LEXICOGRAPHIC DERIVATIVES

Paul I. Barton

Process Systems Engineering Laboratory
Massachusetts Institute of Technology
Cambridge MA 02139

Lexicographic Derivatives

- $\mathbf{f}: X \subset \mathbf{R}^n \to \mathbf{R}^m$ is L-smooth at $\mathbf{x} \in X$ if it is loc. Lip. continuous and directionally differentiable, and if, for any $\mathbf{M} = \begin{bmatrix} \mathbf{m}_{(1)} & \cdots & \mathbf{m}_{(k)} \end{bmatrix} \in \mathbf{R}^{n\times k}$ the following functions exist:

$$\mathbf{f}^{(0)}_{\mathbf{x},\mathbf{M}} : \mathbf{d} \mapsto \mathbf{f}'(\mathbf{x};\mathbf{d})$$

This is the directional derivative mapping, viewed as a function of direction **d**

$$\mathbf{f}^{(1)}_{\mathbf{x},\mathbf{M}} : \mathbf{d} \mapsto [\mathbf{f}^{(0)}_{\mathbf{x},\mathbf{M}}]'(\mathbf{m}_{(1)};\mathbf{d})$$

$$\vdots$$

$$\mathbf{f}^{(k)}_{\mathbf{x},\mathbf{M}} : \mathbf{d} \mapsto [\mathbf{f}^{(k-1)}_{\mathbf{x},\mathbf{M}}]'(\mathbf{m}_{(k)};\mathbf{d})$$

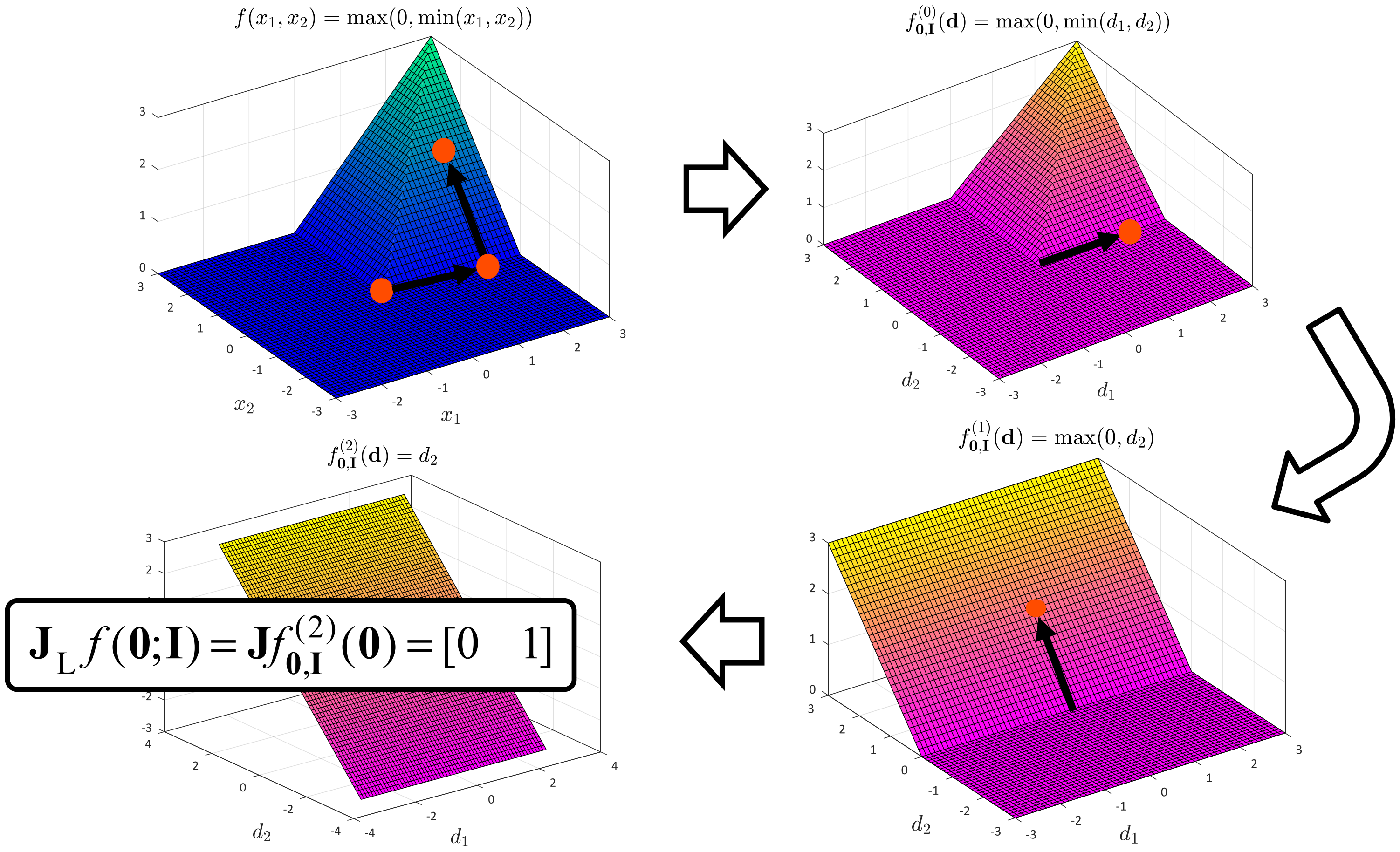
These are higher-order directional derivative mappings; directional derivatives of directional derivatives

- If the columns of $\mathbf{M}$ span $\mathbf{R}^n$, $\mathbf{f}_{\mathbf{x},\mathbf{M}}^{(k)}$ is linear
- If the columns of $\mathbf{M}$ span $\mathbf{R}^n$, the L-derivative is $\mathbf{J}_\mathrm{L}\mathbf{f}(\mathbf{x};\mathbf{M}) := \mathbf{J}\mathbf{f}^{(k)}_{\mathbf{x},\mathbf{M}}(\mathbf{0})$
- Lexicographic subdifferential: $\partial_\mathrm{L}\mathbf{f}(\mathbf{x}) := \{\mathbf{J}_\mathrm{L}\mathbf{f}(\mathbf{x};\mathbf{M}) : \mathbf{M} \in \mathbf{R}^{n\times n},\ \det \mathbf{M} \neq 0\}$

Nesterov (2005)

Lexicographic Differentiation

- Ex.: Probes local derivative information



$$\boxed{\mathbf{J}_{\mathrm{L}} f(\mathbf{0};\mathbf{I}) = \mathbf{J} f^{(2)}_{\mathbf{0},\mathbf{I}}(\mathbf{0}) = [0 \quad 1]}$$

Y. Nesterov (2005)

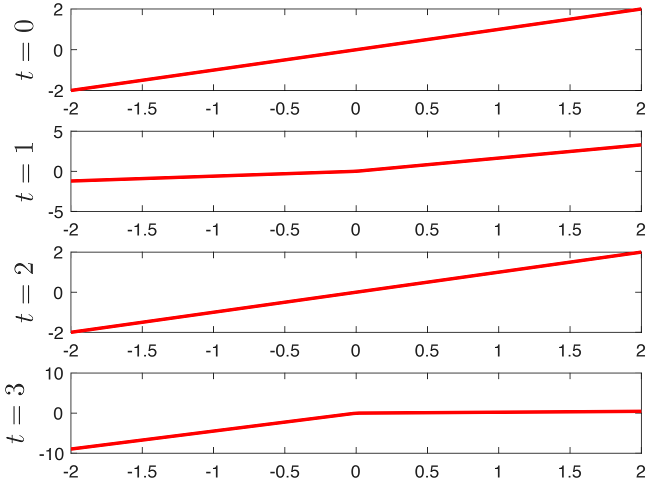
L-smooth Functions

- The following functions are L-smooth:
  - Continuously differentiable (C^1) functions
  - Piecewise differentiable (PC^r) functions
  - Convex functions (e.g. abs, 2-norm)
  - Compositions of L-smooth functions: $\mathbf{x} \mapsto \mathbf{h}(\mathbf{g}(\mathbf{x}))$
  - Integrals of L-smooth functions:

$$\mathbf{x} \mapsto \int_a^b \mathbf{g}(t,\mathbf{x})\ dt$$

  - Solutions of parametric nonsmooth ordinary differential equations (ODEs) and differential-algebraic equations (DAEs) w.r.t. parameter value
  - Solutions of optimization problems (e.g. nonlinear programs) w.r.t. parameter value
  - The list continues to grow….

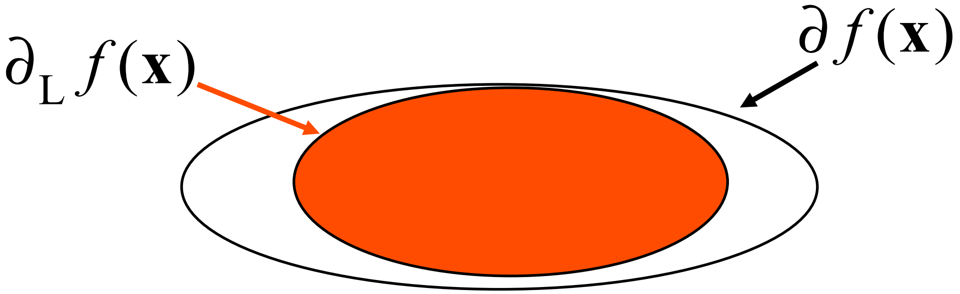
Solution of parametric DAE at snapshots in time



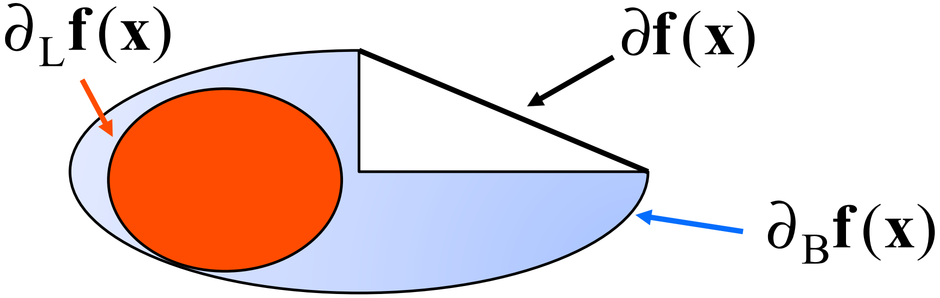
Nesterov (2005), Khan and Barton (2015), Barton et al., *Opt. Meth. & Soft.* (In Press), Stechlinski and Barton (2018)

Generalized Derivatives Landscape

- If $f : X \subset \mathbf{R}^n \to \mathbf{R}$ is an L-smooth, scalar-valued function (e.g. objective function of an optimization problem):



- If $\mathbf{f} : X \subset \mathbf{R}^n \to \mathbf{R}^m$ is PC^1:



- If $\mathbf{f} : X \subset \mathbf{R}^n \to \mathbf{R}^m$ is C^1: $\partial_{\mathrm{L}} \mathbf{f}(\mathbf{x}) = \partial_{\mathrm{B}} \mathbf{f}(\mathbf{x}) = \partial \mathbf{f}(\mathbf{x}) = \{\mathbf{J}\mathbf{f}(\mathbf{x})\}$
- If $\mathbf{f} : X \subset \mathbf{R}^n \to \mathbf{R}^m$ is L-smooth: $\{\mathbf{Ad} : \mathbf{A} \in \partial_{\mathrm{L}} \mathbf{f}(\mathbf{x})\} \subset \{\mathbf{Ad} : \mathbf{A} \in \partial \mathbf{f}(\mathbf{x})\}$ for each $\mathbf{d} \in \mathbf{R}^n$

Nesterov (2005), Khan and Barton (2015), Khan and Barton (2014)

L-smooth Functions & Lexicographic Derivatives

- Story so far:
  - A broad class of functions (PC^r, C^1, convex, all compositions, …) are L-smooth
  - Clarke Jacobian elements are computationally relevant in dedicated nonsmooth numerical methods (e.g. semismooth Newton method) but are difficult to compute automatically
  - L-derivatives are Clarke Jacobian elements (or indistinguishable from Clarke Jacobian matrix-vector products) and are therefore computationally relevant
- Question: Are L-derivatives "easy" to compute in an automated way?
- Answer: Yes! L-derivatives satisfy sharp calculus rules, expressed naturally using LD-derivatives.

