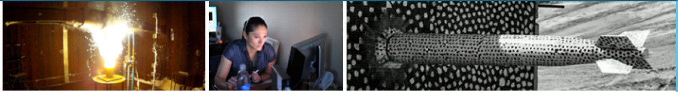
-9253C

Sandia National Laboratories

Optimization-based multigrid-enabled finite element solvers for the Monge-Ampère equation



Kelsey DiPietro, Diana Morales, and Denis Ridzal
ICCOPT
Lehigh University

LDRD

U.S. DEPARTMENT OF ENERGY

NNSA

SAND2021-xxxxx C

Introduction to the Monge-Ampère Equation

Optimal transport problem:

$$\min_{\phi} \int_{\Omega} \mu_0(\mathbf{x})|\phi(\mathbf{x}) - \mathbf{x}|^2 d\mathbf{x},$$
$$\text{subject to } \det(\nabla\phi(\mathbf{x}))\mu_1(\phi(\mathbf{x})) - \mu_0(\mathbf{x}) = 0,$$

where μ_0, μ_1 are the source and target densities.

Introduction to the Monge-Ampère Equation

Optimal transport problem:

$$\min_{\phi} \int_{\Omega} \mu_0(\mathbf{x})|\phi(\mathbf{x}) - \mathbf{x}|^2 d\mathbf{x},$$
$$\text{subject to } \det(\nabla\phi(\mathbf{x}))\mu_1(\phi(\mathbf{x})) - \mu_0(\mathbf{x}) = 0,$$

where μ_0, μ_1 are the source and target densities.

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

Introduction to the Monge-Ampère Equation

Optimal transport problem:

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

$$\text{subject to } \det(\nabla\phi(\mathbf{x}))\mu_1(\phi(\mathbf{x})) - \mu_0(\mathbf{x}) = 0,$$

where μ_0, μ_1 are the source and target densities.

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

$$\det(\nabla\phi) = \det(\nabla(\nabla u)) = \det(H(u)) \quad = \det\begin{pmatrix} \frac{\partial}{\partial x} u_x & \frac{\partial}{\partial x} u_y \\ \frac{\partial}{\partial y} u_x & \frac{\partial}{\partial y} u_y \end{pmatrix} \text{ in 2D}$$

with Hessian H.

Introduction to the Monge-Ampère Equation

Optimal transport problem:

$$\min_{\phi} \int_{\Omega} \mu_0(\mathbf{x})|\phi(\mathbf{x}) - \mathbf{x}|^2 d\mathbf{x},$$
$$\text{subject to } \det(\nabla\phi(\mathbf{x}))\mu_1(\phi(\mathbf{x})) - \mu_0(\mathbf{x}) = 0,$$

where μ_0, μ_1 are the source and target densities.

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

$$\det(\nabla\phi) = \det(\nabla(\nabla u)) = \det(H(u)) \quad = \det\begin{pmatrix} \frac{\partial}{\partial x} u_x & \frac{\partial}{\partial x} u_y \\ \frac{\partial}{\partial y} u_x & \frac{\partial}{\partial y} u_y \end{pmatrix} \text{ in 2D}$$

with Hessian H.

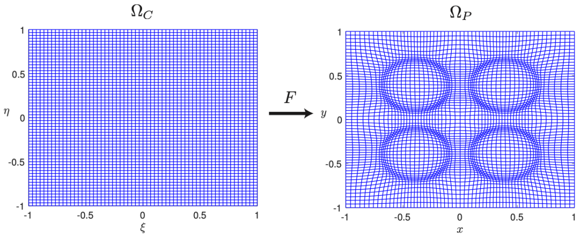
Obtain the Monge-Ampère equation (MAE) with Dirichlet boundary conditions

$$\det(H(\mathbf{u})) = \frac{\mu_0}{\mu_1} = \mathbf{f}, \quad \mathbf{u} \in \Omega,$$
$$\mathbf{u} = g, \quad \mathbf{u} \in \partial\Omega.$$

Applications of the Monge-Ampère Equation

Several applications for the Monge-Ampère equation:

- Inverse problems in seismic imaging.
- Differential geometry.
- Wasserstein Neural Networks.
- Mesh adaptation
  - If u is a convex solution to the Monge-Ampère equation, then $\vec{\mathbf{x}} = \nabla \mathbf{u}$ gives an adaptive mesh in the physical space (assumption of a transport boundary condition).
  - Mesh point locations are determined through a density function $f = \frac{\mu_0}{\mu_1}$.

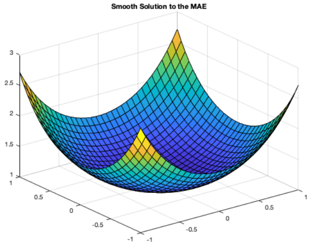


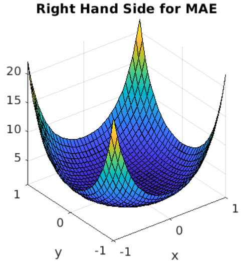
Types of Solutions

- The Monge-Ampère equation admits several possible solution types.

Types of Solutions

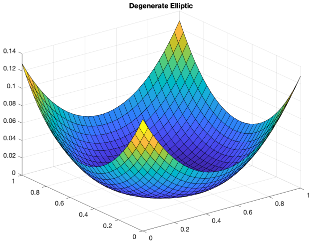
- The Monge-Ampère equation admits several possible solution types.
- Classical solutions: $u \in C^{2,\alpha}(\Omega)$
  - Strongest solution type.
  - **Classical solutions may not always exist**!

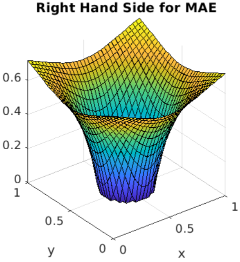




Types of Solutions

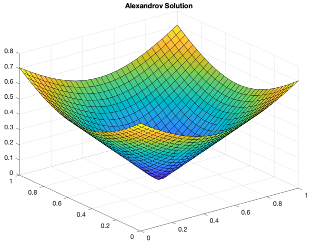
- The Monge-Ampère equation admits several possible solution types.
- Classical solutions: $u \in C^{2,\alpha}(\Omega)$
    - Strongest solution type.
    - **Classical solutions may not always exist**!
- Viscosity solutions $u \in C(\Omega)$.
    - Weak solution type.
    - Requires continuity of $f \in C(\Omega)$.
    - Based on theory of sub and super solutions and elliptic operators.

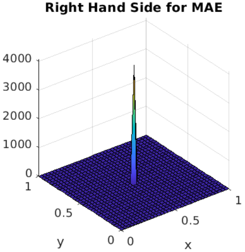




Types of Solutions

- The Monge-Ampère equation admits several possible solution types.
- Classical solutions: $u \in C^{2,\alpha}(\Omega)$
  - Strongest solution type.
  - **Classical solutions may not always exist**!
- Viscosity solutions $u \in C(\Omega)$.
  - Weak solution type.
  - Requires continuity of $f \in C(\Omega)$.
  - Based on theory of sub and super solutions and elliptic operators.
- Alexandrov solutions: $u \in C(\Omega)$.
  - Weakest solution type.
  - **Stable with respect to weak convergence**.
  - Does not require continuity in the source term f.





Conditional Ellipticity for the MAE

Define the Monge-Ampère operator as

$$F(\mathbf{x}, r, H(\mathbf{u})) = \begin{cases} \det H(\mathbf{u}) - f(\mathbf{x}), & \mathbf{x} \in \Omega \\ g(\mathbf{x}) - r, \ \mathbf{x} \in \partial\Omega. \end{cases}$$

Conditional Ellipticity for the MAE

Define the Monge-Ampère operator as

$$F(\mathbf{x}, r, H(\mathbf{u})) = \begin{cases} \det H(\mathbf{u}) - f(\mathbf{x}), & \mathbf{x} \in \Omega \\ g(\mathbf{x}) - r, \ \mathbf{x} \in \partial\Omega. \end{cases}$$

Elliptic if and only if $H(\mathbf{u})$ is restricted to positive semidefinite matrices.

Conditional Ellipticity for the MAE

Define the Monge-Ampère operator as

$$F(\mathbf{x}, r, H(\mathbf{u})) = \begin{cases} \det H(\mathbf{u}) - f(\mathbf{x}), & \mathbf{x} \in \Omega \\ g(\mathbf{x}) - r, \ \mathbf{x} \in \partial\Omega. \end{cases}$$

Elliptic if and only if $H(\mathbf{u})$ is restricted to positive semidefinite matrices.

Limits the set of admissible solutions to convex functions.

The restriction to convex functions significantly complicates numerical approaches.

Conditional Ellipticity for the MAE

Define the Monge-Ampère operator as

$$F(\mathbf{x}, r, H(\mathbf{u})) = \begin{cases} \det H(\mathbf{u}) - f(\mathbf{x}), & \mathbf{x} \in \Omega \\ g(\mathbf{x}) - r, \ \mathbf{x} \in \partial\Omega. \end{cases}$$

Elliptic if and only if $H(\mathbf{u})$ is restricted to positive semidefinite matrices.

