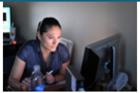
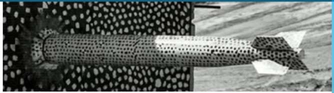
-2945C

National Laboratories

Optimization based solvers for the Monge-Ampére equation with applications to mesh adaptivity

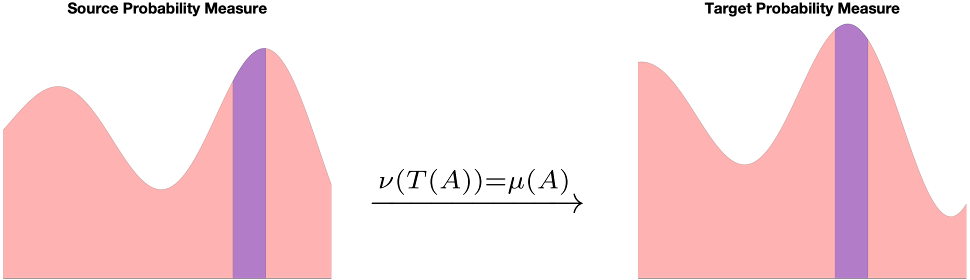
Kelsey DiPietro, Denis Ridzal

East Coast Optimization Meeting

Virtual Conference

LDRD
Laboratory Directed Research

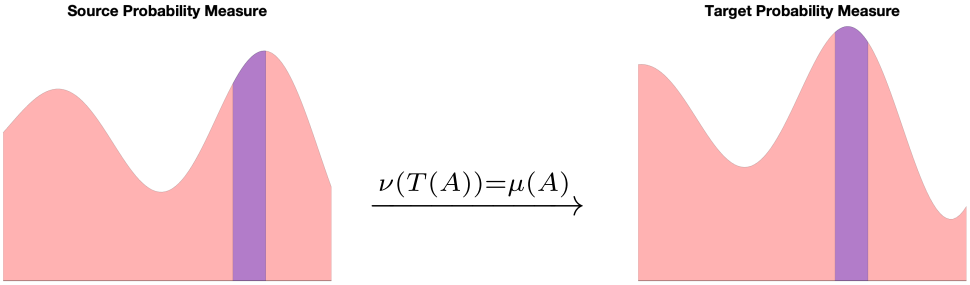
Crash Course in Optimal Transport



Density $\mu \in \mathbf{X}$

Density $\nu \in \mathbf{Y}$

Crash Course in Optimal Transport



Density $\mu \in \mathbf{X}$ Density $\nu \in \mathbf{Y}$

Take away: Obtain the map that transport mass from the source to target with minimal cost.

Relating Optimal Transport to the Monge-Ampère equation

Recall the optimal transport map $\phi(\mu_0) = \mu_1$, find the solution ϕ to the optimal transport problem:

$$\min_{\phi(\mathbf{x})} \int_\Omega \mu_0 |\phi(\mathbf{x}) - \mathbf{x}|^2 dx,$$

$$\text{such that } c(\phi(\mathbf{x})) = \det(\nabla\phi(\mathbf{x}))\mu_1(\phi(\mathbf{x})) - \mu_0(\mathbf{x}) = 0,$$

where $\mu_0 \in \mathbf{X}, \mu_1 \in \mathbf{Y}$ are the source and target densities.

Relating Optimal Transport to the Monge-Ampère equation

Recall the optimal transport map $\phi(\mu_0) = \mu_1$, find the solution ϕ to the optimal transport problem:

$$\min_{\phi(\mathbf{x})} \int_\Omega \mu_0 |\phi(\mathbf{x}) - \mathbf{x}|^2 dx,$$

$$\text{such that } c(\phi(\mathbf{x})) = \det(\nabla\phi(\mathbf{x}))\mu_1(\phi(\mathbf{x})) - \mu_0(\mathbf{x}) = 0,$$

where $\mu_0 \in \mathbf{X}, \mu_1 \in \mathbf{Y}$ are the source and target densities.