Lexicographic Directional (LD-)Derivative

- Extension of classical directional derivative
- LD-derivative of L-smooth function $\mathbf{f}: X \subset \mathbf{R}^n \to \mathbf{R}^m$ at $\mathbf{x} \in X$ in the directions $\mathbf{M} = [\mathbf{m}_{(1)} \ \cdots \ \mathbf{m}_{(k)}] \in \mathbf{R}^{n \times k}$:

$$\mathbf{f}'(\mathbf{x};\mathbf{M}) = [\mathbf{f}^{(0)}_{\mathbf{x},\mathbf{M}}(\mathbf{m}_{(1)}) \ \cdots \ \mathbf{f}^{(k-1)}_{\mathbf{x},\mathbf{M}}(\mathbf{m}_{(k)})]$$

- If **M** is square and nonsingular: $\mathbf{f}'(\mathbf{x};\mathbf{M}) = \mathbf{J}_{\mathrm{L}}\mathbf{f}(\mathbf{x};\mathbf{M})\mathbf{M}$
- If **f** is differentiable at **x**: $\mathbf{f}'(\mathbf{x};\mathbf{M}) = \mathbf{J}\mathbf{f}(\mathbf{x})\mathbf{M}$
- Sharp LD-derivative chain rule: $[\mathbf{f} \circ \mathbf{g}]'(\mathbf{x};\mathbf{M}) = \mathbf{f}'(\mathbf{g}(\mathbf{x});\mathbf{g}'(\mathbf{x};\mathbf{M}))$

Khan and Barton (2015)

Computing L-Derivative from (LD-)Derivative

- Procedure to compute an L-derivative from an LD-derivative:
  1. Choose a nonsingular directions matrix $\mathbf{M}$
  2. Calculate an LD-derivative via sharp calculus rules (e.g. $[\mathbf{f} \circ \mathbf{g}]'(\mathbf{x};\mathbf{M}) = \mathbf{f}'(\mathbf{g}(\mathbf{x});\mathbf{g}'(\mathbf{x};\mathbf{M}))$)
  3. Obtain L-derivative via solving the linear equation system $\mathbf{f}'(\mathbf{x};\mathbf{M}) = \mathbf{J}_{\mathrm{L}}\mathbf{f}(\mathbf{x};\mathbf{M})\mathbf{M}$ for $\mathbf{J}_{\mathrm{L}}\mathbf{f}(\mathbf{x};\mathbf{M})$ (which is unique solution since $\mathbf{M}$ is nonsingular)

Lexicographic Directional Derivative Calculus Rules

- LD-derivative calculus rules for min, max, abs, 2-norm, etc. are based on lexicographical ordering
- Procedure is similar to putting words in alphabetical order. In fact, lexicographical ordering is also known as alphabetical ordering:

generalized inequality using lexicographical ordering

$$\begin{bmatrix}x_1\\x_2\\\vdots\\x_n\end{bmatrix} \prec \begin{bmatrix}y_1\\y_2\\\vdots\\y_n\end{bmatrix} \quad \text{if } x_1 < y_1 \text{ or } (x_1 = y_1 \text{ and } x_2 < y_2 \text{ (or } x_2 = y_2 \text{ and } x_3 < y_3 \text{ (or ...))).} \qquad \begin{bmatrix}x_1\\x_2\\\vdots\\x_n\end{bmatrix} \succeq \begin{bmatrix}y_1\\y_1\\\vdots\\y_n\end{bmatrix} \quad \text{otherwise.}$$

- Ex.

$$\begin{array}{l} \boxed{\begin{array}{ccc} 0 & & 1 \end{array}} \leftarrow 0<1 \\ \left.\begin{array}{ccc} 1 & \prec & 0 \\ 0 & & 1 \end{array}\right\} \text{irrelevant} \end{array} \qquad \begin{array}{l} \boxed{\begin{array}{ccc} 0 & & 0 \end{array}} \leftarrow \text{tie} \\ \boxed{\begin{array}{ccc} 1 & \succ & 0 \end{array}} \leftarrow 1>0 \\ \left.\begin{array}{ccc} 0 & & 1 \end{array}\right\} \text{irrelevant} \end{array} \qquad \begin{array}{l} \boxed{\begin{array}{ccc} 0 & & 0 \end{array}} \leftarrow \text{tie} \\ \boxed{\begin{array}{ccc} 0 & \prec & 0 \end{array}} \leftarrow \text{tie} \\ \boxed{\begin{array}{ccc} 0 & & 1 \end{array}} \leftarrow 0<1 \end{array}$$

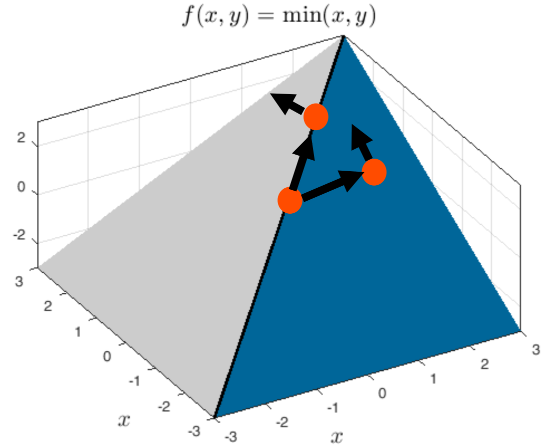
- Putting two words in alphabetical order: $\text{"about"} \prec \text{"above"}$

Lexicographic Directional Derivative Calculus Rules

- Ex. $f(\mathbf{x}) = \min(x_1, x_2)$:

$$\min'\left(\begin{bmatrix} x \\ y \end{bmatrix}; \begin{bmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{bmatrix}\right) = \begin{cases} [m_{11} \quad m_{12}], & \text{if } \begin{bmatrix} x \\ m_{11} \\ m_{12} \end{bmatrix} \prec \begin{bmatrix} y \\ m_{21} \\ m_{22} \end{bmatrix}, \\ [m_{21} \quad m_{22}], & \text{if } \begin{bmatrix} x \\ m_{11} \\ m_{12} \end{bmatrix} \succeq \begin{bmatrix} y \\ m_{21} \\ m_{22} \end{bmatrix}, \end{cases}$$

$$= \mathbf{SLmin}((x, m_{11}, m_{12}), (y, m_{12}, m_{22}))$$



$$\min'\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}; \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\right) = \begin{bmatrix} 0 & 1 \end{bmatrix}, \quad \min'\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}; \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}\right) = \begin{bmatrix} 1 & 0 \end{bmatrix}, \quad \min'\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}; \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}\right) = [1 \quad 1]$$

Lexicographic Directional Derivative Calculus Rules

- LD-Derivative calculus rules for elemental nonsmooth functions:
  - $f(x) = |x| = \mathrm{abs}(x)$:

$$f'(x;[m_1 \ \cdots \ m_k]) = \begin{cases} [m_1 \ \cdots \ m_k]^{\mathrm{T}}, & \text{if } [x \ \ m_1 \ \cdots \ m_k]^{\mathrm{T}} \succeq \mathbf{0}, \\ -[m_1 \ \cdots \ m_k]^{\mathrm{T}}, & \text{if } [x \ \ m_1 \ \cdots \ m_k]^{\mathrm{T}} \prec \mathbf{0}, \end{cases}$$

$$= \mathrm{fsign}(x, m_1, \ldots, m_k)[m_1 \ \cdots \ m_k]^{\mathrm{T}}$$

  - $f(\mathbf{x}) = \|\mathbf{x}\| = \sqrt{x_1^2 + \cdots + x_n^2}$: $f'(\mathbf{x};\mathbf{M}) = (\mathbf{fdir}([\mathbf{x} \ \ \mathbf{M}]))^{\mathrm{T}}$, where

$$\mathbf{fdir}(\mathbf{A}) = \mathbf{fdir}([\mathbf{a}_{(1)} \ \cdots \ \mathbf{a}_{(q)}]) = \begin{cases} \mathbf{0}, & \text{if } \mathbf{A} = \mathbf{0}, \\ \dfrac{\mathbf{a}_{(j^*)}}{\|\mathbf{a}_{(j^*)}\|}, \quad j^* = \min\{j : \mathbf{a}_{(j)} \neq \mathbf{0}\}, & \text{if } \mathbf{A} \neq \mathbf{0} \end{cases}$$

  - $f(\mathbf{x}) = \max(x_1, x_2)$:

$$f'\left(x, y; \begin{bmatrix} \mathbf{M}_1 \\ \mathbf{M}_2 \end{bmatrix}\right) = \mathbf{SLmax}((x, \mathbf{M}_1^{\mathrm{T}}), (y, \mathbf{M}_2^{\mathrm{T}}))$$

  - $f(\mathbf{x}) = \mathrm{mid}(x_1, x_2, x_3)$: $f'(x, y, z; \mathbf{M}) = \mathbf{SLmid}((x, \mathbf{M}_1^{\mathrm{T}}), (y, \mathbf{M}_2^{\mathrm{T}}), (z, \mathbf{M}_3^{\mathrm{T}}))$

Barton, Khan, Stechlinski and Watson, *Opt. Meth. & Soft.* (In Press)

Lexicographic Directional Derivative Calculus Rules

- LD-Derivative calculus rules for function operations:
  - Vector-valued functions: $\mathbf{u}'(\mathbf{x};\mathbf{M}) = (u'_1(\mathbf{x};\mathbf{M}), u'_2(\mathbf{x};\mathbf{M}), \ldots, u'_m(\mathbf{x};\mathbf{M}))$
  - Sums of functions: $[\mathbf{u}+\mathbf{v}]'(\mathbf{x};\mathbf{M}) = \mathbf{u}'(\mathbf{x};\mathbf{M}) + \mathbf{v}'(\mathbf{x};\mathbf{M})$
  - Products of functions: $[uv]'(\mathbf{x};\mathbf{M}) = u'(\mathbf{x};\mathbf{M})v(\mathbf{x}) + u(\mathbf{x})v'(\mathbf{x};\mathbf{M})$
  - Chain rule:
    - If **v** and **u** are L-smooth, $[\mathbf{v}\circ\mathbf{u}]'(\mathbf{x};\mathbf{M}) = \mathbf{v}'(\mathbf{u}(\mathbf{x});\mathbf{u}'(\mathbf{x};\mathbf{M}))$
    - If ψ is C^1 and **u** is L-smooth, $[\psi\circ\mathbf{u}]'(\mathbf{x};\mathbf{M}) = \mathbf{J}\psi(\mathbf{u}(\mathbf{x}))\mathbf{u}'(\mathbf{x};\mathbf{M})$
    - If **v** is L-smooth and ψ is C^1, $[\mathbf{v}\circ\psi]'(\mathbf{x};\mathbf{M}) = \mathbf{v}'(\psi(\mathbf{x});\mathbf{J}\psi(\mathbf{x})\mathbf{M})$

Nonsmooth Automatic Differentiation

- Nonsmooth AD:
  - Same underlying idea as classical AD
  - Nonsmooth AD is achieved by simply adding “nonsmooth” derivative rules (i.e. LD-derivative rules) to classical AD packages
  - …and applying the sharp chain rule

- Other remarks:
  - LD-derivative rules can be added to symbolic differentiation packages, but they still suffer from the same underlying issues outlined earlier
  - LD-derivative rules cannot be added to numerical differentiation packages in the same way; finite differencing is unsuitable for nonsmooth functions (“stepping” over nonsmooth points)

Khan and Barton (2015)

Nonsmooth AD

- Technique for calculating *exact* numerical derivatives
  - Not finite differences (no truncation error)
  - Not symbolic differentiation (no expression manipulation)
  - Applies the LD-derivative chain rule systematically to numerical values

- Ex. $y = f(\mathbf{x}) = \max(0, \min(x_1, x_2))$, at $x_1 = 0, x_2 = 0$ in directions $\mathbf{M} = \mathbf{I}$