Limits the set of admissible solutions to convex functions.

The restriction to convex functions significantly complicates numerical approaches.

We will “sidestep” convexity through a robust nonlinear solver.

Highlighted Numerical Method

- Focus on the mixed finite element method (or *nonvariational* finite element method) by Lakkis and Pryer †.
- Mixed in that we have auxiliary variables that define the Hessian matrix – thus resulting in a system of equations.
- Remaining talk summary:
  1. Mixed finite element method for linear elliptic problems that use the Hessian matrix.
  2. Application to general nonlinear elliptic systems.
  3. Focusing specifically on the Monge-Ampère equation, propose an **optimization-based multigrid-enabled nonlinear solver**.
  4. Present a variety of 2D and 3D examples.

† **Lakkis, Pryer (2011)**, *A finite element method for second order nonvariational elliptic problems*, SISC.

Mixed Finite Element Method for Elliptic Problems

This method directly applies conforming finite elements to elliptic problems dependent on the Hessian matrix †. Uses the notation D^2u to denote the Hessian of u.

General formulation entails finding **u** such that

$$\mathbf{A} : D^2\mathbf{u} = f \text{ in } \Omega \quad \text{and } \mathbf{u} = g \text{ on } \partial\Omega,$$

for $\mathbf{A} : \Omega \to \mathbb{R}^{d\times d}$.

$\mathbf{A} : D^2\mathbf{u} = \text{trace}(\mathbf{A}^T D^2\mathbf{u})$ is the Frobenius inner product.

† **Lakkis, Pryer (2011)**, *A finite element method for second order nonvariational elliptic problems*, SISC.

Mixed Finite Element Method for Elliptic Problems

This method directly applies conforming finite elements to elliptic problems dependent on the Hessian matrix †. Uses the notation $D^2 u$ to denote the Hessian of u.

General formulation entails finding $\mathbf{u}$ such that

$$\mathbf{A} : D^2\mathbf{u} = f \text{ in } \Omega \quad \text{and } \mathbf{u} = g \text{ on } \partial\Omega,$$

for $\mathbf{A} : \Omega \to \mathbb{R}^{d \times d}$.

$\mathbf{A} : D^2\mathbf{u} = \mathrm{trace}(\mathbf{A}^T D^2\mathbf{u})$ is the Frobenius inner product.

Requires a finite element approximation for the Hessian $D^2\mathbf{u}$.

† **Lakkis, Pryer (2011)**, *A finite element method for second order nonvariational elliptic problems*, SISC.

Mixed Finite Element Method for Elliptic Problems

This method directly applies conforming finite elements to elliptic problems dependent on the Hessian matrix †. Uses the notation D^2u to denote the Hessian of u.

General formulation entails finding $\mathbf{u}$ such that

$$\mathbf{A} : D^2\mathbf{u} = f \text{ in } \Omega \quad \text{and } \mathbf{u} = g \text{ on } \partial\Omega,$$

for $\mathbf{A} : \Omega \to \mathbb{R}^{d\times d}$.

$\mathbf{A} : D^2\mathbf{u} = \text{trace}(\mathbf{A}^T D^2\mathbf{u})$ is the Frobenius inner product.

Requires a finite element approximation for the Hessian $D^2\mathbf{u}$.

Allows us to solve an *augmented system* for the elliptic PDE and the Hessian.

† **Lakkis, Pryer (2011)**, *A finite element method for second order nonvariational elliptic problems*, SISC.

Mixed Finite Element Method for Elliptic Problems

Define a piecewise polynomial finite element space $\mathbb{P}^k$ for elements K in triangulation $\mathcal{T}$.

$$\mathbb{V} := \{\Phi \in H^1(\Omega) : \Phi|_K \in \mathbb{P}^k \, \forall K \in \mathcal{T}\},$$

$$\mathring{\mathbb{V}} := \mathbb{V} \cap H_0^1(\Omega) = \{\Phi \in \mathbb{V} : \Phi|_{\partial\Omega} 0\}.$$

Mixed Finite Element Method for Elliptic Problems

Define a piecewise polynomial finite element space $\mathbb{P}^k$ for elements K in triangulation $\mathcal{T}$.

$$\mathbb{V} := \{\Phi \in H^1(\Omega) : \Phi|_K \in \mathbb{P}^k \, \forall K \in \mathcal{T}\},$$

$$\mathring{\mathbb{V}} := \mathbb{V} \cap H_0^1(\Omega) = \{\Phi \in \mathbb{V} : \Phi|_{\partial\Omega} 0\}.$$

For test function $\phi \in H_0^1(\Omega)$, an FE approximation for the problem is given by

$$\langle \mathbf{A} : D^2\mathbf{u}, \phi \rangle = \langle f, \phi \rangle.$$

Still need an approximation for the Hessian $D^2\mathbf{u}$.

Finite Element Hessian

Consider the smooth function $v \in C^2(\Omega) \cap C^1(\overline{\Omega})$ and apply integration by parts to show the Hessian $D^2 v$ of v satisfies

$$\langle D^2 v, \varphi \rangle = -\langle \nabla v D\varphi \rangle + \langle \nabla v \mathbf{n}^T \varphi \rangle_{\partial\Omega}, \text{ for each } \varphi \in H^1(\Omega).$$

Finite Element Hessian

Consider the smooth function $v \in C^2(\Omega) \cap C^1(\overline{\Omega})$ and apply integration by parts to show the Hessian $D^2 v$ of v satisfies

$$\langle D^2 v, \varphi \rangle = -\langle \nabla v D\varphi \rangle + \langle \nabla v \mathbf{n}^T \varphi \rangle_{\partial\Omega}, \text{ for each } \varphi \in H^1(\Omega).$$

Define the generalized Hessian of v, where $D^2 v$ maps to $\mathbb{R}^{d\times d}$.

Generalize for functions $v \in H^1(\Omega)$ with $\nabla v n^T|_{\partial\Omega}$ *in* $\left(H^{1/2}(\partial\Omega)'\right)^{d\times d}$.

$$\langle D^v | \varphi \rangle := -\langle \nabla v D\varphi \rangle + \langle \nabla v n^T | \varphi \rangle_{(H^{1/2}(\partial\Omega))\times H^{1/2}(\partial\Omega)} \text{ for each } \varphi \in H^1(\Omega)$$

.

Finite Element Hessian

Consider the smooth function $v \in C^2(\Omega) \cap C^1(\overline{\Omega})$ and apply integration by parts to show the Hessian $D^2 v$ of v satisfies

$$\langle D^2 v, \varphi \rangle = -\langle \nabla v D\varphi \rangle + \langle \nabla v \mathbf{n}^T \varphi \rangle_{\partial\Omega}, \text{ for each } \varphi \in H^1(\Omega).$$

Define the generalized Hessian of v, where $D^2 v$ maps to $\mathbb{R}^{d \times d}$.

Generalize for functions $v \in H^1(\Omega)$ with $\nabla v n^T|_{\partial\Omega}$ *in* $\left(H^{1/2}(\partial\Omega)'\right)^{d \times d}$.

$$\langle D^v | \varphi \rangle := -\langle \nabla v D\varphi \rangle + \langle \nabla v n^T | \varphi \rangle_{(H^{1/2}(\partial\Omega)) \times H^{1/2}(\partial\Omega)} \text{ for each } \varphi \in H^1(\Omega)$$

. Define the finite element Hessian for $v \in \mathbb{V}$,

$$\langle \mathbf{H} v, \Phi \rangle_{\Omega_\mathcal{T}} = \langle D^2 v | \Phi \rangle = \langle \nabla V \otimes \nabla \Phi \rangle + \langle \nabla V \otimes \mathbf{n} \Phi \rangle_{\partial\Omega} \quad \forall \Phi \in \mathbb{V},$$

where $a \otimes b = ab^T$ for vectors a, b.

Finite Element Convexity for Low-Order Elements

Aguilera and Morin [†] define the potential limitation for low-order finite elements to accurately approximate convex functions .

Define the discrete finite element Hessian:

$$\langle Hu, \varphi \rangle_{ij} = -\int_{\Omega} \partial_i u(\mathbf{x}) \partial_j \phi(\mathbf{x}) d\mathbf{x} + \int_{\partial\Omega} \partial_i u(\mathbf{x}) \varphi(\mathbf{x}) \mathbf{n}_j dS.$$

Finite element convexity: A function $u \in V_h$ is finite element convex with respect to test and trial functions $\{\phi_r^h\}, \{\varphi_s^h\}$ if $H_s^h \succeq 0$ for all $s \in I_{test}^h$.

[†] **Aguilera, Morin (2009)**, *On convex functions and the finite element method*, SINUM.

Finite Element Convexity for Low-Order Elements

Aguilera and Morin [†] define the potential limitation for low-order finite elements to accurately approximate convex functions .

