Optimally transported meshes on the Sphere for Global Atmospheric Modelling

Hilary Weller, Phil Browne, Chris Budd and Mike Cullen

University of Reading
METEOROLOGY

April 2016

r-Adaptivity

- **R**edistribution, keeping mesh topology fixed
- Deform mesh based on a monitor function
- Why?

r-Adaptivity

- **R**edistribution, keeping mesh topology fixed
- Deform mesh based on a monitor function
- Why?
  - No load balancing problems

r-Adaptivity

- **R**edistribution, keeping mesh topology fixed
- Deform mesh based on a monitor function
- Why?
  - No load balancing problems
  - No need to map solution between meshes

r-Adaptivity

- **R**edistribution, keeping mesh topology fixed
- Deform mesh based on a monitor function
- Why?
  - No load balancing problems
  - No need to map solution between meshes
  - Retro-fit to existing codes

r-Adaptivity

- **R**edistribution, keeping mesh topology fixed
- Deform mesh based on a monitor function
- Why?
  - No load balancing problems
  - No need to map solution between meshes
  - Retro-fit to existing codes
- Who not?
  - Distorted meshes
  - Never have complete control of Δx, Δy and Δz independently

r-Adaptivity

- **R**edistribution, keeping mesh topology fixed
- Deform mesh based on a monitor function
- Why?
  - No load balancing problems
  - No need to map solution between meshes
  - Retro-fit to existing codes
- Who not?
  - Distorted meshes
  - Never have complete control of Δx, Δy and Δz independently
- How?

Optimally Transported Meshes in Euclidean Geometry

- How to create a mesh which is equidistributed with respect to a monitor function. ie

$$A_x m(\mathbf{x}) = \text{const}$$

for each cell with area A_x for mesh monitor function $m(\mathbf{x})$.

Optimally Transported Meshes in Euclidean Geometry

- How to create a mesh which is equidistributed with respect to a monitor function. ie

$$A_x m(\mathbf{x}) = \text{const}$$

  for each cell with area A_x for mesh monitor function $m(\mathbf{x})$.
- Define a map from old mesh point, $\boldsymbol{\xi}_i$, to new mesh points, $\mathbf{x}_i$ which is the gradient of a convex potential, ϕ:

$$\mathbf{x} = \boldsymbol{\xi} + \nabla\phi$$

  This leads to an **"optimally transported mesh"**, free of tangling

Optimally Transported Meshes in Euclidean Geometry

- How to create a mesh which is equidistributed with respect to a monitor function. ie

$$A_x m(\mathbf{x}) = \text{const}$$

  for each cell with area A_x for mesh monitor function $m(\mathbf{x})$.
- Define a map from old mesh point, $\boldsymbol{\xi}_i$, to new mesh points, $\mathbf{x}_i$ which is the gradient of a convex potential, ϕ:

$$\mathbf{x} = \boldsymbol{\xi} + \nabla\phi$$

  This leads to an **"optimally transported mesh"**, free of tangling
- The Jacobian determinant of the mesh transform is the change in area of the cells:

$$|\nabla \mathbf{x}| = A_x / A_\xi$$

Optimally Transported Meshes in Euclidean Geometry

- How to create a mesh which is equidistributed with respect to a monitor function. ie

$$A_x m(\mathbf{x}) = \text{const}$$

  for each cell with area A_x for mesh monitor function $m(\mathbf{x})$.
- Define a map from old mesh point, $\boldsymbol{\xi}_i$, to new mesh points, $\mathbf{x}_i$ which is the gradient of a convex potential, ϕ:

$$\mathbf{x} = \boldsymbol{\xi} + \nabla\phi$$

  This leads to an **"optimally transported mesh"**, free of tangling
- The Jacobian determinant of the mesh transform is the change in area of the cells:

$$|\nabla \mathbf{x}| = A_x / A_\xi$$

- Combine these to form a Monge-Ampére equation for the mesh potential, ϕ:

$$\begin{aligned} |\nabla(\boldsymbol{\xi} + \nabla\phi)| m(\mathbf{x}) &= \text{const} \\ \text{or } |I + H(\phi)| &= \frac{\text{const}}{m(\mathbf{x})} \end{aligned}$$

Optimally Transported Meshes in Euclidean Geometry

- How to create a mesh which is equidistributed with respect to a monitor function. ie

$$A_x m(\mathbf{x}) = \text{const}$$

  for each cell with area A_x for mesh monitor function $m(\mathbf{x})$.
- Define a map from old mesh point, $\boldsymbol{\xi}_i$, to new mesh points, $\mathbf{x}_i$ which is the gradient of a convex potential, ϕ:

$$\mathbf{x} = \boldsymbol{\xi} + \nabla\phi$$

  This leads to an **"optimally transported mesh"**, free of tangling
- The Jacobian determinant of the mesh transform is the change in area of the cells:

$$|\nabla \mathbf{x}| = A_x / A_\xi$$

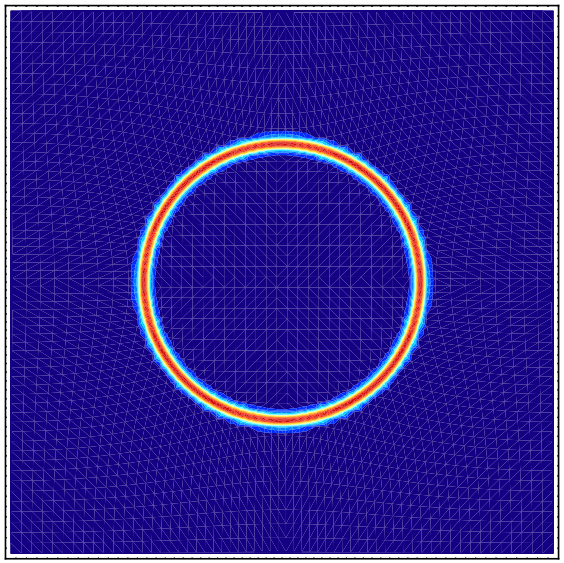
- Combine these to form a Monge-Ampére equation for the mesh potential, ϕ:

$$\begin{aligned} |\nabla(\boldsymbol{\xi} + \nabla\phi)| m(\mathbf{x}) &= \text{const} \\ \text{or } |I + H(\phi)| &= \frac{\text{const}}{m(\mathbf{x})} \end{aligned}$$

- This works fine in Euclidean geometry ...

Numerical Solution of the Monge-Ampère Equation on a Plane (test case from Budd et al. [2015])

Monitor Function, m

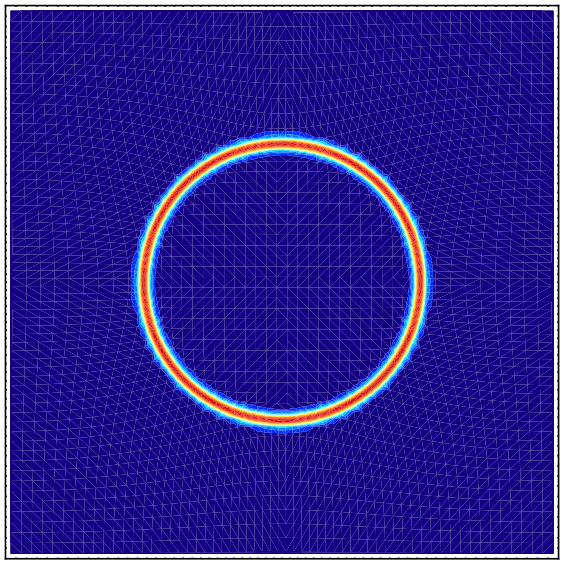


Initial Mesh

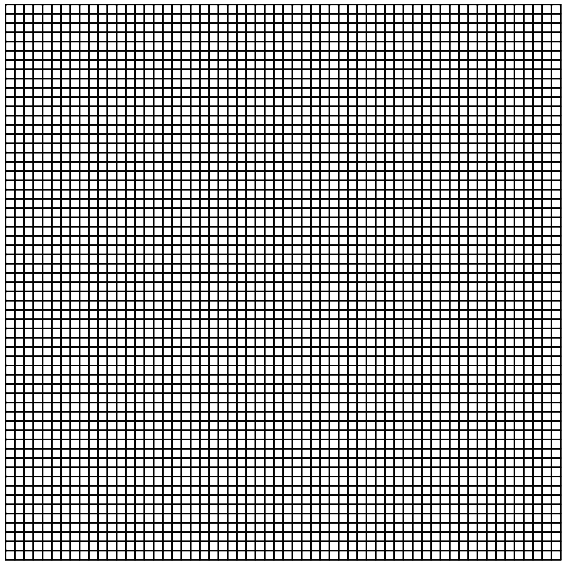
Solution of
$|I + H(\phi)|m(\mathbf{x}) = \text{const}$

Numerical Solution of the Monge-Ampère Equation on a Plane (test case from Budd et al. [2015])

Monitor Function, m



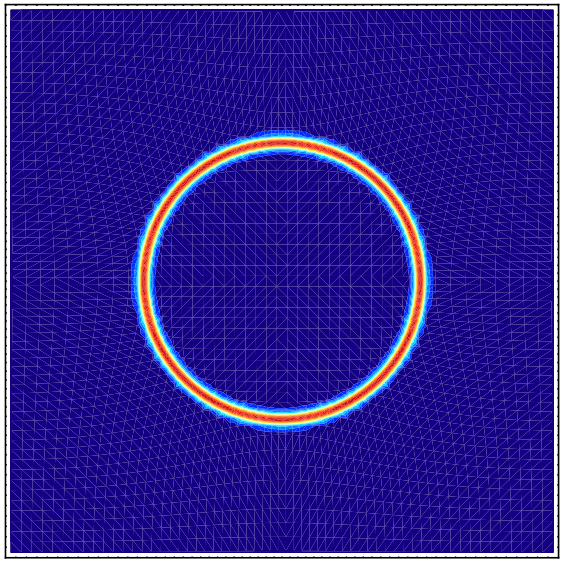
Initial Mesh



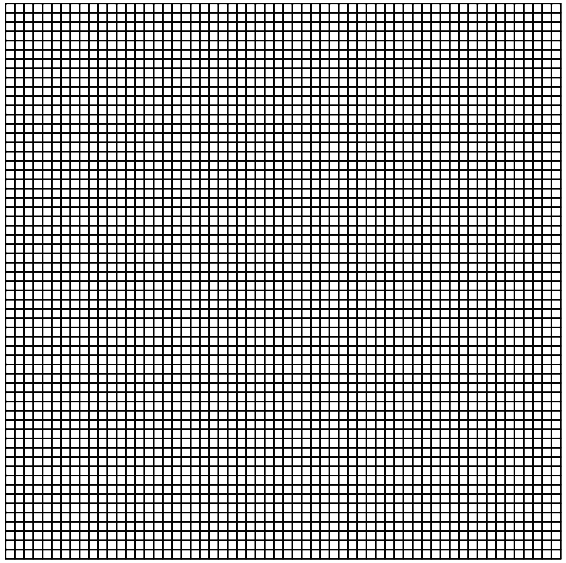
Solution of
$|I + H(\phi)|m(\mathbf{x}) = \text{const}$

Numerical Solution of the Monge-Ampère Equation on a Plane (test case from Budd et al. [2015])