$v_{-1} = x_1$	$v_{-1} = 0$	$\dot{\mathbf{V}}_{-1} = \dot{\mathbf{X}}_1$	$\dot{\mathbf{V}}_{-1} = [1 \quad 0]$
$v_0 = x_2$	$v_0 = 0$	$\dot{\mathbf{V}}_0 = \dot{\mathbf{X}}_2$	$\dot{\mathbf{V}}_0 = [0 \quad 1]$
$v_1 = \min(v_{-1}, v_0)$	$v_1 = 0$	$\dot{\mathbf{V}}_1 = \mathbf{SLmin}((v_{-1}, (\dot{\mathbf{V}}_{-1})^{\mathrm{T}}), (v_0, (\dot{\mathbf{V}}_0)^{\mathrm{T}}))$	$\dot{\mathbf{V}}_1 = [0 \quad 1]$
$v_2 = \max(0, v_1)$	$v_2 = 0$	$\dot{\mathbf{V}}_2 = \mathbf{SLmax}((0,0,0), (v_1, (\dot{\mathbf{V}}_1)^{\mathrm{T}}))$	$\dot{\mathbf{V}}_2 = [0 \quad 1]$
$y = v_2$	$y = 0$	$\dot{\mathbf{Y}} = \dot{\mathbf{V}}_2$	$\dot{\mathbf{Y}} = [0 \quad 1]$

LD-derivative along directions **M=I**

$f(0,0)$

$f'(0,0;\mathbf{I})$

Nonsmooth AD

- Technique for calculating *exact* numerical derivatives
  - Not finite differences (no truncation error)
  - Not symbolic differentiation (no expression manipulation)
  - Applies the LD-derivative chain rule systematically to numerical values

- Ex. $y = f(\mathbf{x}) = \max(0, \min(x_1, x_2))$, at $x_1 = 0, x_2 = 0$ in directions $\mathbf{M} = -\mathbf{I}$

$v_{-1} = x_1$	$v_{-1} = 0$	$\dot{\mathbf{V}}_{-1} = \dot{\mathbf{X}}_1$	$\dot{\mathbf{V}}_{-1} = [-1 \quad 0]$
$v_0 = x_2$	$v_0 = 0$	$\dot{\mathbf{V}}_0 = \dot{\mathbf{X}}_2$	$\dot{\mathbf{V}}_0 = [0 \quad -1]$
$v_1 = \min(v_{-1}, v_0)$	$v_1 = 0$	$\dot{\mathbf{V}}_1 = \mathbf{SLmin}((v_{-1}, (\dot{\mathbf{V}}_{-1})^{\mathrm{T}}), (v_0, (\dot{\mathbf{V}}_0)^{\mathrm{T}}))$	$\dot{\mathbf{V}}_1 = [-1 \quad 0]$
$v_2 = \max(0, v_1)$	$v_2 = 0$	$\dot{\mathbf{V}}_2 = \mathbf{SLmax}((0,0,0), (v_1, (\dot{\mathbf{V}}_1)^{\mathrm{T}}))$	$\dot{\mathbf{V}}_2 = [0 \quad 0]$
$y = v_2$	$y = 0$	$\dot{\mathbf{Y}} = \dot{\mathbf{V}}_2$	$\dot{\mathbf{Y}} = [0 \quad 0]$

LD-derivative along directions **M=-I**

$f(0,0)$

$f'(0,0;-\mathbf{I})$

Summary

- The Clarke Jacobian is a computationally relevant generalized derivative, but is generally difficult to compute in an automated way
- L-derivatives are attractive for several reasons:
  - The class of L-smooth functions is broad (includes C1, PC1, convex functions and **all** compositions)
  - L-derivatives are computationally relevant (i.e. can be supplied to dedicated nonsmooth methods)
  - L-derivatives can be computed in an automated way thanks to sharp calculus rules and nonsmooth automatic differentiation
- LD-derivatives can be computed for singular (or even nonsquare) directions matrices. This is crucial for compositions of problems; e.g. dynamic systems with optimization problems embedded or vice versa

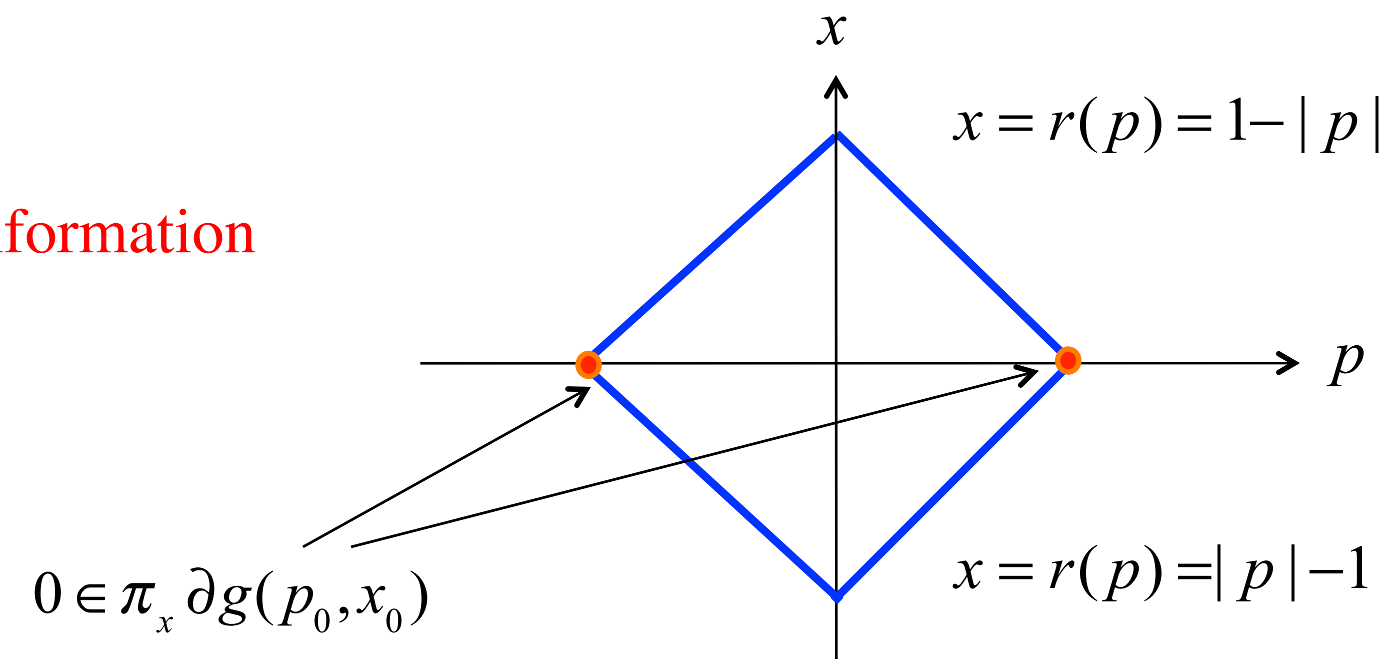
SENSITIVITY ANALYSIS OF NONSMOOTH IMPLICIT FUNCTIONS

Clarke Jacobian Implicit Function Theorem Revisited

- If $\mathbf{g}: P \times X \subset \mathbf{R}^p \times \mathbf{R}^n \to \mathbf{R}^n$ is a loc. Lip. cts. function s.t. $\mathbf{g}(\mathbf{p}_0, \mathbf{x}_0) = \mathbf{0}$ and $\det \mathbf{X} \neq 0$ for all $\mathbf{X} \in \pi_{\mathbf{x}} \partial \mathbf{g}(\mathbf{p}_0, \mathbf{x}_0) = \{\mathbf{X} \in \mathbf{R}^{n \times n} : [\mathbf{Q} \quad \mathbf{X}] \in \partial \mathbf{g}(\mathbf{p}_0, \mathbf{x}_0)\}$ then there exists a Lip. cts. (implicit) function $\mathbf{r}$ such that $\mathbf{g}(\mathbf{p}, \mathbf{r}(\mathbf{p})) = \mathbf{0}$ near $\mathbf{p} = \mathbf{p}_0$

- Ex. $|p| + |x| = 1$

No derivative (sensitivity) information



F. H. Clarke, 1990. Optimization and Nonsmooth Analysis. Philadelphia, PA: SIAM.

L-Smooth Implicit Function Theorem

- If $\mathbf{g}: P \times X \subset \mathbf{R}^p \times \mathbf{R}^n \to \mathbf{R}^n$ is an L-smooth function s.t. $\mathbf{g}(\mathbf{p}_0, \mathbf{x}_0) = \mathbf{0}$ and $\det \mathbf{X} \neq 0$ for all $\mathbf{X} \in \pi_{\mathbf{x}} \partial \mathbf{g}(\mathbf{p}_0, \mathbf{x}_0) = \{\mathbf{X} \in \mathbf{R}^{n \times n} : [\mathbf{Q} \quad \mathbf{X}] \in \partial \mathbf{g}(\mathbf{p}_0, \mathbf{x}_0)\}$ then there exists an L-smooth (implicit) function $\mathbf{r}$ such that $\mathbf{g}(\mathbf{p}, \mathbf{r}(\mathbf{p})) = \mathbf{0}$ near $\mathbf{p} = \mathbf{p}_0$ and for any $\mathbf{P}$, $\mathbf{r}'(\mathbf{p}_0; \mathbf{P}) \equiv \mathbf{X}$ is the solution of

$$\boxed{\mathbf{g}'(\mathbf{p}_0, \mathbf{x}_0; (\mathbf{P}, \mathbf{X})) = \mathbf{0}}$$

Nonsmooth sensitivity system

- Remarks:
  - The matrix $\mathbf{P}$ is the directions matrix
  - Sensitivity system provides generalized derivative information for implicit function $\mathbf{r}$
  - Sensitivity system is nonsmooth (and thus nonlinear), but has a unique solution for any $\mathbf{P}$
  - Computing solution of sensitivity system is practically implementable (more in a bit)

Khan and Barton, *IEEE TAC*. 62 (2017)

Implicit Function Sensitivities: Smooth vs. Nonsmooth

- Smooth sensitivity system:

$$\boxed{\begin{aligned}&\frac{\partial \mathbf{g}}{\partial \mathbf{p}}(\mathbf{p}_0,\mathbf{x}_0)+\frac{\partial \mathbf{g}}{\partial \mathbf{x}}(\mathbf{p}_0,\mathbf{x}_0)\mathbf{X}=\mathbf{0}\\ &\text{s.t.}\quad \mathbf{X}\equiv \mathbf{Jr}(\mathbf{p}_0)\end{aligned}}$$

  - Linear equation system
  - Unique solution given that

$$\det \frac{\partial \mathbf{g}}{\partial \mathbf{x}}(\mathbf{p}_0,\mathbf{x}_0)\neq 0$$

  - Efficient methods for numerical computation

- Nonsmooth sensitivity system:

$$\boxed{\begin{aligned}&\mathbf{g}'(\mathbf{p}_0,\mathbf{x}_0;(\mathbf{P},\mathbf{X}))=\mathbf{0}\\ &\text{s.t.}\ \ \mathbf{X}\equiv \mathbf{r}'(\mathbf{p}_0;\mathbf{P})\end{aligned}}$$

  - Nonsmooth equation system
  - Unique solution given that $\det \mathbf{X}\neq 0$ for all $\mathbf{X}\in\{\mathbf{X}\in\mathbf{R}^{n\times n}: [\mathbf{P}\quad \mathbf{X}]\in\partial\mathbf{g}(\mathbf{p}_0,\mathbf{x}_0)\}$
  - If $\mathbf{g}$ is PC[1], above condition can be replaced by $\mathrm{sign}(\det \mathbf{X})=\pm 1$ for all

$$\mathbf{X}\in\{\mathbf{X}\in\mathbf{R}^{n\times n}: \mathbf{X}_j=\frac{\partial g_{(\delta_i),j}}{\partial \mathbf{x}}(\mathbf{p}_0,\mathbf{x}_0), \delta\in\{1,\ldots,n_{\text{ess}}\}^{|n_{\text{ess}}|}\}$$

  - Practically implementable methods for numerical computation (up next)

Numerical Solution of Nonsmooth Sensitivity System

- Compute solution $\mathbf{X} \equiv \mathbf{r}'(\mathbf{p}_0;\mathbf{P})$ of $\mathbf{g}'(\mathbf{p}_0,\mathbf{x}_0;(\mathbf{P},\mathbf{X})) = \mathbf{0}$ two ways