Define the discrete finite element Hessian:

$$\langle Hu, \varphi \rangle_{ij} = -\int_{\Omega} \partial_i u(\mathbf{x}) \partial_j \phi(\mathbf{x}) d\mathbf{x} + \int_{\partial\Omega} \partial_i u(\mathbf{x}) \varphi(\mathbf{x}) \mathbf{n}_j dS.$$

Finite element convexity: A function $u \in V_h$ is finite element convex with respect to test and trial functions $\{\phi_r^h\}, \{\varphi_s^h\}$ if $H_s^h \succeq 0$ for all $s \in I_{test}^h$.

They note in their experiments that FE convexity is not **always** guaranteed when using linear elements so they cannot prove convergence.

However this does not mean that it is impossible for linear elements to be finite element convex.

[†] **Aguilera, Morin (2009)**, *On convex functions and the finite element method*, SINUM.

Mixed FEM System

Discretized augmented system for a Dirichlet boundary condition, $\mathbf{E}\mathbf{v} = \mathbf{b} - \mathbf{E}_{\mathbf{dc}}\mathbf{b}_{\mathbf{dc}}$.

$$\mathbf{E} = \begin{bmatrix} \mathbf{M} & \mathbf{0} & \dots & \mathbf{0} & -\mathbf{C}_{1,1} \\ \mathbf{0} & \mathbf{M} & \dots & \mathbf{0} & -\mathbf{C}_{1,2} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{M} & -\mathbf{C}_{d,d} \\ \mathbf{B}^{1,1} & \mathbf{B}^{1,2} & \dots & \mathbf{B}^{d,d} & \mathbf{0}. \end{bmatrix}, \quad \mathbf{E}_{\mathbf{dc}} = \begin{bmatrix} -\mathbf{C}^{\mathbf{dc}}_{1,1} \\ -\mathbf{C}^{\mathbf{dc}}_{1,2} \\ \vdots \\ -\mathbf{C}^{\mathbf{dc}}_{d,d} \\ \mathbf{0}. \end{bmatrix}.$$

$$\mathbf{v} = (\mathbf{h}_{1,1}, \mathbf{h}_{1,2}, \dots, \mathbf{h}_{d,d}, \mathbf{u})^T, \mathbf{b} = (\mathbf{0}, \mathbf{0}, \dots, \mathbf{0}, \mathbf{f}), \mathbf{b}_{\mathbf{dc}} = [\mathbf{g}]^T.$$

$$\mathbf{B}^{\alpha,\beta} := \langle \mathring{\Phi}, \mathbf{A}^{\alpha,\beta}\Phi^T \rangle \in \mathbb{R}^{\mathring{N}\times N},$$

$$\mathbf{M} := \langle \Phi, \Phi^T \rangle \in \mathbb{R}^{N\times N},$$

$$\mathbf{C}_{\alpha,\beta} := -\langle \partial_\beta \Phi, \partial_\alpha \mathring{\Phi}^T \rangle + \langle \Phi n_\beta, \partial_\alpha \mathring{\Phi}^T \rangle_{\partial\Omega} \in \mathbb{R}^{\mathring{N}\times N}$$

$$\mathbf{f} := \langle f, \mathring{\Phi} \rangle \in \mathbb{R}^{\mathring{N}}$$

Schur Complement System

If low order elements are used, then $\mathbf{M}$ can be diagonalized using mass lumping and a Schur complement can be used to solve for $\mathbf{u}$ and the auxiliary Hessian variables are recovered separately.

$$\mathbf{D}\mathbf{u} := \sum_{\alpha=1}^{d}\sum_{\beta=1}^{d} \mathbf{B}^{\alpha,\beta}\mathbf{M}^{-1}\mathbf{C}_{\alpha,\beta}\mathbf{u} = \mathbf{f}.$$

$$\begin{aligned}
\mathbf{B}^{\alpha,\beta} &:= \langle \mathring{\Phi}, \mathbf{A}^{\alpha,\beta}\Phi^{T}\rangle \in \mathbb{R}^{\mathring{N}\times N}, \\
\mathbf{M} &:= \langle \Phi, \Phi^{T}\rangle \in \mathbb{R}^{N\times N}, \\
\mathbf{C}_{\alpha,\beta} &:= -\langle \partial_\beta \Phi, \partial_\alpha \mathring{\Phi}^{T}\rangle + \langle \Phi n_\beta, \partial_\alpha \mathring{\Phi}^{T}\rangle_{\partial\Omega} \in \mathbb{R}^{\mathring{N}\times N} \\
\mathbf{f} &:= \langle f, \mathring{\Phi}\rangle \in \mathbb{R}^{\mathring{N}}
\end{aligned}$$

Note: We can reduce the size of the system by using the symmetry of the Hessian, i.e. $h_{a,b} = h_{b,a}$.

Mixed FEM Equivalence to Standard FEM

Equivalence to standard FEM

†For the second order elliptic problem $\mathbf{A} : D^2 u = f$, if the problem coefficients in $\mathbf{A}$ are piecewise constant then the mixed finite element method coincides with the standard FEM $\mathbf{A} : D^2 u = \text{div}(\mathbf{A}\nabla u)$. This implies that $\mathbf{u}$ solves both

$$\mathbf{Du} = \mathbf{f} \quad \text{and} \quad \mathbf{Su} = \mathbf{f}$$

$$\text{where} \quad \mathbf{S} = \sum_{\alpha,\beta=1}^{d} \langle \partial_\beta \mathbf{\Phi}, a^{\alpha,\beta} \partial_\alpha \mathbf{\Phi}^T \rangle.$$

Mixed FEM Equivalence to Standard FEM

Equivalence to standard FEM

[†]For the second order elliptic problem $\mathbf{A} : D^2 u = f$, if the problem coefficients in $\mathbf{A}$ are piecewise constant then the mixed finite element method coincides with the standard FEM $\mathbf{A} : D^2 u = \operatorname{div}(\mathbf{A}\nabla u)$. This implies that $\mathbf{u}$ solves both

$$\mathbf{Du} = \mathbf{f} \quad \text{and} \quad \mathbf{Su} = \mathbf{f}$$

$$\text{where} \quad \mathbf{S} = \sum_{\alpha,\beta=1}^{d} \langle \partial_\beta \mathbf{\Phi}, a^{\alpha,\beta} \partial_\alpha \mathbf{\Phi}^T \rangle.$$

Opens up the possibility to use **algebraic multigrid** to solve the Schur complement system, due to its structure, which resembles the structure of the **discrete Laplacian**.

[†]**Lakkis, Pryer (2011)**, *A finite element method for second order nonvariational elliptic problems*, SISC.

Mixed FEM for Nonlinear Elliptic Problems

General nonlinear elliptic problem††:

$$\mathcal{N}[\mathbf{u}] = F(D^2\mathbf{u}) - \mathbf{f} = 0 \quad \text{in } \Omega.$$

†† **Lakkis, Pryer (2012)**, *A finite element method for nonlinear elliptic problems*, SISC.

Mixed FEM for Nonlinear Elliptic Problems

General nonlinear elliptic problem[††]:

$$\mathcal{N}[\mathbf{u}] = F(D^2\mathbf{u}) - \mathbf{f} = 0 \quad \text{in}\, \Omega.$$

Previous applications use a Newton's method to obtain the linear system of equations

$$\begin{aligned} \mathbf{N}(D^2\mathbf{u}^n) : D^2\mathbf{u}^{n+1} = g(D^2\mathbf{u}^n), \\ \mathbf{N}(\mathbf{X}) := F'(\mathbf{X}), \\ g(\mathbf{X}) := f - F(\mathbf{X}) + F'(\mathbf{X}) : \mathbf{X}. \end{aligned}$$

†† **Lakkis, Pryer (2012)**, *A finite element method for nonlinear elliptic problems*, SISC.

Mixed FEM for Nonlinear Elliptic Problems

General nonlinear elliptic problem[††]:

$$\mathcal{N}[\mathbf{u}] = F(D^2\mathbf{u}) - \mathbf{f} = 0 \quad \text{in}\, \Omega.$$

Previous applications use a Newton's method to obtain the linear system of equations

$$\begin{aligned} \mathbf{N}(D^2\mathbf{u}^n) : D^2\mathbf{u}^{n+1} &= g(D^2\mathbf{u}^n), \\ \mathbf{N}(\mathbf{X}) &:= F'(\mathbf{X}), \\ g(\mathbf{X}) &:= f - F(\mathbf{X}) + F'(\mathbf{X}) : \mathbf{X}. \end{aligned}$$

Solve using the mixed finite element method, given an initial guess $U^0 := \Pi_0 u^0$ for each $n \in \mathbb{N}_0$ find $(U^{n+1}, \mathbf{H}[U^{n+1}]) \in \mathbb{V} \times \mathbb{V}^{d\times d}$ such that,

$$\begin{aligned} < \mathbf{H}[U^{n+1}], \Phi > + \int_\Omega \nabla U^{n+1} \otimes \nabla \Phi - \int_{\partial\Omega} \nabla U^{n+1} \otimes \mathbf{n}\Phi = 0 \quad \forall \Phi \in \mathbb{V}, \\ < \mathbf{N}(\mathbf{H}[U^n]) : \mathbf{H}[U^{n+1}], \Psi >=< g(\mathbf{H}[U^n]), \Psi > \quad \forall \Psi \in \mathbb{V}, \end{aligned}$$

†† **Lakkis, Pryer (2012)**, *A finite element method for nonlinear elliptic problems*, SISC.

