Mesh-Free Applications for Static and Dynamically Changing Node Configurations

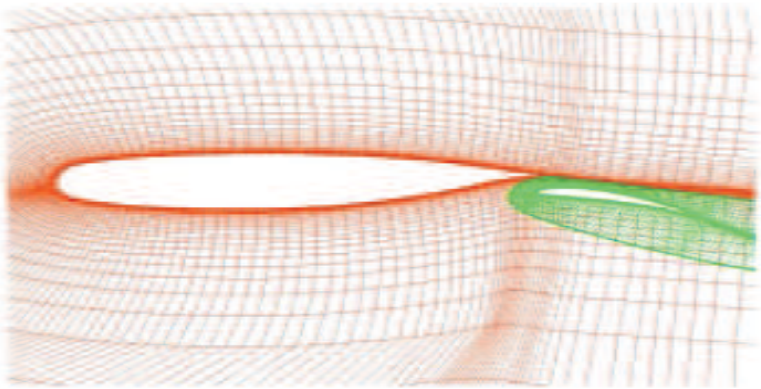
Natasha Flyer
Computational Information Systems Lab
National Center for Atmospheric Research
Boulder, CO

Meshes vs. Mesh-free discretizations

Structured meshes:

FD, DG, FV, Spectral Elements

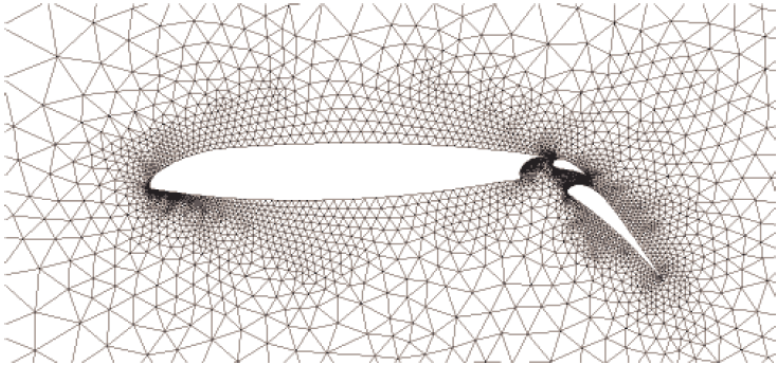
Requires domain decomposition / curvilinear mappings



Unstructured meshes:

FEM, DG, FV, Spectral Elements

Improved geometric flexibility; requires triangles, tetrahedral, etc.

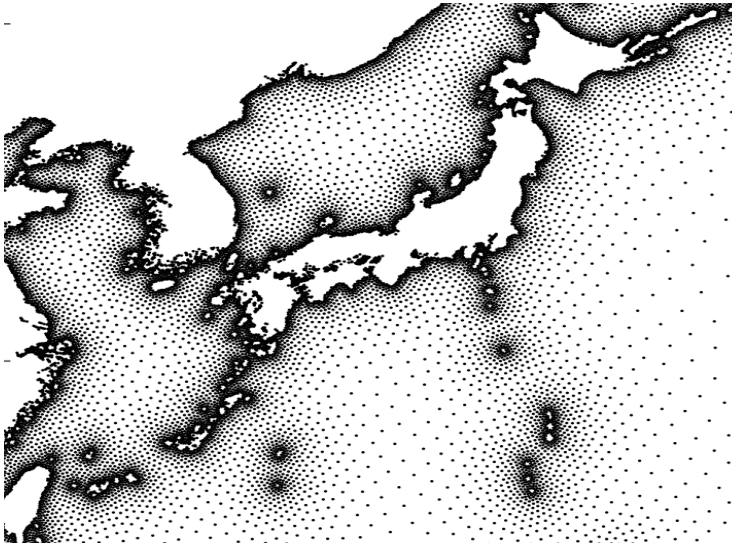


Mesh-free:

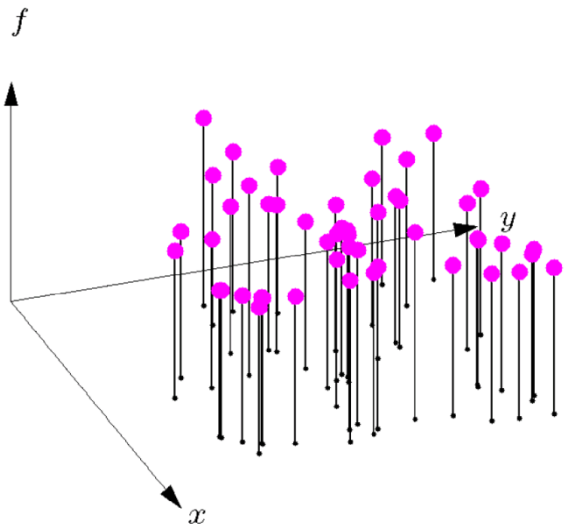
RBF-FD

(Radial basis Func.-generated Finite Differences)

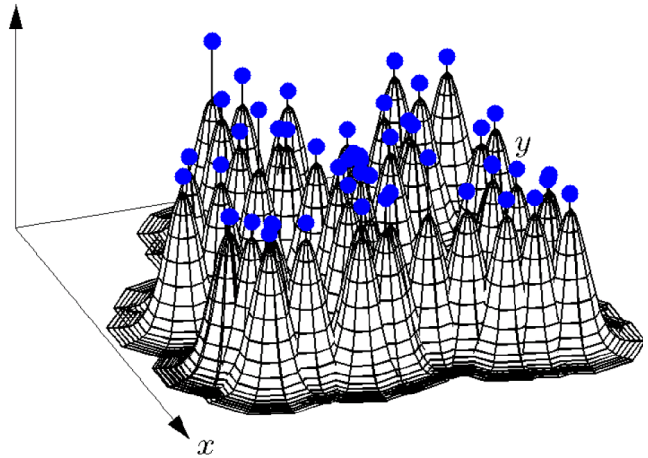
Total geometric flexibility;
needs node locations, but no connectivites,
e.g. no triangles or mappings