1. Classical linear equation system: $\frac{\partial \mathbf{g}_{(i)}}{\partial \mathbf{p}}(\mathbf{p}_0,\mathbf{x}_0)\mathbf{P} + \frac{\partial \mathbf{g}_{(i)}}{\partial \mathbf{x}}(\mathbf{p}_0,\mathbf{x}_0)\mathbf{X} = \mathbf{0}$
   - Cycle through essentially active selection functions satisfying $\det \frac{\partial \mathbf{g}_{(i)}}{\partial \mathbf{x}}(\mathbf{p}_0,\mathbf{x}_0) \neq 0$
   - Verify solution: check if $\mathbf{g}'(\mathbf{p}_0,\mathbf{x}_0;(\mathbf{P},\mathbf{X})) = \mathbf{0}$, otherwise choose new selection fn.
   - Can apply efficient solvers and use techniques such as iterative refinement
   - Only possible if **g** is PC^1
   - Worst-case computational cost: solving n_{ess} linear equation systems

2. Nonsmooth equation system: $\mathbf{g}'(\mathbf{p}_0,\mathbf{x}_0;(\mathbf{P},\mathbf{X})) = \mathbf{0}$
   - Can apply dedicated nonsmooth equation-solving methods (e.g. nonsmooth Newton's method or LP-Newton method)
   - Can apply recently developed branch-locking techniques (Khan, OM&S, 2017) when solving the system columnwise
   - Computational cost unclear at present

Khan and Barton, *IEEE TAC*. 62 (2017); Khan, *OM&S* (In Press)

Summary

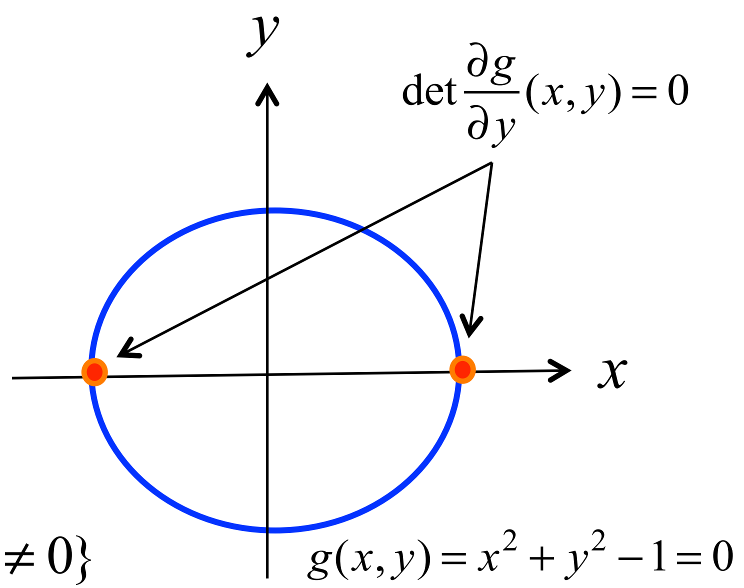
- The L-smooth Implicit Function Theorem augments the Clarke Jacobian Implicit Function Theorem with generalized derivative information
- The nonsmooth sensitivity system is nonlinear but has a unique solution from which an L-derivative can be computed (given a nonsingular directions matrix)
- Practically implementable methods are available to compute the solution of the nonsmooth sensitivity system

NONSMOOTH DIFFERENTIAL EQUATIONS

Differential-Algebraic Equations

- Consider the semi-explicit differential-algebraic equations (DAEs):

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{y}(t))$$
$$\mathbf{0} = \mathbf{g}(\mathbf{x}(t), \mathbf{y}(t))$$
$$\mathbf{x}(t_0) = \mathbf{x}_0$$



  - Consistent initialization: $\mathbf{0} = \mathbf{g}(\mathbf{x}_0, \mathbf{y}_0)$
  - Consistency set: $(\mathbf{x}(t), \mathbf{y}(t)) \in \mathrm{G}_C = \{(\mathbf{x}, \mathbf{y}) : \mathbf{g}(\mathbf{x}, \mathbf{y}) = \mathbf{0}\}$
  - Regularity set (index-1): $(\mathbf{x}(t), \mathbf{y}(t)) \in \mathrm{G}_R = \{(\mathbf{x}, \mathbf{y}) : \det \frac{\partial \mathbf{g}}{\partial \mathbf{y}}(\mathbf{x}, \mathbf{y}) \neq 0\}$
  - Underlying ODE: $\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{y}(t))$

nonsingular equivalent to differentiation index 1

$$\dot{\mathbf{y}}(t) = -\left(\frac{\partial \mathbf{g}}{\partial \mathbf{y}}(\mathbf{x}(t), \mathbf{y}(t))\right)^{-1} \frac{\partial \mathbf{g}}{\partial \mathbf{x}}(\mathbf{x}(t), \mathbf{y}(t))\, \mathbf{f}(\mathbf{x}(t), \mathbf{y}(t))$$

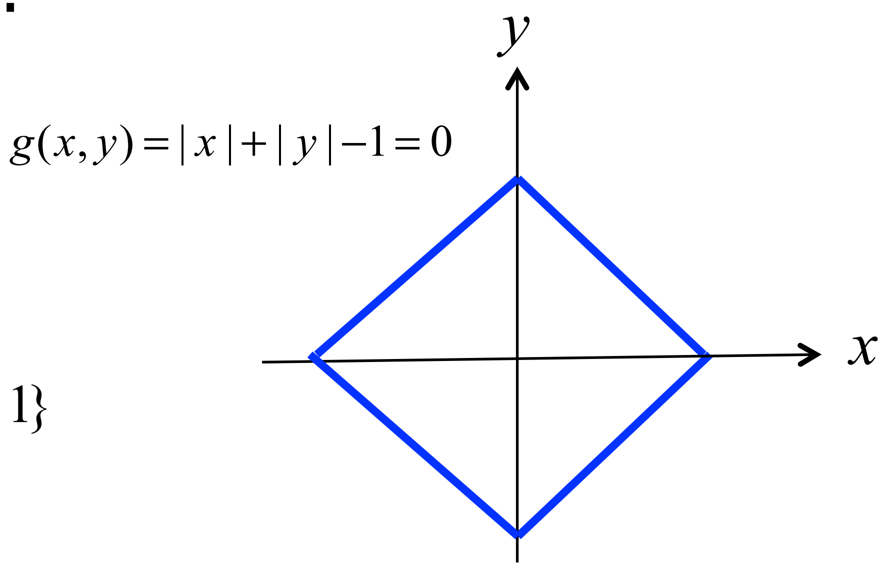
- Note: ODEs are a special case of DAEs

Kunkel and Mehrmann, 2006. EMS; Brenan, Campbell and Petzold, 1996. SIAM; Scott and Barton, Numerische Mathematik, 125 (2013)

Nonsmooth DAEs

- Consider the following nonsmooth DAEs:

$$\dot{x}(t) = 1$$
$$1 = |x(t)| + |y(t)|$$
$$x(t_0) = x_0$$



  - Consistent initialization: $1 = |x_0| + |y_0|$
  - Consistency set: $(x(t), y(t)) \in G_C = \{(x,y) : |x| + |y| = 1\}$
  - Regularity set (index-1): $(x(t), y(t)) \in G_R = ??$
  - Underlying ODE: $??$

- Classical index-1 DAE theory is established via implicit function theorem and classical ODE theory
- Idea: apply nonsmooth implicit function theorem and nonsmooth ODE theory (well established) to nonsmooth DAE

Well-Posedness of Nonsmooth DAEs

- Nonsmooth semi-explicit DAEs:

$$\dot{\mathbf{x}}(t) = \mathbf{f}(t, \mathbf{x}(t), \mathbf{y}(t))$$
$$\mathbf{0} = \mathbf{g}(t, \mathbf{x}(t), \mathbf{y}(t))$$
$$\mathbf{x}(t_0) = \mathbf{x}_0$$

  - **f** is discontinuous w.r.t. *t*, continuous w.r.t. $\mathbf{x}, \mathbf{y}$, and **g** is locally Lipschitz
  - Consistency set $(t, \mathbf{x}(t), \mathbf{y}(t)) \in G_C = \{(t, \mathbf{x}, \mathbf{y}) : \mathbf{g}(t, \mathbf{x}, \mathbf{y}) = \mathbf{0}\}$
  - Regularity set (generalized differentiation index-1):

$$(t, \mathbf{x}(t), \mathbf{y}(t)) \in G_R = \{(t, \mathbf{x}, \mathbf{y}) : \det \mathbf{Y} \neq 0, \text{ for all } \mathbf{Y} \in \pi_{\mathbf{y}} \partial \mathbf{g}(t, \mathbf{x}, \mathbf{y})\}$$

nonsmooth implicit function theorem can be applied

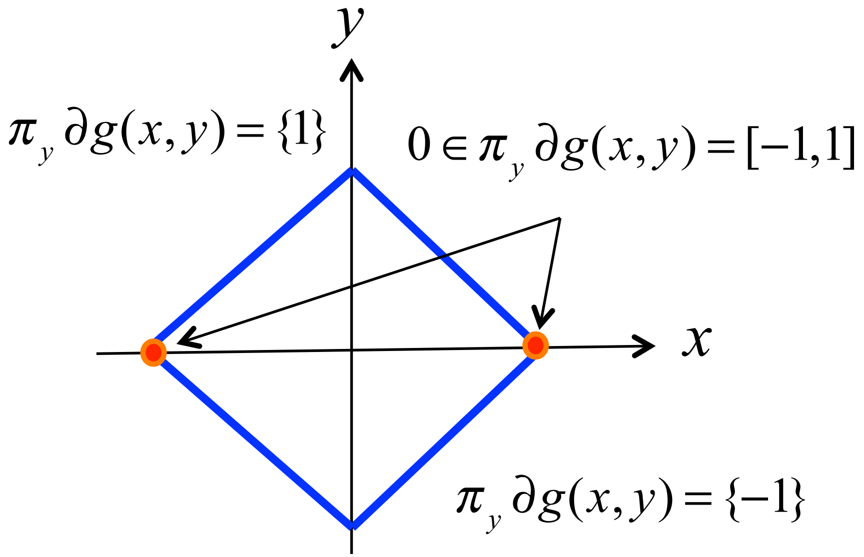
- Well-posedness results:
  - Existence of (local) solutions: $(\mathbf{x}_0, \mathbf{y}_0) \in G_C \cap G_R$
  - Uniqueness of a solution: $\{(\mathbf{x}(t), \mathbf{y}(t)) : t \in T\} \subset G_C \cap G_R$ and **f** locally Lipschitz
  - Continuation of solutions: a regular solution (i.e. generalized diff. index-1) can be extended

Stechlinski and Barton, *Journal Diff. Eqns*. 262 (2017)

Well-Posedness of Nonsmooth DAEs

- Ex. continued:

$$\dot{x}(t) = 1 \qquad 1 = |x(t)| + |y(t)| \qquad x(t_0) = x_0 \quad \Rightarrow \quad \pi_y \partial g(x,y) = \begin{cases} \{-1\}, & \text{if } y < 0 \\ [-1,1], & \text{if } y = 0 \\ \{1\}, & \text{if } y > 0 \end{cases}$$



  - **f** is PC[1] and **g** is PC[1]
  - Consistency set: $G_C = \{(x,y) : |x| + |y| = 1\}$
  - Regularity set: $G_R = \{(x,y) : y \neq 0\}$

- Existence and uniqueness of a "regular" solution:

$$(x_0, y_0) \in G_C \cap G_R = \{(x,y) : |x_0| + |y_0| = 1,\ y_0 \neq 0\}$$

- Indeed, unique regular solution is $(x(t), y(t)) = \begin{cases} \left(t + x_0, 1 - \left|t + x_0\right|\right), & \text{if } y_0 > 0, \\ \left(t + x_0, -1 + \left|t + x_0\right|\right), & \text{if } y_0 < 0, \end{cases}$

Stechlinski and Barton, *Journal Diff. Eqns*. 262 (2017)