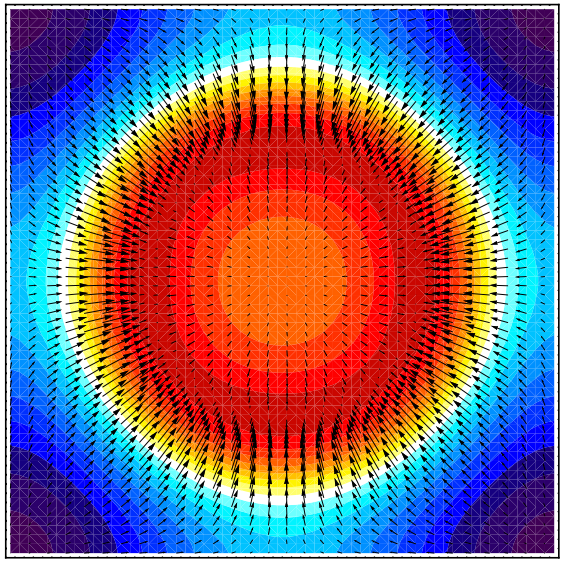
Monitor Function, m



Initial Mesh

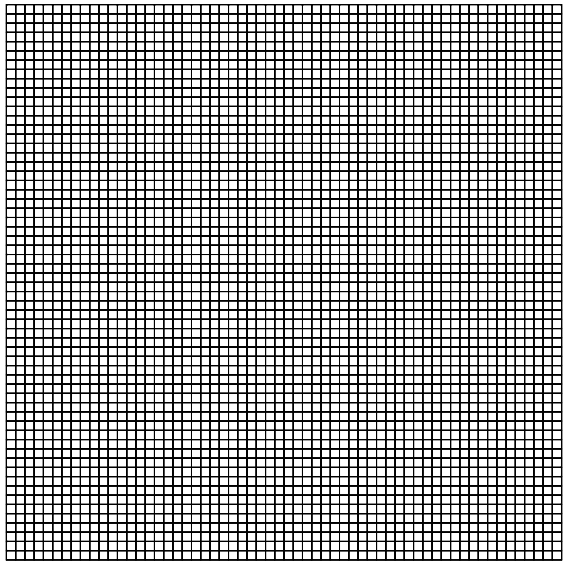


Solution of $|I + H(\phi)|m(\mathbf{x}) = \text{const}$



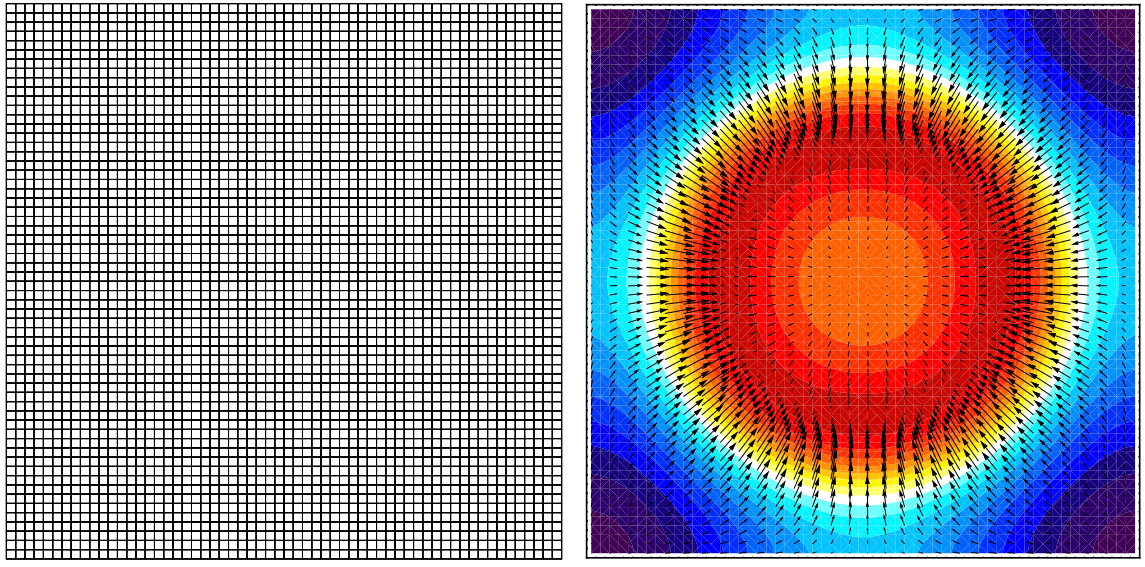
Initial Mesh + $\nabla\phi$ gives an equidistributed mesh

Initial Mesh, $\boldsymbol{\xi}$ $+\nabla\phi$ $=\mathbf{x}$



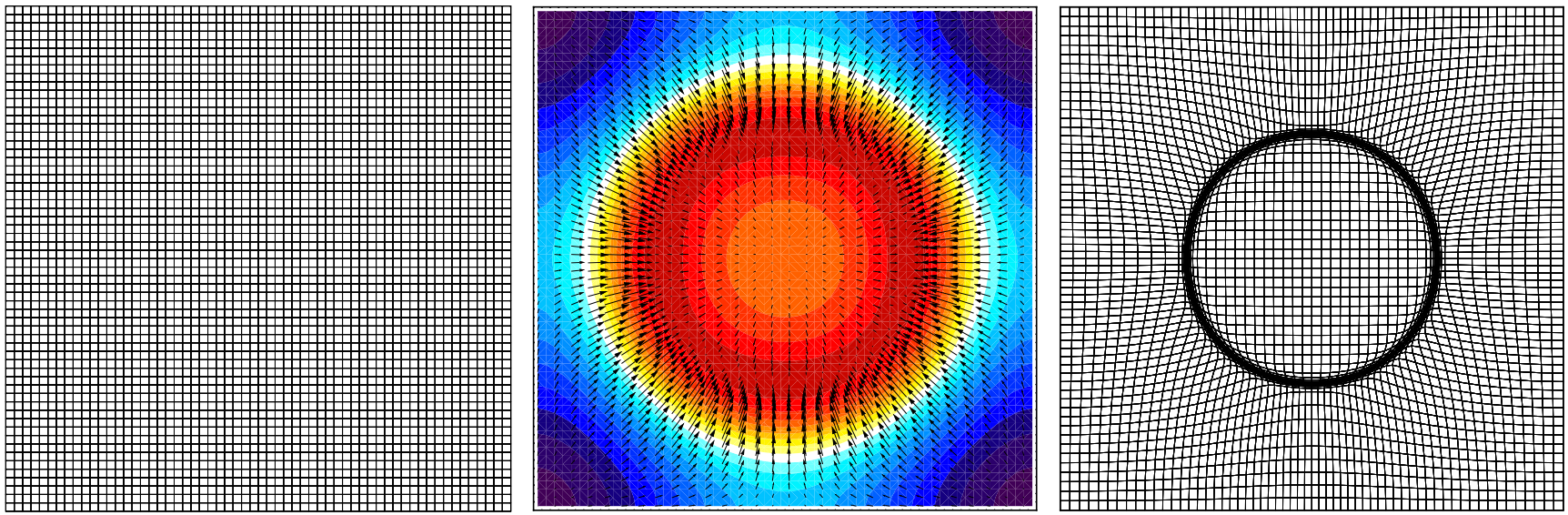
Initial Mesh + $\nabla\phi$ gives an equidistributed mesh

Initial Mesh, $\boldsymbol{\xi}$ $+\nabla\phi$ $=\mathbf{x}$



Initial Mesh + $\nabla\phi$ gives an equidistributed mesh

Initial Mesh, $\boldsymbol{\xi}$ $+\nabla\phi$ $=\mathbf{x}$



But on the Surface of a Sphere

- The gradient of a potential does not map to another point on the sphere
- $\therefore$ cannot formulate a Monge-Ampère equation for the mesh pontential

Instead:

But on the Surface of a Sphere

- The gradient of a potential does not map to another point on the sphere
- $\therefore$ cannot formulate a Monge-Ampère equation for the mesh pontential

  Instead:
- Use exponential maps to map from old to new mesh:

$$\mathbf{x} = \boldsymbol{\xi} + \exp_{\xi} \nabla\phi$$

-

But on the Surface of a Sphere

- The gradient of a potential does not map to another point on the sphere
- $\therefore$ cannot formulate a Monge-Ampère equation for the mesh pontential

  Instead:
- Use exponential maps to map from old to new mesh:

$$\mathbf{x} = \boldsymbol{\xi} + \exp_{\xi} \nabla\phi$$

- Solve the optimal transport problem more directly:

$$\frac{A_x}{A_\xi} = \frac{\text{const}}{m(\mathbf{x})} \tag{1}$$

  How?

But on the Surface of a Sphere

- The gradient of a potential does not map to another point on the sphere
- $\therefore$ cannot formulate a Monge-Ampère equation for the mesh pontential

  Instead:
- Use exponential maps to map from old to new mesh:

$$\mathbf{x} = \boldsymbol{\xi} + \exp_{\xi} \nabla\phi$$

- Solve the optimal transport problem more directly:

$$\frac{A_x}{A_\xi} = \frac{\text{const}}{m(\mathbf{x})} \tag{1}$$

  How?
- Informed by the Euclidean case, linearise eqn (1) (assuming tangent plane):

$$\frac{A_x}{A_\xi} \approx |I + H(\phi)| \approx$$

But on the Surface of a Sphere

- The gradient of a potential does not map to another point on the sphere
- ∴ cannot formulate a Monge-Ampère equation for the mesh pontential

  Instead:
- Use exponential maps to map from old to new mesh:

$$\mathbf{x} = \boldsymbol{\xi} + \exp_{\xi} \nabla\phi$$

- Solve the optimal transport problem more directly:

$$\frac{A_x}{A_\xi} = \frac{\text{const}}{m(\mathbf{x})} \tag{1}$$

  How?
- Informed by the Euclidean case, linearise eqn (1) (assuming tangent plane):

$$\frac{A_x}{A_\xi} \approx |I + H(\phi)| \approx 1 + \nabla^2\phi$$

-

But on the Surface of a Sphere

- The gradient of a potential does not map to another point on the sphere
- $\therefore$ cannot formulate a Monge-Ampère equation for the mesh pontential

  Instead:
- Use exponential maps to map from old to new mesh:

$$\mathbf{x} = \boldsymbol{\xi} + \exp_{\xi} \nabla\phi$$

- Solve the optimal transport problem more directly:

$$\frac{A_x}{A_\xi} = \frac{\text{const}}{m(\mathbf{x})} \tag{1}$$

  How?
- Informed by the Euclidean case, linearise eqn (1) (assuming tangent plane):

$$\frac{A_x}{A_\xi} \approx |I + H(\phi)| \approx 1 + \nabla^2\phi$$

- and create Poincaré iterations (with iteration counter n) with non-linear terms treated explicitly:

$$1 + \nabla^2\phi^{n+1} = 1 + \nabla^2\phi^n - \frac{A_x}{A_\xi} + \frac{\text{const}^n}{m(\mathbf{x}^n)}$$

Iterative Solution

This is unstable if the non-linear terms are too big. Therefore under-relax:

$$\alpha\nabla^2\phi^{(n+1)} = \alpha\nabla^2\phi^{(n)} - \frac{A_x}{A_\xi} + \frac{\text{const}^n}{m\left(\mathbf{x}^n\right)}$$

where $\alpha > 1$. How to find α?

Iterative Solution

This is unstable if the non-linear terms are too big. Therefore under-relax:

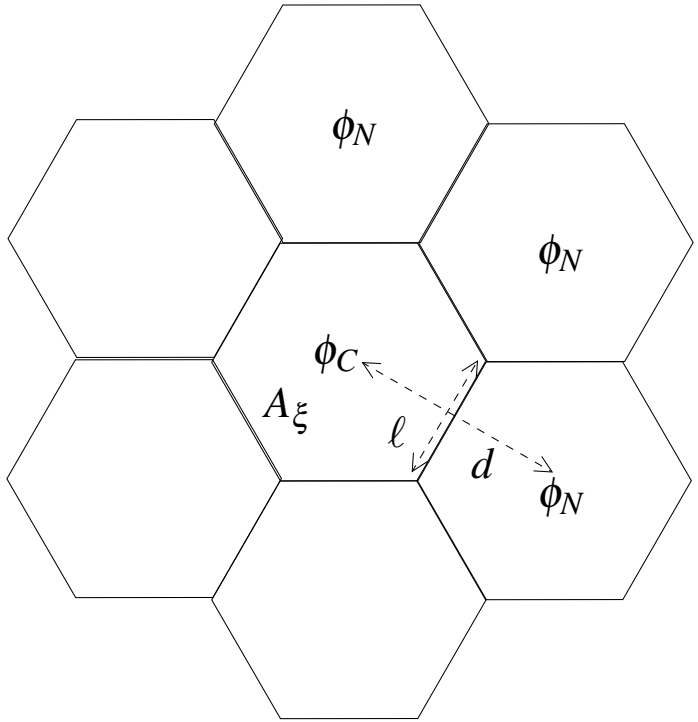
$$\alpha\nabla^2\phi^{(n+1)} = \alpha\nabla^2\phi^{(n)} - \frac{A_x}{A_\xi} + \frac{\text{const}^n}{m\left(\mathbf{x}^n\right)}$$

where $\alpha > 1$. How to find α?

$$\alpha^{n+1} = \max\left(\alpha^n,\ 4\max\left({}^1\!/_4,\ \max\left(\left|\frac{A_x}{A_\xi} - \frac{\text{const}^n}{m\left(\mathbf{x}^n\right)}\right|\right)\right)\right)$$

Spatial Discretisation

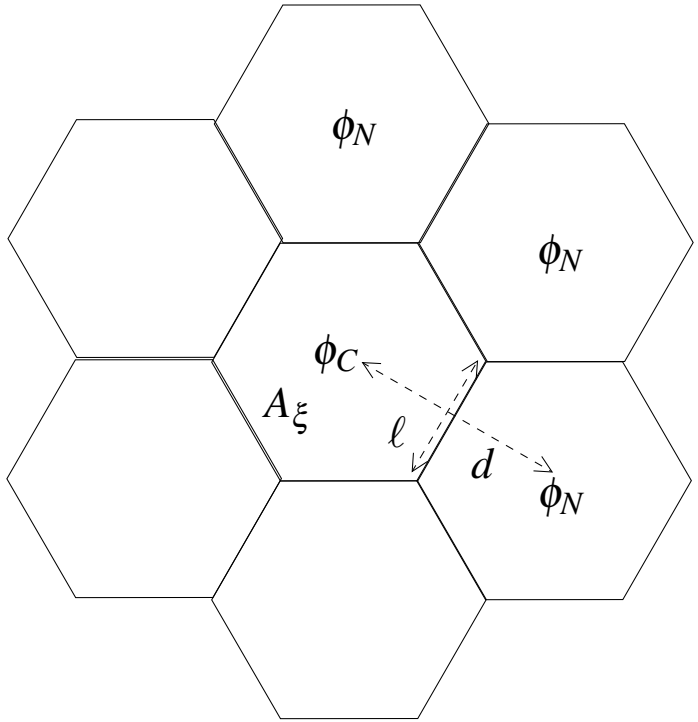
- A standard finite volume discretisation for $\nabla^2\phi$ on an arbitrary mesh:



$$\nabla^2\phi \approx \frac{1}{A_\xi}\sum_{\text{edges}} \frac{\phi_N - \phi_C}{d}\ell$$

