Convex V-models for nonconvex minimization

Claudia Sagastizábal

mailto:sagastiz@impa.br, http://www.impa.br/~sagastiz

Optimization Day, Grenoble, Nov. 5, 2014

Joint work with R. mifflin

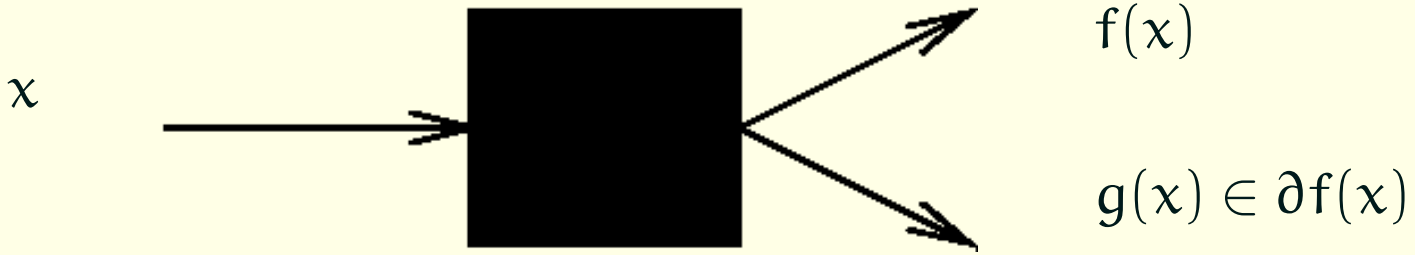
With thanks to:
AFOSR Grant FA9550-08-1-0370, NSF Grant DMS 0707205,
and CNPq & Faperj from Brazil

NSO Algorithms

For a nonsmooth function, solving

$$\min f(x)$$

with a black-box method



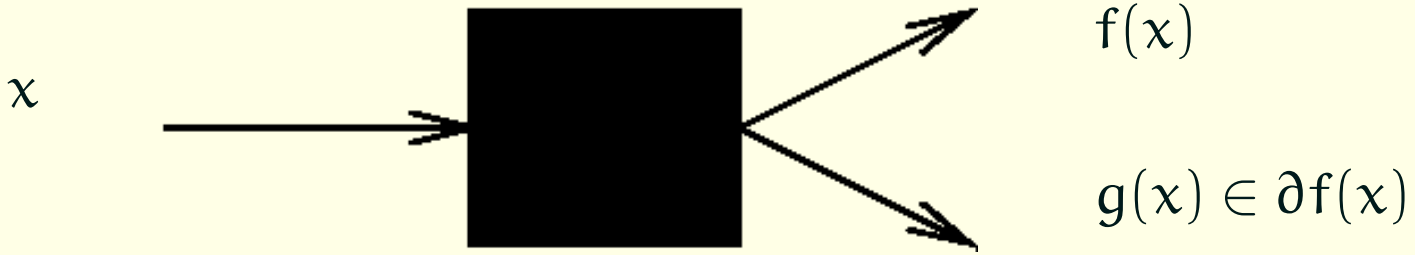
is doomed to slow convergence speed.

NSO Algorithms

For a nonsmooth function, solving

$$\min f(x)$$

with a black-box method



is doomed to slow convergence speed.

Better performance possible by exploiting structure

How does structure appear?

– Implicitly

  U-Lagrangian

  VU-decomposition

  partly smooth functions

– Explicitly

  as a sum

  as a composition

How does structure appear?

– Implicitly

U-Lagrangian

VU-decomposition

partly smooth functions

How does structure appear?

– Implicitly

U-Lagrangian

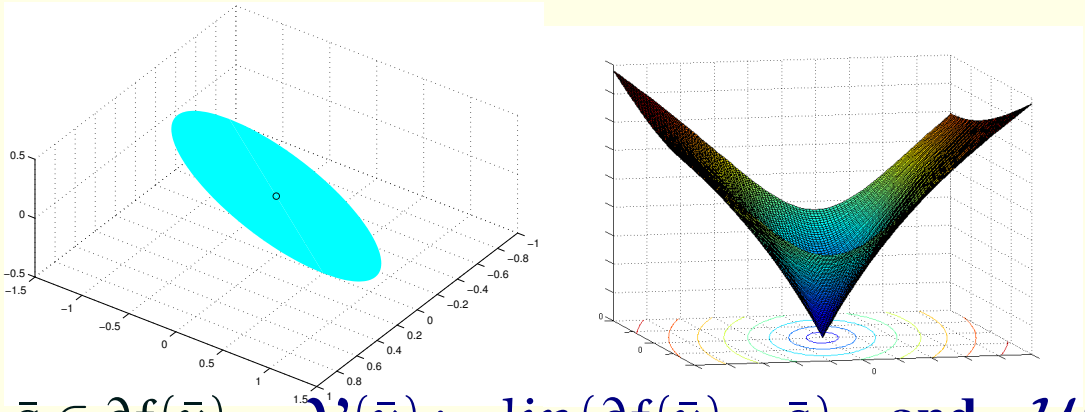
VU-decomposition

partly smooth functions

$$f(x_1,x_2,x_3) = \tfrac{1}{2}x_1^2 + \tfrac{1}{2}\ln\left(1+\sqrt{(x_1^2-2x_2)^2+(x_3-x_2)^2}\right)$$

$x^* = (0,0,0)$ is a stationary point (minimizer)