Assumption that the map ϕ is the gradient of a convex function u, $\phi = \nabla u$

Recover the Monge-Ampère equation with a transport boundary condition:

$$\det(H(u)) = u_{xx}u_{yy} - u_{xy}^2 = \frac{\mu_0}{\mu_1}, \quad u \in \mathbf{X}, \quad \nabla u(\mathbf{X}) = \mathbf{Y}, \quad u \in \partial\mathbf{X}.$$

Relating Optimal Transport to the Monge-Ampère equation

Recall the optimal transport map $\phi(\mu_0) = \mu_1$, find the solution ϕ to the optimal transport problem:

$$\min_{\phi(\mathbf{x})} \int_\Omega \mu_0 |\phi(\mathbf{x}) - \mathbf{x}|^2 dx,$$

$$\text{such that } c(\phi(\mathbf{x})) = \det(\nabla\phi(\mathbf{x}))\mu_1(\phi(\mathbf{x})) - \mu_0(\mathbf{x}) = 0,$$

where $\mu_0 \in \mathbf{X}, \mu_1 \in \mathbf{Y}$ are the source and target densities.

Assumption that the map ϕ is the gradient of a convex function u, $\phi = \nabla u$

Recover the Monge-Ampère equation with a transport boundary condition:

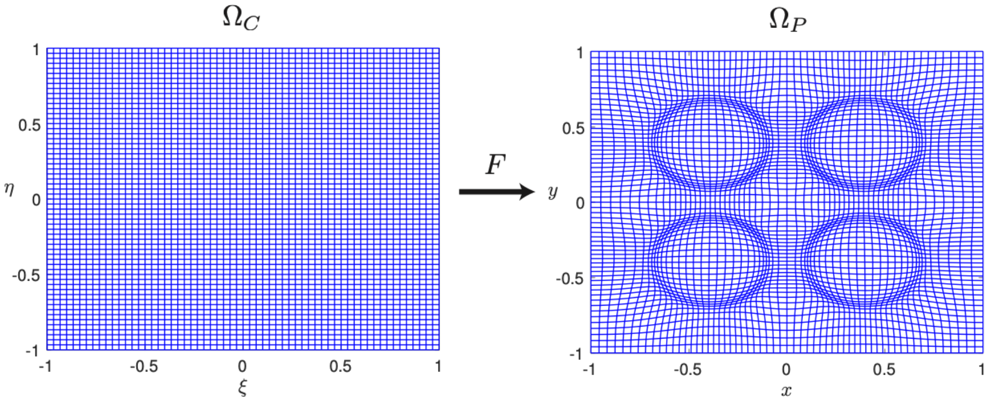
$$\det(H(u)) = u_{xx}u_{yy} - u_{xy}^2 = \frac{\mu_0}{\mu_1}, \quad u \in \mathbf{X}, \quad \nabla u(\mathbf{X}) = \mathbf{Y}, \quad u \in \partial\mathbf{X}.$$

We focus solving the Monge-Ampère equation to generate a **dynamic, optimally transported adaptive mesh.**

Links to Mesh Adaptation

If u is a convex solution to the Monge-Ampere equation, then $(x, y) = \nabla u$ **gives an adaptive mesh in the physical space.**

Mesh adaptation is determined through a monitor function based on the solution to PDE being solved.



Improving Numerical Methods for the MAE

- Desire **efficient and robust** finite element solvers for the Monge-Ampère equation

Improving Numerical Methods for the MAE

- Desire **efficient and robust** finite element solvers for the Monge-Ampère equation
  - The faster the solution the easier it is to integrate r-adapting meshing into a toolbox.
- Want a solver interface that fits well into existing Sandia architectures.
  - Use the Trilinos package Intrepid for the finite element discretization.
  - Utilize the Intrelab interface to run real time examples using Matlab.

Improving Numerical Methods for the MAE

- Desire **efficient and robust** finite element solvers for the Monge-Ampère equation
  - The faster the solution the easier it is to integrate r-adapting meshing into a toolbox.
- Want a solver interface that fits well into existing Sandia architectures.
  - Use the Trilinos package Intrepid for the finite element discretization.
  - Utilize the Intrelab interface to run real time examples using Matlab.
- Need **low-order approximations** in order to exploit parallel solution methods.

Mixed Finite Element Method for the MAE

Recall the MAE:

$$\det(D^2 u) = f \quad \text{in}\, \Omega, \quad \nabla u(\mathbf{X}) = \mathbf{Y} \quad \text{on}\, \partial\Omega.$$

Mixed Finite Element Method for the MAE

Recall the MAE:

$$\det(D^2 u) = f \quad \text{in}\, \Omega, \quad \nabla u(\mathbf{X}) = \mathbf{Y} \quad \text{on}\, \partial\Omega.$$

Boundary condition is approximated with a signed distance function. Discretize using a mixed finite element method.

$$\langle \mathbf{H}[U], \mathbf{\Phi} \rangle = -\int_{\Omega} \nabla U \otimes \nabla \mathbf{\Phi} + \int_{\partial\Omega} \nabla \otimes \mathbf{n}\mathbf{\Phi}, \quad \forall \mathbf{\Phi} \in \mathbb{V},$$

$$\langle \mathbf{F}(\mathbf{H}[U]), \mathbf{\Phi} \rangle = \langle f, \mathbf{\Phi} \rangle, \quad \forall \mathbf{\Phi} \in \mathbb{V}.$$

The discretization imposes an $\mathcal{O}(h)$ error near the boundary, uniqueness needs to be imposed with a mean value constraint.

