



Software and Tools
for Computational
Engineering

RWTHAACHEN
UNIVERSITY

Convex Relaxations of Fixed-point Iterations and their Subgradients

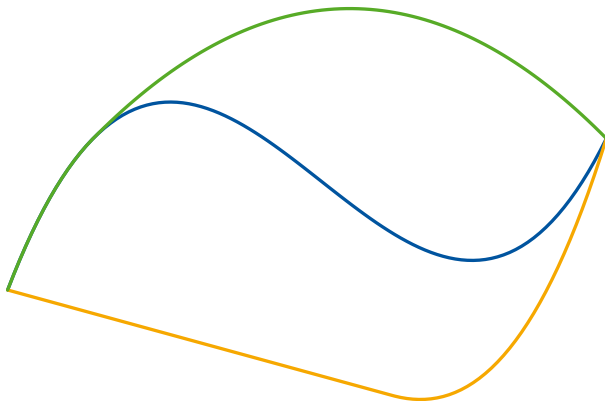
EuroAD 2026

Neil Kichler 

kichler@stce.rwth-aachen.de

Motivation

Interested in **convex relaxations (underestimators)** of **functions**.



Motivation

Context

Deterministic Global Optimization (DGO)

- ↪ Mixed-Integer Nonlinear Programming (MINLP)
- ↪ typically requires **explicit** factorable functions
- ↪ a computational DAG

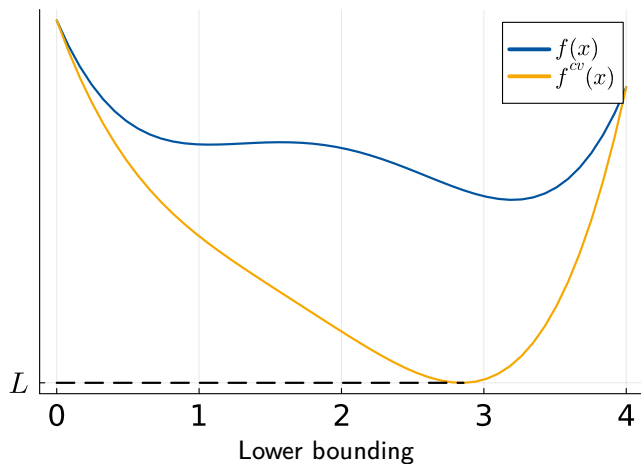
Goal

Extend applicability to larger class of problems

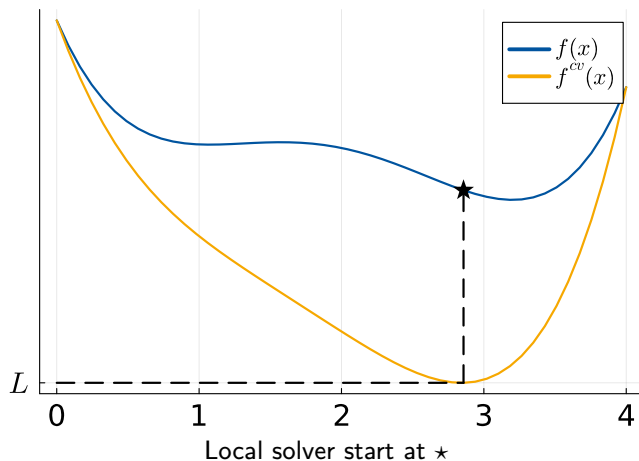
- ↪ Implicit functions
- ↪ Fixed-point iterations
- ↪ ...

