Proximal Algorithms for Large-scale Convex Nonsmooth Optimization

Laurent Condat

Visual Computing Center & AI Initiative
King Abdullah Univ. of Science and Technology (KAUST), Saudi Arabia

Optimization

"Nothing takes place in the world whose meaning is not that of some maximum or minimum"

— **Leonhard Euler** ~1750

Optimization

$$\text{Find } x^\star \in \underset{x \in \mathcal{X}}{\arg\min}\ \Psi(x)$$

Optimization

$$\text{Find } x^\star \in \underset{x \in \mathcal{X}}{\arg\min}\ \Psi(x)$$

$\mathcal{X}$ is a real Hilbert space

Optimization

$$\text{Find } x^\star \in \arg\min_{x \in \mathcal{X}} \Psi(x) = \sum_{m=1}^{M} g_m(L_m x)$$

with

- linear operators $L_m : \mathcal{X} \to \mathcal{U}_m$
- real Hilbert spaces $\mathcal{X}, \mathcal{U}_m$
- functions g_m

Optimization

$$\text{Find } x^\star \in \arg\min_{x \in \mathcal{X}} \Psi(x) = \sum_{m=1}^{M} g_m(L_m x)$$

$$\text{Example: } \Psi(x) = \|Ax - y\|^2 + \|x\|_1$$

Optimization

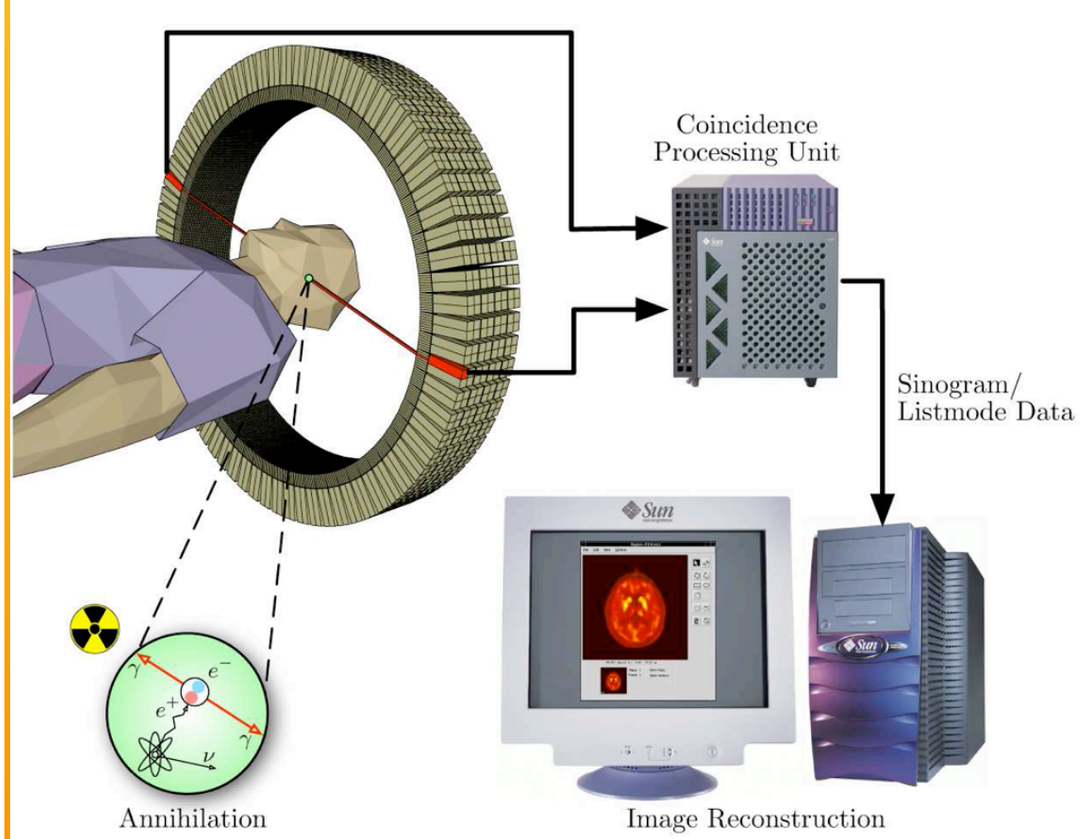
$$\text{Find } x^\star \in \arg\min_{x \in \mathcal{X}} \Psi(x) = \sum_{m=1}^{M} g_m(L_m x)$$

