Smoothness in nonsmooth optimization

(Newtonian ideas for partly smooth equations)

Adrian Lewis

Joint work with: D. Drusvyatskiy, X.Y. Han, A. Ioffe, J. Liang, M.L. Overton, T. Tian, C. Wylie

September 2020

ORIE Cornell One World Optimization Seminar
**Question 1:** Why is smoothness often inherent in nonsmooth optimization and equations?
Introduction: three questions

**Question 1:** Why is smoothness often inherent in nonsmooth optimization and equations?

**Question 2:** Can Newton methods use this smoothness?

**Question 3:** Superlinear convergence for black box nonsmooth optimization?

Example: Eigenvalue optimization.
Introduction: three questions

**Question 1:** Why is smoothness often inherent in nonsmooth optimization and equations?

**Question 2:** Can Newton methods use this smoothness?

**Example:** $\min_Q f$ becomes

$-\nabla f(x) \in N_Q(x)$.

Projected gradient methods

$x \leftarrow \text{Proj}_Q (x - \gamma \nabla f(x))$

identify smoothness in $Q$.

**Newtonian acceleration?**
Question 1: Why is smoothness often inherent in nonsmooth optimization and equations?

Question 2: Can Newton methods use this smoothness?

Example: \( \min_Q f \) becomes
\[-\nabla f(x) \in N_Q(x).\]
Projected gradient methods
\[x \leftarrow \text{Proj}_Q \left( x - \gamma \nabla f(x) \right) \]
identify smoothness in \( Q \).
Newtonian acceleration?

Question 3: Superlinear convergence for black box nonsmooth optimization?
**Question 1:** Why is smoothness often inherent in nonsmooth optimization and equations?

**Question 2:** Can Newton methods use this smoothness?

**Example:** $\min_Q f$ becomes $-\nabla f(x) \in N_Q(x)$. Projected gradient methods $x \leftarrow \text{Proj}_Q (x - \gamma \nabla f(x))$ identify smoothness in $Q$. Newtonian acceleration?

**Question 3:** Superlinear convergence for black box nonsmooth optimization?

**Example:** Eigenvalue optimization.

[Graph showing black box evaluations for Bundle Newton, Bundle Method, and BFGS methods]

1/24
Inherent structure: an example

The numerical radius of an $n$-by-$n$ complex matrix $A$, $\rho(A) = \max_{\|u\|_2 = 1} |u^*Au|$, satisfies the "power inequality" (Berger '65): for $k = 1, 2, ...$

$$\|A^k\|_2 \leq \rho(A^k) \leq \left(\rho(A)\right)^k,$$

and so controls transient stability in dynamics $x \leftarrow Ax$.

Optimizing $\rho$ often results in unusual matrices. Example: For random matrices $Y$, proximal matrices $A$ minimizing $\rho(A) + \lambda \|A - Y\|_2$ often have disc fields of values $\{u^*Au: \|u\|_2 = 1\}$.
The **numerical radius** of an \( n \)-by-\( n \) complex matrix \( A \),

\[
\rho(A) = \max_{\|u\| = 1} |u^* Au|,
\]

satisfies the “power inequality” (Berger ’65): for \( k = 1, 2, \ldots \)

\[
\frac{1}{2} \|A^k\|_2 \leq \rho(A^k) \leq (\rho(A))^k,
\]

and so controls transient stability in dynamics \( x \leftarrow Ax \).
The numerical radius of an \( n \)-by-\( n \) complex matrix \( A \),

\[
\rho(A) = \max_{\|u\|=1} |u^*Au|,
\]
satisfies the “power inequality” (Berger '65): for \( k = 1, 2, \ldots \)

\[
\frac{1}{2} \|A^k\|_2 \leq \rho(A^k) \leq (\rho(A))^k,
\]
and so controls transient stability in dynamics \( x \leftarrow Ax \).

Optimizing \( \rho \) often results in unusual matrices...
The **numerical radius** of an $n$-by-$n$ complex matrix $A$,

$$
\rho(A) = \max_{\|u\| = 1} |u^* Au|,
$$

satisfies the “power inequality” (Berger ’65): for $k = 1, 2, \ldots$

$$
\frac{1}{2} \|A^k\|_2 \leq \rho(A^k) \leq (\rho(A))^k,
$$

and so controls transient stability in dynamics $x \leftarrow Ax$.

**Optimizing** $\rho$ often results in unusual matrices.

**Example:** For random matrices $Y$, **proximal** matrices $A$ minimizing

$$
\rho(A) + \lambda \|A - Y\|^2
$$

often have **disc fields of values** $\{u^* Au : \|u\| = 1\}$. 
Random proximal points (via \texttt{cvx}) are often disk matrices.
Random proximal points (via \texttt{cvx}) are often disk matrices

\[ 1 - \frac{\text{inner radius}}{\rho(A)} \]

(via \texttt{chebfun})

algebraic deviation
from disk

distance to
singularity

distance to null
Jordan block
Random proximal points (via cvx) are often disk matrices

\[ 1 - \frac{\text{inner radius}}{\rho(A)} \] (via chebfun)

algebraic deviation from disk

distance to singularity

distance to null

Jordan block

\[ \sigma_{\min}(\hat{A}) \sigma_{n-1}(\hat{A}^2) \max\{1-\lambda_{\max}(\hat{Z}), \lambda_{\min}(\hat{Z})-1\} \]

\[ 1-\lambda_{n+1}(\Phi \hat{A}(\hat{Z})) \]

\[ \sigma_{\min}(\sigma_{\max}(p_1 p_2 \cdots p_n)) \min_{w \in T} \{(\lambda_1-\lambda_2)(w^* \hat{A} + w \hat{A}^*)\} \]
Random proximal points (via \texttt{cvx}) are often disk matrices

\begin{align*}
1 - \frac{\text{inner radius}}{\rho(A)} \\
\text{(via chebfun)}
\end{align*}

algebraic deviation from disk

distance to singularity

distance to null Jordan block

\textbf{Why...?} (Disk matrices comprise a small set, of codimension $2n$).
Nonsmooth optimization usually involves structured sets. Random instances are solved on manifolds of solutions under perturbation.

\[ \{ g_i(x) \leq 0 \} \]  
relative to active set \( \{ g_j(x) = 0 \} \)

\[ \text{PSD matrices } S^n_+ \]  
relative to \( \{ X \in S^n_+ : \text{rank}(X) = k \} \)

\[ \text{smooth } f^+ ||\cdot||_1 \]  
relative to fixed sparsity pattern

\[ \text{numerical radius } \rho \]  
relative to disk matrices (L-Overton '20, Han-L '20)

History: generalized "active constraints" in nonlinear programming (Burke-More '88), "identifiable surfaces" (Wright '93), "VU decomposition" (Mifflin-Sagastizabal '00)
Nonsmooth optimization usually involves structured sets. Random instances are solved on manifolds of solutions under perturbation.

- \{\text{smooth } g_i(x) \leq 0\} \text{ relative to active set } \{x : g_j(x) = 0\}
Nonsmooth optimization usually involves structured sets. Random instances are solved on manifolds of solutions under perturbation.