Deterministic Global Optimization
 $\hookrightarrow$ Spatial Branch & Bound

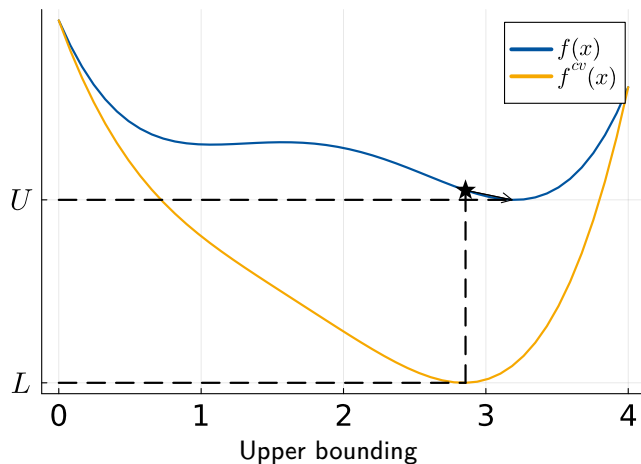
Deterministic Global Optimization



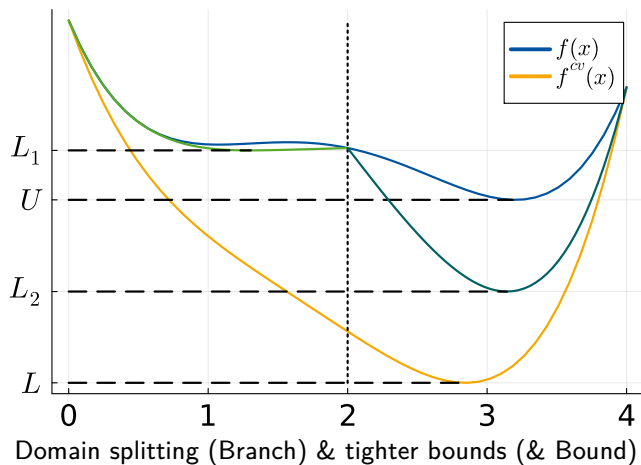
Deterministic Global Optimization



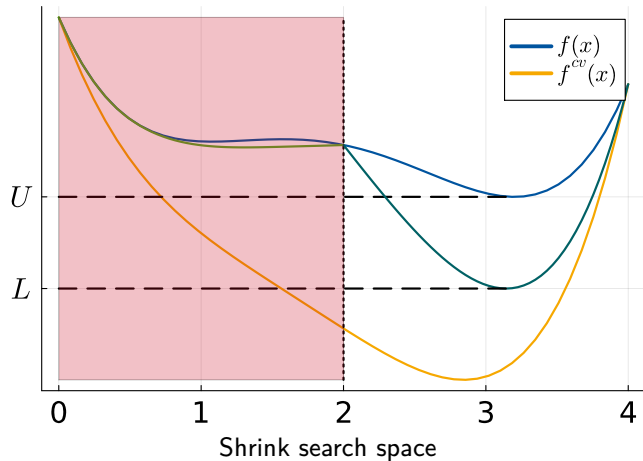
Deterministic Global Optimization



Deterministic Global Optimization

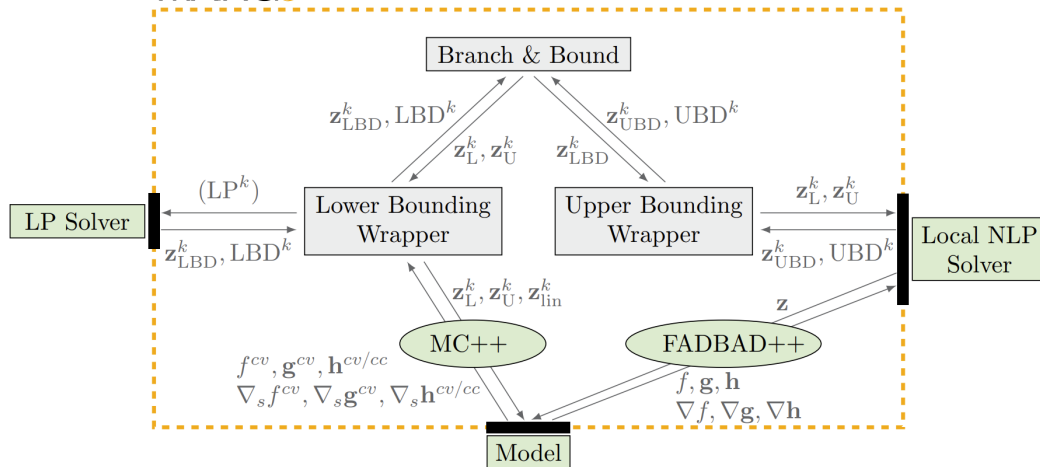


Deterministic Global Optimization



Deterministic Global Optimization

MAiNGO 



“MAiNGO - McCormick-based Algorithm for mixed-integer Nonlinear Global Optimization”, Bongartz et al. (2018)