zero subgradient $\in \partial f(x^*)$

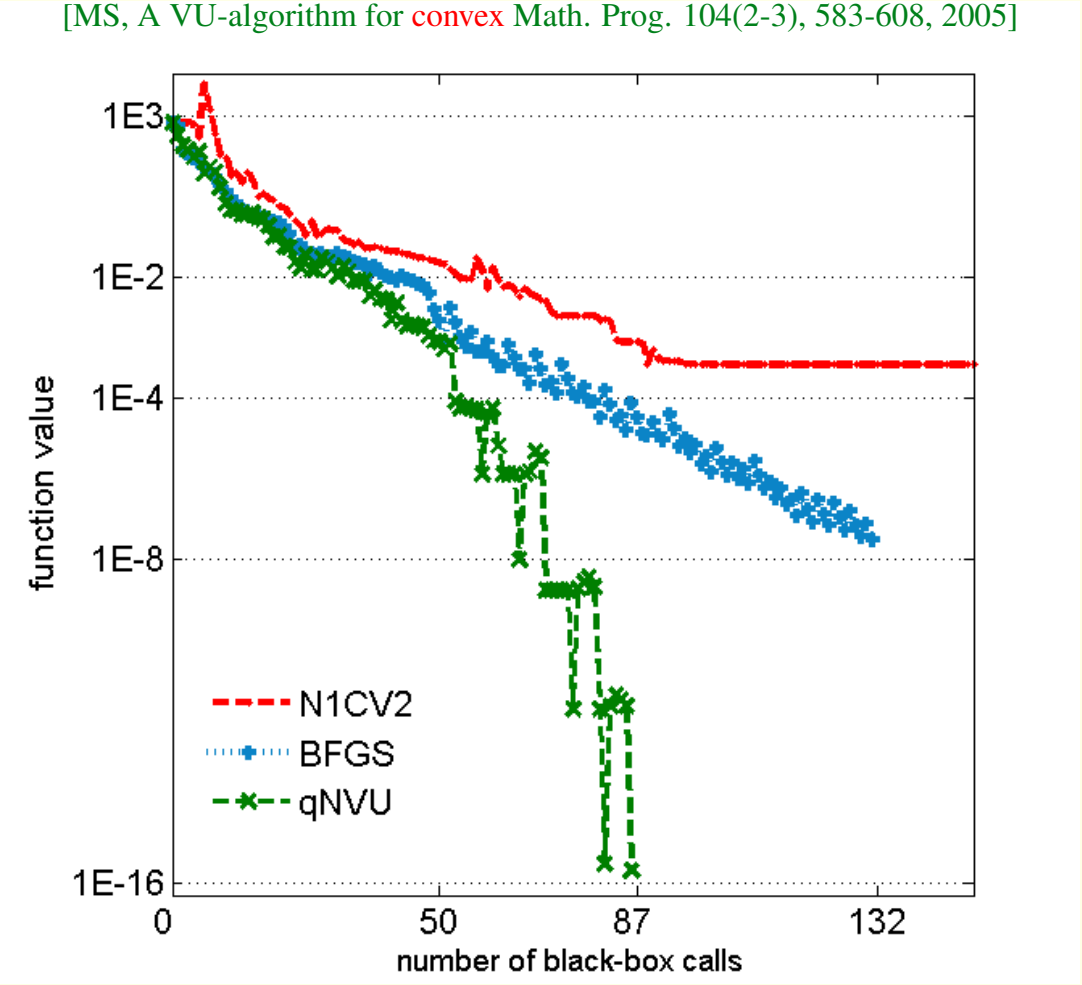


$\bar{g} \in \partial f(\bar{x})$, $\mathcal{V}(\bar{x}) := \mathrm{lin}(\partial f(\bar{x}) - \bar{g})$ **and** $\mathcal{U}(\bar{x}) := \mathcal{V}(\bar{x})^\perp$

CONVEX CASE

Lewis and Overton 8-variable half-and-half function

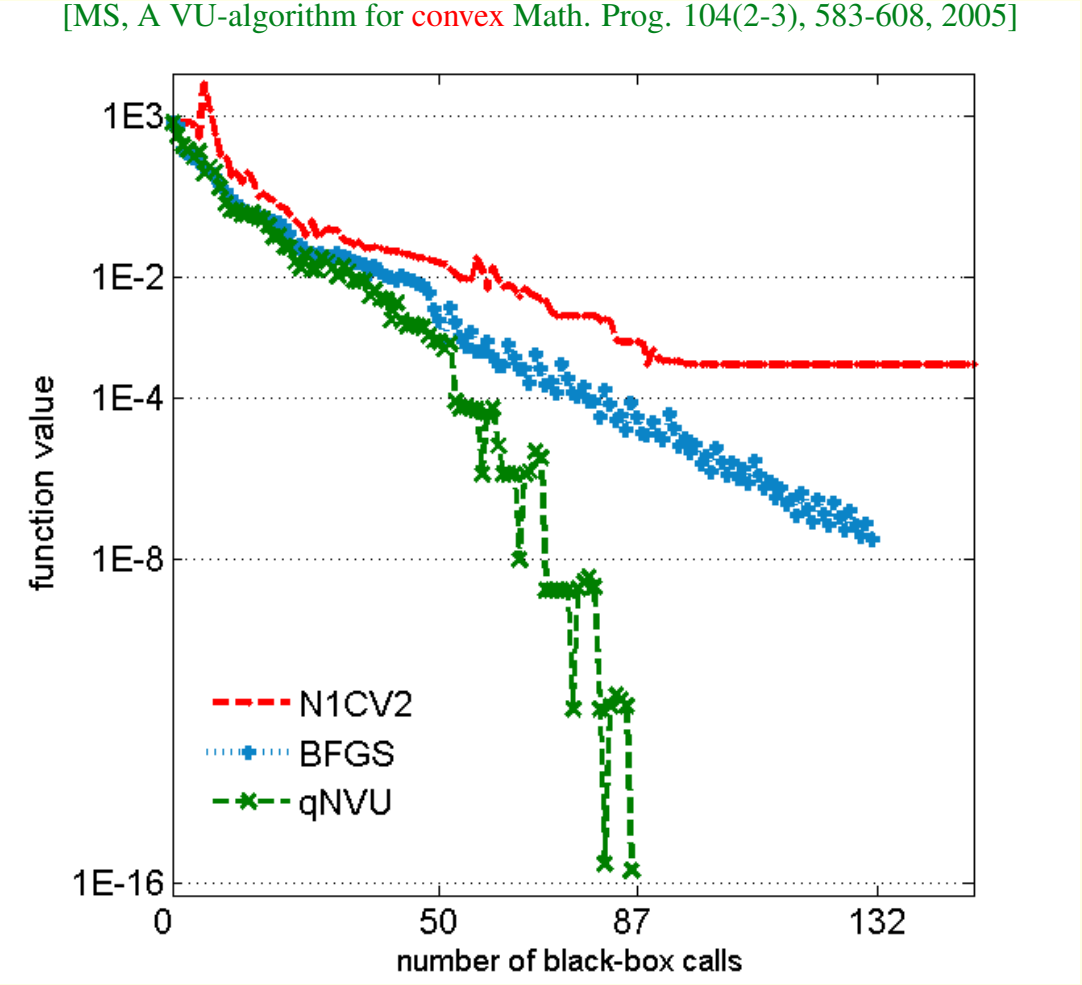
[MS, A VU-algorithm for convex Math. Prog. 104(2-3), 583-608, 2005]



Sublinear, linear, and superlinear convergence - convex case

Lewis and Overton 8-variable half-and-half function

[MS, A VU-algorithm for convex Math. Prog. 104(2-3), 583-608, 2005]



Sublinear, linear, and superlinear convergence -

What can be done for nonconvex case?

How does structure appear?

– Implicitly

  - U-Lagrangian
  - VU-decomposition
  - partly smooth functions

} digging tools

– Explicitly

  - as a sum
  - as a composition

} blackboxes

How does structure appear?

– Implicitly

- U-Lagrangian
- VU-decomposition
- partly smooth functions

} digging tools

– Explicitly

- as a sum
- as a composition

} $\neq$ black boxes

Explicit Structure:

Opening the Black Box

Psyche Opening the Golden Box, 1903 -J.W. Waterhouse

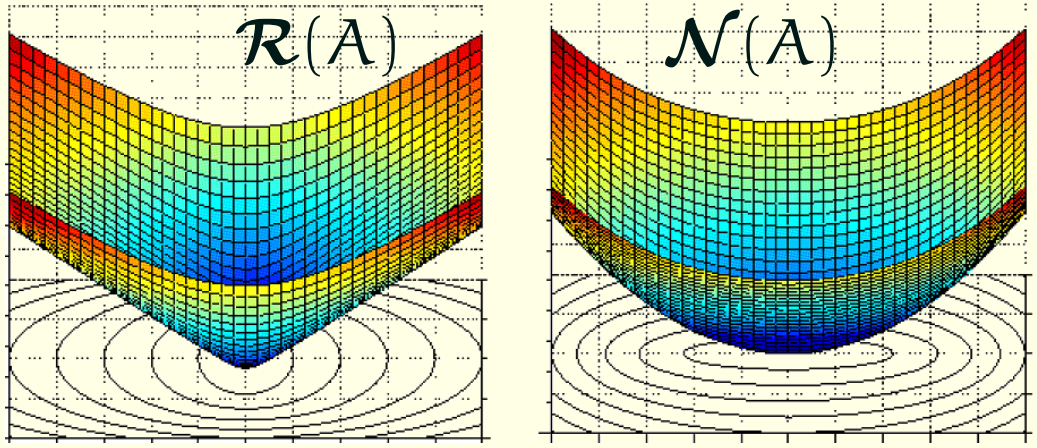
The half-and-half function

For $x \in \mathbb{R}^n$, given matrices $A \succeq 0$, $B \succ 0$,

$$f(x) = \sqrt{\langle x, Ax \rangle} + \langle x, Bx \rangle$$

has a unique minimizer at 0.