- \{\text{smooth } g_i(x) \leq 0\} \text{ relative to active set } \{x : g_j(x) = 0\}
- \text{PSD matrices } S_+^n \text{ relative to } \{X \in S_+^n : \text{rank}(X) = k\}
Nonsmooth optimization usually involves structured sets. Random instances are solved on manifolds of solutions under perturbation.

- \{\text{smooth} \ g_i(x) \leq 0\} \text{ relative to active set} \ \{x : g_j(x) = 0\}
- \text{PSD matrices} \ S^n_+ \text{ relative to} \ \{X \in S^n_+ : \text{rank}(X) = k\}
- \text{smooth} \ f + || \cdot ||_1 \text{ relative to fixed sparsity pattern}
Nonsmooth optimization usually involves structured sets. Random instances are solved on manifolds of solutions under perturbation.

- \{\text{smooth } g_i(x) \leq 0\} \text{ relative to active set } \{x : g_j(x) = 0\}
- PSD matrices $S_n^+$ relative to \{\(X \in S_n^+ : \text{rank}(X) = k\)\}
- smooth $f + \| \cdot \|_1$ relative to fixed sparsity pattern
- numerical radius $\rho$ relative to disk matrices

(L-Overton '20, Han-L '20)
Nonsmooth optimization usually involves structured sets. Random instances are solved on manifolds of solutions under perturbation.

- \{\text{smooth } g_i(x) \leq 0\} \text{ relative to active set } \{x : g_j(x) = 0\}
- PSD matrices \( S^n_+ \) relative to \( \{X \in S^n_+ : \text{rank}(X) = k\} \)
- smooth \( f + \| \cdot \|_1 \) relative to fixed sparsity pattern
- numerical radius \( \rho \) relative to disk matrices

(L-Overton ’20, Han-L ’20)

History: generalized “active constraints” in nonlinear programming (Burke-Moré ’88), “identifiable surfaces” (Wright ’93), “\( \mathcal{VU} \) decomposition” (Mifflin-Sagastizábal ’00)…
Partial smoothness and algorithms

Diverse first-order methods identify the manifold (L-Hare '04...), which drives the local convergence.

Example: $\ell_1$ regularization for sparsity. Proximal gradient iterates settle on a sparsity pattern (Hale-Yin-Zhang '08).

$$\min_x f(x) = h(x) + \lambda \|x\|_*$$

Example: Nuclear norm regularization for low-rank optimization. Proximal gradient (singular value thresholding) iterates settle on a fixed-rank manifold, then converge linearly to the solution. (Liang-Fadili-Peyré '18)
Diverse first-order methods identify the manifold (L-Hare ’04...), which drives the local convergence.
Partial smoothness and algorithms

Diverse first-order methods identify the manifold (L-Hare '04...), which drives the local convergence.

**Example:** $l_1$ regularization for sparsity. Proximal gradient iterates settle on a sparsity pattern (Hale-Yin-Zhang '08).
Diverse first-order methods identify the manifold (L-Hare '04...), which drives the local convergence.

**Example:** $l_1$ regularization for sparsity. Proximal gradient iterates settle on a sparsity pattern (Hale-Yin-Zhang '08).

$$\min_x f(x) = h(x) + \lambda \|x\|_*$$
Partial smoothness and algorithms

Diverse first-order methods identify the manifold (L-Hare ’04...), which drives the local convergence.

**Example:** $l_1$ regularization for sparsity. Proximal gradient iterates settle on a sparsity pattern (Hale-Yin-Zhang ’08).

\[
\min_x f(x) = h(x) + \lambda \|x\|_*
\]

**Example:** Nuclear norm regularization for low-rank optimization. Proximal gradient (singular value thresholding) iterates settle on a fixed-rank manifold, then converge linearly to the solution.

(Liang-Fadili-Peyré ’18)
Generalized equations

Shift focus from optimization to optimality conditions: $x$ minimizes $f \Rightarrow 0 \in \partial f(x)$.

Generalize: $0 \in \Phi(u)$ for set-valued operator $\Phi$ on $\mathbb{R}^n$.

• Variational inequalities
  
  Find $x \in Q$ so $F(x)^T(z - x) \geq 0$ for all $z \in Q$: equivalently, $0 \in F(x) + \mathcal{N}Q(x)$.

• Composite optimization
  
  $\min x h(c(x))$ for convex $h$ on $\mathbb{R}^m$ and smooth $c$ into $\mathbb{R}^m$.

  Stationarity: $0 \in (\nabla c(x)^T y - y) + \partial h(c(x))$. 
Generalized equations

Shift focus from optimization to optimality conditions:

\[ x \text{ minimizes } f \implies 0 \in \partial f(x). \]
Shift focus from optimization to optimality conditions:

\[ x \text{ minimizes } f \Rightarrow 0 \in \partial f(x). \]

Generalize: \( 0 \in \Phi(u) \) for set-valued operator \( \Phi \) on \( \mathbb{R}^n \).
Generalized equations

Shift focus from optimization to optimality conditions:

\[ x \text{ minimizes } f \implies 0 \in \partial f(x). \]

Generalize: \( 0 \in \Phi(u) \) for set-valued operator \( \Phi \) on \( \mathbb{R}^n \).

- **Variational inequalities**
  
  Find \( x \in Q \) so \( F(x)^T(z - x) \geq 0 \) for all \( z \in Q \): equivalently,
  
  \[ 0 \in F(x) + N_Q(x). \]
Shift focus from optimization to optimality conditions:

\[ x \text{ minimizes } f \implies 0 \in \partial f(x). \]

Generalize: \( 0 \in \Phi(u) \) for set-valued operator \( \Phi \) on \( \mathbb{R}^n \).

- **Variational inequalities**
  
  Find \( x \in Q \) so \( F(x)^T(z - x) \geq 0 \) for all \( z \in Q \): equivalently,
  
  \[ 0 \in F(x) + N_Q(x). \]

- **Composite optimization**
  
  \( \min_x h(c(x)) \) for convex \( h \) on \( \mathbb{R}^m \) and smooth \( c \) into \( \mathbb{R}^m \).
  
  Stationarity:
  
  \[ 0 \in \begin{pmatrix} \nabla c(x)^T y \\ -y \end{pmatrix} + \begin{pmatrix} 0 \\ \partial h(c(x)) \end{pmatrix} \]
A set-valued operator Φ is \textit{partly smooth} at a solution \( \overline{u} \) for given data \( \overline{v} \).
A set-valued operator $\Phi$ is partly smooth at a solution $\bar{u}$ for given data $\bar{v}$ if

- $\text{gph } \Phi = \{(u, v) : v \in \Phi(u)\}$ is a manifold around $(\bar{u}, \bar{v})$.
- $\text{proj} : \text{gph } \Phi \rightarrow \mathbb{R}^n : (u, v) \mapsto u$ is constant rank around $(\bar{u}, \bar{v})$... (i.e. the projection of the graph’s tangent space has locally constant dimension).
A set-valued operator $\Phi$ is partly smooth at a solution $\bar{u}$ for given data $\bar{v}$ if

- $\text{gph } \Phi = \{(u, v) : v \in \Phi(u)\}$ is a manifold around $(\bar{u}, \bar{v})$.
- $\text{proj} : \text{gph } \Phi \to \mathbb{R}^n : (u, v) \mapsto u$ is constant rank around $(\bar{u}, \bar{v})$...
  (i.e. the projection of the graph’s tangent space has locally constant dimension).