Goal

Relaxations of fixed-point iterations

Convex Relaxations for fixed-points

Have:

$$x_{\star}(p) := \phi(x_{\star}(p), p), \quad \forall p \in P.$$

Want:

Relaxation of $p \mapsto x_{\star}(p)$

Existing relaxation methods

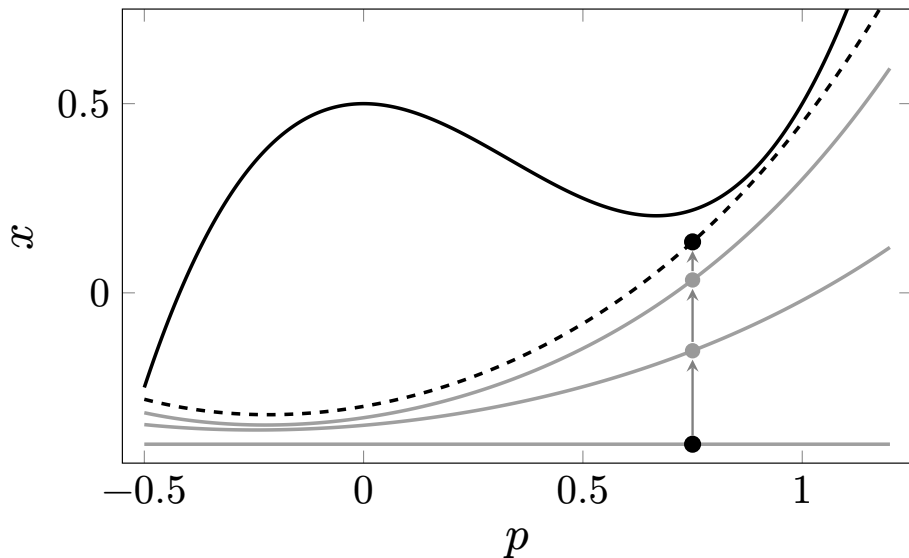
Direct methods:

- ↪ Successive-substitution relaxations (Scott et al. [2])
- ↪ Custom Gauss-Seidel solver relaxations (Stuber et al. [3])

Optimization-based:

- ↪ Optimize implicit functions $f^{cv}(x) \leq 0 \leq f^{cc}(x)$ (Cao and Khan [4]).

Direct Successive-substitution relaxations [2]



Practical considerations for fixed-points

Assumptions¹

Existence and uniqueness

Assumption 3.1 There exists $\mathbf{x} : P \rightarrow \mathbb{R}^{n_x}$ such that $\mathbf{x}(\mathbf{p}) = \phi(\mathbf{x}(\mathbf{p}), \mathbf{p}), \forall \mathbf{p} \in P$, and an interval $[\mathbf{x}^L, \mathbf{x}^U] \equiv X \in \mathbb{IR}^{n_x}$ is known such that $\mathbf{x}(P) \subset X$ and $\mathbf{x}(\mathbf{p})$ is unique in X for all $\mathbf{p} \in P$.

a contraction mapping:

If $\phi(\cdot, \mathbf{p})$ is a contraction mapping on X for every $\mathbf{p} \in P$,

¹which are directly taken from the papers that developed the existing methods [2, 3] (not the case for [4]).

Can we learn from AD?

“Solve first, then differentiate!”²

— $\exists$ AD person, $\forall$ days

²Perform implicit differentiation after reaching the optimality/fixed-point condition.

Can we learn from AD?