Spatial Discretisation

- A standard finite volume discretisation for $\nabla^2\phi$ on an arbitrary mesh:

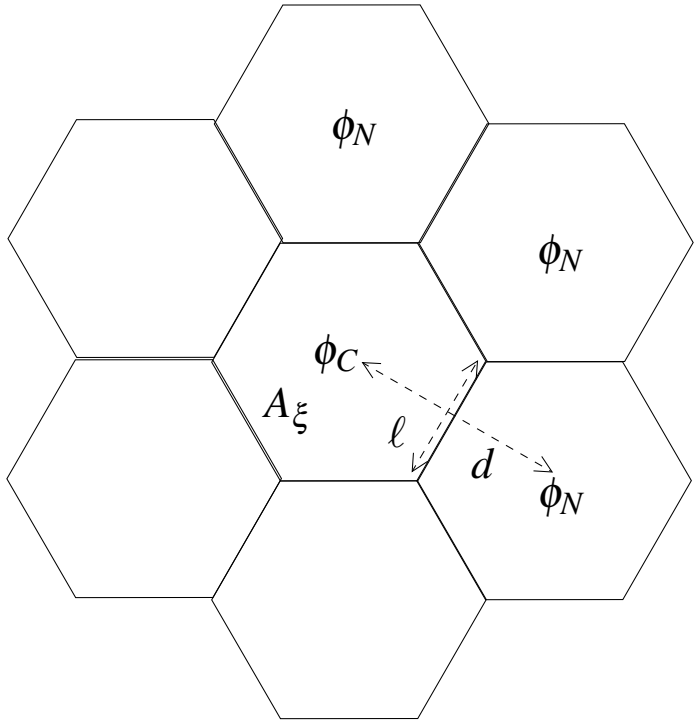


$$\nabla^2\phi \approx \frac{1}{A_\xi}\sum_{\text{edges}} \frac{\phi_N - \phi_C}{d}\ell$$

- No discretisation needed for A_x/A_ξ

Spatial Discretisation

- A standard finite volume discretisation for $\nabla^2\phi$ on an arbitrary mesh:

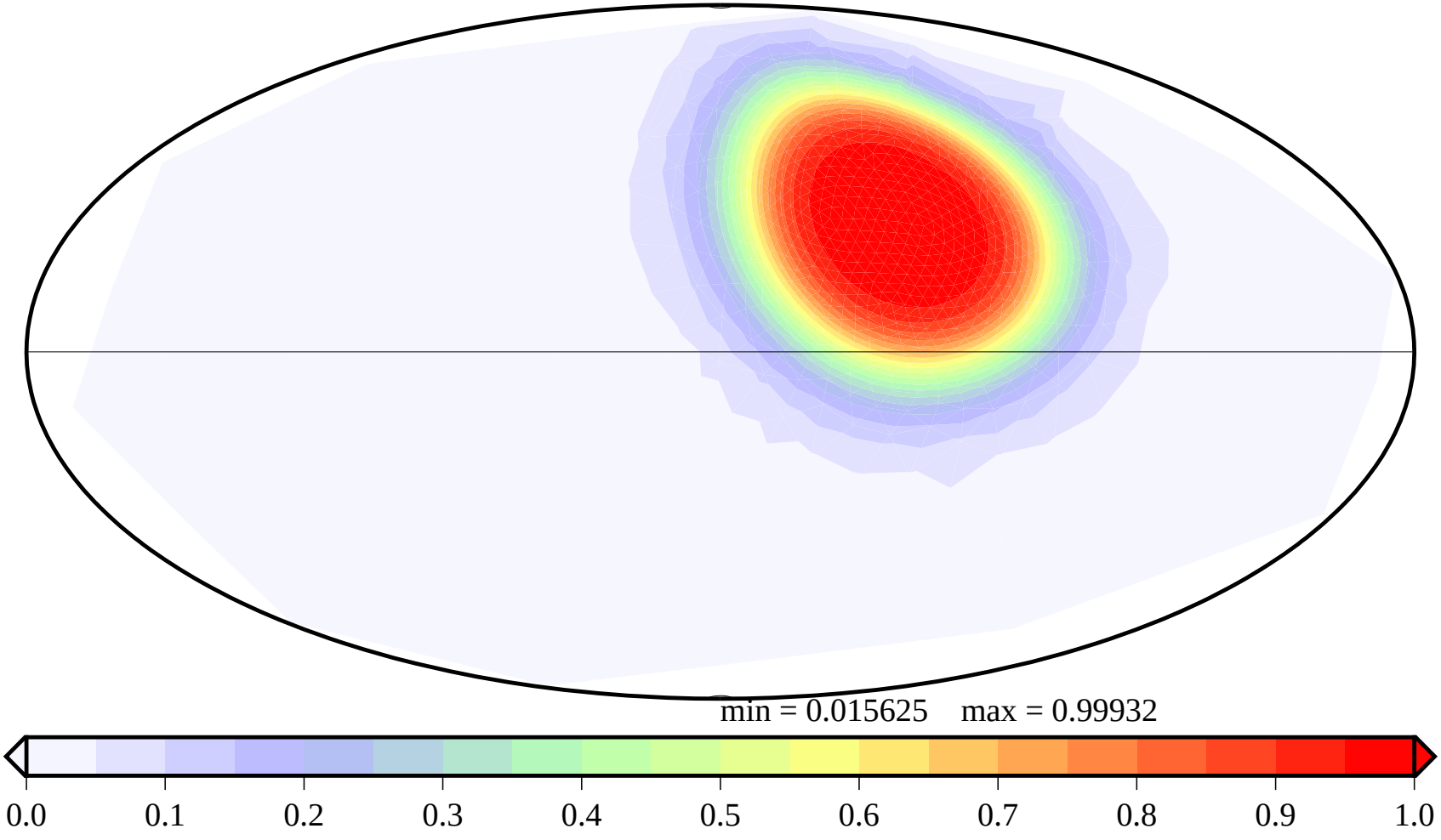


$$\nabla^2\phi \approx \frac{1}{A_\xi}\sum_{\text{edges}} \frac{\phi_N - \phi_C}{d}\ell$$

- No discretisation needed for A_x/A_ξ
- Solution is very sensitive to the discretisation of $\nabla\phi$ (at the vertices)
  - Least squares fit using values of ϕ at 6 hexagons around a vertex

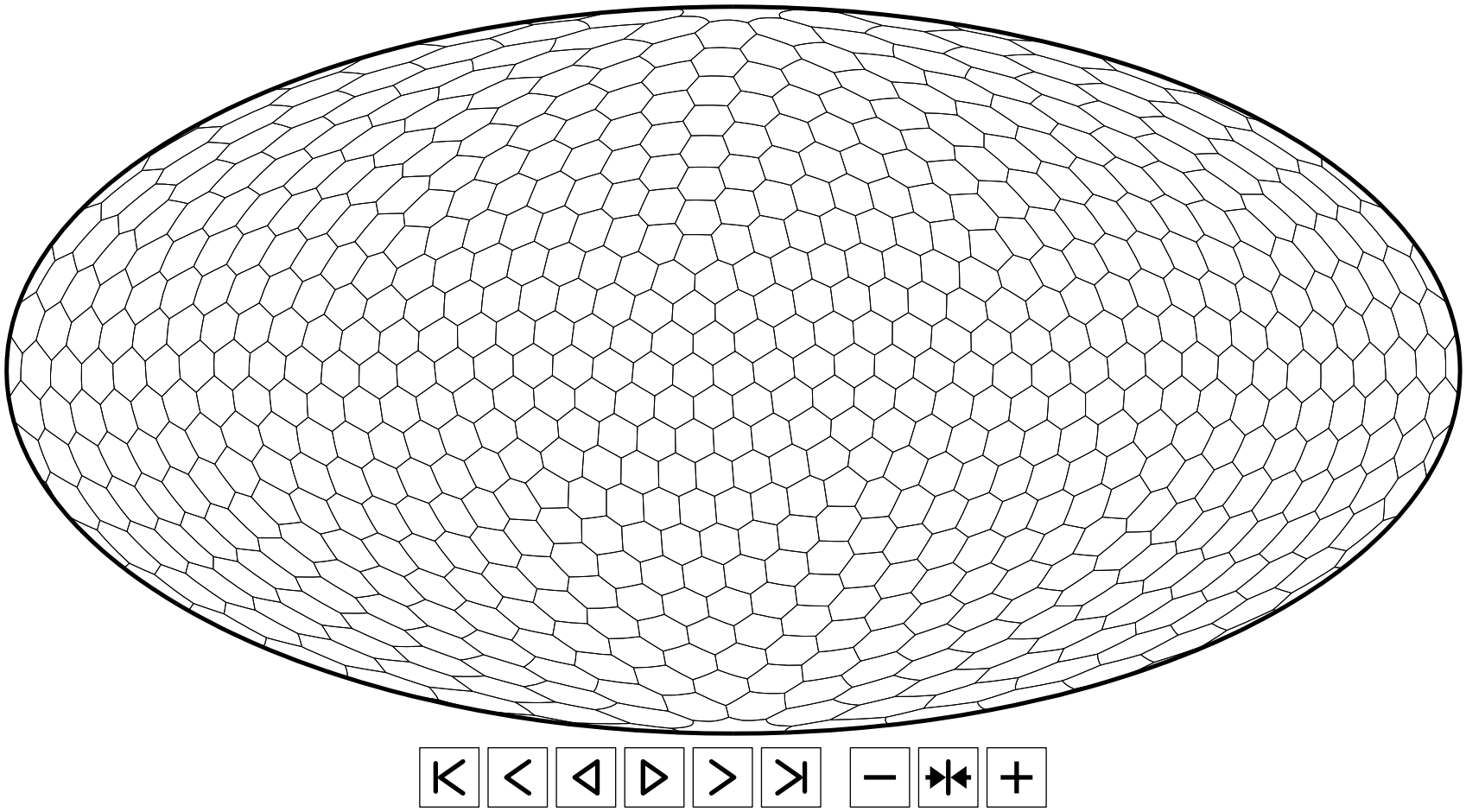
An Optimally Transported Mesh

Given the monitor function on the sphere:



