March 12-16, 2018

ICERM Workshop on "Fast Algorithms for Generating Static and Dynamically Changing Point Configurations"

Improving Particle Methods

Robert Krasny (Michigan)

collaborators

Weihua Geng (Southern Methodist University)

Peter Bosler (Sandia National Laboratories)

Ling Xu (Michigan)

support

NSF DMS-1418966

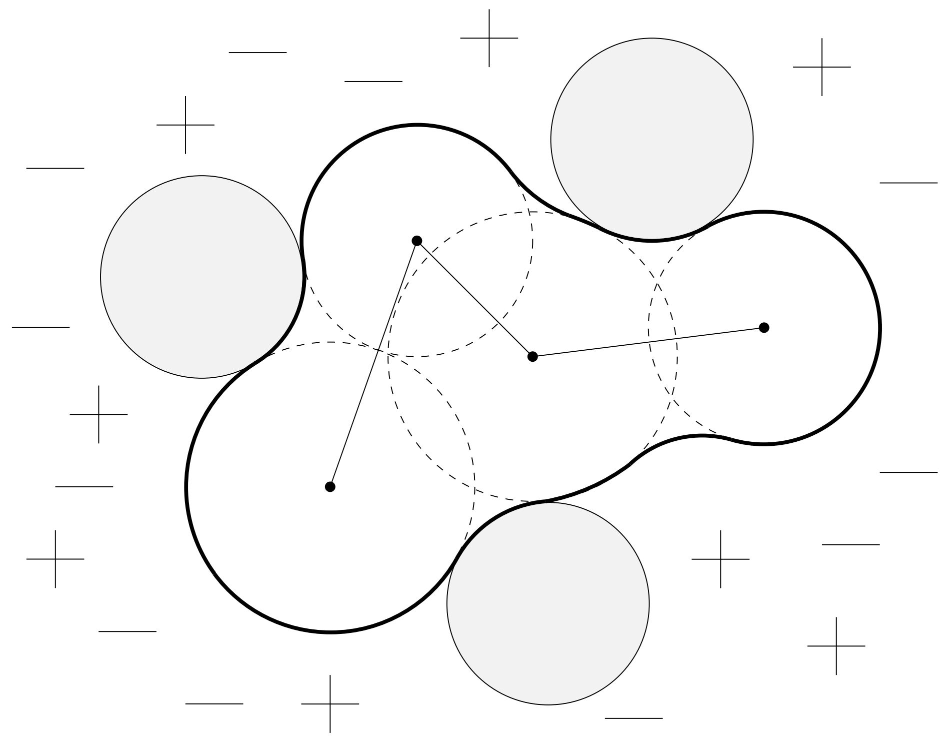
ONR N00014-14-1-0075

MICDE (Michigan Institute for Computational Discovery and Engineering)

outline

1. protein-solvent electrostatics
2. incompressible fluid flow

1. protein-solvent electrostatics



atomic charges

Van der Waals radius

water molecules

molecular surface

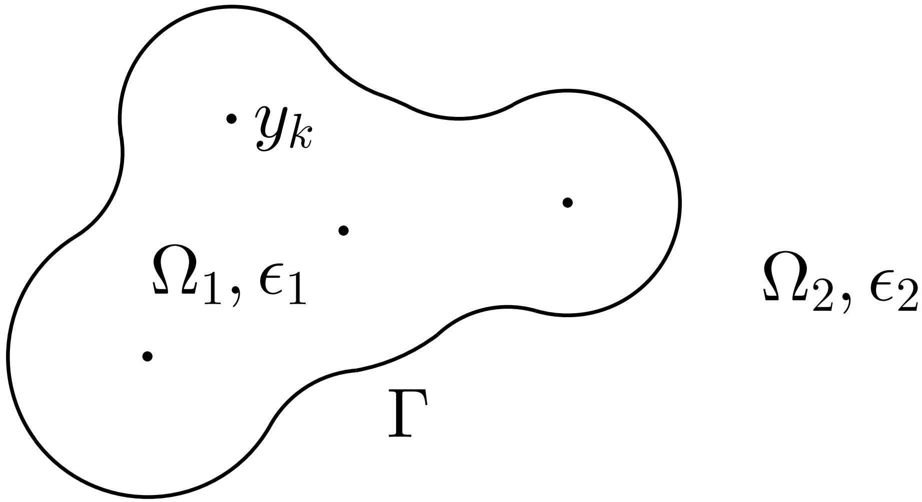
dissolved salt ions

goal

$\phi(x)$: electrostatic potential

E_{sol} : electrostatic solvation energy

Poisson-Boltzmann implicit solvent model



Ω_1 : protein domain : $\epsilon_1 \nabla^2 \phi(x) = -\sum_{k=1}^{N_c} q_k \delta(x - y_k)$

Ω_2 : solvent domain : $\epsilon_2 \nabla^2 \phi(x) - \bar{\kappa}^2 \phi(x) = 0$: PB equation

Γ : molecular surface : $\phi_1 = \phi_2$, $\epsilon_1 \frac{\partial \phi_1}{\partial n} = \epsilon_2 \frac{\partial \phi_2}{\partial n}$

far-field boundary condition : $\lim_{|x| \to \infty} \phi(x) = 0$

boundary integral formulation Juffer et al. (1991)

$$G_0(x,y) = \frac{1}{4\pi|x-y|} \ : \text{ Coulomb potential}$$

$$G_\kappa(x,y) = \frac{e^{-\kappa|x-y|}}{4\pi|x-y|} \ : \text{ screened Coulomb potential } , \ \kappa^2 = \bar{\kappa}^2/\epsilon_2$$

$$\tfrac{1}{2}\left(1+\tfrac{\epsilon_2}{\epsilon_1}\right)\phi(x) \ = \int_\Gamma \Big[K_1(x,y)\partial_n\phi(y) + K_2(x,y)\phi(y)\Big] dS_y + S_1(x)$$

$$\tfrac{1}{2}\left(1+\tfrac{\epsilon_1}{\epsilon_2}\right)\partial_n\phi(x) = \int_\Gamma \Big[K_3(x,y)\partial_n\phi(y) + K_4(x,y)\phi(y)\Big] dS_y + S_2(x)$$

$$K_1 = G_0 - G_\kappa \ , \qquad\qquad K_2 = \frac{\epsilon_2}{\epsilon_1}\partial_{n_y}G_\kappa - \partial_{n_y}G_0$$

$$K_4 = \partial^2_{n_x n_y}(G_\kappa - G_0) \ , \ K_3 = \partial_{n_x}G_0 - \tfrac{\epsilon_1}{\epsilon_2}\partial_{n_x}G_\kappa$$

$$S_1(x) = \frac{1}{\epsilon_1}\sum_{k=1}^{N_c} q_k G_0(x,y_k) \ , \ S_2(x) = \frac{1}{\epsilon_1}\sum_{k=1}^{N_c} q_k \partial_{n_x} G_0(x,y_k)$$

discretization

triangulation + centroid collocation

x_i : centroid , A_i : area , $i = 1 : N$

$$\frac{1}{2}\left(1+\frac{\epsilon_2}{\epsilon_1}\right)\phi_i = \sum_{\substack{j=1 \\ j\neq i}}^{N} \Big[K_1(x_i,x_j)\partial_n\phi_j + K_2(x_i,x_j)\phi_j\Big]A_j + S_1(x_i)$$

$$\frac{1}{2}\left(1+\frac{\epsilon_1}{\epsilon_2}\right)\partial_n\phi_i = \sum_{\substack{j=1 \\ j\neq i}}^{N} \Big[K_3(x_i,x_j)\partial_n\phi_j + K_4(x_i,x_j)\phi_j\Big]A_j + S_2(x_i)$$

$\Rightarrow$ linear system : $Ax = b$

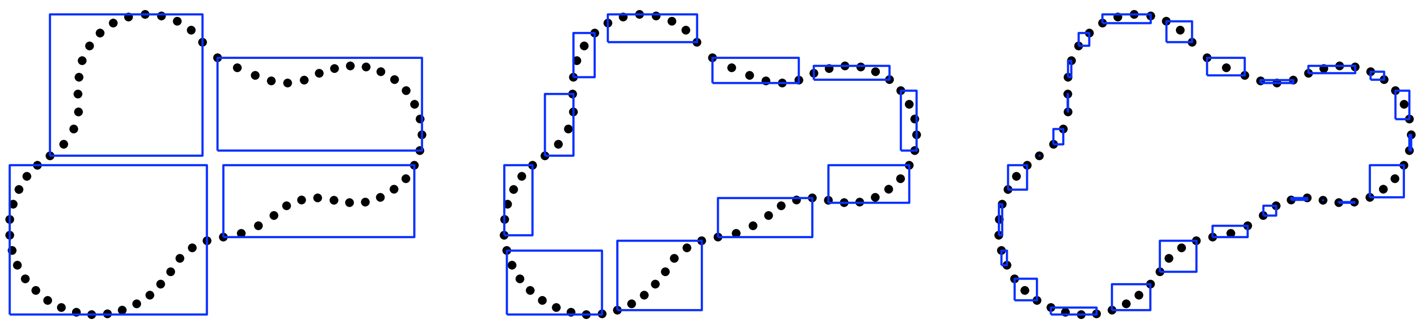
- GMRES
- matrix-vector product at each step
- fast summation $\rightarrow$ treecode

treecode : higher-order version of Barnes-Hut (1986)

$$\phi_i = \sum_{\substack{j=1 \\ j\neq i}}^{N} q_j G(x_i, x_j) \ , \ \ i = 1, \ldots, N \ : \ \text{particle-particle} \ , \ O(N^2)$$

$$\approx \sum_c \sum_{k=0}^{p} a_k(x_i, x_c) M^k(c) \ : \ \text{particle-cluster} \ , \ O(N \log N)$$

- Cartesian coordinates
- recurrence relations for $a_k(x_i, x_c) = \frac{1}{k!} D_y^k G(x, x_c)$
- geometrically adaptive



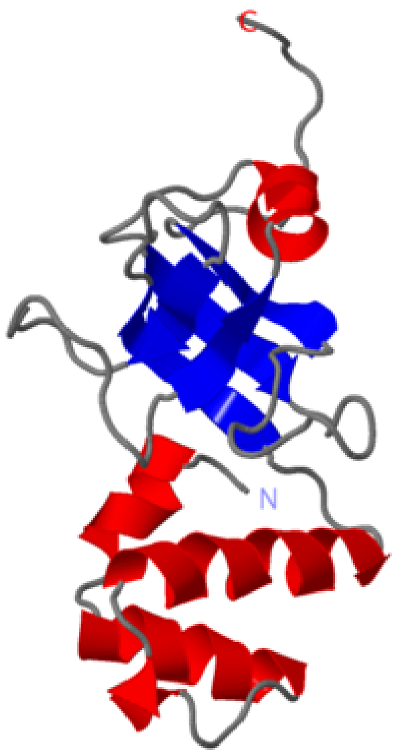
- low memory usage, good parallel efficiency