Mixed FEM Applied to the MAE

Recall the MAE:

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

Mixed FEM Applied to the MAE

Recall the MAE:

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

Newton's method applied to the MAE:

$$< \mathbf{H}[U^{n+1}], \Phi > + \int_{\Omega} \nabla U^{n+1} \otimes \Phi - \int_{\partial\Omega} \nabla U^{n+1} \otimes \mathbf{n}\Phi = \mathbf{0} \quad \forall \Phi \in \mathbb{V}$$

$$< \mathbf{Cof}(D^2 U^n) : \mathbf{H}[U^{n+1}], \Psi >= < f + \mathbf{det} D^2 U^n, \Psi > \ \forall \Psi \in \mathring{\mathbb{V}}.$$

Mixed FEM Applied to the MAE

Recall the MAE:

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

Newton's method applied to the MAE:

$$< \mathbf{H}[U^{n+1}], \Phi > + \int_\Omega \nabla U^{n+1} \otimes \Phi - \int_{\partial\Omega} \nabla U^{n+1} \otimes \mathbf{n}\Phi = \mathbf{0} \quad \forall \Phi \in \mathbb{V}$$

$$< \mathbf{Cof}(D^2 U^n) : \mathbf{H}[U^{n+1}], \Psi >= < f + \mathbf{det} D^2 U^n, \Psi > \ \forall \Psi \in \mathring{\mathbb{V}}.$$

Where $\mathbf{Cof}(D^2 U^n)$ is the cofactor matrix given by

$$2D: \quad \mathbf{Cof}(D^2 U^n) = \begin{pmatrix} U^n_{yy} & -U^n_{xy} \\ -U^n_{yx} & U^n_{xx} \end{pmatrix},$$

$$3D: \quad \mathbf{Cof}(D^2 U^n) = \begin{pmatrix} U_{yy}U_{zz} - U^2_{yz} & U_{yz}U_{xz} - U_{xy}U_{zz} & U_{xy}U_{yz} - U_{yy}U_{xz} \\ U_{yz}U_{xz} - U_{xy}U_{zz} & U_{xx}U_{zz} - U^2_{xz} & U_{xy}U_{xz} - U_{xx}U_{yz} \\ U_{xy}U_{yz} - U_{yy}U_{xz} & U_{xy}U_{xz} - U_{xx}U_{yz} & U_{xx}U_{yy} - U^2_{xy} \end{pmatrix}$$

The Mixed FEM for $\mathbb{P}^1$ Elements

Previous applications of the method are limited to $\mathbb{P}^k$ elements with $k \geq 2$. [†]

Further asserted in a follow-up paper [††] that $\mathbb{P}^1$ elements can only be used with a gradient recovery operator.

[†] **Lakkis, Pryer (2011)**, *A finite element method for second order nonvariational elliptic problems*, SJSC.

The Mixed FEM for $\mathbb{P}^1$ Elements

Previous applications of the method are limited to $\mathbb{P}^k$ elements with $k \geq 2$. [†]

Further asserted in a follow-up paper [††] that $\mathbb{P}^1$ elements can only be used with a gradient recovery operator.

List of assertions about the method:

[†] **Lakkis, Pryer (2011)**, *A finite element method for second order nonvariational elliptic problems*, SJSC.
[††] **Kawecki, Lakkis, Pryer (2018)**, *A finite element method for the Monge-Ampère equation with optimal transport boundary condition*, arXiv.

The Mixed FEM for $\mathbb{P}^1$ Elements

Previous applications of the method are limited to $\mathbb{P}^k$ elements with $k \geq 2$. †

Further asserted in a follow-up paper †† that $\mathbb{P}^1$ elements can only be used with a gradient recovery operator.

List of assertions about the method:

- Newton's method and damped Newton's methods do not converge for $\mathbb{P}^1$.

† **Lakkis, Pryer (2011)**, *A finite element method for second order nonvariational elliptic problems*, SJSC.
†† **Kawecki, Lakkis, Pryer (2018)**, *A finite element method for the Monge-Ampère equation with optimal transport boundary condition*, arXiv.

The Mixed FEM for $\mathbb{P}^1$ Elements

Previous applications of the method are limited to $\mathbb{P}^k$ elements with $k \geq 2$. †

Further asserted in a follow-up paper †† that $\mathbb{P}^1$ elements can only be used with a gradient recovery operator.

List of assertions about the method:

- Newton's method and damped Newton's methods do not converge for $\mathbb{P}^1$.
- Finite element convexity cannot be preserved during the Newton's iterations. **?**

†**Lakkis, Pryer (2011)**, *A finite element method for second order nonvariational elliptic problems*, SJSC.
††**Kawecki, Lakkis, Pryer (2018)**, *A finite element method for the Monge-Ampère equation with optimal transport boundary condition*, arXiv.

The Mixed FEM for $\mathbb{P}^1$ Elements

Previous applications of the method are limited to $\mathbb{P}^k$ elements with $k \geq 2$. [†]

Further asserted in a follow-up paper [††] that $\mathbb{P}^1$ elements can only be used with a gradient recovery operator.

List of assertions about the method:

- Newton's method and damped Newton's methods do not converge for $\mathbb{P}^1$.
- Finite element convexity cannot be preserved during the Newton's iterations. **?**
- The method only converges for $\mathbb{P}^k$, $k \geq 2$, so using Schur complements becomes impractical.**?**

[†] **Lakkis, Pryer (2011)**, *A finite element method for second order nonvariational elliptic problems*, SJSC.
[††] **Kawecki, Lakkis, Pryer (2018)**, *A finite element method for the Monge-Ampère equation with optimal transport boundary condition*, arXiv.

The Mixed FEM for $\mathbb{P}^1$ Elements

Previous applications of the method are limited to $\mathbb{P}^k$ elements with $k \geq 2$. †

Further asserted in a follow-up paper †† that $\mathbb{P}^1$ elements can only be used with a gradient recovery operator.

List of assertions about the method:

- Newton's method and damped Newton's methods do not converge for $\mathbb{P}^1$.
- Finite element convexity cannot be preserved during the Newton's iterations. **?**
- The method only converges for $\mathbb{P}^k$, $k \geq 2$, so using Schur complements becomes impractical.**?**
- Convergence with $\mathbb{P}^1$ elements and Newton's method requires an auxiliary gradient recovery operator.**?**

† **Lakkis, Pryer (2011)**, *A finite element method for second order nonvariational elliptic problems*, SJSC.
†† **Kawecki, Lakkis, Pryer (2018)**, *A finite element method for the Monge-Ampère equation with optimal transport boundary condition*, arXiv.

The Mixed FEM for $\mathbb{P}^1$ Elements

Previous applications of the method are limited to $\mathbb{P}^k$ elements with $k \geq 2$. [†]

Further asserted in a follow-up paper [††] that $\mathbb{P}^1$ elements can only be used with a gradient recovery operator.

List of assertions about the method:

- Newton's method and damped Newton's methods do not converge for $\mathbb{P}^1$.
- Finite element convexity cannot be preserved during the Newton's iterations. **?**
- The method only converges for $\mathbb{P}^k$, $k \geq 2$, so using Schur complements becomes impractical.**?**
- Convergence with $\mathbb{P}^1$ elements and Newton's method requires an auxiliary gradient recovery operator.**?**

Motivates using a robust optimization-based solver for the low-order system.

[†] **Lakkis, Pryer (2011)**, *A finite element method for second order nonvariational elliptic problems*, SJSC.
[††] **Kawecki, Lakkis, Pryer (2018)**, *A finite element method for the Monge-Ampère equation with optimal transport boundary condition*, arXiv.

An Optimization-based Nonlinear Solver

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

$$c(x) = 0,$$

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

An Optimization-based Nonlinear Solver

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

$$c(x) = 0,$$

  where c is the nonlinear residual function, $c : \mathcal{X} \to \mathcal{C}$.
- Verified that Newton's method and damped Newton's method do not converge for $\mathbb{P}^1$ mixed finite element discretizations.

An Optimization-based Nonlinear Solver

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

$$c(x) = 0,$$

  where c is the nonlinear residual function, $c : \mathcal{X} \to \mathcal{C}$.
- Verified that Newton's method and damped Newton's method do not converge for $\mathbb{P}^1$ mixed finite element discretizations.
- **Observation:** Solving the problem

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

  with a *composite-step sequential quadratic programming (SQP)* solver **converges without exception** for a variety of MAE examples, using $\mathbb{P}^1$ elements.
- Simplify the nonlinear solution method by using the normal step, and eliminating the tangential step, of the SQP method†.