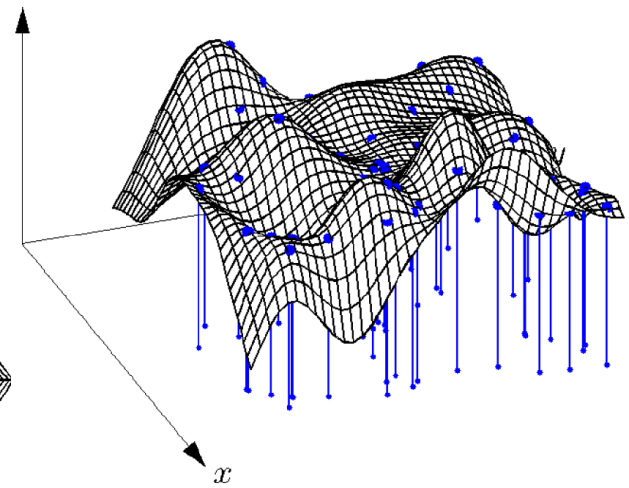
General RBF-FD Concept



Scattered data in 2D

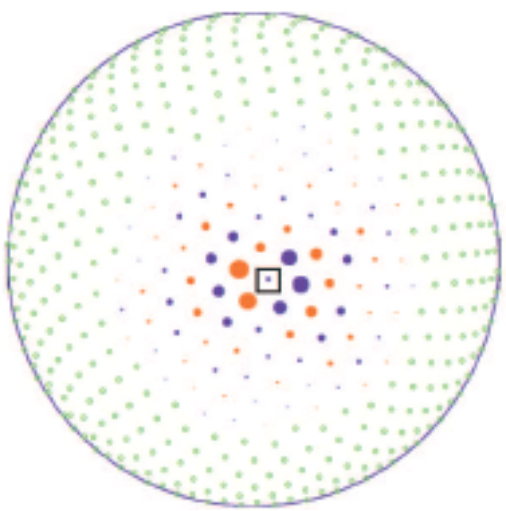


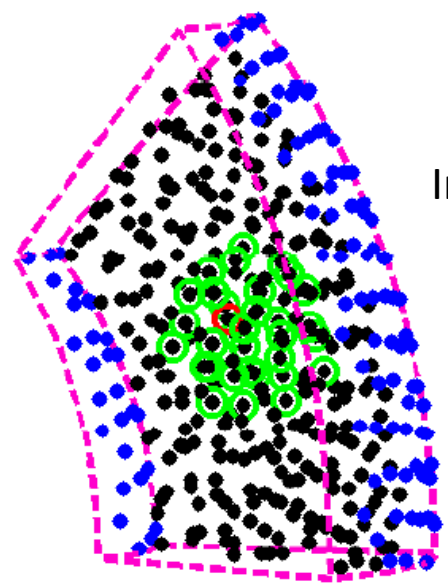
Collocate RBF, e.g. 2D Gaussians



Find linear combination of RBF that fits all the data

On a sphere

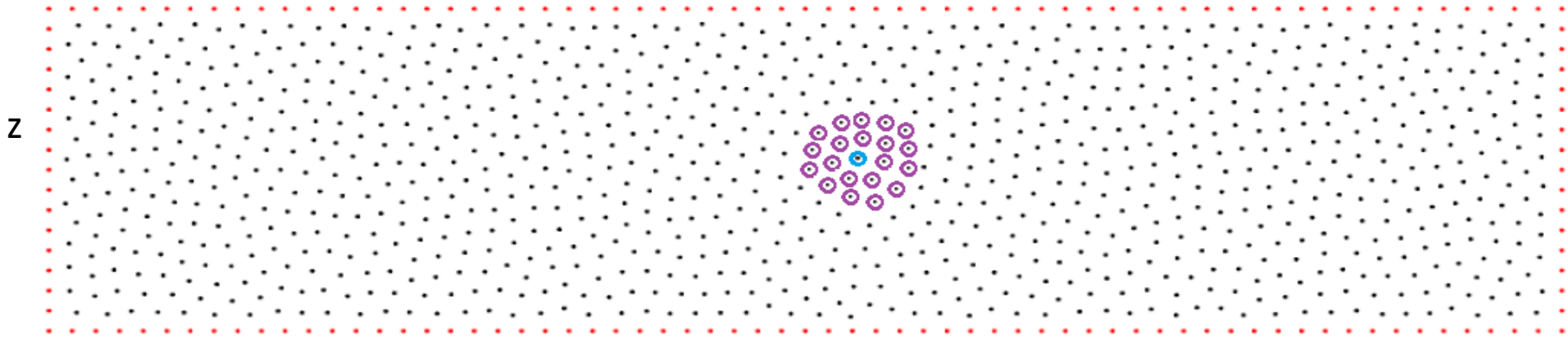


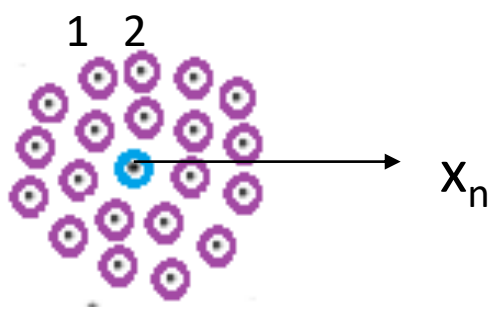


In a 3D

Simplicity of RBF-FD: Mesh-Free method

Ex.: Stencil of *n* = 21 nodes





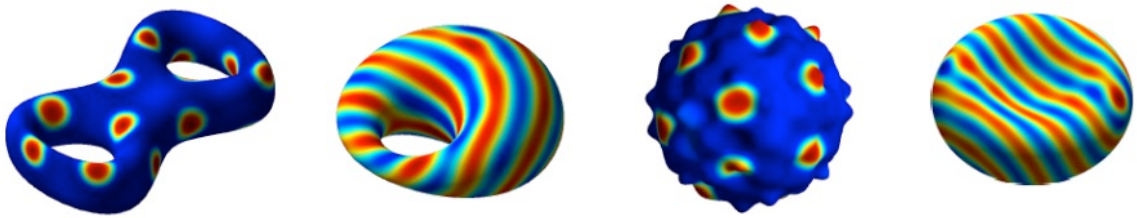
Get Gaussian matrix **A**, use Gaussian elimination to solve for weights

$$\begin{bmatrix} Exp(-r^2{}_{11}) & Exp(-r^2{}_{12}) & \dots Exp(-r^2{}_{1n}) \\ \vdots & \ddots & \vdots \\ Exp(-r^2{}_{n1}) & Exp(-r^2{}_{n2}) & \dots . Exp(-r^2{}_{nn}) \end{bmatrix} \begin{bmatrix} w_1 \\ \vdots \\ w_n \end{bmatrix} = \begin{bmatrix} d/dx[Exp(-r^2{}_1)]|_{\circ} \\ \vdots \\ d/dx[Exp(-r^2{}_n)]|_{\circ} \end{bmatrix}$$

Coding RBF-FD Method is FAST and EASY

```
IDX = knnsearch(xyz,xyz,'K',n);    % n is stencil size
for k  = 1:N                       % Loop over all points N in domain
   X   = xyz(IDX(k,:),:);          % nodes in the kth stencil
   r2  = (X(:,1) - X(:,1)').^2 + …
         (X(:,2) - X(:,2)').^2 + …
         (X(:,3) - X(:,3)').^2 ;   % Distance matrix
    A   = exp(-r2);                % RBF-FD matrix
   RHS_dx = -2*(X(:,1) – X(k,1)).* exp(-r2);   % derivative of GA w.r.t x,y,z
   RHS_dy = -2*(X(:,2) – X(k,2)).* exp(-r2);
   RHS_dz = -2*(X(:,3) – X(k,3)).* exp(-r2);
   Dx(k,:) = A\RHS_dx;             % Differentiation matrices (DM)
   Dy(k,:) = A\RHS_dy;
   Dz(k,:) = A\RHS_dz;
end
```

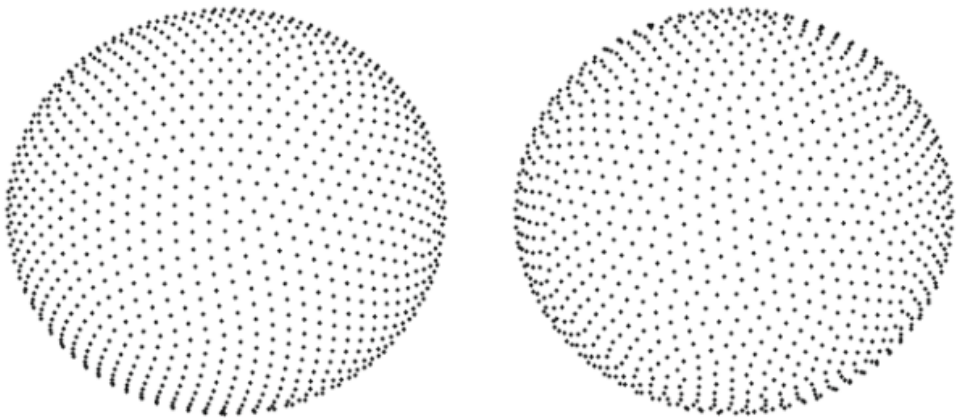
Have DMs for any geometry and point distribution in 3D space