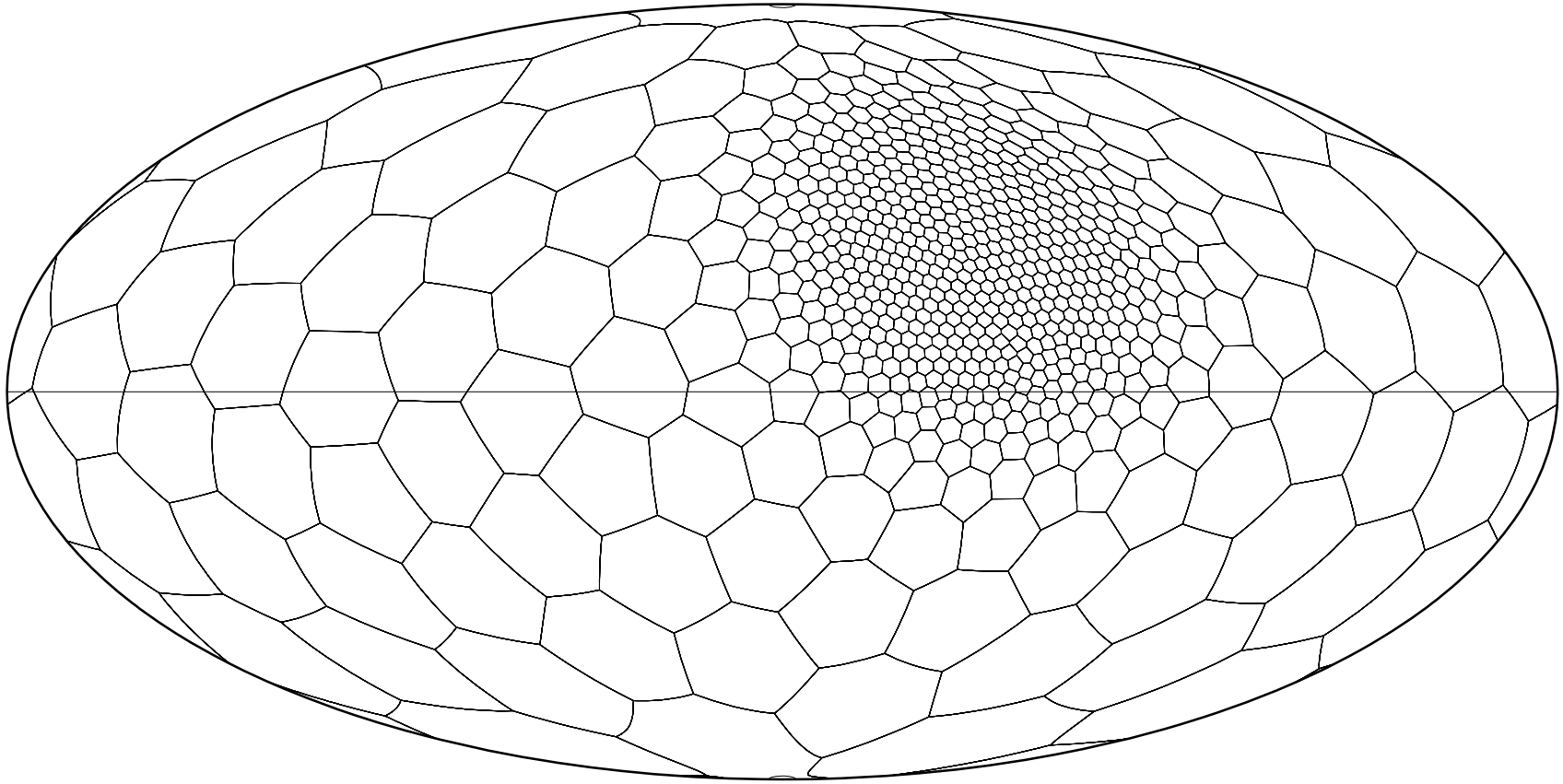
An Optimally Transported Mesh

The optimally transported mesh is calculated iteratively:



Compared with Lloyd's Algorithm

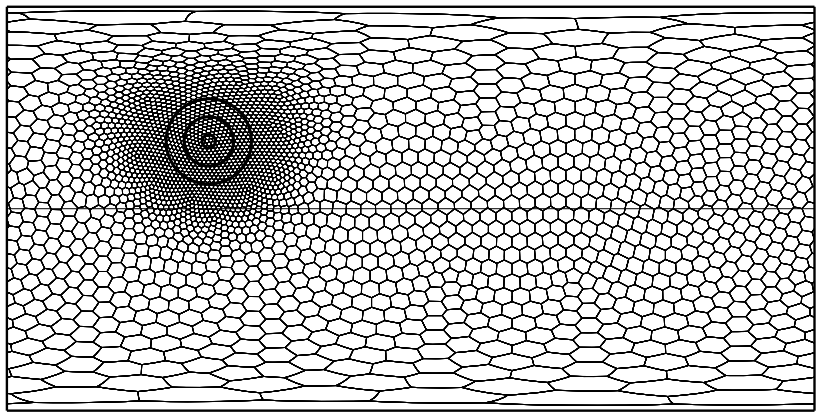
Iterative Mesh Optimisation using Lloyd's algorithm

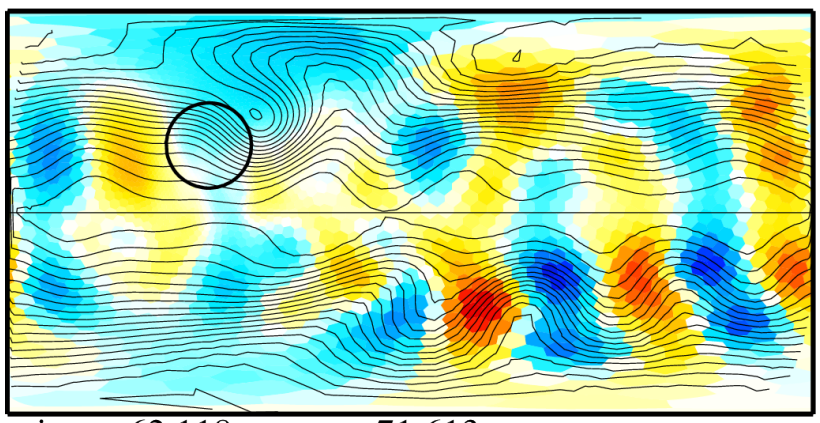


Shallow Water Flow Over a Mountain

Errors after 15 days using a static Lloyd algorithm mesh versus a static optimally transported mesh using the hexagonal C-grid [Ringler et al., 2010]

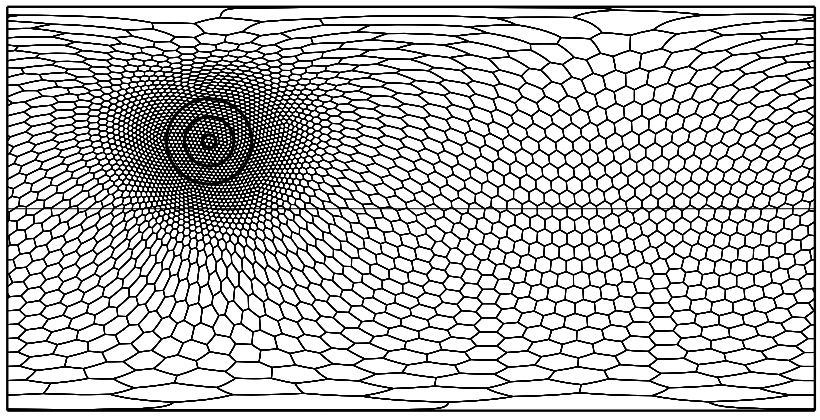
Lloyd

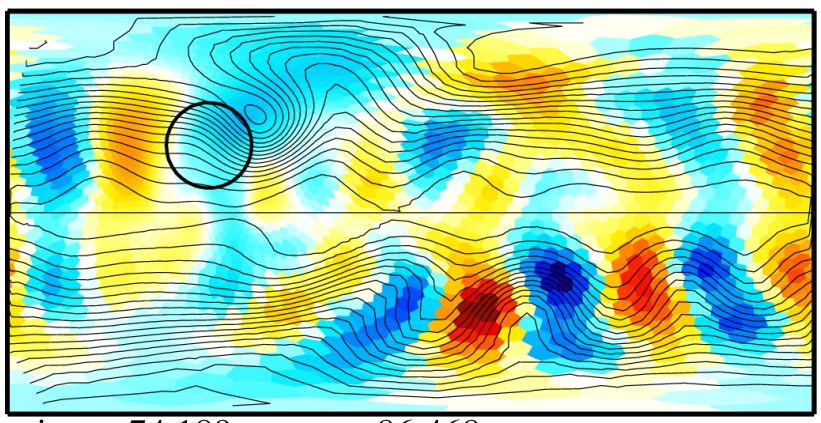




min = −62.118 max = 71.613

Optimally transported

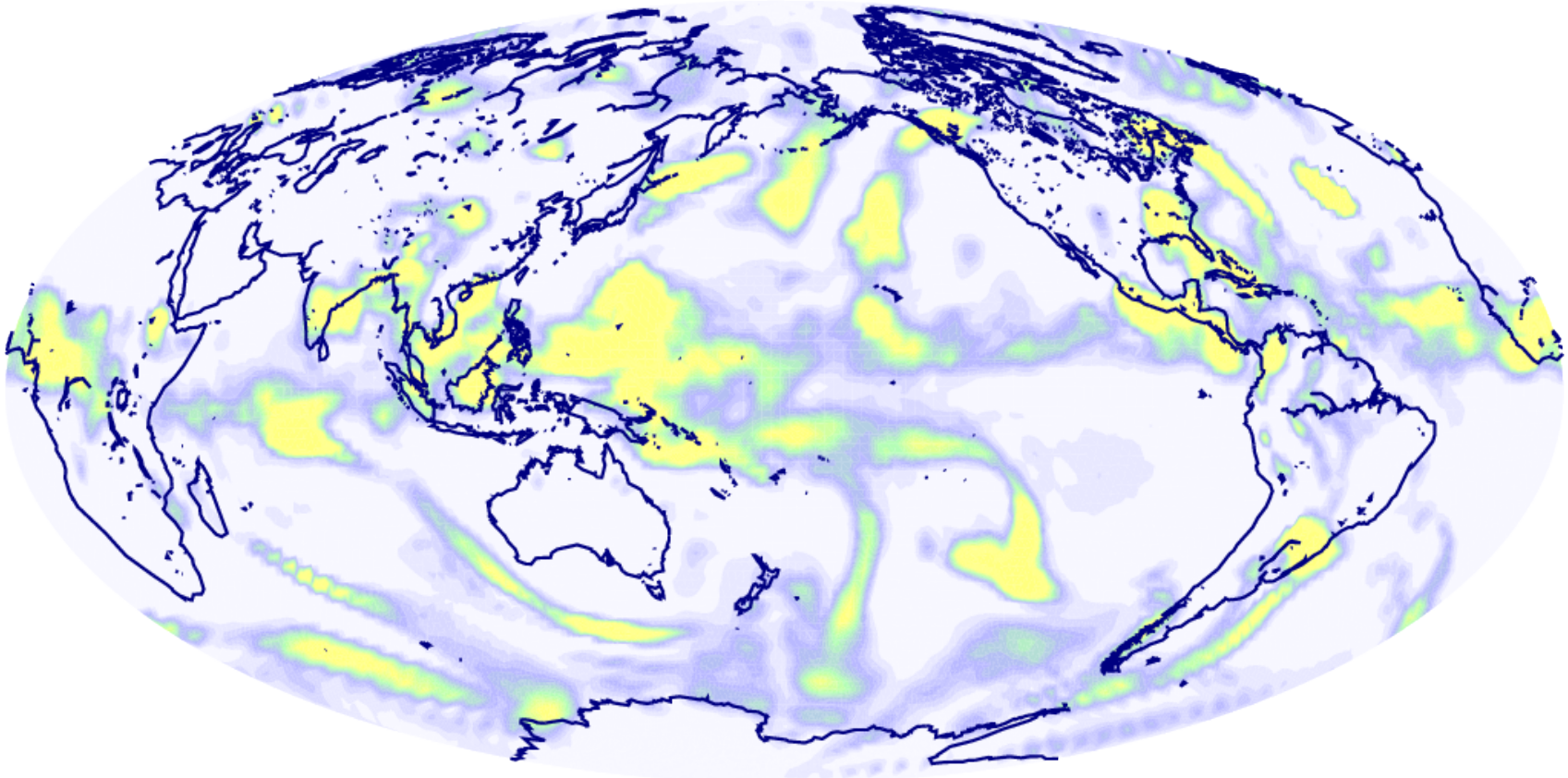




min = −74.189 max = 86.468

Precipitation as a Monitor Function

$$m = \frac{p + p_{\min}}{p_{\max} + p_{min}} \text{ where } p_{\min} = 10^{-5}\text{kgm}^{-2}\text{s}^{-1},\ p_{\max} = 8.73 \times 10^{-4}\text{kgm}^{-2}\text{s}^{-1}$$



Using daily average precipitation rate, 1-12 Oct 2012, from the NOAA-CIRES 20th Century Reanalysis version 2 (Compo et al, 2011, `http://www.esrl.noaa.gov/psd/data/gridded/data.20thC_ReanV2.html`)

See Weller, Browne, Budd, and Cullen [2016]

Conclusions so far ...

Monge-Ampére Equation on the Sphere

- Exponential maps used to create optimally transported meshes on the sphere
- New linearisation to create a semi-implicit algorithm
- Generates meshes similar to Lloyd's algorithm but faster

Conclusions so far ...