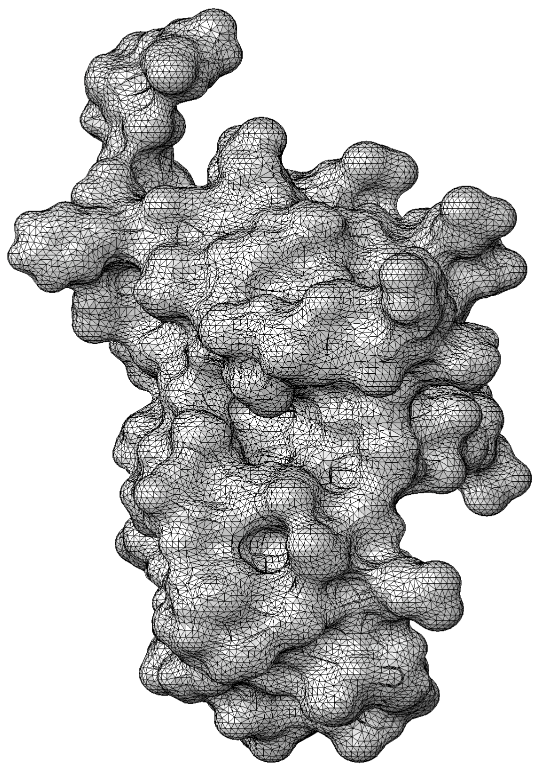
Li-Johnston-K (2009) JCP 228

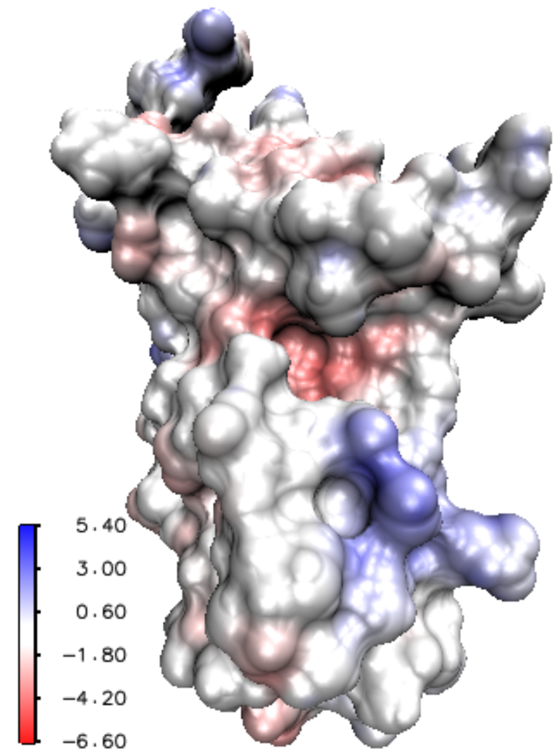
example : protein 1A63

RNA binding domain of E. coli rho factor

2069 atoms , triangulation by MSMS with $N = 132196$







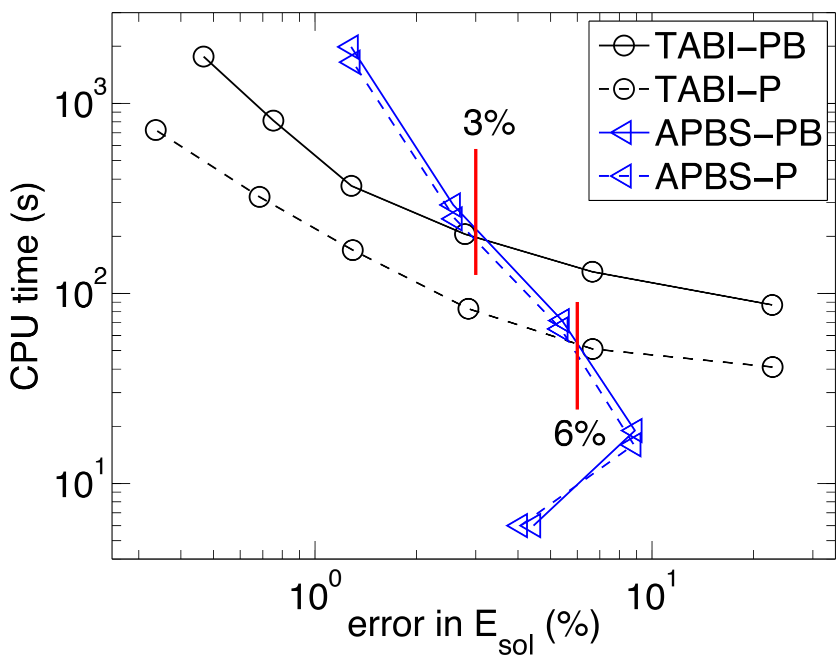
example : protein 1A63

compare 2 codes

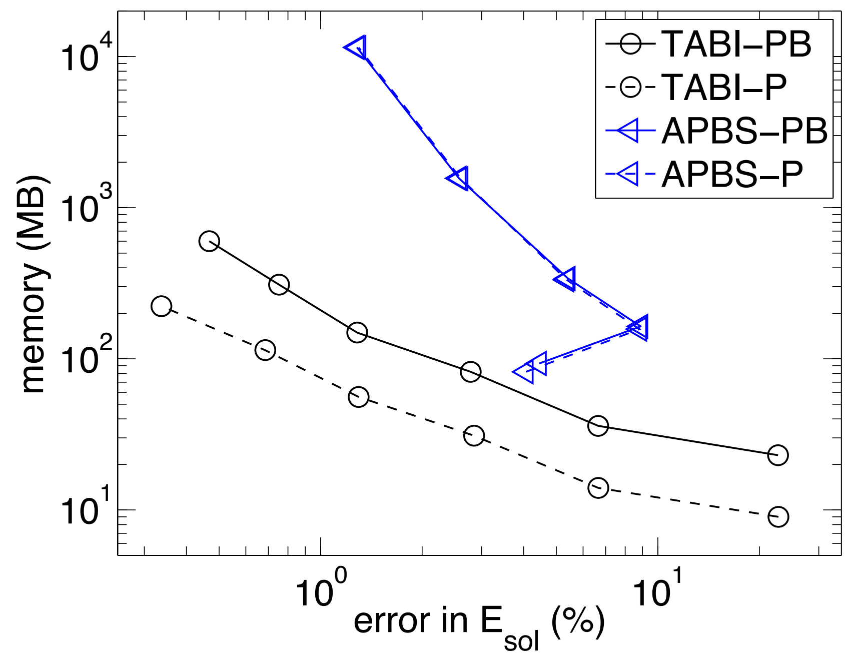
TABI : $N = 20\mathrm{K} \to 500\mathrm{K}$

APBS : grid $= 65 \times 33^2 \to 513 \times 321^2$, $h_{\max} = 1.63\,\text{Å} \to 0.13\,\text{Å}$

CPU time / error



memory / error



Geng-K (2013) JCP 247

example : protein 1A63

TABI parallel performance , $N = 132196$, error in $E_{\mathrm{sol}} \approx 1.3\%$

Poisson-Boltzmann

# processors	run time (s)	speedup	parallel efficiency (%)
1	799.3	1.00	100.0
2	410.0	1.95	97.5
4	223.8	3.57	89.3
8	123.7	6.46	80.9

Poisson

# processors	run time (s)	speedup	parallel efficiency (%)
1	324.4	1.00	100.0
2	166.7	1.95	97.3
4	92.6	3.50	87.6
8	51.0	6.36	79.5

PROTEIN SCIENCE 2018 VOL 27:112-128

THE PROTEIN SOCIETY

TOOLS FOR PROTEIN SCIENCE

Improvements to the APBS biomolecular solvation software suite

Elizabeth Jurrus,[1] Dave Engel,[1] Keith Star,[1] Kyle Monson,[1] Juan Brandi,[1] Lisa E. Felberg,[2] David H. Brookes,[2] Leighton Wilson,[3] Jiahui Chen,[4] Karina Liles,[1] Minju Chun,[1] Peter Li,[1] David W. Gohara,[5] Todd Dolinsky,[6] Robert Konecny,[7] David R. Koes,[8] Jens Erik Nielsen,[9] Teresa Head-Gordon,[2] Weihua Geng,[4] Robert Krasny,[3] Guo-Wei Wei,[10] Michael J. Holst,[7] J. Andrew McCammon,[7] and Nathan A. Baker[1,11]*

[1]Pacific Northwest National Laboratory, Richland, Washington
[2]University of California, Berkeley, California
[3]University of Michigan, Ann Arbor, Michigan
[4]Southern Methodist University, Dallas, Texas
[5]St. Louis University, St. Louis, Missouri
[6]FoodLogiQ, Durham, North Carolina
[7]University of California San Diego, San Diego, California
[8]University of Pittsburgh, Pittsburgh, Pennsylvania
[9]Protein Engineering, Novozymes A/S, Copenhagen, Denmark
[10]Michigan State University, East Lansing, Michigan
[11]Brown University, Providence, Rhode Island