AD Solve first, then differentiate!²

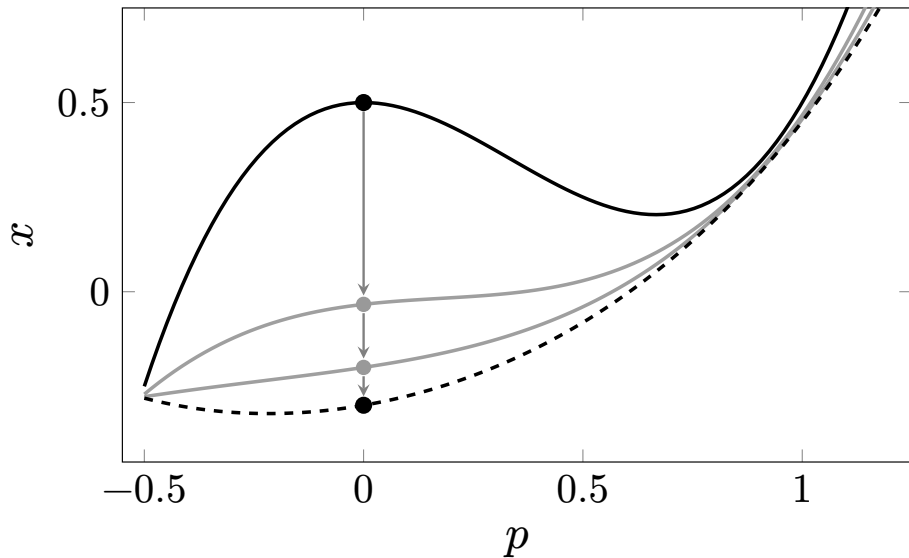
²Perform implicit differentiation after reaching the optimality/fixed-point condition.

Can we learn from AD?

AD	Solve first, then differentiate! ²
Convex Relaxations	Solve first, then relax?

²Perform implicit differentiation after reaching the optimality/fixed-point condition.

Converge-then-relax method sketch



Convex Relaxations for fixed-points

Assumption 1: There exists a $\phi : X \times P \rightarrow X$ which is a contraction mapping³.

Converge-then-relax

Require: Assumption 1 holds, $x_0 \in X$, $p \in P = [p^L, p^U]$, tolerance ϵ .

Obtain $x_\star(p)$ using ϕ $\triangleright$ through, e.g., fixed-point iteration, root-finding

$x_0^{cv} \leftarrow x_\star(p)$ $\triangleright$ lift $x_\star(p)$ into space of convex relaxations

Obtain $x_\star^{cv}$ using ϕ^{cv} , warm starting with x_0^{cv}

return $x_\star^{cv}$

³on X uniformly for every $p \in P$.

When does *converge-then-relax* work?

Key property

What are the key properties required for such an algorithm to work?

I claim:

It "only" needs to be a **uniform contraction mapping**.

Theorem (Convex-relaxed contraction mapping)

Let $\phi : X \times P \rightarrow X$ be a uniform contraction mapping, with $X = [x^L, x^U]$, $P = [p^L, p^U]$. The fixed-point mapping $p \mapsto x_\star^{cv}(p)$ of the convex envelope $\text{conv } \phi$ is a convex underestimator contraction of the fixed point mapping $p \mapsto x_\star(p)$, defined implicitly by $x_\star(p) = \phi(x_\star(p), p)$.

Meaning

$p \mapsto x_\star^{cv}(p)$ is convex,

$x_\star^{cv}(p) = (\text{conv } \phi)(x_\star^{cv}(p), p)$,

$x_\star^{cv}(\cdot)$ is unique in P ,

$x_\star^{cv}(p) \leq x_\star(p)$ for all $p \in P$.

Theorem (Convex-relaxed contraction mapping)

Let $\phi : X \times P \rightarrow X$ be a uniform contraction mapping, with $X = [x^L, x^U]$, $P = [p^L, p^U]$. The fixed-point mapping $p \mapsto x_\star^{cv}(p)$ of the convex envelope $\text{conv } \phi$ is a convex underestimator contraction of the fixed point mapping $p \mapsto x_\star(p)$, defined implicitly by $x_\star(p) = \phi(x_\star(p), p)$.

Corollary

Practical convex relaxation techniques, e.g.,

- (Generalized) McCormick Relaxations [2]
- αBB
- Taylor models
- ...

are applicable as long as they are uniform contractions in $X \times P$.

Example: Scott [2]

Consider

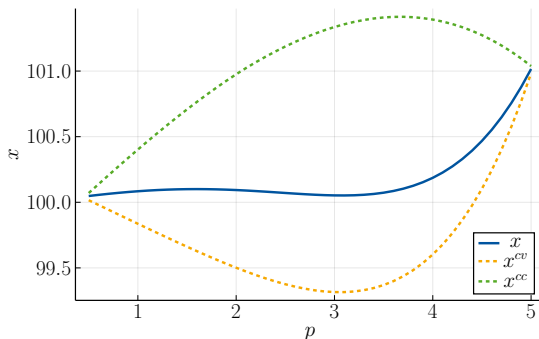
$$x \leftarrow \phi(x, p),$$

$$\phi(x, p) = \left(p - \frac{p^3}{6} + \frac{p^5}{120} \right) x^{-\frac{1}{2}} + 100$$

$$p \in P = [0.5, 5.0]$$

$$x \in X = [97.9, 103.1]$$

Same result, orders of magnitude speedup!