Dependence of Solutions of Nonsmooth DAEs on Parameters

- Nonsmooth semi-explicit DAEs:

$$\begin{aligned}
\dot{\mathbf{x}}(t,\mathbf{p}) &= \mathbf{f}(t,\mathbf{p},\mathbf{x}(t,\mathbf{p}),\mathbf{y}(t,\mathbf{p}))\\
\mathbf{0} &= \mathbf{g}(t,\mathbf{p},\mathbf{x}(t,\mathbf{p}),\mathbf{y}(t,\mathbf{p}))\\
\mathbf{x}(t_0,\mathbf{p}) &= \mathbf{f}_0(\mathbf{p})
\end{aligned}$$

  - Consistency set: $(t,\mathbf{p},\mathbf{x}(t,\mathbf{p}),\mathbf{y}(t,\mathbf{p})) \in \mathrm{G}_C = \{(t,\mathbf{p},\mathbf{x},\mathbf{y}) : \mathbf{g}(t,\mathbf{p},\mathbf{x},\mathbf{y}) = \mathbf{0}\}$
  - Regularity set (generalized differentiation index-1):

$$(t,\mathbf{p},\mathbf{x}(t,\mathbf{p}),\mathbf{y}(t,\mathbf{p})) \in \mathrm{G}_R = \{(t,\mathbf{p},\mathbf{x},\mathbf{y}) : \det\mathbf{Y} \neq 0, \text{ for all } \mathbf{Y} \in \pi_{\mathbf{y}}\,\partial\mathbf{g}(t,\mathbf{p},\mathbf{x},\mathbf{y})\}$$

- A regular solution $(\mathbf{x}(t,\mathbf{p}),\mathbf{y}(t,\mathbf{p}))$ is:
  - Continuous w.r.t. **p** if **f** is cts. and **g** is locally Lipschitz
  - Lipschitz w.r.t. **p** if **f** is locally Lipschitz and **g** is locally Lipschitz
  - L-smooth w.r.t. **p** if **f** is L-smooth and **g** is L-smooth

can we calculate LD-derivatives?...

Stechlinski and Barton, *Journal Diff. Eqns*. 262 (2017)

Smooth DAEs
Classical Dynamic Sensitivities

- Smooth semi-explicit DAEs:

$$\dot{\mathbf{x}}(t,\mathbf{p}) = \mathbf{f}(t,\mathbf{p},\mathbf{x}(t,\mathbf{p}),\mathbf{y}(t,\mathbf{p}))$$
$$\mathbf{0} = \mathbf{g}(t,\mathbf{p},\mathbf{x}(t,\mathbf{p}),\mathbf{y}(t,\mathbf{p}))$$
$$\mathbf{x}(t_0,\mathbf{p}) = \mathbf{f}_0(\mathbf{p})$$

- A regular solution $(\mathbf{x}(t,\mathbf{p}),\mathbf{y}(t,\mathbf{p}))$ is C^1 w.r.t. **p** (from diff. index-1)

- Sensitivities: $\mathbf{s}_x \equiv \dfrac{\partial \mathbf{x}}{\partial \mathbf{p}}, \ \mathbf{s}_y \equiv \dfrac{\partial \mathbf{y}}{\partial \mathbf{p}}$

Dynamic Optimization of Smooth DAEs

- Sequential approach (e.g. single or multiple shooting):

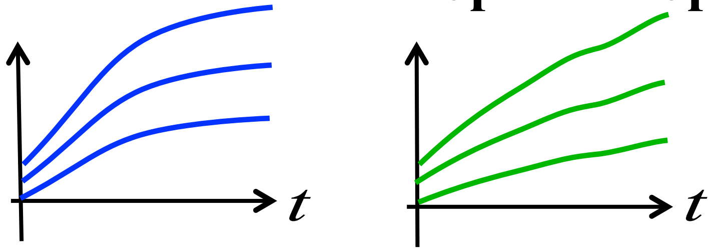
Semi-explicit index-1 DAEs

$$\dot{\mathbf{x}}(t,\mathbf{p}) = \mathbf{f}(t,\mathbf{p},\mathbf{x}(t,\mathbf{p}),\mathbf{y}(t,\mathbf{p}))$$
$$\mathbf{0} = \mathbf{g}(t,\mathbf{p},\mathbf{x}(t,\mathbf{p}),\mathbf{y}(t,\mathbf{p}))$$

Linear sensitivity DAEs

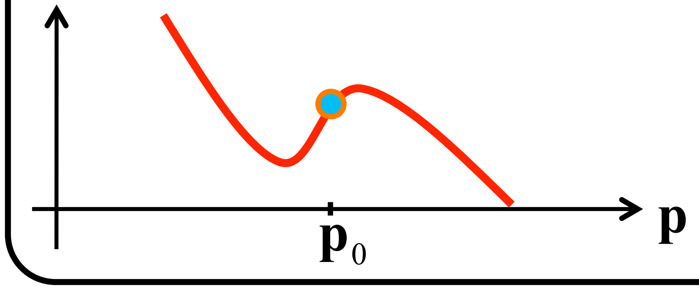
$$\dot{\mathbf{s}}_x = \frac{\partial \mathbf{f}}{\partial \mathbf{p}} + \frac{\partial \mathbf{f}}{\partial \mathbf{x}}\mathbf{s}_x + \frac{\partial \mathbf{f}}{\partial \mathbf{y}}\mathbf{s}_y$$
$$\mathbf{0} = \frac{\partial \mathbf{g}}{\partial \mathbf{p}} + \frac{\partial \mathbf{g}}{\partial \mathbf{x}}\mathbf{s}_x + \frac{\partial \mathbf{g}}{\partial \mathbf{y}}\mathbf{s}_y$$

$\mathbf{x}(t,\mathbf{p}_0),\mathbf{y}(t,\mathbf{p}_0)$ $\quad \mathbf{s}_x = \frac{\partial \mathbf{x}}{\partial \mathbf{p}},\ \mathbf{s}_y = \frac{\partial \mathbf{y}}{\partial \mathbf{p}}$



Update **p** via optimization

$\phi(\mathbf{x}(t_f,\mathbf{p}),\mathbf{y}(t_f,\mathbf{p}))$



Dynamic Optimization of Nonsmooth DAEs

- Sequential approach in nonsmooth setting:

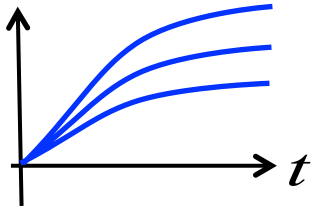
Semi-explicit index-1 DAEs

$$\dot{\mathbf{x}}(t,\mathbf{p}) = \mathbf{f}(t,\mathbf{p},\mathbf{x}(t,\mathbf{p}),\mathbf{y}(t,\mathbf{p}))$$
$$\mathbf{0} = \mathbf{g}(t,\mathbf{p},\mathbf{x}(t,\mathbf{p}),\mathbf{y}(t,\mathbf{p}))$$

Sensitivity DAEs

$$\cancel{\dot{\mathbf{s}}_x = \frac{\partial \mathbf{f}}{\partial \mathbf{p}} + \frac{\partial \mathbf{f}}{\partial \mathbf{x}}\mathbf{s}_x + \frac{\partial \mathbf{f}}{\partial \mathbf{y}}\mathbf{s}_y}$$
$$\cancel{\mathbf{0} = \frac{\partial \mathbf{g}}{\partial \mathbf{p}} + \frac{\partial \mathbf{g}}{\partial \mathbf{x}}\mathbf{s}_x + \frac{\partial \mathbf{g}}{\partial \mathbf{y}}\mathbf{s}_y}$$

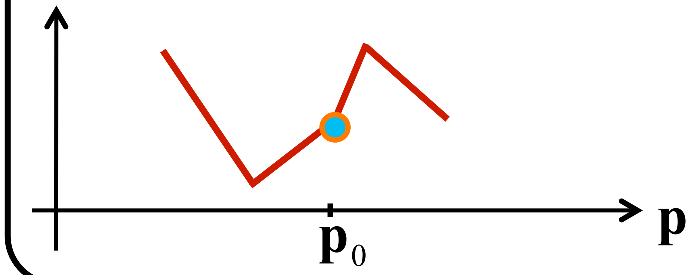
$\mathbf{x}(t,\mathbf{p}_0),\mathbf{y}(t,\mathbf{p}_0)$ $\quad \cancel{\mathbf{s}_x = \frac{\partial \mathbf{x}}{\partial \mathbf{p}},\ \mathbf{s}_y = \frac{\partial \mathbf{y}}{\partial \mathbf{p}}}$



???

Update **p** via optimization

$\phi(\mathbf{x}(t_f,\mathbf{p}),\mathbf{y}(t_f,\mathbf{p}))$



Nonsmooth ODEs
Classical Dynamic Sensitivities

- Nonsmooth ODE case:

$$\dot{\mathbf{x}}(t,\mathbf{p}) = \mathbf{f}(t,\mathbf{x}(t,\mathbf{p}))$$
$$\mathbf{x}(t_0,\mathbf{p}) = \mathbf{p}$$

- Goal: given reference parameter $\mathbf{p}_0$, characterize (local) sensitivity information by computing element of $\partial[\mathbf{x}(t,\cdot)](\mathbf{p}_0)$
- Linear Newton Approximation (Pang & Stewart, 2009; Clarke, 1980):

$$\boxed{\Gamma(\tau) = \mathrm{conv}\left\{\mathbf{X}(\tau) : \dot{\mathbf{X}}(t) \in \partial[\mathbf{f}_t](\mathbf{x}(t,\mathbf{p}_0))\mathbf{X}(t);\, \mathbf{X}(0) = \mathbf{I}\right\}}$$

  - Pros: relatively easy to evaluate
  - Cons: Satisfies $\partial[\mathbf{x}(t,\cdot)](\mathbf{p}_0) \subset \Gamma(t)$; does not reduce to derivative when $\mathbf{x}(t,\cdot)$ is C^1; does not reduce to subdifferential when $\mathbf{x}(t,\cdot)$ is convex; no sufficient optimality condition

Nonsmooth ODEs
Classical Dynamic Sensitivities

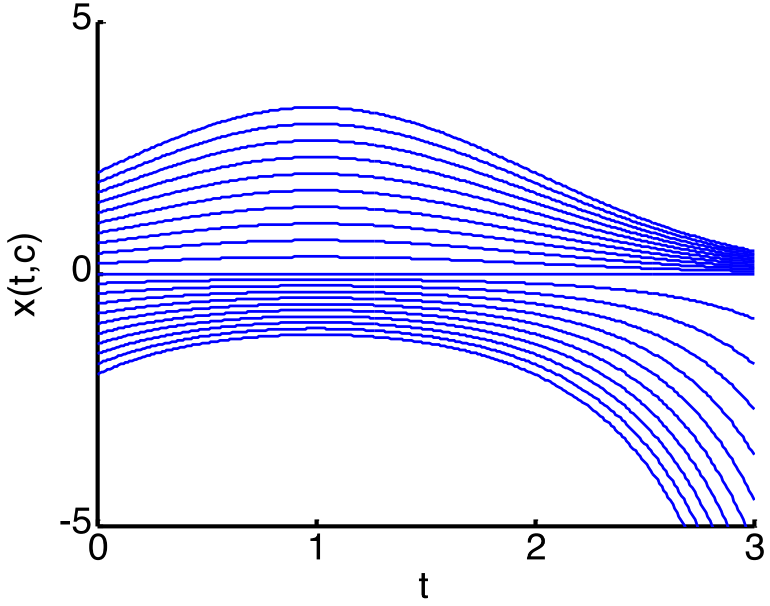
- Linear Newton Approximation (LNA):