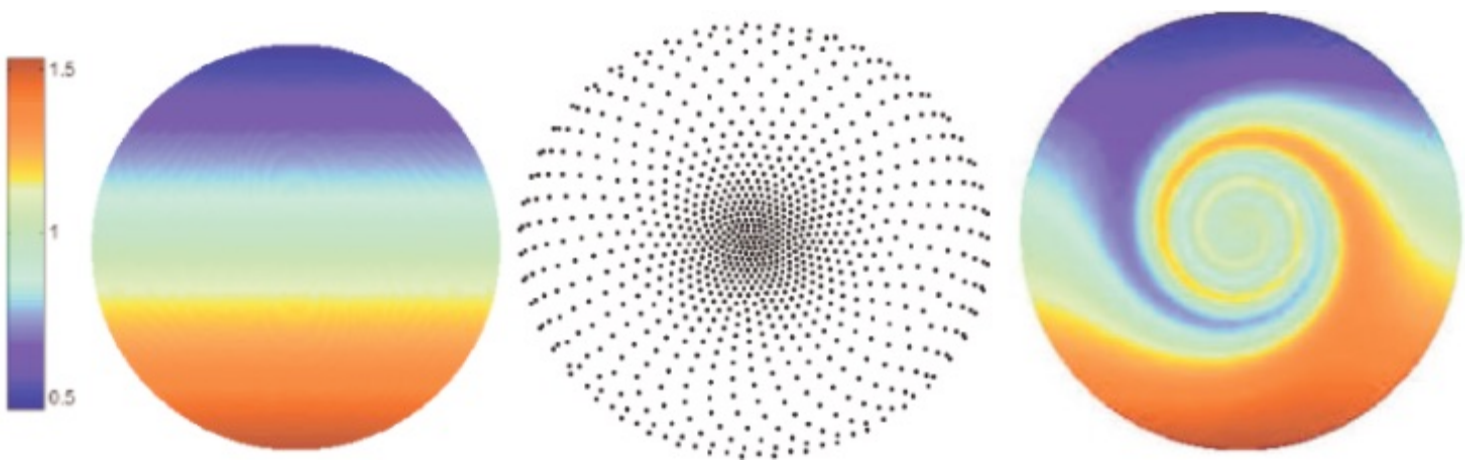
Translating Static Node Refinement (Flyer and Lehto, 2010, *JCP*)

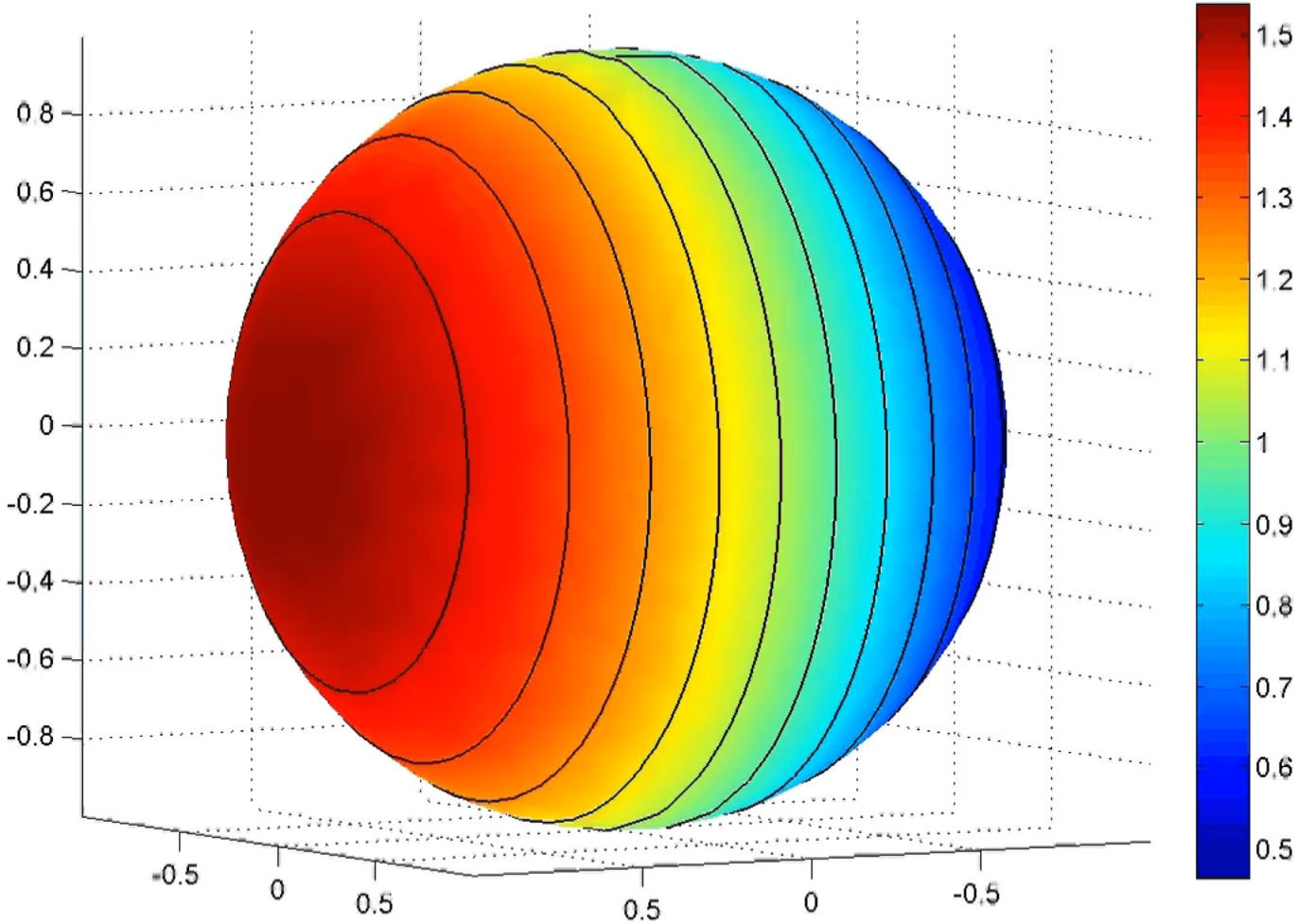
$$\partial h/\partial t = \vec{U}(\omega(lat.), long., t) \cdot \nabla h$$

http://web.maths.unsw.edu.au/~rsw/Sphere/



(a) ME Nodes (b) MD Nodes




0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-0.5
0
0.5
0.5
0
-0.5
1.5
1.4
1.3
1.2
1.1
1
0.9
0.8
0.7
0.6
0.5

Comparison between ME, MD, Refined, and other methods

Method	*N*	Δt (mins.)	RMS error
RBF, refined	900	60	5e-3
RBF, ME	3136	60	4e-3
RBF, MD	3136	60	5e-3
DG	9600	6	7e-3
FV	38,400	30	2e-3
FV, AMR	2500 to 165K	Variable	2e-3

How we Cluster Nodes and Variable Shape Parameter

Fine features in flow are formed where ω is large

$\Longrightarrow$ use ω to assign charge distribution for node repel

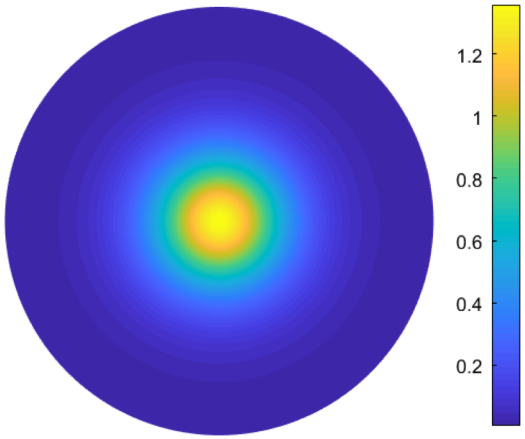
$$\omega(lat.) \propto \mathrm{sech}^2(lat.)\tanh(lat.) \Longrightarrow q(lat.) = [0.1 + c * \omega(lat.)]^{-1}$$

When clustering nodes, the shape parameter of the Gaussian Exp[-(εr)^2] must scale over the domain to avoid ill-conditioned matrices and Runge phenomena