comparison of software for molecular surface

MSMS

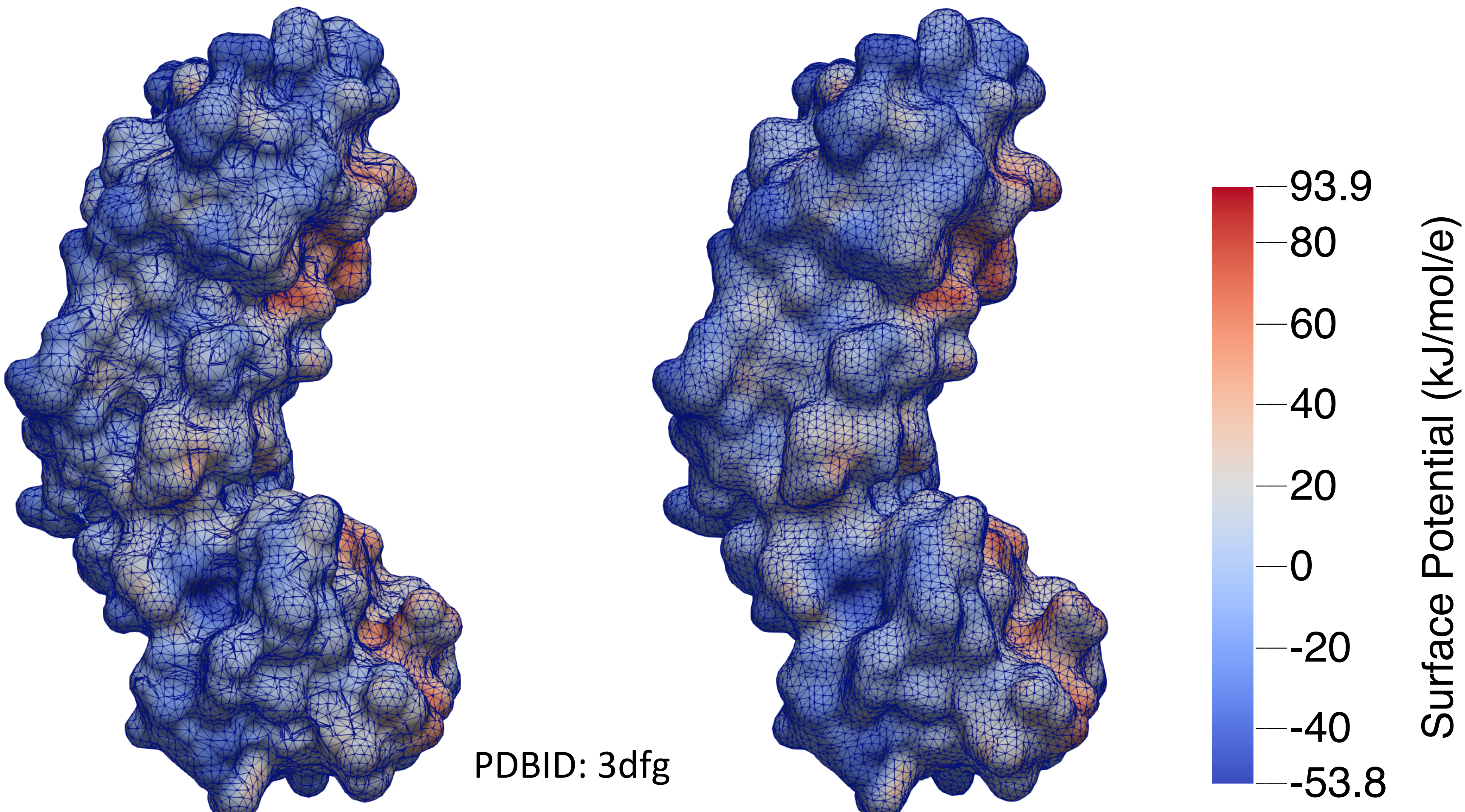
N = 29867, iter = 62

E_sol = -10443 kJ/mol

NanoShaper

N = 29224, iter = 9

E_sol = -10675 kJ/mol



outline

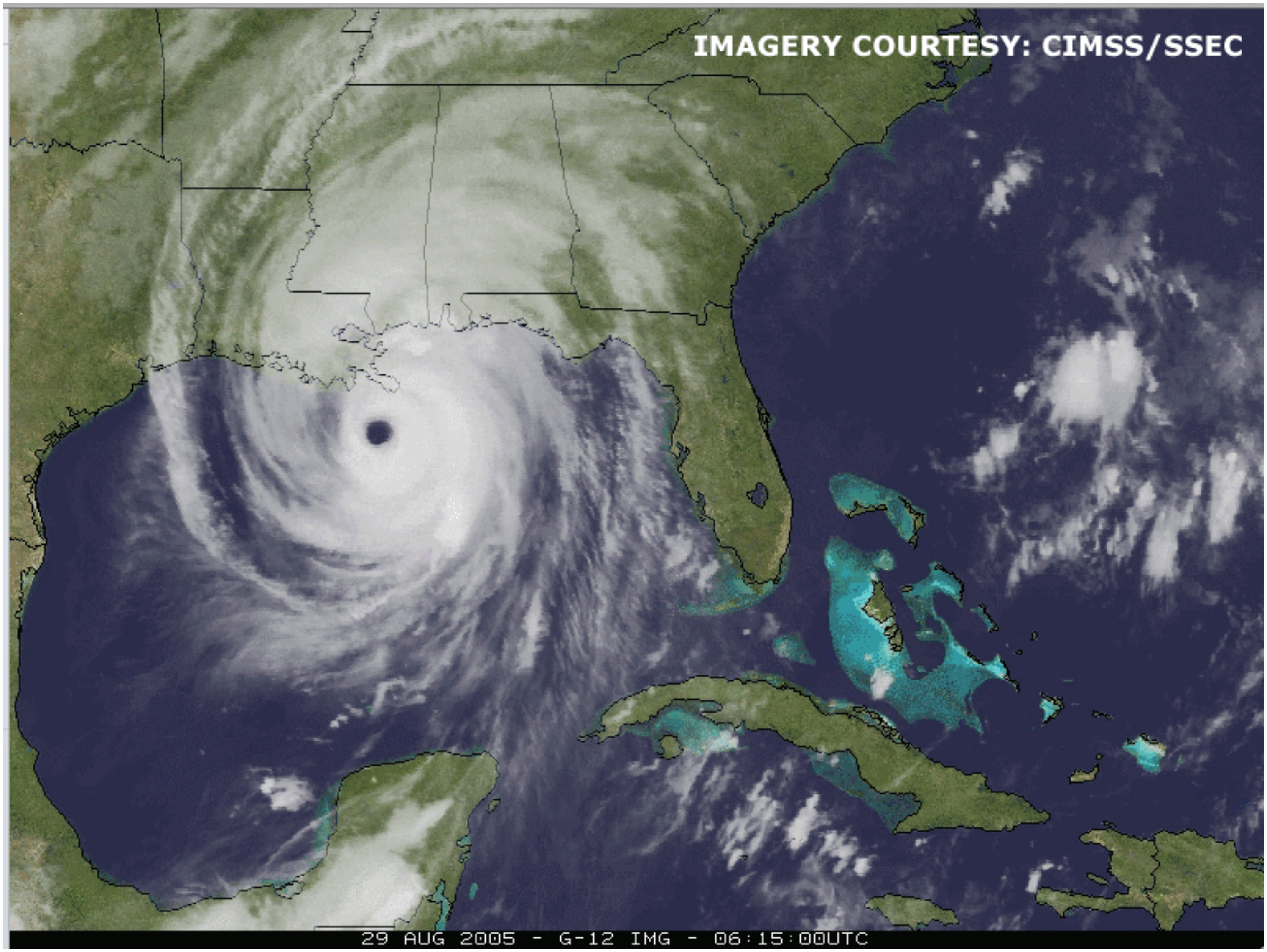
1. protein-solvent electrostatics
2. incompressible fluid flow

   - tracer transport on a sphere

   - vortex dynamics on a sphere

   - vortex dynamics in 2D

motivation : atmosphere



tracer transport on a sphere

given : $q_0(x)$, $u(x,t)$, $x \in S$

problem : determine $q(x,t)$ for $t > 0$

Eulerian form

$$\frac{\partial q}{\partial t}(x,t) + u \cdot \nabla q(x,t) = 0$$

Lagrangian form

flow map : $a \to x(a,t)$, $a \in S$

$$\frac{\partial}{\partial t} x(a,t) = u(x(a,t),t) \ , \ x(a,0) = a$$

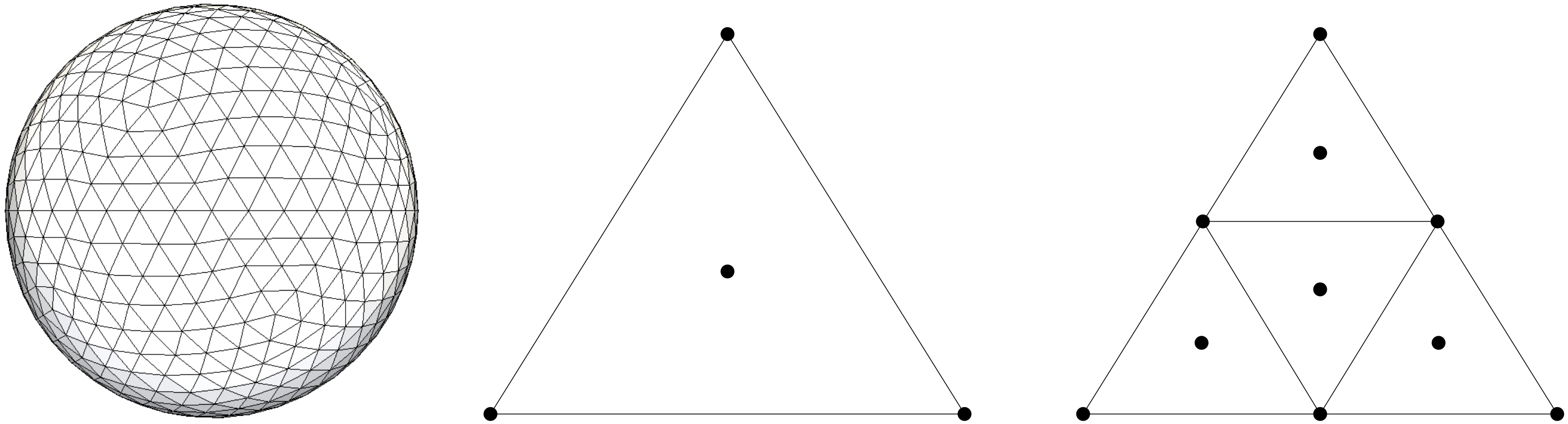
$$q(x(a,t),t) = q_0(a)$$

Lagrangian particle method

particles : $x_j(t) \approx x(a_j, t)$, $x_j(0) = a_j$, $j = 1 : M$

panels : $P_k(t) = x(P_k(0), t)$, $S = \cup_{k=1}^{N} P_k(t)$, $k = 1 : N$

icosahedral triangulation



Bosler-Kent-K-Jablonowski (2017) JCP 340

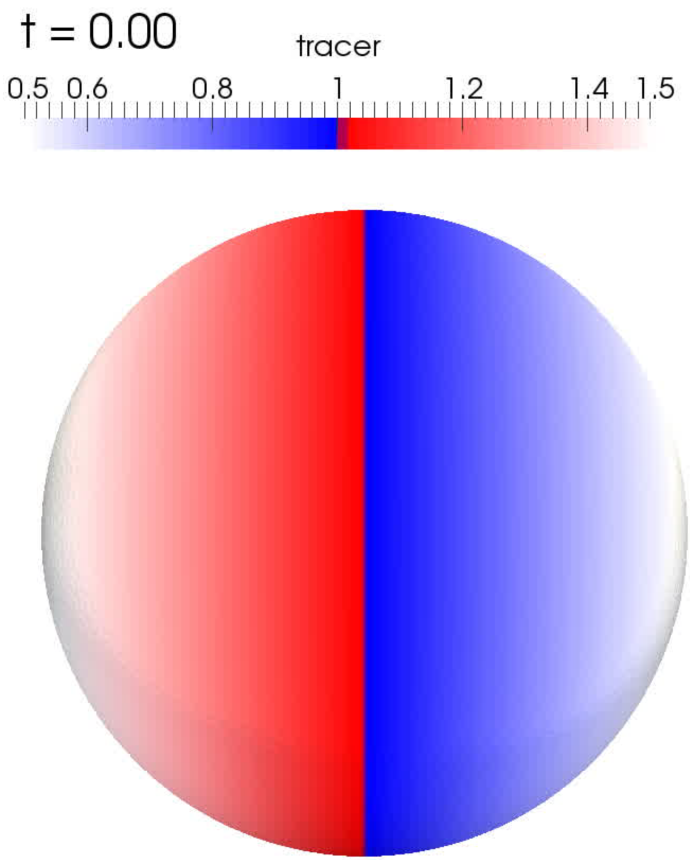
particle advection

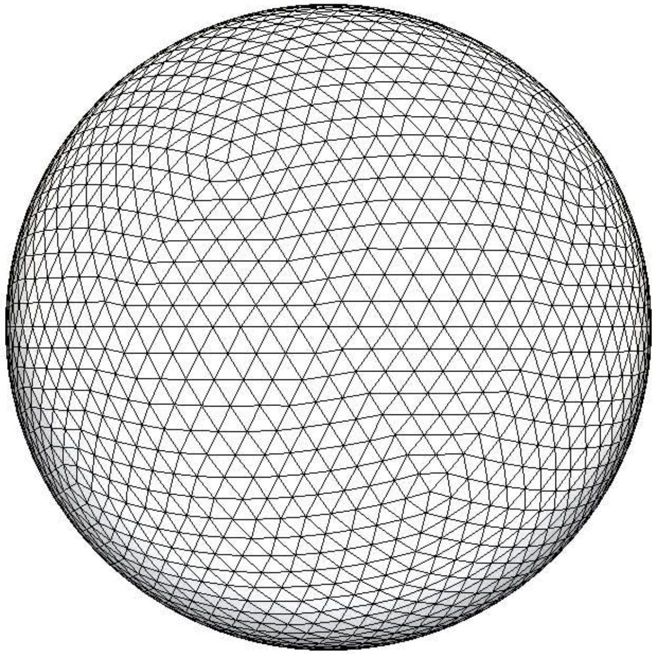
$$\frac{d}{dt}x_j(t) = u(x_j, t)\ ,\ x_j(0) = a_j$$

$$q_j = q_0(a_j)$$

problem : particles become disordered for $t > 0$

test case 1 : moving vortices flow





movie

particle advection

$$\frac{d}{dt}x_j(t) = u(x_j, t)\ ,\ x_j(0) = a_j$$

$$q_j = q_0(a_j)$$

problem : particles become disordered for $t > 0$

solution : remeshing : $\{x_j^{old}\} \to \{x_j^{new}\}$, $\{q_j^{new}\} =$?

