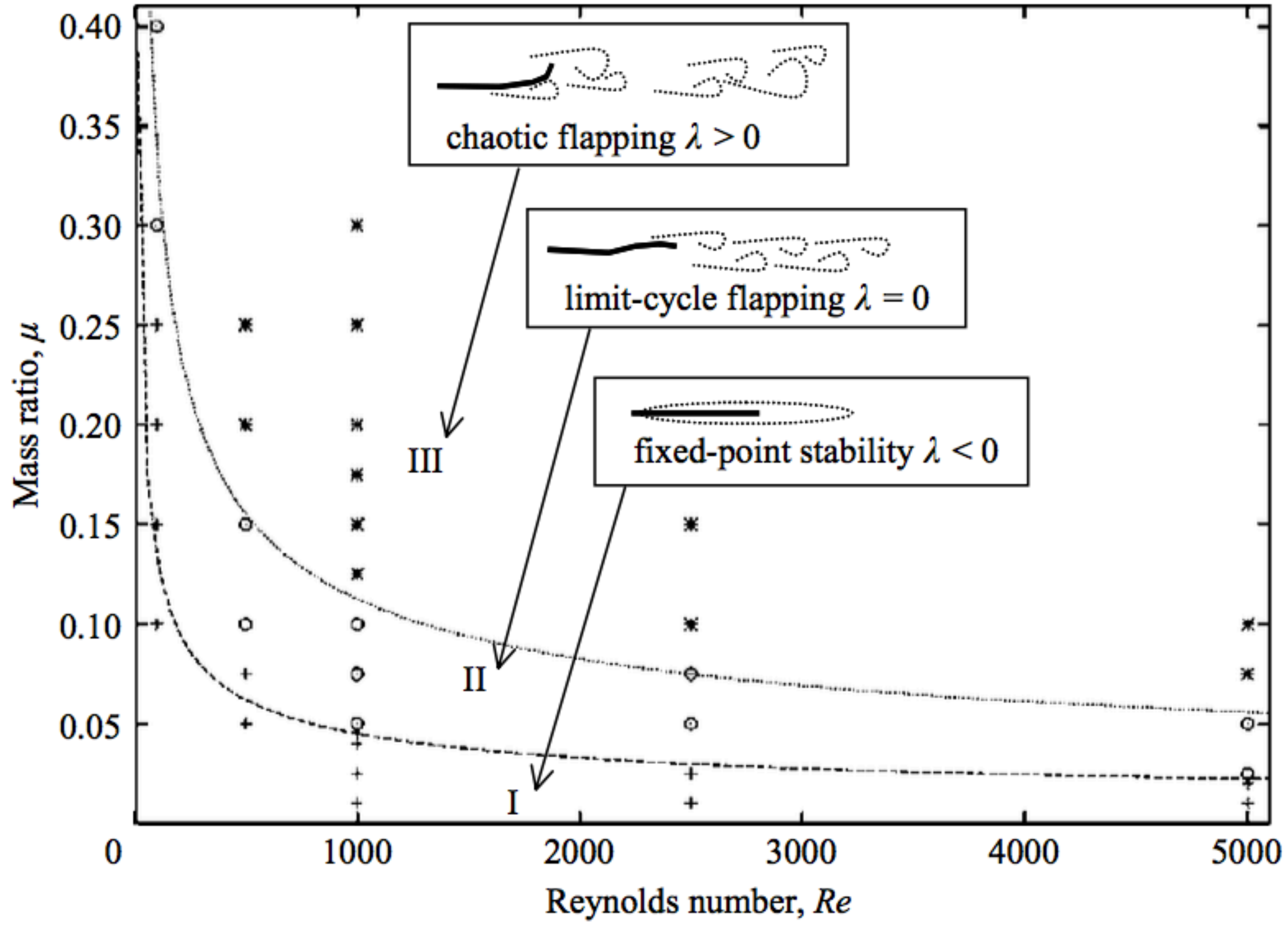


FIGURE 13. Differences between the velocity fields in the reference simulation (using a 5040×5040 grid) and the coarsest simulation (using a 360×360 grid). Plots are shown at $t = 0.5$ for six of the converegence tests. The colors in each panel are normalized differently by a maximum value V. The thick black lines mark the fluid–structure interfaces. The thin dashed lines are contours of the components of the reference map. Simulation parameters are $(\rho_f, \mu_f, \rho_s, G) = (1, 10^{-3}, 1, 1)$.

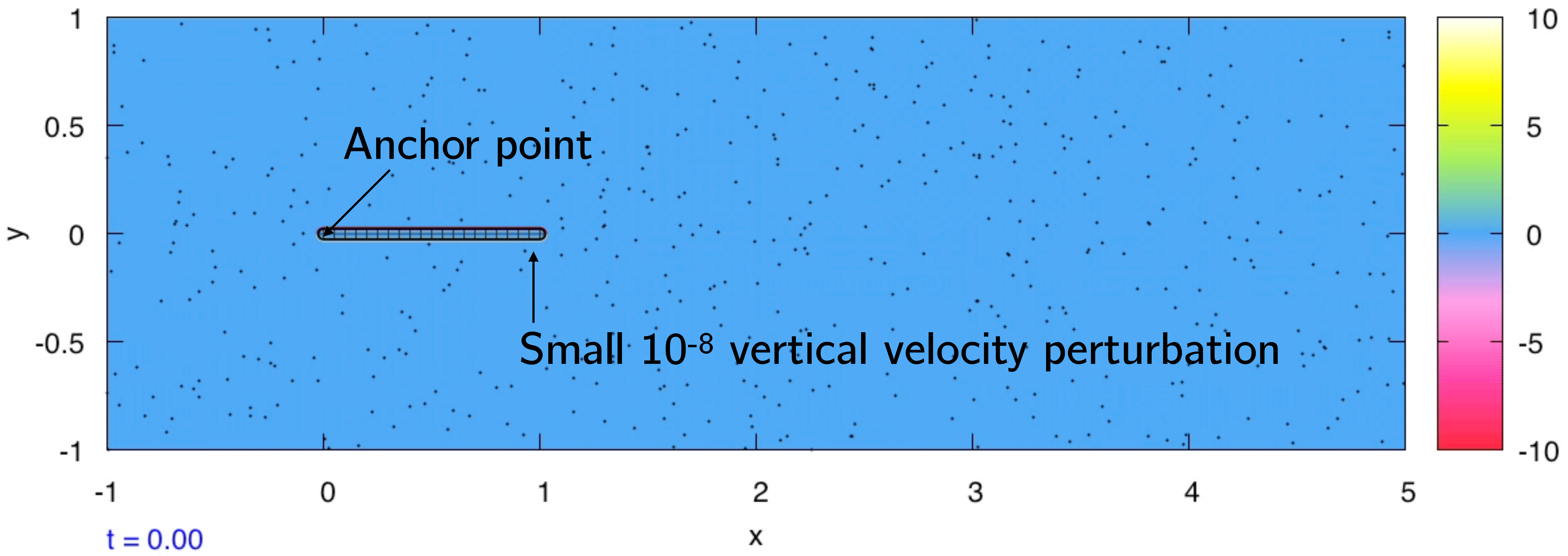
Flag flapping stability
(Connell & Yue, J. Fluid Mech. 2007)

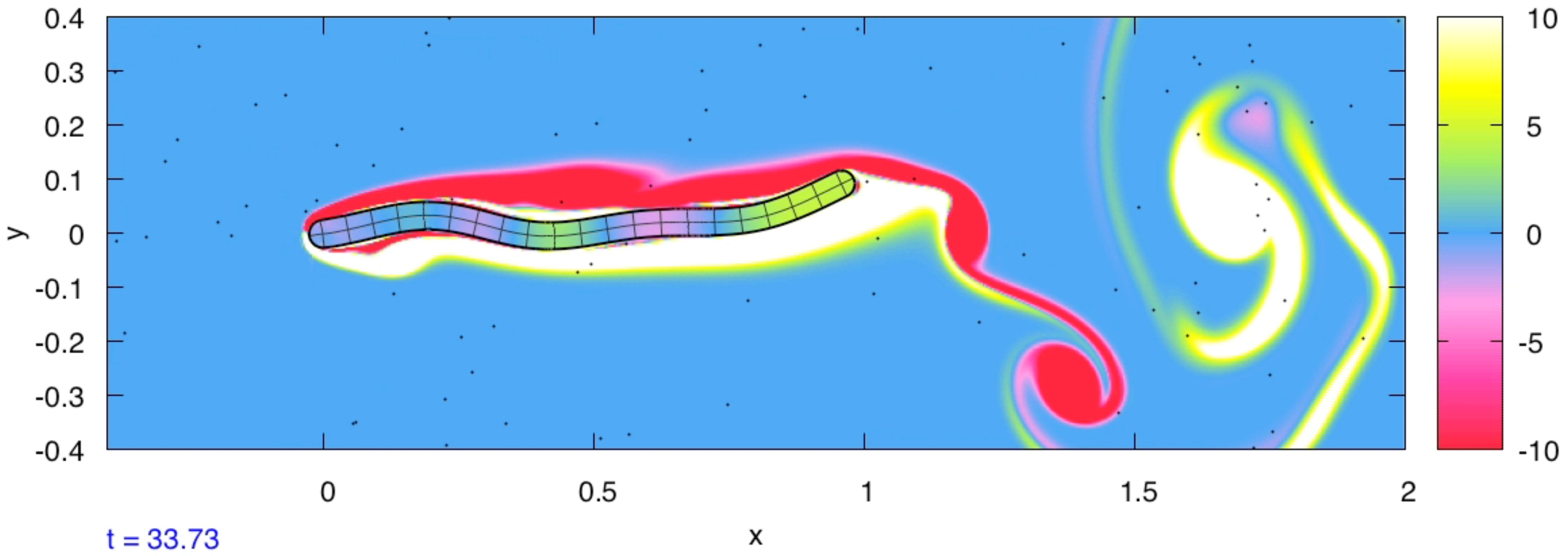
- Examine stability of flapping flag in terms of four dimensionless parameters:
  - Reynolds number, Re
  - Bending rigidity, k_B
  - Mass ratio, mu
  - (Aspect ratio)