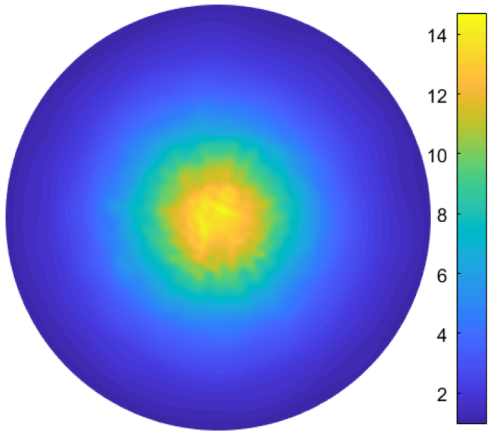
Rule-of-Thumb: $\varepsilon \propto$ Inverse of Euclidean distance to nearest neighbor

$\omega(lat.) \Longrightarrow q(lat.) \Longrightarrow$ Node distribution $\Longrightarrow \varepsilon$

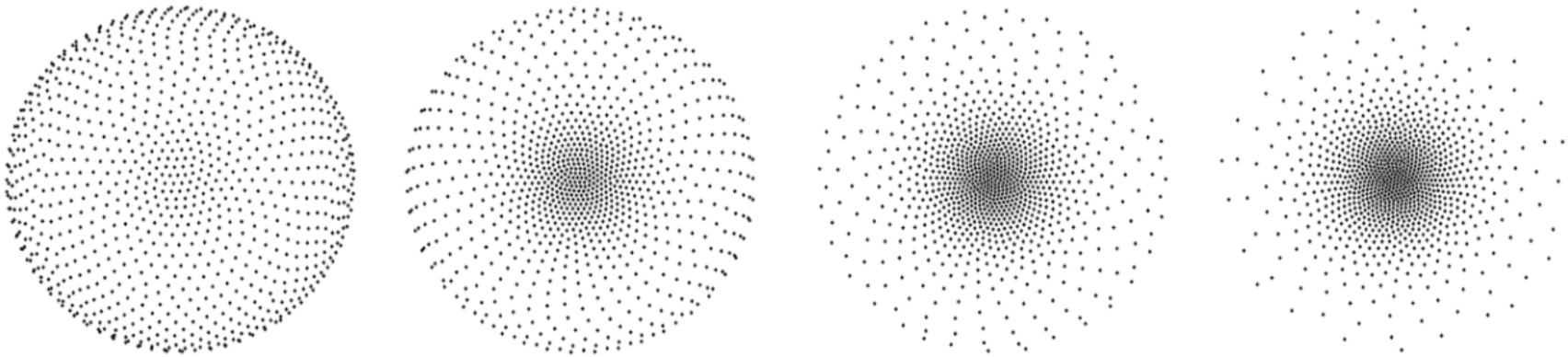
$\omega(lat.)$



ε

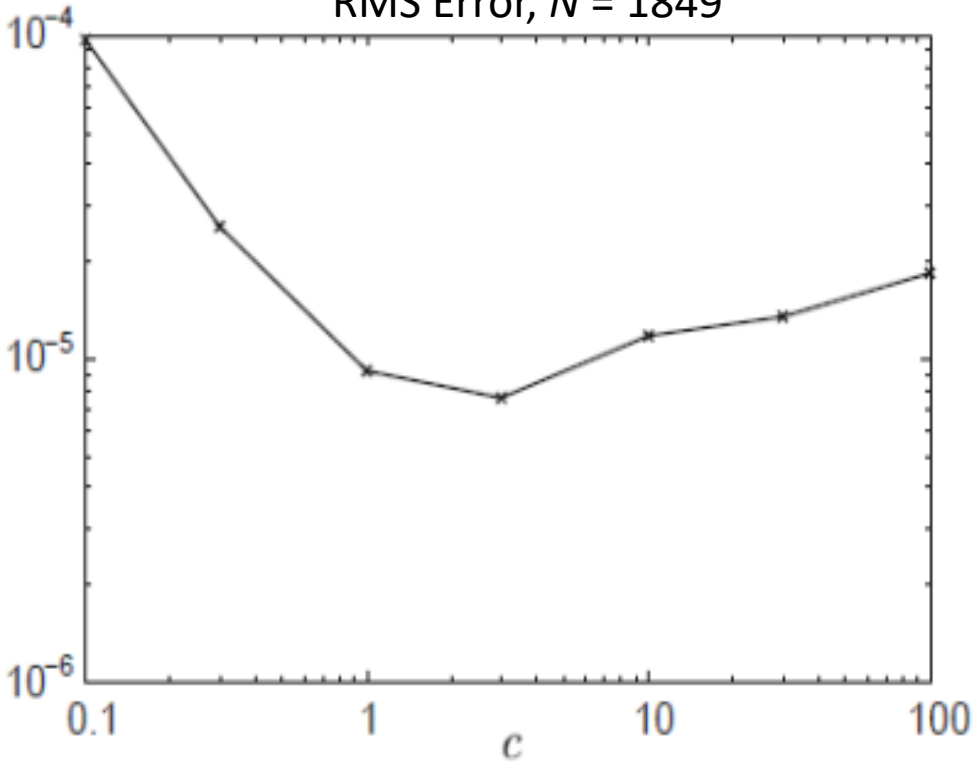


Effect of Clustering on Error



$c = 0.1$ $c = 1$ $c = 10$ $c = 100$

RMS Error, $N = 1849$



Dynamic Node Refinement: Simple Tropical Cyclogenesis

(in collaboration with Erik Lehto)

Barotropic Vorticity Equation

$$\frac{\partial \zeta}{\partial t} + u\frac{\partial \zeta}{\partial x} + v\frac{\partial \zeta}{\partial y} + \beta v = 0 \qquad \nabla^2 \psi = \zeta$$

ψ = streamfunction

ζ = relative vorticity

$$u = -\frac{\partial \psi}{\partial y} \quad v = \frac{\partial \psi}{\partial x}$$

- We do not know apriori where fine features will occur
- Need a <u>good</u> Monitor Function for node adaptation
- Generally, takes the form $M = M(\nabla f)$, f is a physical feature of the flow

