Nonsmooth Optimization: Stable Descent and Sparsity Preservation

Ying Cui
Department of Industrial Engineering and Operations Research
University of California, Berkeley

Joint work with Hanyang Li (UC Berkeley) and Jake Roth (University of Minnesota)

Nonsmooth functions

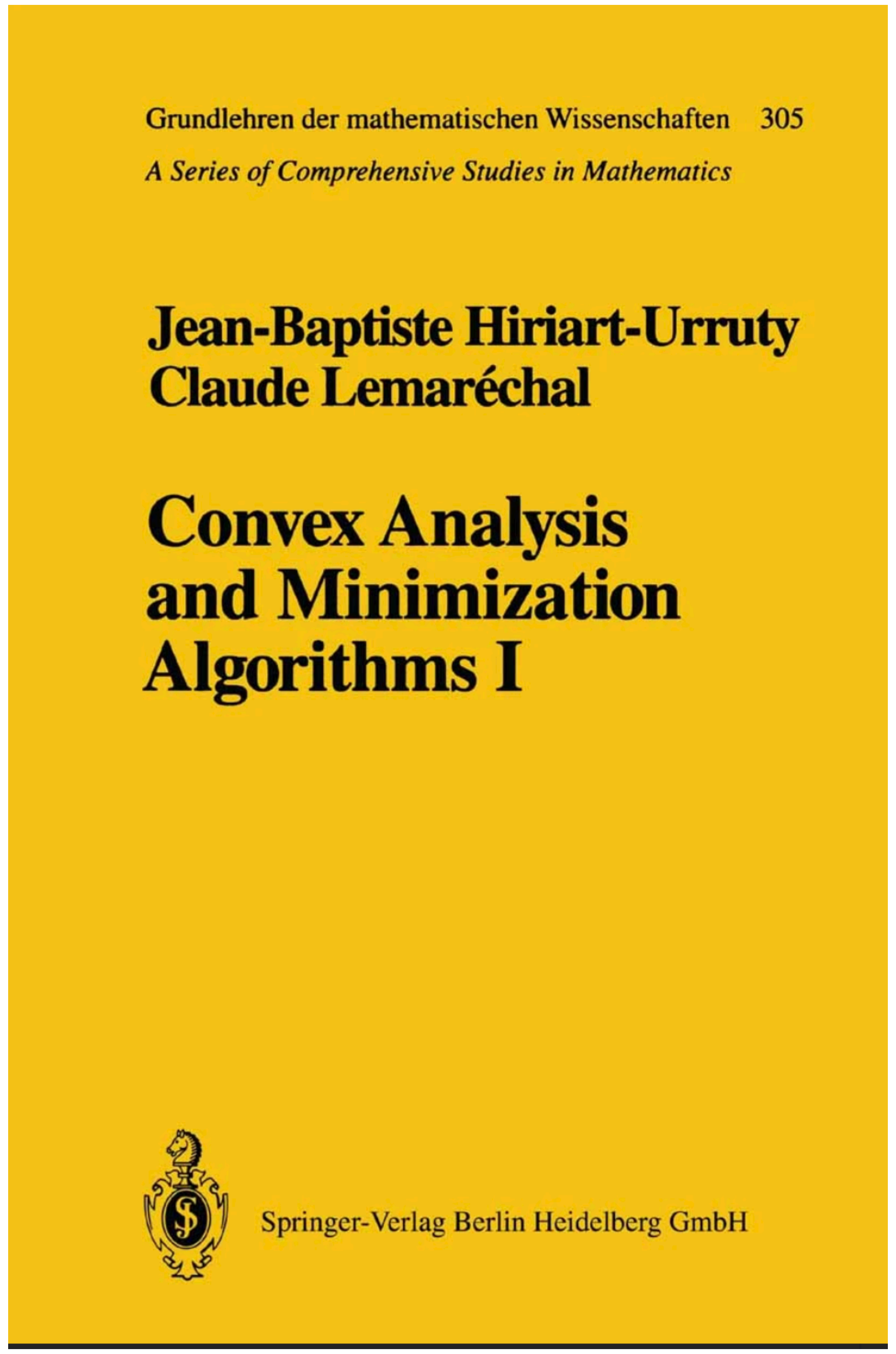
- A locally Lipschitz continuous function is differentiable almost everywhere
- Nonsmooth functions:
  - gradients not continuously vary (at the kinks)
  - second derivatives grow unboundedly
- ``Smoothing functions” may still suffer from unstable gradients when the smoothing parameter is very small

Nonsmooth optimization

``…Unfortunately, there is ***no clear-cut*** *between functions that are* ***smooth*** *(whence the field application such algorithms) and functions that are* ***not*** *(whence requiring methods from nonsmooth optimization)…*

…A sound algorithm for convex minimization should therefore not ignore its parents…”

from the monograph Convex Analysis and Minimization Algorithms



Nonsmooth optimization

Part 1: Stable Descent Directions

when the function is also nonconvex

Part 2: Benefit from Nonsmoothness

if the (sparsity) structure is properly preserved

Stable descent directions

If f is convex, **steepest descent direction** at x

$$g_x := \operatorname*{argmin}_{\|d\|_2=1} f'(x; d) = \left\{ -\frac{v}{\|v\|_2} : v = \operatorname*{argmin}_{v \in \partial f(x)} \|v\| \right\}$$

- When f is smooth, $g_x = -\nabla f(x)/\|\nabla f(x)\|_2$

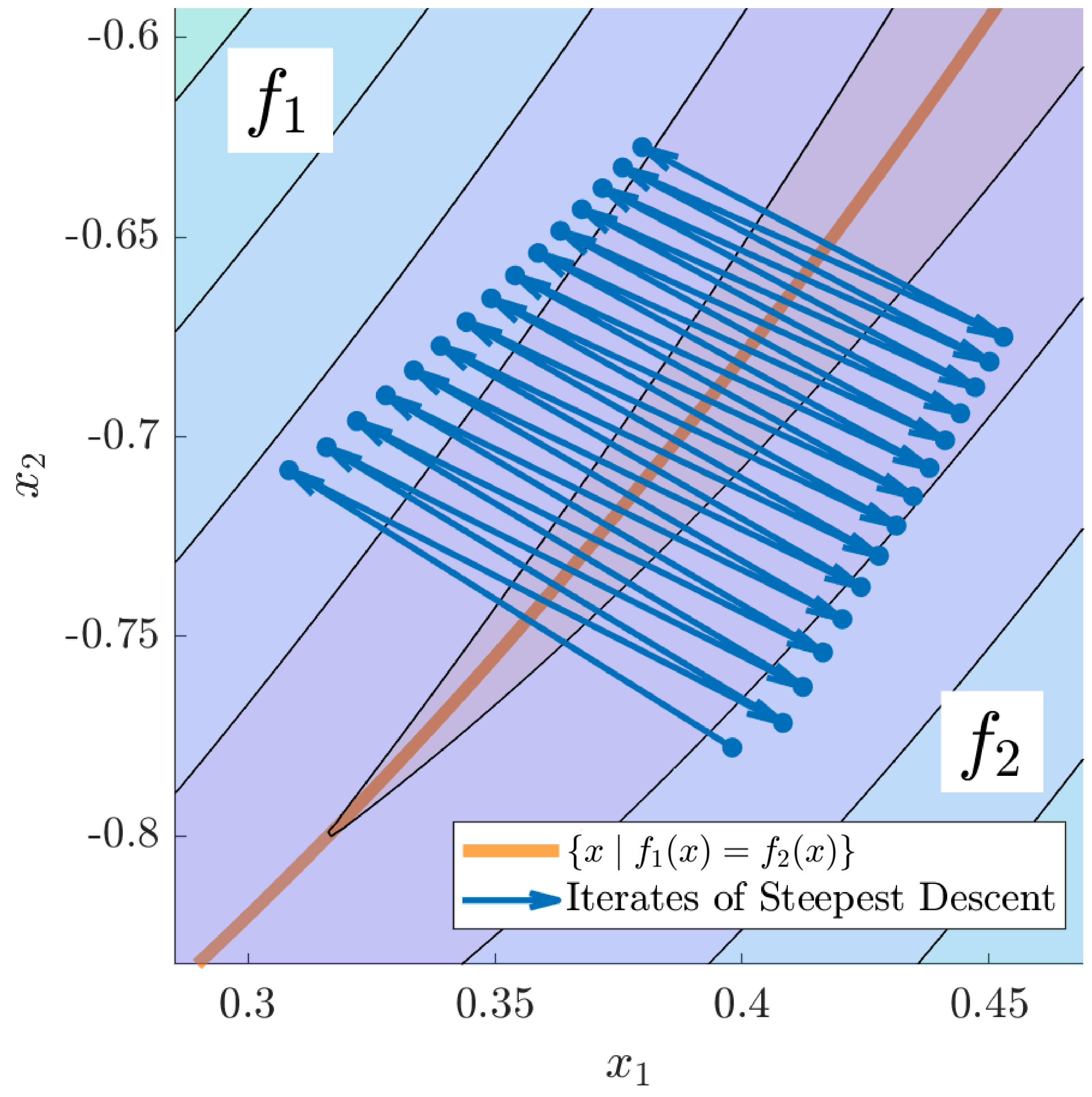
Stable descent directions

If f is convex, **steepest descent direction** at x:

$$g_x := \operatorname*{argmin}_{\|d\|_2=1} f'(x; d) = \left\{ -\frac{v}{\|v\|_2} : v = \operatorname*{argmin}_{v \in \partial f(x)} \|v\| \right\}$$

- When f is smooth, $g_x = -\nabla f(x)/\|\nabla f(x)\|_2$
- When f is nonsmooth at x, g_x is discontinuous

Think about $f(x) = \max\{f_1(x), f_2(x)\}$ near $f_1(x) = f_2(x)$



Stable descent directions

If f is convex, **steepest descent direction** at x:

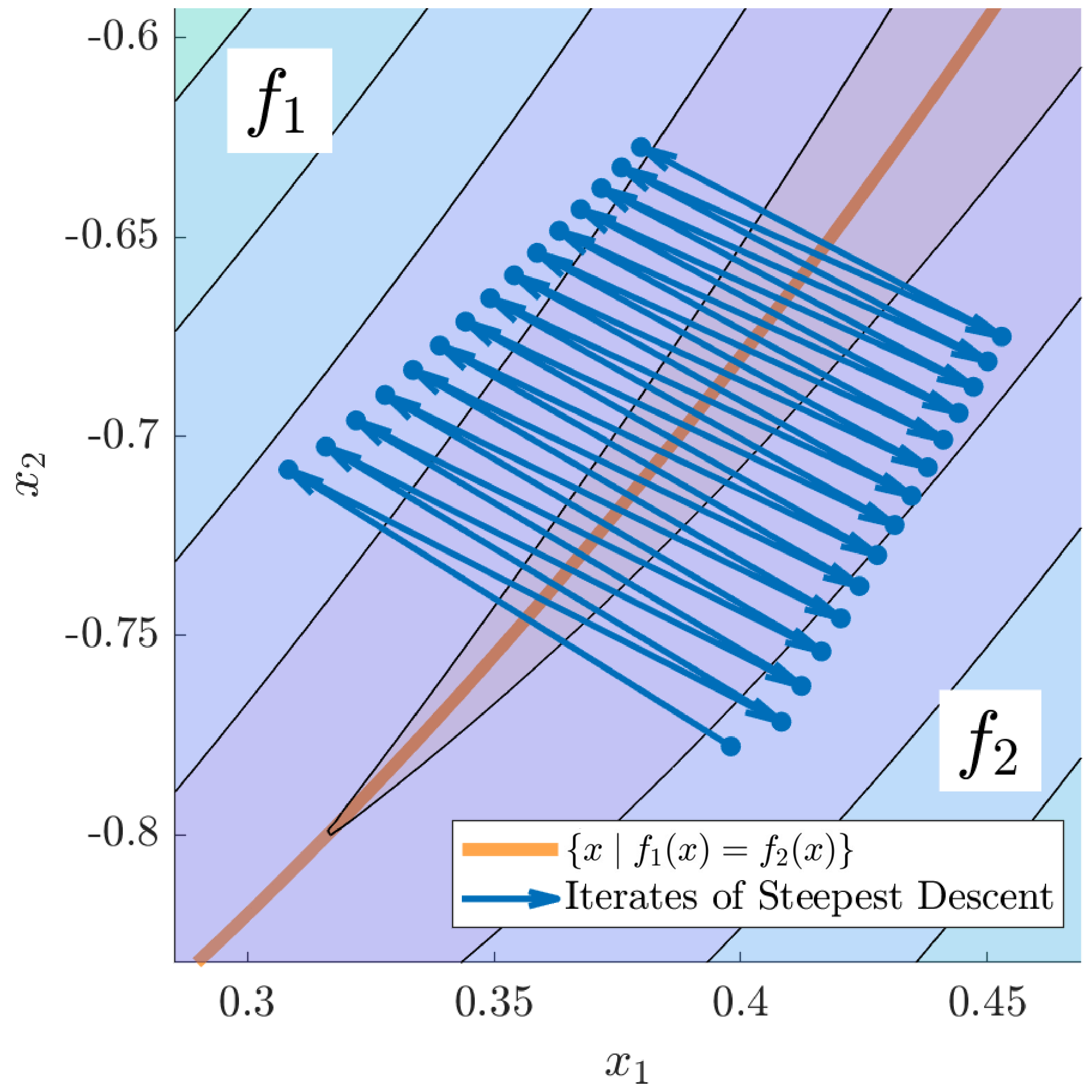
$$g_x := \underset{\|d\|_2=1}{\operatorname{argmin}} f'(x; d) = \left\{ -\frac{v}{\|v\|_2} : v = \underset{v \in \partial f(x)}{\operatorname{argmin}} \|v\| \right\}$$