Flag flapping ($mu=0.46$, $k_B=0.001$, $Re=3000$)

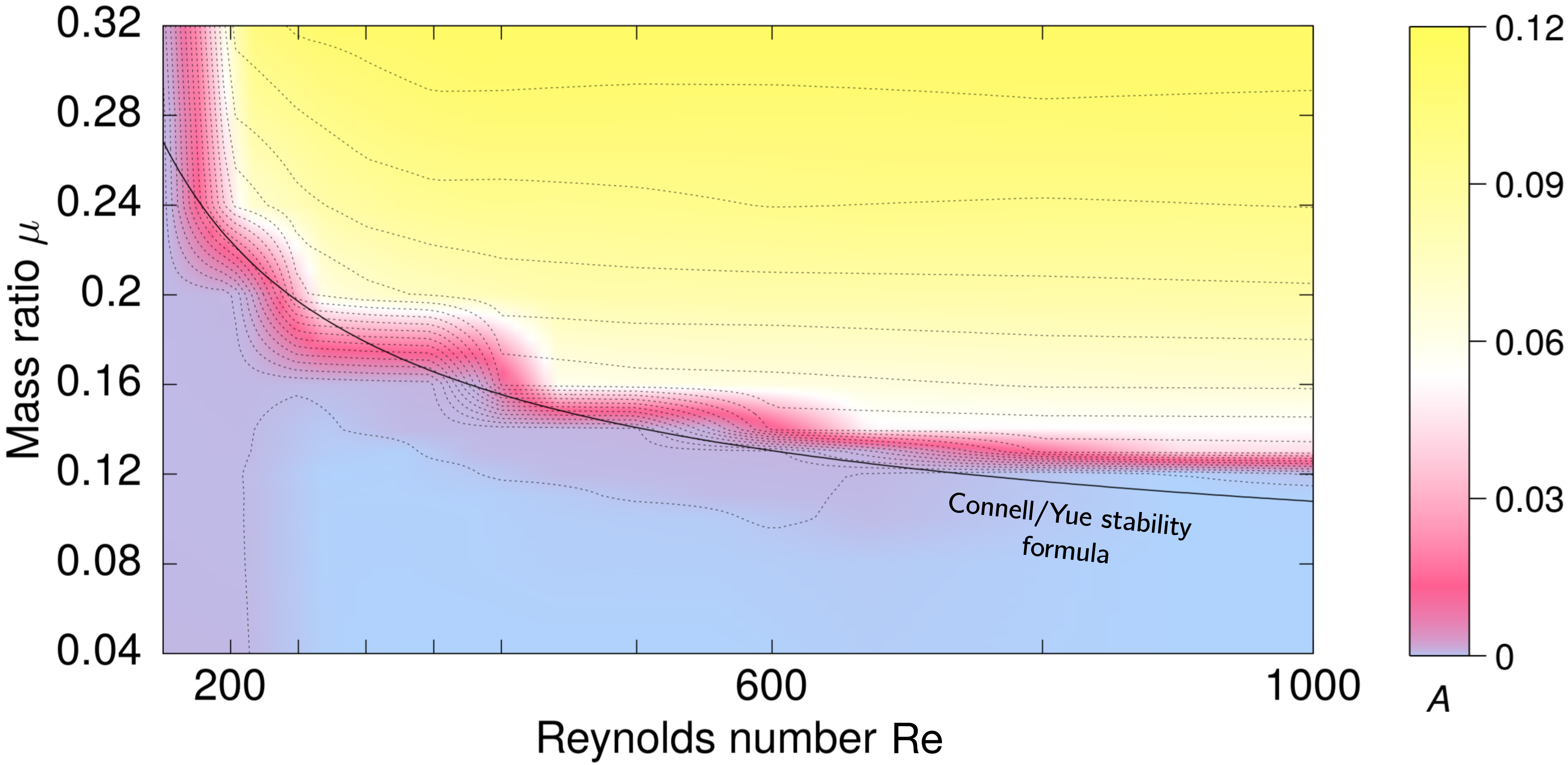




(Colors show vorticity)

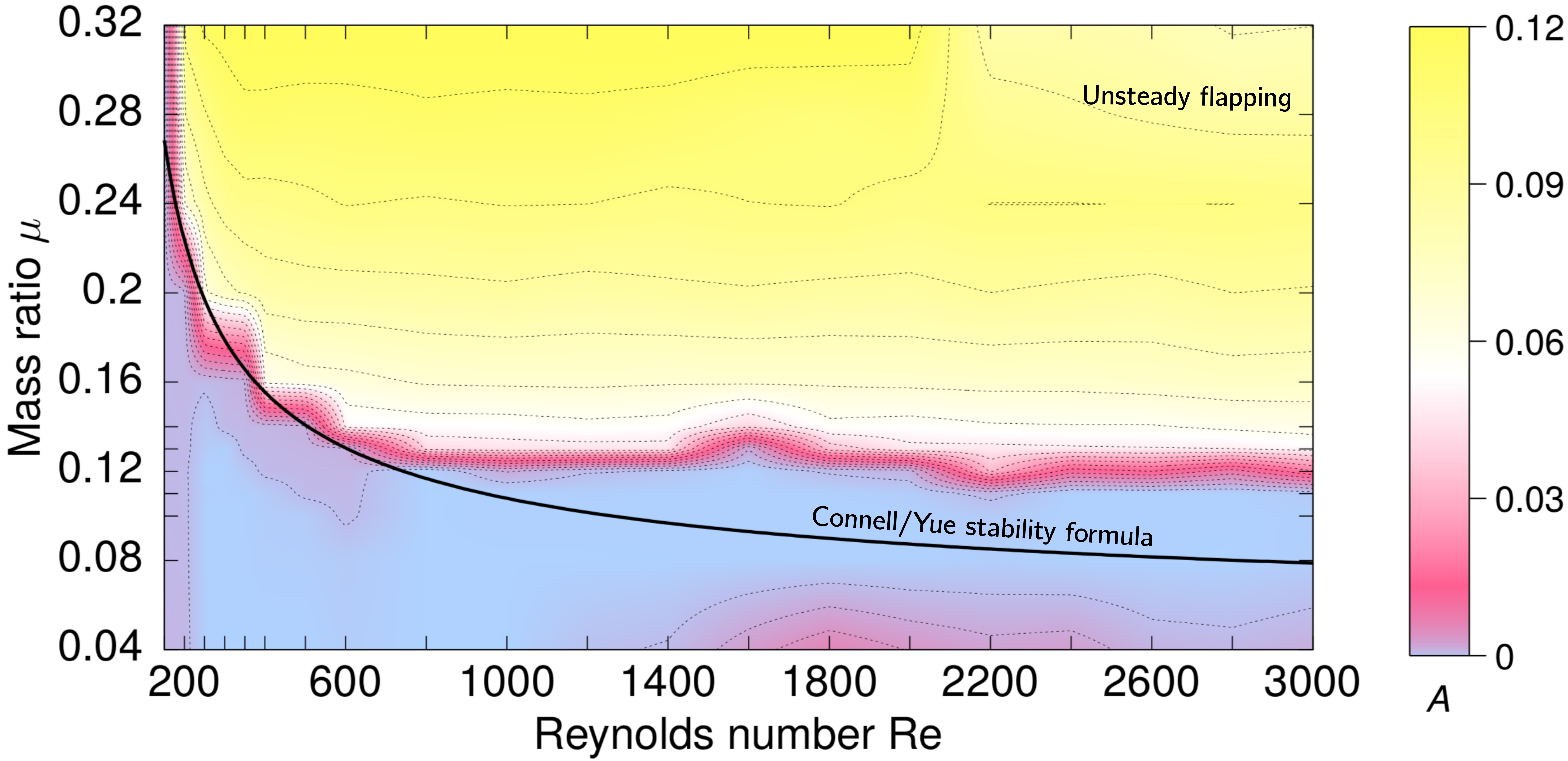
Flapping amplitude of the tail

- Track perpendicular displacement of tail from $t = 120$ to $t = 160$
- Apply Fourier transform to find flapping amplitude A