Example: Stuber 3.26 [3]

$$x \leftarrow \phi(p, x), \quad \phi(p, x) = -\frac{(x^2 + 4)}{p}$$

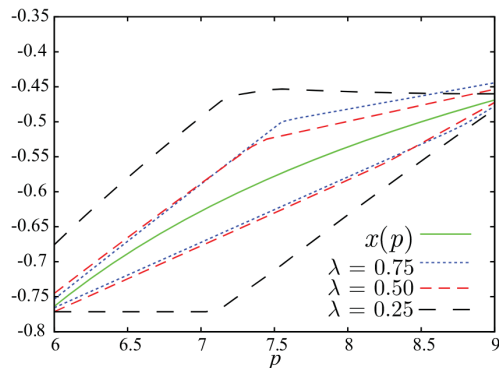
with:

$$p \in P = [6, 9],$$

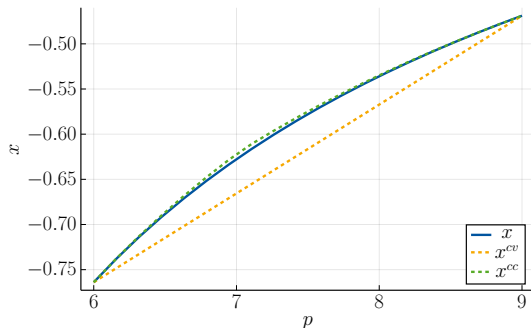
$$x \in \begin{cases} X^1 = [-0.78, -0.4] \\ X^2 = [-10.0, -5.0] \end{cases} \quad (\text{e.g., via Interval Newton})$$

Example: Stuber 3.26 [3]

Results for X^1 :



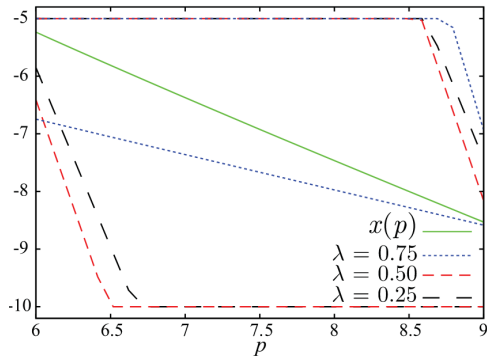
(a) Direct relaxation



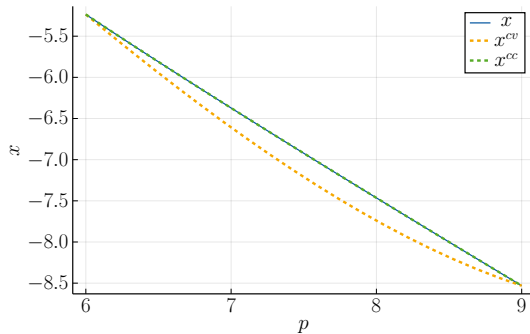
(b) Solve-then-relax

Example: Stuber 3.26 [3]

Results for X^2 :



(a) Direct relaxation



(b) Solve-then-relax

Example: Cao and Khan [4, Example 2]

Let

$$f(P, V) = \left(P + a \frac{n^2}{V^2} \right) (V - nb) - nRT = 0,$$

where

$$R = 0.0820574,$$

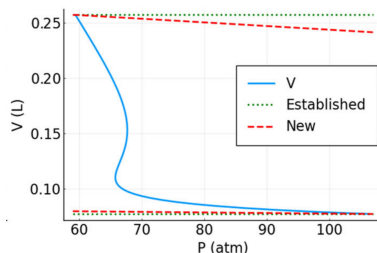
$$a = 3.610,$$

$$b = 0.0429,$$

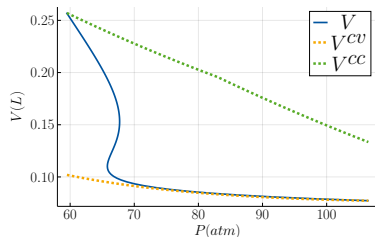
$$T = 297.77,$$

$$P = [59, 107].$$

(a) Optimization-based



(b) Solve-then-relax



...and their subgradients?