† **Heinkenschloss, Ridzal (2014).** *A matrix-free trust-region SQP method for equality constrained optimization*, SIOPT.

SQP Summary

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

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

where $c'(x_k)$ is the MAE Jacobian at iterate x_k, $c(x_k)$ is the residual, and Δ_k is the trust-region radius.

SQP Summary

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

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

  where $c'(x_k)$ is the MAE Jacobian at iterate x_k, $c(x_k)$ is the residual, and Δ_k is the trust-region radius.
- **Minimization of linearized residual with trust regions for global convergence.**

SQP Summary

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

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

  where $c'(x_k)$ is the MAE Jacobian at iterate x_k, $c(x_k)$ is the residual, and Δ_k is the trust-region radius.
- **Minimization of linearized residual with trust regions for global convergence.**
- Related to the Levenberg-Marquardt method for nonlinear least-squares problems.

SQP Summary

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

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

  where $c'(x_k)$ is the MAE Jacobian at iterate x_k, $c(x_k)$ is the residual, and Δ_k is the trust-region radius.
- **Minimization of linearized residual with trust regions for global convergence.**
- Related to the Levenberg-Marquardt method for nonlinear least-squares problems.
- Can also be viewed as the Gauss-Newton method with trust regions.

SQP Summary

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

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

  where $c'(x_k)$ is the MAE Jacobian at iterate x_k, $c(x_k)$ is the residual, and Δ_k is the trust-region radius.
- **Minimization of linearized residual with trust regions for global convergence.**
- Related to the Levenberg-Marquardt method for nonlinear least-squares problems.
- Can also be viewed as the Gauss-Newton method with trust regions.
- Significantly more robust than damped Newton.

SQP Summary

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

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

  where $c'(x_k)$ is the MAE Jacobian at iterate x_k, $c(x_k)$ is the residual, and Δ_k is the trust-region radius.
- **Minimization of linearized residual with trust regions for global convergence.**
- Related to the Levenberg-Marquardt method for nonlinear least-squares problems.
- Can also be viewed as the Gauss-Newton method with trust regions.
- Significantly more robust than damped Newton.
- We compute the feasibility step using **Powell's dogleg method**.

Dogleg Method

Cauchy point

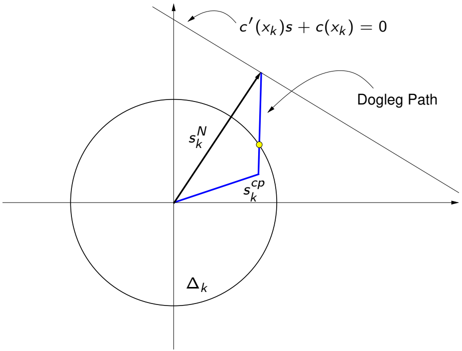
$$s_k^{cp} = \min_{\alpha \geq 0, \|s\|_{\mathcal{X}} \leq \Delta_k} \|c'(x_k)s + c(x_k)\|_{\mathcal{C}}^2$$

$$\text{subject to} \quad s = -\alpha c'(x_k)^* c(x_k)$$

Newton point

$s_k^N =$ minimum norm solution of

$$\min_s \quad \|c'(x_k)s + c(x_k)\|_{\mathcal{C}}^2$$

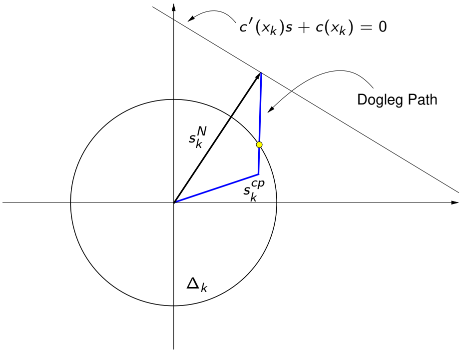


Dogleg Method

Cauchy point

$$s_k^{cp} = \min_{\alpha \geq 0, \|s\|_{\mathcal{X}} \leq \Delta_k} \|c'(x_k)s + c(x_k)\|_{\mathcal{C}}^2$$

$$\text{subject to} \quad s = -\alpha c'(x_k)^* c(x_k)$$



Newton point

s_k^N = minimum norm solution of

$$\min_s \|c'(x_k)s + c(x_k)\|_{\mathcal{C}}^2$$

For an invertible $c'(x_k)$, the Newton point s_k^N can be computed as $s_k^N = -(c'(x_k))^{-1}c(x_k)$.

Dogleg Method

Cauchy point

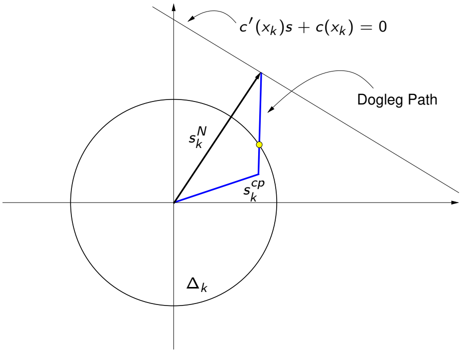
$$s_k^{cp} = \min_{\alpha \geq 0, \|s\|_{\mathcal{X}} \leq \Delta_k} \|c'(x_k)s + c(x_k)\|_{\mathcal{C}}^2$$

$$\text{subject to} \quad s = -\alpha c'(x_k)^* c(x_k)$$

Newton point

$s_k^N =$ minimum norm solution of

$$\min_s \|c'(x_k)s + c(x_k)\|_{\mathcal{C}}^2$$



For an invertible $c'(x_k)$, the Newton point s_k^N can be computed as $s_k^N = -(c'(x_k))^{-1} c(x_k)$.

Step: $s_k = s_k^{cp} + s_k^N$.

If $\|s_k\|_{\mathcal{X}} > \Delta_k$, compute $s_k = s_k^{cp} + \tau s_k^N$, with $\left\| s_k^{cp} + \tau (s_k^N - s_k^{cp}) \right\|_{\mathcal{X}}^2 = \Delta_k^2$.

Step Acceptance

Calculate ratio of actual and predicted reduction,

$$r = \frac{\|c(x_k)\|_{\mathcal{C}}^2 - \|c(x_k + s_k)\|_{\mathcal{C}}^2}{\|c(x_k)\|_{\mathcal{C}}^2 - \|c'(x_k)s_k + c(x_k)\|_{\mathcal{C}}^2} .$$

Step Acceptance

Calculate ratio of actual and predicted reduction,

$$r = \frac{\|c(x_k)\|_{\mathcal{C}}^2 - \|c(x_k + s_k)\|_{\mathcal{C}}^2}{\|c(x_k)\|_{\mathcal{C}}^2 - \|c'(x_k)s_k + c(x_k)\|_{\mathcal{C}}^2} .$$

If $r > \eta$, for some specified threshold $\eta > 0$, accept the step, $x_{k+1} = x_k + s_k$.

Step Acceptance

Calculate ratio of actual and predicted reduction,

$$r = \frac{\|c(x_k)\|_{\mathcal{C}}^2 - \|c(x_k + s_k)\|_{\mathcal{C}}^2}{\|c(x_k)\|_{\mathcal{C}}^2 - \|c'(x_k)s_k + c(x_k)\|_{\mathcal{C}}^2}.$$

If $r > \eta$, for some specified threshold $\eta > 0$, accept the step, $x_{k+1} = x_k + s_k$.

Otherwise set $x_{k+1} = x_k$ and the decrease the trust-region radius, e.g., $\Delta_{k+1} = \frac{1}{2}\|s_k\|_{\mathcal{X}}$.

Step Acceptance

Calculate ratio of actual and predicted reduction,

$$r = \frac{\|c(x_k)\|_{\mathcal{C}}^2 - \|c(x_k + s_k)\|_{\mathcal{C}}^2}{\|c(x_k)\|_{\mathcal{C}}^2 - \|c'(x_k)s_k + c(x_k)\|_{\mathcal{C}}^2}.$$

If $r > \eta$, for some specified threshold $\eta > 0$, accept the step, $x_{k+1} = x_k + s_k$.

Otherwise set $x_{k+1} = x_k$ and the decrease the trust-region radius, e.g., $\Delta_{k+1} = \frac{1}{2}\|s_k\|_{\mathcal{X}}$.

Algorithm terminates if the norm of the residual is less than a specific tolerance or the maximum number of iterations is reached.

Cauchy Point Computation

Recall $c'(x_k)$ is the Jacobian of the discretized Monge-Ampère equation evaluated at the step x_k, $c'(x_k)^*$ is the adjoint of the Jacobian, $c(x_k)$ is the residual of the Monge-Ampère equation at x_k, and Δ_k is the trust region radius at step k.

Cauchy Point Computation

Recall $c'(x_k)$ is the Jacobian of the discretized Monge-Ampère equation evaluated at the step x_k, $c'(x_k)^*$ is the adjoint of the Jacobian, $c(x_k)$ is the residual of the Monge-Ampère equation at x_k, and Δ_k is the trust region radius at step k.