- When f is smooth, $g_x = -\nabla f(x)/\|\nabla f(x)\|_2$
- When f is nonsmooth at x, g_x is discontinuous

Think about $f(x) = \max\{f_1(x), f_2(x)\}$ near $f_1(x) = f_2(x)$

— unstable (zigzag phenomenon)

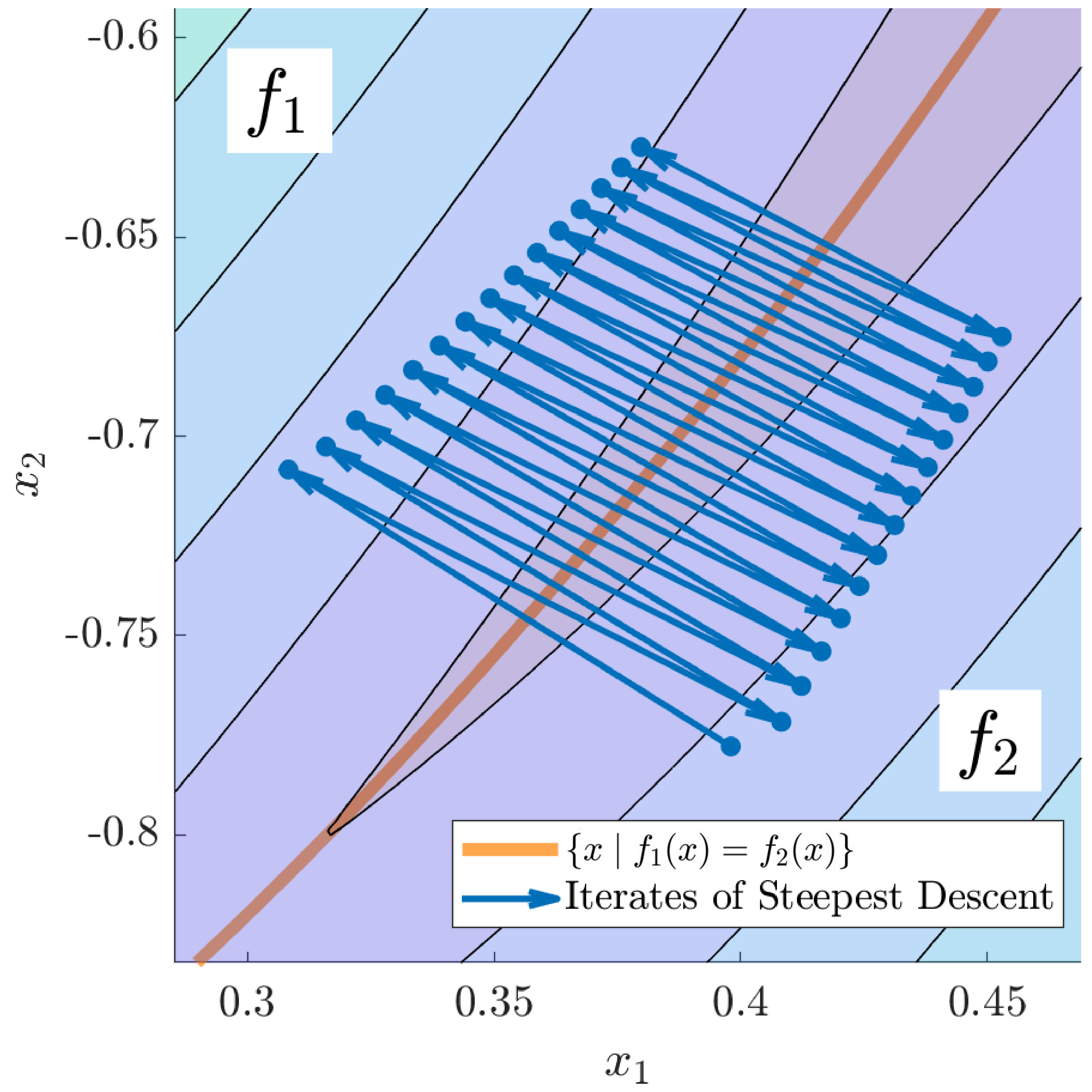
— may converge to non-stationary points (even with exact line search)



Stable descent directions

If f is convex, **steepest descent direction** at x:

$$g_x := \operatorname*{argmin}_{\|d\|_2=1} f'(x; d) = \left\{ -\frac{v}{\|v\|_2} : v = \operatorname*{argmin}_{v \in \partial f(x)} \|v\| \right\}$$



- **Improvement:** $g_x \xrightarrow{\text{stablize in } x} ??$

Stable descent directions

1. Goldstein-type methods

Idea: ϵ-neighborhood of x^k stabilizes the direction

Goldstein ϵ-subdifferential $\partial_\epsilon^G f(x) := \operatorname{conv}\left\{ \bigcup_{\|z-x\|\le\epsilon} \partial f(z) \right\}$

Stable descent directions

1. Goldstein-type methods

Idea: ϵ-neighborhood of x^k stabilizes the direction

Goldstein ϵ-subdifferential $\partial_\epsilon^G f(x) := \operatorname{conv}\left\{ \bigcup_{\|z-x\|\leq\epsilon} \partial f(z) \right\}$

$$x^{k+1} = x^k - \epsilon \frac{g_k}{\|g_k\|} \quad \text{with} \quad g_k := \operatorname*{argmin}_{v \in \partial_\epsilon^G f(x^k)} \|v\|$$

Stable descent directions

1. Goldstein-type methods

Idea: ϵ-neighborhood of x^k stabilizes the direction

Goldstein ϵ-subdifferential $\partial^G_\epsilon f(x) := \mathrm{conv}\left\{ \bigcup_{\|z-x\|\le\epsilon} \partial f(z) \right\}$

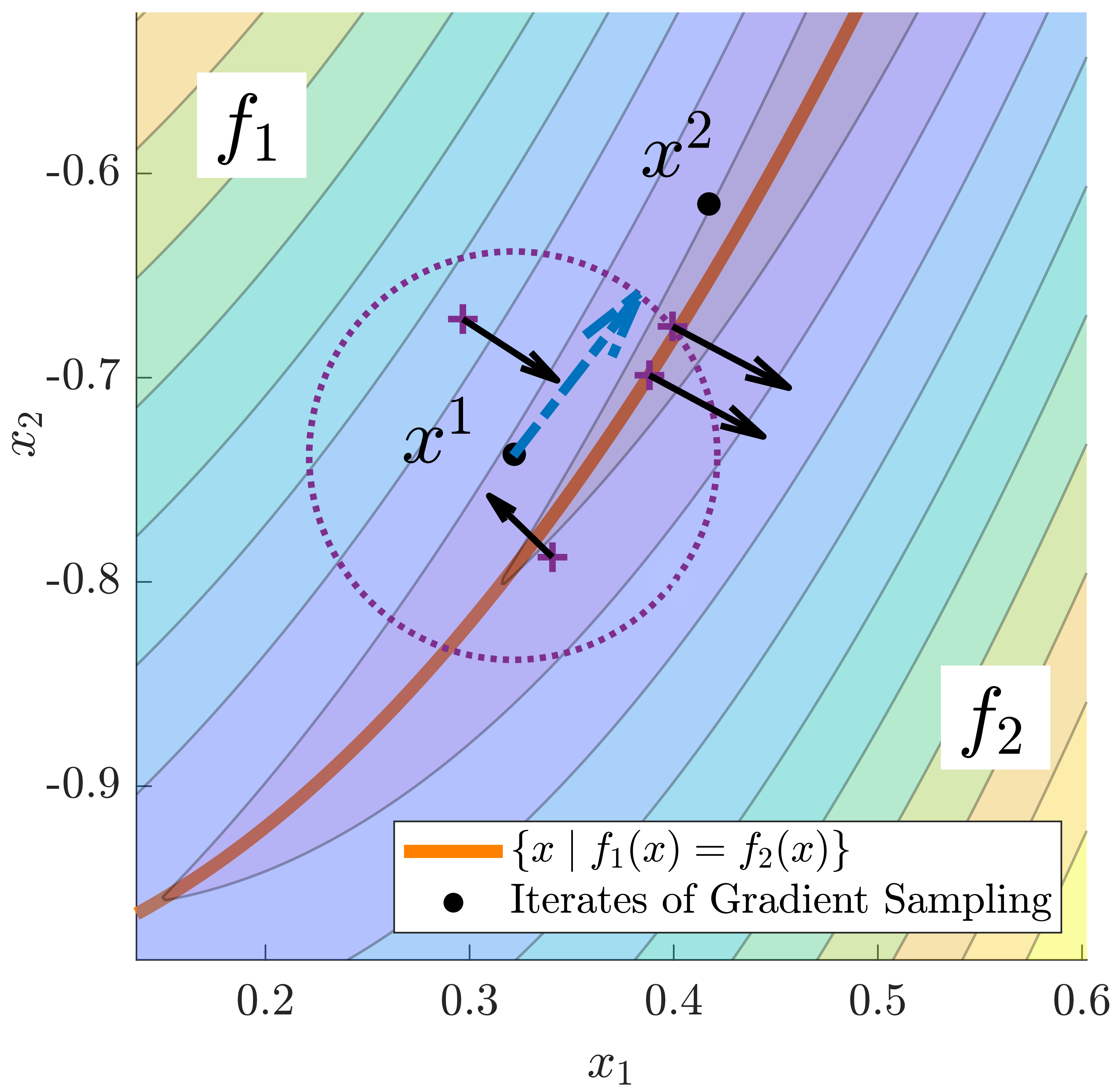
$$x^{k+1} = x^k - \epsilon \frac{g_k}{\|g_k\|} \quad \text{with} \quad g_k := \operatorname*{argmin}_{v \in \partial^G_\epsilon f(x^k)} \|v\|$$

Practical issue: computation of g_k

$\xrightarrow{\text{approx.}}$ *Gradient Sampling [Burke, Lewis, Overton '02], [Burke, Lewis, Overton '05], [Kiwiel '07], [Curtis and Que '13], [Burke et.al.2020]*