Monge-Ampére Equation on the Sphere

- Exponential maps used to create optimally transported meshes on the sphere
- New linearisation to create a semi-implicit algorithm
- Generates meshes similar to Lloyd's algorithm but faster
- Not fast enough ... too many iterations to find the optimally transported meshes

Numerical Solution Techniques for the Monge-Ampére Equation

- On the sphere we have used *fixed-point iterations*:

$$\alpha\nabla^2\phi^{(n+1)} = \alpha\nabla^2\phi^{(n)} - |I + H(\phi^n)| + \frac{\text{const}^n}{m\left(\mathbf{x}^n\right)}$$

Numerical Solution Techniques for the Monge-Ampére Equation

- On the sphere we have used *fixed-point iterations*:

$$\alpha\nabla^2\phi^{(n+1)} = \alpha\nabla^2\phi^{(n)} - |I + H(\phi^n)| + \frac{\text{const}^n}{m(\mathbf{x}^n)}$$

- Budd and Williams [2006] proposed solving a *Parabolic Monge-Ampére Equation*:

$$\frac{\partial\phi}{\partial t} - \gamma\frac{\partial\nabla^2\phi}{\partial t} = \left(|I + H(\phi)|\, m(\mathbf{x})\right)^{1/d}$$

  I will also describe

Numerical Solution Techniques for the Monge-Ampére Equation

- On the sphere we have used *fixed-point iterations*:

$$\alpha\nabla^2\phi^{(n+1)} = \alpha\nabla^2\phi^{(n)} - |I + H(\phi^n)| + \frac{\text{const}^n}{m(\mathbf{x}^n)}$$

- Budd and Williams [2006] proposed solving a *Parabolic Monge-Ampére Equation*:

$$\frac{\partial\phi}{\partial t} - \gamma\frac{\partial\nabla^2\phi}{\partial t} = \left(|I + H(\phi)|\, m(\mathbf{x})\right)^{1/d}$$

I will also describe

- *Fixed-point iterations with adaptive linearisation*
- A *Newton method*

In Euclidean Geometry (work on the sphere to follow)

Fixed-point Iterations with Adaptive Linearisation

- For the fixed-point iterations, we used $|I + H(\phi)| \approx 1 + \nabla^2 \phi$ which was derived by linearising about $\phi = 0$.

Fixed-point Iterations with Adaptive Linearisation

- For the fixed-point iterations, we used $|I + H(\phi)| \approx 1 + \nabla^2 \phi$ which was derived by linearising about $\phi = 0$.
- Instead linearise about the previous iteration

$$\phi^{n+1} \approx \phi^n + \phi'$$

Fixed-point Iterations with Adaptive Linearisation

- For the fixed-point iterations, we used $|I + H(\phi)| \approx 1 + \nabla^2 \phi$ which was derived by linearising about $\phi = 0$.
- Instead linearise about the previous iteration

$$\phi^{n+1} \approx \phi^n + \phi'$$

which gives: $$|I + H(\phi^{n+1})| \approx |I + H(\phi^n)| + \nabla \cdot A^n \nabla \phi'$$

where $A^n = \begin{pmatrix} 1 + \phi^n_{yy} & -\phi^n_{xy} \\ -\phi^n_{xy} & 1 + \phi^n_{xx} \end{pmatrix}$ in 2D Euclidean geometry.

Fixed-point Iterations with Adaptive Linearisation

- For the fixed-point iterations, we used $|I + H(\phi)| \approx 1 + \nabla^2 \phi$ which was derived by linearising about $\phi = 0$.
- Instead linearise about the previous iteration

$$\phi^{n+1} \approx \phi^n + \phi'$$

which gives: $$|I + H(\phi^{n+1})| \approx |I + H(\phi^n)| + \nabla \cdot A^n \nabla \phi'$$

where $A^n = \begin{pmatrix} 1 + \phi^n_{yy} & -\phi^n_{xy} \\ -\phi^n_{xy} & 1 + \phi^n_{xx} \end{pmatrix}$ in 2D Euclidean geometry.

- So instead we solve for ϕ' and the fixed-point iterations become:

$$\nabla \cdot A^n \nabla \phi' = -|I + H(\phi^n)| + \frac{\text{const}^n}{m(\mathbf{x}^n)}$$

Fixed-point Iterations with Adaptive Linearisation

- For the fixed-point iterations, we used $|I+H(\phi)| \approx 1+\nabla^2\phi$ which was derived by linearising about $\phi = 0$.
- Instead linearise about the previous iteration

$$\phi^{n+1} \approx \phi^n + \phi'$$

which gives: $$|I+H(\phi^{n+1})| \approx |I+H(\phi^n)| + \nabla \cdot A^n \nabla \phi'$$

where $A^n = \begin{pmatrix} 1+\phi^n_{yy} & -\phi^n_{xy} \\ -\phi^n_{xy} & 1+\phi^n_{xx} \end{pmatrix}$ in 2D Euclidean geometry.

- So instead we solve for ϕ' and the fixed-point iterations become:

$$\nabla \cdot A^n \nabla \phi' = -|I+H(\phi^n)| + \frac{\text{const}^n}{m(\mathbf{x}^n)}$$

- To ensure that this equation remains elliptic, we may need to modify A to force it to be definite (all positive eigenvalues):

$$B^n = A^n + \gamma I \quad \text{where} \quad \gamma = \begin{cases} 0 & \text{if } \min\sigma[A^n] > 0 \\ \varepsilon - \min\sigma[A^n] & \text{if } \min\sigma[A^n] \leq 0 \end{cases}$$

Fixed-point Iterations with Adaptive Linearisation

- For the fixed-point iterations, we used $|I+H(\phi)| \approx 1+\nabla^2\phi$ which was derived by linearising about $\phi = 0$.
- Instead linearise about the previous iteration

$$\phi^{n+1} \approx \phi^n + \phi'$$

which gives: $$|I+H(\phi^{n+1})| \approx |I+H(\phi^n)| + \nabla \cdot A^n \nabla \phi'$$

where $A^n = \begin{pmatrix} 1+\phi^n_{yy} & -\phi^n_{xy} \\ -\phi^n_{xy} & 1+\phi^n_{xx} \end{pmatrix}$ in 2D Euclidean geometry.

- So instead we solve for ϕ' and the fixed-point iterations become:

$$\nabla \cdot A^n \nabla \phi' = -|I+H(\phi^n)| + \frac{\text{const}^n}{m(\mathbf{x}^n)}$$

- To ensure that this equation remains elliptic, we may need to modify A to force it to be definite (all positive eigenvalues):

$$B^n = A^n + \gamma I \quad \text{where} \quad \gamma = \begin{cases} 0 & \text{if } \min\sigma[A^n] > 0 \\ \varepsilon - \min\sigma[A^n] & \text{if } \min\sigma[A^n] \leq 0 \end{cases}$$

- This is nearly a Newton solver ...

Newton Solver

- The adaptive fixed-point iterations are:

$$\nabla \cdot A^n \nabla \phi' = -|I + H(\phi^n)| + \frac{\text{const}}{m(\mathbf{x})} \tag{2}$$

Newton Solver

- The adaptive fixed-point iterations are:

$$\nabla \cdot A^n \nabla \phi' = -|I + H(\phi^n)| + \frac{\text{const}}{m(\mathbf{x})} \tag{2}$$

- To make this into a Newton solver we also need to linearise the $\frac{c}{m(\mathbf{x})}$ term:

$$\begin{aligned} \frac{c}{m}\left(\mathbf{x}^{n+1}\right) &\approx \frac{c}{m}\left(\mathbf{x}^n\right) + \nabla_x\left(\frac{c}{m}\left(\mathbf{x}^n\right)\right) \cdot \left(\mathbf{x}^{n+1} - \mathbf{x}^n\right) \\ &= \frac{c}{m}\left(\mathbf{x}^n\right) + \nabla_x\left(\frac{c}{m}\left(\mathbf{x}^n\right)\right) \cdot \nabla \phi' \end{aligned}$$

Newton Solver

- The adaptive fixed-point iterations are:

$$\nabla \cdot A^n \nabla \phi' = -|I + H(\phi^n)| + \frac{\text{const}}{m(\mathbf{x})} \tag{2}$$

- To make this into a Newton solver we also need to linearise the $\frac{c}{m(\mathbf{x})}$ term:

$$\begin{aligned} \frac{c}{m}\left(\mathbf{x}^{n+1}\right) &\approx \frac{c}{m}\left(\mathbf{x}^n\right) + \nabla_x\left(\frac{c}{m}\left(\mathbf{x}^n\right)\right) \cdot \left(\mathbf{x}^{n+1} - \mathbf{x}^n\right) \\ &= \frac{c}{m}\left(\mathbf{x}^n\right) + \nabla_x\left(\frac{c}{m}\left(\mathbf{x}^n\right)\right) \cdot \nabla \phi' \end{aligned}$$

- Substituting into eqn(2) gives

$$\nabla \cdot A^n \nabla \phi' = -|I + H(\phi^n)| + \frac{c}{m}\left(\mathbf{x}^n\right) + \nabla_x\left(\frac{c}{m}\left(\mathbf{x}^n\right)\right) \cdot \nabla \phi' \tag{3}$$

which is like an advection-diffusion equation for ϕ' with tensorial diffusion A^n and advecting velocity $\nabla_x\left(\frac{c}{m}\left(\mathbf{x}^n\right)\right)$.

Newton Solver

- The adaptive fixed-point iterations are:

$$\nabla \cdot A^n \nabla \phi' = -|I + H(\phi^n)| + \frac{\text{const}}{m(\mathbf{x})} \tag{2}$$

- To make this into a Newton solver we also need to linearise the $\frac{c}{m(\mathbf{x})}$ term:

$$\begin{aligned} \frac{c}{m}\left(\mathbf{x}^{n+1}\right) &\approx \frac{c}{m}\left(\mathbf{x}^n\right) + \nabla_x\left(\frac{c}{m}\left(\mathbf{x}^n\right)\right) \cdot \left(\mathbf{x}^{n+1} - \mathbf{x}^n\right) \\ &= \frac{c}{m}\left(\mathbf{x}^n\right) + \nabla_x\left(\frac{c}{m}\left(\mathbf{x}^n\right)\right) \cdot \nabla \phi' \end{aligned}$$

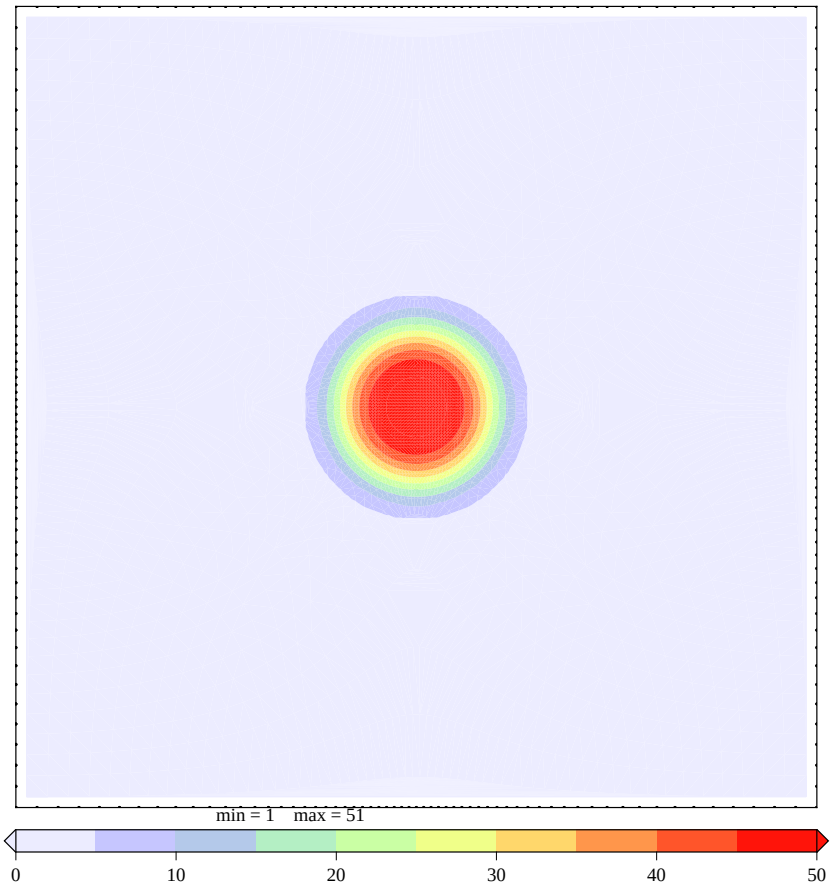
- Substituting into eqn(2) gives

$$\nabla \cdot A^n \nabla \phi' = -|I + H(\phi^n)| + \frac{c}{m}\left(\mathbf{x}^n\right) + \nabla_x\left(\frac{c}{m}\left(\mathbf{x}^n\right)\right) \cdot \nabla \phi' \tag{3}$$

  which is like an advection-diffusion equation for ϕ' with tensorial diffusion A^n and advecting velocity $\nabla_x\left(\frac{c}{m}\left(\mathbf{x}^n\right)\right)$.
- Eqn (3) is a linear equation for ϕ' which is straightforward to solve (after spatial discretisation)