$$\text{Example: } \Psi(x) = \|Ax - y\|^2 + R(x)$$

Motivation: image processing

$$\text{Find } x^\star \in \underset{x \in \mathcal{X}}{\arg\min}\ \Psi(x) = \sum_{m=1}^{M} g_m(L_m x)$$

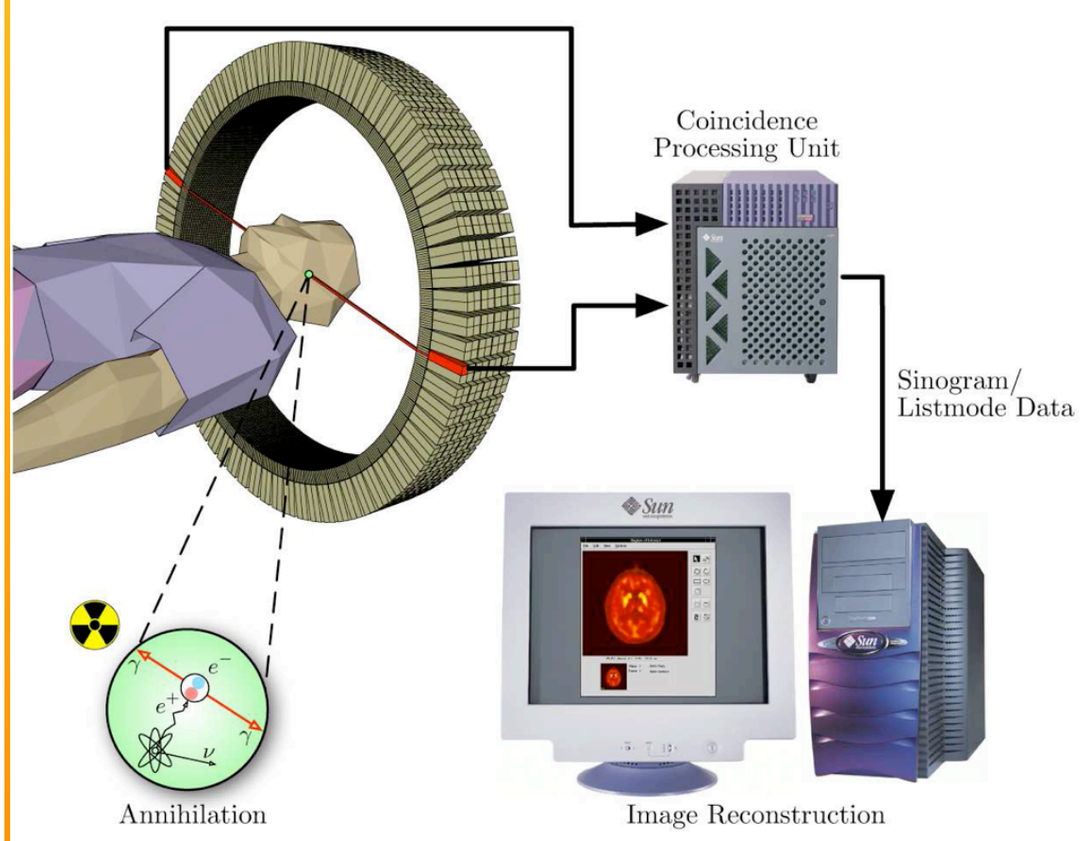
$$\text{Example: } \Psi(x) = \|Ax - y\|^2 + R(x)$$