On $\mathcal{N}(A)$ the function is not differentiable, and the first term vanishes: $f|_{\mathcal{N}(A)}$ looks smooth.

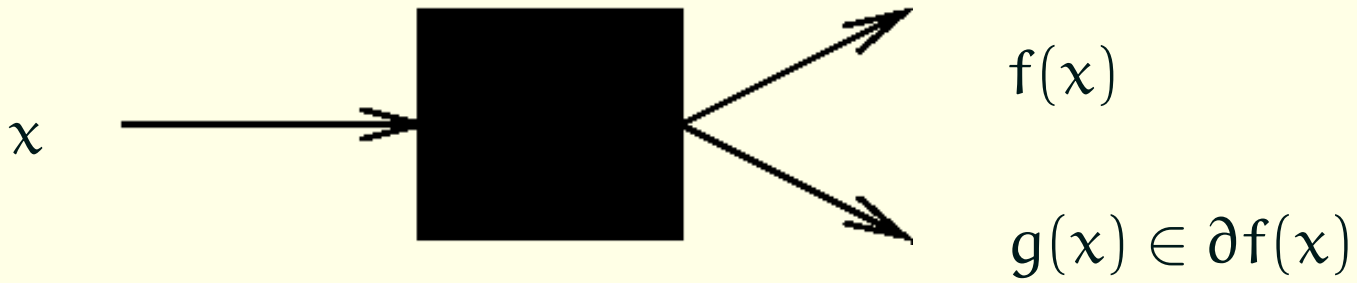


The half-and-half function has several interesting structures

If no structure at all

$$f(x) = \sqrt{\langle x, Ax \rangle} + \langle x, Bx \rangle$$

This defines the black box:



The half-and-half function has several interesting structures

Sum structure

$$f(x) = f_1(x) + f_2(x) \text{ with } \begin{cases} f_1(x) = \sqrt{\langle x, Ax \rangle} \\ f_2(x) = \langle x, Bx \rangle \quad \text{(smooth } \Psi\text{)} \end{cases}$$

This defines a sum black box:

$$f_1(x), f_2(x)$$

$$g_j(x) \in \partial f_j(x)_{j=1,2}$$

The half-and-half function has several interesting structures

Composite structure

$$f(x) = (h \circ c)(x) \text{ with } \begin{cases} c(x) = (x, \langle x, Bx \rangle) \in \mathbb{R}^{n+1} \\ h(C) = \sqrt{\langle C_{1:n}, AC_{1:n} \rangle} + C_{n+1} \end{cases}$$

for C smooth and h positively homogeneous

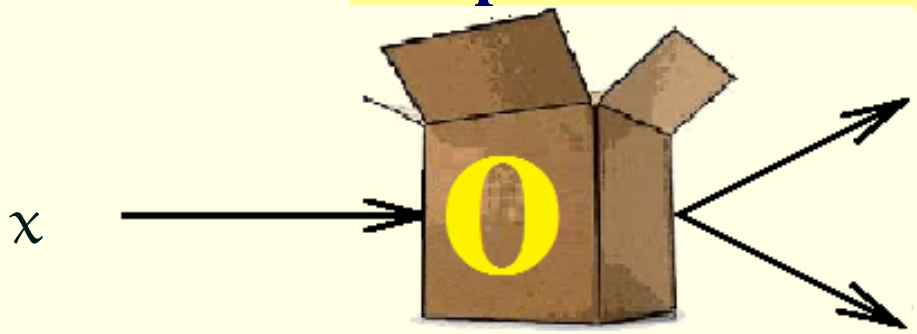
The half-and-half function has several interesting structures

Composite structure

$$f(x) = (h \circ c)(x) \text{ with } \begin{cases} c(x) = (x, \langle x, Bx \rangle) \in \mathbb{R}^{n+1} \\ h(C) = \sqrt{\langle C_{1:n}, AC_{1:n} \rangle} + C_{n+1} \end{cases}$$

for C smooth and h positively homogeneous

This defines a composite black box:



$C := c(x)$ **and** $h(C)$

Jacobian $c'(x)$ **and**

$G(C) \in \partial h(C)$

The half-and-half function has several interesting structures

Missing information structure

Suppose not all of A/B is known/accessible,

so that only estimates are available for f

This defines a noisy black box:

$$f_x \approx f(x)$$

$$g_x \approx g(x) \in \partial f(x)$$

The half-and-half function has several interesting structures

Missing information structure

Suppose not all of A/B is known/accessible,

so that only estimates are available for f

This defines a noisy black box:

$$f_x \approx f(x)$$

$$g_x \approx g(x) \in \partial f(x)$$

Not considered in this talk

(Convex: OlSagLem, NonConvex: HareSagSol)

The Bottom Line

- The same function can be seen in different manners
- Depending on how the oracle outputs its f/g information, different structure is revealed:
  - A sum of two nonsmooth functions
  - A sum of a nonsmooth function and a smooth one
  - A composition of a nonsmooth function with a smooth mapping

The Bottom Line

– The same function can be seen in different manners

– Depending on how the oracle outputs its f/g information, different structure is revealed:

  – A sum of two nonsmooth functions

  – A sum of a nonsmooth function and a smooth one

  – A composition of a nonsmooth function with a smooth mapping

  • What is the impact of structure in a bundle algorithm?

f/g

The Bottom Line

– The same function can be seen in different manners

– Depending on how the oracle outputs its f/g information, different structure is revealed:

  – A sum of two nonsmooth functions

  – A sum of a nonsmooth function and a smooth one

  – A composition of a nonsmooth function with a smooth mapping

- What is the impact of structure in a bundle algorithm?
- To which extent does the convergence analysis rely on the specific f/g output?

The Bottom Line

– The same function can be seen in different manners

– Depending on how the oracle outputs its f/g information, different structure is revealed:

  – A sum of two nonsmooth functions

  – A sum of a nonsmooth function and a smooth one

  – A composition of a nonsmooth function with a smooth mapping

- What is the impact of structure in a bundle algorithm?
- To which extent does the convergence analysis rely on the specific f/g output?
- Is the composite structure worth looking at?

Is the composite structure worth looking at?

- **Max-functions:**

$$f(x) = \max\{f_1(x), \ldots, f_m(x)\} \text{ so } \begin{cases} c_i(x) = f_i(x), i = 1:m \\ h(c) = \max(c_1, \ldots, c_m) \end{cases}$$

- **Sparse least squares (compressed sensing):**

$$f(x) = \tfrac{1}{2}|Rx - b|_2^2 + \tau|x|_1 \text{ so } \begin{cases} c_1(x) = \tfrac{1}{2}|Rx - b|_2^2 \\ c_{i+1}(x) = x_i, i = 1:m \\ h(c) = c_1 + \tau|c_{2:n+1}| \end{cases}$$

- **Sum of norms** $f(x) = \sum_{i=1}^m |f_i(x)|$ **so** $\begin{cases} c_i(x) = f_i(x), i = 1:m \\ h(c) = \sum_{i=1}^m |c_i| \end{cases}$
- **Sum of leading eigenvalues, ℓ_1-penalization of NLPs, etc**

Is the composite structure worth looking at?