1. direct remeshing

$$\{q_j^{new}\} = I(\{x_j^{new}\}; \{x_j^{old}\}, \{q_j^{old}\})$$

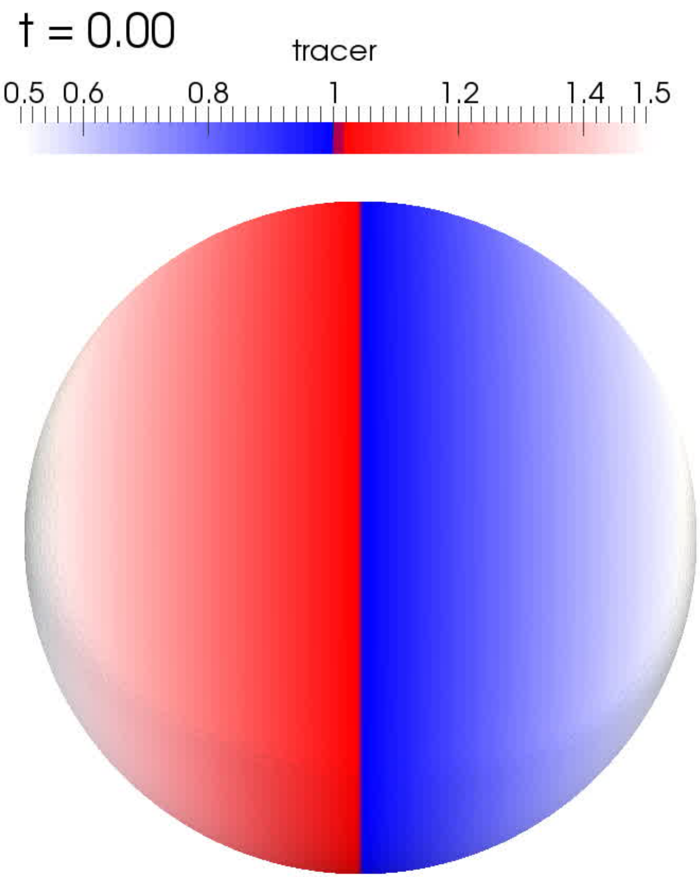
2. indirect remeshing

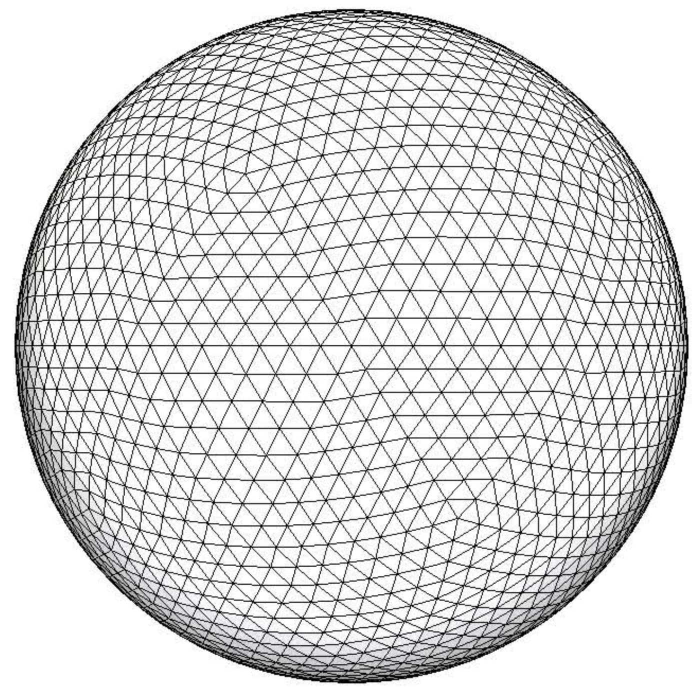
$$\{a_j^{new}\} = I(\{x_j^{new}\}; \{x_j^{old}\}, \{a_j^{old}\})$$

$$\{q_j^{new}\} = q_0(\{a_j^{new}\})$$

I : STRIPACK/SSRFPACK (Renka 1997)