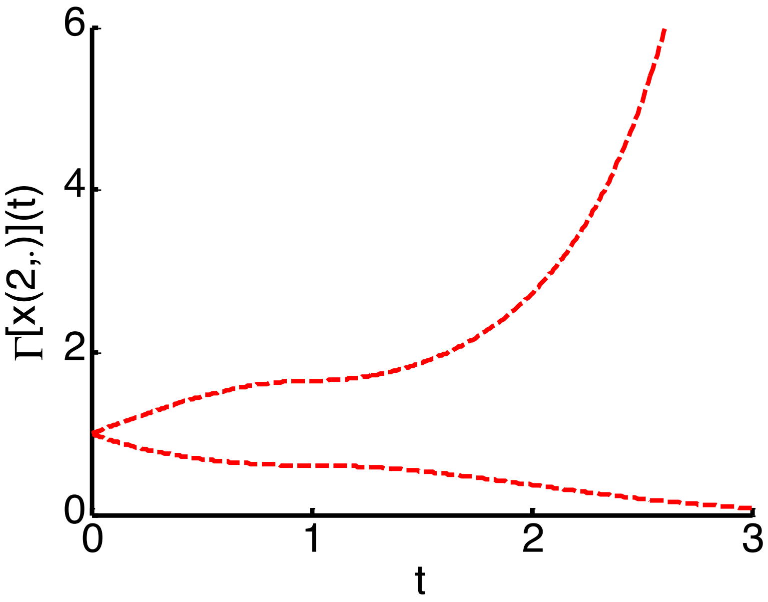
$$\Gamma(\tau) = \mathrm{conv}\left\{\mathbf{X}(\tau) : \dot{\mathbf{X}}(t) \in \partial[\mathbf{f}_t](\mathbf{x}(t,\mathbf{p}_0))\mathbf{X}(t);\ \mathbf{X}(0) = \mathbf{I}\right\}$$

- Ex. $\dot{x}(t,p) = (1-t)\,|\,x(t,p)\,|$

  $x(0,p) = p$

  - The solution $x(2,\cdot)$ is C^1 and convex w.r.t. p at p=0
  - The LNA is calculated as $\Gamma[2(t,\cdot)](0) = [1/e, e]$, but $\partial[x(2,\cdot)](0) = \{1\} = \left\{\frac{\partial x}{\partial p}(2,0)\right\}$





Nonsmooth ODEs
Dynamic LD-Derivatives

- Nonsmooth ODEs:

$$\dot{\mathbf{x}}(t,\mathbf{p}) = \mathbf{f}(t,\mathbf{p},\mathbf{x}(t,\mathbf{p}))$$
$$\mathbf{x}(t_0,\mathbf{p}) = \mathbf{f}_0(\mathbf{p})$$

  - If **f** and $\mathbf{f}_0$ are L-smooth functions, then $\mathbf{x}(t,\mathbf{p})$ is L-smooth w.r.t. **p**

- Nonsmooth ODE sensitivity system:

$$\boxed{\dot{\mathbf{X}}(t) = [\mathbf{f}_t]'(\mathbf{p}_0,\mathbf{x}(t,\mathbf{p}_0);(\mathbf{M},\mathbf{X}(t))), \quad \mathbf{X}(0) = \mathbf{f}_0{}'(\mathbf{p}_0;\mathbf{M})}$$

  - LD-derivative mapping $t \mapsto [\mathbf{x}(t,\cdot)]'(\mathbf{p}_0;\mathbf{M})$ is unique solution of sensitivity system
  - If **M** is nonsingular, then an L-derivative can be computed for any t via the linear equation system $\mathbf{X}(t) = \mathbf{J}_\mathrm{L}[\mathbf{x}(t,\cdot)](\mathbf{p}_0;\mathbf{M})\mathbf{M}$
  - If **f** and $\mathbf{f}_0$ are C^1 and **M**=**I** then the classical sensitivity system is recovered:

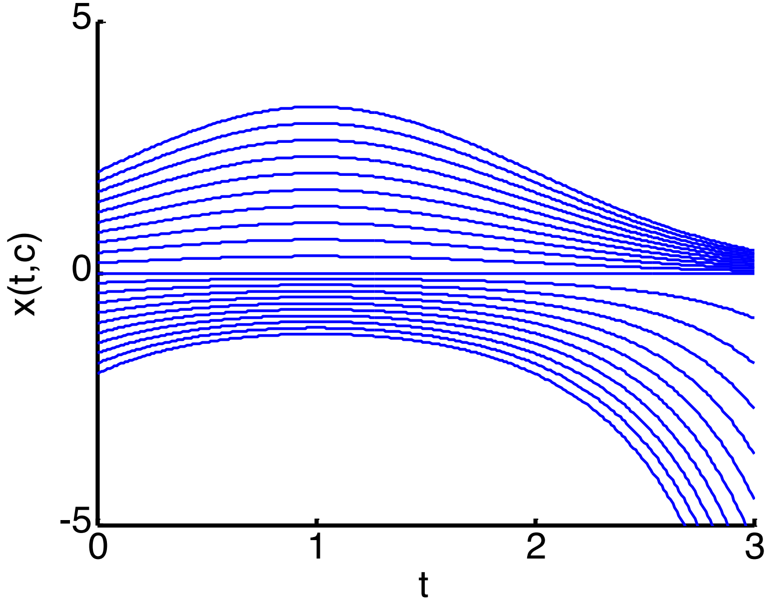
$$\dot{\mathbf{X}}(t) = [\mathbf{f}_t]'(\mathbf{p}_0,\mathbf{x}(t,\mathbf{p}_0);(\mathbf{I},\mathbf{X}(t))) = \mathbf{J}\mathbf{f}_t(\mathbf{p}_0,\mathbf{x}(t,\mathbf{p}_0))\begin{bmatrix}\mathbf{I}\\ \mathbf{X}(t)\end{bmatrix} = \frac{\partial \mathbf{f}}{\partial \mathbf{p}}(t,\mathbf{p}_0,\mathbf{x}(t,\mathbf{p}_0)) + \frac{\partial \mathbf{f}}{\partial \mathbf{x}}(t,\mathbf{p}_0,\mathbf{x}(t,\mathbf{p}_0))\mathbf{X}(t)$$

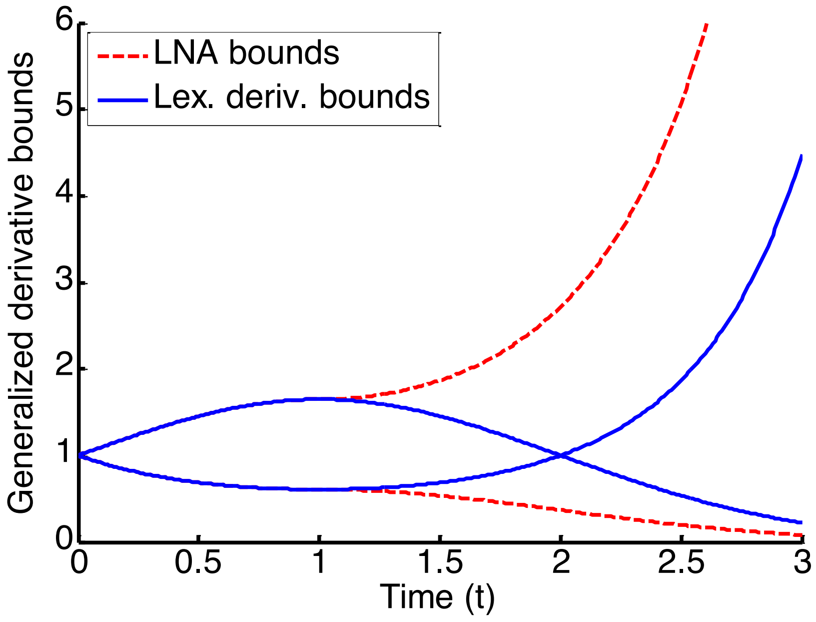
Khan and Barton, *Journal Opt. Theory Appl.* 163 (2014)

Nonsmooth ODEs
Dynamic LD-Derivatives

- Ex. continued: $\dot{x}(t,p) = (1-t)\,|x(t,p)|$
  $x(0,p) = p$
  - Nonsmooth sensitivity system: $\dot{X}(t) = (1-t)\mathrm{fsign}(x(t,p_0), X(t))X(t) = (1-t)\,|X(t)|$
    $X(0) = m$

  whose unique solution $X(t) = [x(t,\cdot)]'(0;m)$ satisfies $X(2) = X(0) = m = \dfrac{\partial x}{\partial p}(2,0)m$ ✓





Nonsmooth DAEs Dynamic LD-Derivatives

- Nonsmooth DAEs:
$$\dot{\mathbf{x}}(t,\mathbf{p}) = \mathbf{f}(t,\mathbf{p},\mathbf{x}(t,\mathbf{p}),\mathbf{y}(t,\mathbf{p}))$$
$$\mathbf{0} = \mathbf{g}(\mathbf{p},\mathbf{x}(t,\mathbf{p}),\mathbf{y}(t,\mathbf{p}))$$
$$\mathbf{x}(t_0,\mathbf{p}) = \mathbf{f}_0(\mathbf{p})$$
  - If **f** and **g** and $\mathbf{f}_0$ are L-smooth functions, then $\mathbf{x}(t,\mathbf{p})$ and $\mathbf{y}(t,\mathbf{p})$ are L-smooth w.r.t. **p**
- Nonsmooth DAE sensitivity system:

$$\boxed{\begin{aligned}&\dot{\mathbf{X}}(t) = [\mathbf{f}_t]'(\mathbf{p}_0,\mathbf{x}(t,\mathbf{p}_0),\mathbf{y}(t,\mathbf{p}_0);(\mathbf{M},\mathbf{X}(t),\mathbf{Y}(t)))\\ &\mathbf{0} = [\mathbf{g}_t]'(\mathbf{p}_0,\mathbf{x}(t,\mathbf{p}_0),\mathbf{y}(t,\mathbf{p}_0);(\mathbf{M},\mathbf{X}(t),\mathbf{Y}(t)))\\ &\mathbf{X}(0) = \mathbf{f}_0{}'(\mathbf{p}_0;\mathbf{M})\end{aligned}}$$

  - LD-derivative mappings $t \mapsto [\mathbf{x}(t,\cdot)]'(\mathbf{p}_0;\mathbf{M})$ and $t \mapsto [\mathbf{y}(t,\cdot)]'(\mathbf{p}_0;\mathbf{M})$ uniquely solve the nonsmooth sensitivity system
  - If **M** is nonsingular, then L-derivatives can be computed for any t
  - If **f**, **g** and $\mathbf{f}_0$ are C^1 and **M=I** then the classical sensitivity DAE system is recovered

Stechlinski and Barton, *Journal Opt. Theory. Appl.* (2016)

Dynamic Optimization of Nonsmooth DAEs

- Sequential approach in nonsmooth setting:

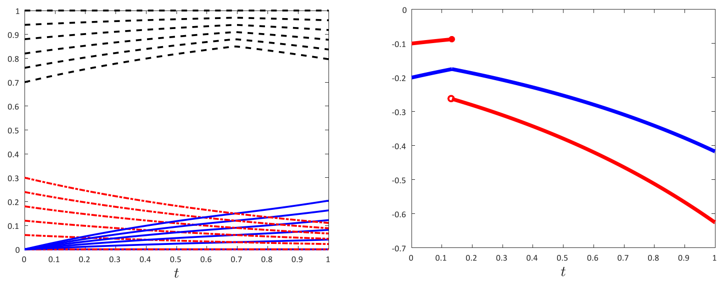
Semi-explicit index-1 DAEs

$$\dot{\mathbf{x}}(t,\mathbf{p}) = \mathbf{f}(\mathbf{p},\mathbf{x}(t,\mathbf{p}),\mathbf{y}(t,\mathbf{p}))$$
$$\mathbf{0} = \mathbf{g}(\mathbf{p},\mathbf{x}(t,\mathbf{p}),\mathbf{y}(t,\mathbf{p}))$$

LD-Derivative Sensitivity DAEs

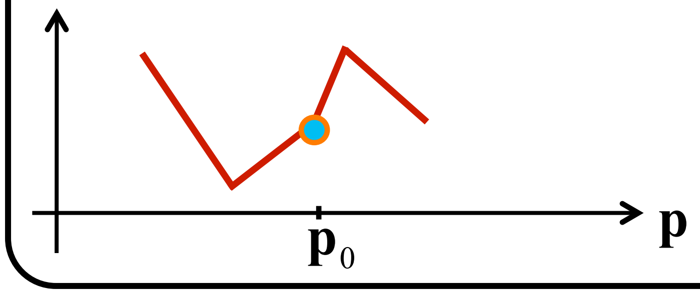
$$\dot{\mathbf{X}} = [\mathbf{f}]'(\mathbf{p}_0,\mathbf{x},\mathbf{y};(\mathbf{M},\mathbf{X},\mathbf{Y}))$$
$$\mathbf{0} = [\mathbf{g}]'(\mathbf{p}_0,\mathbf{x},\mathbf{y};(\mathbf{M},\mathbf{X},\mathbf{Y}))$$