- **Max-functions:**

$$f(x) = \max\{f_1(x), \ldots, f_m(x)\} \text{ so } \begin{cases} c_i(x) = f_i(x), i = 1:m \\ h(c) = \max(c_1, \ldots, c_m) \end{cases}$$

- **Sparse least squares (compressed sensing):**

$$f(x) = \tfrac{1}{2}|Rx - b|_2^2 + \tau|x|_1 \text{ so } \begin{cases} c_1(x) = \tfrac{1}{2}|Rx - b|_2^2 \\ c_{i+1}(x) = x_i, i = 1:m \\ h(c) = c_1 + \tau|c_{2:n+1}| \end{cases}$$

- **Sum of norms** $f(x) = \sum_{i=1}^m |f_i(x)|$ **so** $\begin{cases} c_i(x) = f_i(x), i = 1:m \\ h(c) = \sum_{i=1}^m |c_i| \end{cases}$
- **Sum of leading eigenvalues, ℓ_1-penalization of NLPs, etc**

YES!

NOTE: f **may fail to be convex**

Impact of Structure Knowledge

Black-box information defines pieces that put together create a model M of the function f. The model is used to define iterates not too far away from a "good" past iterate, $\hat{x}$.

At iteration i,

$$y^{i+1} \textbf{ minimizes } M(y) + \frac{1}{2}\mu|y - \hat{x}|^2$$

Impact of Structure Knowledge

Black-box information defines pieces that put together create a model M of the function f. The model is used to define iterates not too far away from a "good" past iterate, $\hat{x}$.

At iteration i,

$$y^{i+1} \text{ minimizes } M(y) + \frac{1}{2}\mu|y-\hat{x}|^2$$

"pieces" chosen to make minimization simple (convex QP)

for example, "piece"=linearization:

$$y^i \longrightarrow \blacksquare \begin{array}{l} f^i = f(y^i) \\ g^i = g(y^i) \end{array} \qquad \Longrightarrow \qquad f^i + \left\langle g^i, y - y^i \right\rangle$$

Impact of Structure Knowledge

Black-box information defines pieces that put together create a model M of the function f. The model is used to define iterates not too far away from a "good" past iterate, $\hat{x}$.

At iteration i,

$$y^{i+1} \text{ minimizes } M(y) + \frac{1}{2}\mu|y - \hat{x}|^2$$

"pieces" chosen to make minimization simple (convex QP)

for example, "piece"=linearization:

$$y^i \longrightarrow \blacksquare \begin{array}{l} f^i = f(y^i) \\ g^i = g(y^i) \end{array} \qquad \Longrightarrow M(y) = \max_i \left\{ f^i + \left\langle g^i, y - y^i \right\rangle \right\}$$

$$y^{i+1} = \arg\min_x M(y) + \frac{1}{2}\mu|y - \hat{x}|^2$$

for example, "piece"=linearization:

$$y^i \longrightarrow \blacksquare \begin{array}{l} f^i = f(y^i) \\ g^i = g(y^i) \end{array} \qquad \Longrightarrow M(y) = \max_i \left\{ f^i + \left\langle g^i, y - y^i \right\rangle \right\}$$

Some jargon

$\hat{x}$ **is a serious point**

$\bigcup_i (y^i, f^i, g^i)$ **is the bundle** B

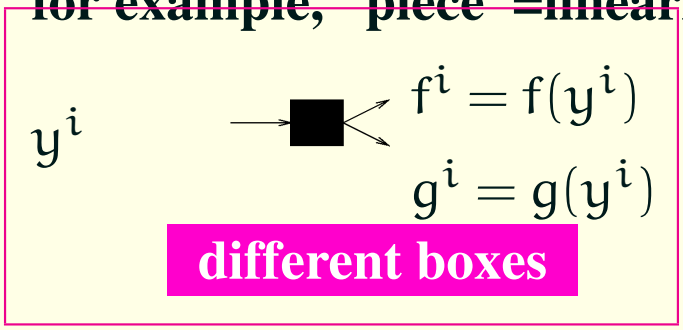
If y^{i+1} **gives sufficient decrease for** f **(in terms of a progress measure),**

it becomes the next $\hat{x}$

Otherwise, it is declared a null point

$$y^{i+1} = \arg\min_x M(y) + \frac{1}{2}\mu|y - \hat{x}|^2$$

for example, "piece"=linearization:



$$\Longrightarrow M(y) = \max_i \left\{ f^i + \left\langle g^i, y - y^i \right\rangle \right\}$$

different models

Some jargon

$\hat{x}$ **is a serious point**

$\bigcup_i (y^i, f^i, g^i)$ **is the bundle** B

If y^{i+1} **gives sufficient decrease for** f **(in terms of a progress measure),**

it becomes the next $\hat{x}$

Otherwise, it is declared a null point

Back to our questions

OK! Is the composite structure worth looking at?

OK! What is the impact of structure in a bundle algorithm?

- **To which extent does the convergence analysis rely on the specific f/g output?**

Back to our questions

OK! Is the composite structure worth looking at?

OK! What is the impact of structure in a bundle algorithm?

- **To which extent does the convergence analysis**

  rely on the specific f/g **output?**

  rely on the specific model?

Back to our questions

OK! Is the composite structure worth looking at?

OK! What is the impact of structure in a bundle algorithm?

- **To which extent does the convergence analysis**

  rely on the specific f/g output?

  rely on the specific model?

What about convexity?

The setting

$\min_{y \in \mathbb{R}^n} f(y)$; f locally Lipschitz,

only one (Clarke) gradient $g(y)$,

computed by a black box at each y.

Black-box information used to build a model function

$M(d) \approx f(\hat{x} + d)$ **near a center** $\hat{x}$

To which extent does the convergence analysis

rely on the specific model?

Some Model Functions

$$f(y) \quad \Rightarrow \quad M(d) \quad = \quad \max_{i \in B} \left\{ f^i + \left\langle g^i, \hat{x} + d - y^i \right\rangle \right\} \qquad (y = \hat{x} + d)$$

f_1

f_1

h

Some Model Functions

$$f(y) \quad \Rightarrow \quad M(d) \quad = \quad \max_{i \in B} \left\{ f^i + \left\langle g^i, \hat{x} + d - y^i \right\rangle \right\} \qquad (y = \hat{x} + d)$$

$$= \quad f(\hat{x}) + \max_{i \in B} \left\{ -e^i + \left\langle g^i, d \right\rangle \right\} \quad \textbf{(blind CP)}$$

f_1

f_1

h

Some Model Functions

$(y = \hat{x} + d)$

$$f(y) \quad \Rightarrow \quad M(d) \quad = \quad \max_{i \in B} \left\{ f^i + \left\langle g^i, \hat{x} + d - y^i \right\rangle \right\}$$

$$= \quad f(\hat{x}) + \max_{i \in B} \left\{ -e^i + \left\langle g^i, d \right\rangle \right\} \textbf{ (blind CP)}$$

$$f_1(y) + \Psi(y) \quad \Rightarrow \quad M(d) \quad = \quad \textbf{blind CP for } f_1$$

$$+ \Psi(\hat{x}) + \left\langle \nabla \Psi(\hat{x}), d \right\rangle \textbf{ (structured CP)}$$

f_1

h

Some Model Functions

$(y = \hat{x} + d)$

$$f(y) \quad \Rightarrow \quad M(d) \quad = \quad \max_{i \in B} \left\{ f^i + \left\langle g^i, \hat{x} + d - y^i \right\rangle \right\}$$

$$= \quad f(\hat{x}) + \max_{i \in B} \left\{ -e^i + \left\langle g^i, d \right\rangle \right\} \textbf{ (blind CP)}$$

$$f_1(y) + \Psi(y) \quad \Rightarrow \quad M(d) \quad = \quad \textbf{blind CP for } f_1$$

$$+ \Psi(\hat{x}) + \left\langle \nabla \Psi(\hat{x}), d \right\rangle \textbf{ (structured CP)}$$

$$\Rightarrow \quad M(d) \quad = \quad \textbf{blind CP for } f_1 + \Psi(\hat{x}) + \left\langle \nabla \Psi(\hat{x}), d \right\rangle$$

$$+ \frac{1}{2} \left\langle d, \nabla^2 \Psi(\hat{x}) d \right\rangle \textbf{ (structured nonpolyhedral)}$$