Mixed Finite Element Method for the MAE

Recall the MAE:

$$\det(D^2u) = f \quad \text{in}\, \Omega, \quad \nabla u(\mathbf{X}) = \mathbf{Y} \quad \text{on}\, \partial\Omega.$$

Boundary condition is approximated with a signed distance function. Discretize using a mixed finite element method.

$$\langle \mathbf{H}[U], \mathbf{\Phi} \rangle = -\int_{\Omega} \nabla U \otimes \nabla \mathbf{\Phi} + \int_{\partial\Omega} \nabla \otimes \mathbf{n}\mathbf{\Phi}, \quad \forall \mathbf{\Phi} \in \mathbb{V},$$
$$\langle \mathbf{F}(\mathbf{H}[U]), \mathbf{\Phi} \rangle = \langle f, \mathbf{\Phi} \rangle, \quad \forall \mathbf{\Phi} \in \mathbb{V}.$$

The discretization imposes an $\mathcal{O}(h)$ error near the boundary, uniqueness needs to be imposed with a mean value constraint.

$$\mathbf{E} = \begin{bmatrix} \mathbf{M} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{C}_{1,1} & 0 \\ \mathbf{0} & \mathbf{M} & \mathbf{0} & \mathbf{0} & -\mathbf{C}_{1,2} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{M} & \mathbf{0} & -\mathbf{C}_{2,1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{M} & -\mathbf{C}_{2,2} & \mathbf{0} \\ \mathbf{B}^{1,1} & \mathbf{B}^{1,2} & \mathbf{B}^{2,1} & \mathbf{B}^{2,2} & \mathbf{0} & \langle 1, \mathbf{\Phi}^T \rangle \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \langle \mathbf{\Phi}, 1 \rangle & 0 \end{bmatrix}, \mathbf{v} = \begin{pmatrix} \mathbf{h}_{1,1} \\ \mathbf{h}_{1,2} \\ \mathbf{h}_{2,1} \\ \mathbf{h}_{2,2} \\ \mathbf{u} \\ \lambda \end{pmatrix}, \quad \mathbf{b} = \begin{pmatrix} \langle \Psi^*(\mathbf{n}), \mathbf{\Phi}^T \rangle \\ \mathbf{0} \\ \mathbf{0} \\ \langle \Psi^*(\mathbf{n}), \mathbf{\Phi}^T \rangle \\ \mathbf{f} \\ 0 \end{pmatrix},$$

$C_{1,1} = -\langle \partial_1 \mathbf{\Phi}, \partial_1 \mathbf{\Phi}^T \rangle + \langle \mathbf{\Phi}\mathbf{n}_2, \partial_2 \mathbf{\Phi}^T \rangle_{\partial\Omega} \in \mathbb{R}^{N \times N}$, $C_{2,2} = -\langle \partial_2 \mathbf{\Phi}, \partial_2 \mathbf{\Phi}^T \rangle + \langle \mathbf{\Phi}\mathbf{n}_1, \partial_1 \mathbf{\Phi}^T \rangle_{\partial\Omega} \in \mathbb{R}^{N \times N}$

Can the Mixed Method be used with $\mathbb{P}^1$ Elements?

- Much of the literature claims the above methods cannot converge for $\mathbb{P}^1$ elements.

Can the Mixed Method be used with $\mathbb{P}^1$ Elements?

- Much of the literature claims the above methods cannot converge for $\mathbb{P}^1$ elements.
- Show that Newton's and damped Newton's methods cannot converge unless:
  1. It is discretized with higher order elements (such as $\mathbb{P}^2$).[†]

Can the Mixed Method be used with $\mathbb{P}^1$ Elements?

- Much of the literature claims the above methods cannot converge for $\mathbb{P}^1$ elements.
- Show that Newton's and damped Newton's methods cannot converge unless:
  1. It is discretized with higher order elements (such as $\mathbb{P}^2$).[†]
  2. Additional gradient variables are used.[††]

Can the Mixed Method be used with $\mathbb{P}^1$ Elements?