Directional Derivatives

Joint-convexity provides the existence of directional-derivatives

$$(x^{cv})'(p; h) = (\phi^{cv})'((x^{cv}(p), p); ((x^{cv})'(p; h), h)).$$

Direct tangent AD

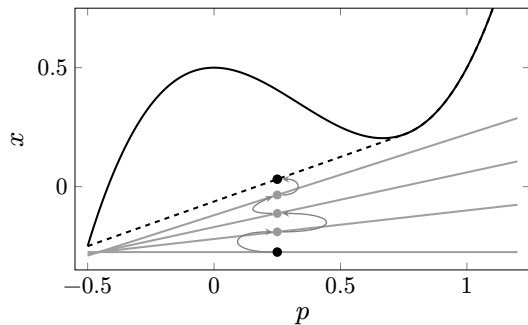
- 1: $\dot{x}_0 \leftarrow 0$
 - 2: $\dot{p} \leftarrow h$
 - 3: **repeat**
 - 4: $(x_{k+1}, \dot{x}_{k+1}) \leftarrow \dot{\phi}^{cv}((x_k, \dot{x}_k), (p, \dot{p}))$
 - 5: **until** $\|x_{k+1} - x_k\| \leq \epsilon$
 - 6: **return** $(x_k, \dot{x}_k)$
-

Solve-then-differentiate

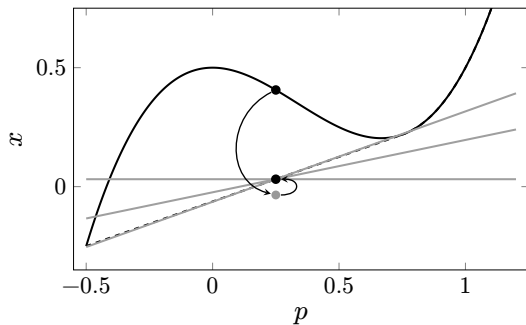
- 1: $x \leftarrow x_{\star}^{cv}(p)$
 - 2: $v_0 \leftarrow 0$
 - 3: **repeat**
 - 4: $v_{k+1} \leftarrow (\phi^{cv})'((x, p); (v_k, h))$
 - 5: **until** $\|v_{k+1} - v_k\| \leq \epsilon$
 - 6: **return** v_k
-

Given: Contraction $\phi : X \times P \rightarrow X$, $p \in P$, direction h , tolerance ϵ .

Direct vs. Fixed-point Directional Derivative



(a) Direct relaxation + tangent AD



(b) Solve-then-differentiate

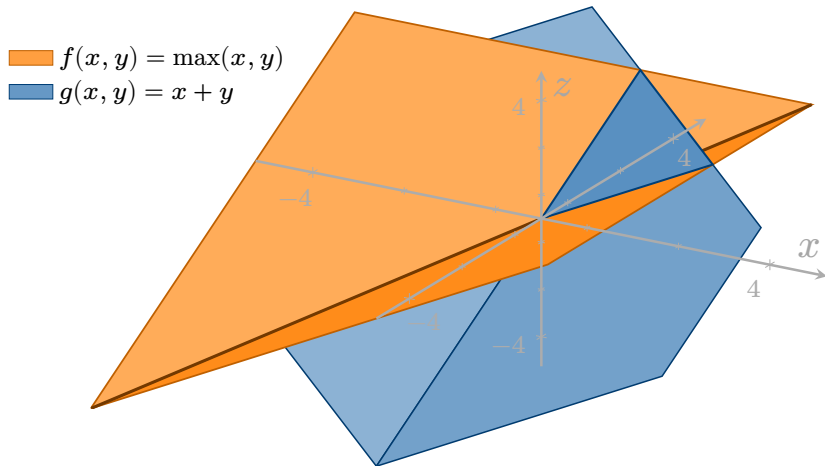
Existence of IFT

An implicit function theorem (IFT) exists not just for differentiable functions!