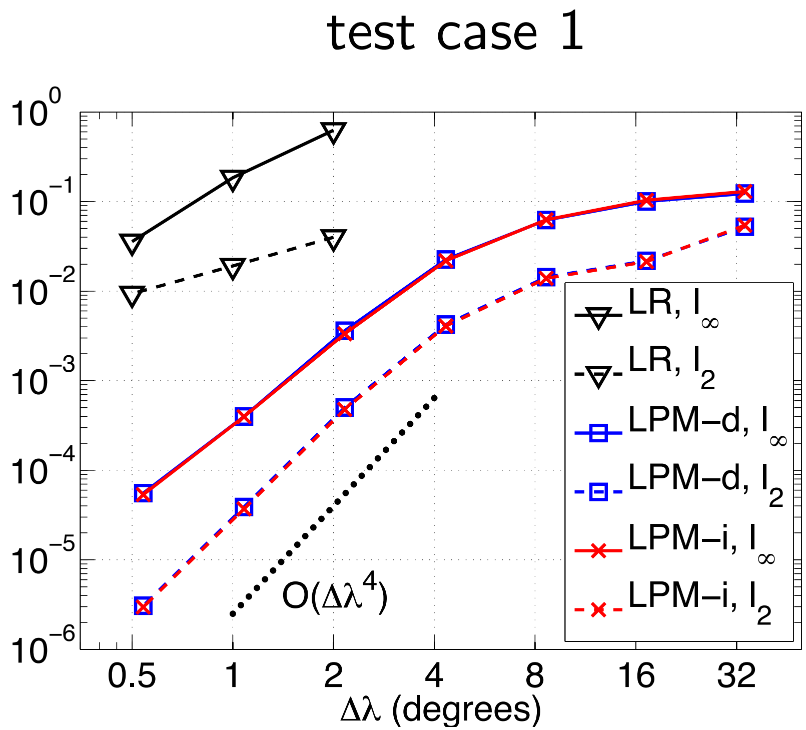
test case 1 : moving vortices flow

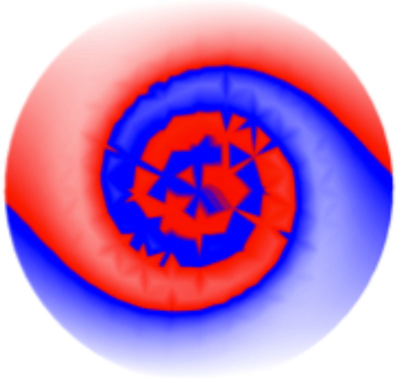




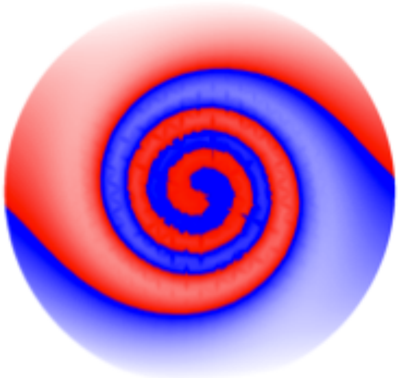
movie

tracer error

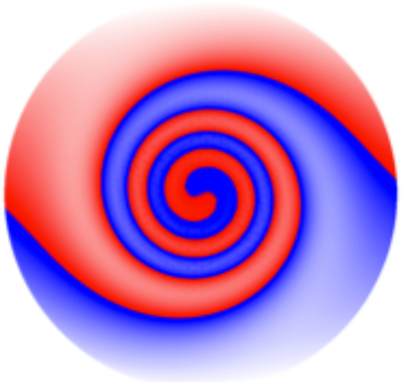




$\Delta\lambda = 8.64°$



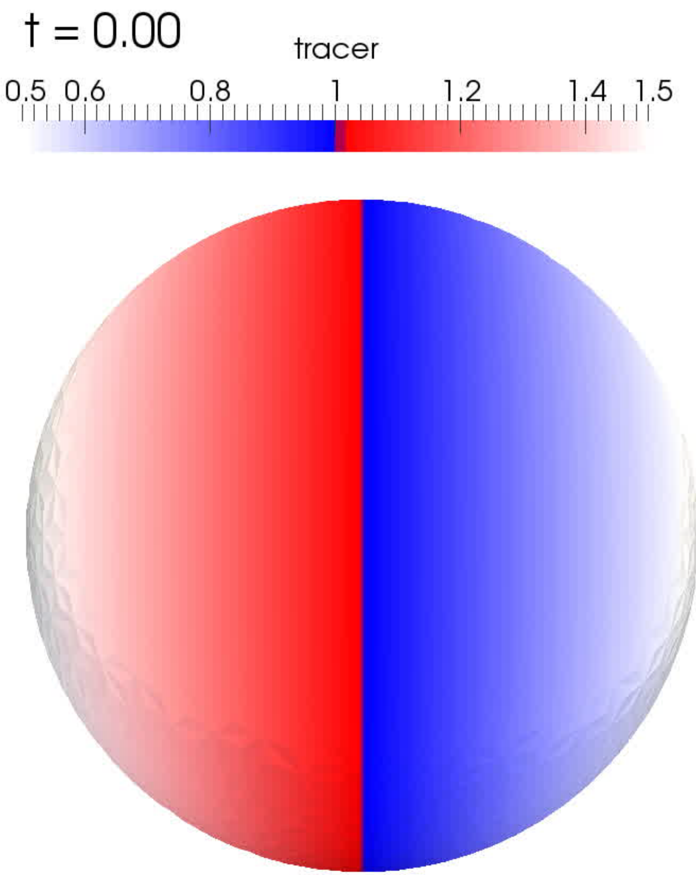
$\Delta\lambda = 4.33°$



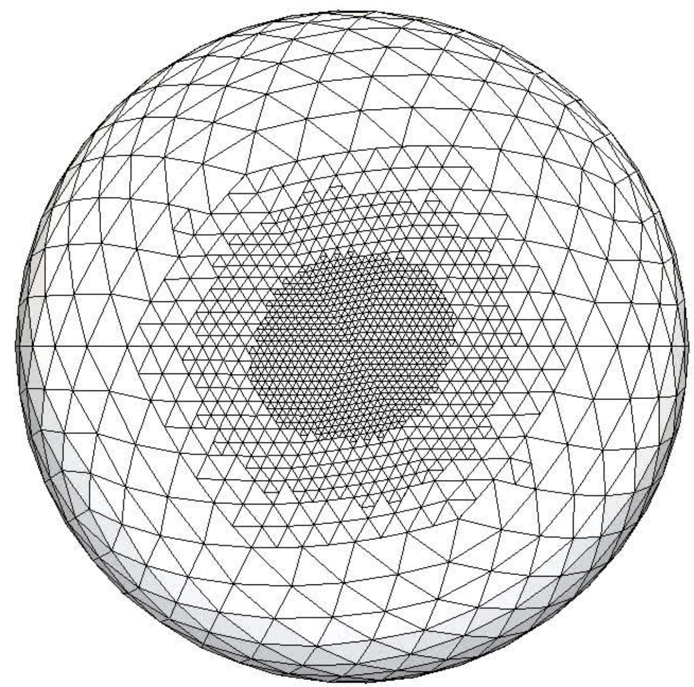
$\Delta\lambda = 2.16°$

test case 1 : moving vortices flow

- adaptive panel refinement

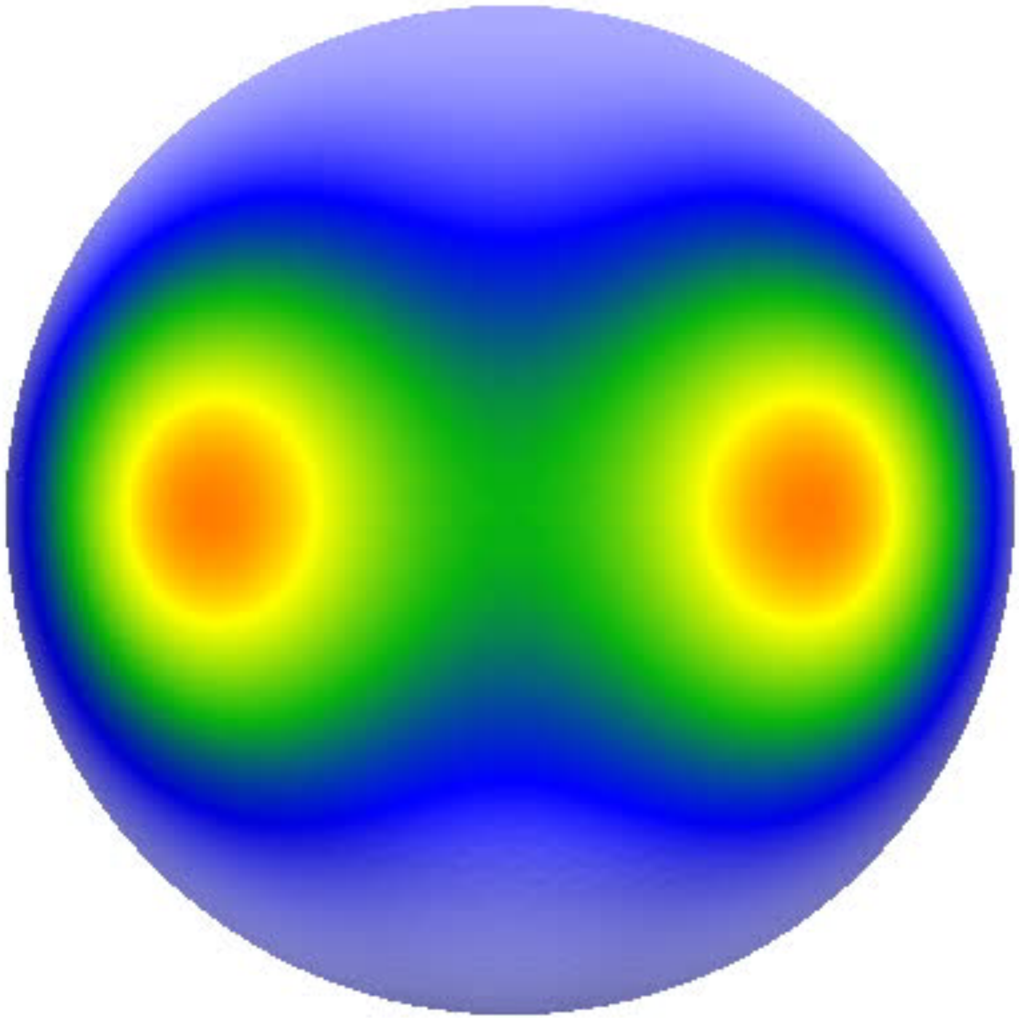


$$\zeta(\mathbf{x}_k)A_k < \epsilon\,\Gamma_{\max}$$



movie

test case 2 : reversing-deformational flow
- Gaussian hills tracer

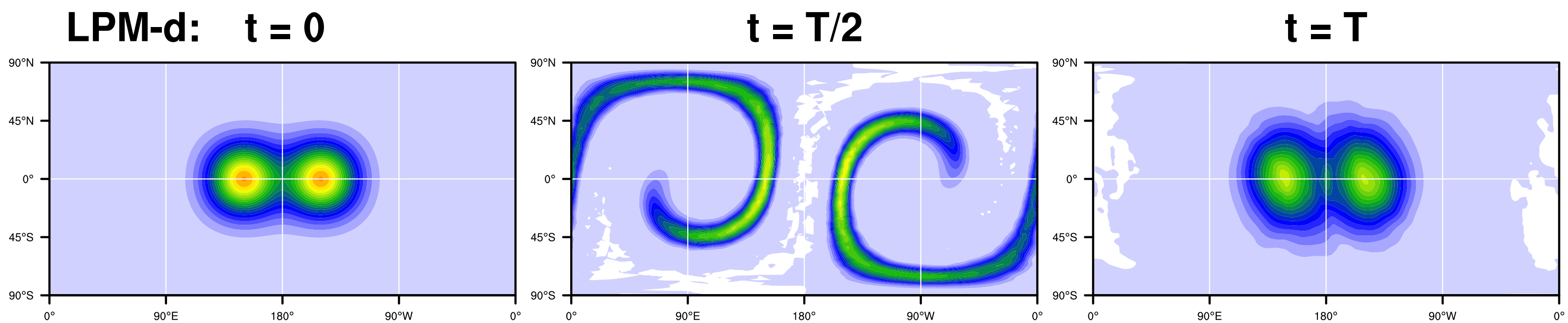


movie

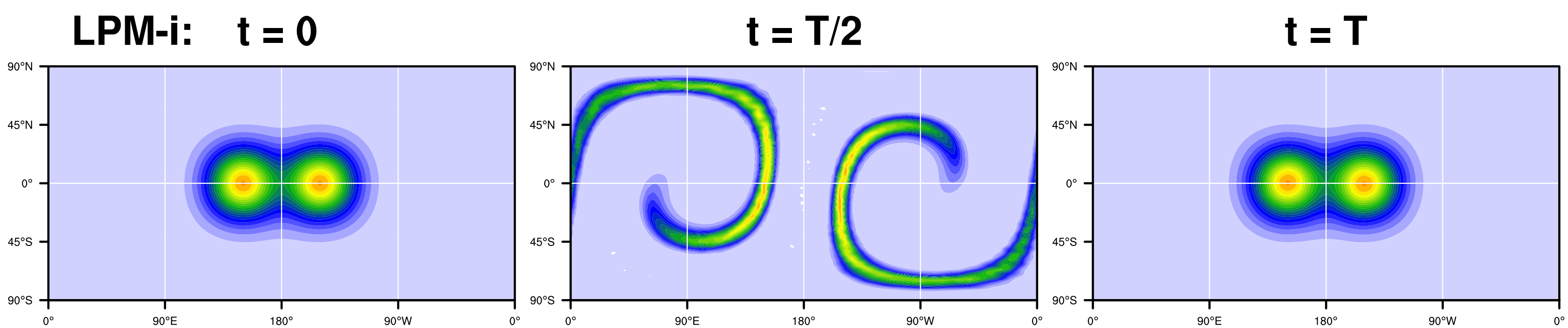
test case 2 : reversing-deformational flow

Gaussian-hills tracer , grid spacing $\Delta\lambda = 8.64^\circ$

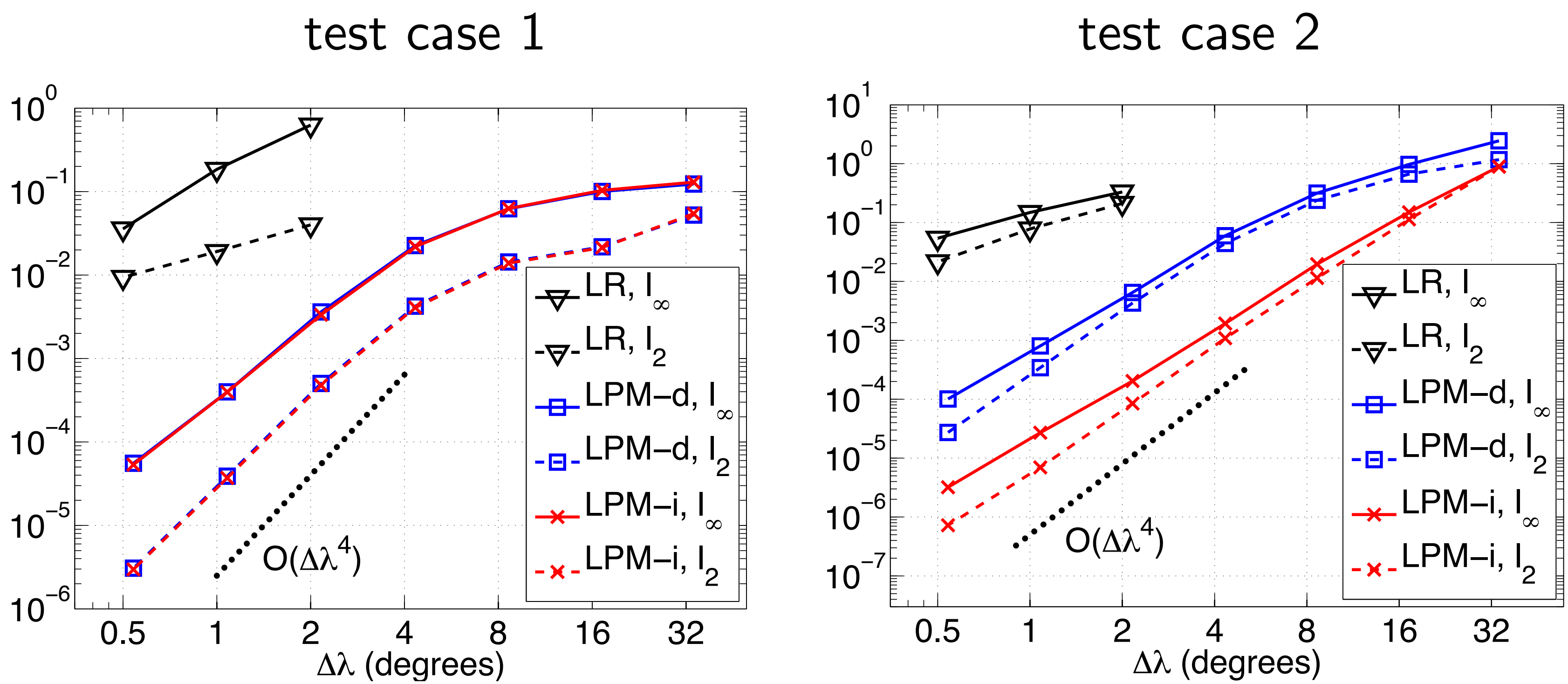
direct remeshing



indirect remeshing



tracer error



vortex dynamics on a sphere (partial list!)

point vortices

Kimura-Okamoto (1987), Newton-Sakajo (2007) . . .

vortex patches

Dritschel (1989, 2004), Dritschel-Polvani (1992)

Crowdy-Cloke (2003), Surana-Crowdy (2008) . . .

vortex sheets

Sakajo (2009) . . .

present work : smooth vorticity

barotropic vorticity equation : Eulerian form

velocity : $\mathbf{u} = \nabla\psi \times \mathbf{x}$

vorticity : $\zeta = \nabla \times \mathbf{u}$

Poisson equation : $\nabla^2\psi = -\zeta$

vortex dynamics : $\dfrac{\partial\zeta}{\partial t} + \mathbf{u} \cdot \nabla(\zeta + f) = 0$