First Results of Newton Method

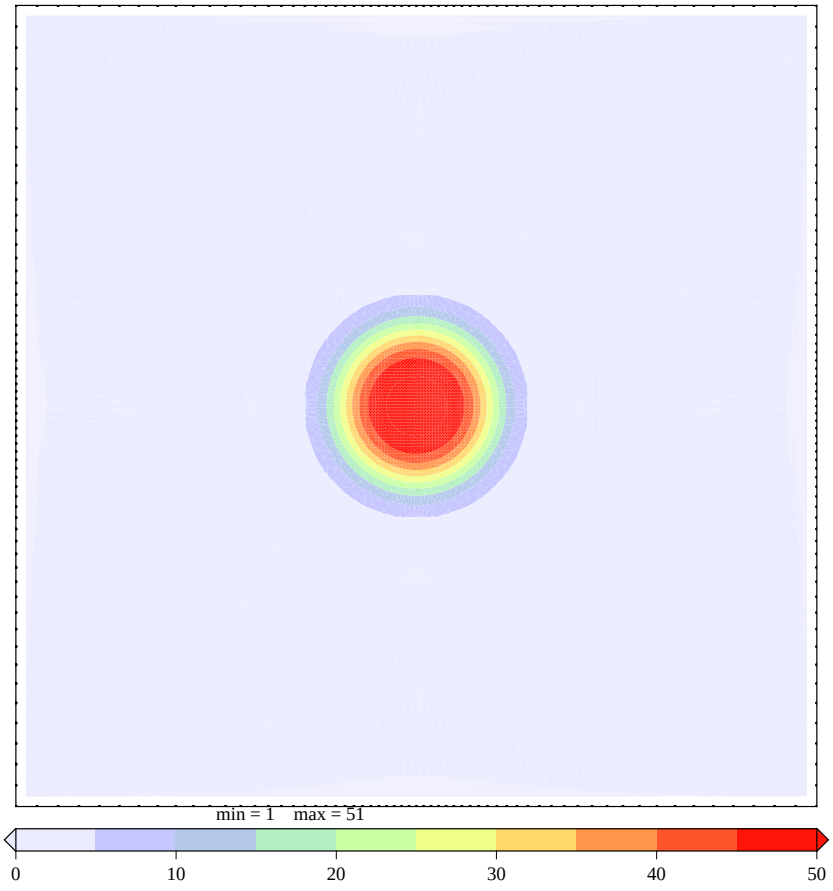
Monitor function, m



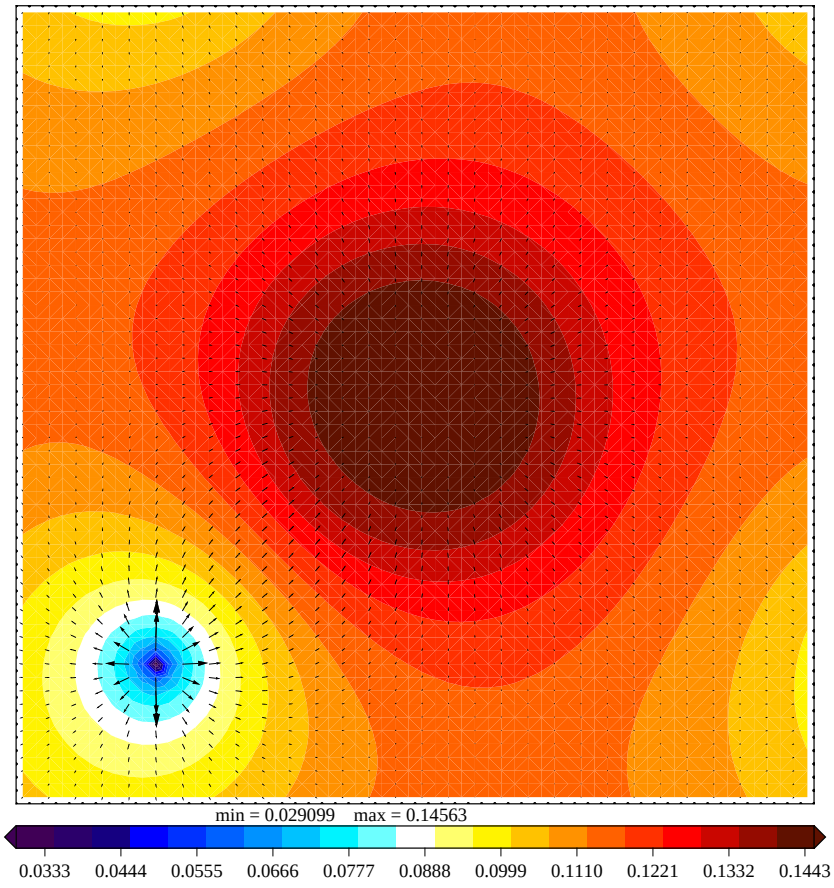
ϕ and $\nabla\phi$ after 1 iteration

First Results of Newton Method

Monitor function, m

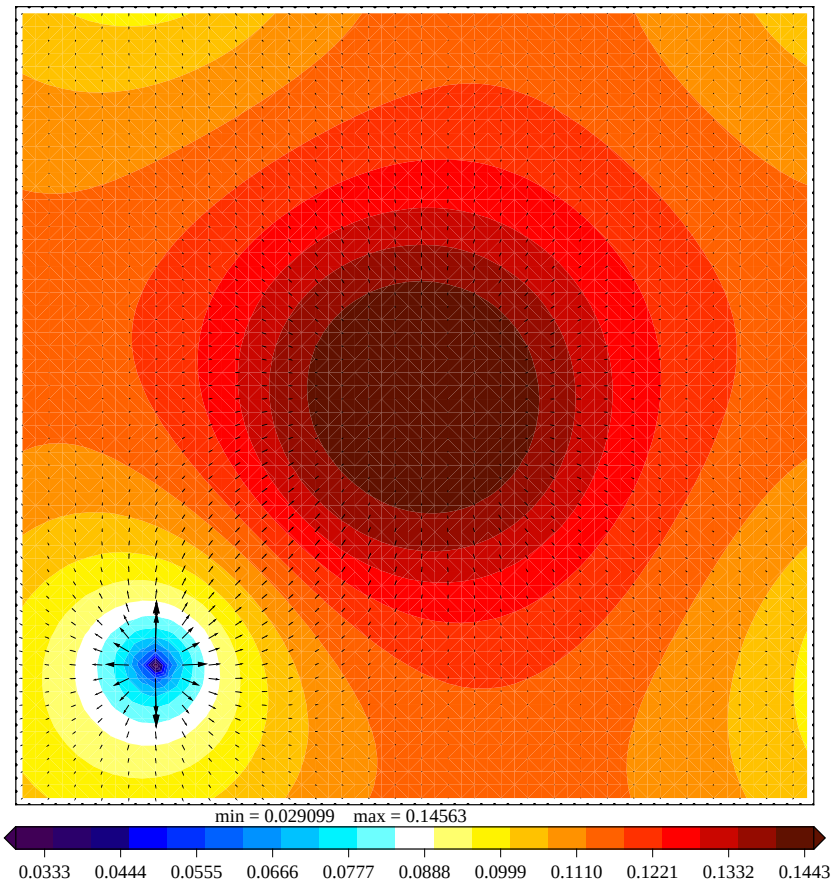


ϕ and $\nabla\phi$ after 1 iteration

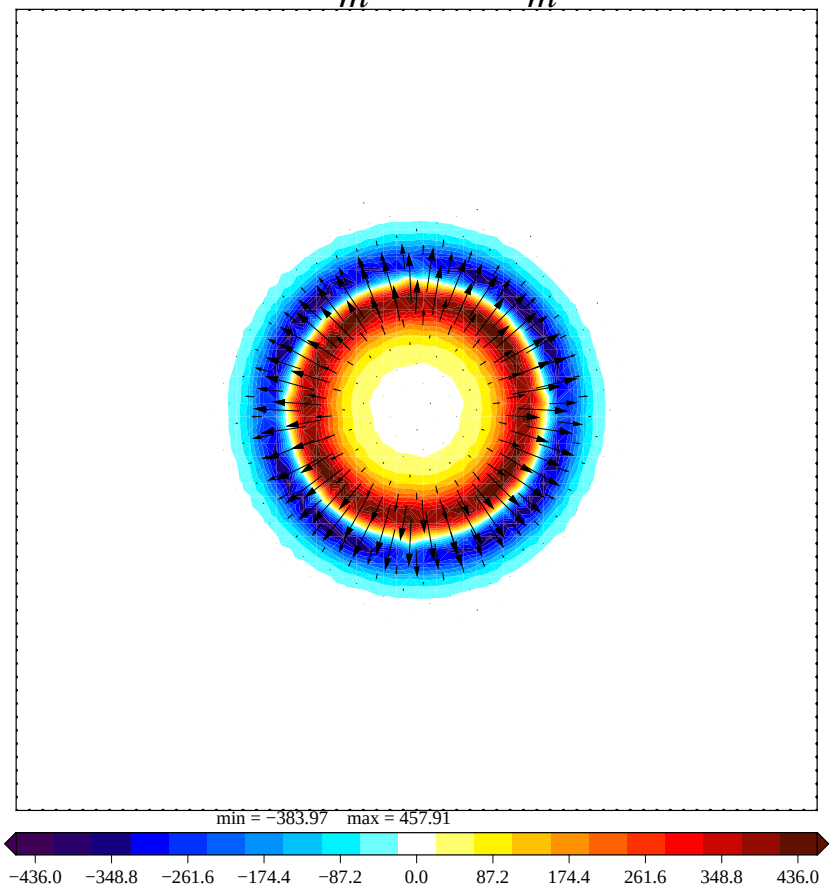


First Results of Newton Method

ϕ and $\nabla\phi$



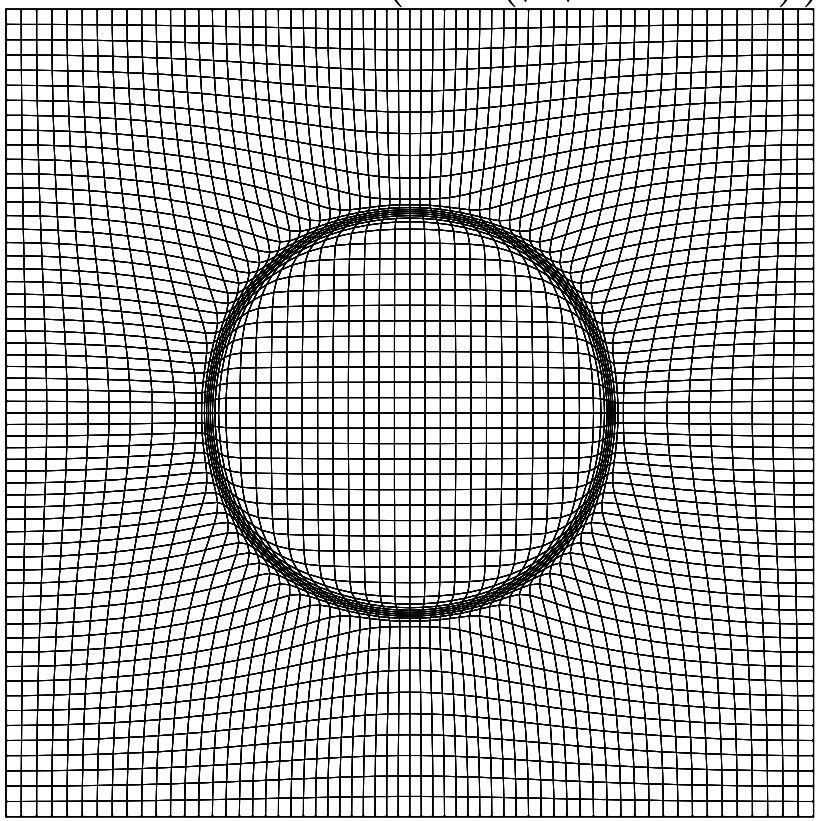
$\nabla^2 \frac{c}{m}$ and $\nabla \frac{c}{m}$



Results of Test Cases

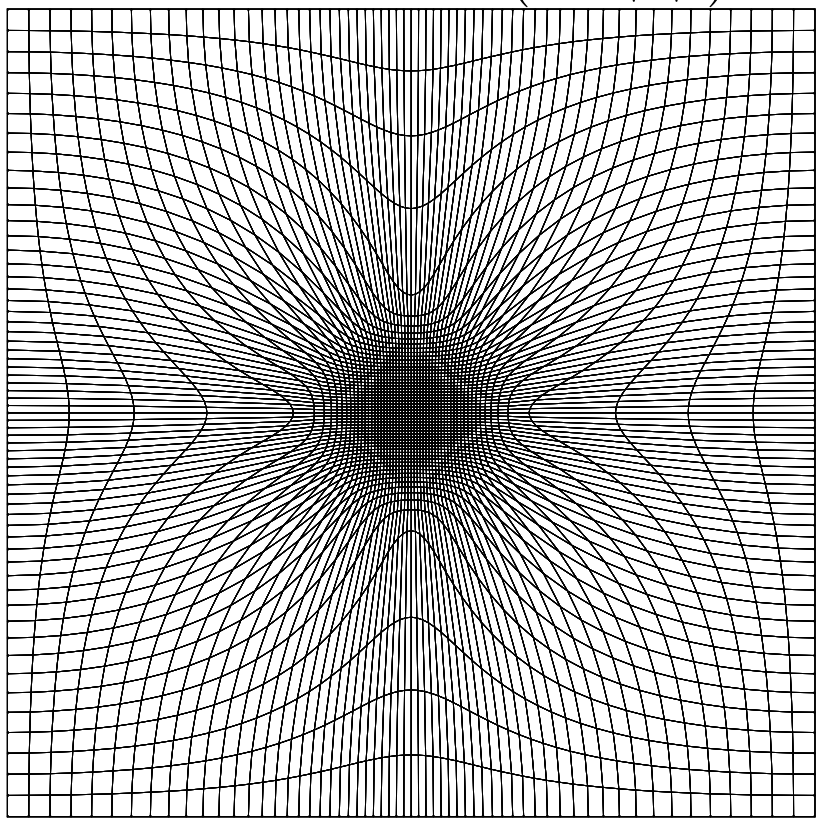
All techniques converge to the same solutions