Asymptotic solvers then identify the active manifold

$$\mathcal{M} = \text{proj}(\text{gph } \Phi \text{ around } (\bar{u}, \bar{v})), $$

since $v_k \in \Phi(u_k)$ with $(u_k, v_k) \to (\bar{u}, \bar{v})$ implies $u_k \in \mathcal{M}$ eventually.
Basic example: partly smooth sets

For closed convex (or “prox-regular”) \( S \subset \mathbb{R}^n \), suppose \( \bar{x} \) solves

\[
\min_{x \in S} \langle \bar{y}, x \rangle \quad \text{and hence} \quad \bar{y} \in N_S(x).
\]
Basic example: partly smooth sets

For closed convex (or “prox-regular”) $S \subset \mathbb{R}^n$, suppose $\bar{x}$ solves
\[
\min_{x \in S} \langle \bar{y}, x \rangle \quad \text{and hence} \quad \bar{y} \in N_S(x).
\]

If $S$ contains a ridge manifold $M$ (the normal cone $N_S(x)$ depends on $x \in M$ continuously and spans $N_M(x)$), and nondegeneracy holds ($\bar{y} \in \text{ri}(N_S(\bar{x}))$), then the operator $N_S$ is partly smooth at $\bar{x}$ for $\bar{y}$, with active manifold $M$. 
Basic example: partly smooth sets

For closed convex (or “prox-regular”) $S \subset \mathbb{R}^n$, suppose $\bar{x}$ solves

$$\min_{x \in S} \langle \bar{y}, x \rangle$$

and hence $\bar{y} \in N_S(\bar{x})$.

If $S$ contains a ridge manifold $\mathcal{M}$ (the normal cone $N_S(x)$ depends on $x \in \mathcal{M}$ continuously and spans $N_\mathcal{M}(x)$), and nondegeneracy holds ($\bar{y} \in \text{ri}(N_S(\bar{x}))$), then the operator $N_S$ is partly smooth at $\bar{x}$ for $\bar{y}$, with active manifold $\mathcal{M}$.

So if $S$ is convex and $\bar{x}$ is unique, projected gradient iterations

$$x \leftarrow \text{proj}_Q(x - \alpha \bar{y})$$

converge to $\bar{x}$ (if $\alpha$ small) and identify $\mathcal{M}$. 
Example: max functions of degree $k$

Given a decomposition

$$f(x) = \max_{i=1}^{k} f_i(x),$$

using smooth components $f_i$, call $\bar{x}$ a strictly active critical point when the values $f_i(\bar{x})$ are all equal, and the system

$$\sum_i \lambda_i = 1, \quad \sum_i \lambda_i \nabla f_i(\bar{x}) = 0$$

has a unique solution, which furthermore has each $\lambda_i > 0$.

Then $x \mapsto \partial f(x) = \text{conv}\{\nabla f_i(x) : f_i(x) = f(x)\}$ is partly smooth at $\bar{x}$ for 0 relative to the active manifold $M$ of points where each $f_i$ has equal value.
Example: max functions of degree $k$

Given a decomposition

$$f(x) = \max_{i=1,...,k} f_i(x),$$

using smooth components $f_i$, 

...
Given a decomposition

\[ f(x) = \max_{i=1,...,k} f_i(x), \]

using smooth components \( f_i \), call \( \bar{x} \) a **strictly active** critical point when the values \( f_i(\bar{x}) \) are all equal, and the system

\[
\sum_i \lambda_i = 1, \quad \sum_i \lambda_i \nabla f_i(\bar{x}) = 0
\]

has a unique solution, which furthermore has each \( \lambda_i > 0 \).
Example: max functions of degree $k$

Given a decomposition

$$f(x) = \max_{i=1,\ldots,k} f_i(x),$$

using smooth components $f_i$, call $\bar{x}$ a strictly active critical point when the values $f_i(\bar{x})$ are all equal, and the system

$$\sum_i \lambda_i = 1, \quad \sum_i \lambda_i \nabla f_i(\bar{x}) = 0$$

has a unique solution, which furthermore has each $\lambda_i > 0$. Then

$$x \mapsto \partial f(x) = \operatorname{conv}\{\nabla f_i(x) : f_i(x) = f(x)\}$$

is partly smooth at $\bar{x}$ for 0 relative to the active manifold $\mathcal{M}$ of points where each $f_i$ has equal value.
Sard-type behavior: partial smoothness is common

Sard's Theorem: almost no values of smooth operators are critical.

What about set-valued operators and generalized equations on $\mathbb{R}^n$?

Consider a semi-algebraic operator $\Phi$ with $n$-dimensional graph:

$$\text{gph } \Phi = \bigcup_{i=1}^{q} \bigcap_{j=1}^{r} \{ (x, y) \in \mathbb{R}^{2n} : p_{ij}(x, y) \leq \text{or} < 0 \}$$

for polynomials $p_{ij}$.

Eg: Subdifferentials, monotone operators. . .

Theorem:

Around generic data $y$, there are smooth maps $G_i$ so $\Phi^{-1} = \{ G_1, \ldots, G_k \}$ (possibly empty).

$\Phi$ is partly smooth for $y$ at each solution $x_i = G_i(y)$, and $\text{gph } \Phi$ intersects ($\mathbb{R}^n \times \{ y \}$) transversally at $(x_i, y)$.

(Ioffe '07, Bolte. . . '11, Drusvyatskiy. . . '16, Lee. . . '19, L-Tian)
Sard-type behavior: partial smoothness is common

Sard’s Theorem: almost no values of smooth operators are critical.
Sard-type behavior: partial smoothness is common

Sard’s Theorem: almost no values of smooth operators are critical.
What about set-valued operators and generalized equations on $\mathbb{R}^n$?
Sard-type behavior: partial smoothness is common

Sard’s Theorem: almost no values of smooth operators are critical. What about set-valued operators and generalized equations on $\mathbb{R}^n$?

Consider a semi-algebraic operator $\Phi$ with $n$-dimensional graph:

$$\text{gph } \Phi = \bigcup_{i=1}^{q} \bigcap_{j=1}^{r} \left\{ (x, y) \in \mathbb{R}^{2n} : p_{ij}(x, y) \left\{ \begin{array}{c} \leq \nonumber \\ \text{or} \nonumber \\ < \end{array} \right\} 0 \right\}$$

for polynomials $p_{ij}$.
Sard-type behavior: partial smoothness is common

Sard’s Theorem: almost no values of smooth operators are critical. What about set-valued operators and generalized equations on $\mathbb{R}^n$?

Consider a semi-algebraic operator $\Phi$ with $n$-dimensional graph:

$$\text{gph } \Phi = \bigcup_{i=1}^{q} \bigcap_{j=1}^{r} \left\{ (x, y) \in \mathbb{R}^{2n} : p_{ij}(x, y) \begin{cases} \leq \\ \text{or} \\ < \end{cases} 0 \right\}$$

for polynomials $p_{ij}$. Eg: Subdifferentials, monotone operators...
Sard-type behavior: partial smoothness is common

Sard’s Theorem: almost no values of smooth operators are critical. What about set-valued operators and generalized equations on $\mathbb{R}^n$?