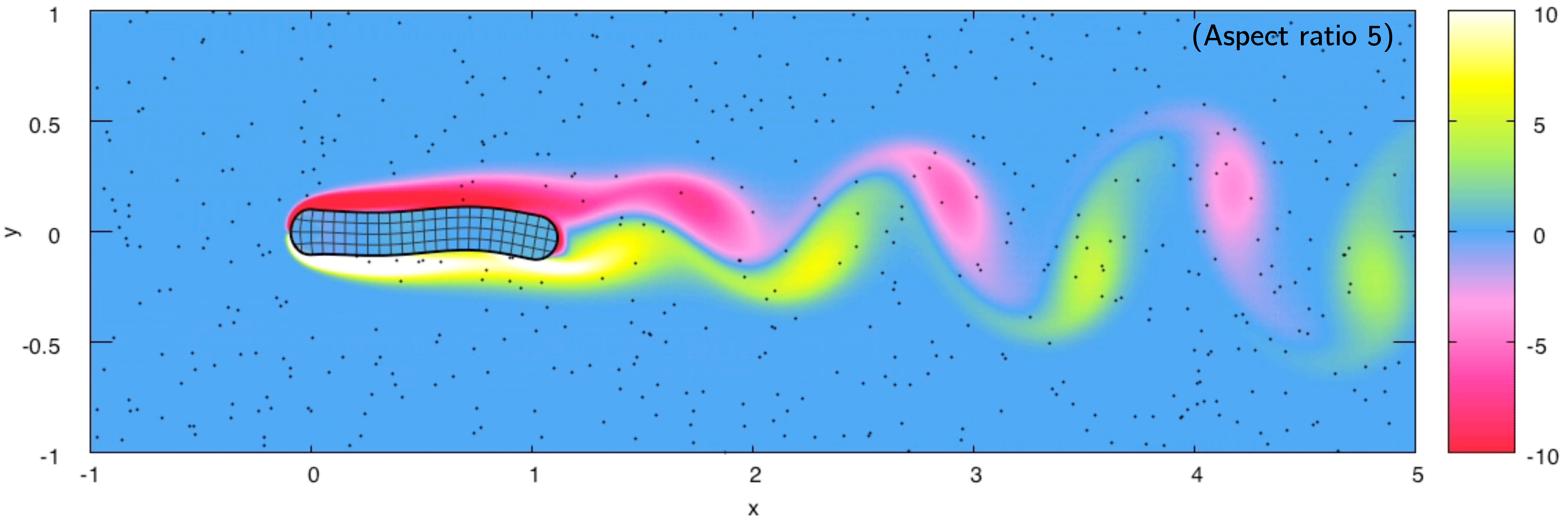


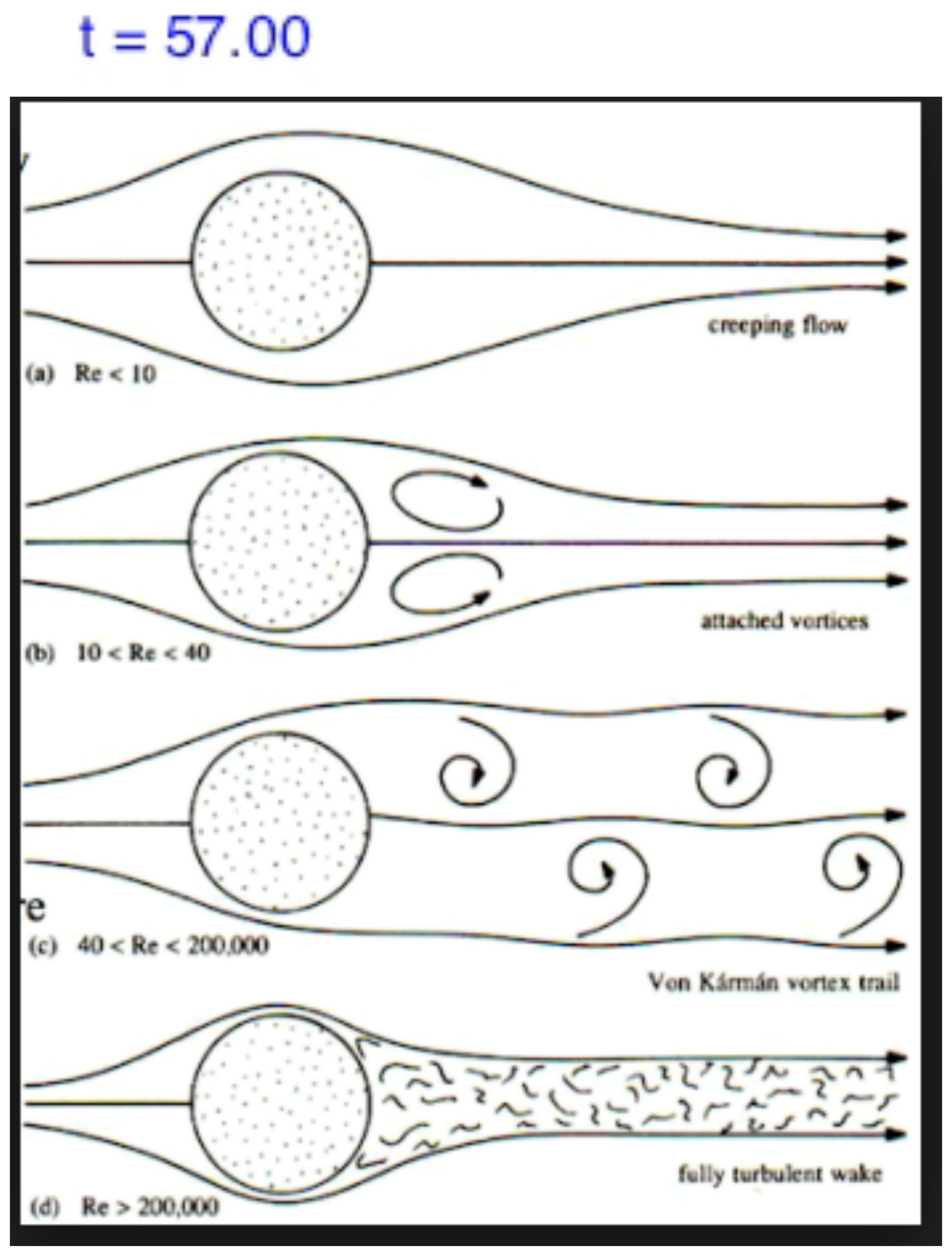
Flapping amplitude of the tail

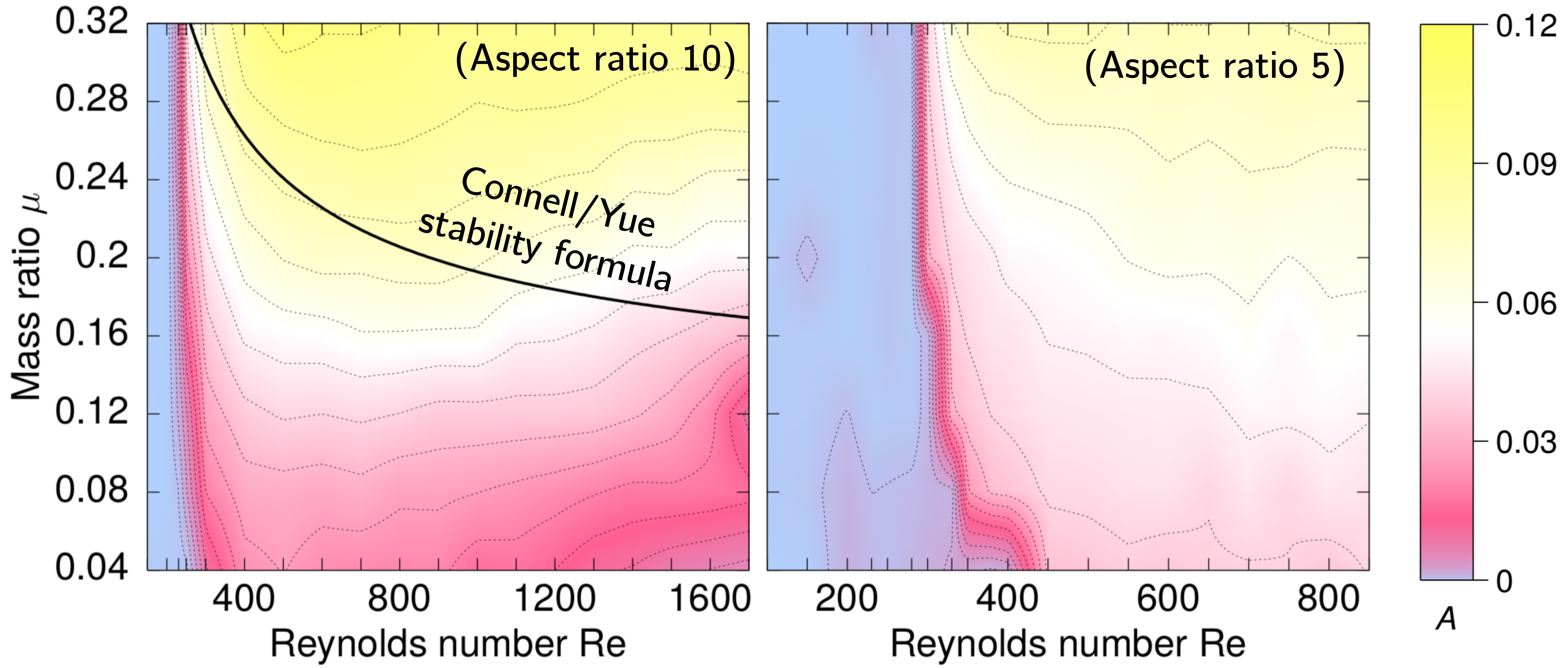
- Track perpendicular displacement of tail from $t = 120$ to $t = 160$
- Apply Fourier transform to find flapping amplitude A