Motivation: image processing

$$\text{Find } x^\star \in \operatorname*{arg\,min}_{x \in \mathcal{X}} \ \Psi(x) = \sum_{m=1}^{M} g_m(L_m x)$$

$$\text{Example: } \Psi(x) = \|Ax - y\|^2 + R(x)$$



LC, "Discrete total variation: New definition and minimization", 2017

LC, "A generic proximal algorithm...", 2014

Convex optimization

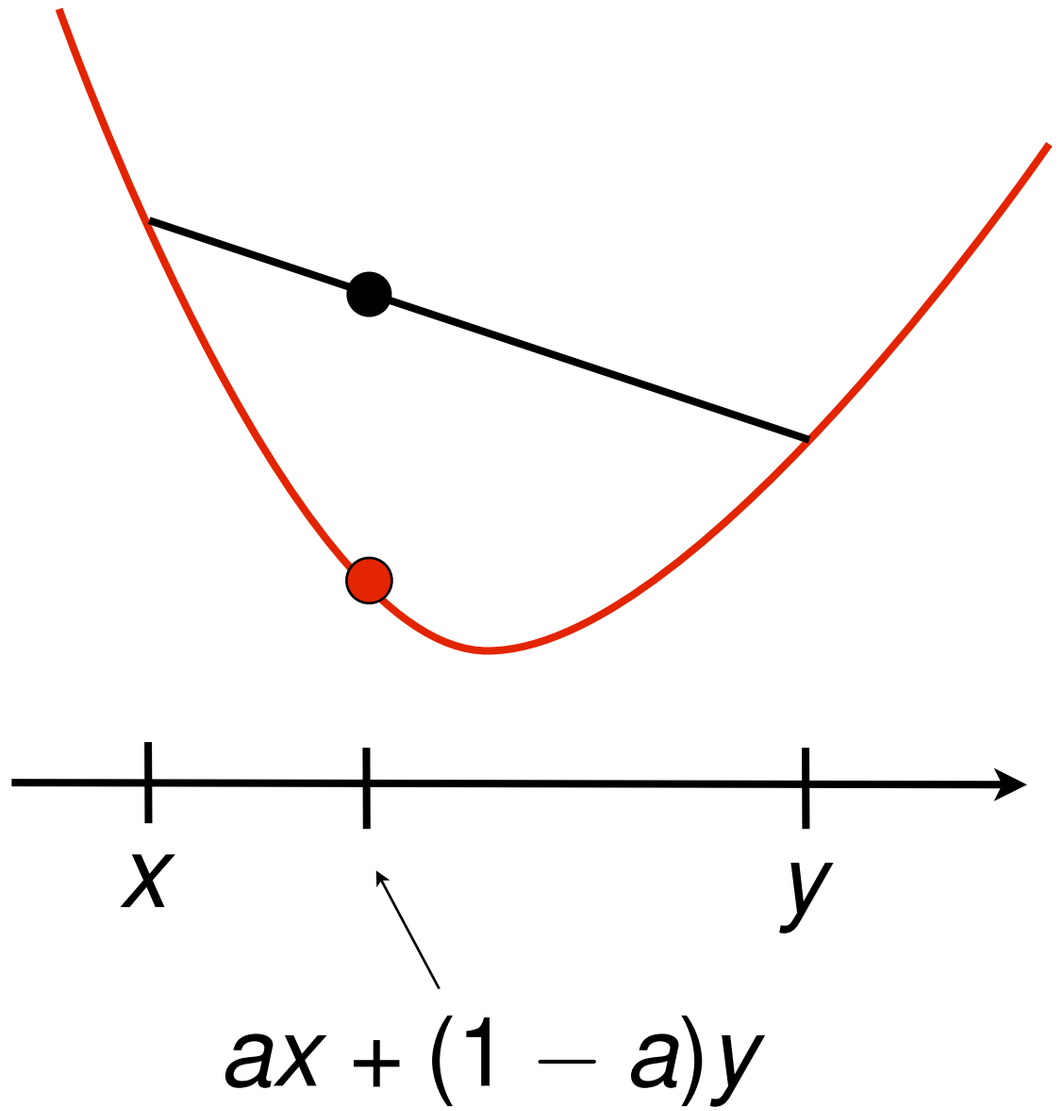
$$\text{Find } x^\star \in \arg\min_{x \in \mathcal{X}} \Psi(x) = \sum_{m=1}^{M} g_m(L_m x)$$

with

- linear operators $L_m : \mathcal{X} \to \mathcal{U}_m$
- real Hilbert spaces $\mathcal{X}, \mathcal{U}_m$
- **convex** functions $g_m : \mathcal{U}_m \to \mathbb{R} \cup \{+\infty\}$