Consider a semi-algebraic operator $\Phi$ with $n$-dimensional graph:

$$
gph \Phi = \bigcup_{i=1}^{q} \bigcap_{j=1}^{r} \{(x, y) \in \mathbb{R}^{2n} : p_{ij}(x, y) \left\{ \leq \right\} 0 \}$$

for polynomials $p_{ij}$. Eg: Subdifferentials, monotone operators . . .

**Theorem**  Around generic data $y$, there are smooth maps $G_i$ so

$$
\Phi^{-1} = \{ G_1, \ldots, G_k \} \quad (\text{possibly empty}).
$$
Sard-type behavior: partial smoothness is common

Sard’s Theorem: almost no values of smooth operators are critical. What about set-valued operators and generalized equations on $\mathbb{R}^n$?

Consider a semi-algebraic operator $\Phi$ with $n$-dimensional graph:

$$\text{gph } \Phi = \bigcup_{i=1}^{q} \bigcap_{j=1}^{r} \{ (x, y) \in \mathbb{R}^{2n} : p_{ij}(x, y) \begin{cases} \leq \\ < \end{cases} 0 \}$$

for polynomials $p_{ij}$. Eg: Subdifferentials, monotone operators...

**Theorem** Around generic data $y$, there are smooth maps $G_i$ so

$$\Phi^{-1} = \{ G_1, \ldots, G_k \} \text{ (possibly empty).}$$

$\Phi$ is partly smooth for $y$ at each solution $x_i = G_i(y)$,
Sard-type behavior: partial smoothness is common

Sard’s Theorem: almost no values of smooth operators are critical. What about set-valued operators and generalized equations on $\mathbb{R}^n$?

Consider a semi-algebraic operator $\Phi$ with $n$-dimensional graph:

$$\text{gph } \Phi = \bigcup_{i=1}^{q} \bigcap_{j=1}^{r} \left\{ (x, y) \in \mathbb{R}^{2n} : p_{ij}(x, y) \begin{cases} \leq \text{ or } < \\ < \end{cases} 0 \right\}$$

for polynomials $p_{ij}$. Eg: Subdifferentials, monotone operators...

**Theorem**  Around generic data $y$, there are smooth maps $G_i$ so

$$\Phi^{-1} = \{ G_1, \ldots, G_k \} \quad \text{(possibly empty)}.$$

$\Phi$ is partly smooth for $y$ at each solution $x_i = G_i(y)$, and

$$\text{gph } \Phi \text{ intersects } (\mathbb{R}^n \times \{ y \}) \text{ transversally at } (x_i, y).$$

(Ioffe '07, Bolte... ’11, Drusvyatskiy... ’16, Lee... ’19, L-Tian)
Semi-Newton iterations for $0 \in \Phi(u)$ (L-Wylie '20)

Recast as set intersection: find a point $z = (u, 0)$ where $X = \text{gph } \Phi$ intersects $Y = \mathbb{R}^n \times \{0\}$.

Assume transversality: $N_X(z) \cap N_Y(z) = \{0\}$.

Step 1: Linearize $X$; intersect with $Y$.

Step 2: Restore to $X$ via a Lipschitz map fixing $z$.

Linearize around $v \in \Phi(u)$; solve for $u': u' + \text{Proj } M(u') = v + \text{Proj } \Phi(u + 0)(0)$.
Recast as set intersection: find a point \( z = (u, 0) \) where
\[
X = \text{gph } \Phi \quad \text{intersects} \quad Y = \mathbb{R}^n \times \{0\}.
\]
Recast as set intersection: find a point $z = (u, 0)$ where $X = \text{gph } \Phi$ intersects $Y = \mathbb{R}^n \times \{0\}$.

Assume transversality: $N_X(z) \cap N_Y(z) = \{0\}$. 
Recast as set intersection: find a point \( z = (u, 0) \) where
\[
X = \text{gph } \Phi \quad \text{intersects} \quad Y = \mathbb{R}^n \times \{0\}.
\]
Assume transversality: \( N_X(z) \cap N_Y(z) = \{0\} \).

**Step 1:** Linearize \( X \); intersect with \( Y \).
Semi-Newton iterations for $0 \in \Phi(u)$

Recast as set intersection: find a point $z = (u, 0)$ where

$$X = \text{gph } \Phi \quad \text{intersects} \quad Y = \mathbb{R}^n \times \{0\}.$$ 

Assume transversality: $N_X(z) \cap N_Y(z) = \{0\}$.

**Step 1:** Linearize $X$; intersect with $Y$.

**Step 2:** Restore to $X$ via a Lipschitz map fixing $z$. 
Recast as set intersection: find a point $z = (u, 0)$ where

$$X = \text{gph} \Phi \quad \text{intersects} \quad Y = \mathbb{R}^n \times \{0\}.$$

Assume transversality: $N_X(z) \cap N_Y(z) = \{0\}$.

**Step 1:** Linearize $X$; intersect with $Y$.

Linearize around $v \in \Phi(u)$; solve for $u'$

**Step 2:** Restore to $X$ via a Lipschitz map fixing $z$. 
Semi-Newton iterations for $0 \in \Phi(u)$ (L-Wylie ’20)

Recast as set intersection: find a point $z = (u, 0)$ where

$$X = \text{gph } \Phi \text{ intersects } Y = \mathbb{R}^n \times \{0\}.$$ 

Assume transversality: $N_X(z) \cap N_Y(z) = \{0\}$.

**Step 1:** Linearize $X$; intersect with $Y$.

Linearize around $v \in \Phi(u)$; solve for $u'$

**Step 2:** Restore to $X$ via a Lipschitz map fixing $z$.

$$u^+ = \text{Proj}_{\mathcal{M}}(u'); \quad v^+ = \text{Proj}_{\Phi(u^+)}(0)$$
Newtonian methods for partly smooth optimization $0 \in \partial f(x)$ are interesting, but typically need \textit{structural} knowledge of $\partial f$. 
Newtonian methods for partly smooth optimization $0 \in \partial f(x)$ are interesting, but typically need *structural* knowledge of $\partial f$.

Classical special case: sequential quadratic programming.
Newtonian methods for partly smooth optimization $0 \in \partial f(x)$ are interesting, but typically need \textit{structural} knowledge of $\partial f$.

Classical special case: sequential quadratic programming.

More generally, semismooth Newton methods: Klatte-Kummer '02, Facchinei-Pang '03, Izmailov-Solodov '14, Gfrerer-Outrata '19.
Newtonian methods for partly smooth optimization $0 \in \partial f(x)$ are interesting, but typically need structural knowledge of $\partial f$.

Classical special case: sequential quadratic programming.

More generally, semismooth Newton methods: Klatte-Kummer '02, Facchinei-Pang '03, Izmailov-Solodov '14, Gfrerer-Outrata '19.

With just an oracle for linear approximations to convex $f$ at input points, bundle methods are appealing (Sagastizábal '18 ICM).
Newtonian methods for partly smooth optimization $0 \in \partial f(x)$ are interesting, but typically need structural knowledge of $\partial f$.

Classical special case: sequential quadratic programming.

More generally, semismooth Newton methods: Klatte-Kummer ’02, Facchinei-Pang ’03, Izmailov-Solodov ’14, Gfrerer-Outrata ’19.