Coriolis parameter : $f = 2\Omega\sin\theta$

Bosler-Wang-Jablonowski-K (2014) FDR 46

barotropic vorticity equation : Lagrangian form

flow map : $\mathbf{x} = \mathbf{x}(\vec{\alpha}, t)$

vorticity : $\zeta = \zeta(\mathbf{x}, t)$

stream function : $\psi(\mathbf{x}) = -\frac{1}{4\pi}\int_S \log(1 - \mathbf{x}\cdot\tilde{\mathbf{x}})\zeta(\tilde{\mathbf{x}})dA(\tilde{\mathbf{x}})$

Biot-Savart law : $\frac{d\mathbf{x}}{dt} = -\frac{1}{4\pi}\int_S \frac{\mathbf{x}\times\tilde{\mathbf{x}}}{1 - \mathbf{x}\cdot\tilde{\mathbf{x}}}\zeta(\tilde{\mathbf{x}}, t)dA(\tilde{\mathbf{x}})$

vortex dynamics : $\frac{d\zeta}{dt} = -2\Omega\frac{dz}{dt}$

discretization

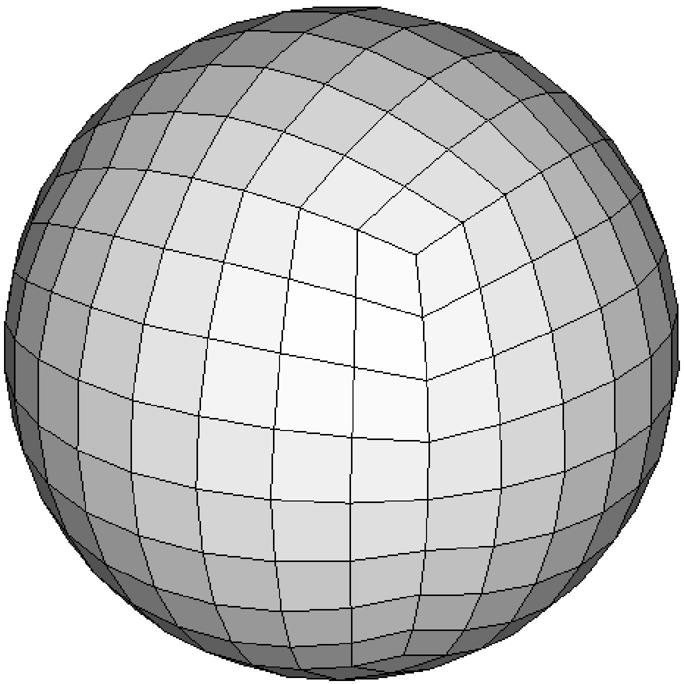
$$\mathcal{S} = \cup_{k=1}^{N} A_k$$

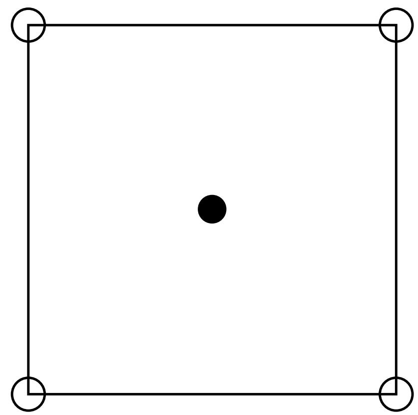
panels : cubed sphere

particles : $\mathbf{x}_j(t) = \mathbf{x}(\vec{\alpha}_j, t)$

$$\frac{d\mathbf{x}_j}{dt} = -\frac{1}{4\pi} \sum_{\substack{k=1 \\ k \neq j}} \frac{\mathbf{x}_j \times \mathbf{x}_k}{1 - \mathbf{x}_j \cdot \mathbf{x}_k} \zeta_k A_k$$

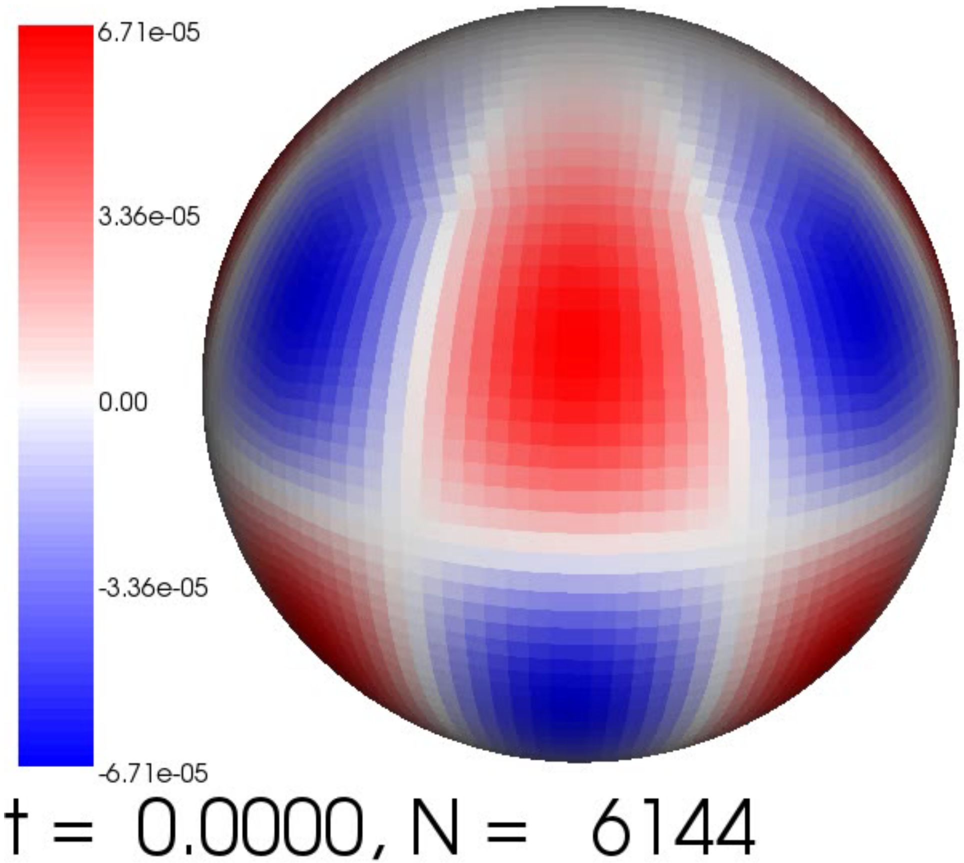
vorticity : $\displaystyle \frac{d\zeta_j}{dt} = -2\Omega \frac{dz_j}{dt}$