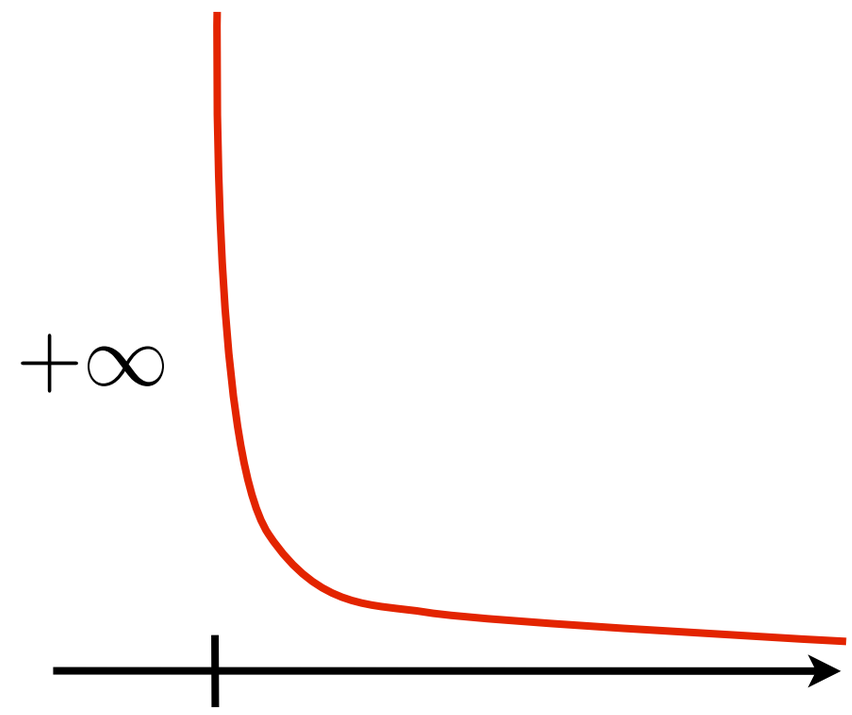
Convex functions

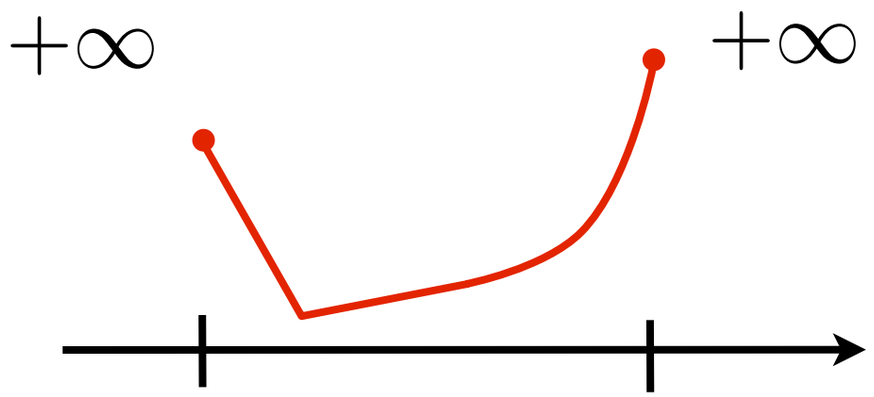
f is **convex** if $\forall x, y \in \mathcal{X}$ and $a \in [0, 1]$,
$f(ax + (1 - a)y) \leq af(x) + (1 - a)f(y)$