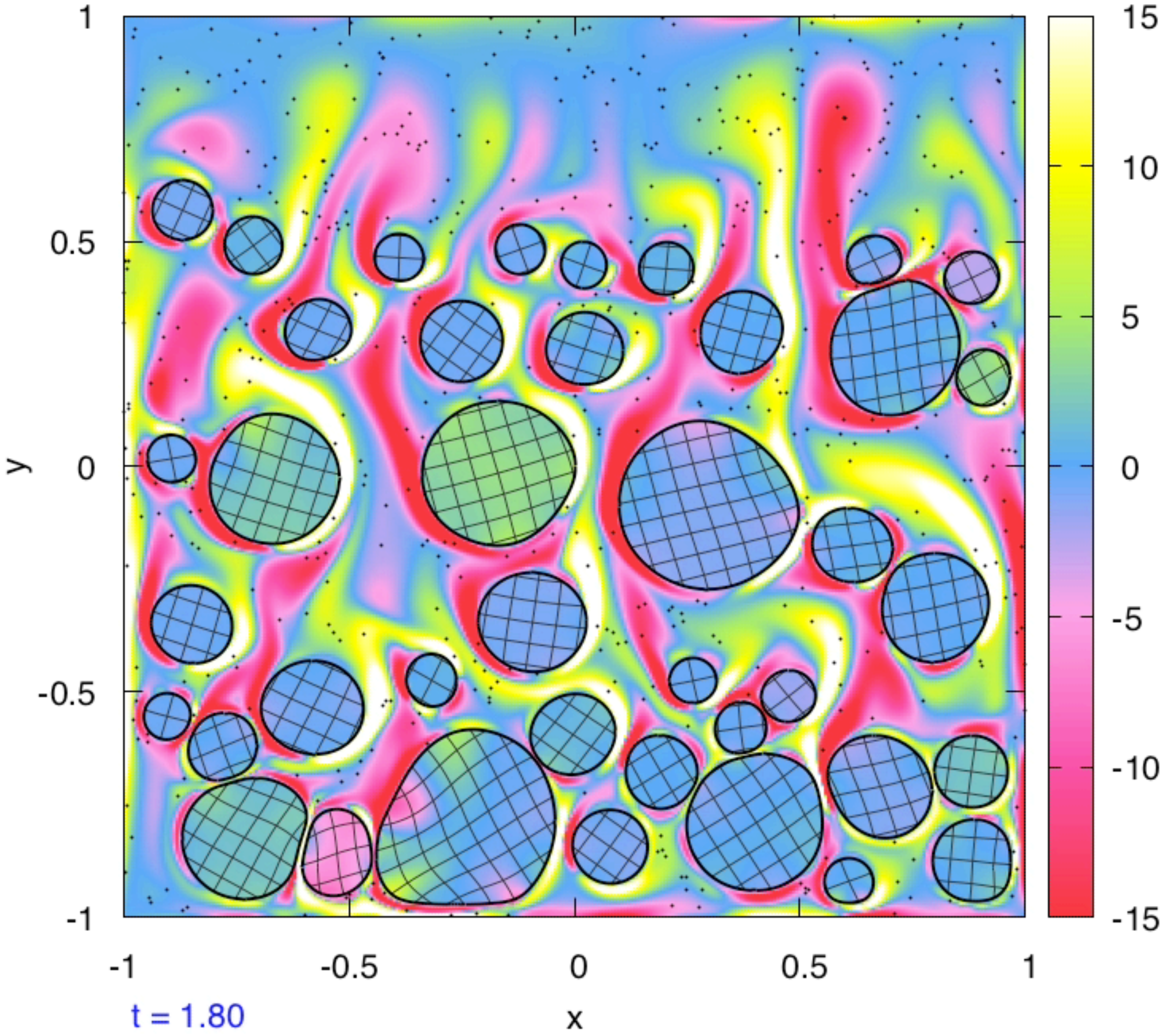
Bulky flappers





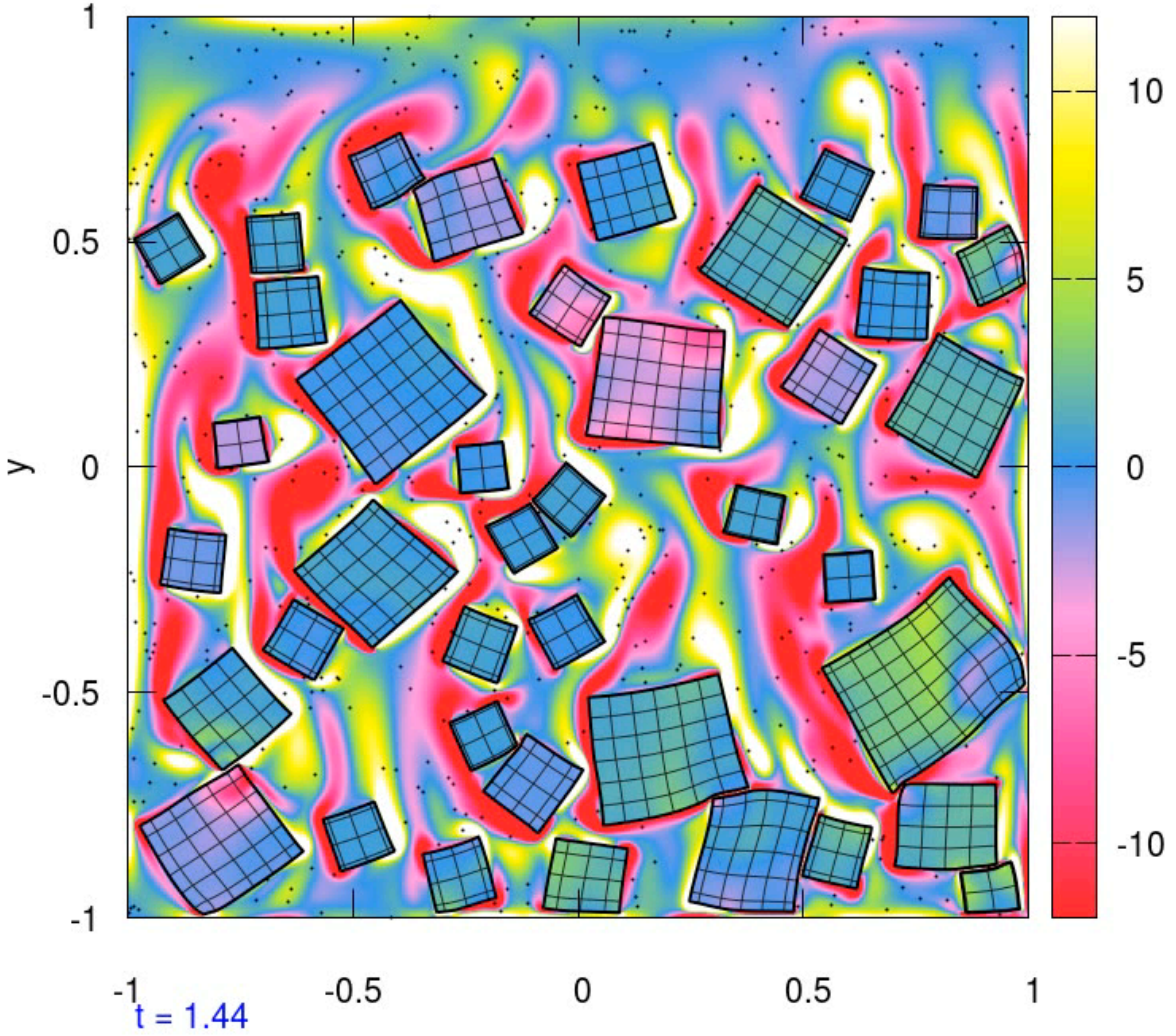


Multi-body contact