First compute

$$s_k^{cp} = -\frac{\|c'(x_k)^* c(x_k)\|_{\mathcal{X}}^2}{\|c'(x_k)c'(x_k)^* c(x_k)\|_{\mathcal{C}}^2} c'(x_k)^* c(x_k).$$

Cauchy Point Computation

Recall $c'(x_k)$ is the Jacobian of the discretized Monge-Ampère equation evaluated at the step x_k, $c'(x_k)^*$ is the adjoint of the Jacobian, $c(x_k)$ is the residual of the Monge-Ampère equation at x_k, and Δ_k is the trust region radius at step k.

First compute

$$s_k^{cp} = -\frac{\|c'(x_k)^* c(x_k)\|_{\mathcal{X}}^2}{\|c'(x_k)c'(x_k)^* c(x_k)\|_{\mathcal{C}}^2} c'(x_k)^* c(x_k).$$

If $\|s_k\|_{\mathcal{X}}^{cp} > \Delta_k$, then set $s_k = \frac{\Delta_k}{\|s_k^{cp}\|} s_k^{cp}$.

Cauchy Point Computation

Recall $c'(x_k)$ is the Jacobian of the discretized Monge-Ampère equation evaluated at the step x_k, $c'(x_k)^*$ is the adjoint of the Jacobian, $c(x_k)$ is the residual of the Monge-Ampère equation at x_k, and Δ_k is the trust region radius at step k.

First compute

$$s_k^{cp} = -\frac{\|c'(x_k)^* c(x_k)\|_{\mathcal{X}}^2}{\|c'(x_k)c'(x_k)^* c(x_k)\|_{\mathcal{C}}^2} c'(x_k)^* c(x_k).$$

If $\|s_k\|_{\mathcal{X}}^{cp} > \Delta_k$, then set $s_k = \frac{\Delta_k}{\|s_k^{cp}\|} s_k^{cp}$.

Cauchy point computation is inexpensive, as it does not involve application of $(c'(x_k))^{-1}$.

Global Convergence

Algorithm converges from any initial guess under the following assumptions:

The step s_k must satisfy the boundedness condition

$$\|s_k\|_{\mathcal{X}} \leq \kappa_1 \|c(x_k)\|_{\mathcal{C}},$$

and the sufficient decrease condition

$$\|c(x_k)\|_{\mathcal{C}}^2 - \|c'(x_k)s_k + c(x_k)\|_{\mathcal{C}}^2 \geq \kappa_2 \|c(x_k)\|_{\mathcal{C}} \min\left\{\kappa_3 \|c(x_k)\|_{\mathcal{C}}, \Delta_k\right\},$$

where $\kappa_1, \kappa_2, \kappa_3 > 0$ are independent of k.

Global Convergence

Algorithm converges from any initial guess under the following assumptions:

The step s_k must satisfy the boundedness condition

$$\|s_k\|_{\mathcal{X}} \leq \kappa_1 \|c(x_k)\|_{\mathcal{C}},$$

and the sufficient decrease condition

$$\|c(x_k)\|_{\mathcal{C}}^2 - \|c'(x_k)s_k + c(x_k)\|_{\mathcal{C}}^2 \geq \kappa_2 \|c(x_k)\|_{\mathcal{C}} \min\{\kappa_3 \|c(x_k)\|_{\mathcal{C}}, \Delta_k\},$$

where $\kappa_1, \kappa_2, \kappa_3 > 0$ are independent of k.

Satisfied by the Cauchy point!!!

Global Convergence

Algorithm converges from any initial guess under the following assumptions:

The step s_k must satisfy the boundedness condition

$$\|s_k\|_{\mathcal{X}} \leq \kappa_1 \|c(x_k)\|_{\mathcal{C}},$$

and the sufficient decrease condition

$$\|c(x_k)\|_{\mathcal{C}}^2 - \|c'(x_k)s_k + c(x_k)\|_{\mathcal{C}}^2 \geq \kappa_2 \|c(x_k)\|_{\mathcal{C}} \min\left\{\kappa_3 \|c(x_k)\|_{\mathcal{C}}, \Delta_k\right\},$$

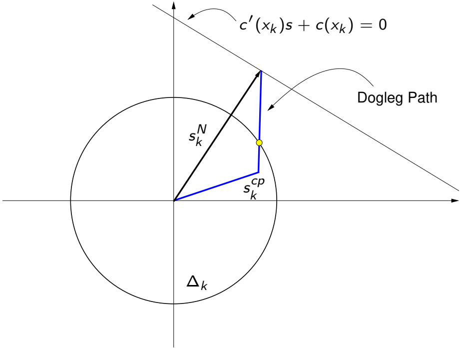
where $\kappa_1, \kappa_2, \kappa_3 > 0$ are independent of k.

Satisfied by the Cauchy point!!!

Must ensure that the Newton point does not destroy the progress made by the Cauchy point.

Newton Point Computation

Newton point: $s_k^N = -(c'(x_k))^{-1} c(x_k)$

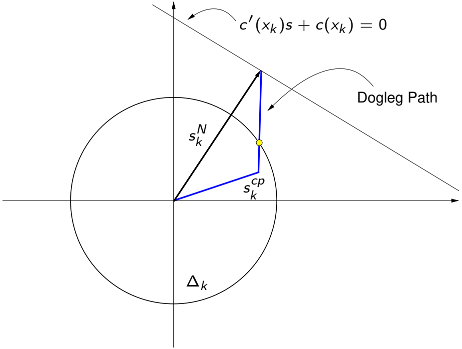


Newton Point Computation

Newton point: $s_k^N = -(c'(x_k))^{-1}c(x_k)$

Instead of solving for s_k^N, we solve for

$$\delta s_k = s_k^N - s_k^{cp}$$



Newton Point Computation

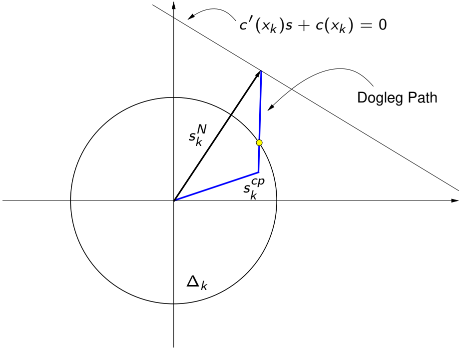
Newton point: $s_k^N = -(c'(x_k))^{-1} c(x_k)$

Instead of solving for s_k^N, we solve for

$$\delta s_k = s_k^N - s_k^{cp}$$

In other words, solve

$$c'(x_k)\delta s_k = -(c'(x_k)s_k^{cp} + c(x_k))$$



Newton Point Computation

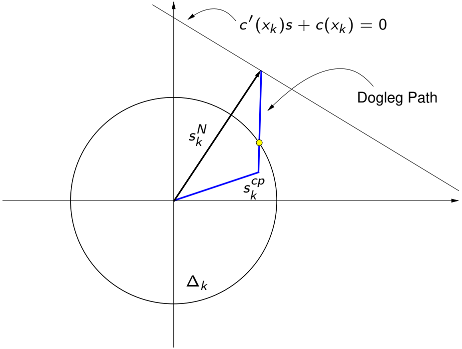
Newton point: $s_k^N = -(c'(x_k))^{-1}c(x_k)$

Instead of solving for s_k^N, we solve for

$$\delta s_k = s_k^N - s_k^{cp}$$

In other words, solve

$$c'(x_k)\delta s_k = -(c'(x_k)s_k^{cp} + c(x_k))$$



If the linear system is solved iteratively with nonzero error vector e, i.e.,

$$c'(x_k)\delta s_k = -(c'(x_k)s_k^{cp} + c(x_k)) + e,$$

then for global convergence it is sufficient to require $\|e\|_{\mathcal{C}} \,/\, \|c'(x_k)s_k^{cp} + c(x_k)\|_{\mathcal{C}} \leq 1$.
$\longrightarrow$ Any reasonable relative residual will do!!!