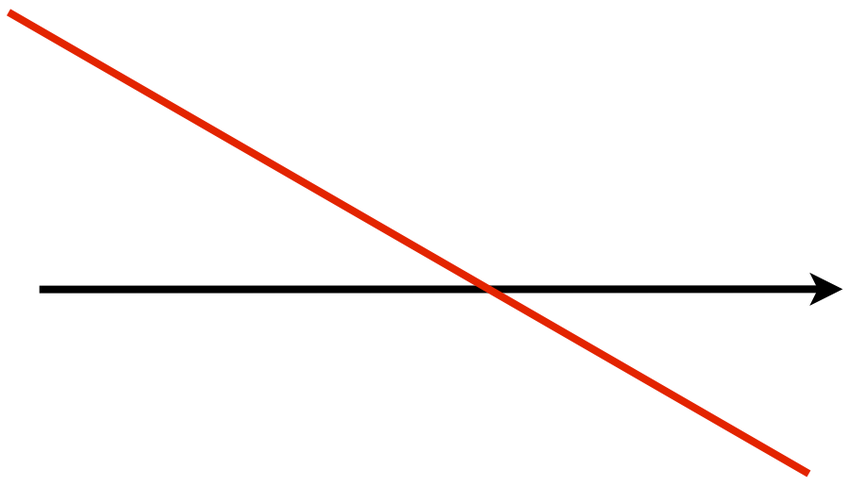


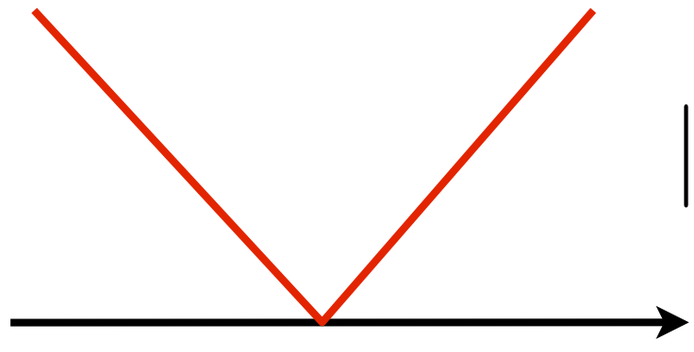
Convex functions

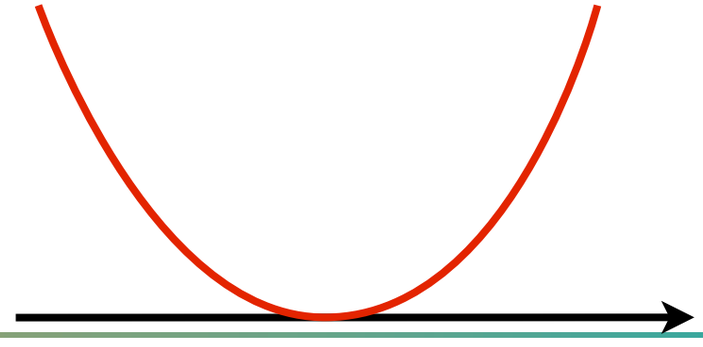
Some convex functions: $\mathbb{R} \to \mathbb{R} \cup \{+\infty\}$