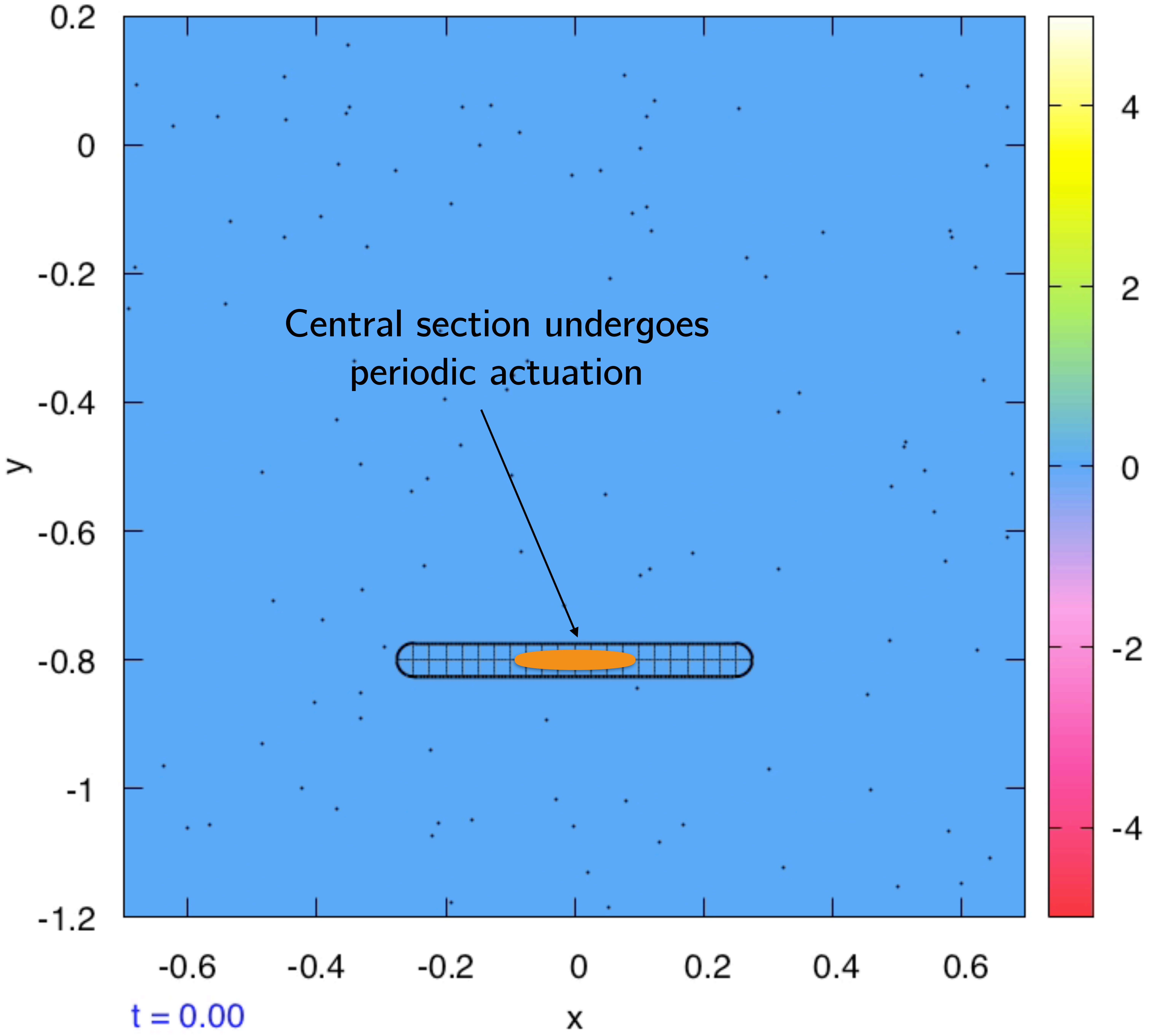
(Colors show vorticity)

Multi-body contact



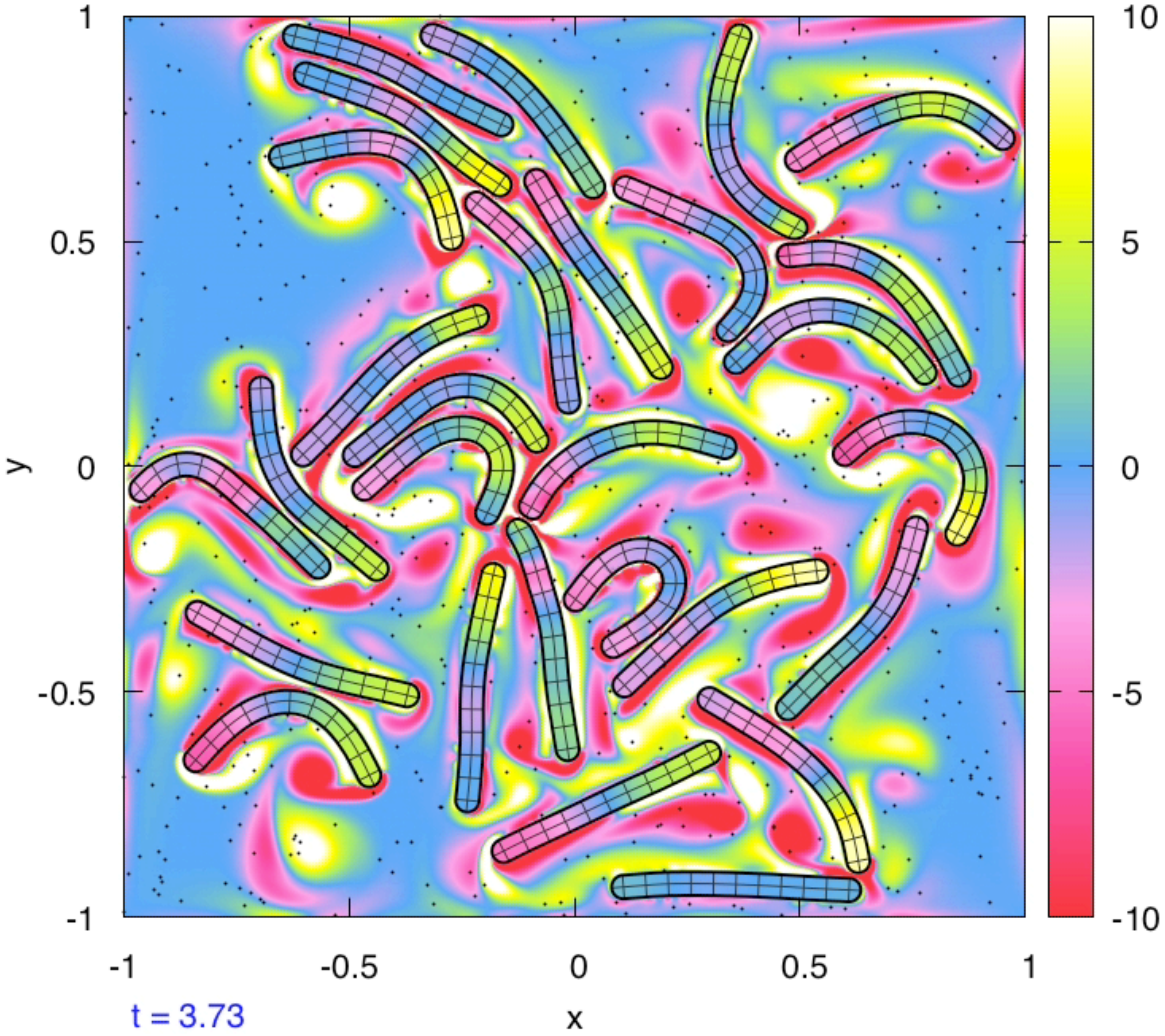
(Colors show vorticity)