$m = 1 + 10\ \mathrm{sech}^2\left(200\left(|\mathbf{x}|^2 - 0.25^2\right)\right)$



Ring mesh

$m = 50 + 10\ \mathrm{sech}^2\left(100\ |\mathbf{x}|^2\right)$

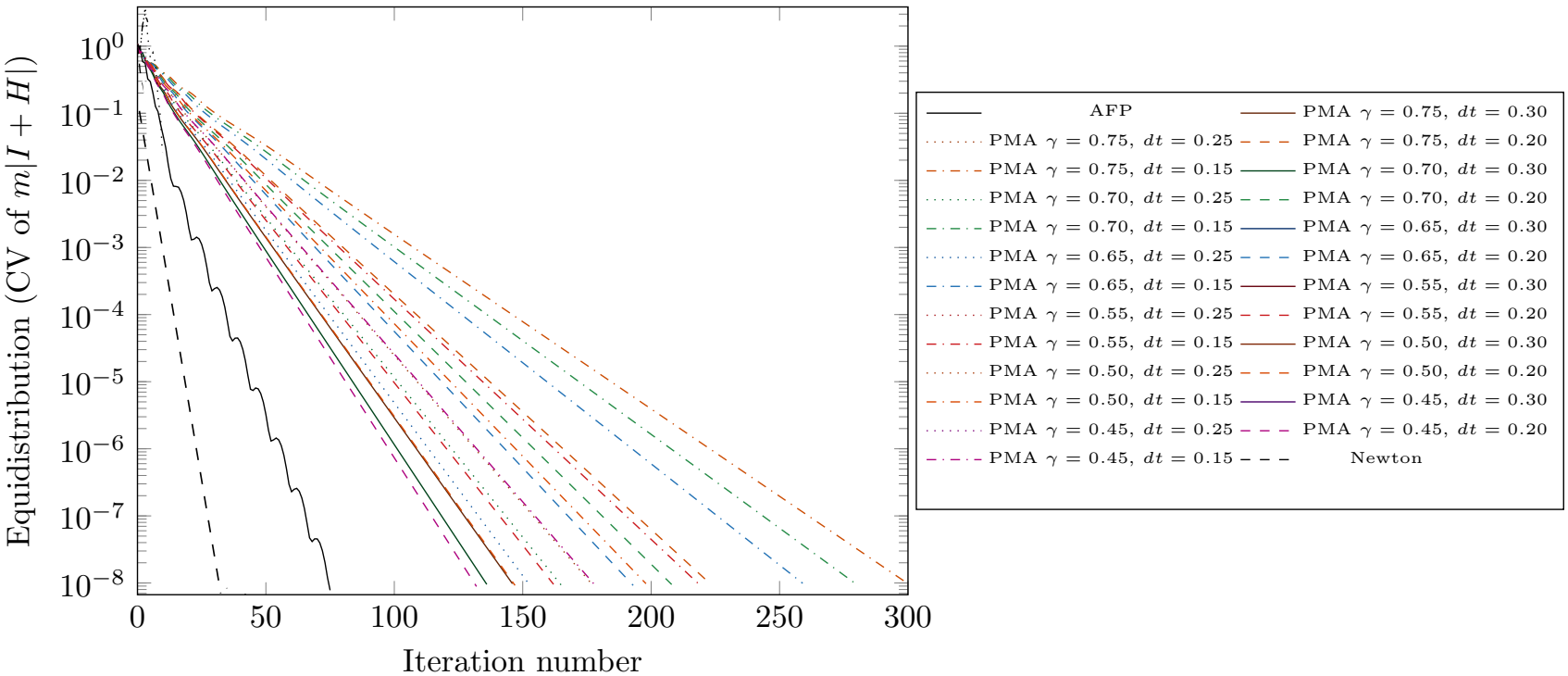


Bell mesh

But how fast do they converge ...

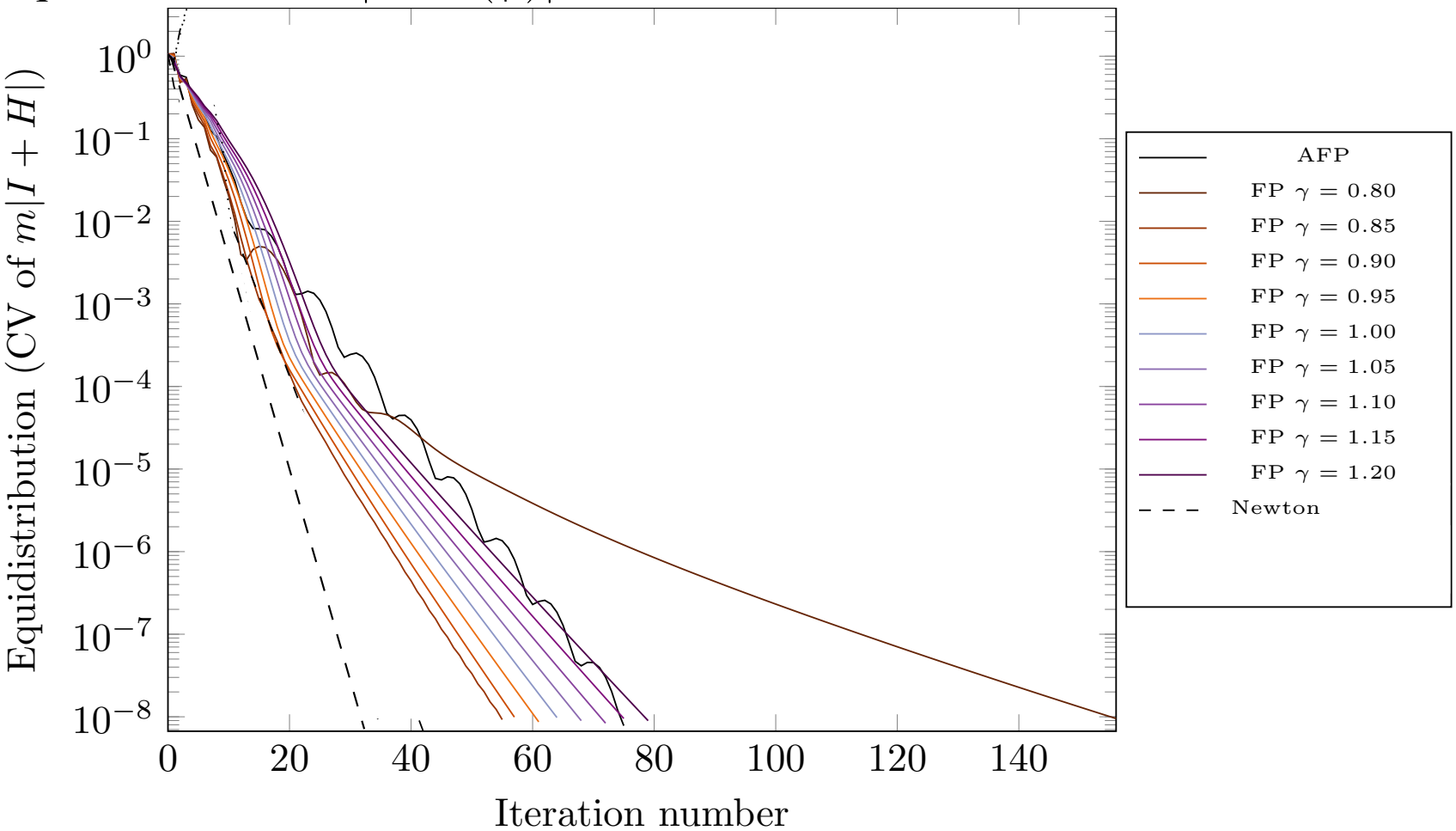
Results for the Ring Mesh

Number of Iterations to Convergence (measured by variance of equidistribution, $m|I+H(\phi)|$)



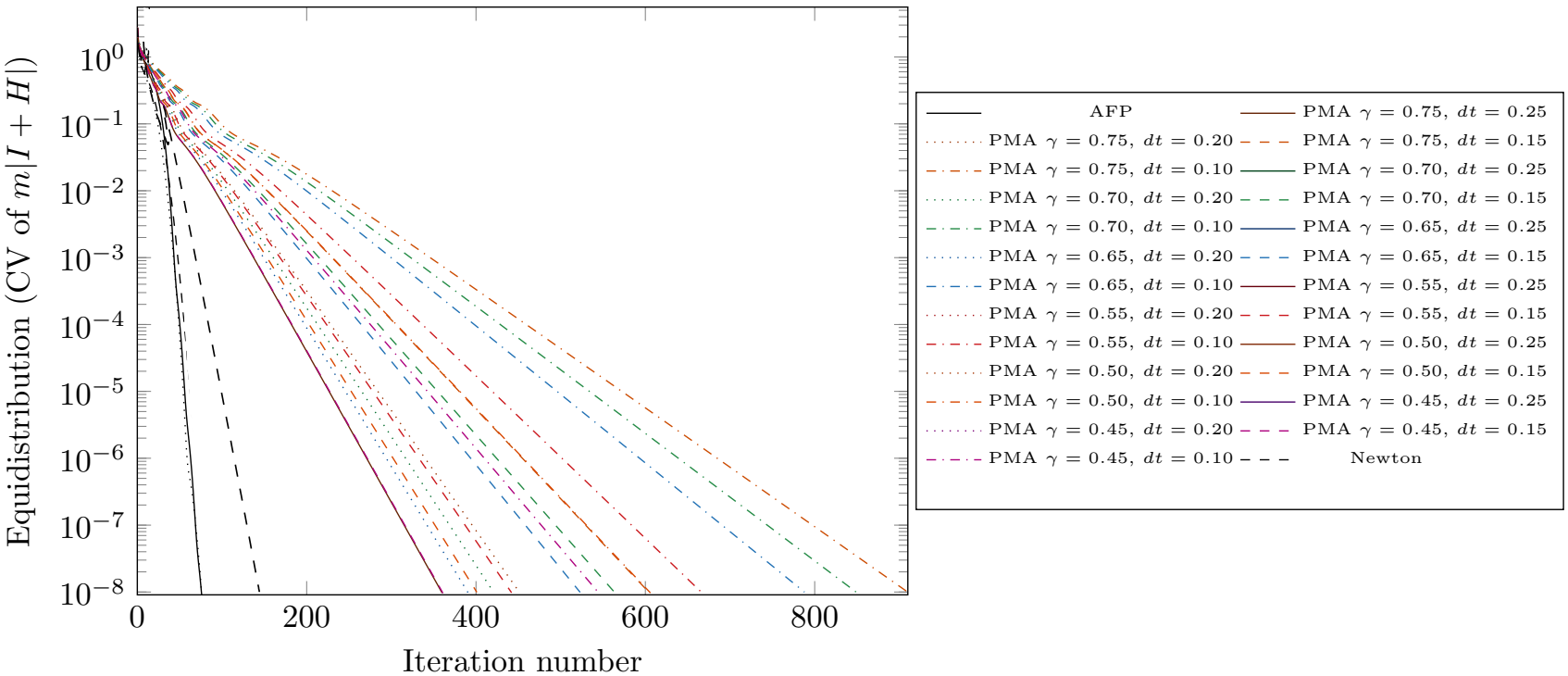
Results for the Ring Mesh

Number of Iterations to Convergence (measured by variance of equidistribution, $m|I+H(\phi)|$)