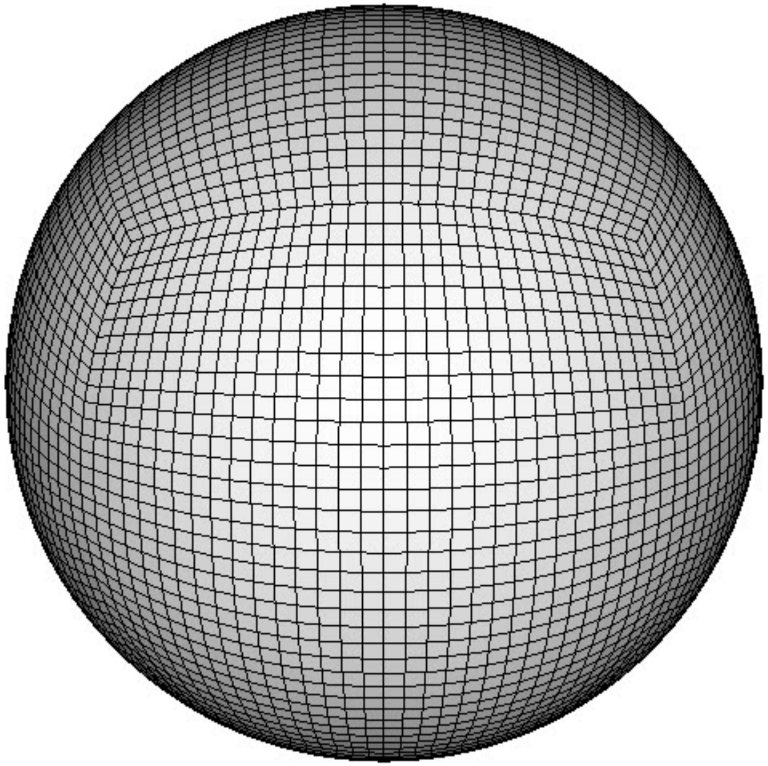


Rossby-Haurwitz wave – no remeshing

vorticity



panels

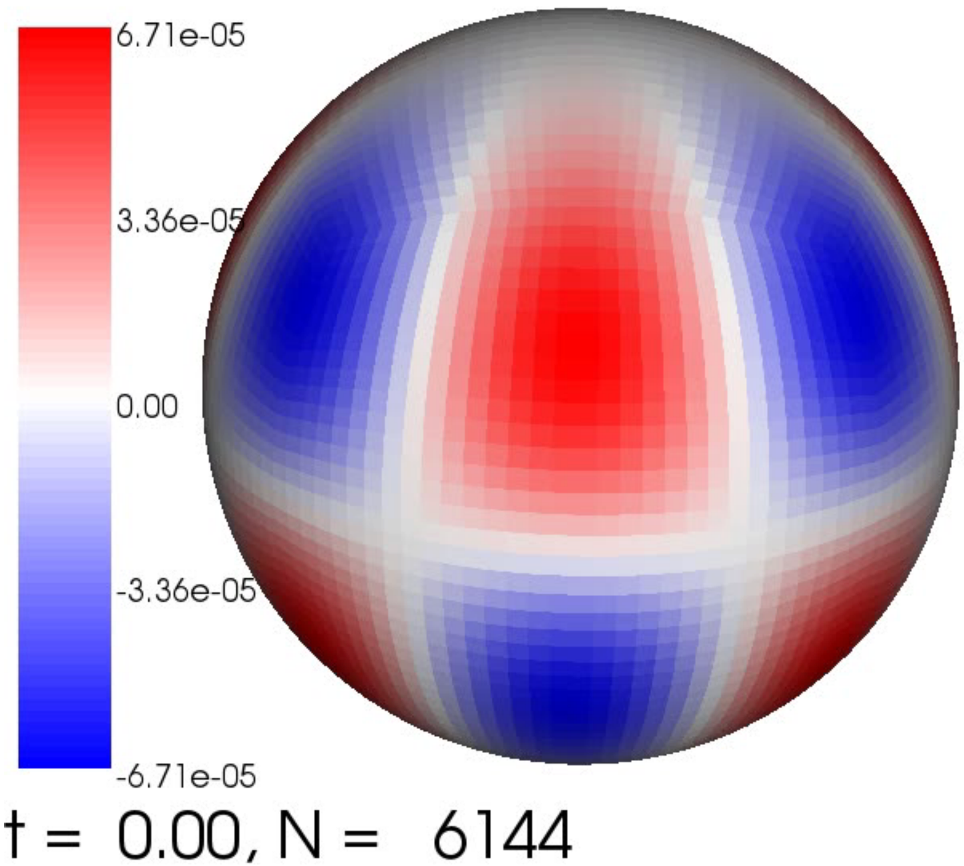


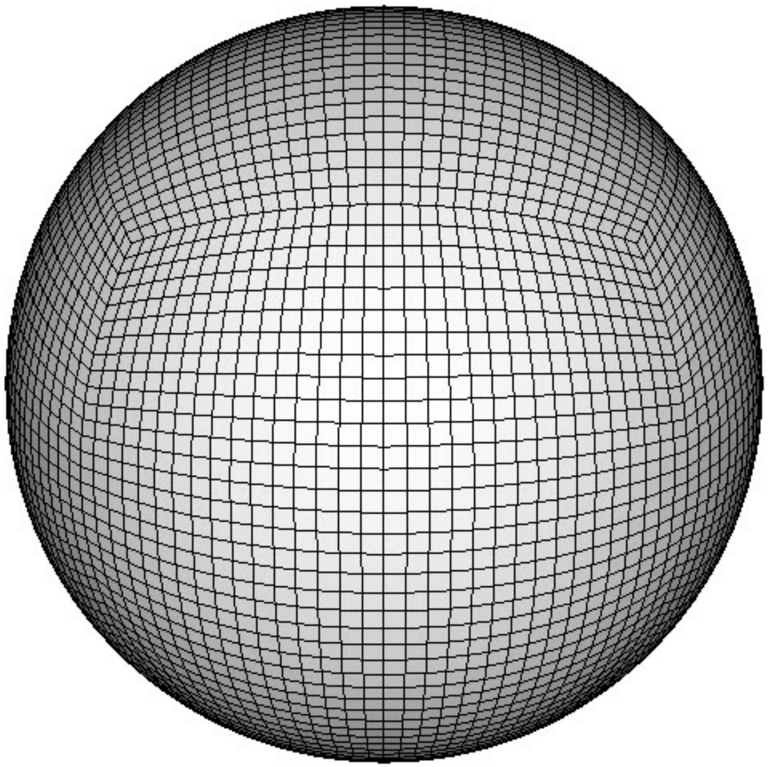
movie

Rossby-Haurwitz wave – remeshing

vorticity

panels



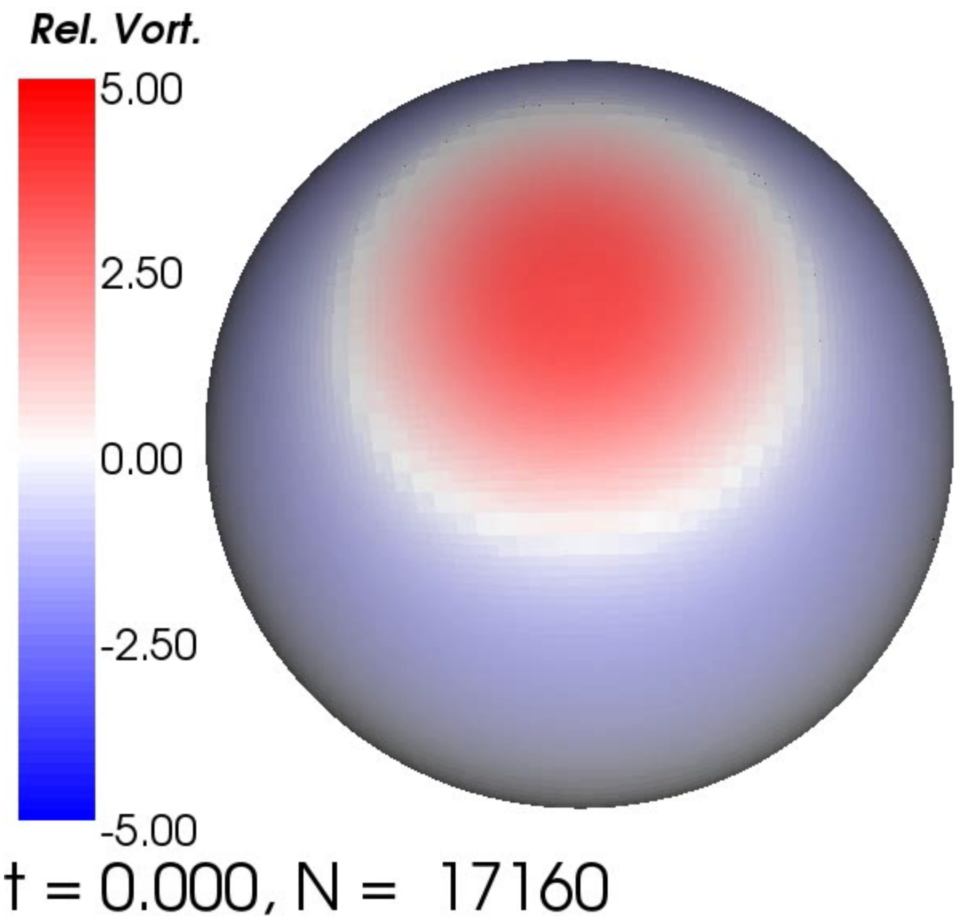


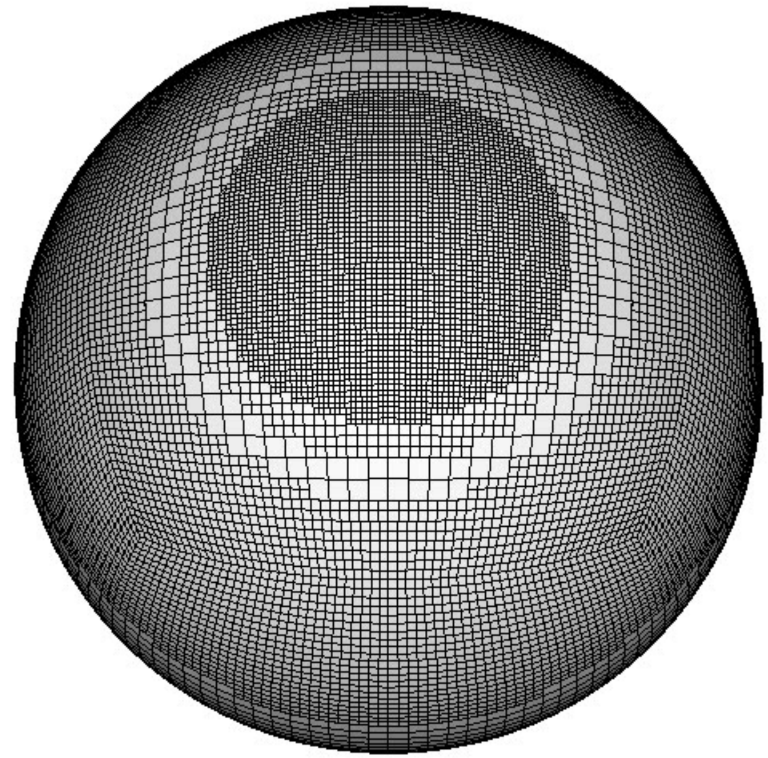
movie

disruption of the polar vortex

motivated by Juckes & McIntyre (1987)

remesh + refine





movie

fluid flow in 2D : incompressible, inviscid

Eulerian form

$$\frac{\partial \omega}{\partial t} + \mathbf{u} \cdot \nabla \omega = 0 \ , \ \nabla \cdot \mathbf{u} = 0$$

$$\mathbf{u} = \nabla^{\perp} \psi \ , \ \nabla^2 \psi = -\omega$$

Lagrangian form

$$\mathbf{x}(\mathbf{a}, t) \ , \ \frac{d\mathbf{x}}{dt} = \mathbf{u}(\mathbf{x}, t) \ , \ \mathbf{x}(\mathbf{a}, 0) = \mathbf{a}$$

$$\mathbf{u}(\mathbf{x}, t) = \int K_\delta(\mathbf{x}, \mathbf{y}) \omega(\mathbf{y}, t) d\mathbf{y} \ , \ K_\delta(\mathbf{x}, \mathbf{y}) = -\nabla^{\perp} \frac{\ln(|\mathbf{x} - \mathbf{y}|^2 + \delta^2)}{4\pi}$$

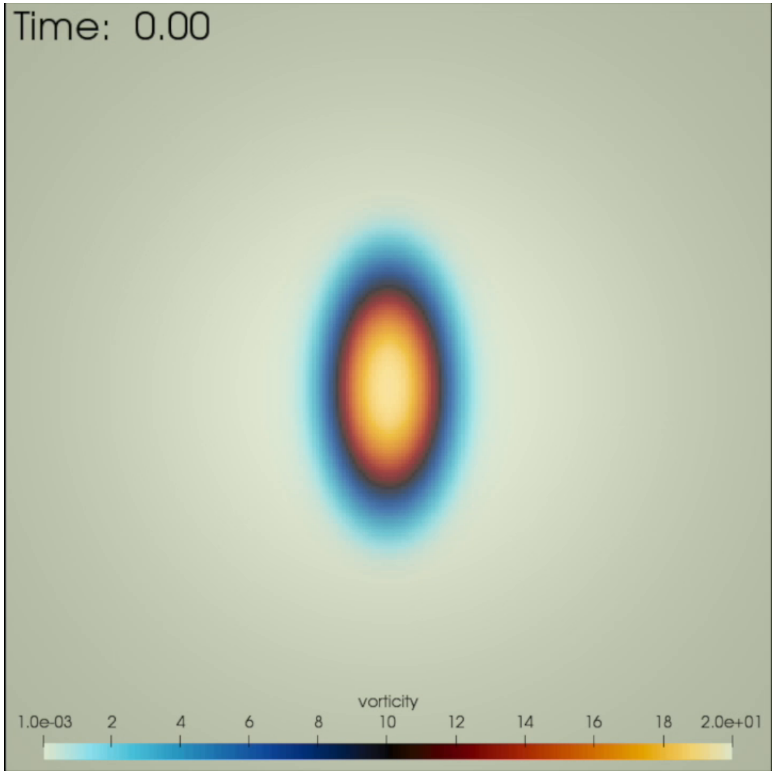
$$\omega(\mathbf{x}, t) = \omega_0(\mathbf{a})$$

- particle/panel method + remeshing, AMR, treecode

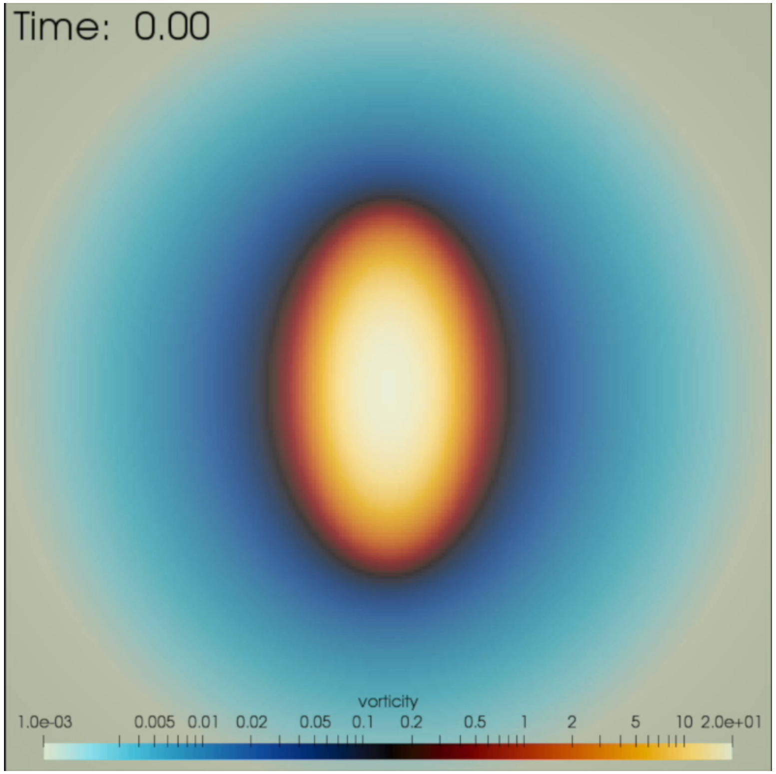
elliptic vortex

Koumoutsakos (1997)