$$M(x, y, t) = \sqrt{1 + \frac{|\nabla \zeta|^2}{\alpha}}$$

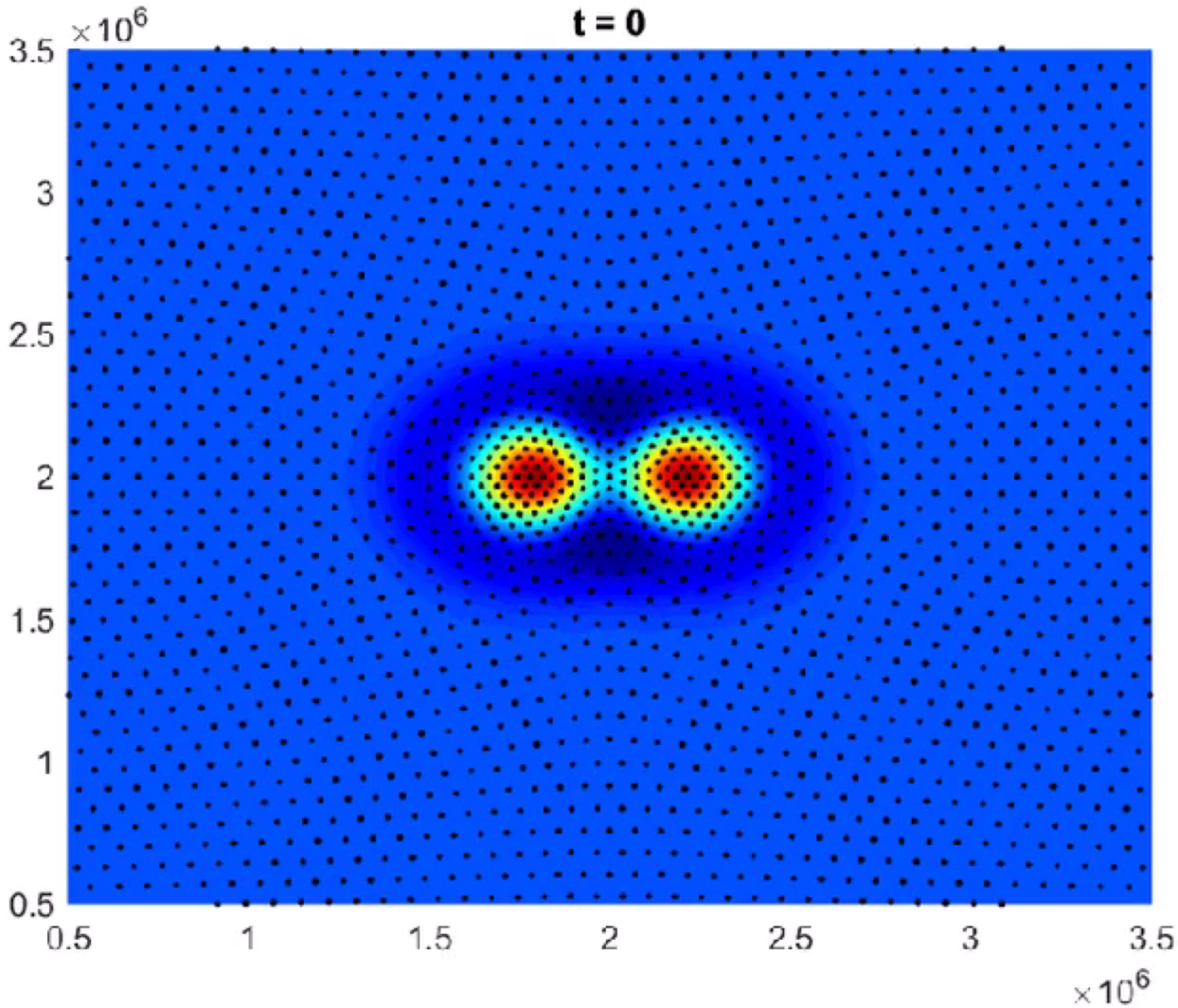
α is a scaling parameter

Steps in implementation

1. M at a given time is approximated with RBFs
2. Assign charge distribution according $1/M$ and repel
3. Evaluate solution at new pts. via RBF interpolation
4. Calculate spatially variable ε and recalculate RBF differentiation matrices

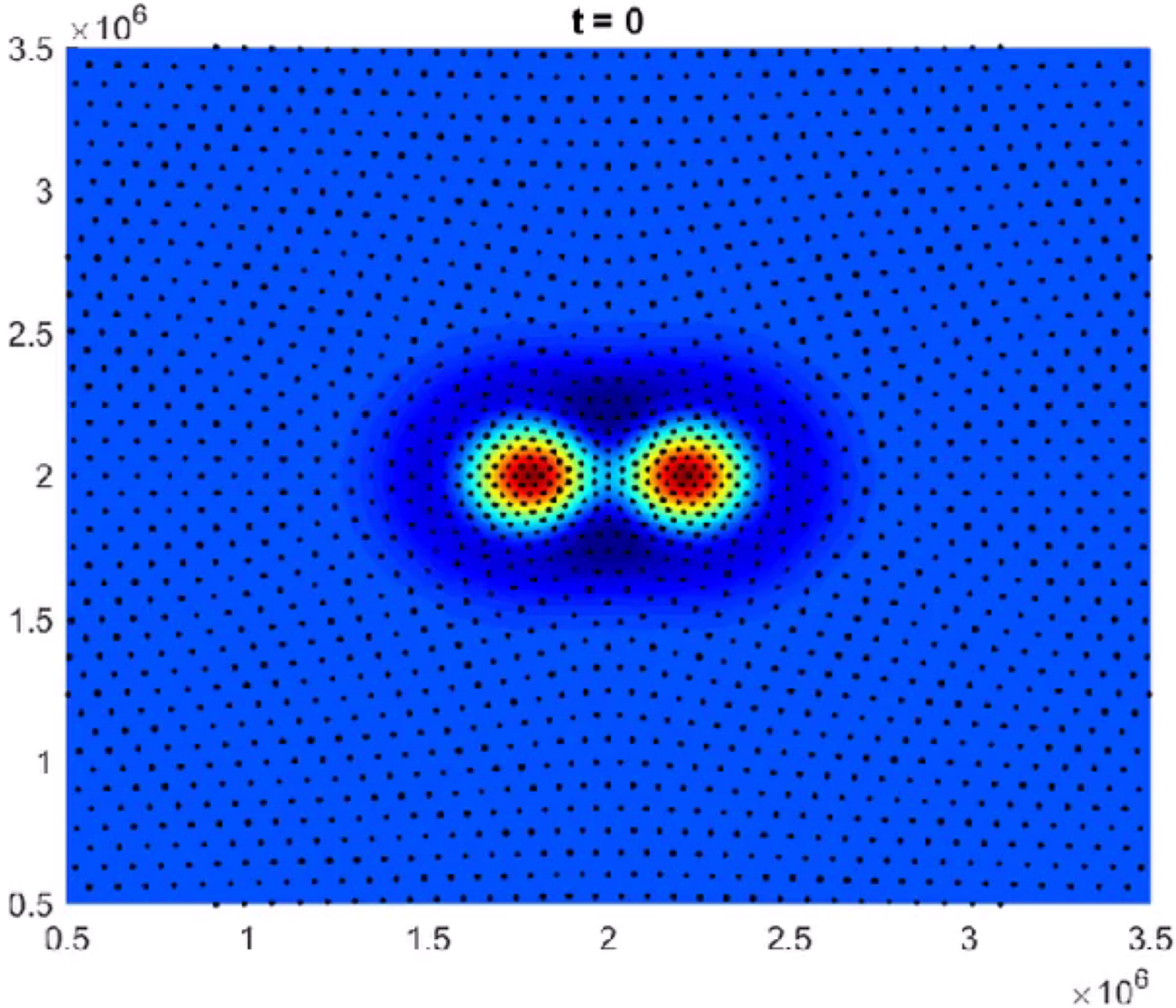
3500km x 3500km

Separation Dist: 400km



3500km x
3500km

Separation
Dist: 410km

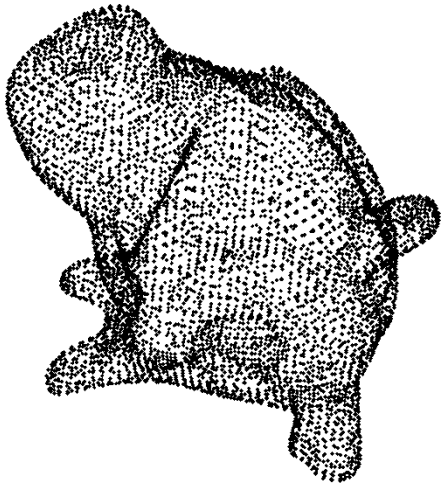



```
fname=['movie\mov' num2str(n) 'img' num2str(i)];
```

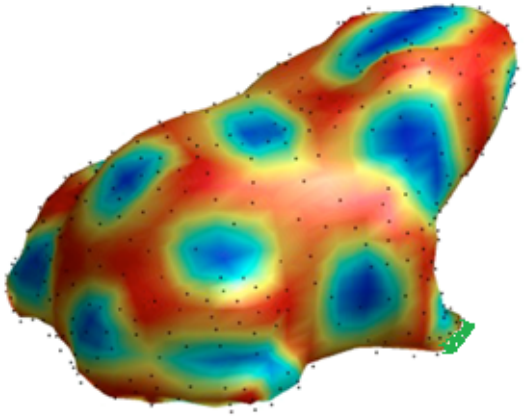
Coupled Reaction-diffusion equations over irregular surfaces (Piret, JCP 2012)

The *Brusselator equations* (Alan Turning) model pattern formation in nature,
Solved by RBF over the surface of a frog