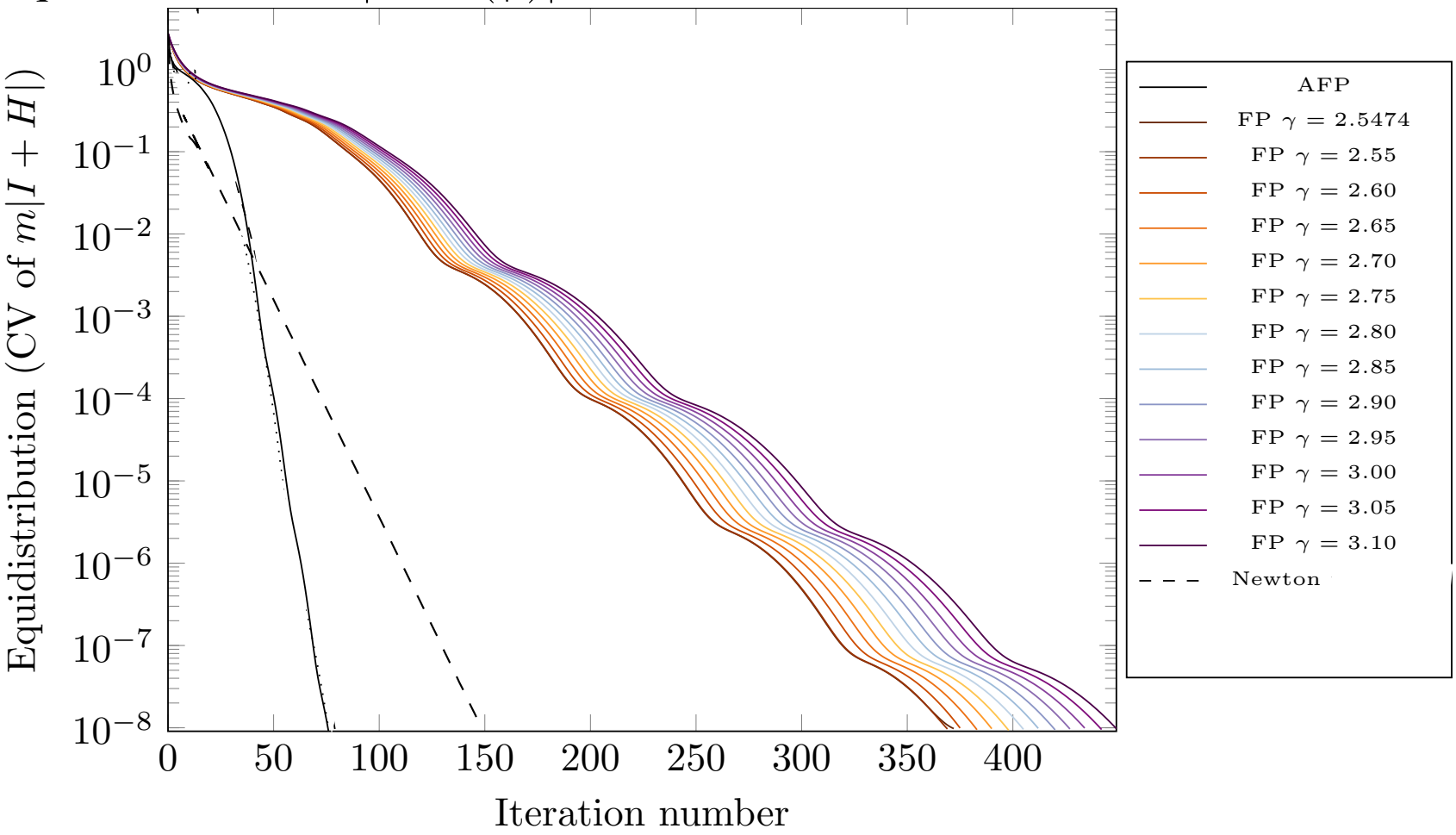
Results for the Bell Mesh

Number of Iterations to Convergence (measured by variance of equidistribution, $m|I+H(\phi)|$)

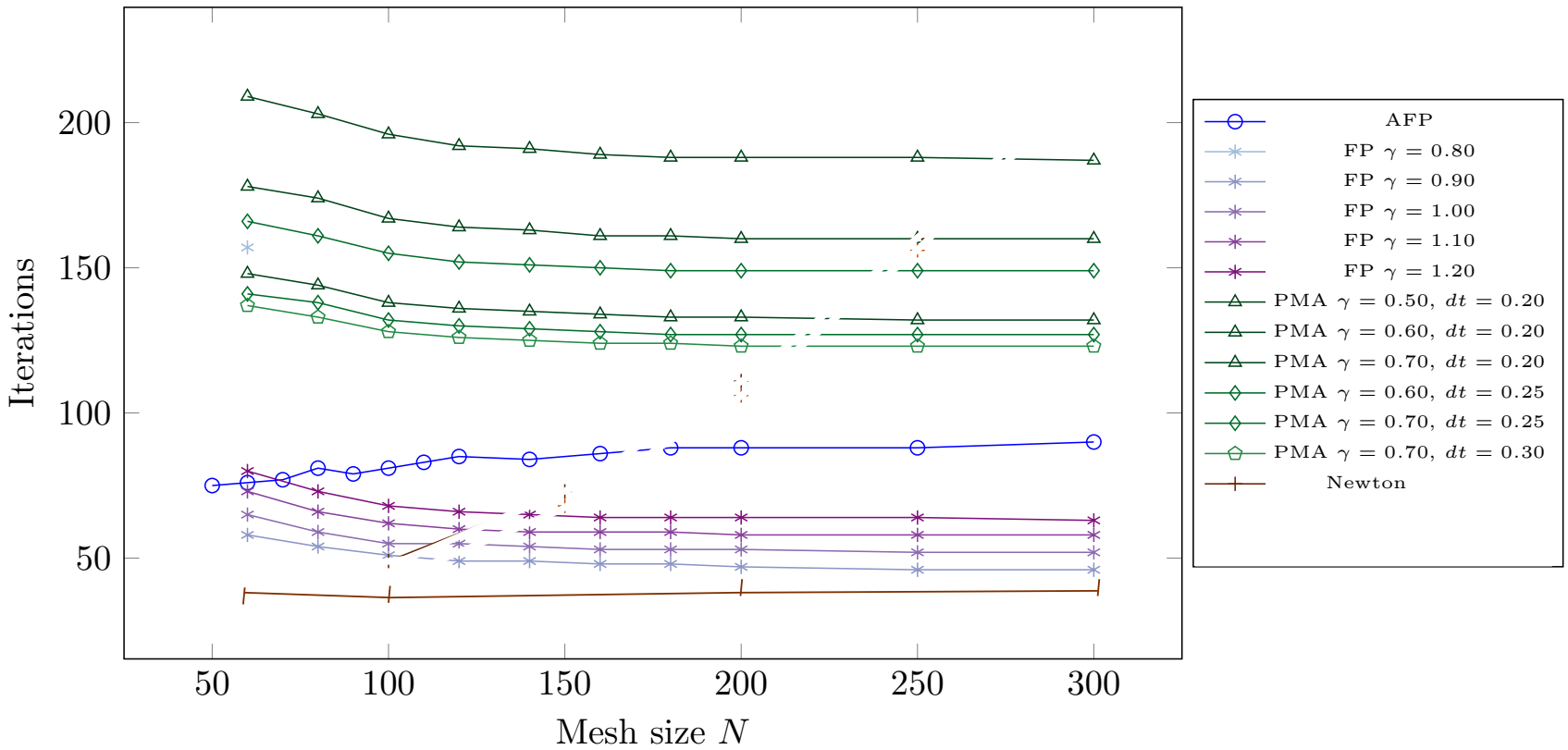


Results for the Bell Mesh

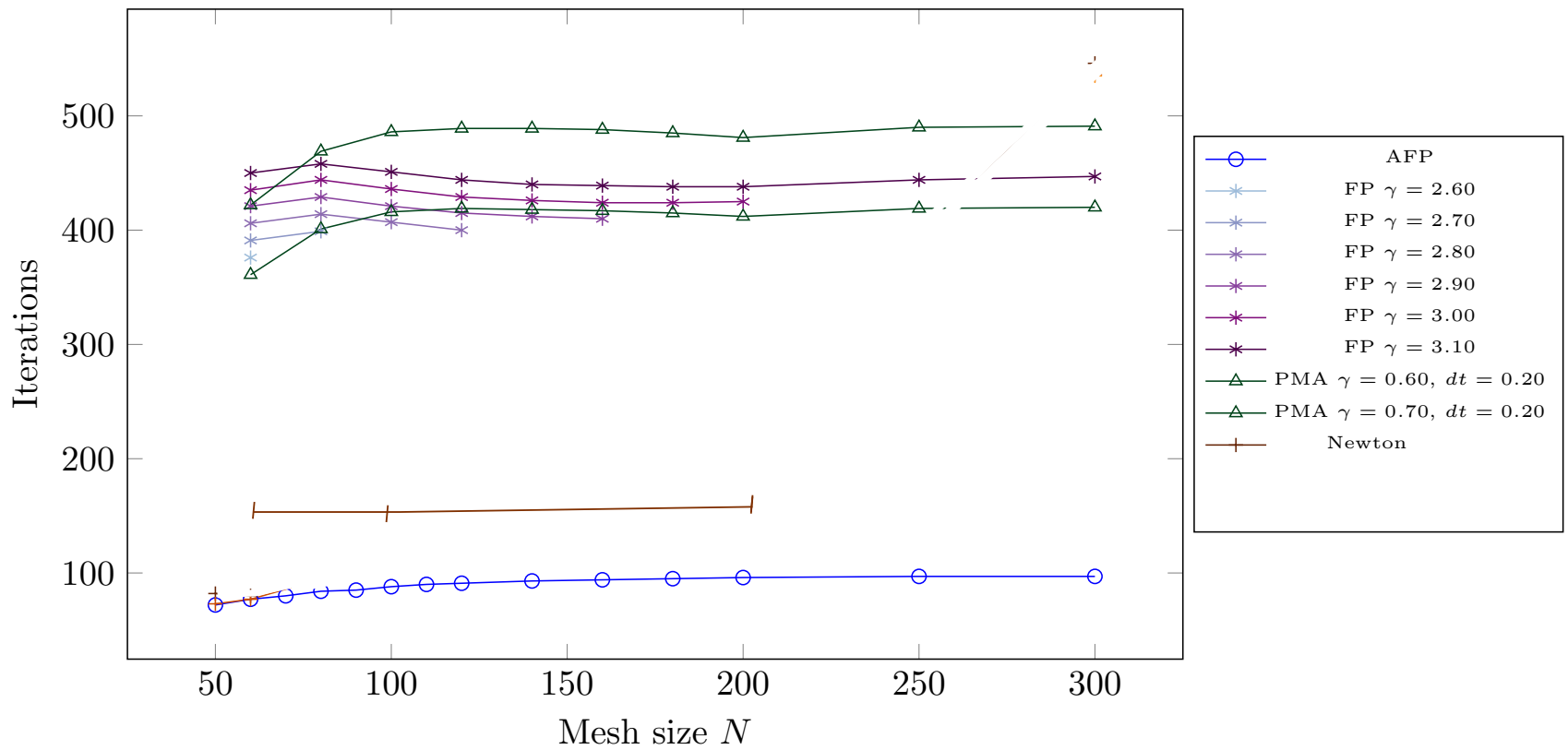
Number of Iterations to Convergence (measured by variance of equidistribution, $m|I+H(\phi)|$)



Scaling With Problem Size for the Ring



Scaling With Problem Size for the Bell



Conclusions

Monge-Ampére Equation on the Sphere

- Exponential maps used to create optimally transported meshes on the sphere
- New linearisation to create a semi-implicit algorithm
- Generates meshes similar to Lloyd's algorithm but faster

Iterative Algorithms

Conclusions

Monge-Ampére Equation on the Sphere

- Exponential maps used to create optimally transported meshes on the sphere
- New linearisation to create a semi-implicit algorithm
- Generates meshes similar to Lloyd's algorithm but faster

Iterative Algorithms

- Iterations with adaptive linearisation are:
  - Faster than solution of Parabolic Monge-Ampére equation
  - Faster than fixed point iterations linearising about zero
  - Not dependent on any free parameters
  - Cost linearly dependent on problem size

Conclusions

Monge-Ampére Equation on the Sphere

- Exponential maps used to create optimally transported meshes on the sphere
- New linearisation to create a semi-implicit algorithm
- Generates meshes similar to Lloyd's algorithm but faster

Iterative Algorithms

- Iterations with adaptive linearisation are:
  - Faster than solution of Parabolic Monge-Ampére equation
  - Faster than fixed point iterations linearising about zero
  - Not dependent on any free parameters
  - Cost linearly dependent on problem size
- Newton method is:
  - Faster and slower than adaptive linearisation
  - Cost linearly dependent on problem size

C.J. Budd and J.F. Williams. Parabolic Monge-Ampère methods for blow-up problems in several spatial dimensions. *J. Phys. A*, 39(19):5425–5444, 12 May 2006.

C.J. Budd, R.D. Russell, and E. Walsh. The geometry of r-adaptive meshes generated using optimal transport methods. *J. Comput. Phys.*, 282: 113–137, 2015.

T.D. Ringler, J. Thuburn, W.C. Skamarock, and J.B. Klemp. A unified approach to energy conservation and potential vorticity dynamics for arbitrarily-structured C-grids. *J. Comput. Phys.*, 229:3065–3090, 2010.

H. Weller, P. Browne, C. Budd, and M. Cullen. Mesh adaptation on the sphere using optimal transport and the numerical solution of a Monge-Ampère type equation. *J. Comput. Phys.*, 308:102–123, 2016.