Some Model Functions

$(y = \hat{x} + d)$

$$f(y) \quad \Rightarrow \quad M(d) = \max_{i \in B} \left\{ f^i + \left\langle g^i, \hat{x} + d - y^i \right\rangle \right\}$$
$$= f(\hat{x}) + \max_{i \in B} \left\{ -e^i + \left\langle g^i, d \right\rangle \right\} \textbf{ (blind CP)}$$

$$f_1(y) + \Psi(y) \quad \Rightarrow \quad M(d) = \textbf{blind CP for } f_1$$
$$+ \Psi(\hat{x}) + \left\langle \nabla \Psi(\hat{x}), d \right\rangle \textbf{ (structured CP)}$$
$$\Rightarrow \quad M(d) = \textbf{blind CP for } f_1 + \Psi(\hat{x}) + \left\langle \nabla \Psi(\hat{x}), d \right\rangle$$
$$+ \tfrac{1}{2} \left\langle d, \nabla^2 \Psi(\hat{x}) d \right\rangle \textbf{ (structured nonpolyhedral)}$$

$$(h \circ c)(y) \quad \Rightarrow \quad M(d) = h \circ (c(\hat{x}) + c'(\hat{x}) d) \textbf{ (composite nonpolyhedral)}$$

Some Model Functions

$(y = \hat{x} + d)$

$$
\begin{array}{lcll}
f(y) \Rightarrow M(d) & = & \max\limits_{i \in B} \left\{ f^i + \left\langle g^i, \hat{x} + d - y^i \right\rangle \right\} & \\
& = & f(\hat{x}) + \max\limits_{i \in B} \left\{ -e^i + \left\langle g^i, d \right\rangle \right\} & \textbf{(blind CP)} \\
f_1(y) + \Psi(y) \Rightarrow M(d) & = & \textbf{blind CP for } f_1 & \\
& & + \Psi(\hat{x}) + \left\langle \nabla \Psi(\hat{x}), d \right\rangle & \textbf{(structured CP)} \\
& \Rightarrow M(d) & = & \textbf{blind CP for } f_1 + \Psi(\hat{x}) + \left\langle \nabla \Psi(\hat{x}), d \right\rangle \\
& & + \frac{1}{2} \left\langle d, \nabla^2 \Psi(\hat{x}) d \right\rangle & \textbf{(structured nonpolyhedral)} \\
(h \circ c)(y) \Rightarrow M(d) & = & h \circ (c(\hat{x}) + c'(\hat{x}) d) & \textbf{(composite nonpolyhedral)} \\
& M(d) & = & \textbf{blind CP for } h \circ \left(c(\hat{x}) + c'(\hat{x}) d \right) \\
& & & \textbf{(composite polyhedral)}
\end{array}
$$

Some Model Functions

$(y = \hat{x} + d)$

$$f(y) \Rightarrow M(d) = \max_{i \in B} \left\{ f^i + \left\langle g^i, \hat{x} + d - y^i \right\rangle \right\}$$
$$= f(\hat{x}) + \max_{i \in B} \left\{ -e^i + \left\langle g^i, d \right\rangle \right\} \textbf{ (blind CP)}$$

$$f_1(y) + \Psi(y) \Rightarrow M(d) = \textbf{blind CP for } f_1$$
$$+ \Psi(\hat{x}) + \left\langle \nabla \Psi(\hat{x}), d \right\rangle \textbf{ (structured CP)}$$

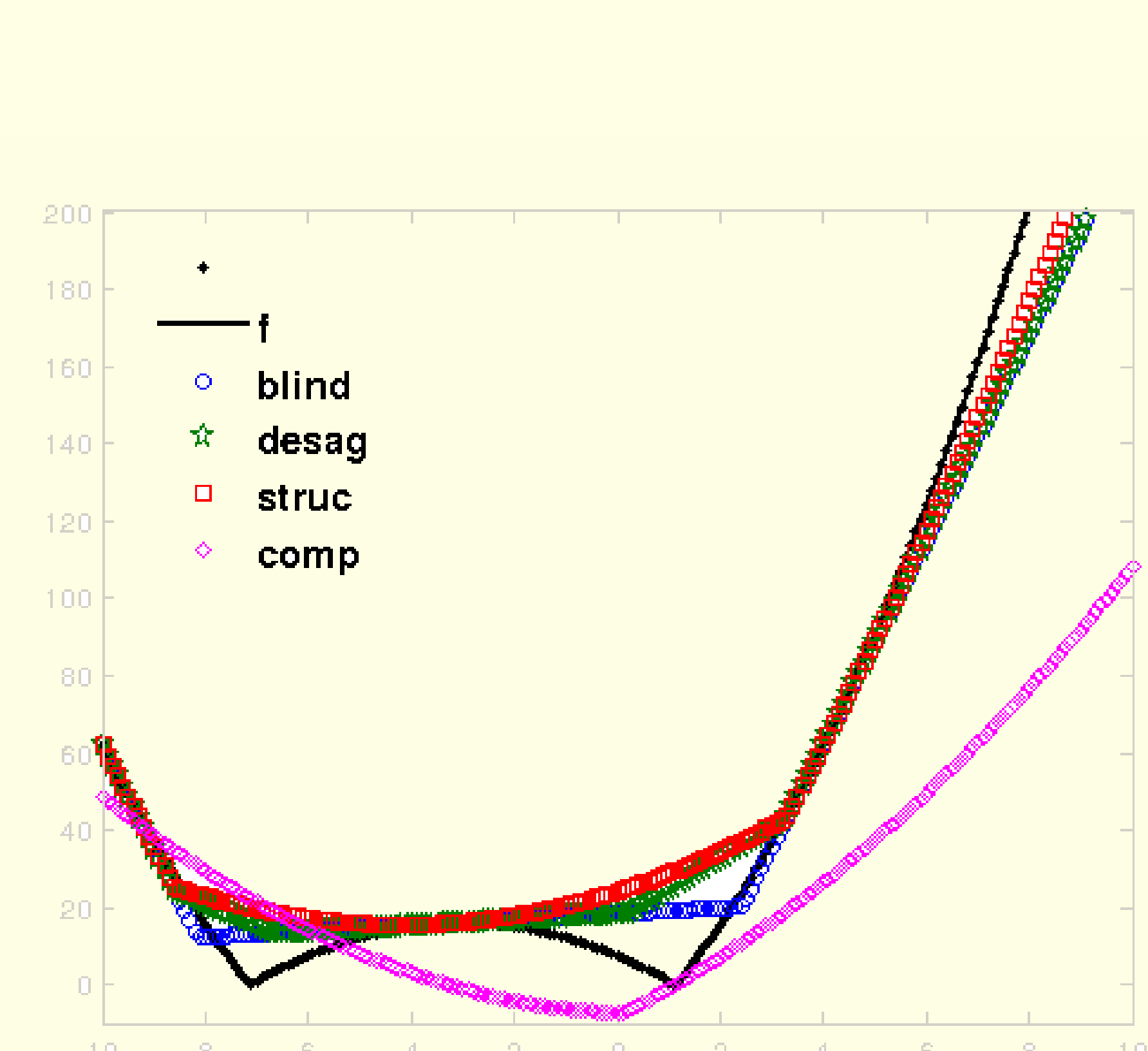
$$\Rightarrow M(d) = \textbf{blind CP for } f_1 + \Psi(\hat{x}) + \left\langle \nabla \Psi(\hat{x}), d \right\rangle$$
$$+ \frac{1}{2} \left\langle d, \nabla^2 \Psi(\hat{x}) d \right\rangle \textbf{ (structured nonpolyhedral)}$$