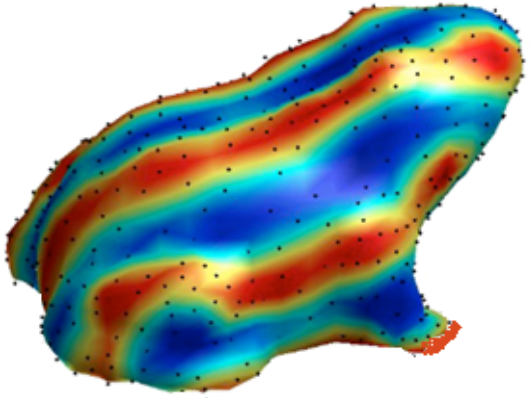
Snapshots from a computed time evolution for two different parameter regimes



Tabasara rain frog

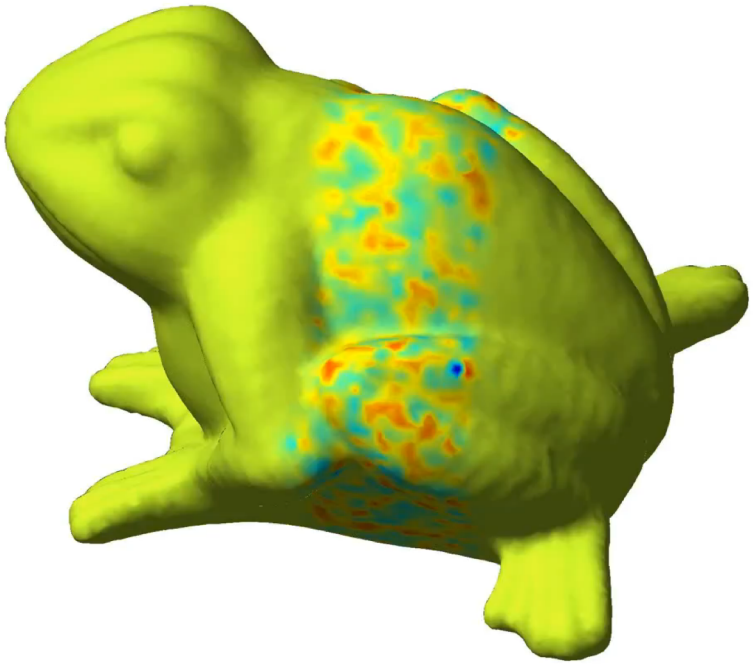


Poison dart frog



- RBF Node layout
- AIM@Shape Online Repository

Movie Courtesy of Grady Wright



Node generation algorithms

Iterative-Type Schemes

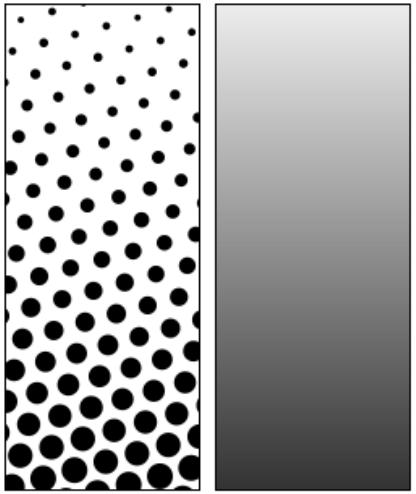
For a given number of nodes, the quality of the distribution depends on how soon the iteration is stopped.

- Min. energy distribution
- Voronoi relaxation
- Delaunay triangulations
  - DistMesh (Persson-Strang)
  - Gmsh (Geuzaine-Remacle)

Advancing-Front Type Schemes

For a given number of nodes, Start at a boundary and advance forward until the domain is filled.

- Dithering for half-tone images



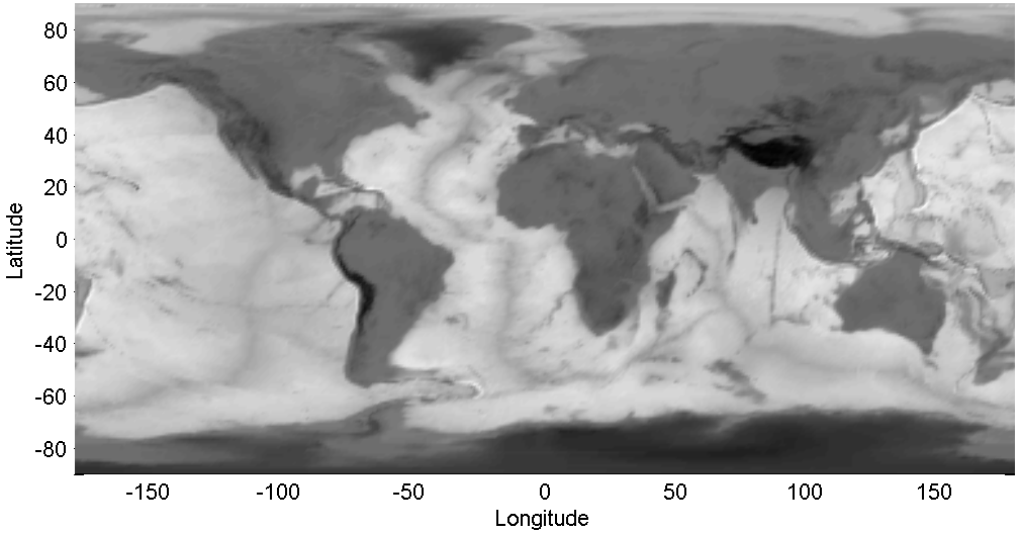
Notice nice hex pattern. Instead of changing width of dots, we change density of dots. We start at bottom boundary and march upward until domain is filled

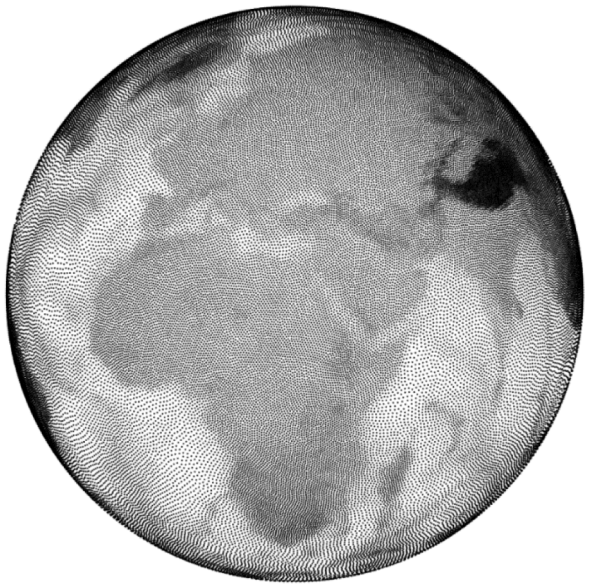
Half-tone image Human eye

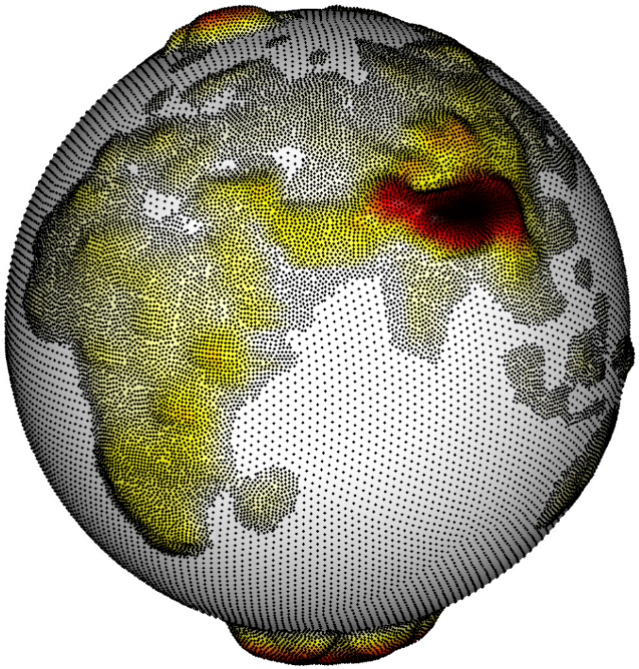
Distributing variable node density on sphere

(Fornberg and Flyer, 2015)