INGD [Zhang, Lin, Jegelka, Sra, Jadbabaie '20],

NTD [Davis, Jiang '23], …



Stable descent directions

1. Goldstein-type methods

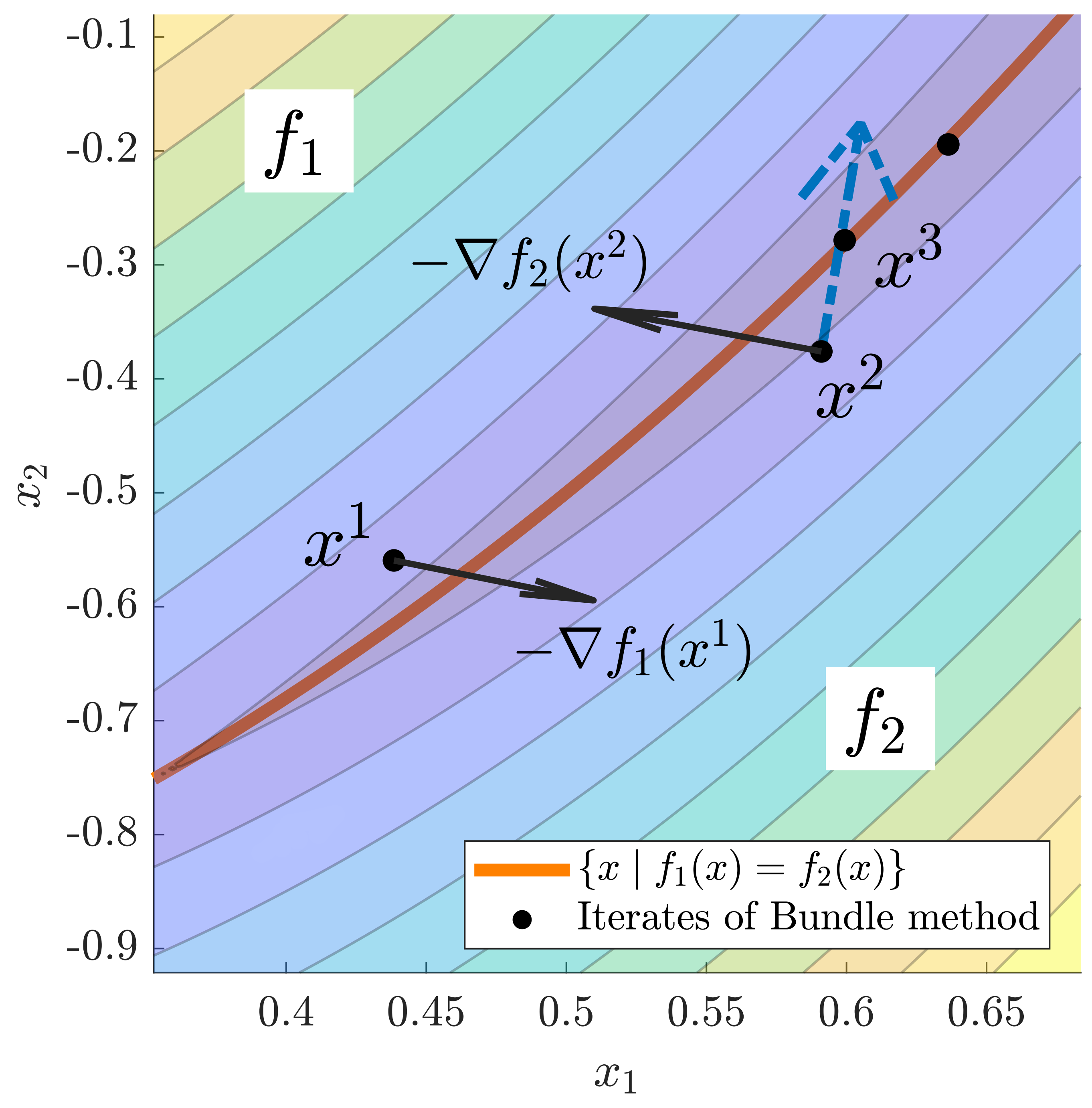
Idea: ϵ-neighborhood of x^k stabilizes the direction

2. Bundle-type methods

Idea: ϵ-neighborhood of $f(x^k)$ stabilizes the direction

"Bundle": subgradients & function values over past iterations

$$\begin{matrix} \{v_1 \in \partial f(x^1), & v_2 \in \partial f(x^2), & \cdots, & v_k \in \partial f(x^k)\} \\ \{\quad f(x^1), & f(x^2), & \cdots, & f(x^k) \quad\} \end{matrix}$$



A unified interpretation

$$x^{k+1} = x^k - \alpha_k \cdot g_k \quad \text{with} \quad g_k := \operatorname*{argmin}_{v \in S_k} \|v\|$$

✗ Steepest descent method: $S_k = \partial f(x^k)$

1. Goldstein-type methods $S_k = \partial^G_{\epsilon_k} f(x^k) = \operatorname{conv}\left\{ \bigcup_{\|z - x^k\| \le \epsilon_k} \partial f(z) \right\}$

Idea: ϵ-neighborhood of x^k stabilizes the direction

2. Bundle-type methods $S_k = \partial_{\epsilon_k} f(x^k) = \left\{ v \mid f(z) \ge f(x^k) + v^\top (z - x^k) - \epsilon_k, \forall z \right\}$ if f is convex

Idea: ϵ-neighborhood of $f(x^k)$ stabilizes the direction

A unified interpretation

$$x^{k+1} = x^k - \alpha_k \cdot g_k \quad \text{with} \quad g_k := \underset{v \in S_k}{\operatorname{argmin}} \|v\|$$

$$S_k = \partial^G_{\epsilon_k} f(x^k) = \operatorname{conv}\left\{ \bigcup_{\|z - x^k\| \le \epsilon_k} \partial f(z) \right\}$$

$$S_k = \partial_{\epsilon_k} f(x^k) = \left\{ v \mid f(z) \ge f(x^k) + v^\top (z - x^k) - \epsilon_k, \forall z \right\}$$

Stable descent directions $\xrightarrow{\text{need}}$ **combine (sub)gradients in some "neighborhood"**

A similar story for variance reduction / momentum in stochastic optimization

Stable descent directions

$$x^{k+1} = x^k - \alpha_k \cdot g_k \quad \text{with} \quad g_k := \underset{v \in S_k}{\operatorname{argmin}} \|v\|$$

1. Goldstein-type methods

$$S_k = \partial^G_{\epsilon_k} f(x^k) = \operatorname{conv}\left\{ \bigcup_{\|z - x^k\| \le \epsilon_k} \partial f(z) \right\}$$

Almost impossible to compute (deterministically)

$$S_k = \partial_{\epsilon_k} f(x^k) = \left\{ v \mid f(z) \ge f(x^k) + v^\top (z - x^k) - \epsilon_k, \forall z \right\}$$

Stable descent directions

$$x^{k+1} = x^k - \alpha_k \cdot g_k \quad \text{with} \quad g_k := \underset{v \in S_k}{\operatorname{argmin}} \|v\|$$

$$S_k = \partial^G_{\epsilon_k} f(x^k) = \operatorname{conv}\left\{ \bigcup_{\|z - x^k\| \le \epsilon_k} \partial f(z) \right\}$$

2. Bundle-type methods $S_k = \partial_{\epsilon_k} f(x^k) = \left\{ v \mid f(z) \ge f(x^k) + v^\top (z - x^k) - \epsilon_k, \forall z \right\}$ if f is convex

Only well understood (complexity & convergence rate) for convex optimization

Stable descent directions

$$x^{k+1} = x^k - \alpha_k \cdot g_k \quad \text{with} \quad g_k := \operatorname*{argmin}_{v \in S_k} \|v\|$$

$$S_k = \partial^G_{\epsilon_k} f(x^k) = \operatorname{conv}\left\{ \bigcup_{\|z - x^k\| \le \epsilon_k} \partial f(z) \right\}$$

Idea: ϵ-neighborhood of x^k stabilizes the direction

Not imply each other

$$S_k = \partial_{\epsilon_k} f(x^k) = \left\{ v \mid f(z) \ge f(x^k) + v^\top (z - x^k) - \epsilon_k, \forall z \right\}$$

Idea: ϵ-neighborhood of $f(x^k)$ stabilizes the direction

Abstract theory & new construction for stable descent methods of nonsmooth optimization?

$$S_k = \partial^G_{\epsilon_k} f(x^k) = \mathrm{conv}\left\{ \bigcup_{\|z-x^k\|\le\epsilon_k} \partial f(z) \right\}$$

Almost impossible to compute (deterministically)

Not imply each other

$$S_k = \partial_{\epsilon_k} f(x^k) = \left\{ v \mid f(z) \ge f(x^k) + v^\top (z - x^k) - \epsilon_k, \forall z \right\}$$

Only well understood (complexity & convergence rate) for weakly convex optimization

Abstract theory of stable descent

A map $G : \mathbb{R}^n \times (0,\infty) \rightrightarrows \mathbb{R}^m$ is a descent-oriented ϵ-subdifferential for f if

(G1) Outer jointly limit in (x, ϵ) stays in the Clarke subdifferential:

$$\limsup_{\epsilon \downarrow 0,\, x \to \bar{x}} G(x, \epsilon) \subset \partial f(\bar{x})$$

``Gradient consistency''

(G2) Separate limits yield the minimal norm subgradient:

$$\lim_{\epsilon \downarrow 0} \left(\limsup_{x \to \bar{x}} G(x, \epsilon) \right) = \operatorname{argmin}\{\|v\| \mid v \in \partial f(\bar{x})\}$$

Abstract theory of stable descent

A map $G : \mathbb{R}^n \times (0,\infty) \rightrightarrows \mathbb{R}^m$ is a descent-oriented ϵ-subdifferential for f if

(G1) Outer joint limit in (x, ϵ) stays in the Clarke subdifferential:

$$\limsup_{\epsilon \downarrow 0,\, x \to \bar{x}} G(x, \epsilon) \subset \partial f(\bar{x})$$

(G2) Separate limits yield the minimal norm subgradient:

$$\lim_{\epsilon \downarrow 0} \left(\limsup_{x \to \bar{x}} G(x, \epsilon) \right) = \operatorname{argmin}\{\|v\| \mid v \in \partial f(\bar{x})\}$$

descent

stability in x

steepest descent direction $G : (x, \epsilon) \mapsto \operatorname{argmin}\{\|v\| \mid v \in \partial f(x)\}$ violates (G2)

Abstract theory of stable descent

The framework covers:

- **Goldstein direction**: $G : (x, \epsilon) \mapsto \operatorname{argmin}\{\|v\| \mid v \in \partial_\epsilon^G f(x)\}$
- **Bundle direction** (when f is convex): $G : (x, \epsilon) \mapsto \operatorname{argmin}\{\|v\| \mid v \in \partial_\epsilon f(x)\}$
- **Gradient of Moreau envelope** (when f is convex): $G : (x, \epsilon) \mapsto \nabla e_\epsilon f(x)$ with $e_\epsilon f(x) := \inf_z \{f(z) + \|z - x\|^2/(2\epsilon)\}$

Abstract theory of stable descent

Iterate scheme: $x^{k+1} = x^k - \eta_k g^k$ for some $g^k \in G(x^k, \epsilon_k)$

Asymptotic convergence: With a proper line search scheme to find ϵ_k and η_k, any accumulation point $\bar{x}$ of $\{x^k\}$ is a stationary point, i.e., $0 \in \partial f(\bar{x})$.