Efficient Solution of the Linear System

- Solve the linear system using the Schur complement,
  $\mathbf{Du} := \sum_{\alpha=1}^{d} \sum_{\beta=1}^{d} \mathbf{B}^{\alpha,\beta} \mathbf{M}^{-1} \mathbf{C}_{\alpha,\beta} \mathbf{u} = \mathbf{f}$, for $\alpha, \beta = 1, \ldots, d$,

$$\mathbf{B}^{\alpha,\beta} := \langle \mathring{\mathbf{\Phi}}, \mathbf{Cof}(\mathbf{H}[U])^{\alpha,\beta}) \mathbf{\Phi}^T \rangle \in \mathbb{R}^{\mathring{N} \times N},$$

$$\mathbf{C}_{\alpha,\beta} := -\langle \partial_\beta \Phi, \partial_\alpha \mathring{\Phi}^T \rangle + \langle \Phi n_\beta, \partial_\alpha \mathring{\Phi}^T \rangle_{\partial\Omega} \in \mathbb{R}^{\mathring{N} \times N}$$

$$\mathbf{M} := \langle \Phi, \Phi^T \rangle \in \mathbb{R}^{N \times N}, \quad \mathbf{f} := \langle f, \mathring{\Phi} \rangle \in \mathbb{R}^{\mathring{N}},\ f = -(c'(x_k) s_k^{cp} + c(x_k))$$

Efficient Solution of the Linear System

- Solve the linear system using the Schur complement,
  $\mathbf{Du} := \sum_{\alpha=1}^{d} \sum_{\beta=1}^{d} \mathbf{B}^{\alpha,\beta}\mathbf{M}^{-1}\mathbf{C}_{\alpha,\beta}\mathbf{u} = \mathbf{f}$, for $\alpha, \beta = 1, \dots, d$,

$$\mathbf{B}^{\alpha,\beta} := \langle \mathring{\boldsymbol{\Phi}}, \mathbf{Cof}(\mathbf{H}[U])^{\alpha,\beta})\boldsymbol{\Phi}^T \rangle \in \mathbb{R}^{\mathring{N} \times N},$$

$$\mathbf{C}_{\alpha,\beta} := -\langle \partial_\beta \Phi, \partial_\alpha \mathring{\Phi}^T \rangle + \langle \Phi n_\beta, \partial_\alpha \mathring{\Phi}^T \rangle_{\partial\Omega} \in \mathbb{R}^{\mathring{N} \times N}$$

$$\mathbf{M} := \langle \Phi, \Phi^T \rangle \in \mathbb{R}^{N \times N}, \quad \mathbf{f} := \langle f, \mathring{\Phi} \rangle \in \mathbb{R}^{\mathring{N}}, \; f = -(c'(x_k)s_k^{cp} + c(x_k))$$

- Schur complement of the linear system $\mathbf{Du} = \mathbf{f}$ is solved using an **algebraic multigrid method**.

Efficient Solution of the Linear System

- Solve the linear system using the Schur complement,
  $\mathbf{D}\mathbf{u} := \sum_{\alpha=1}^{d} \sum_{\beta=1}^{d} \mathbf{B}^{\alpha,\beta}\mathbf{M}^{-1}\mathbf{C}_{\alpha,\beta}\mathbf{u} = \mathbf{f}$, for $\alpha, \beta = 1, \ldots, d$,

$$\mathbf{B}^{\alpha,\beta} := \langle \mathring{\boldsymbol{\Phi}}, \mathbf{Cof}(\mathbf{H}[U])^{\alpha,\beta})\boldsymbol{\Phi}^T\rangle \in \mathbb{R}^{\mathring{N}\times N},$$

$$\mathbf{C}_{\alpha,\beta} := -\langle \partial_\beta \Phi, \partial_\alpha \mathring{\Phi}^T \rangle + \langle \Phi n_\beta, \partial_\alpha \mathring{\Phi}^T \rangle_{\partial\Omega} \in \mathbb{R}^{\mathring{N}\times N}$$

$$\mathbf{M} := \langle \Phi, \Phi^T \rangle \in \mathbb{R}^{N\times N}, \quad \mathbf{f} := \langle f, \mathring{\Phi} \rangle \in \mathbb{R}^{\mathring{N}}, \; f = -(c'(x_k)s_k^{cp} + c(x_k))$$

- Schur complement of the linear system $\mathbf{D}\mathbf{u} = \mathbf{f}$ is solved using an **algebraic multigrid method**.
- Use the Trilinos package Muelu to set up and solve the resulting system.
  - Symmetric Gauss-Siedel smoother.
  - GMRES used for solving the resulting system.
  - Future plans to explore Petrov-Galerkin methods for nonsymmetric linear systems.[†]

[†] **Sala, Tuminaro (2008)**, *A new Petrov-Galerkin smoothed aggregation preconditioner for nonsymmetric linear systems*, SISC.

Efficient Solution of the Linear System

- Solve the linear system using the Schur complement,
  $\mathbf{Du} := \sum_{\alpha=1}^{d} \sum_{\beta=1}^{d} \mathbf{B}^{\alpha,\beta}\mathbf{M}^{-1}\mathbf{C}_{\alpha,\beta}\mathbf{u} = \mathbf{f}$, for $\alpha, \beta = 1, \ldots, d$,

$$\mathbf{B}^{\alpha,\beta} := \langle \mathring{\boldsymbol{\Phi}}, \mathbf{Cof}(\mathbf{H}[U])^{\alpha,\beta})\boldsymbol{\Phi}^T \rangle \in \mathbb{R}^{\mathring{N}\times N},$$

$$\mathbf{C}_{\alpha,\beta} := -\langle \partial_\beta \Phi, \partial_\alpha \mathring{\Phi}^T \rangle + \langle \Phi n_\beta, \partial_\alpha \mathring{\Phi}^T \rangle_{\partial\Omega} \in \mathbb{R}^{\mathring{N}\times N}$$

$$\mathbf{M} := \langle \Phi, \Phi^T \rangle \in \mathbb{R}^{N\times N}, \quad \mathbf{f} := \langle f, \mathring{\Phi} \rangle \in \mathbb{R}^{\mathring{N}}, \; f = -(c'(x_k)s_k^{cp} + c(x_k))$$

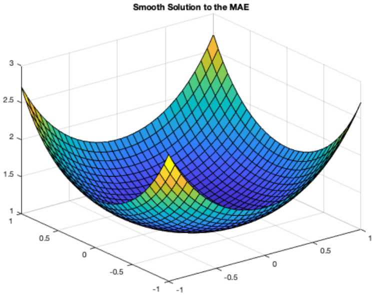
- Schur complement of the linear system $\mathbf{Du} = \mathbf{f}$ is solved using an **algebraic multigrid method**.
- Use the Trilinos package Muelu to set up and solve the resulting system.
  - Symmetric Gauss-Siedel smoother.
  - GMRES used for solving the resulting system.
  - Future plans to explore Petrov-Galerkin methods for nonsymmetric linear systems.[†]
- Hessian terms $\mathbf{h}_{\alpha,\beta}$ are recovered separately using the definition of the finite element Hessian.

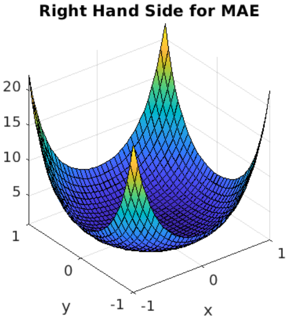
[†]**Sala,Tuminaro (2008)**, *A new Petrov-Galerkin smoothed aggregation preconditioner for nonsymmetric linear systems*, SISC.

Results - 2D Smooth Solution

Exact Solution: $u = \exp\left(\frac{|\mathbf{x}|^2}{2}\right), \ u \in C^\infty.$

Classical solution of the Monge-Ampère equation.





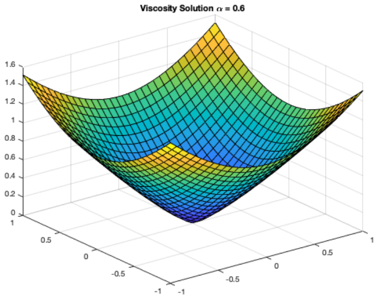
Results - 2D Smooth Solution

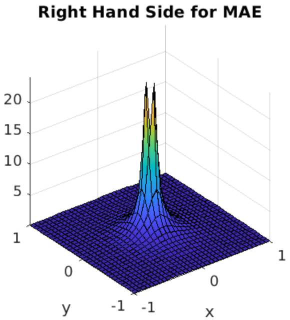
Grid Size	$\|u-u_e\|_2^2$	Rate	$\|u-u_e\|_\infty$	Rate
$2^5 \times 2^5$	3.40e-03	-	4.40e-03	-
$2^6 \times 2^6$	8.60e-04	1.98	1.10e-03	2
$2^7 \times 2^7$	2.16e-04	2.0	2.88e-04	1.93
$2^8 \times 2^8$	5.40e-05	2.0	7.27e-05	1.99

Standard smooth problem used in every numerical paper for the MAE, we attain **optimal** L_2 error estimates.

Results - 2D Viscosity Solution

Exact solution: $u(\mathbf{x}) = |\mathbf{x}|^{2\alpha}$, $\alpha \in (1/2, 3/4)$, singular derivatives at $(0,0)$, $\alpha = 0.6$.