linear scale

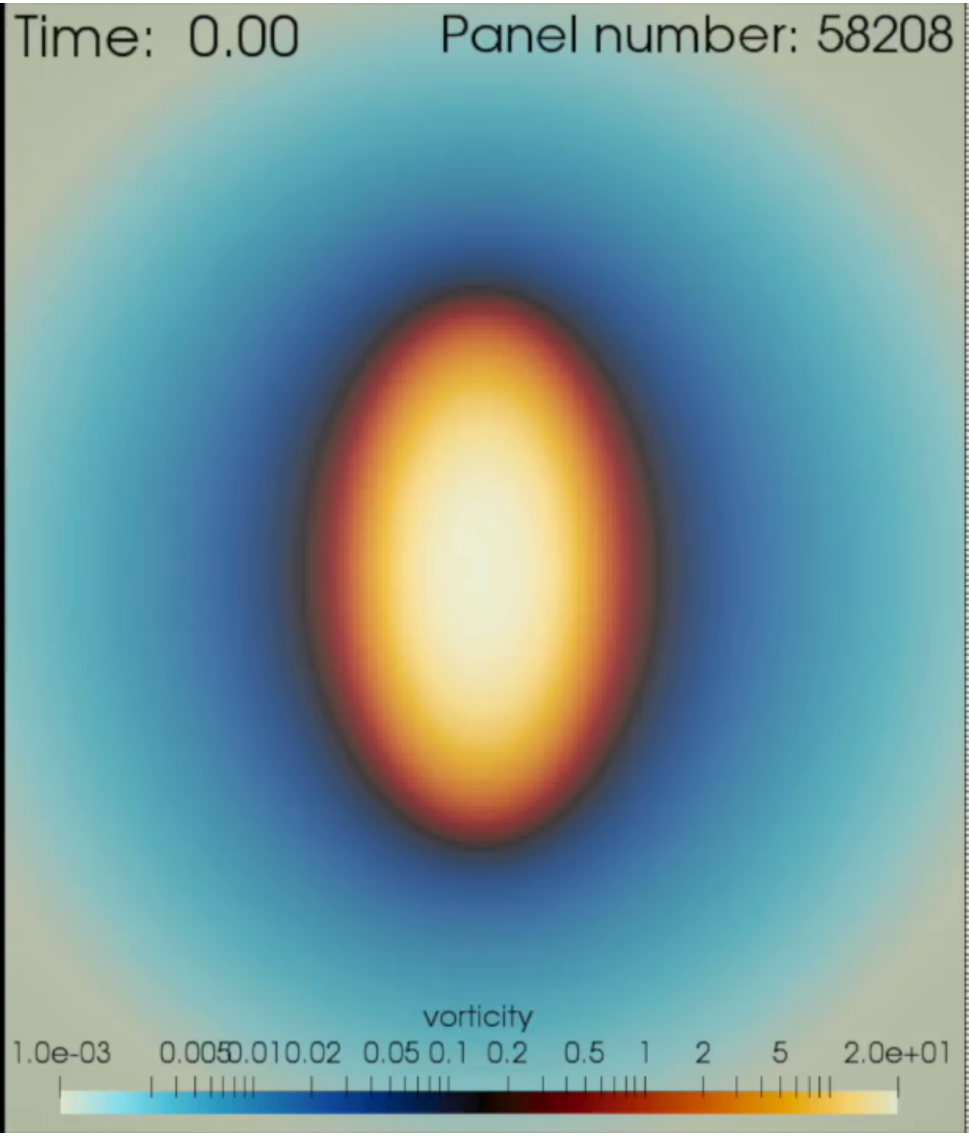


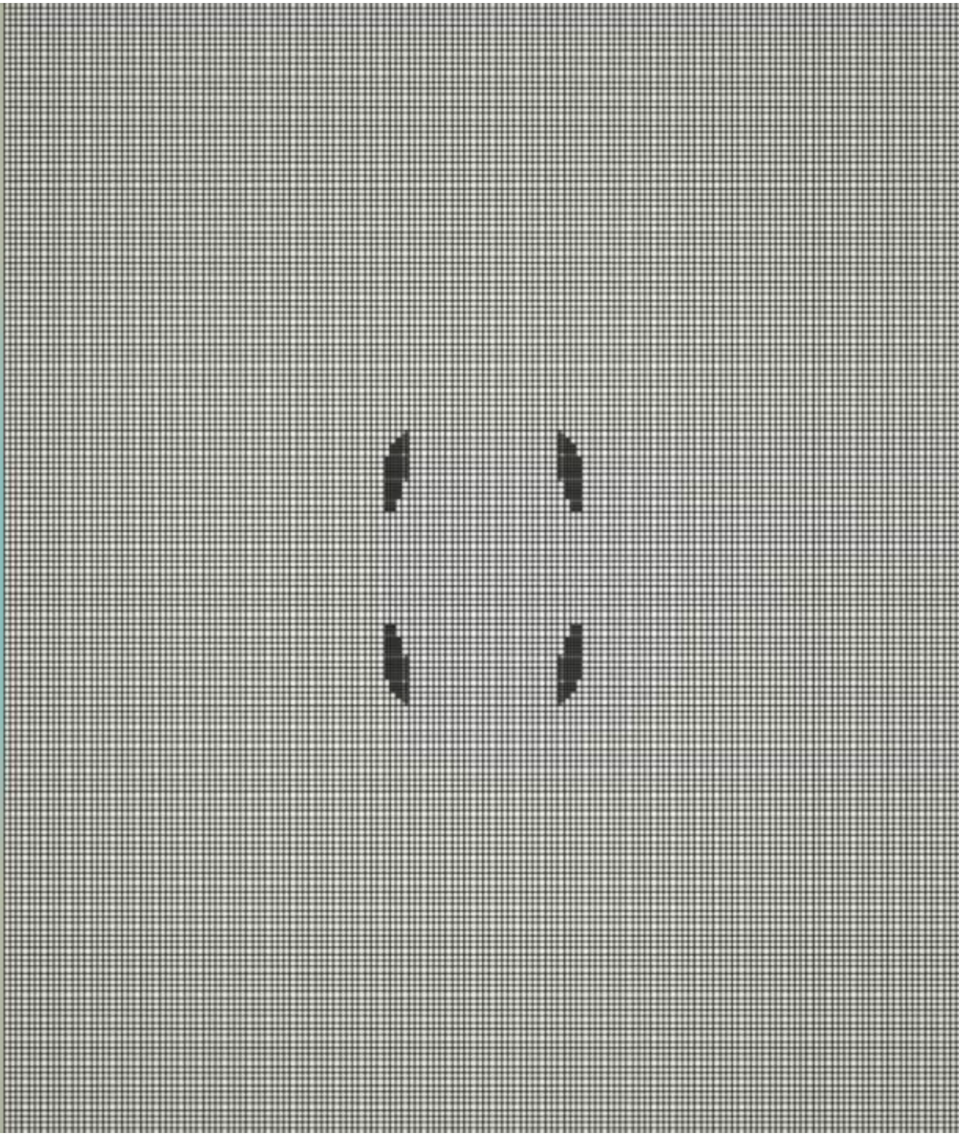
logarithmic scale



elliptic vortex

adaptive mesh



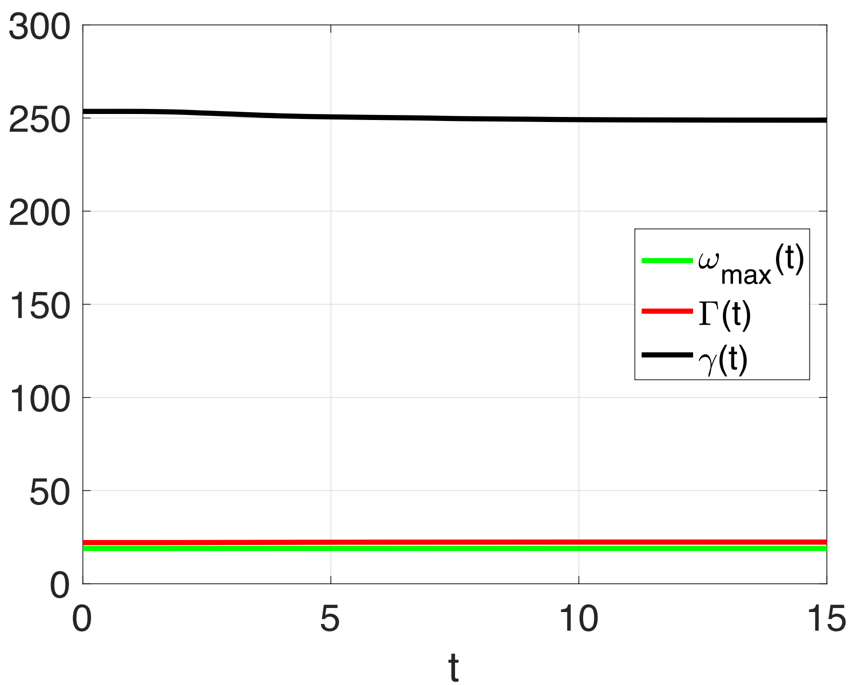


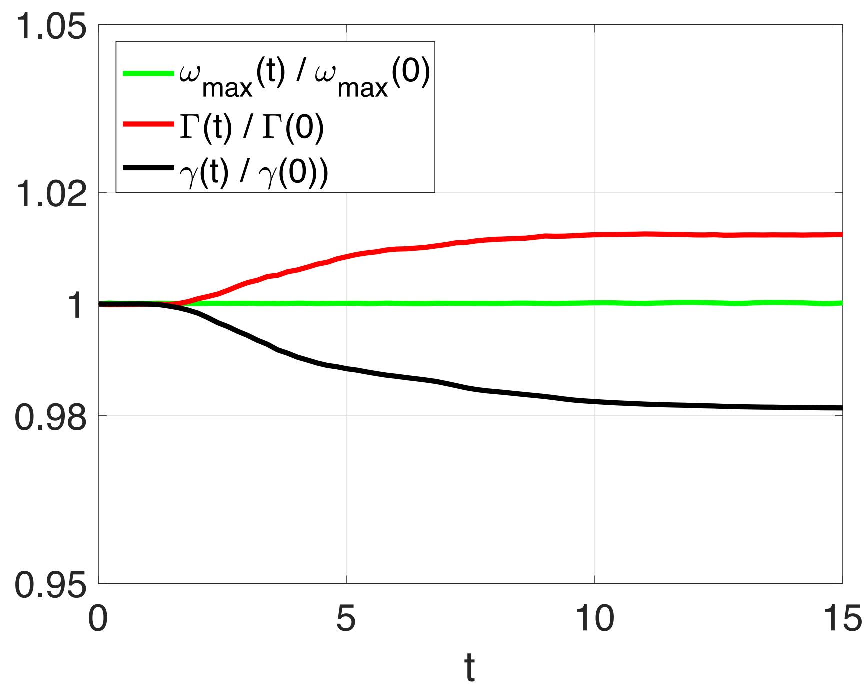
elliptic vortex 1 : conserved quantities

maximum vorticity : $\omega_{\max}(t) = \max_{\mathbf{x}} \omega(\mathbf{x}, t)$

circulation : $\Gamma(t) = \int \omega(\mathbf{x}, t) dA$

enstrophy : $\gamma(t) = \int \omega^2(\mathbf{x}, t) dA$





thanks to my collaborators -

thanks to my collaborators -

Weihua Geng

Peter Bosler

Ling Xu

current/future projects

protein-solvent electrostatics

- binding energy of protein-ligand complex
- protein pKa

incompressible fluid flow

- viscosity, 3D, solid boundaries
- shallow water equations