Abstract theory & new construction for stable descent methods of nonsmooth optimization?

$$S_k = \partial^G_{\epsilon_k} f(x^k) = \mathrm{conv}\left\{ \bigcup_{\|z-x^k\|\le\epsilon_k} \partial f(z) \right\}$$

Almost impossible to compute (deterministically)

$$S_k = \partial_{\epsilon_k} f(x^k) = \{ v \mid f(z) \ge f(x^k) + v^\top (z - x^k) - \epsilon_k, \forall z \}$$

Only well understood (complexity & convergence rate) for weakly convex optimization

New construction: a toy example

For a piecewise smooth function $f(x) = \max\{f_1(x), f_2(x)\}$,

$$\partial f(x) = \left\{ \bar{y}_1 \nabla f_1(x) + \bar{y}_2 \nabla f_2(x) \;\middle|\; \bar{y} \in \operatorname*{argmax}_{\{y \geq 0 | y_1 + y_2 = 1\}} \left[y_1 f_1(x) + y_2 f_2(x) \right] \right\},$$

New construction: a toy example

For a piecewise smooth function $f(x) = \max\{f_1(x), f_2(x)\}$,

$$\partial f(x) = \left\{ \bar{y}_1 \nabla f_1(x) + \bar{y}_2 \nabla f_2(x) \;\middle|\; \bar{y} \in \underset{\{y \geq 0 | y_1 + y_2 = 1\}}{\operatorname{argmax}} \left[y_1 f_1(x) + y_2 f_2(x) \right] \right\},$$

needs to be stabilized & yields descent

Nesterov's smoothing $\bar{y} = \underset{\{y \geq 0 | y_1 + y_2 = 1\}}{\operatorname{argmax}} \left[y_1 f_1(x) + y_2 f_2(x) - \epsilon \phi(y) \right]$

for some strongly cvx ϕ may not yield a descent direction of f

New construction: a toy example

For a piecewise smooth function $f(x) = \max\{f_1(x), f_2(x)\}$,

$$\partial f(x) = \left\{ \bar{y}_1 \nabla f_1(x) + \bar{y}_2 \nabla f_2(x) \;\middle|\; \bar{y} \in \underset{\{y \geq 0 | y_1 + y_2 = 1\}}{\operatorname{argmax}} \left[y_1 f_1(x) + y_2 f_2(x) \right] \right\},$$

For any $\epsilon > 0$, define

Subgradient regularization

$$G(x, \epsilon) = \left\{ \bar{y}_1^\epsilon \nabla f_1(x) + \bar{y}_2^\epsilon \nabla f_2(x) \;\middle|\; \bar{y}^\epsilon \in \underset{\{y \geq 0 | y_1 + y_2 = 1\}}{\operatorname{argmax}} \left[y_1 f_1(x) + y_2 f_2(x) - \frac{\epsilon}{2} \| y_1 \nabla f_1(x) + y_2 \nabla f_2(x) \|^2 \right] \right\}$$

New construction: a toy example

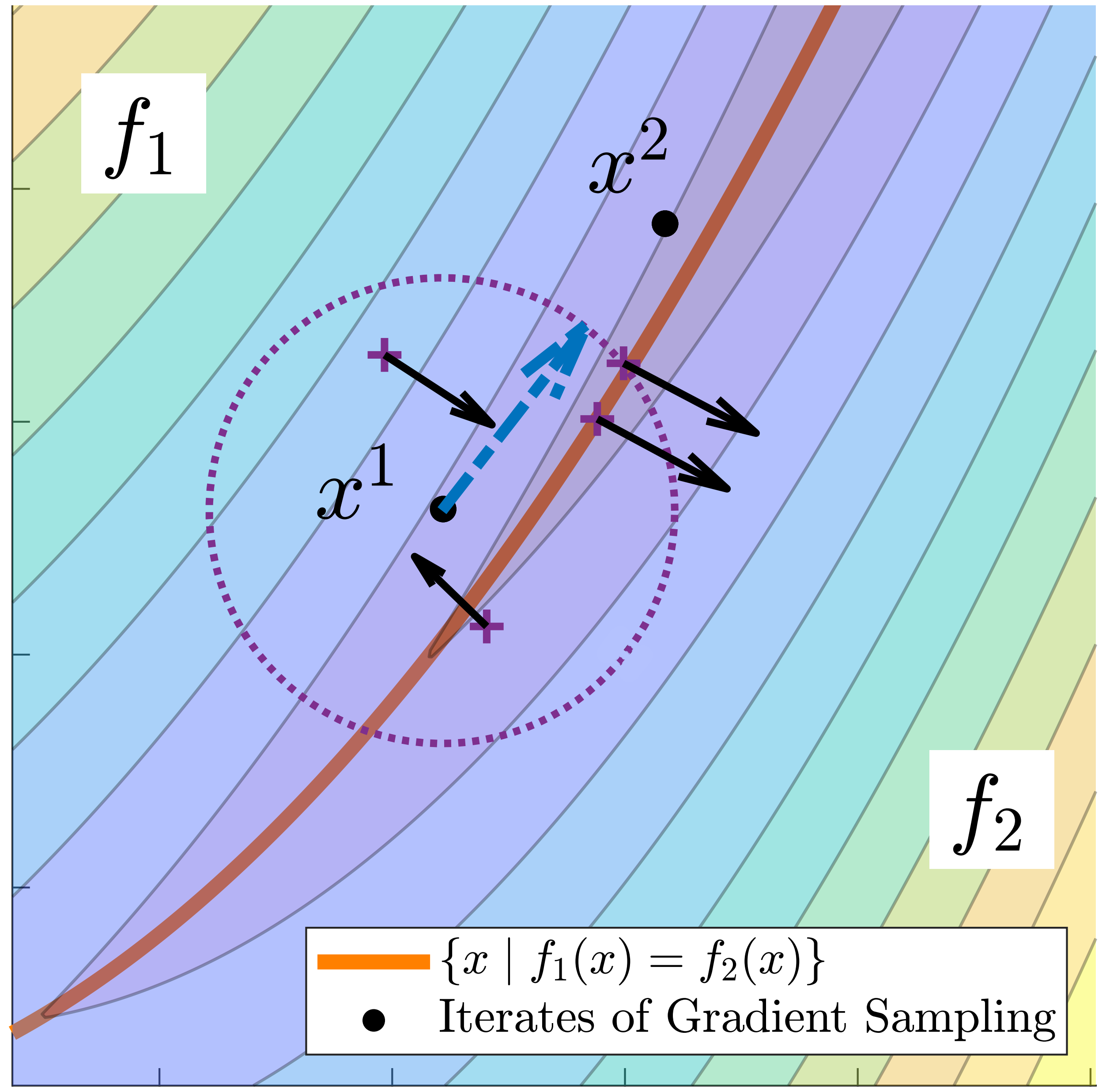
For any $\epsilon > 0$, define

$$G(x,\epsilon) = \left\{ \bar{y}_1^\epsilon \nabla f_1(x) + \bar{y}_2^\epsilon \nabla f_2(x) \;\middle|\; \bar{y}^\epsilon \in \underset{\{y \geq 0 | y_1 + y_2 = 1\}}{\operatorname{argmax}} \left[y_1 f_1(x) + y_2 f_2(x) - \frac{\epsilon}{2} \| y_1 \nabla f_1(x) + y_2 \nabla f_2(x) \|^2 \right] \right\}$$

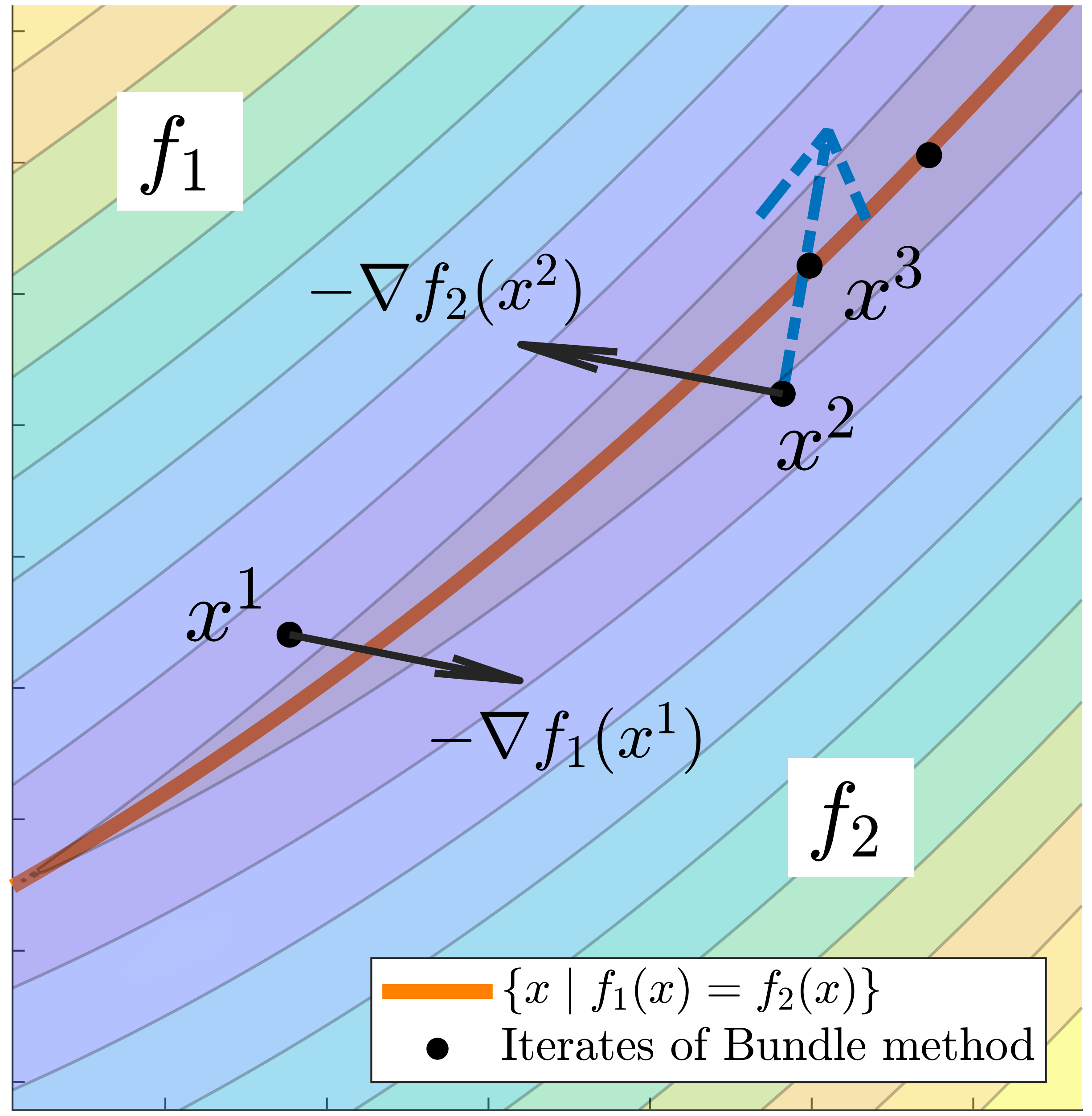
Fact: G is a descent-oriented ϵ-subdifferential, yielding a **stable descent** direction

New construction: a toy example

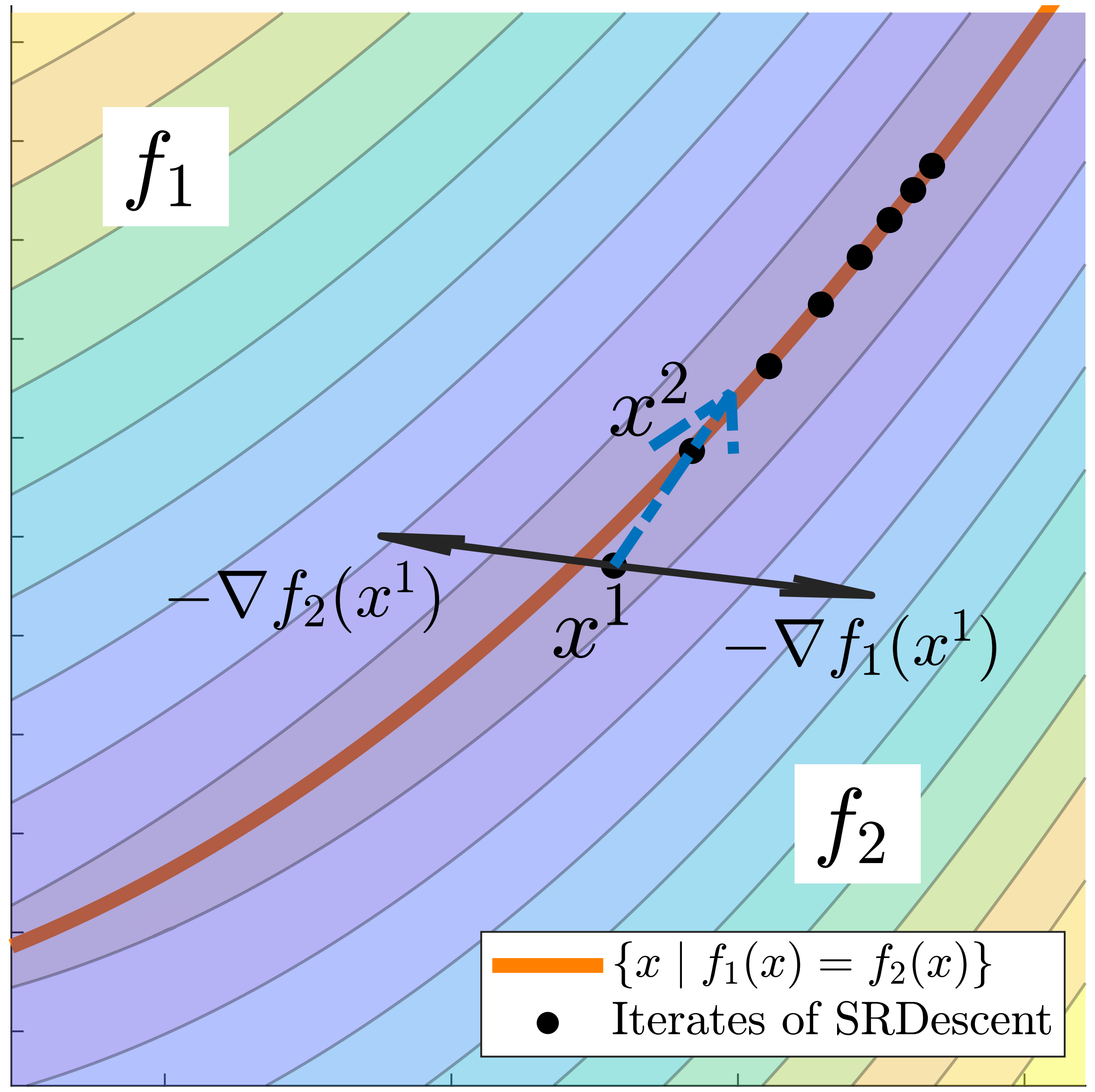
$$G(x,\epsilon) = \left\{ \bar{y}_1^{\epsilon}\,\nabla f_1(x) + \bar{y}_2^{\epsilon}\,\nabla f_2(x) \;\middle|\; \bar{y}^{\epsilon} \in \operatorname*{argmax}_{\{y \geq 0 \mid y_1 + y_2 = 1\}} \left[y_1 f_1(x) + y_2 f_2(x) - \frac{\epsilon}{2} \|y_1 \nabla f_1(x) + y_2 \nabla f_2(x)\|^2 \right] \right\}$$