Results - 2D Viscosity Solution

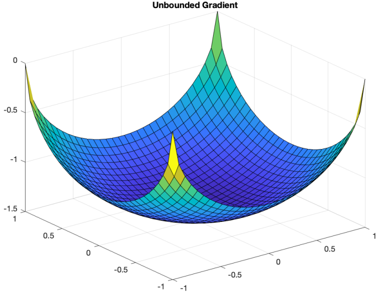
Grid Size	$\|u - u_e\|_2^2$	L_2 Rate	LP L_2 Rate[†]	$\|u - u_e\|_\infty$	∞ Error Rate
$2^5 \times 2^5$	0.1249	-	-	0.1841	-
$2^6 \times 2^6$	0.0937	0.41	0.42	0.1381	0.41
$2^7 \times 2^7$	0.0705	0.41	0.41	0.1036	0.41
$2^8 \times 2^8$	0.0531	0.41	0.41	0.0777	0.42

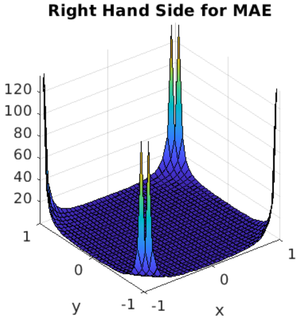
Compared to the paper by Lakkis and Pryer[†] which uses $\mathbb{P}^2$ elements to solve for the Monge-Ampère equation using a Newton's method. We attain similar errors with $\mathbb{P}^1$ element method.

Lakkis, Pryer (2011), *A finite element method for second order nonvariational elliptic problems*, SJSC.

Results - 2D Unbounded Gradient

Exact solution: $u(x, y) = -\sqrt{2 - x^2 - y^2}$, $u \in C^\infty(\Omega) \cap C^0(\Omega)$.





Results - 2D Unbounded Gradient

Grid Size	$\|u-u_e\|_2^2$	L_2 Rate	LP Rate†	$\|u-u_e\|_\infty$	FO Max Error††
$2^5 \times 2^5$	6.4e-03	-	-	8.62e-02	1.74e-03
$2^6 \times 2^6$	2.3e-03	1.48	1.62	6.10e-02	5.9e-04
$2^7 \times 2^7$	8.2460e-04	1.48	1.60	4.32e-02	2.0e-04
$2^8 \times 2^8$	2.9253e-04	1.50	1.55	3.05e-02	1.5e-04

Compared to the $\mathbb{P}^2$ mixed finite element method by Lakkis and Pryer † and the wide stencil, monotone finite difference schemes by Froese and Oberman ††.

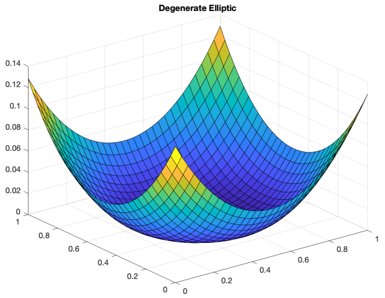
Lakkis, Pryer (2011), *A finite element method for second order nonvariational elliptic problems*, SJSC.

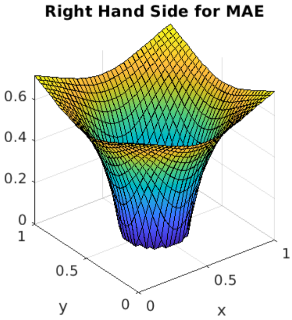
Froese, Oberman (2013), *Convergent filtered schemes for the Monge-Ampère partial differential equation*, SJNA.

Results - 2D Degenerate Elliptic

Exact solution: $u(\mathbf{x}) = \frac{1}{2}(\max(|\mathbf{x} - \mathbf{x}_0| - 0.2, 0))^2$, $\mathbf{x}_0 = (0.5, 0.5)$.

Forcing term: $f(\mathbf{x}) = \max\left(1 - \frac{0.2}{|\mathbf{x}-\mathbf{x}_0|}, 0\right)$.





Results - 2D Degenerate Elliptic

Grid Size	$\|u - u_e\|_2^2$	Rate	$\|u - u_e\|_\infty$	NNZ Max Error†	FO Max Error††
$2^5 \times 2^5$	2.18e-04	-	4.22e-04	5.4e-04	3.73e-04
$2^6 \times 2^6$	6.04e-05	1.8516	1.39e-04	2.8e-04	1.51e-04
$2^7 \times 2^7$	2.00e-05	1.5915	5.51e-05	1.5e-4	9.2e-05
$2^8 \times 2^8$	6.48e-06	1.6282	2.22e-05	7.8e-05	3.8e-05

Compared to two-scale (unstructured wide scale) finite element method by Nochetto et al. † and the wide stencil, montone finite differences schemes by Frose and Oberman ††.

Nochetto, Ntogkas, Zhang (2019), *Two-Scale method for the Monge-Ampère equation: convergence to the viscosity solution*. Mathematics of Computation.

Froese, Oberman (2013), *Convergent filtered schemes for the Monge-Ampère partial differential equation*, SJNA.

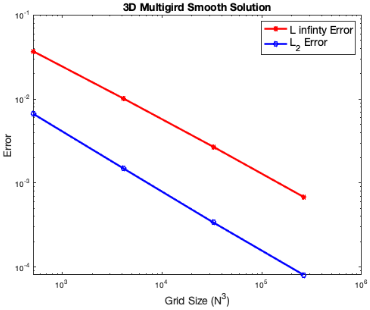
Results - 3D Smooth Solution

$$f = (1 + x^2 + y^2 + z^2)\exp\left(\frac{3(x^2 + y^2 + z^2)}{2}\right)$$

$$u = \exp\left(\frac{x^2 + y^2 + z^2}{2}\right)$$

$$\Omega = (0, 1)^3$$

Taken from: **Liu et al. (2016)**, *A multigrid scheme for 3D Monge-Ampère equations*, IJCM.



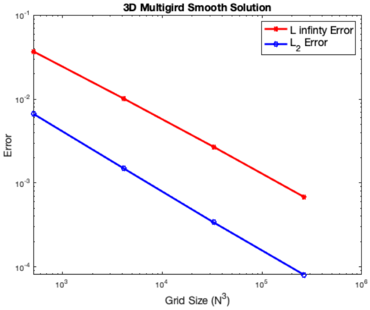
Convergence results for the 3D smooth problem.

Grid Size N	$\|u - u_e\|_2^2$	Rate	$\|u - u_e\|_\infty$	Rate	Max GMRES
8	6.70e-03	-	3.72e-02	-	3
16	1.50e-03	2.16	1.02e-02	1.87	3
32	3.39e-04	2.15	2.70e-03	1.89	3
64	8.06e-05	2.07	6.84e-04	1.92	5

Results - 3D Smooth Solution

$$f = (\sin(x)+1)(\sin(y)+1)(\sin(z)+1)$$
$$u = -\sin(x) - \sin(y) - \sin(z) + (x^2+y^2+z^2)/2$$
$$\Omega = (0,\pi)^3$$

Taken from: **Liu et al. (2016)**, *A multigrid scheme for 3D Monge-Ampère equations*, IJCM.



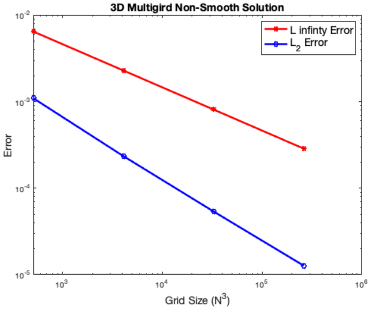
Convergence results for the 3D smooth problem.

Grid Size N	$\|u-u_e\|_2^2$	Rate	$\|u-u_e\|_\infty$	Rate	Max GMRES
8	2.09e-01	-	7.12e-02	-	4
16	4.84e-02	2.11	1.82e-02	1.97	4
32	1.16e-02	2.06	4.60e-03	1.98	3
64					

Results - 3D Non-smooth Solution

$$f = (1/16)(x^2 + y^2 + z^2)^{-3/4}$$
$$u = (1/3)(x^2 + y^2 + z^2)^{3/4}$$
$$\Omega = (0,1)^3$$

Taken from: **Liu et al. (2016)**, *A multigrid scheme for 3D Monge-Ampère equations*, IJCM.



Convergence results for the 3D non-smooth problem.

Grid Size N	$\|u - u_e\|_2^2$	Rate	$\|u - u_e\|_\infty$	Rate	Max GMRES
8	1.10e-03	-	6.50e-03	-	3
16	2.35e-04	1.50	2.30e-03	2.23	4
32	5.38e-05	1.50	8.13e-04	2.13	5
64	1.27e-05	1.50	1.27e-05	2.08	6

Conclusion

- Presented an optimization-based solver for any standard low-order finite element discretization of nonlinear elliptic problems.
- Our optimization-based algorithm combined with a $\mathbb{P}^1$ discretization demonstrates good results with solid numerical error and convergence rates.
- While the method does not directly impose convexity like other discretizations for the Monge-Ampère equation, we have reason to believe that it is provably convergent to a solution of the MAE.
- Future work will further leverage the built in machinery for inexact methods that can offer additional speed ups for the methods.
- Potential to build in convexity constraints or other boundary conditions (such as the optimal transport condition) directly into the objective function of the optimization solver.

Temporary page!

LaTeX was unable to guess the total number of pages correctly. As there was some unproces
data that should have been added to the final page this extra page has been added to receiv
If you rerun the document (without altering it) this surplus page will go away, because LaTe
now knows how many pages to expect for this document.