$$(h \circ c)(y) \Rightarrow M(d) = h \circ (c(\hat{x}) + c'(\hat{x}) d) \textbf{ (composite nonpolyhedral)}$$

$$M(d) = \textbf{blind CP for } h \circ \left(c(\hat{x}) + c'(\hat{x}) d \right)$$

(composite polyhedral)

or we can add second order terms for c, etc


f
blind
desag
struc
comp
200
180
160
140
120
100
80
60
40
20
0

Some Model Functions

$(y = \hat{x} + d)$

$$f(y) \Rightarrow M(d) = \max_{i \in B} \left\{ f^i + \left\langle g^i, \hat{x} + d - y^i \right\rangle \right\}$$
$$= f(\hat{x}) + \max_{i \in B} \left\{ -e^i + \left\langle g^i, d \right\rangle \right\} \textbf{ (blind CP)}$$

$$f_1(y) + \Psi(y) \Rightarrow M(d) = \textbf{blind CP for } f_1$$
$$+ \Psi(\hat{x}) + \left\langle \nabla \Psi(\hat{x}), d \right\rangle \textbf{ (structured CP)}$$
$$\Rightarrow M(d) = \textbf{blind CP for } f_1 + \Psi(\hat{x}) + \left\langle \nabla \Psi(\hat{x}), d \right\rangle$$
$$+ \tfrac{1}{2} \left\langle d, \nabla^2 \Psi(\hat{x}) d \right\rangle \textbf{ (structured nonpolyhedral)}$$

$$(h \circ c)(y) \Rightarrow M(d) = h \circ (c(\hat{x}) + c'(\hat{x}) d) \textbf{ (composite nonpolyhedral)}$$
$$M(d) = \textbf{blind CP for } h \circ \left(c(\hat{x}) + c'(\hat{x}) d \right)$$
$$\textbf{(composite polyhedral)}$$

or we can add second order terms for c, etc

What are the least conditions we can set on M??

Model and f at the center: lower type ($M(0) \le f(\hat{x})$)

$$
\begin{aligned}
f(\hat{x}) \quad &\Rightarrow \quad M(0) \quad = \quad \max_{i \in B} \left\{ f^i + \left\langle g^i, \hat{x} - y^i \right\rangle \right\} \\
&\qquad\qquad = \quad f(\hat{x}) + \max_{i \in B} \left\{ -e^i + \left\langle g^i, 0 \right\rangle \right\} \text{ (blind CP)} \\
f_1(\hat{x}) + \Psi(\hat{x}) \quad &\Rightarrow \quad M(0) \quad = \quad \text{blind CP for } f_1 \\
&\qquad\qquad + \Psi(\hat{x}) + \left\langle \nabla \Psi(\hat{x}), 0 \right\rangle \text{ (structured CP)} \\
&\Rightarrow \quad M(0) \quad = \quad \text{blind CP for } f_1 + \Psi(\hat{x}) \\
&\qquad\qquad + \tfrac{1}{2} \left\langle 0, \nabla^2 \Psi(\hat{x}) 0 \right\rangle \text{ (structured nonpolyhedral)} \\
(h \circ c)(\hat{x}) \quad &\Rightarrow \quad M(0) \quad = \quad h \circ (c(\hat{x}) + c'(\hat{x}) 0) \text{ (composite nonpolyhedral)} \\
&\qquad\qquad M(0) \quad = \quad \text{blind CP for } h \circ \Big(c(\hat{x}) + c'(\hat{x}) 0 \Big) \\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \text{(composite polyhedral)}
\end{aligned}
$$

or we can add second order terms for c, etc

Proximal Bundle Method iteration elements

For a prox-parameter $\mu > 0$ **and a**
model function $M(d) \approx f(\hat{x}+d)$ **near a center** $\hat{x}$
search direction $d(\hat{x}) := \arg\min M(\cdot) + \frac{1}{2}\mu|\cdot|^2$**,**

Proximal Bundle Method iteration elements

For a prox-parameter $\mu > 0$ **and a**
model function $M(d) \approx f(\hat{x}+d)$ **near a center** $\hat{x}$
search direction $d(\hat{x}) := \arg\min M(\cdot) + \frac{1}{2}\mu|\cdot|^2$**,**
model Gradient

$$G(\hat{x}) := -\mu d(\hat{x}) \in \partial M(d(\hat{x}))$$

Affine support for the model

$$A(d) := M(d(\hat{x})) + \langle G(\hat{x}), d - d(\hat{x})\rangle$$

Error $E(\hat{x}) := M(0) - A(0)$

Proximal Bundle Method iteration elements

For a prox-parameter $\mu > 0$ **and a**
model function $M(d) \approx f(\hat{x}+d)$ **near a center** $\hat{x}$
search direction $d(\hat{x}) := \arg\min M(\cdot) + \frac{1}{2}\mu|\cdot|^2$**,**
model Gradient

$$G(\hat{x}) := -\mu d(\hat{x}) \in \partial M(d(\hat{x}))$$

Affine support for the model

$$A(d) := M(d(\hat{x})) + \langle G(\hat{x}), d - d(\hat{x})\rangle$$

Error $E(\hat{x}) := M(0) - A(0)$

How general is this algorithmic scheme?

Various Bundle Methods iteration elements

A parameter $\mu = \mu(d(\hat{x})) > 0$ **and a**
model function $M(d) \approx f(\hat{x}+d)$ **near a center** $\hat{x}$
search direction and model Gradient

$$G(\hat{x}) := -\mu d(\hat{x}) \in \partial M(d(\hat{x}))$$

Affine support for the model

$$A(d) := M(d(\hat{x})) + \left\langle G(\hat{x}), d - d(\hat{x}) \right\rangle$$

Error $E(\hat{x}) := M(0) - A(0)$

Various Bundle Methods iteration elements

A parameter $\mu = \mu(d(\hat{x})) > 0$ **and a**
model function $M(d) \approx f(\hat{x}+d)$ **near a center** x
search direction and model Gradient

$$G(\hat{x}) := -\mu d(\hat{x}) \in \partial M(d(\hat{x}))$$

Affine support for the model

$$A(d) := M(d(\hat{x})) + \left\langle G(\hat{x}), d - d(\hat{x}) \right\rangle$$