Gradient Sampling



Bundle method



Subgradient Regularization

combining (sub)gradients at nearby points

Reduction to the prox-linear method

Prox-linear method to solve $f(x) = \max\{f_1(x), f_2(x)\}$:

(Fletcher, 1982, Burke and Ferris, 1995, Lewis and Wright, 2016, Drusvyatskiy and Paquette, 2019…)

linearization

$$x^{k+1} = \operatorname*{argmin}_x \left\{ \max_{1\le i\le 2} \left\{ f_i(x^k) + \nabla f_i(x^k)^\top (x - x^k) \right\} + \frac{1}{2\epsilon}\|x - x^k\|^2 \right\},$$

Reduction to the prox-linear method

Prox-linear method to solve $f(x) = \max\{f_1(x), f_2(x)\}$:

(Fletcher, 1982, Burke and Ferris, 1995, Lewis and Wright, 2016, Drusvyatskiy and Paquette, 2019…)

linearization

$$x^{k+1} = \operatorname*{argmin}_x \left\{ \max_{1 \le i \le 2} \left\{ f_i(x^k) + \nabla f_i(x^k)^\top (x - x^k) \right\} + \frac{1}{2\epsilon}\|x - x^k\|^2 \right\},$$

$$= \operatorname*{argmin}_x \left\{ \max_{\{y \ge 0 \mid y_1 + y_2 = 1\}} \sum_{i=1}^{2} y_i \left(f_i(x^k) + \nabla f_i(x^k)^\top (x - x^k) \right) + \frac{1}{2\epsilon}\|x - x^k\|^2 \right\}$$

Reduction to the prox-linear method

Prox-linear method to solve $f(x) = \max\{f_1(x), f_2(x)\}$:

(Fletcher, 1982, Burke and Ferris, 1995, Lewis and Wright, 2016, Drusvyatskiy and Paquette, 2019...)

linearization

$$x^{k+1} = \operatorname*{argmin}_{x} \left\{ \max_{1 \le i \le 2} \left\{ f_i(x^k) + \nabla f_i(x^k)^\top (x - x^k) \right\} + \frac{1}{2\epsilon} \|x - x^k\|^2 \right\},$$

$$= \operatorname*{argmin}_{x} \left\{ \max_{\{y \ge 0 \mid y_1 + y_2 = 1\}} \sum_{i=1}^{2} y_i \left(f_i(x^k) + \nabla f_i(x^k)^\top (x - x^k) \right) + \frac{1}{2\epsilon} \|x - x^k\|^2 \right\}$$

$$= x^k - \epsilon \left[\bar{y}_1 \nabla f_1(x^k) + \bar{y}_2 \nabla f_2(x^k) \right]$$

where $\bar{y} \in \operatorname*{argmax}_{y \in \Delta^2} \left\{ y_1 f_1(x) + y_2 f_2(x) - \frac{\epsilon}{2} \|y_1 \nabla f_1(x) + y_2 \nabla f_2(x)\|^2 \right\}$

Reduction to the prox-linear method

Prox-linear method to solve $f(x) = \max\{f_1(x), f_2(x)\}$:

(Fletcher, 1982, Burke and Ferris, 1995, Lewis and Wright, 2016, Drusvyatskiy and Paquette, 2019...)

linearization

$$x^{k+1} = \operatorname*{argmin}_x \left\{ \max_{1 \le i \le 2} \left\{ f_i(x^k) + \nabla f_i(x^k)^\top (x - x^k) \right\} + \frac{1}{2\epsilon} \|x - x^k\|^2 \right\},$$

$$= \operatorname*{argmin}_x \left\{ \max_{\{y \ge 0 | y_1 + y_2 = 1\}} \sum_{i=1}^{2} y_i \big(f_i(x^k) + \nabla f_i(x^k)^\top (x - x^k) \big) + \frac{1}{2\epsilon} \|x - x^k\|^2 \right\}$$

$$= x^k - \epsilon \left[\bar{y}_1 \nabla f_1(x^k) + \bar{y}_2 \nabla f_2(x^k) \right]$$

= Subgradient regularization

where $\bar{y} \in \operatorname*{argmax}_{y \in \Delta^2} \left\{ y_1 f_1(x) + y_2 f_2(x) - \frac{\epsilon}{2} \|y_1 \nabla f_1(x) + y_2 \nabla f_2(x)\|^2 \right\}$

Reduction to the prox-linear method

- A dual interpretation of the prox-linear method
- can be extended to composite function (convex) ∘ (smooth) by conjugate duality

$$= \operatorname*{argmin}_{x} \left\{ \max_{\{y \ge 0 | y_1 + y_2 = 1\}} \sum_{i=1}^{2} y_i \left(f_i(x^k) + \nabla f_i(x^k)^\top (x - x^k) \right) + \frac{1}{2\epsilon} \|x - x^k\|^2 \right\}$$

$$= x^k - \epsilon \left[\bar{y}_1 \nabla f_1(x^k) + \bar{y}_2 \nabla f_2(x^k) \right]$$

= Subgradient regularization

where $\bar{y} \in \operatorname*{argmax}_{y \in \Delta^2} \left\{ y_1 f_1(x) + y_2 f_2(x) - \frac{\epsilon}{2} \|y_1 \nabla f_1(x) + y_2 \nabla f_2(x)\|^2 \right\}$

Subgradient regularization beyond composite structure

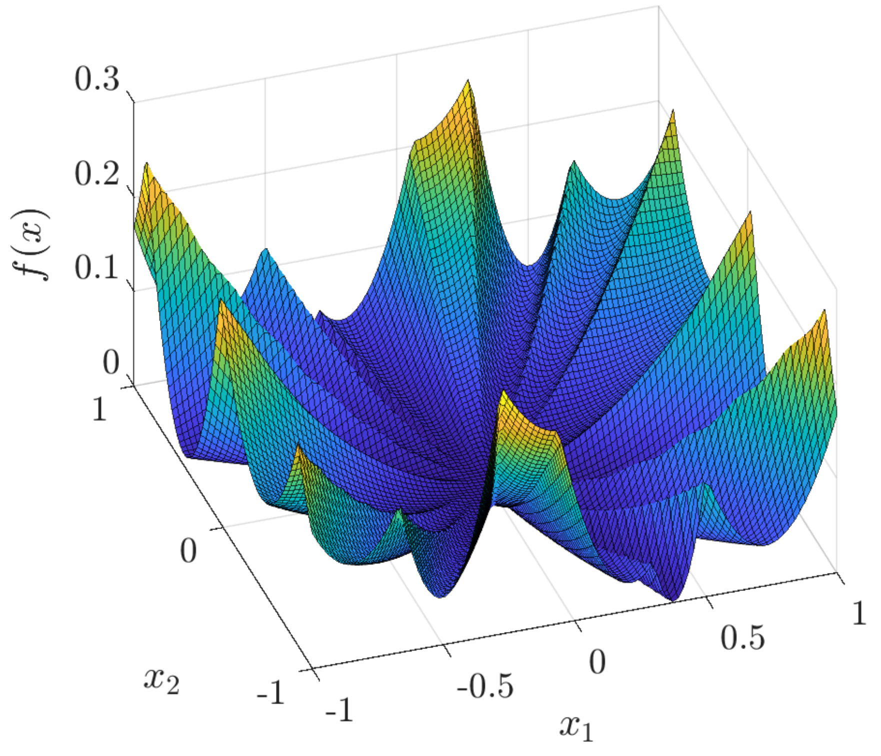
For the marginal function:

$$f(x) \triangleq \left[\max_{y} \ \varphi_0(x,y) \ \text{ subject to } \varphi_j(x,y) \leq 0,\, j = 1,\cdots,r \right]$$

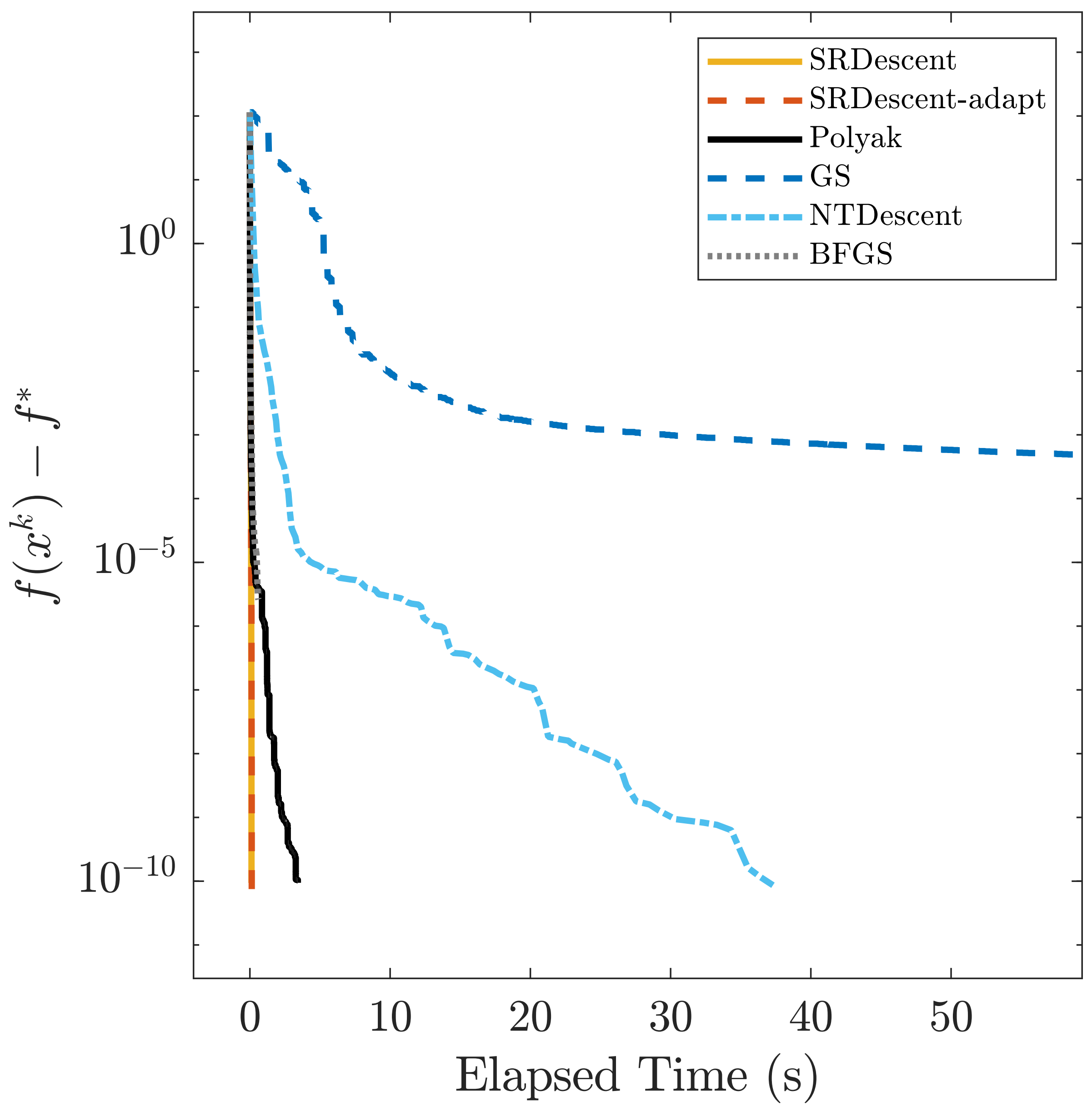
- Characterize $\partial f(x)$, and apply subgradient regularization
- Yield a stable descent direction
- Does not need f to be (weakly) convex
- Can be applied to the two-stage stochastic programs (f : the recourse function)