Sufficient condition	IFT	Convex contraction
Strict semidifferentiability	✓	✗
Strong metric (sub-)regularity	✓	✓
Lipschitz homeomorphism	✓	✓

Subgradients

Problem: In higher-dimensions, directional-derivatives $\nRightarrow$ subgradients.



Aim: Linearize the nonlinear directional derivatives in such a way that they form elements of the (B-)subdifferential (if PC^1).

Lexicographic-directional (LD-)derivatives [5]

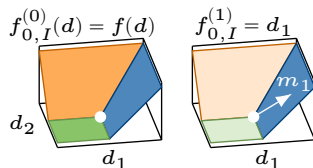
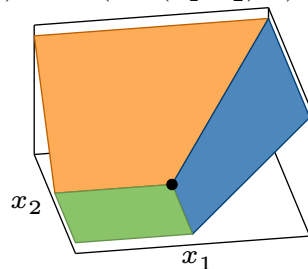
$f : X \subseteq \mathbb{R}^n \rightarrow \mathbb{R}^m$ is **L-smooth** at $x \in X$ if it is loc. Lip. cont. and dir. differentiable and for any $M = [m_1 \dots m_k] \in \mathbb{R}^{n \times k}$, we have

$$\begin{aligned} f_{x,M}^{(0)} : d &\mapsto f'(x; d) \\ f_{x,M}^{(1)} : d &\mapsto (f_{x,M}^{(0)})'(m_1; d) \\ &\vdots \\ f_{x,M}^{(k)} : d &\mapsto (f_{x,M}^{(k-1)})'(m_k; d). \end{aligned}$$

LD-derivative

$$f'(x; M) = [f_{x,M}^{(0)}(m_1) \dots f_{x,M}^{(k-1)}(m_k)]$$

$$f(x) = \max(\max(x_1, x_2), 0)$$



Lexicographic-directional (LD-)derivatives [5]

Chain rule

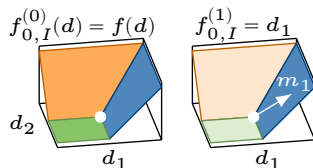
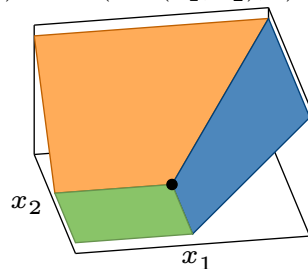
Propagation

$$(f \circ g)'(x; M) = f'(g(x); g'(x; M))$$

on individual LD-derivatives!

$\hookrightarrow$ (generalized) tangent AD [6]

$$f(x) = \max(\max(x_1, x_2), 0)$$



Lexicographic-directional (LD-)derivatives [5]

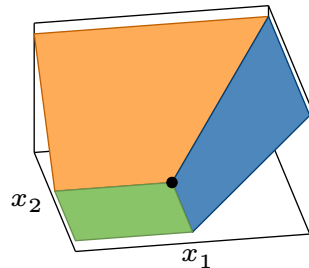
L-smooth IFT

Obtain unique solution N for seed matrix M by solving equation system

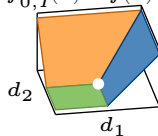
$$R'((\bar{p}, \bar{x}), (M, N)) = 0_{n_x \times k},$$

where $R(p, x) = x - \phi^{cv}(x, p)$.

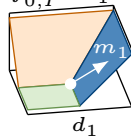
$$f(x) = \max(\max(x_1, x_2), 0)$$



$$f_{0,I}^{(0)}(d) = f(d)$$



$$f_{0,I}^{(1)} = d_1$$



Lexicographic-directional (LD-)derivatives [5]

L-smooth IFT

Obtain unique solution N for seed matrix M by solving equation system

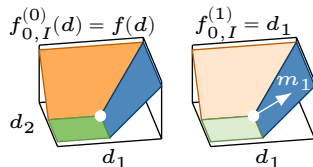
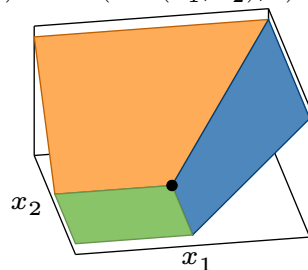
$$R'((\bar{p}, \bar{x}), (M, N)) = 0_{n_x \times k},$$

where $R(p, x) = x - \phi^{cv}(x, p)$.