- Much of the literature claims the above methods cannot converge for $\mathbb{P}^1$ elements.
- Show that Newton's and damped Newton's methods cannot converge unless:
  1. It is discretized with higher order elements (such as $\mathbb{P}^2$).[†]
  2. Additional gradient variables are used.[††]
- These restrictions **hinder** the use of fast solution algorithms.

Can the Mixed Method be used with $\mathbb{P}^1$ Elements?

- Much of the literature claims the above methods cannot converge for $\mathbb{P}^1$ elements.
- Show that Newton's and damped Newton's methods cannot converge unless:
  1. It is discretized with higher order elements (such as $\mathbb{P}^2$).[†]
  2. Additional gradient variables are used.[††]
- These restrictions **hinder** the use of fast solution algorithms.
- If solving the Monge-Ampère is too computationally expensive, providing adaptivity through the MAE could be cost prohibitive for large scale problems.

Can the Mixed Method be used with $\mathbb{P}^1$ Elements?

- Much of the literature claims the above methods cannot converge for $\mathbb{P}^1$ elements.
- Show that Newton's and damped Newton's methods cannot converge unless:
  1. It is discretized with higher order elements (such as $\mathbb{P}^2$).[†]
  2. Additional gradient variables are used.[††]
- These restrictions **hinder** the use of fast solution algorithms.
- If solving the Monge-Ampère is too computationally expensive, providing adaptivity through the MAE could be cost prohibitive for large scale problems.

Objective: Can we find a low order finite element approximation for the Monge-Ampère Equation?

Implementing for Low Order Elements

Using low-order finite elements is crucial for creating highly efficient solvers for the Monge-Ampère equation.

Our improved method uses an optimization-based nonlinear solver.

- Combine auxiliary and primal variables into $x = (\mathbf{h}_{1,1}, \mathbf{h}_{1,2}, \mathbf{h}_{2,1}, \mathbf{h}_{2,2}, \mathbf{u})$. We solve the nonlinear equation:

$$c(x) = 0,$$

where c is the nonlinear residual function, $c : \mathcal{X} \to \mathcal{C}$.

Implementing for Low Order Elements

Using low-order finite elements is crucial for creating highly efficient solvers for the Monge-Ampère equation.

Our improved method uses an optimization-based nonlinear solver.

- Combine auxiliary and primal variables into $x = (\mathbf{h}_{1,1}, \mathbf{h}_{1,2}, \mathbf{h}_{2,1}, \mathbf{h}_{2,2}, \mathbf{u})$. We solve the nonlinear equation:

$$c(x) = 0,$$

  where c is the nonlinear residual function, $c : \mathcal{X} \to \mathcal{C}$.

- We have verified that Newton's method and damped Newton's method do not converge for $\mathbb{P}^1$ finite element discretizations of NVFEM.

Implementing for Low Order Elements

Using low-order finite elements is crucial for creating highly efficient solvers for the Monge-Ampère equation.

Our improved method uses an optimization-based nonlinear solver.

- Combine auxiliary and primal variables into $x = (\mathbf{h}_{1,1}, \mathbf{h}_{1,2}, \mathbf{h}_{2,1}, \mathbf{h}_{2,2}, \mathbf{u})$. We solve the nonlinear equation:

$$c(x) = 0,$$

  where c is the nonlinear residual function, $c : \mathcal{X} \to \mathcal{C}$.
- We have verified that Newton's method and damped Newton's method do not converge for $\mathbb{P}^1$ finite element discretizations of NVFEM.
- Our sequential quadratic programming (SQP) solver, where we solve the problem as

$$\begin{aligned} &\min \; 0 \\ &\text{subject to} \; c(x) = 0, \end{aligned}$$

  converges without exception for a variety of MAE examples, using $\mathbb{P}^1$ elements.

An Optimization-based Nonlinear Solver

- The SQP solver is a **composite-step method** that coordinates two steps: a step toward optimality and a step toward feasibility.

An Optimization-based Nonlinear Solver

- The SQP solver is a **composite-step method** that coordinates two steps: a step toward optimality and a step toward feasibility.
- At every nonlinear iteration k, the feasibility step or **normal step** n solves the trust-region subproblem:

$$\begin{aligned} \min \quad & \|c'(x_k)n + c(x_k)\|_{\mathcal{C}}^2 \\ \text{subject to} \quad & \|n\|_{\mathcal{X}} \le \Delta_k, \end{aligned}$$

where $c'(x_k)$ is the MAE Jacobian at iterate x_k and $c(x_k)$ is the residual.