With just an oracle for linear approximations to convex $f$ at input points, bundle methods are appealing (Sagastizábal ’18 ICM).

- “Null” steps enhance a cutting plane model.
- “Serious” steps sufficiently decrease the objective.
- Partial smoothness (“$\nabla U$”) can accelerate the serious steps.
Newtonian methods for partly smooth optimization \( 0 \in \partial f(x) \) are interesting, but typically need \textit{structural} knowledge of \( \partial f \).

Classical special case: sequential quadratic programming.

More generally, semismooth Newton methods: Klatte-Kummer '02, Facchinei-Pang '03, Izmailov-Solodov '14, Gfrerer-Outrata '19.

With just an \textit{oracle} for \textit{linear} approximations to convex \( f \) at input points, \textit{bundle methods} are appealing (Sagastizábal '18 ICM).

- "Null” steps enhance a cutting plane model.
- "Serious” steps sufficiently decrease the objective.
- Partial smoothness ("\( \nabla U \)") can accelerate the serious steps.

But can we reduce oracle calls using \textit{quadratic} approximations?
Second-order oracles for nonsmooth optimization

Convex $f : \mathbb{R}^n \to \mathbb{R}$ are twice differentiable off a negligible set $N$.

Black-box methods (bundle, BFGS) typically never encounter $N$.

What if an oracle returns $f(x), \nabla f(x), \nabla^2 f(x)$ for $x \not\in N$?

Aim: find a bundle $S$ of $k$ reference points, with small diameter $\max \{ \|s - s'\| : s, s' \in S \}$ and small optimality measure $\text{dist}(0, \text{conv}(\nabla f(S)))$.

Intuition: if the bundle size $k$ is large enough, $\lim_{S \to \{x\}} \text{conv}(\nabla f(S)) = \partial f(x)$. 

13/24
Convex $f : \mathbb{R}^n \rightarrow \mathbb{R}$ are twice differentiable off a negligible set $N$. 
Convex $f : \mathbb{R}^n \to \mathbb{R}$ are twice differentiable off a negligible set $N$. Black-box methods (bundle, BFGS) typically never encounter $N$. 

Aim: find a bundle $S$ of $k$ reference points, with small diameter $\max \{\|s - s'\| : s, s' \in S\}$ and small optimality measure $\text{dist}(0, \text{conv}(\nabla f(S)))$.

Intuition: if the bundle size $k$ is large enough, $\lim_{S \to \{x\}} \text{conv}(\nabla f(S)) = \partial f(x)$. 

13/24
Second-order oracles for nonsmooth optimization

Convex $f : \mathbb{R}^n \rightarrow \mathbb{R}$ are twice differentiable off a negligible set $N$. Black-box methods (bundle, BFGS) typically never encounter $N$. What if an oracle returns $f(x), \nabla f(x), \nabla^2 f(x)$ for $x \notin N$?
Convex $f : \mathbb{R}^n \rightarrow \mathbb{R}$ are twice differentiable off a negligible set $N$. Black-box methods (bundle, BFGS) typically never encounter $N$. What if an oracle returns $f(x), \nabla f(x), \nabla^2 f(x)$ for $x \notin N$?

**Aim:** find a bundle $S$ of $k$ reference points, with small diameter

$$\max \{ \|s - s'\| : s, s' \in S \}$$

and small optimality measure

$$\text{dist} \left( 0, \text{conv} (\nabla f(S)) \right).$$
Convex $f : \mathbb{R}^n \to \mathbb{R}$ are twice differentiable off a negligible set $N$. Black-box methods (bundle, BFGS) typically never encounter $N$. What if an oracle returns $f(x), \nabla f(x), \nabla^2 f(x)$ for $x \not\in N$?

**Aim:** find a bundle $S$ of $k$ reference points, with small diameter

$$\max\{\|s - s'\| : s, s' \in S\}$$

and small optimality measure

$$\text{dist}\left(0, \text{conv}\left(\nabla f(S)\right)\right).$$

Intuition: if the bundle size $k$ is large enough,

$$\lim_{S \to \{x\}} \text{conv}\left(\nabla f(S)\right) = \partial f(x).$$
For each of the $k$ current reference points $s \in S$, use the oracle to form the linear and quadratic approximations:

- Linear approximation: $l_s(x) = f(s) + \nabla f(s)^T(x - s)$
- Quadratic approximation: $q_s(x) = l_s(x) + \frac{1}{2}(x - s)^T \nabla^2 f(s)(x - s)$

- Choose bundle weights $\lambda_s$ solving \[
\min \{ \| \sum_{s \in S} \lambda_s \nabla f(s) \| : \lambda \geq 0, \sum_{s \in S} \lambda_s = 1 \}.
\]
- Choose a new reference point $x$ solving \[
\min \{ \sum_{s \in S} \lambda_s q_s(x) : l_s(s) \text{ equal for all } s \in S \}.
\]
- Replace $s \in S$ minimizing $\| \nabla f(s) - \nabla f(x) \|$ with $x$. 

\[14/24\]
For each of the $k$ current reference points $s \in S$, use the oracle to form the linear and quadratic approximations

$$l_s(x) = f(s) + \nabla f(s)^T(x - s)$$
$$q_s(x) = l_s(x) + \frac{1}{2}(x - s)^T \nabla^2 f(s)(x - s).$$
A $k$-bundle Newton method

For each of the $k$ current reference points $s \in S$, use the oracle to form the linear and quadratic approximations

\[
\begin{align*}
    l_s(x) &= f(s) + \nabla f(s)^T(x - s) \\
    q_s(x) &= l_s(x) + \frac{1}{2}(x - s)^T\nabla^2 f(s)(x - s). 
\end{align*}
\]

- Choose bundle weights $\lambda_s$ solving

\[
\min \left\{ \left\| \sum_{s \in S} \lambda_s \nabla f(s) \right\| : \lambda \geq 0, \sum_s \lambda_s = 1 \right\}.
\]
A $k$-bundle Newton method (L-Wylie ’19)

For each of the $k$ current reference points $s \in S$, use the oracle to form the linear and quadratic approximations

$$l_s(x) = f(s) + \nabla f(s)^T (x - s)$$

$$q_s(x) = l_s(x) + \frac{1}{2} (x - s)^T \nabla^2 f(s)(x - s).$$

- Choose bundle weights $\lambda_s$ solving

$$\min \left\{ \left\| \sum_{s \in S} \lambda_s \nabla f(s) \right\| : \lambda \geq 0, \sum_s \lambda_s = 1 \right\}.$$

- Choose a new reference point $x$ solving

$$\min \left\{ \sum_s \lambda_s q_s(x) : l_s(s) \text{ equal for all } s \in S \right\}.$$
A $k$-bundle Newton method  

For each of the $k$ current reference points $s \in S$, use the oracle to form the linear and quadratic approximations

$$l_s(x) = f(s) + \nabla f(s)^T (x - s)$$

$$q_s(x) = l_s(x) + \frac{1}{2} (x - s)^T \nabla^2 f(s) (x - s).$$

- Choose bundle weights $\lambda_s$ solving

$$\min \left\{ \left\| \sum_{s \in S} \lambda_s \nabla f(s) \right\| : \lambda \geq 0, \sum_s \lambda_s = 1 \right\}.$$

- Choose a new reference point $x$ solving

$$\min \left\{ \sum_s \lambda_s q_s(x) : l_s(s) \text{ equal for all } s \in S \right\}.$$

- Replace $s \in S$ minimizing $\| \nabla f(s) - \nabla f(x) \|$ with $x$.  