Computable in practice

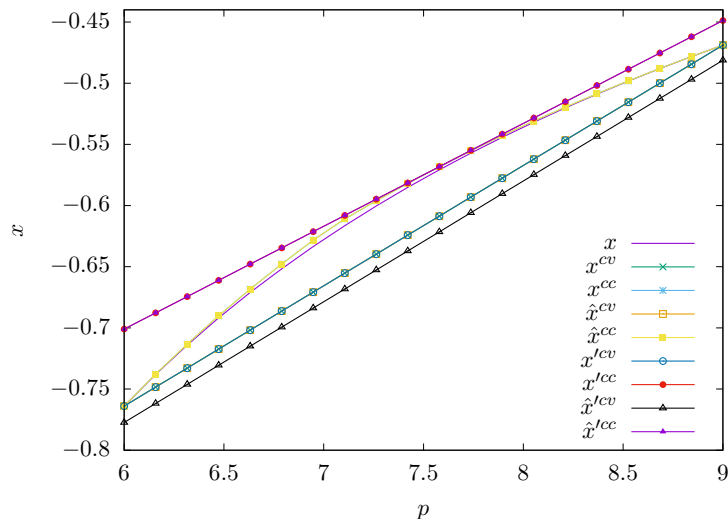
- ↪ Nonsmooth Newton [7]
- ↪ Linear solves on active selections [8]
- ↪ Branch-locking [9]

$$f(x) = \max(\max(x_1, x_2), 0)$$



Affine underestimator

Must deal with numerical error: Uses `mccormick<tangent<interval<T>>>`.



Cao and Khan [4, Ex. 1] & Stuber et al. [3, Ex. 3.26]

Takeaway

Relaxations

Converge-then-relax is a promising, potentially **faster** and **tighter** method.

Implicit Function Theorem

IFT exists not just for differentiable functions:

- ↪ Strong metric (sub-)regularity
- ↪ Lipschitz homeomorphism

Subgradients

Lexicographic-directional derivatives provide subgradients.

- ↪ Extended AD thanks to sharp chain rule!

Takeaway

Relaxations

Converge-then-relax is a promising, potentially **faster** and **tighter** method.

Implicit Function Theorem

IFT exists not just for differentiable functions:

- ↪ Strong metric (sub-)regularity
- ↪ Lipschitz homeomorphism

Subgradients

Lexicographic-directional derivatives provide subgradients.

- ↪ Extended AD thanks to sharp chain rule!

Thank you!
Questions?



↪ slides online 

References

- [1] D. Bongartz et al. *MAiNGO - McCormick-based Algorithm for mixed-integer Nonlinear Global Optimization*. Technical Report. Process Systems Engineering (AVT.SVT), RWTH Aachen University, 2018.
- [2] Joseph K. Scott et al. "Generalized McCormick relaxations". In: *Journal of Global Optimization* 51.4 (Dec. 2011), pp. 569–606.
- [3] Matthew D. Stuber et al. "Convex and concave relaxations of implicit functions". In: *Optimization Methods and Software* 30.3 (2015), pp. 424–460.
- [4] Huiyi Cao and Kamil A Khan. "General convex relaxations of implicit functions and inverse functions". In: *Journal of Global Optimization* 86.3 (2023), pp. 545–572.
- [5] Yu. Nesterov. "Lexicographic differentiation of nonsmooth functions". In: *Mathematical Programming* 104.2 (Nov. 2005), pp. 669–700.
- [6] Kamil A. Khan and Paul I. Barton. "A vector forward mode of automatic differentiation for generalized derivative evaluation". In: *Optimization Methods and Software* 30.6 (Nov. 2015), pp. 1185–1212.
- [7] Peter Stechlin et al. "Generalized Sensitivity Analysis of Nonlinear Programs". In: *SIAM Journal on Optimization* 28.1 (Jan. 2018), pp. 272–301.
- [8] Kamil A. Khan and Paul I. Barton. "Generalized Derivatives for Hybrid Systems". In: *IEEE Transactions on Automatic Control* 62.7 (July 2017), pp. 3193–3208.
- [9] Kamil A. Khan. "Branch-locking AD techniques for nonsmooth composite functions and nonsmooth implicit functions". In: *Optimization Methods and Software* (2017).