An Optimization-based Nonlinear Solver

- The SQP solver is a **composite-step method** that coordinates two steps: a step toward optimality and a step toward feasibility.
- At every nonlinear iteration k, the feasibility step or **normal step** n solves the trust-region subproblem:

$$\begin{aligned} \min \quad & \|c'(x_k)n + c(x_k)\|_{\mathcal{C}}^2 \\ \text{subject to} \quad & \|n\|_{\mathcal{X}} \leq \Delta_k, \end{aligned}$$

  where $c'(x_k)$ is the MAE Jacobian at iterate x_k and $c(x_k)$ is the residual.
- **Minimization of linearized residual with trust regions for global convergence.**
- Our low order discretization allows for mass lumping and a faster sparse solve.

An Optimization-based Nonlinear Solver

- The SQP solver is a **composite-step method** that coordinates two steps: a step toward optimality and a step toward feasibility.
- At every nonlinear iteration k, the feasibility step or **normal step** n solves the trust-region subproblem:

$$\begin{aligned} \min \quad & \|c'(x_k)n + c(x_k)\|_{\mathcal{C}}^2 \\ \text{subject to} \quad & \|n\|_{\mathcal{X}} \leq \Delta_k, \end{aligned}$$

  where $c'(x_k)$ is the MAE Jacobian at iterate x_k and $c(x_k)$ is the residual.
- **Minimization of linearized residual with trust regions for global convergence.**
- Significantly more robust than damped Newton.
- Our low order discretization allows for mass lumping and a faster sparse solve.

An Optimization-based Nonlinear Solver

- The SQP solver is a **composite-step method** that coordinates two steps: a step toward optimality and a step toward feasibility.
- At every nonlinear iteration k, the feasibility step or **normal step** n solves the trust-region subproblem:

$$\begin{aligned} \min \quad & \|c'(x_k)n + c(x_k)\|_{\mathcal{C}}^2 \\ \text{subject to} \quad & \|n\|_{\mathcal{X}} \leq \Delta_k, \end{aligned}$$

  where $c'(x_k)$ is the MAE Jacobian at iterate x_k and $c(x_k)$ is the residual.
- **Minimization of linearized residual with trust regions for global convergence.**
- Significantly more robust than damped Newton.
- Our low order discretization allows for mass lumping and a faster sparse solve.
- Can solve a smooth problem to machine precision on a 256×256 mesh in about 24 seconds.

An Optimization-based Nonlinear Solver

- The SQP solver is a **composite-step method** that coordinates two steps: a step toward optimality and a step toward feasibility.
- At every nonlinear iteration k, the feasibility step or **normal step** n solves the trust-region subproblem:

$$\begin{aligned} \min \quad & \|c'(x_k)n + c(x_k)\|_{\mathcal{C}}^2 \\ \text{subject to} \quad & \|n\|_{\mathcal{X}} \leq \Delta_k, \end{aligned}$$

  where $c'(x_k)$ is the MAE Jacobian at iterate x_k and $c(x_k)$ is the residual.
- **Minimization of linearized residual with trust regions for global convergence.**
- Significantly more robust than damped Newton.
- Our low order discretization allows for mass lumping and a faster sparse solve.
- Can solve a smooth problem to machine precision on a 256×256 mesh in about 24 seconds.
- **Crucial for providing optimal transport based mesh adaptation for large scale problems.**

Results – Sine Wave

Right hand side density of the Monge-Ampère equation:

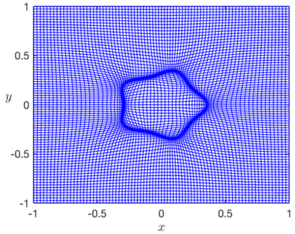
$$f = \frac{1}{0.65 + 10\exp(-50(y - 0.5 - 0.25\sin(2\pi x))^2)}.$$

Results – Bitmap Adaptation

Meshing to a bitmap, density for black pixels = 1, density for white pixels = 0.

Conclusions

- Fast, low order finite element approximations for the Monge-Ampère equation.
- Utilizes $\mathbb{P}^1$ finite elements to efficient solve the MAE.
- Accurately solves the MAE for various solution types seen in the literature.
- Plan to use for mesh adaptivity in convex domains where FD methods fail.



Grid Size	Comp. Time
$2^5 \times 2^5$	.42 sec
$2^6 \times 2^6$	1.27 sec
$2^7 \times 2^7$	4.91 sec
$2^8 \times 2^8$	24.5 sec
$2^9 \times 2^9$	160.9 sec

Please feel free to contact me at for more information and further discussion (kdipiet@sandia.gov)