Actuated solids for biolocomotion



(Colors show vorticity)

Active fluid model

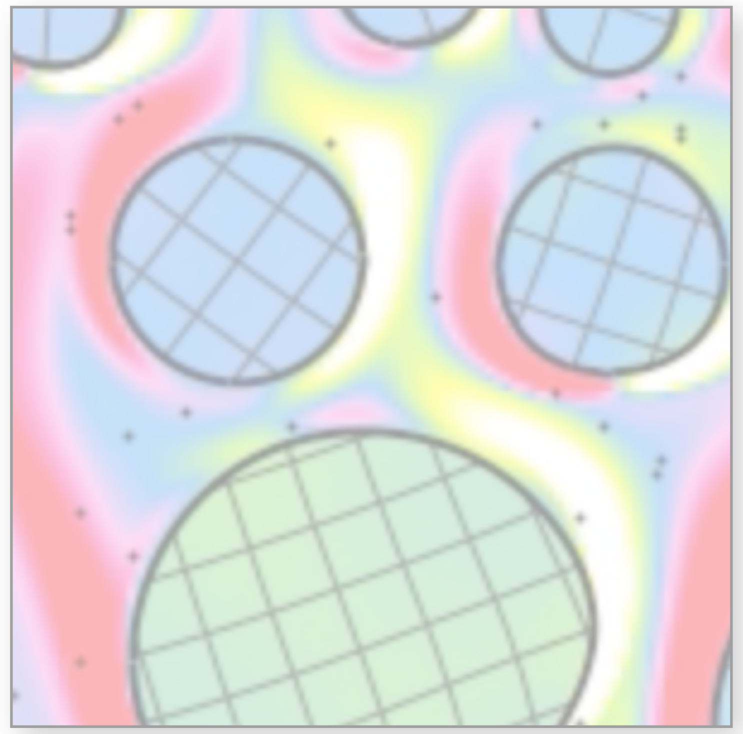


(Colors show vorticity)

Talk outline

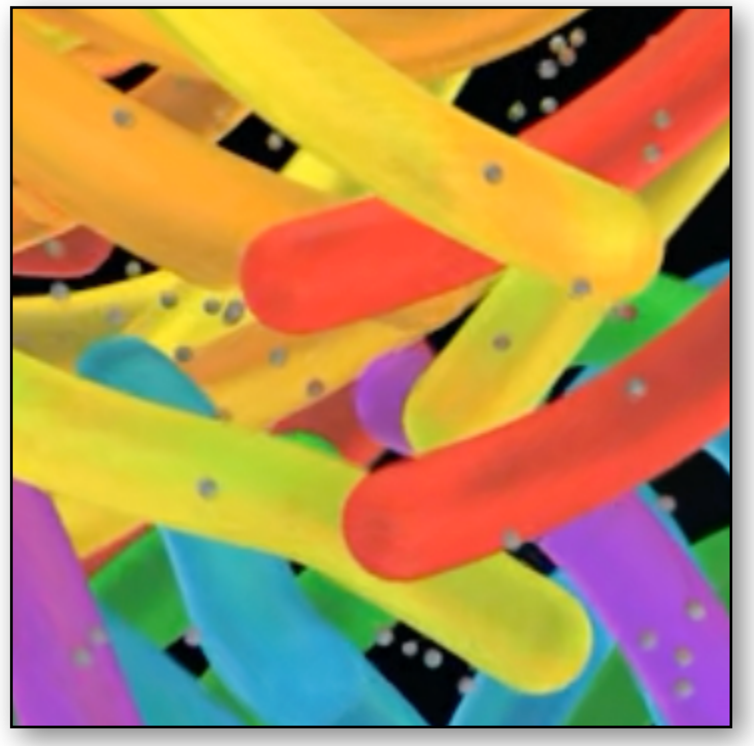
1. The reference map technique

A conceptual overview of the method



2. Reference map simulation for incompressible fluids

A numerical implementation for simulating flag-flapping and multi-body contact

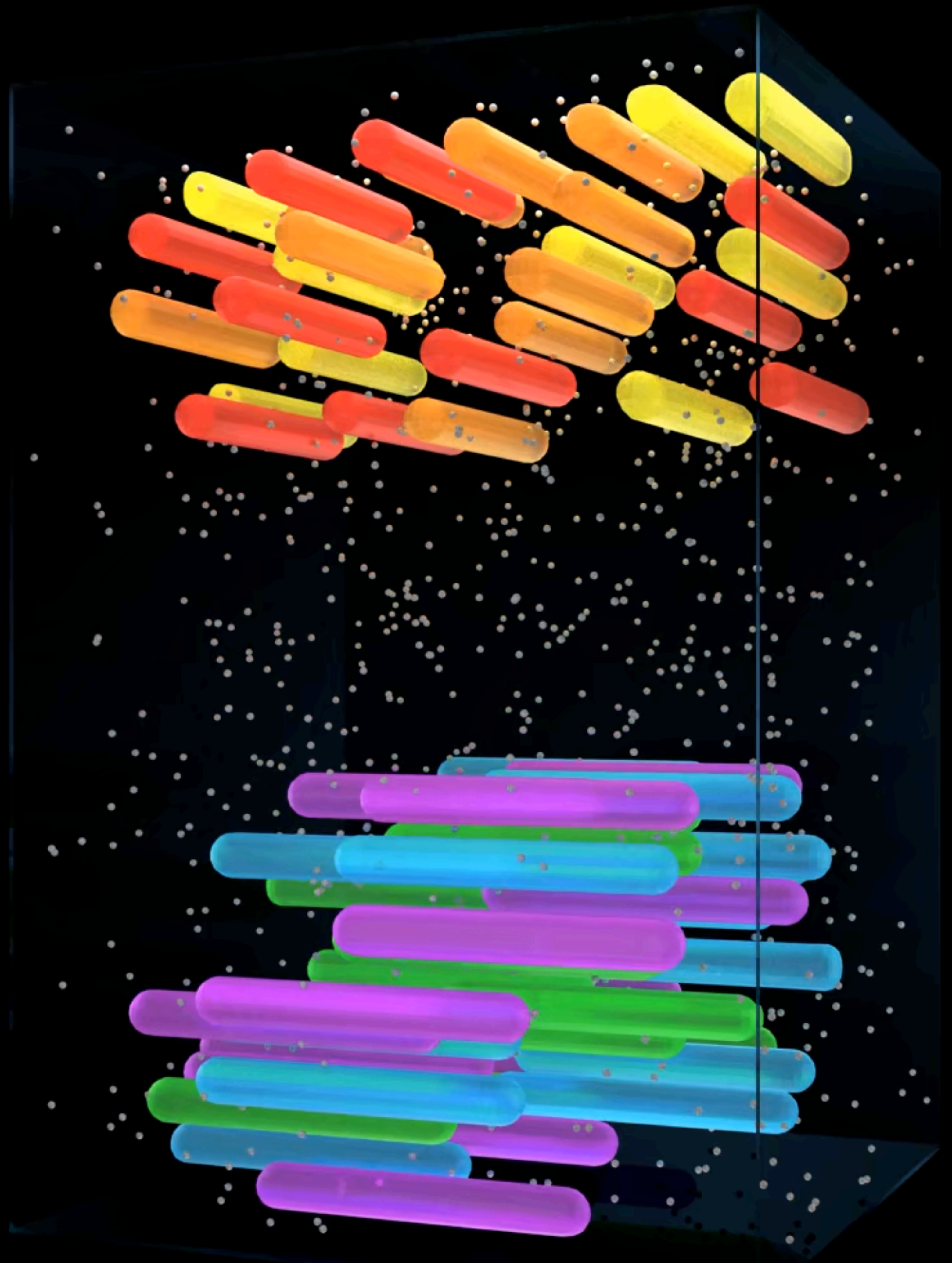


3. Applications and extensions

Three-dimensional simulation, higher-order discretizations

Three-dimensional simulation

- Methods naturally generalize to 3D simulation, but many computational challenges:
  - Run in parallel using domain decomposition. Use Message-Passing Interface (MPI) library to communicate between processors
  - Key challenge in projection step to solve very large linear systems in parallel—use custom multigrid solver
  - Multiple shapes leads to multiple reference maps—requires special storage