Below: Gray scale rendering of the file topo.mat in Matlab's Mapping toolbox







Top right: Advanicng Front Algorithm

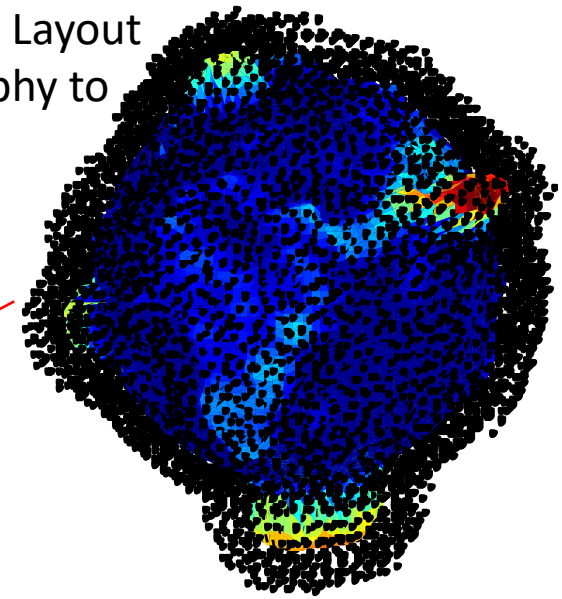
N = 105,419 nodes rendering of the topo map above
Computational speed in MATLAB still around 11,000 nodes per second.

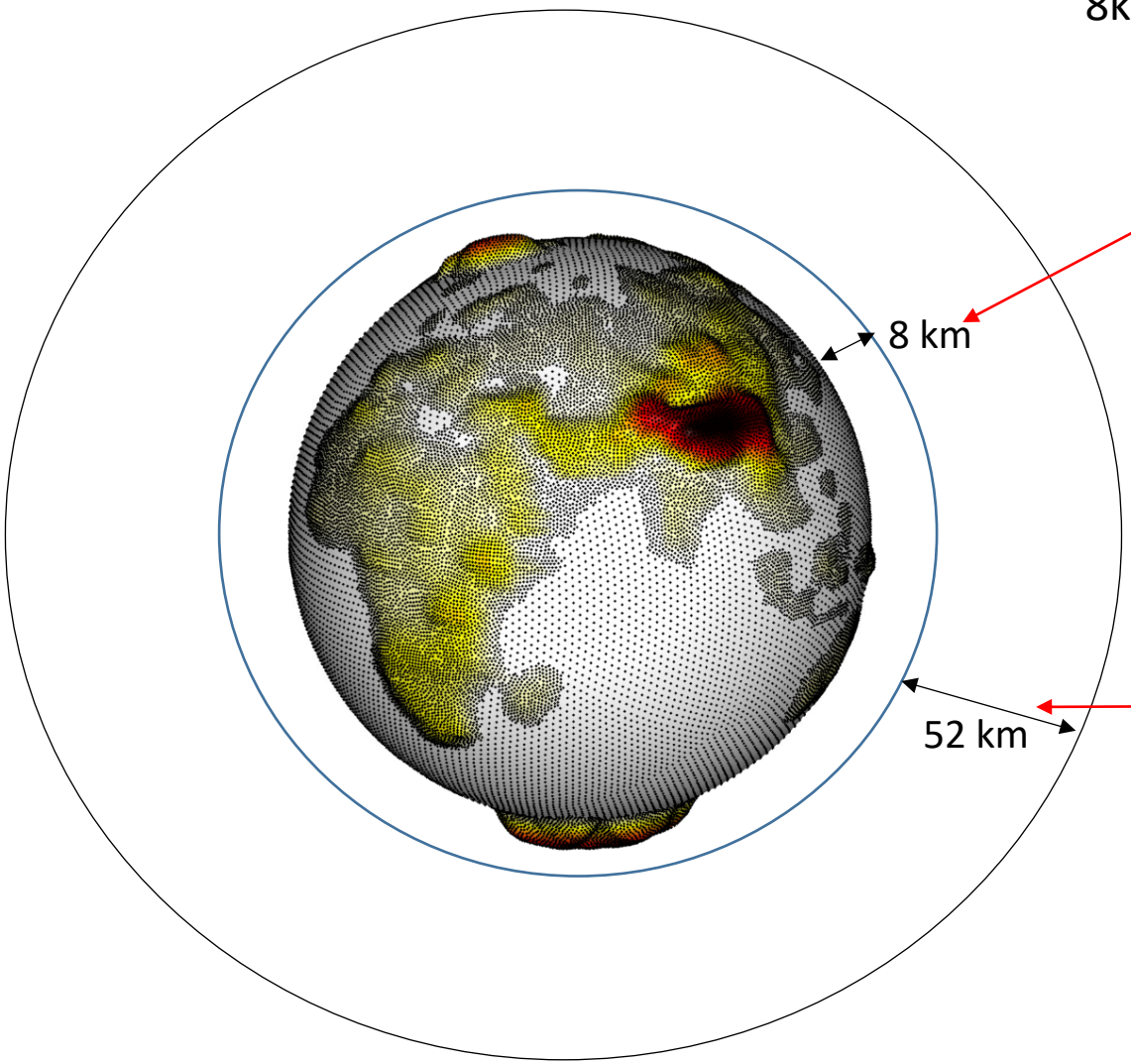
Next step in modeling (Bayona et al. 2015) :
Take elevation physically taken into account

3D Elliptic PDE: Modeling Electrical Currents in the Atmosphere

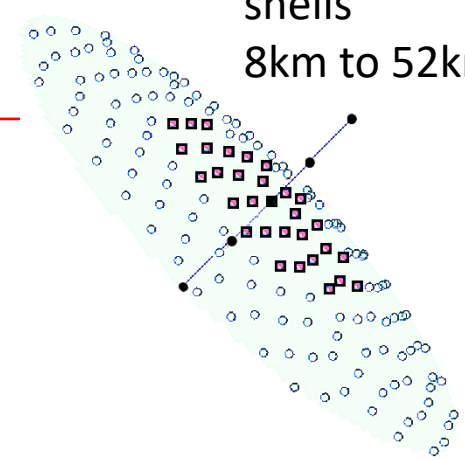
$$\nabla_{3D} \cdot \left(\underbrace{\sigma(x,y,z)}_{\text{Conductivity}} \nabla_{3D} \underbrace{u}_{\text{Elec. potential}}\right) = \underbrace{S(x,y,z)}_{\text{Thunderstorm sources (NASA measured data)}}$$