Error $E(\hat{x}) := M(0) - A(0)$

Note 1: $E(\hat{x}) \geq 0$ **by subgradient inequality (**M **is convex)**

Various Bundle Methods iteration elements

A parameter $\mu = \mu(d(\hat{x})) > 0$ **and a**
model function $M(d) \approx f(\hat{x}+d)$ **near a center** x
search direction and model Gradient

$$G(\hat{x}) := -\mu d(\hat{x}) \in \partial M(d(\hat{x}))$$

Affine support for the model

$$A(d) := M(d(\hat{x})) + \left\langle G(\hat{x}), d - d(\hat{x}) \right\rangle$$

Error $E(\hat{x}) := M(0) - A(0)$

Note 1: $E(\hat{x}) \geq 0$ **by subgradient inequality (**M **is convex)**

Note 2: $d(\hat{x})$ **may provide next iterate (**$y_+ = \hat{x} + d(\hat{x})$**),**

or we may linesearch for next iterate ($y(t) = \hat{x} + td(\hat{x})$**),**

or we may curvesearch ($y(\mu) = \hat{x} + d_\mu(\hat{x})$**), without changing** M

Stationarity result

Model conditions

[M_0] The model is convex

[M_1] The model is lower at the center: $M(0) \leq f(\hat{x})$

[M_2] (finite version) When the error is zero, the model gradient is an f-subgradient: $E(\hat{x}) = 0 \Longrightarrow G(\hat{x}) \in \partial f(\hat{x})$

Stationarity result (finite version)

Model conditions

[M_0] The model is convex

[M_1] The model is lower at the center: $M(0) \leq f(\hat{x})$

[M_2] (finite version) When the error is zero, the model gradient is an f-subgradient: $E(\hat{x}) = 0 \Longrightarrow G(\hat{x}) \in \partial f(\hat{x})$

Theorem For the **progress measure**

$D(\hat{x}) := f(\hat{x}) - M(d(\hat{x})) - \frac{\mu}{2}|d(\hat{x})|^2$

(i) $D(\hat{x}) = f(\hat{x}) - M(0) + E(\hat{x}) + \frac{1}{2\mu}|G(\hat{x})|^2$ and, hence, $M_1 \Longrightarrow D(\hat{x}) \geq 0$.

(ii) If $M_{1,2}$ hold, then either $D(\hat{x}) = 0$ or $d(\hat{x}) = 0$ implies $0 = G(\hat{x}) \in \partial f(\hat{x})$, i.e. $\hat{x}$ is stationary for f.

Stationarity result (asymptotic)

Model conditions

$[M_0]$ The model is convex

$[M_1]$ The model is lower at the center: $M(0) \leq f(\hat{x})$

$[M_2]$ (algorithmic version) When the error $\to 0$ the model gradient is an f-subgradient in the limit:

$E(\hat{x}^k) \to 0 \Longrightarrow G(\hat{x}^\infty) \in \partial f(\hat{x}^\infty)$

Stationarity result (asymptotic)

Model conditions

$[M_0]$ The model is convex

$[M_1]$ The model is lower at the center: $M(0) \leq f(\hat{x})$

$[M_2]$ (algorithmic version) When the error $\to 0$ the model gradient is an f-subgradient in the limit:

$E(\hat{x}^k) \to 0 \Longrightarrow G(\hat{x}^\infty) \in \partial f(\hat{x}^\infty)$

Stationarity result (asymptotic)

Model conditions

[M_0] The model is convex

[M_1] The model is lower at the center: $M(0) \leq f(\hat{x})$

[M_2] (algorithmic version) When the error $\to 0$ the model gradient is an f-subgradient in the limit:
$E(\hat{x}^k) \to 0 \Longrightarrow G(\hat{x}^\infty) \in \partial f(\hat{x}^\infty)$

For nonconvex f, condition M_2 ensured by **selecting** B-points via a **linesearch** ($d(\hat{x}) \neq 0$),

Linesearch

A subalgorithm detecting if $y_\ell = \hat{x}^k + t_\ell d(\hat{x}^k)$ is either

- a serious step: $\hat{x}^{k+1} = y_\ell$ and black-box information at y_ℓ enters the model; or
- a null step: black-box information at y_ℓ enters the model; or
- adjusts t_ℓ to try with $y_{\ell+1}$.

Linesearch

A subalgorithm detecting if $y_\ell = \hat{x}^k + t_\ell d(\hat{x}^k)$ is either

- a serious step: $\hat{x}^{k+1} = y_\ell$ and black-box information at y_ℓ enters the model; or
- a null step: black-box information at y_ℓ enters the model; or
- adjusts t_ℓ to try with $y_{\ell+1}$.

Main features

1. For semismooth functions it has finite termination
2. Declares serious/null step checking the progress measure and also if **$\mathcal{V}$-approximation is good enough**
3. Null-step condition is the weakest in the market

Null Step Condition

Only requires that at fixed center $\hat{x}$

- $M_+(d) \geq A(d)$
- $M_+(d) \geq L(d) := f(\hat{x}) - e_\ell + \langle g_\ell, d \rangle$

for e_ℓ and g_ℓ defined with black-box information at y_ℓ

Null Step Condition

Only requires that at fixed center $\hat{x}$

- $M_+(d) \geq A(d)$
- $M_+(d) \geq L(d) := f(\hat{x}) - e_\ell + \langle g_\ell, d \rangle$

for e_ℓ and g_ℓ defined with black-box information at y_ℓ

Note 1: if f convex, $g_\ell = g(y_\ell)$ and e_ℓ linearization error

Null Step Condition

Only requires that at fixed center $\hat{x}$

- $M_+(d) \geq A(d)$
- $M_+(d) \geq L(d) := f(\hat{x}) - e_\ell + \langle g_\ell, d \rangle$

for e_ℓ and g_ℓ defined with black-box information at y_ℓ

Note 1: if f convex, $g_\ell = g(y_\ell)$ and e_ℓ linearization error

Note 2: this does not mean M is **polyhedral**

Null Step Condition

Only requires that at fixed center $\hat{x}$

- $M_+(d) \geq A(d)$
- $M_+(d) \geq L(d) := f(\hat{x}) - e_\ell + \langle g_\ell, d \rangle$

for e_ℓ and g_ℓ defined with black-box information at y_ℓ

Note 1: if f convex, $g_\ell = g(y_\ell)$ and e_ℓ linearization error

Note 2: this does not mean M is **polyhedral**

Note 3: if f nonconvex, define g_ℓ and e_ℓ checking **curvature**

Null Step Condition

Only requires that at fixed center $\hat{x}$

- $M_+(d) \geq A(d)$
- $M_+(d) \geq L(d) := f(\hat{x}) - e_\ell + \langle g_\ell, d \rangle$

for e_ℓ and g_ℓ defined with black-box information at y_ℓ

Note 1: if f convex, $g_\ell = g(y_\ell)$ and e_ℓ linearization error

Note 2: this does not mean M is **polyhedral**

Note 3: if f nonconvex, define g_ℓ and e_ℓ checking

curvature: now the bundle $B = \bigcup_i (y^i, f^i, g^i, H^i)$

includes matrices H^i

the curvature is $\kappa_\ell := \langle y_\ell - \hat{x}, H_\ell(y_\ell - \hat{x}) \rangle$

Curvature (nonconvex f, polyhedral M)

Having the bundle $B = \bigcup_i (y^i, f^i, g^i, H^i)$ and

$\kappa_\ell := \langle y_\ell - \hat{x}, H_\ell (y_\ell - \hat{x}) \rangle$, define

$$g_\ell = \begin{cases} g(y_\ell) & \text{if } \kappa_\ell \geq 0 \\ g(y_\ell) + H_\ell (y_\ell - \hat{x}) & \text{if } \kappa_\ell < 0 \end{cases}$$