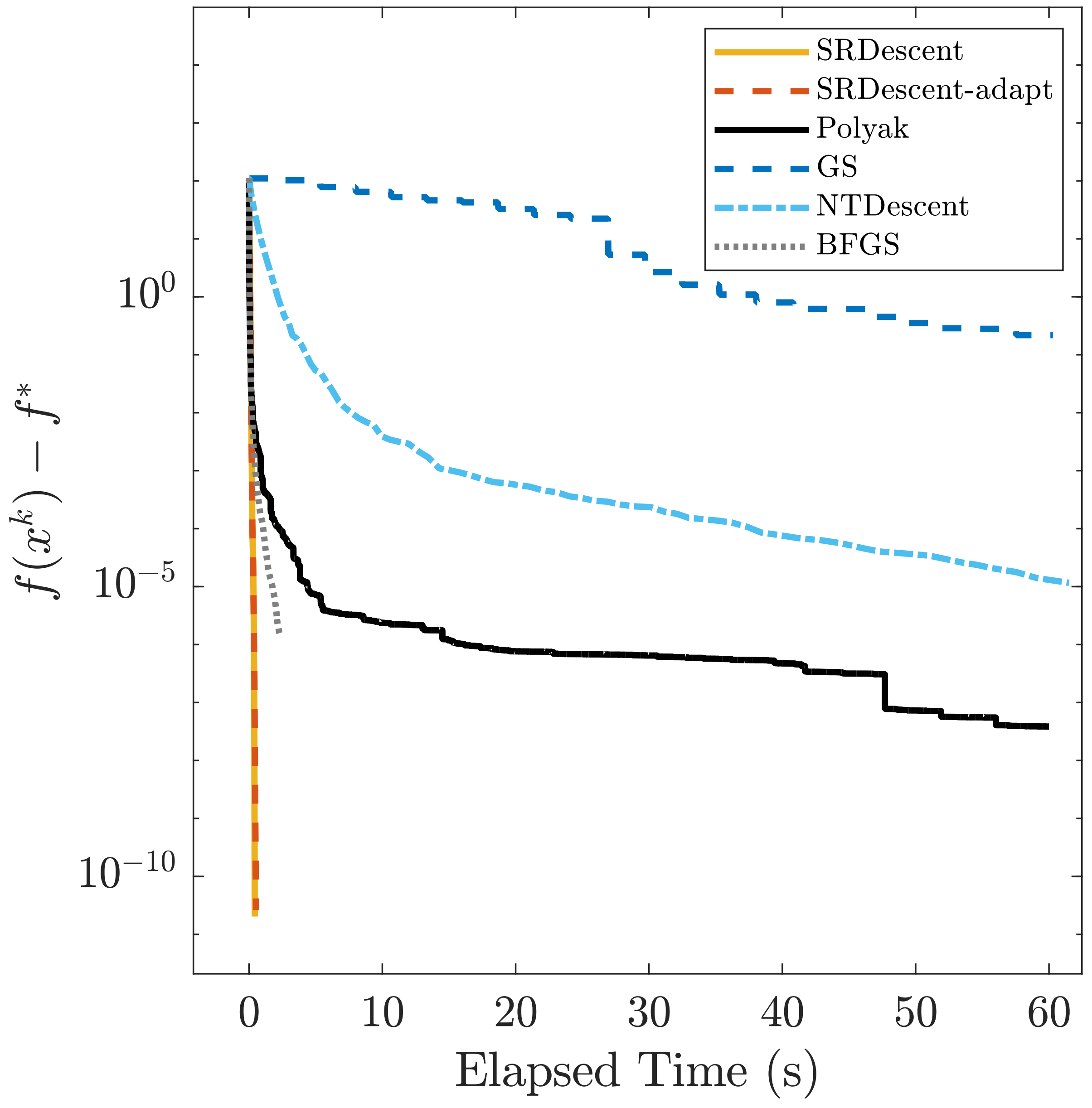
Numerical results for nonconvex cases

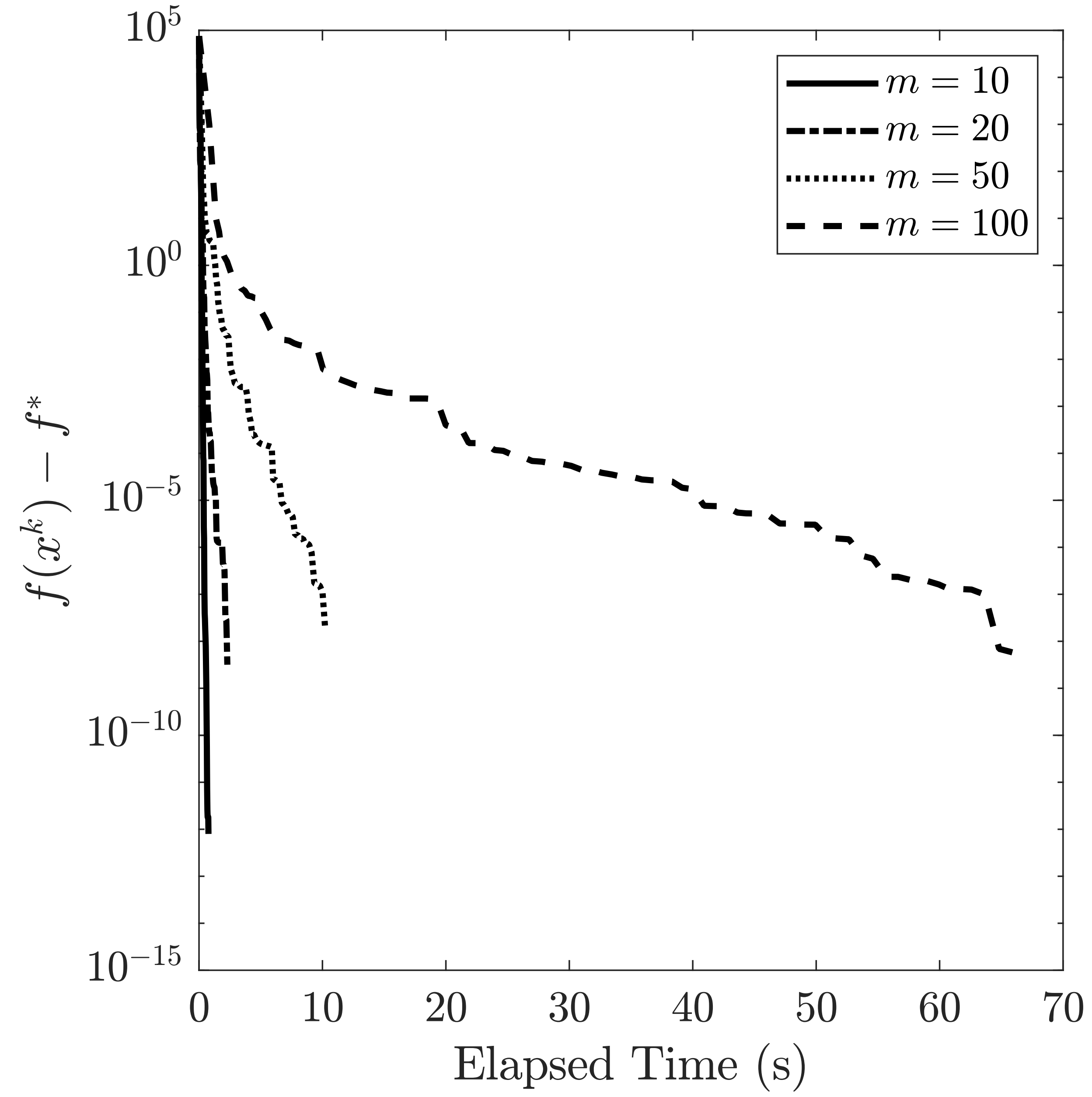
$$f(x) = \min_{1\le i\le m} \frac{1}{2}\|A_i x - b_i\|^2$$



$$f(x) = \min_{y\in\mathbb{R}^m} \left\{(c + Dx)^\top y + \frac{1}{2} y^\top Q y + \|x\|^4\right\}$$
$$\text{subject to } \; b - Ax - \mathbf{1} \le Wy \le b - Ax\,.$$







Nonsmooth Optimization

Part 1: Stable Descent Directions

when the function is also nonconvex

Part 2: Benefit from Nonsmoothness

if the (sparsity) structure is properly preserved

Superquantile

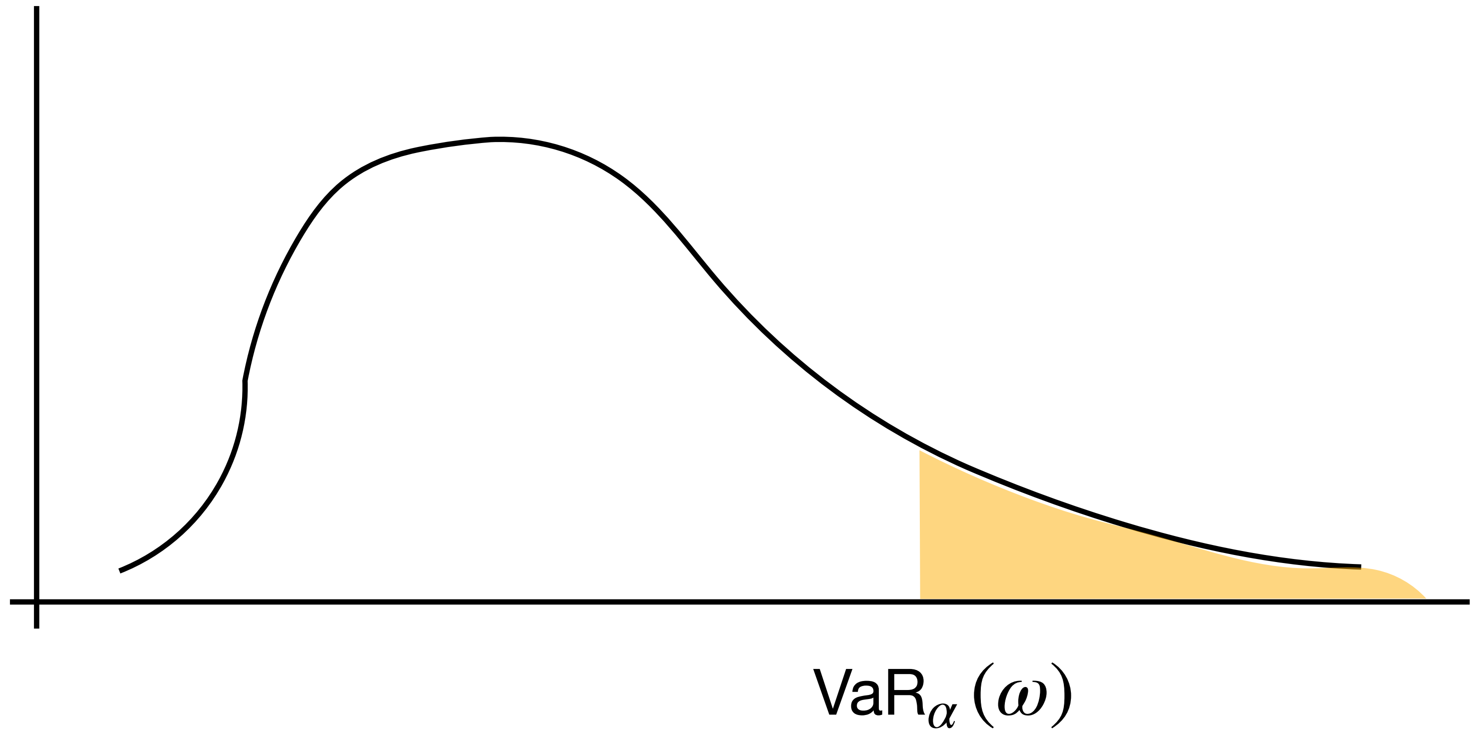
- Superquantile / conditional value-at-risk (CVaR)