$\mathbf{x}(t,\mathbf{p}_0),\mathbf{y}(t,\mathbf{p}_0)$ "$\mathbf{s}_x$" $= \mathbf{X}$, "$\mathbf{s}_y$" $= \mathbf{Y}$



Update **p** via optimization

$\phi(\mathbf{x}(t_f,\mathbf{p}),\mathbf{y}(t_f,\mathbf{p}))$



DAE Sensitivities: Nonsmooth vs. Smooth

- Smooth vs. nonsmooth cases:

  - Nonsmooth DAE sensitivities:

$$\boxed{\begin{aligned}\dot{\mathbf{X}} &= [\mathbf{f}]'(\mathbf{p}_0, \mathbf{x}, \mathbf{y}; (\mathbf{M}, \mathbf{X}, \mathbf{Y}))\\ \mathbf{0} &= [\mathbf{g}]'(\mathbf{p}_0, \mathbf{x}, \mathbf{y}; (\mathbf{M}, \mathbf{X}, \mathbf{Y}))\\ \mathbf{X}(t_0) &= [\mathbf{f}_0]'(\mathbf{p}_0; \mathbf{M})\end{aligned}}$$

  - Nonsmooth (and nonlinear) DAE system
  - Unique solution and unique initialization
  - $\mathbf{X}$ continuous, $\mathbf{Y}$ discontinuous

  - Smooth DAE sensitivities:

$$\boxed{\begin{aligned}\dot{\mathbf{s}}_x &= \frac{\partial \mathbf{f}}{\partial \mathbf{p}} + \frac{\partial \mathbf{f}}{\partial \mathbf{x}}\mathbf{s}_x + \frac{\partial \mathbf{f}}{\partial \mathbf{y}}\mathbf{s}_y\\ \mathbf{0} &= \frac{\partial \mathbf{g}}{\partial \mathbf{p}} + \frac{\partial \mathbf{g}}{\partial \mathbf{x}}\mathbf{s}_x + \frac{\partial \mathbf{g}}{\partial \mathbf{y}}\mathbf{s}_y\\ \mathbf{s}_x(t_0) &= \mathbf{J}\mathbf{f}_0(\mathbf{p}_0)\end{aligned}}$$

  - Linear DAE system
  - Unique solution and unique initialization
  - $\mathbf{s}_x$, $\mathbf{s}_y$ continuous

Stechlinski and Barton, *Journal Opt. Theory. Appl.* (2016)

Simple Flash Process: Well-Posedness

- Nonsmooth DAE model of simple const. *P* flash:

$$\dot{H}(t) = U\left(T_{out} - T(t)\right)$$

$$M = M_L(t) + M_V(t)$$

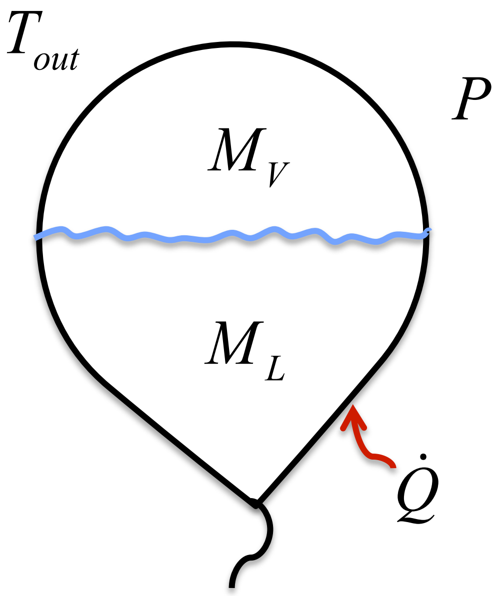
$$H(t) = Mh_V(t) - M_L(t)\Delta h_{vap}\left(T(t)\right)$$

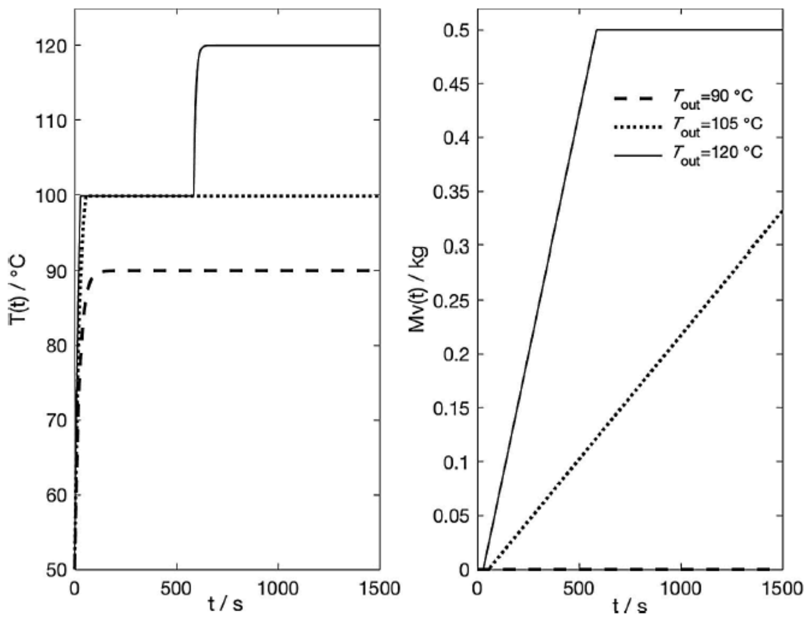
$$h_V(t) = Cp\left(T(t) - T_0\right)$$

$$\log\left(P^{sat}(t)\right) = A - B/\left(T(t) + C\right)$$

$$\boxed{0 = \operatorname{mid}\left(M_V(t), P - P^{sat}(T(t)), -M_L(t)\right)}$$

$$\vdots$$





- Does there exist a (regular) solution?
  - Yes, under appropriate initial conditions and some simplifying assumptions $\pi_y \partial \mathbf{g}(H, T, M_L)$ contains no singular matrices. This implies existence and uniqueness of a regular solution (since right-hand side functions are PC[1])

Stechlinski, Patrascu and Barton, *Comp. And Chem. Eng.* (In Press)

Simple Flash Process: Sensitivities

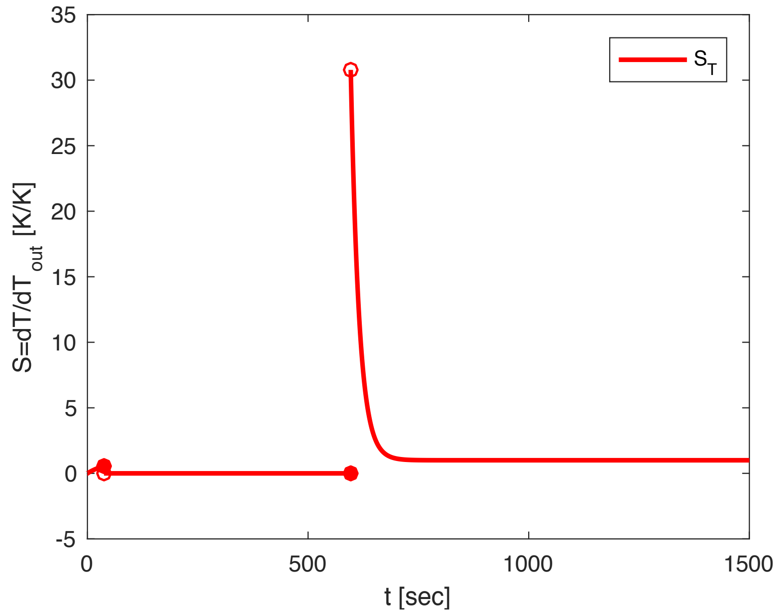
- Nonsmooth sensitivities of simple const. *P* flash:

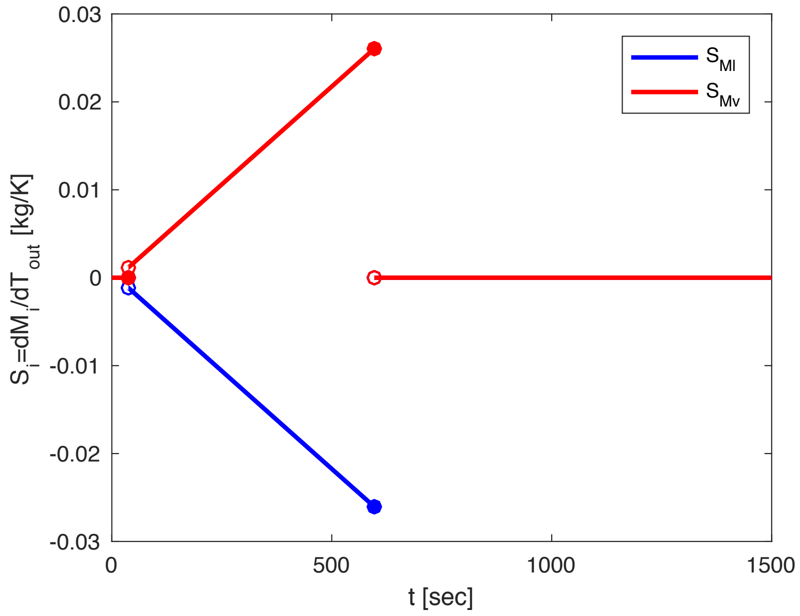
$$\dot{S}_H(t) = U\left(1 - S_T(t)\right)$$

$$S_H(t) = MCpS_T(t) - \Delta h_{vap}{}'\left(T(t)\right)S_T(t)$$

$$0 = \text{mid}'\left(M_V(t), P - P_{sat}\left(T(t)\right), -M_L(t); (S_V(t), -P_{\text{sat}}{}'(T(t))S_T(t), -S_L(t))\right)$$

$$S_V(t) = -S_L(t)$$





- No notion of mode sequence needed

Nonsmooth Dynamical Systems

- Nonsmooth ODEs/DAEs/hybrid automata

- Open loop optimal control with nonsmooth ODE/DAEs:

$$\begin{aligned}&\inf_{\mathbf{p}} \Phi(\mathbf{p}) \equiv \phi(t_f, \mathbf{p}, \mathbf{u}(t_f, \mathbf{p}), \mathbf{x}(t_f, \mathbf{p}, \mathbf{u}), \mathbf{y}(t_f, \mathbf{p}, \mathbf{u}))\\ &\text{s.t.} \quad (\mathbf{x}, \mathbf{y}) \text{ satisfy nonsmooth DAE system}\end{aligned}$$

- ODEs with LPs embedded:

$$\begin{aligned}\dot{\mathbf{x}}(t,\mathbf{p}) &= \mathbf{f}(\mathbf{p}, \mathbf{x}(t,\mathbf{p}), h(\mathbf{x}(t,\mathbf{p})))\\ h(\mathbf{x}(t,\mathbf{p})) &= \min_{\mathbf{v}} \mathbf{c}^{\mathrm{T}}\mathbf{v}\\ &\quad \text{s.t. } \mathbf{A}\mathbf{v} = \mathbf{b}(\mathbf{x}(t,\mathbf{p}))\\ &\qquad\quad \mathbf{v} \geq \mathbf{0}\end{aligned}$$

- Etc…

Khan and Barton, *Journal Opt. Theory Appl.* 163 (2014); Stechlinski and Barton, 55^{th} *CDC*. (2016);
Höffner, Khan, and Barton. Automatica. 63 (2016); Khan and Barton, 53^{rd} *CDC*. (2014);
Barton, Khan, Stechlinski, Watson, *Opt. Meth. & Soft.* (In Press);

Summary

- Nonsmooth ODEs and DAEs possess a strong mathematical theory (recently for DAEs)
- Easy-to-use and solve models that act as a natural framework in many physical problems
- Open to tractable numerical implementations
- Applicable to a wide range of process operations
- Future work in adjoint sensitivities (?) and discontinuous dynamical systems

SENSITIVITY ANALYSIS OF OPTIMIZATION PROBLEMS

Parametric Nonlinear Programs (NLPs)