Curvature (nonconvex f, polyhedral M)

Having the bundle $B = \bigcup_i (y^i, f^i, g^i, H^i)$ and

$\kappa_\ell := \langle y_\ell - \hat{x}, H_\ell(y_\ell - \hat{x}) \rangle$, define

$$g_\ell = \begin{cases} g(y_\ell) & \text{if } \kappa_\ell \geq 0 \\ g(y_\ell) + H_\ell(y_\ell - \hat{x}) & \text{if } \kappa_\ell < 0 \end{cases}$$

and

$$e_\ell = \begin{cases} \text{lin.error} & \text{if } \kappa_\ell \geq 0 \\ \max\left(\text{lin.error} - \frac{1}{2}\kappa_\ell, s|y_\ell - \hat{x}|^2\right) & \text{if } \kappa_\ell < 0 \end{cases}$$

Curvature (nonconvex f, polyhedral M)

Having the bundle $B = \bigcup_i (y^i, f^i, g^i, H^i)$ and

$\kappa_\ell := \langle y_\ell - \hat{x}, H_\ell(y_\ell - \hat{x}) \rangle$, define

$$g_\ell = \begin{cases} g(y_\ell) & \text{if } \kappa_\ell \geq 0 \\ g(y_\ell) + H_\ell(y_\ell - \hat{x}) & \text{if } \kappa_\ell < 0 \end{cases}$$

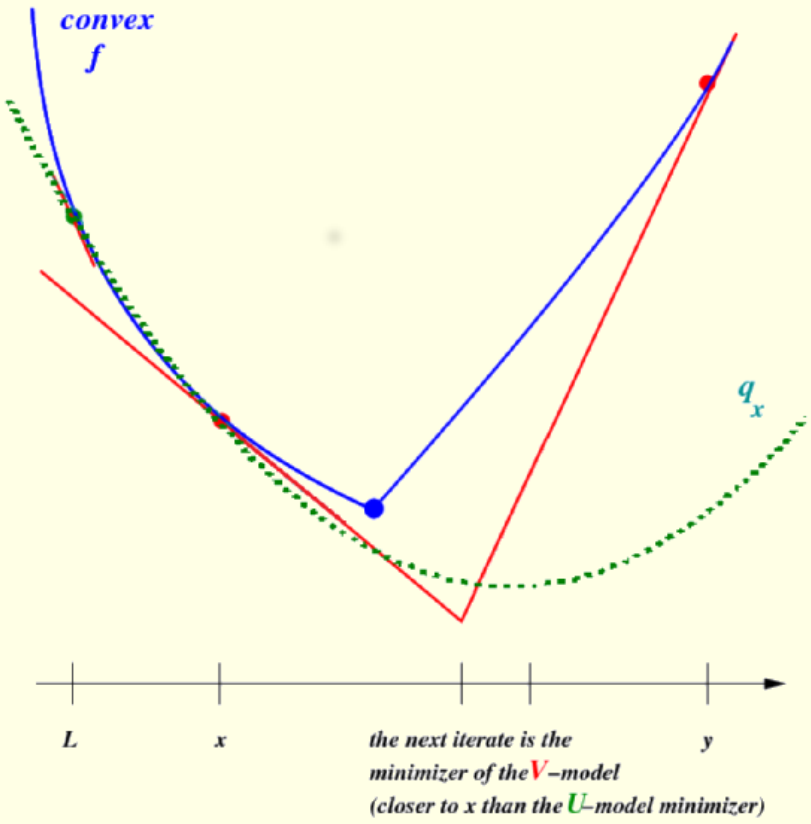
and

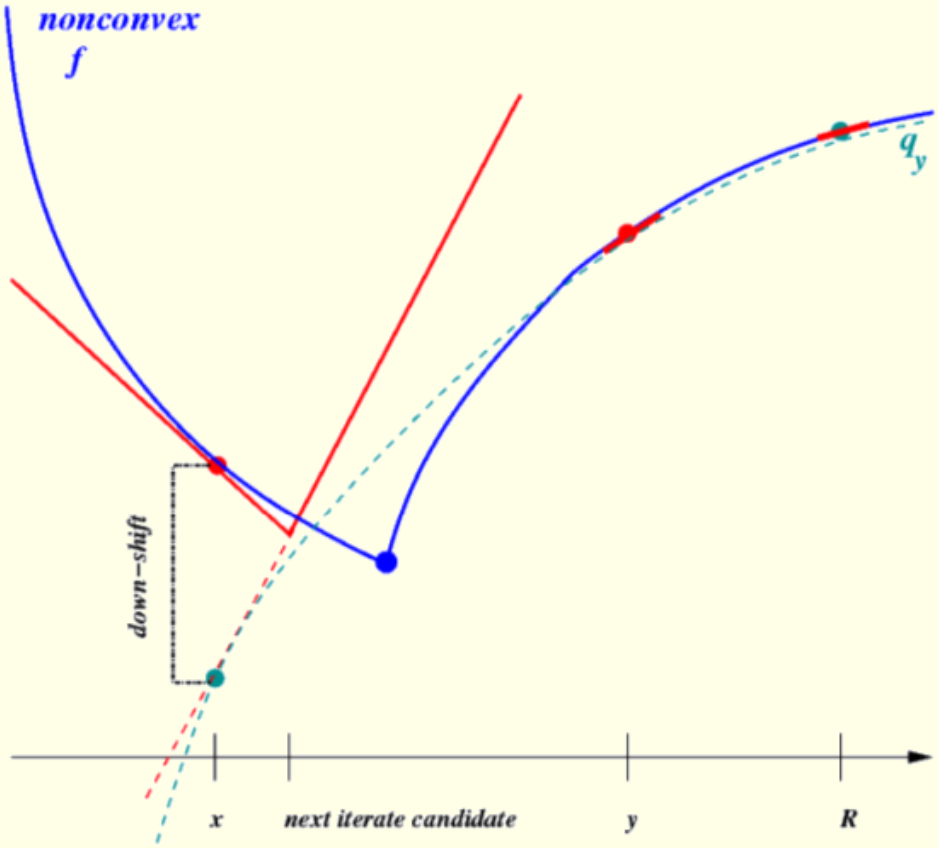
$$e_\ell = \begin{cases} \text{lin.error} & \text{if } \kappa_\ell \geq 0 \\ \max\left(\text{lin.error} - \frac{1}{2}\kappa_\ell, s|y_\ell - \hat{x}|^2\right) & \text{if } \kappa_\ell < 0 \end{cases}$$

Note: Safeguard s ensures $e_\ell \geq 0$ and, hence, $M(0) \leq f(\hat{x})$

Curvature (nonconvex f, polyhedral M)

Bundle matrices H computed via quadratic approximations





Final Comments

- Brings a model-view into the nonconvex setting
- Weakest NS condition relies heavily on conjugacy and Convex Analysis ($\neq$ Correa and Lemaréchal NS analysis)
- The linesearch extends the one-dimensional superlinearly convergent method by Lemaréchal and Mifflin 1982

This is difficult stuff

Final Comments

- Brings a model-view into the nonconvex setting
- Weakest NS condition relies heavily on conjugacy and Convex Analysis ($\neq$ Correa and Lemaréchal NS analysis)
- The linesearch extends the one-dimensional superlinearly convergent method by Lemaréchal and Mifflin 1982
- **This is difficult stuff**