[Ben-Tal and Teboulle, Rockafellar and Uryasev, Rockafellar and Royset…]

$$\text{CVaR}_\alpha(\omega) = \text{Average of the worst } (1-\alpha)100\% \text{ outcomes of } \omega$$

$$= \min_\eta \eta + \frac{1}{1-\alpha}\mathbb{E}[\max(\omega - \eta, 0)]$$

- ``top-k-sum" in machine learning



Superquantile optimization

$$\begin{aligned} \min_{x \in X} \quad & \theta(x) + \text{CVaR}_{\alpha_0}[\, f_0(x, \omega) \,] \\ \text{s.t.} \quad & \text{CVaR}_{\alpha_i}[\, f_i(x, \omega) \,] \le r_i, \quad i = 1, \cdots, L \end{aligned}$$

- The problem is convex if θ, $\{f_i(\bullet, \omega)\}_{i=0}^{L}$, X are convex
- Financial decisions, operational plans, military strategies, engineering designs, machine learning, statistical models…[see the survey paper by Royset (2023)]

Expectation vs Superquantile

$$\mathbb{E}[\, f(x,\omega)\,] \approx \frac{1}{S}\sum_{s=1}^{S} f(x,\omega^s)$$

- **Separable**: samples are equally important

$$\text{CVaR}_\alpha\,[\, f(x,\omega)\,] \approx \frac{1}{\lfloor 1/(1-\alpha)\rfloor}\sum_{s=1}^{\lfloor 1/(1-\alpha)\rfloor} f(x,\omega^{[s]})$$

where $f(x,\omega^{[1]}) \geq f(x,\omega^{[2]}) \geq \cdots \geq f(x,\omega^{[S]})$

- **Non-separable**: samples are not equally important —> only care about **tail** expectation

Expectation vs Superquantile

$$\mathbb{E}[\, f(x,\omega)\,] \approx \frac{1}{S}\sum_{s=1}^{S} f(x,\omega^s)$$

- Can take an **arbitrary** sample to estimate the function value and the (sub)gradient

$$\text{CVaR}_\alpha[\, f(x,\omega)\,] \approx \frac{1}{\lfloor 1/(1-\alpha)\rfloor}\sum_{s=1}^{\lfloor 1/(1-\alpha)\rfloor} f(x,\omega^{[s]})$$

where $f(x,\omega^{[1]}) \geq f(x,\omega^{[2]}) \geq \cdots \geq f(x,\omega^{[S]})$

- $f(x,\omega^s)$ has to belong to the **right-tail** to generate a non-trivial (sub)gradient

Expectation vs Superquantile

$$\mathbb{E}[\,f(x,\omega)\,] \approx \frac{1}{S}\sum_{s=1}^{S} f(x,\omega^s)$$

$$\text{CVaR}_\alpha\,[\,f(x,\omega)\,] \approx \frac{1}{\lfloor 1/(1-\alpha)\rfloor}\sum_{s=1}^{\lfloor 1/(1-\alpha)\rfloor} f(x,\omega^{[s]})$$

where $f(x,\omega^{[1]}) \geq f(x,\omega^{[2]}) \geq \cdots \geq f(x,\omega^{[S]})$

- **Function evaluations can be expensive**, e.g., recourse functions, neural networks; in fact, even if $f(\,\bullet\,,\omega)$ is affine when the number of scenarios is large.

Superquantile optimization

reduce the number of evaluations for function values and (sub)gradients

a second-order method?

Superquantile optimization

reduce the number of evaluations for function values and (sub)gradients

↓

a second-order method?

↓

expensive to formulate ``Hessian” matrices + solve linear equations?

Superquantile optimization

reduce the number of evaluations for function values and (sub)gradients

↓

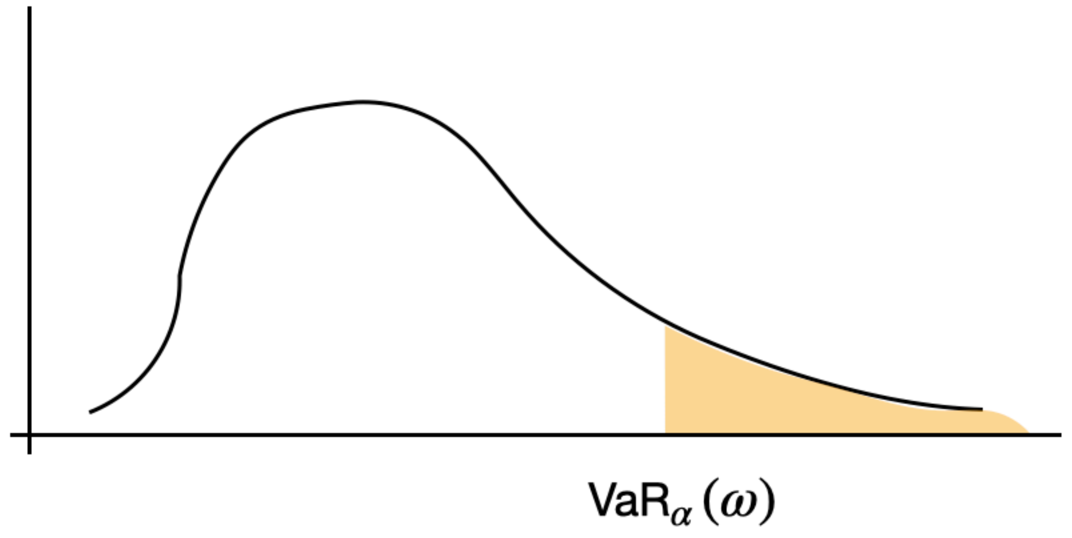
second order method?

↓

cheap

~~expensive~~ to formulate “Hessian” matrices + solve linear equations!

Blessing of the tail risk: “Hessian” is sparse



only a small proportion of scenarios matters

Partial augmented Lagrangian

Consider a simplified problem: linear objective, one CVaR constraint, no side constraints

$$\begin{array}{ll} \underset{x}{\text{minimize}} & c^\top x \\ \text{subject to} & CVaR_\alpha \left[\{a_i^\top x + b_i\}_{i=1}^S \right] \le r \end{array}$$

Partial augmented Lagrangian

$$\begin{array}{ll} \underset{x}{\text{minimize}} & c^\top x \\ \text{subject to} & CVaR_\alpha \left[\{a_i^\top x + b_i\}_{i=1}^{S} \right] \leq r \end{array}$$

$\Rightarrow$

$$\begin{array}{ll} \underset{x,y}{\text{minimize}} & c^\top x \\ \text{subject to} & y = Ax + b, \quad CVaR_\alpha \left[y \right] \leq r \end{array}$$

Partial augmented Lagrangian

$$\underset{x}{\text{minimize}} \quad c^\top x \qquad \text{subject to} \quad CVaR_\alpha\left[\{a_i^\top x + b_i\}_{i=1}^S\right] \leq r$$

$$\underset{x,y}{\text{minimize}} \quad c^\top x \qquad \text{subject to} \quad y = Ax + b, \quad CVaR_\alpha\left[y\right] \leq r$$

partial augmented Lagrangian

(partial Moreau envelope of the dual)

$$\min_x \; c^\top x + \frac{\sigma}{2}\left\| \Pi_{\text{CVaR}_\alpha(\cdot)\leq r}(Ax + b - \widetilde{\lambda}/\sigma) - (Ax + b - \widetilde{\lambda}/\sigma)\right\|^2$$

projection onto the top-k-sum level set, **preserve nonsmoothness**

Partial augmented Lagrangian

$$\underset{x}{\text{minimize}} \quad c^\top x$$
$$\text{subject to} \quad CVaR_\alpha\left[\{a_i^\top x + b_i\}_{i=1}^S\right] \leq r$$

$$\underset{x,y}{\text{minimize}} \quad c^\top x$$
$$\text{subject to} \quad y = Ax + b, \quad CVaR_\alpha\left[y\right] \leq r$$

$$\min_x \; c^\top x + \frac{\sigma}{2}\left\| \Pi_{\mathrm{CVaR}_\alpha(\bullet)\leq r}(Ax + b - \widetilde{\lambda}/\sigma) - (Ax + b - \widetilde{\lambda}/\sigma) \right\|^2$$

continuously differentiable

$$c + \sigma A^\top \left[Ax + b - \widetilde{\lambda}/\sigma - \Pi_{\mathrm{CVaR}_\alpha(\bullet)\leq r}(Ax + b - \widetilde{\lambda}/\sigma) \right] = 0$$

optimality condition

Partial augmented Lagrangian

$$\underset{x}{\text{minimize}} \quad c^\top x$$
$$\text{subject to} \quad CVaR_\alpha\left[\{a_i^\top x + b_i\}_{i=1}^S\right] \leq r$$

$$\underset{x,y}{\text{minimize}} \quad c^\top x$$
$$\text{subject to} \quad y = Ax + b, \quad CVaR_\alpha[y] \leq r$$

$$\min_x \quad c^\top x + \frac{\sigma}{2}\left\| \Pi_{\mathrm{CVaR}_\alpha(\bullet)\leq r}(Ax + b - \widetilde{\lambda}/\sigma) - (Ax + b - \widetilde{\lambda}/\sigma) \right\|^2$$

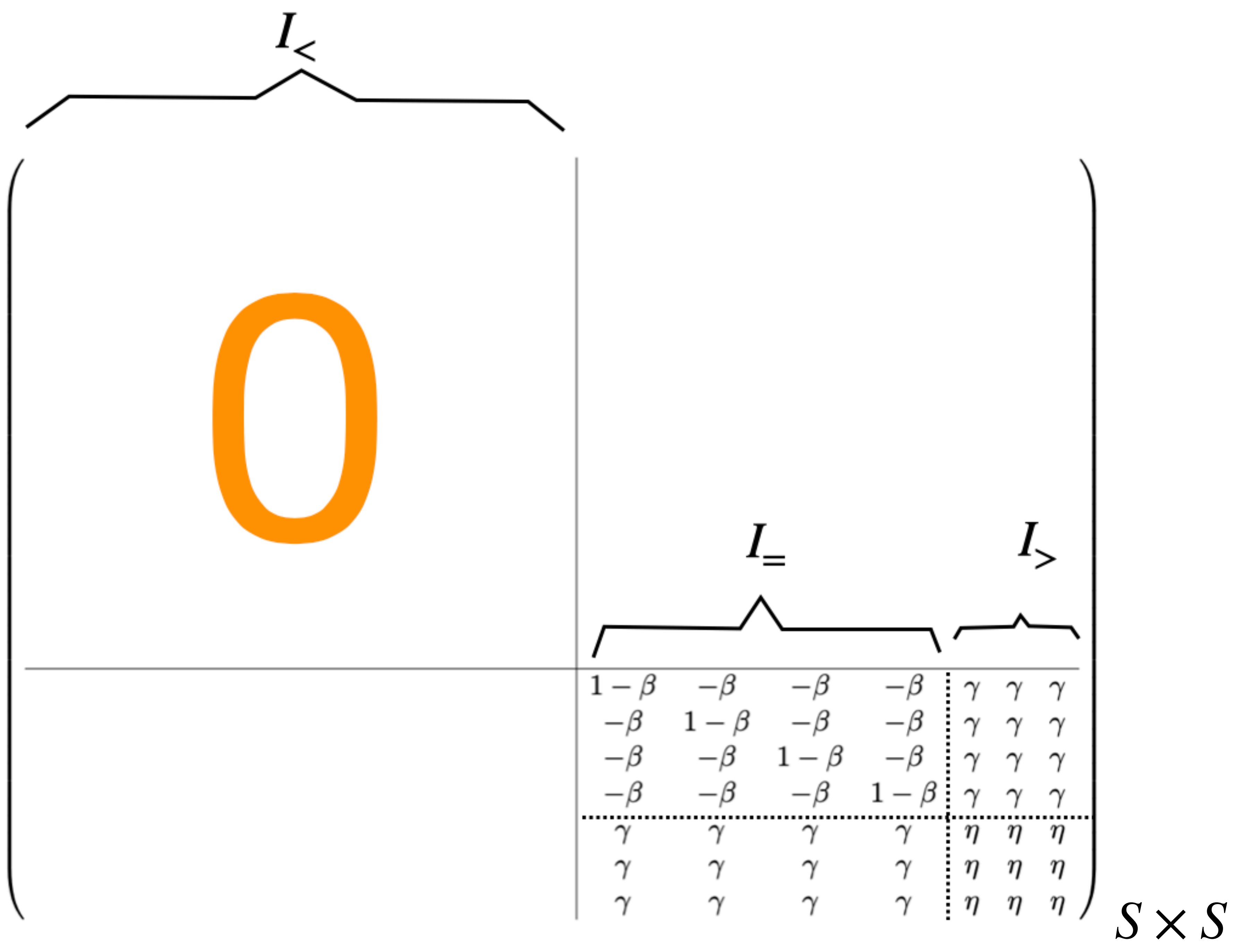
$$c + \sigma A^\top\left[Ax + b - \widetilde{\lambda}/\sigma - \Pi_{\mathrm{CVaR}_\alpha(\bullet)\leq r}(Ax + b - \widetilde{\lambda}/\sigma)\right] = 0$$

piecewise affine equation $\rightarrow$ semismooth Newton

Generalized Jacobian (``Hessian")

$$A^\top \left[Ax - \Pi_{\mathrm{CVaR}_\alpha(\bullet)\le r}\left(Ax + b - \widetilde{\lambda}/\sigma\right) \right] = \mathrm{rhs}$$