(L-Wylie '19)
Motivating the bundle Newton method

Given a current bundle of reference points \( s \in S \), model

\[
f(x) \approx \max_{s \in S} f_s(x):
\]

unknown smooth component \( f_s \) matches \( f \) to 2nd order around \( s \).
Motivating the bundle Newton method

Given a current bundle of reference points $s \in S$, model

$$f(x) \approx \max_{s \in S} f_s(x):$$

unknown smooth component $f_s$ matches $f$ to 2nd order around $s$.

Optimize model via sequential quadratic programming steps on

$$\left\{ \begin{array}{l}
\text{minimize} \quad t \\
\text{subject to} \quad f_s(x) \leq t \quad (s \in S).
\end{array} \right.$$
Motivating the bundle Newton method

Given a current bundle of reference points $s \in S$, model

$$f(x) \approx \max_{s \in S} f_s(x):$$

unknown smooth component $f_s$ matches $f$ to 2nd order around $s$.

Optimize model via sequential quadratic programming steps on

$$\begin{cases} 
\text{minimize} & t \\
\text{subject to} & f_s(x) \leq t \quad (s \in S).
\end{cases}$$

• Estimate multipliers via least squares.
Motivating the bundle Newton method

Given a current bundle of reference points $s \in S$, model

$$f(x) \approx \max_{s \in S} f_s(x) :$$

unknown smooth component $f_s$ matches $f$ to 2nd order around $s$.

Optimize model via sequential quadratic programming steps on

$$\left\{\begin{array}{l}
\text{minimize} \quad t \\
\text{subject to} \quad f_s(x) \leq t \quad (s \in S).
\end{array}\right.$$ 

- Estimate multipliers via least squares.
- Minimize the quadratic approximation of the Lagrangian...
Motivating the bundle Newton method

Given a current bundle of reference points $s \in S$, model

$$f(x) \approx \max_{s \in S} f_s(x) :$$

unknown smooth component $f_s$ matches $f$ to 2nd order around $s$.

Optimize model via sequential quadratic programming steps on

$$\begin{cases} 
\text{minimize} & t \\
\text{subject to} & f_s(x) \leq t \quad (s \in S).
\end{cases}$$

- Estimate multipliers via least squares.
- Minimize the quadratic approximation of the Lagrangian...
- ...subject to the linearized constraints (assumed all active).
Motivating the bundle Newton method

Given a current bundle of reference points $s \in S$, model

$$f(x) \approx \max_{s \in S} f_s(x):$$

unknown smooth component $f_s$ matches $f$ to 2nd order around $s$.

Optimize model via sequential quadratic programming steps on

$$\begin{aligned}
\text{minimize} & \quad t \\
\text{subject to} & \quad f_s(x) \leq t \quad (s \in S).
\end{aligned}$$

- Estimate multipliers via least squares.
- Minimize the quadratic approximation of the Lagrangian.
- ...subject to the linearized constraints (assumed all active).

New point $x$ improves model’s most closely matching component.
Theorem  If $f$ decomposes as max function of degree $k$,

$$f(x) = \max_{i=1,...,k} f_i(x),$$

where each component $f_i$ is smooth

Note: The required bundle size $k$ and the partly smooth geometry of the active manifold $M$ are related:

$$k + \text{dim } M = n + 1.$$
**Theorem**  If $f$ decomposes as max function of degree $k$, 

$$f(x) = \max_{i=1,\ldots,k} f_i(x),$$

where each component $f_i$ is smooth and **strongly convex**
Theorem. If $f$ decomposes as max function of degree $k$,

$$f(x) = \max_{i=1,\ldots,k} f_i(x),$$

where each component $f_i$ is smooth and strongly convex around a strictly active critical point $\bar{x}$,
Theorem  If \( f \) decomposes as max function of degree \( k \),

\[
f(x) = \max_{i=1,\ldots,k} f_i(x),
\]

where each component \( f_i \) is smooth and strongly convex around a strictly active critical point \( \bar{x} \), and initial \( S = \{s_1, \ldots, s_k\} \) is a full bundle near \( \bar{x} \), meaning

\[
f_i(s_i) > f_j(s_i) \quad \text{whenever} \quad i \neq j,
\]
Theorem: If $f$ decomposes as max function of degree $k$,

$$f(x) = \max_{i=1,\ldots,k} f_i(x),$$

where each component $f_i$ is smooth and strongly convex around a strictly active critical point $\bar{x}$, and initial $S = \{s_1, \ldots, s_k\}$ is a full bundle near $\bar{x}$, meaning

$$f_i(s_i) > f_j(s_i) \quad \text{whenever } i \neq j,$$

then $k$-bundle Newton converges $k$-step quadratically to $\bar{x}$. 

Note: The required bundle size $k$ and the partly smooth geometry of the active manifold $M$ are related:

$$k + \dim M = n + 1.$$ 

The classical Newton method has $k = 1$. 

16/24
**Theorem**  If $f$ decomposes as max function of degree $k$,

$$f(x) = \max_{i=1,\ldots,k} f_i(x),$$

where each component $f_i$ is smooth and strongly convex around a strictly active critical point $\bar{x}$, and initial $S = \{s_1,\ldots,s_k\}$ is a full bundle near $\bar{x}$, meaning

$$f_i(s_i) > f_j(s_i) \quad \text{whenever } i \neq j,$$

then $k$-bundle Newton converges $k$-step quadratically to $\bar{x}$.

**Note:** The required bundle size $k$ and the partly smooth geometry of the active manifold $M$ are related:

$$k + \dim M = n + 1.$$
Theorem  If $f$ decomposes as max function of degree $k$,

$$f(x) = \max_{i=1,\ldots,k} f_i(x),$$

where each component $f_i$ is smooth and strongly convex around a strictly active critical point $\bar{x}$, and initial $S = \{s_1,\ldots,s_k\}$ is a full bundle near $\bar{x}$, meaning

$$f_i(s_i) > f_j(s_i) \quad \text{whenever } i \neq j,$$

then $k$-bundle Newton converges $k$-step quadratically to $\bar{x}$.

Note: The required bundle size $k$ and the partly smooth geometry of the active manifold $\mathcal{M}$ are related:

$$k + \dim \mathcal{M} = n + 1.$$

The classical Newton method has $k = 1$. 
A simple max function example

\[ f(x, y) = 2x^2 + y^2 + |x^2 - y| \]
A simple max function example

\[ f(x, y) = 2x^2 + y^2 + |x^2 - y| \]

Objective value against oracle calls.
A simple max function example

\[ f(x, y) = 2x^2 + y^2 + |x^2 - y| \]

Objective value against oracle calls.

- Naive proximal bundle ✪
A simple max function example

\[ f(x, y) = 2x^2 + y^2 + |x^2 - y| \]

Objective value against oracle calls.