- Consider the following parametric NLP:

$$\begin{aligned} \min_{\mathbf{x}} \quad & f(\mathbf{p},\mathbf{x}) \\ \text{s.t.} \quad & \mathbf{g}(\mathbf{p},\mathbf{x}) \le \mathbf{0} \end{aligned}$$

- Goal: given $\mathbf{p}_0$ and corresponding minimizer $\mathbf{x}_0$, calculate $\frac{\partial \mathbf{x}}{\partial \mathbf{p}}(\mathbf{p}_0)$ to characterize $\mathbf{x}(\mathbf{p})$ near $\mathbf{p} = \mathbf{p}_0$

- Note: this is different than calculating a minimizer, for which there are established methods

KKT Equation System

- Consider the following parametric NLP:

$$\min_{\mathbf{x}} \quad f(\mathbf{p},\mathbf{x})$$
$$\text{s.t.} \quad \mathbf{g}(\mathbf{p},\mathbf{x}) \leq \mathbf{0}$$

- A point $(\mathbf{p}_0,\mathbf{x}_0,\boldsymbol{\mu}_0)$ is called a Karush-Kuhn-Tucker (KKT) point if it satisfies the following equations:

$$\nabla_{\mathbf{x}} f(\mathbf{p}_0,\mathbf{x}_0)+\sum_{i=1}^{m}\mu_i \nabla_{\mathbf{x}}\mathbf{g}(\mathbf{p}_0,\mathbf{x}_0)=\mathbf{0}, \longleftarrow \text{stationarity}$$

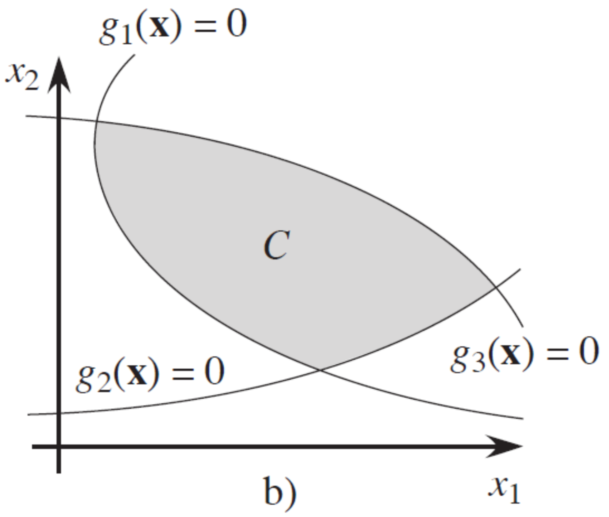
$$\mathbf{g}(\mathbf{p},\mathbf{x}) \leq \mathbf{0}, \longleftarrow \text{primal feasibility}$$

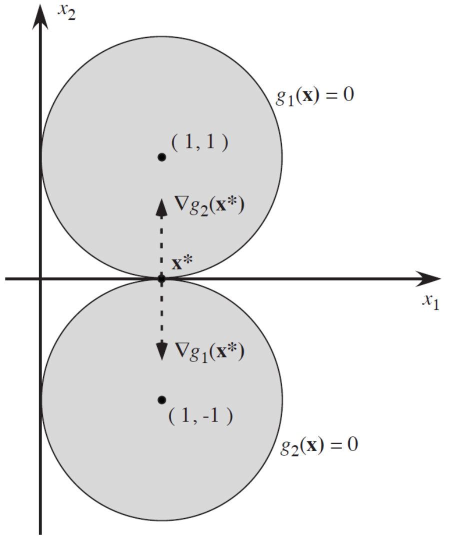
$$\boldsymbol{\mu} \geq \mathbf{0}, \longleftarrow \text{dual feasibility}$$

$$\mu_i g_i(\mathbf{p}_0,\mathbf{x}_0)=0, \quad i=1,\ldots,m \longleftarrow \text{complementary slackness}$$

Regularity of NLP KKT Points

- Linear independence constraint qualification (LICQ) holds at $(\mathbf{p}_0,\mathbf{x}_0)$: the set of vectors $\{\nabla_{\mathbf{x}} g_i(\mathbf{p}_0,\mathbf{x}_0) : i \in A(\mathbf{p}_0,\mathbf{x}_0)\}$ are linearly independent, where $A(\mathbf{p}_0,\mathbf{x}_0) = \{i : g_i(\mathbf{p}_0,\mathbf{x}_0) = 0\}$ is the set of active constraints





- Strong second-order sufficient condition (SSOSC) holds at $(\mathbf{p}_0,\mathbf{x}_0,\boldsymbol{\mu}_0)$:
$\mathbf{d}^{\mathrm{T}}\nabla^2_{\mathbf{xx}} L(\mathbf{p}_0,\mathbf{x}_0,\boldsymbol{\mu}_0)\mathbf{d} > 0$, for all $\mathbf{d} \neq \mathbf{0}$ s.t. $(\nabla_{\mathbf{x}} g_i(\mathbf{p}_0,\mathbf{x}_0))^{\mathrm{T}}\mathbf{d} = 0$, $i \in A^+(\mathbf{p}_0,\mathbf{x}_0,\boldsymbol{\mu}_0)$

$$A^+(\mathbf{p}_0,\mathbf{x}_0,\boldsymbol{\mu}_0) = \{i : g_i(\mathbf{p}_0,\mathbf{x}_0) = 0 < \mu_{0,i}\} \text{ is the strongly active set}$$

Classical Sensitivity System

- Assumptions: KKT point $(\mathbf{p}_0, \mathbf{x}_0, \boldsymbol{\mu}_0)$ satisfies LICQ and SSOSC and strict complementarity (i.e. $g_i(\mathbf{p}_0, \mathbf{x}_0) < \mu_{0,i}$ for all $i = 1, \ldots, m$)

- Then $\mathbf{x}(\mathbf{p})$, $\boldsymbol{\mu}(\mathbf{p})$ are smooth near $\mathbf{p} = \mathbf{p}_0$ and sensitivities satisfy linear equation system:

$$\begin{bmatrix} \nabla^2_{\mathbf{xx}} L & \nabla_{\mathbf{x}} \mathbf{g}_{A^+} \\ -(\nabla_{\mathbf{x}} \mathbf{g}_{A^+})^{\mathrm{T}} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \dfrac{\partial \mathbf{x}}{\partial \mathbf{p}} \\ \dfrac{\partial \boldsymbol{\mu}_{A^+}}{\partial \mathbf{p}} \end{bmatrix} = \begin{bmatrix} -\nabla^2_{\mathbf{xp}} L \\ (\nabla_{\mathbf{p}} \mathbf{g}_{A^+})^{\mathrm{T}} \end{bmatrix}$$

$$\frac{\partial \boldsymbol{\mu}_{A^-}}{\partial \mathbf{p}} = \mathbf{0}$$

Fiacco and McCormick, 1968. Wiley

Nonsmooth KKT Equation System

- Let $(\mathbf{p}_0, \mathbf{x}_0, \boldsymbol{\mu}_0)$ be an NLP KKT point; i.e.

$$\nabla_{\mathbf{x}} L(\mathbf{p}_0, \mathbf{x}_0, \boldsymbol{\mu}_0) = \mathbf{0}$$
$$\mathbf{0} \le -\mathbf{g}(\mathbf{p}_0, \mathbf{x}_0) \perp \boldsymbol{\mu}_0 \ge \mathbf{0}$$

notation: $0 \le a \perp b \ge 0 \Leftrightarrow a \ge 0, b \ge 0, ab = 0$

- Observe that $a \ge 0, b \ge 0, ab = 0$ is equivalent to $\min(a,b) = 0$, so that

$$\boldsymbol{\Phi}(\mathbf{p}, \mathbf{x}, \boldsymbol{\mu}) = \begin{bmatrix} \nabla_{\mathbf{x}} L(\mathbf{p}, \mathbf{x}, \boldsymbol{\mu}) \\ \mathbf{min}(\mathbf{g}(\mathbf{p}, \mathbf{x}), \boldsymbol{\mu}) \end{bmatrix} = \mathbf{0}$$

- Idea**:** regularity conditions allow for application of the nonsmooth implicit function theorem to the nonsmooth KKT equation system
- Nonsmooth sensitivity system: which simplifies to…

Nonsmooth Sensitivity System

- Assumptions: KKT point $(\mathbf{p}_0, \mathbf{x}_0, \boldsymbol{\mu}_0)$ satisfies LICQ and SSOSC

- Then $\mathbf{x}(\mathbf{p})$, $\boldsymbol{\mu}(\mathbf{p})$ are PC[1] near $\mathbf{p} = \mathbf{p}_0$ and sensitivities $\mathbf{X} = \mathbf{x}'(\mathbf{p}_0;\mathbf{P})$ $\mathbf{U} = \boldsymbol{\mu}'(\mathbf{p}_0;\mathbf{P})$ (uniquely) satisfy nonsmooth equation system:

$$\begin{bmatrix} \nabla^2_{\mathbf{xx}} L & \nabla_{\mathbf{x}} \mathbf{g}_{A^+ \cup A^0} \\ -(\nabla_{\mathbf{x}} \mathbf{g}_{A^+ \cup A^0})^{\mathrm{T}} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{X} \\ \mathbf{U}_{A^+ \cup A^0} \end{bmatrix} = \begin{bmatrix} -\nabla^2_{\mathbf{xp}} L \\ (\nabla_{\mathbf{p}} \mathbf{g}_{A^+})^{\mathrm{T}} \end{bmatrix} \mathbf{P}$$

$$\mathbf{LMmin}(-(\nabla_{\mathbf{p}} \mathbf{g}_{A^0})^{\mathrm{T}} \mathbf{P} - (\nabla_{\mathbf{x}} \mathbf{g}_{A^0})^{\mathrm{T}} \mathbf{X}, \mathbf{U}_{A^0}) = \mathbf{0}$$

Stechlinski, Khan and Barton, SIAM Journal on Optimization. (2018)

Sensitivity Systems: Smooth vs. Nonsmooth

- Assumptions: KKT point $(\mathbf{p}_0, \mathbf{x}_0, \boldsymbol{\mu}_0)$ satisfies LICQ and SSOSC and strict complementarity

- Then $\mathbf{x}(\mathbf{p}),\ \boldsymbol{\mu}(\mathbf{p})$ are smooth near $\mathbf{p} = \mathbf{p}_0$ and sensitivities (uniquely) satisfy linear equation system:

$$\begin{bmatrix} \nabla^2_{\mathbf{xx}} L & \nabla_{\mathbf{x}} \mathbf{g}_{A^+ \cancel{\cup A^0}} \\ -(\nabla_{\mathbf{x}} \mathbf{g}_{A^+ \cup A^0})^{\mathrm{T}} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{X} \\ \mathbf{U}_{A^+ \cancel{\cup A^0}} \end{bmatrix} = \begin{bmatrix} -\nabla^2_{\mathbf{xp}} L \\ (\nabla_{\mathbf{p}} \mathbf{g}_{A^+})^{\mathrm{T}} \end{bmatrix} \cancel{\mathbf{P}}$$

$$\cancel{\mathbf{LMmin}(-(\nabla_{\mathbf{p}} \mathbf{g}_{A^0})^{\mathrm{T}} \mathbf{P} - (\nabla_{\mathbf{x}} \mathbf{g}_{A^0})^{\mathrm{T}} \mathbf{X}, \mathbf{U}_{A^0}) = \mathbf{0}}$$

Summary

- Nonsmooth NLP sensitivity system admits a unique solution, which is computationally relevant generalized derivative information
- Recovers classical theory of Fiacco and McCormick in absence of weakly active sets
- Numerical solution is based on same ideas as calculating LD-derivative of nonsmooth implicit function. There are three approaches:
  - Cycle through selection functions (i.e. solve a number of classical sensitivity systems)
  - Directly solve nonsmooth sensitivity systems (e.g. via nonsmooth Newton methods), which can be improved by fathoming weakly active constraints along the way in the spirit of branch-locking techniques
  - Solve sequence of (convex?) QPs
- Extension to nonunique multipliers is underway, where current results only yield directional derivative information (Ralph & Dempe), VIs via natural or normal maps

Acknowledgments

- Kamil Khan, Peter Stechlinski
- Novartis
- Statoil