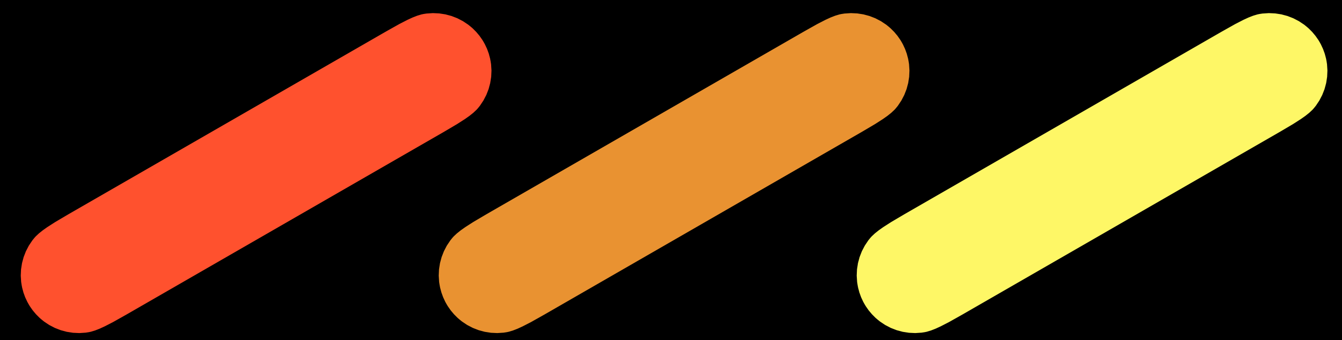


Central cross-section

(8 processors, 96x96x144 grid)
(3D rendering by Yue Sun)

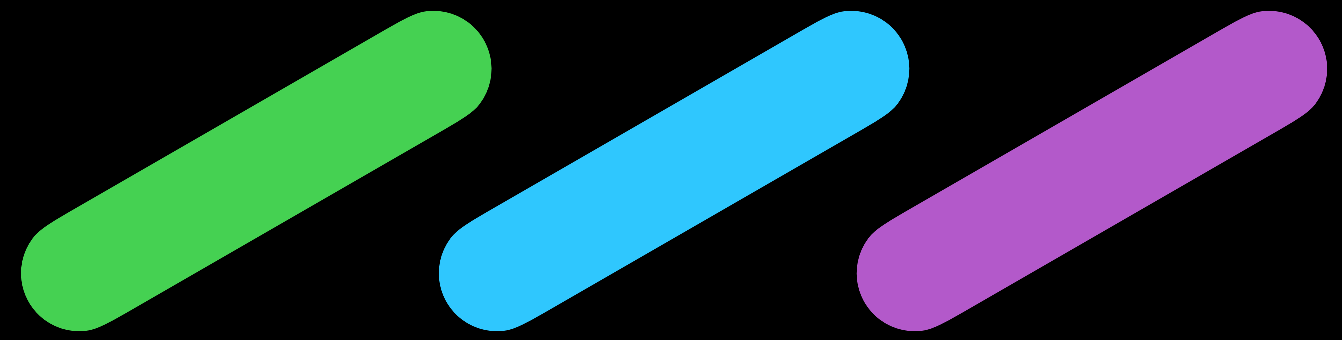
Y. L. Lin, N. J. Derr, and C. H. Rycroft, Proc. Natl. Acad. Sci. **119**, e2105338118 (2022).

Three-dimensional simulation



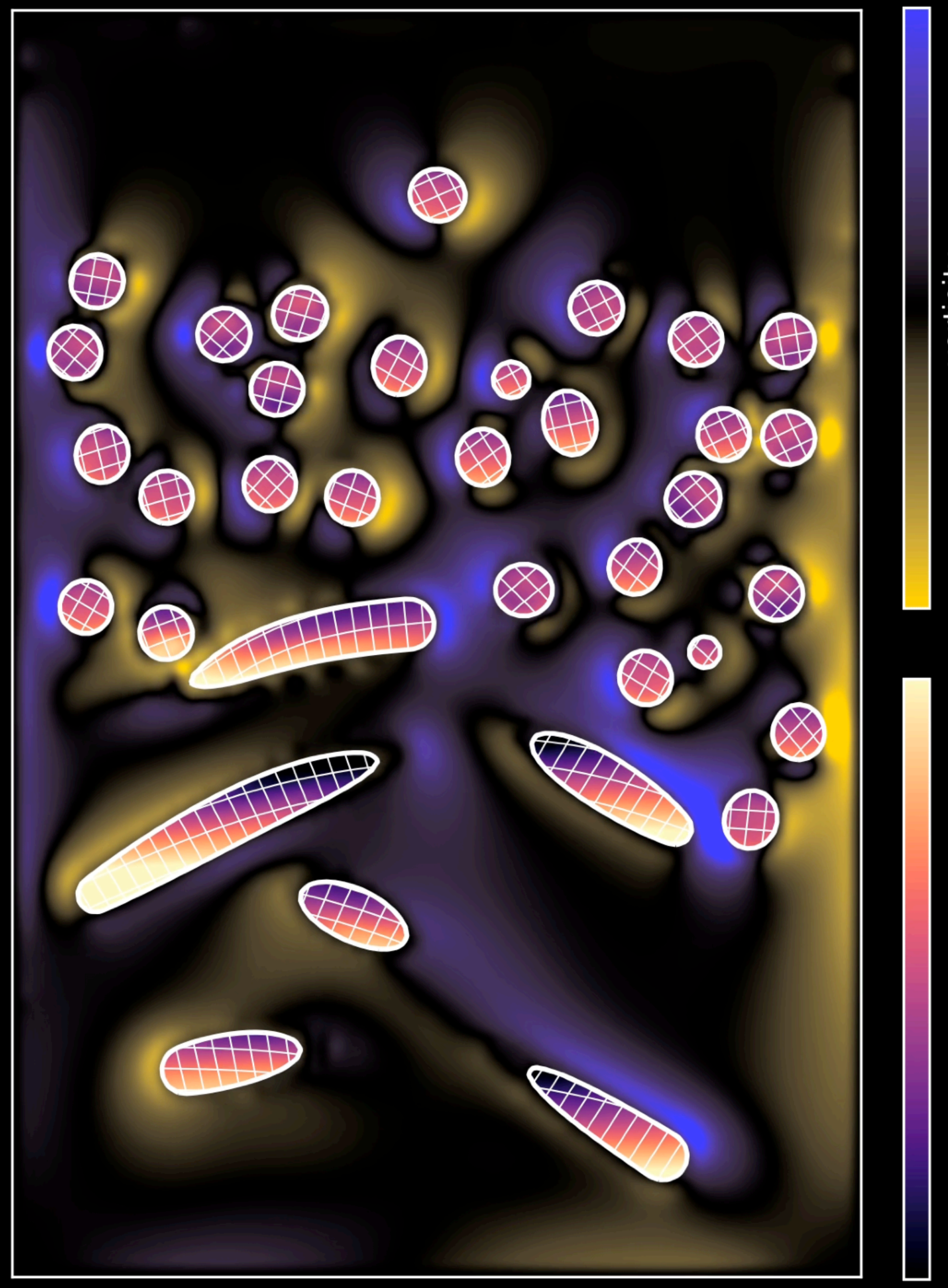
Heavy rods

(density 1.25)



Buoyant rods

(density 0.8)



Central cross-section

(8 processors, 96x96x144 grid)
(3D rendering by Yue Sun)

Y. L. Lin, N. J. Derr, and C. H. Rycroft, Proc. Natl. Acad. Sci. **119**, e2105338118 (2022).

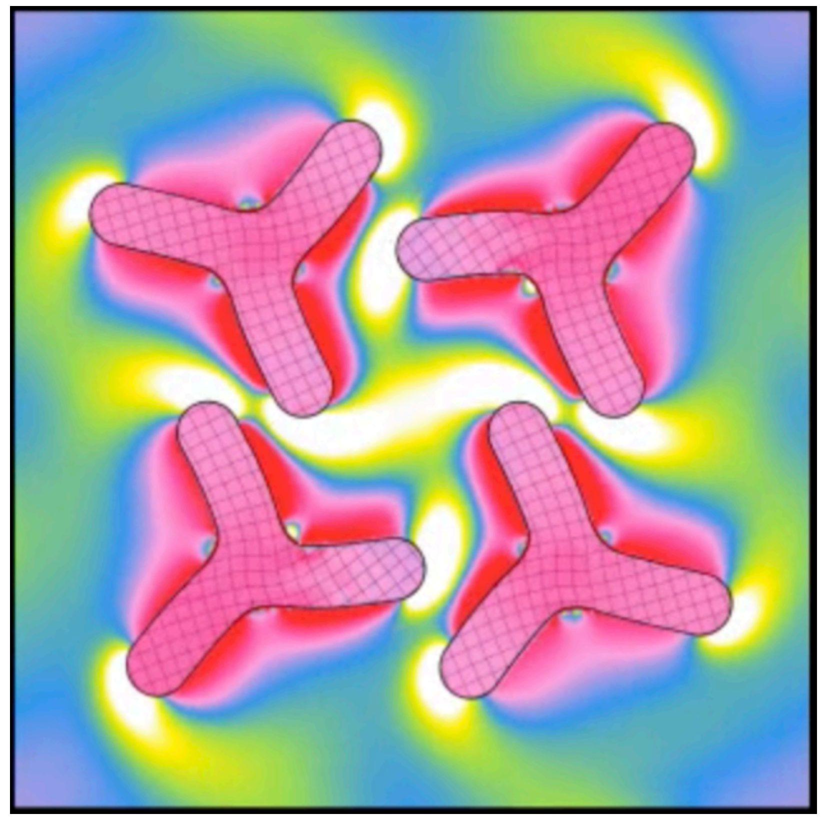
Conclusions

- The reference map technique is a simple, flexible technique for simulating finite-strain solid mechanics on a fixed grid
- Many applications for fluid–structure interaction, many body contact, and modeling of complex biological materials