- **Naive proximal bundle** *
- **Nonsmooth BFGS** ○
A simple max function example

\[ f(x, y) = 2x^2 + y^2 + |x^2 - y| \]

Objective value against oracle calls.

- Naive proximal bundle *
- Nonsmooth BFGS ○
- 2-bundle Newton
A simple max function example

\[ f(x, y) = 2x^2 + y^2 + |x^2 - y| \]

Objective value against oracle calls.

- Naive proximal bundle *
- Nonsmooth BFGS ○
- 2-bundle Newton
  (initiated from proximal bundle)
A simple max function example

\[ f(x, y) = 2x^2 + y^2 + |x^2 - y| \]

Objective value against oracle calls.

- Naive proximal bundle *
- Nonsmooth BFGS ○
- 2-bundle Newton
  (initiated from proximal bundle)
  - objective X
A simple max function example

\[ f(x, y) = 2x^2 + y^2 + |x^2 - y| \]

Objective value against oracle calls.

- Naive proximal bundle *
- Nonsmooth BFGS ○
- 2-bundle Newton (initiated from proximal bundle)
  - objective X
  - optimality measure \( \triangle \)
A simple max function example

\[ f(x, y) = 2x^2 + y^2 + |x^2 - y| \]

Objective value against oracle calls.

- Naive proximal bundle *
- Nonsmooth BFGS ○
- 2-bundle Newton (initiated from proximal bundle)
  - objective \( X \)
  - optimality measure \( \triangle \)
  - bundle diameter \( \Diamond \)
A simple max function example

\[ f(x, y) = 2x^2 + y^2 + |x^2 - y| \]

Objective value against oracle calls.

- Naive proximal bundle \( \star \)
- Nonsmooth BFGS \( \circ \)
- 2-bundle Newton (initiated from proximal bundle)
  - objective \( \times \)
  - optimality measure \( \triangle \)
  - bundle diameter \( \diamond \)
An eigenvalue optimization example

\[ f(s, t, u, v) = \lambda_{\text{max}} \begin{bmatrix} 0 & s & t \\ s & 1+u & v \\ t & v & 1-u \end{bmatrix} \]
An eigenvalue optimization example

\[ f(s, t, u, v) = \lambda_{\max} \begin{bmatrix} 0 & s & t \\ s & 1+u & v \\ t & v & 1-u \end{bmatrix} - 1 \]

Not a max function around its minimizer, zero.
An eigenvalue optimization example

\[ f(s, t, u, v) = \lambda_{\text{max}} \begin{bmatrix} 0 & s & t \\ s & 1+u & v \\ t & v & 1-u \end{bmatrix} - 1 \]

Not a max function around its minimizer, zero. But…

…partly smooth relative to a 2-dimensional manifold.
An eigenvalue optimization example

\[ f(s, t, u, v) = \lambda_{\text{max}} \begin{bmatrix} 0 & s & t \\ s & 1+u & v \\ t & v & 1-u \end{bmatrix} - 1 \]

Not a max function around its minimizer, zero. But...
...partly smooth relative to a 2-dimensional manifold.

- Proximal bundle *
- BFGS ○
- 3-bundle Newton
  - objective X
  - optimality △
  - diameter ◊
An eigenvalue optimization example

\[ f(s, t, u, v) = \lambda_{\text{max}} \begin{bmatrix} 0 & \frac{s}{1+u} & \frac{t}{v} \\ \frac{s}{t} & 1+u & 1-u \end{bmatrix} - 1 \]

Not a max function around its minimizer, zero. But...

...partly smooth relative to a 2-dimensional manifold.

- Proximal bundle *
- BFGS ∘
- 3-bundle Newton
  - objective X
  - optimality △
  - diameter ◊

![Graph showing convergence of optimization algorithms](image)

<table>
<thead>
<tr>
<th>Value</th>
<th>iterations</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>10⁻⁴</td>
<td>10</td>
</tr>
<tr>
<td>10⁻⁸</td>
<td>20</td>
</tr>
<tr>
<td>10⁻¹²</td>
<td>30</td>
</tr>
</tbody>
</table>

iterations 0 10 20 30
The proof for max functions

Consider a maximum of smooth strongly convex components,

\[ f(x) = \max_{i=1,\ldots,k} f_i(x), \]

with a strictly active critical point \( \bar{x} \) on the active manifold \( \mathcal{M} \).
The proof for max functions

Consider a maximum of smooth strongly convex components,

\[ f(x) = \max_{i=1,\ldots,k} f_i(x), \]

with a strictly active critical point \( \bar{x} \) on the active manifold \( \mathcal{M} \).

Near \( x \) on the active manifold \( \mathcal{M} \), full bundles \( S = \{s_1, \ldots, s_k\} \) (so \( f_i(s_i) > f_j(s_i) \) for \( i \neq j \)) approximate the subdifferential:

\[ \partial f(x) \approx \text{conv} \nabla f(S), \]
The proof for max functions

Consider a maximum of smooth strongly convex components,

\[ f(x) = \max_{i=1,\ldots,k} f_i(x), \]

with a strictly active critical point \( \bar{x} \) on the active manifold \( M \).

Near \( x \) on the active manifold \( M \), full bundles \( S = \{s_1, \ldots, s_k\} \) (so \( f_i(s_i) > f_j(s_i) \) for \( i \neq j \)) approximate the subdifferential:

\[ \partial f(x) \approx \text{conv } \nabla f(S), \]

and the Hessians \( \nabla^2 f(s_i) \) predict curvature on \( M \).
The proof for max functions

Consider a maximum of smooth strongly convex components,

$$f(x) = \max_{i=1,\ldots,k} f_i(x),$$

with a strictly active critical point $\bar{x}$ on the active manifold $\mathcal{M}$.

Near $x$ on the active manifold $\mathcal{M}$, full bundles $S = \{s_1, \ldots, s_k\}$ (so $f_i(s_i) > f_j(s_i)$ for $i \neq j$) approximate the subdifferential:

$$\partial f(x) \approx \text{conv} \nabla f(S),$$

and the Hessians $\nabla^2 f(s_i)$ predict curvature on $\mathcal{M}$.

Partly smooth geometry then ensures $\hat{x} - \bar{x} = O(|\bar{x} - S|^2)$. 
The proof for max functions

Consider a maximum of smooth strongly convex components,

\[ f(x) = \max_{i=1,\ldots,k} f_i(x), \]

with a strictly active critical point \( \bar{x} \) on the active manifold \( \mathcal{M} \).

Near \( x \) on the active manifold \( \mathcal{M} \), full bundles \( S = \{s_1, \ldots, s_k\} \) (so \( f_i(s_i) > f_j(s_i) \) for \( i \neq j \)) approximate the subdifferential:

\[ \partial f(x) \approx \text{conv} \nabla f(S), \]

and the Hessians \( \nabla^2 f(s_i) \) predict curvature on \( \mathcal{M} \).

Partly smooth geometry then ensures \( \hat{x} - \bar{x} = O(|\bar{x} - S|^2) \).

Updating \( s_i \leftarrow \hat{x} \) keeps the bundle full, because \( \bar{x} \) is strictly active.
The proof for max functions

Consider a maximum of smooth strongly convex components,

$$f(x) = \max_{i=1,...,k} f_i(x),$$

with a strictly active critical point $\bar{x}$. on the active manifold $\mathcal{M}$.