3D Node Layout
Topography to
8km

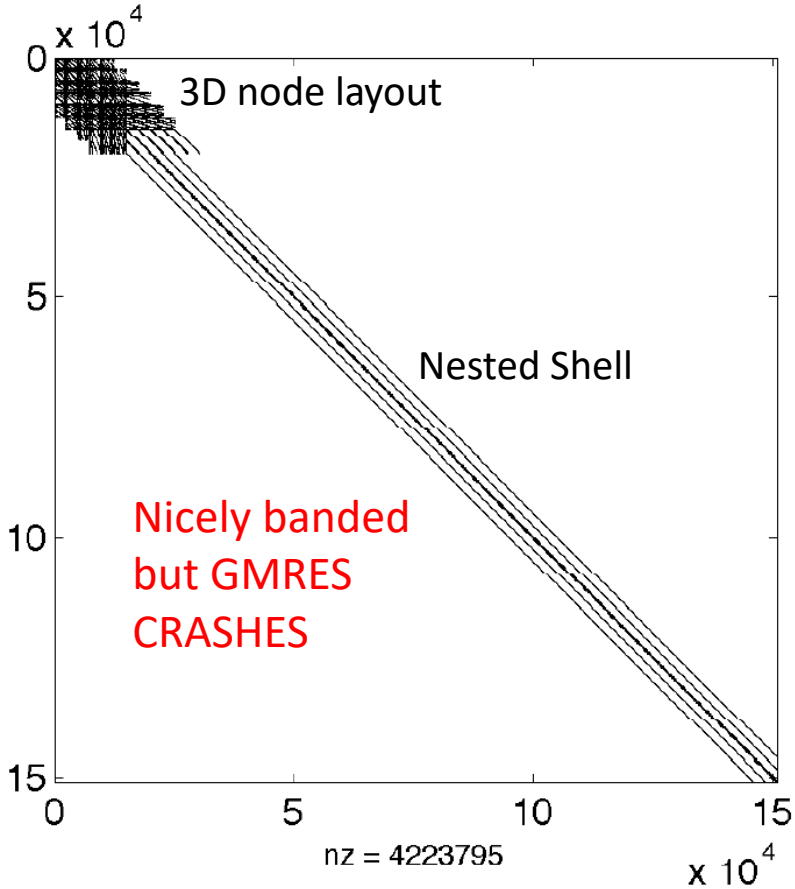




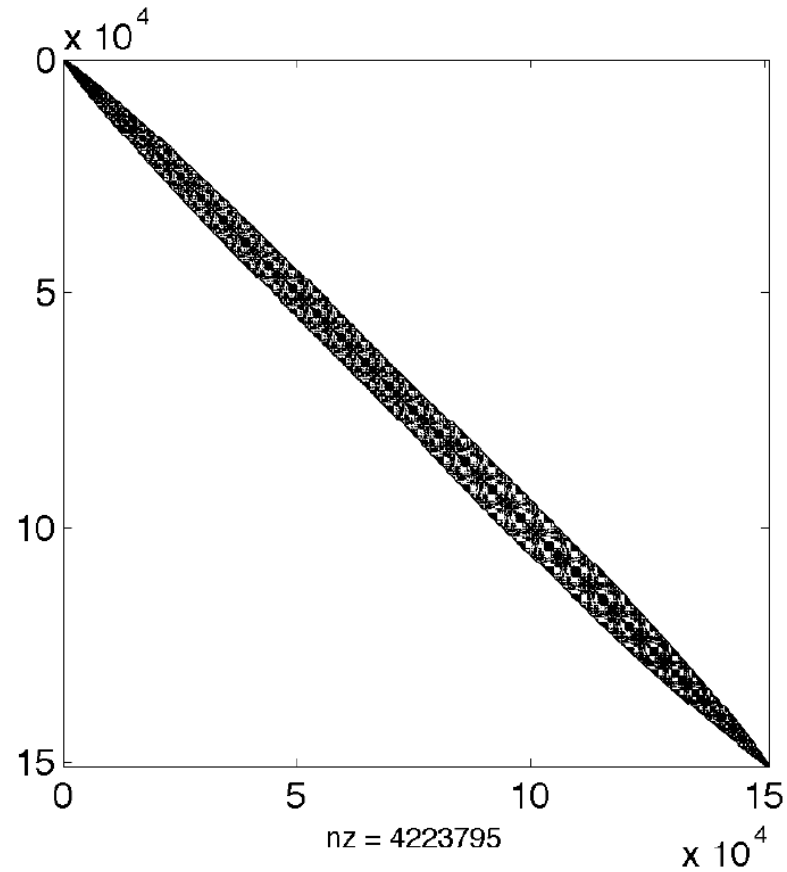
Nested spherical
shells
8km to 52km



Sparsity pattern of 3D elliptic operator (99.998% zeros)



Before any node reordering



After using reverse Cuthill- McKee

Result: Testing with data, 4.2M nodes
100 km. lat. – long. By 600m vertical, 31 mins on laptop using GMRES

GitHub Open Source Code: Bayona et al. , A 3-D RBF-FD solver for modelling the atmospheric Global Electric Circuit with topography (GEC-RBFFD v1.0), Geosci. Model Dev. 2015.

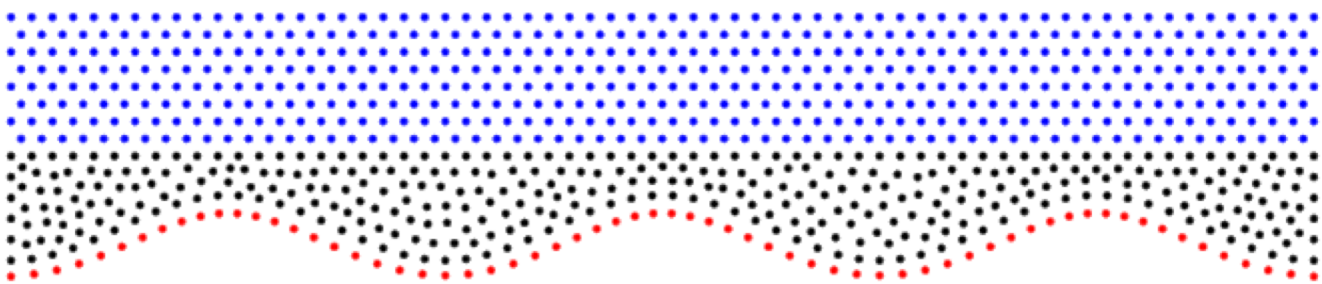
2D Compressible Navier-Stokes with Topography using RBF-FD

$$\frac{\partial \mathbf{u}}{\partial t} = -\left(\mathbf{u}\cdot\nabla\right)\mathbf{u} - c_p\theta\nabla P - g\mathbf{k} + \mu\Delta\mathbf{u}, \qquad \text{momentum}$$

$$\frac{\partial \theta}{\partial t} = -\left(\mathbf{u}\cdot\nabla\right)\theta + \mu\Delta\theta, \qquad \text{energy}$$

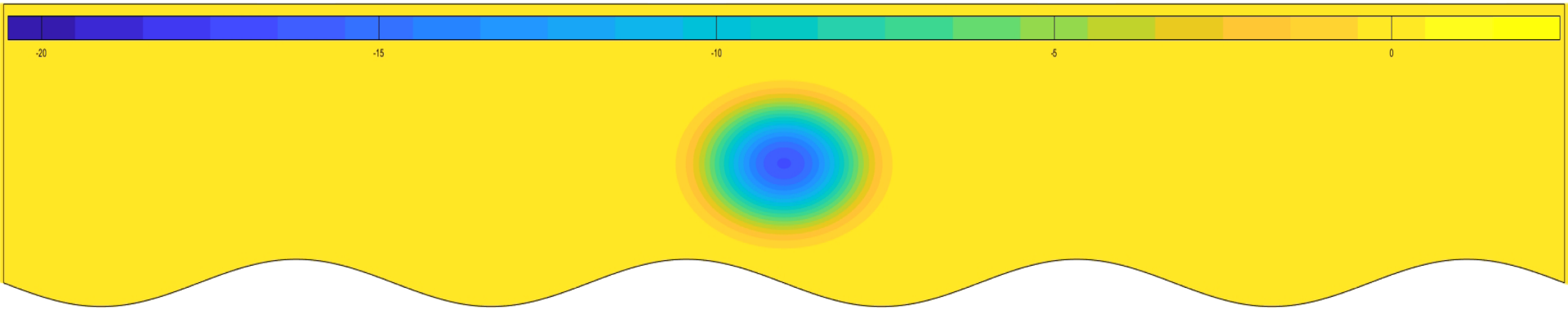
$$\frac{\partial P}{\partial t} = -\left(\mathbf{u}\cdot\nabla\right)P - \frac{R}{c_v}\left(\nabla\cdot\mathbf{u}\right)P, \qquad \text{mass}$$

Schematic node layout



Movie Courtesy of Gregory A. Barnett

Simulation of a cold downdraught in a dry atmosphere at 300K



Recent Review Material for RBF

1. N. Flyer, G.B. Wright, and B. Fornberg, 2014. *Radial basis function-generated finite differences: A mesh-free method for computational geosciences*, Handbook of Geomathematics, Springer-Verlag

2. B. Fornberg and N. Flyer, 2015 *Solving PDEs with Radial Basis Functions*, Acta Numerica.

3. B. Fornberg and N. Flyer, 2015 *A Primer on Radial Basis Functions with Applications to the Geosciences*, SIAM Press.