- Generalized Jacobian: $A^\top (I - J) A$



Generalized Jacobian (``Hessian")

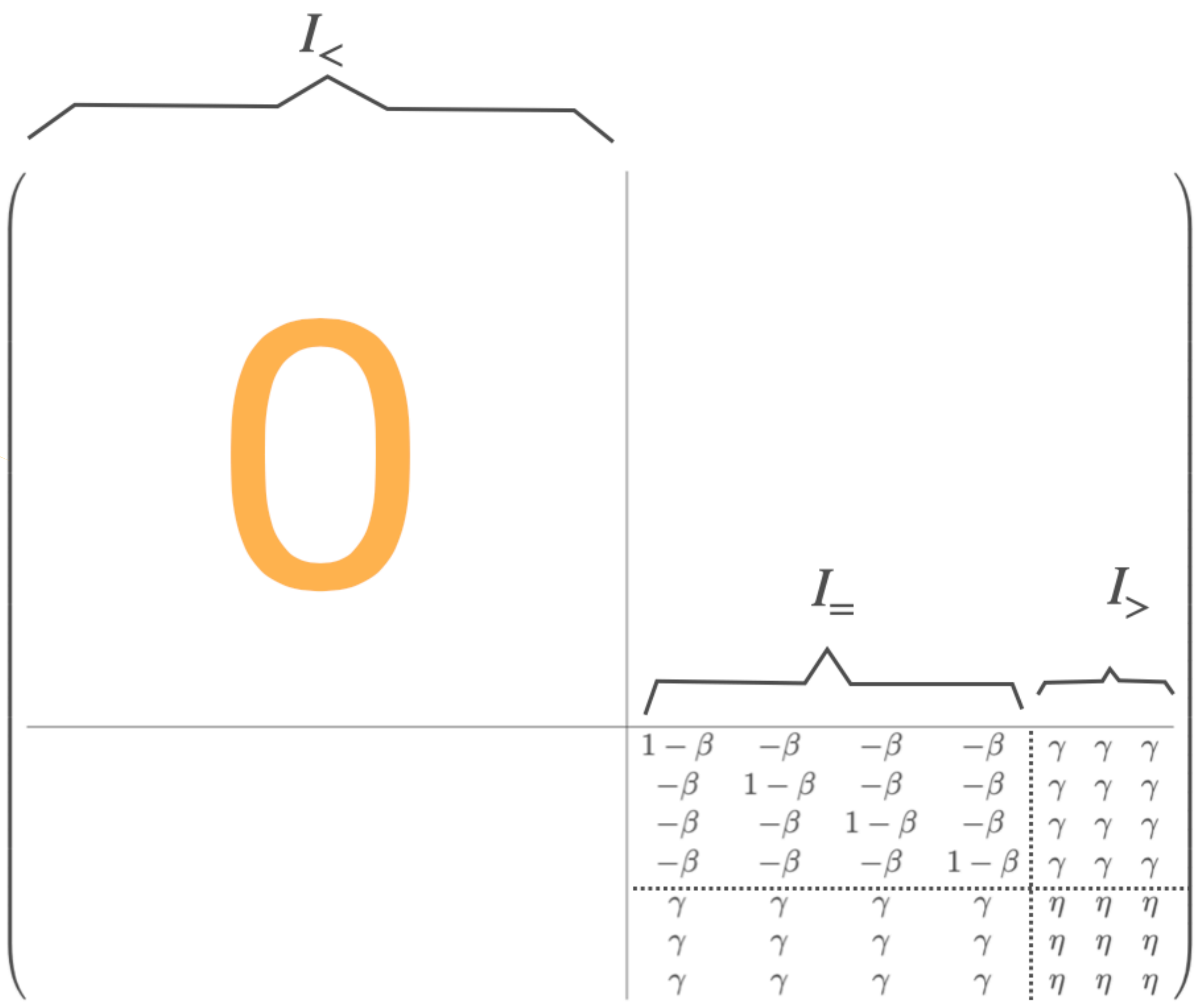
Generalized Jacobian: $A^\top (I - J) A$

$$||$$

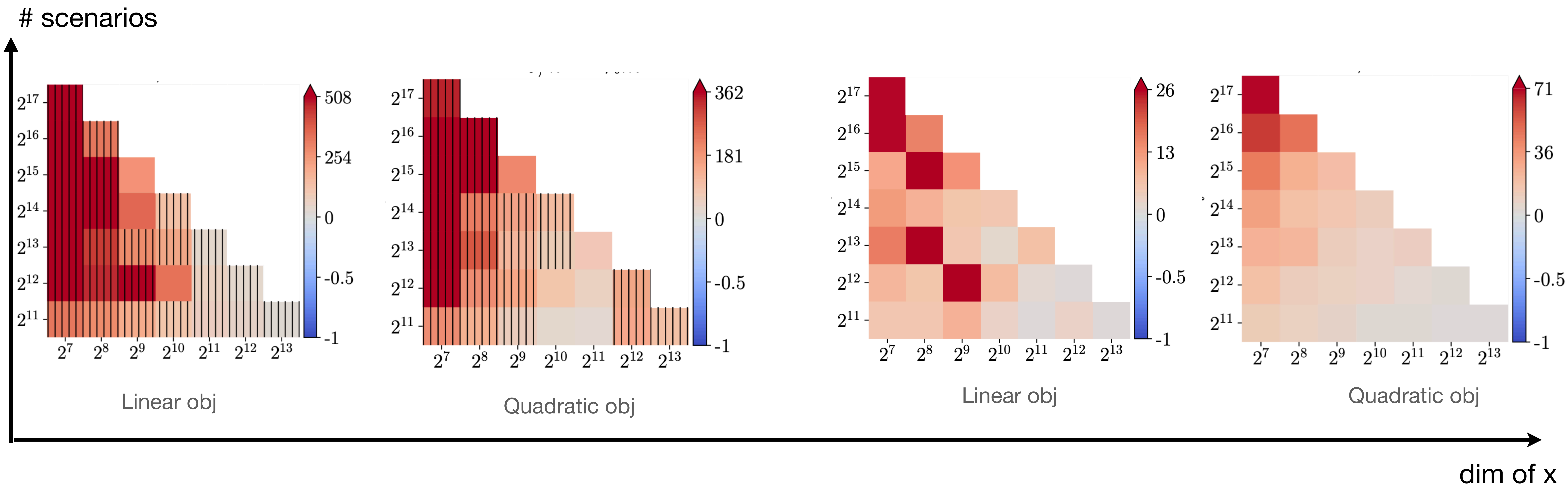
$$\widetilde{A}^\top \widetilde{A}$$

where $\widetilde{A} \in \mathbb{R}^{(|I_=|+1)\times n}$

usually $\ll S$!!



Comparison with OSQP & Gurobi



Compare with OSQP for low-accurate solutions (1e-3)

Compare with Gurobi for high-accurate solutions (1e-6)

Thank you!

Hanyang Li and Ying Cui. *Subgradient Regularization: A Descent-Oriented Subgradient Method for Nonsmooth Optimization* (2025).

Hanyang Li and Ying Cui. *Variational Theory and Algorithms for a Class of Asymptotically Approachable Nonconvex Problems.* Mathematics of Operations Research (2025).

Hanyang Li and Ying Cui. *A Decomposition Algorithm for Two-Stage Stochastic Programs with Nonconvex Recourse Functions*. SIAM Journal on Optimization (2024).

Jake Roth and Ying Cui. *Optimization with superquantile constraints: a fast computational approach* (2024).

Algorithm

for $k = 0,1,\cdots$

for $i = 0,1,\cdots$

Generate a direction $g^{k,i} \in G(x^k, \epsilon_{k,0}\, 2^{-i})$

if $\exists \eta_k \in \left\{\epsilon_{k,0}, \cdots, \epsilon_{k,0}\, 2^{-i}\right\}$ with $f\left(x^k - \eta_k g^{k,i}\right) \le f(x^k) - \alpha \eta_k \|g^{k,i}\|^2$

Update $x^{k+1} = x^k - \eta_k g^{k,i}$ and **break**

if $\|g^{k,i}\| \le \nu_k$

line-search

Update $\epsilon_{k+1,0} = \epsilon_{k,0}/2$ and $\nu_{k+1} = \nu_k/2$

else set $\epsilon_{k+1,0} = \epsilon_{k,0}$ and $\nu_{k+1} = \nu_k$

Algorithm

for $k = 0,1,\cdots$

  for $i = 0,1,\cdots$

    Generate a direction $g^{k,i} \in G(x^k, \epsilon_{k,0}\, 2^{-i})$

    if $\exists \eta_k \in \left\{\epsilon_{k,0}, \cdots, \epsilon_{k,0}\, 2^{-i}\right\}$ with $f\left(x^k - \eta_k g^{k,i}\right) \le f(x^k) - \alpha\eta_k \|g^{k,i}\|^2$

      Update $x^{k+1} = x^k - \eta_k g^{k,i}$ and **break**

  if $\|g^{k,i}\| \le \nu_k$

line-search

    Update $\epsilon_{k+1,0} = \epsilon_{k,0}/2$ and $\nu_{k+1} = \nu_k/2$

  else set $\epsilon_{k+1,0} = \epsilon_{k,0}$ and $\nu_{k+1} = \nu_k$

- The inner-loop terminates for sufficiently large i ($\exists$ descent directions at x^k)

Algorithm

for $k = 0,1,\cdots$

for $i = 0,1,\cdots$

Generate a direction $g^{k,i} \in G(x^k, \epsilon_{k,0}\, 2^{-i})$

if $\exists \eta_k \in \left\{\epsilon_{k,0}, \cdots, \epsilon_{k,0}\, 2^{-i}\right\}$ with $f\left(x^k - \eta_k g^{k,i}\right) \leq f(x^k) - \alpha\eta_k \|g^{k,i}\|^2$

Update $x^{k+1} = x^k - \eta_k g^{k,i}$ and **break**

if $\|g^{k,i}\| \leq \nu_k$

line-search

Update $\epsilon_{k+1,0} = \epsilon_{k,0}/2$ and $\nu_{k+1} = \nu_k/2$

else set $\epsilon_{k+1,0} = \epsilon_{k,0}$ and $\nu_{k+1} = \nu_k$

Theorem: Any accumulation point $\bar{x}$ of $\{x^k\}$ is a stationary point, i.e., $0 \in \partial f(\bar{x})$.

Idea: $\{x^k\}$ will not converge to a non-stationary point [$G(x, \epsilon)$ is "stable" in x]:

If x^k close to a non-stationary point $\bar{x}$ $\Rightarrow$ $G(x^k, \epsilon)$ close to $G(\bar{x}, \epsilon)$ [for a fixed $\epsilon > 0$]

$\Rightarrow$ x^k escapes $\bar{x}$ for sufficiently small ϵ