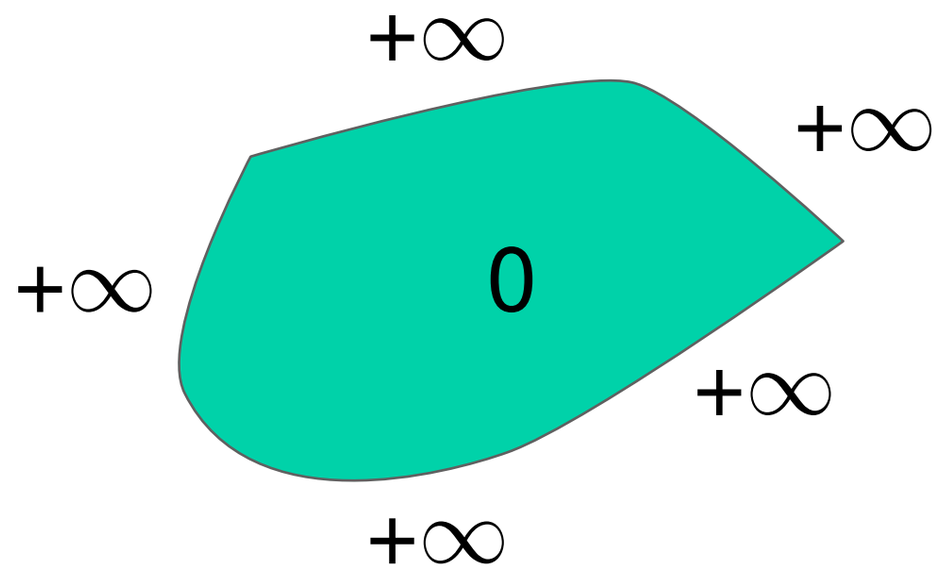




Indicator functions

For a closed convex set $\Omega \subset \mathcal{X}$, its **indicator function** is

$$I_\Omega(x) = \begin{cases} 0 & \text{if } x \in \Omega, \\ +\infty & \text{else.} \end{cases}$$

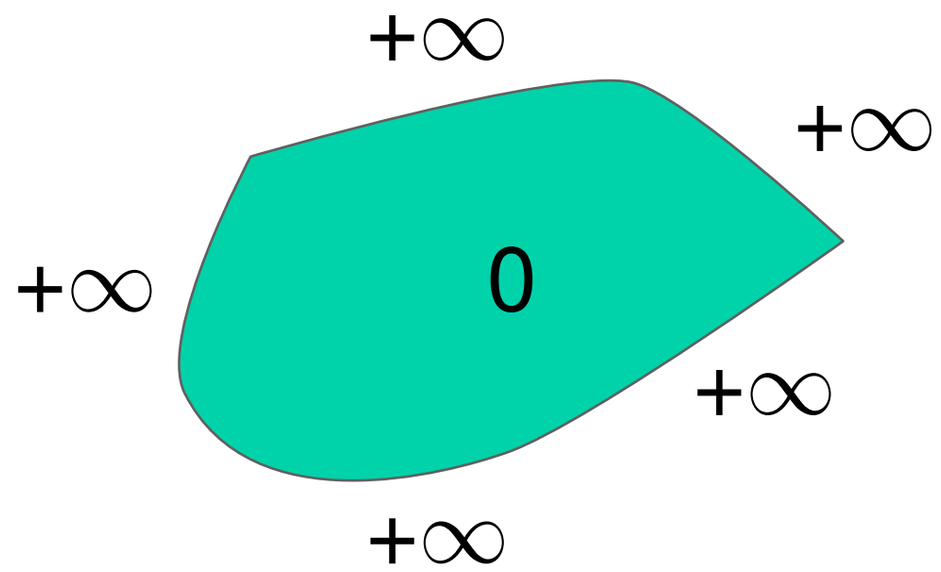


I_Ω is convex.

Indicator functions

For a closed convex set $\Omega \subset \mathcal{X}$, its **indicator function** is

$$I_\Omega(x) = \begin{cases} 0 & \text{if } x \in \Omega, \\ +\infty & \text{else.} \end{cases}$$



I_Ω is convex.

Note: $I_{\Omega_1} + I_{\Omega_2} = I_{\Omega_1 \cap \Omega_2}$

Constrained optimization

$$\text{Find } x^\star \in \underset{x \in \mathcal{X}}{\arg\min}\ \Psi(x) = \sum_{m=1}^{M} g_m(L_m x)$$

encompasses the presence of constraints:

$$\underset{x \in \Omega}{\text{minimize}}\ f(x) \equiv \underset{x \in \mathcal{X}}{\text{minimize}}\ f(x) \ \text{s.t.}\ x \in \Omega$$

$$\equiv \underset{x \in \mathcal{X}}{\text{minimize}}\ f(x) + \imath_\Omega(x)$$