Near $x$ on the active manifold $\mathcal{M}$, full bundles $S = \{s_1, \ldots, s_k\}$ (so $f_i(s_i) > f_j(s_i)$ for $i \neq j$) approximate the subdifferential:

$$\partial f(x) \approx \text{conv } \nabla f(S),$$

and the Hessians $\nabla^2 f(s_i)$ predict curvature on $\mathcal{M}$.

Partly smooth geometry then ensures $\hat{x} - \bar{x} = O(|\bar{x} - S|^2)$.

Updating $s_i \leftarrow \hat{x}$ keeps the bundle full, because $\bar{x}$ is strictly active.

Each reference point $s_i$ updates within $k$ steps, by strong convexity.
Finding an initial bundle

Black-box methods for finding a minimizer $\bar{x}$ for nonsmooth $f$, like

- bundle methods (Lemaréchal, Wolfe ’70’s)
- BFGS (L-Overton ’13)
- gradient sampling (Burke-L-Overton ’05)
Finding an initial bundle

Black-box methods for finding a minimizer $\bar{x}$ for nonsmooth $f$, like

- bundle methods (Lemaréchal, Wolfe ’70’s)
- BFGS (L-Overton ’13)
- gradient sampling (Burke-L-Overton ’05)

asymptotically generate subdifferential approximations:

$$\partial f(\bar{x}) \approx \text{conv} (\nabla f(\Omega))$$

for sets $\Omega$ of points near $\bar{x}$. 
Finding an initial bundle

Black-box methods for finding a minimizer $\bar{x}$ for nonsmooth $f$, like

- bundle methods (Lemaréchal, Wolfe ’70’s)
- BFGS (L-Overton ’13)
- gradient sampling (Burke-L-Overton ’05)

asymptotically generate subdifferential approximations:

$$\partial f(\bar{x}) \approx \text{conv} (\nabla f(\Omega))$$

for sets $\Omega$ of points near $\bar{x}$. So, we could choose

$$k = \dim \left( \text{affine span} (\nabla f(\Omega)) \right) \quad \text{(numerically)}$$
Finding an initial bundle

Black-box methods for finding a minimizer $\bar{x}$ for nonsmooth $f$, like

- bundle methods (Lemaréchal, Wolfe ’70’s)
- BFGS (L-Overton ’13)
- gradient sampling (Burke-L-Overton ’05)

asymptotically generate subdifferential approximations:

$$\partial f(\bar{x}) \approx \text{conv}(\nabla f(\Omega))$$

for sets $\Omega$ of points near $\bar{x}$. So, we could choose

$$k = \dim\left(\text{affine span}(\nabla f(\Omega))\right) \quad \text{(numerically)}$$

and initial $S \subset \Omega$ of size $k$ with $\nabla f(S)$ affinely independent.
Example: eigenvalue optimization

For random 25-by-25 symmetric matrices, minimize $\lambda_{\text{max}}$ for

$$A(x) = A_0 + x_1 A_1 + x_2 A_2 + \ldots + x_{50} A_{50}.$$
Example: eigenvalue optimization

For random 25-by-25 symmetric matrices, minimize $\lambda_{\text{max}}$ for

$$A(x) = A_0 + x_1 A_1 + x_2 A_2 + \ldots + x_{50} A_{50}.$$  

Active manifold, where $\lambda_{\text{max}}(A(x))$ has multiplicity 6, has dim 30.
Example: eigenvalue optimization

For random 25-by-25 symmetric matrices, minimize $\lambda_{\text{max}}$ for

$$A(x) = A_0 + x_1A_1 + x_2A_2 + \ldots + x_{50}A_{50}.$$ 

Active manifold, where $\lambda_{\text{max}}(A(x))$ has multiplicity 6, has dim 30.

$f - \min f$

![Graph showing black box evaluations and performance comparison between Bundle Newton, Bundle Method, and BFGS algorithms.](image-url)
Example: eigenvalue optimization

For random 25-by-25 symmetric matrices, minimize $\lambda_{\text{max}}$ for

$$A(x) = A_0 + x_1 A_1 + x_2 A_2 + \ldots + x_{50} A_{50}.$$ 

Active manifold, where $\lambda_{\text{max}}(A(x))$ has multiplicity 6, has dim 30.

![Graphs showing optimization performance](image-url)
Extensions... 

- Avoiding Hessians... using automatic differentiation ✓
- With a linearly convergent first-order analogue ✓
- Extending the local convergence analysis... to nonconvex max functions ✓
- To partly smooth functions ??
- Globalizing the algorithm ??
Extensions...

- Avoiding Hessians...
Extensions.

- Avoiding Hessians.
  - ...using automatic differentiation
Extensions...

- Avoiding Hessians...
  - ...using automatic differentiation ✓
• Avoiding Hessians...
  • ...using automatic differentiation ✓
  • ...with a linearly convergent first-order analogue
• Avoiding Hessians...
  • ...using automatic differentiation ✓
  • ...with a linearly convergent first-order analogue ✓?
Extensions...

- Avoiding Hessians...
  - ...using automatic differentiation ✓
  - ...with a linearly convergent first-order analogue ✓?
- Extending the local convergence analysis...
Extensions...

- Avoiding Hessians...
  - ... using automatic differentiation ✓
  - ... with a linearly convergent first-order analogue ✓?
- Extending the local convergence analysis...
  - ... to nonconvex max functions
Extensions...

- Avoiding Hessians...
  - ...using automatic differentiation ✓
  - ...with a linearly convergent first-order analogue ✓?

- Extending the local convergence analysis...
  - ...to nonconvex max functions ✓
Extensions.

• Avoiding Hessians.
  • ... using automatic differentiation ✓
  • ... with a linearly convergent first-order analogue ✓?

• Extending the local convergence analysis.
  • ... to nonconvex max functions ✓
  • ... to partly smooth functions
Extensions...

- Avoiding Hessians...
  - ...using automatic differentiation ✓
  - ...with a linearly convergent first-order analogue ✓?
- Extending the local convergence analysis...
  - ...to nonconvex max functions ✓
  - ...to partly smooth functions ??
• Avoiding Hessians...
  • ...using automatic differentiation ✓
  • ...with a linearly convergent first-order analogue ✓?
• Extending the local convergence analysis...
  • ...to nonconvex max functions ✓
  • ...to partly smooth functions ??
• Globalizing the algorithm
Extensions...

- Avoiding Hessians...
  - ...using automatic differentiation ✓
  - ...with a linearly convergent first-order analogue ✓?
- Extending the local convergence analysis...
  - ...to nonconvex max functions ✓
  - ...to partly smooth functions ??
- Globalizing the algorithm ??
Partial smoothness is a simple differential-geometric idea that captures the generic interplay between smooth and nonsmooth geometry in concrete variational problems, illuminating the analysis and design of algorithms.
References

L and M.L. Overton, “Partial smoothness of the numerical radius at matrices whose fields of values are disks”, *SIMAX* 2020.

X.Y. Han and L, “Disk matrices and the proximal mapping for the numerical radius”, *arXiv:2004.14542*

L and J. Liang, “Partial smoothness and constant rank”, *arXiv:1807.03134*.