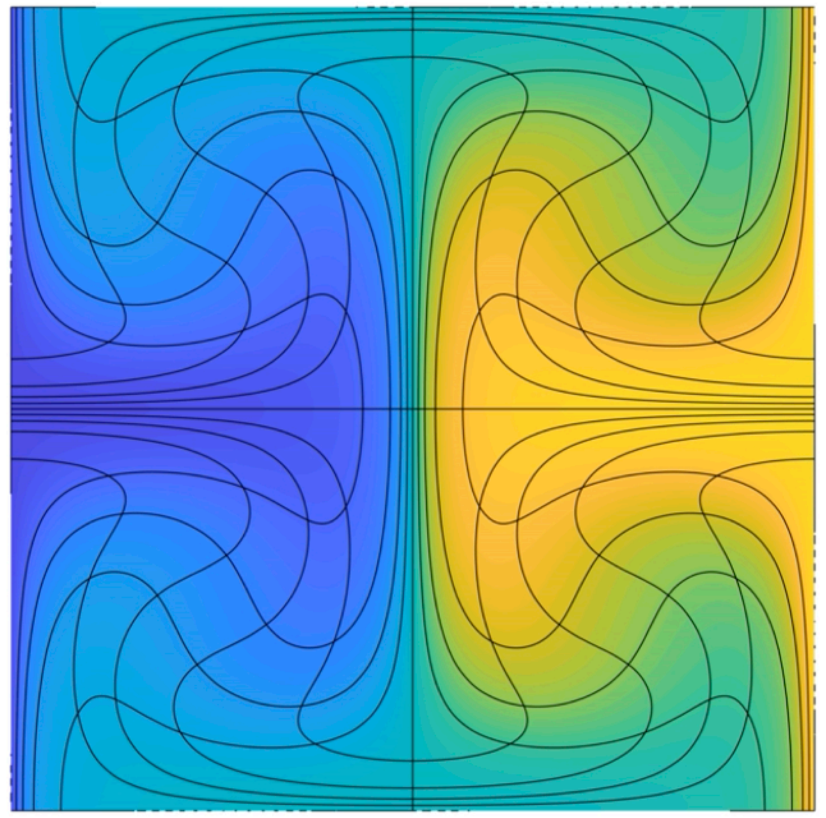
Chris Rycroft
chr@math.wisc.edu
https://people.math.wisc.edu/~chr

K. Kamrin, C. H. Rycroft, and J.-C. Nave, J. Mech. Phys. Solids **60**, 1952–1969 (2012).
B. Valkov, C. H. Rycroft, and K. Kamrin, J. Appl. Mech. **82**, 041011 (2015).
C. H. Rycroft, C.-H. Wu, Y. Yu, and K. Kamrin, J. Fluid Mech **898**, A9 (2020).
X. Wang, K. Kamrin, and C. H. Rycroft, Phys. Fluids 34, 032604 (2022).
Y. L. Lin, N. J. Derr, and C. H. Rycroft, Proc. Natl. Acad. Sci. **119**, e2105338118 (2022).
https://people.math.wisc.edu/~chr/events/rmt_talk.html
Y. Sun *et al.*, Gallery of Fluid Motion, V0045 (2021).

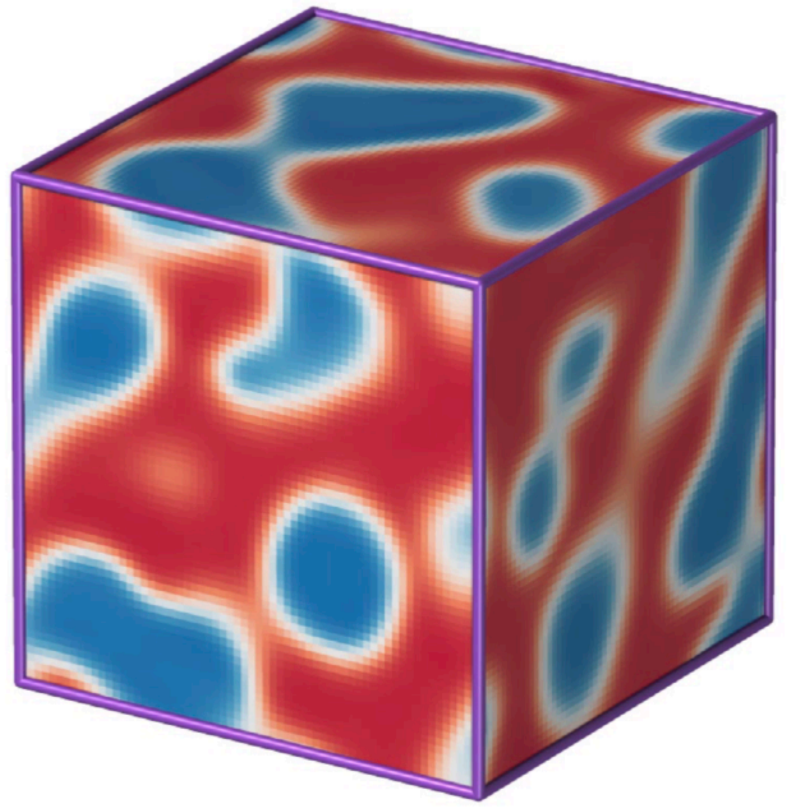
Next steps



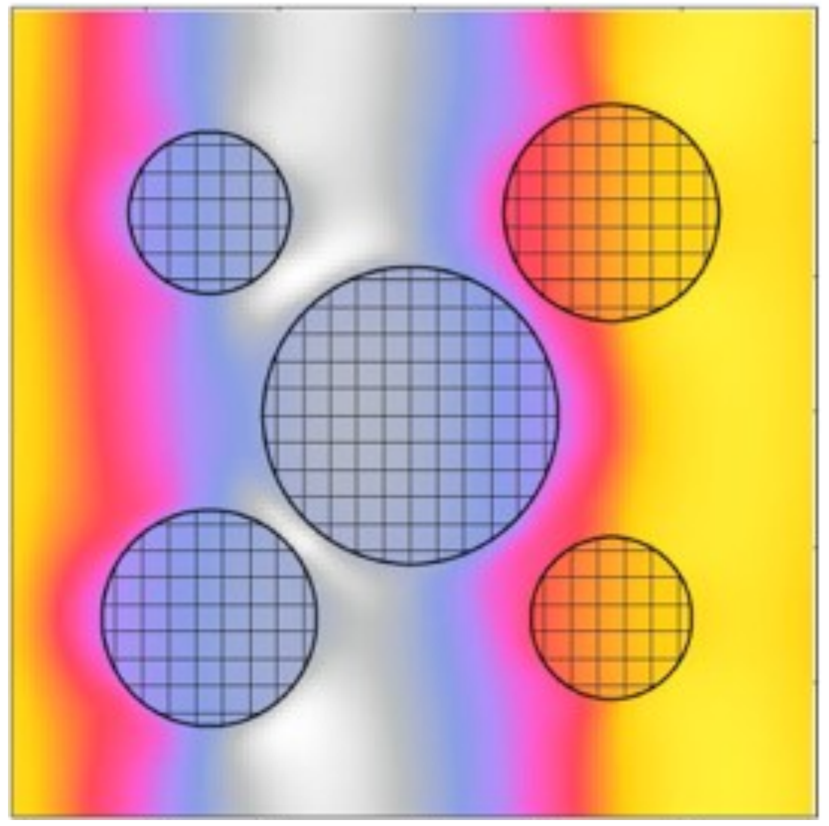
Lattice Boltzmann implementation
(By Yue Sun)



Discontinuous Galerkin implementation
(By Dan Fortunato)



Simulation and modeling of poroelastic materials
(By Nicholas Derr)



Modeling sea ice dynamics
(With Nadiya Mahomed, Alfred Bogaers, and Sebastian Skatulla)