Optimization algorithms

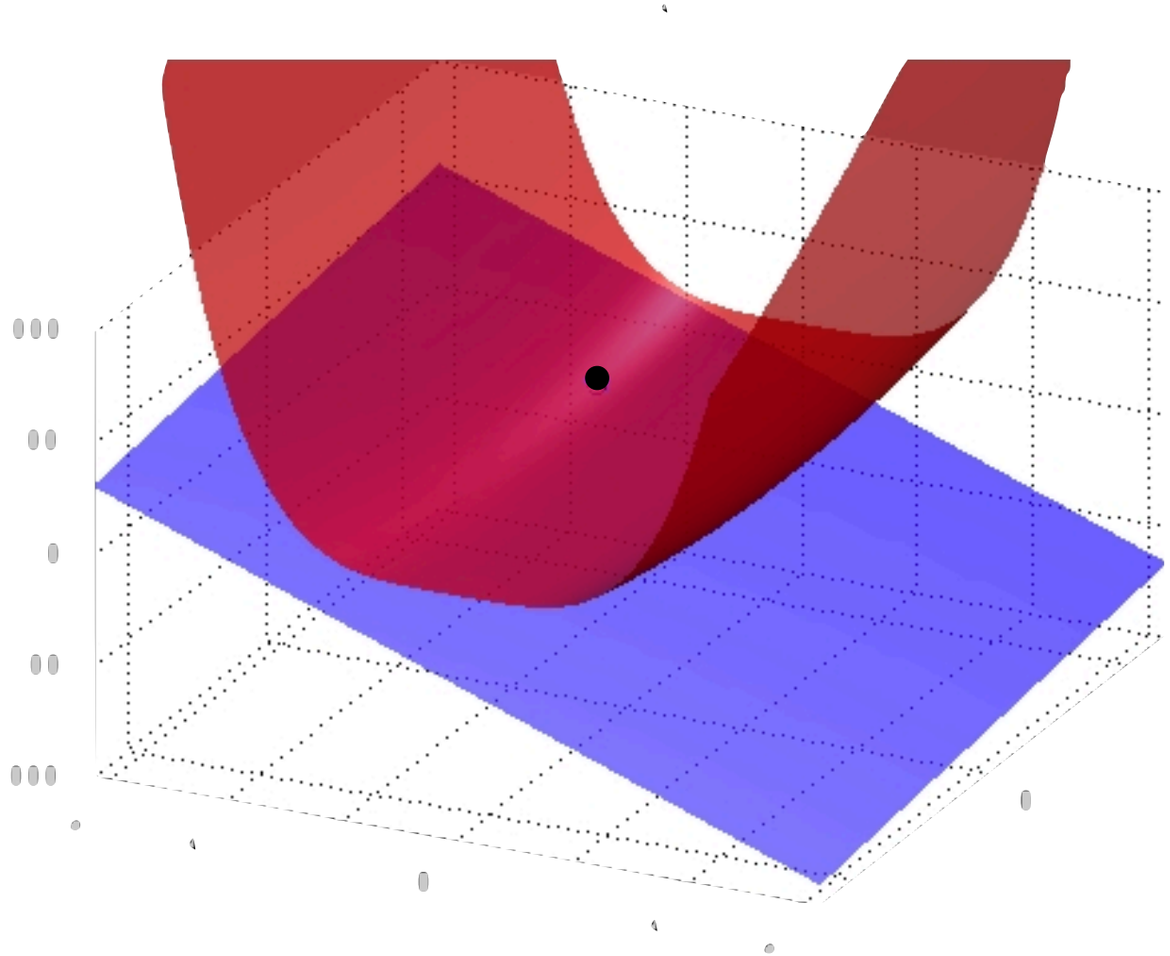
$$\text{Find } x^\star \in \arg\min_{x \in \mathcal{X}} \Psi(x) = \sum_{m=1}^{M} g_m(L_m x)$$

iterative algorithm computing
$x^{k+1} = T(x^k)$
with x^k converging to $x^\star = T(x^\star)$

The gradient

$f : \mathcal{X} \to \mathbb{R}$ is differentiable (=smooth) at x
if there exists a unique element $\nabla f(x) \in \mathcal{X}$ such that

$$\forall e \in \mathcal{X}, f(x + e) = f(x) + \langle e, \nabla f(x) \rangle + o(\|e\|)$$



The gradient

$f : \mathcal{X} \to \mathbb{R}$ is differentiable (=smooth) at x
if there exists a unique element $\nabla f(x) \in \mathcal{X}$ such that

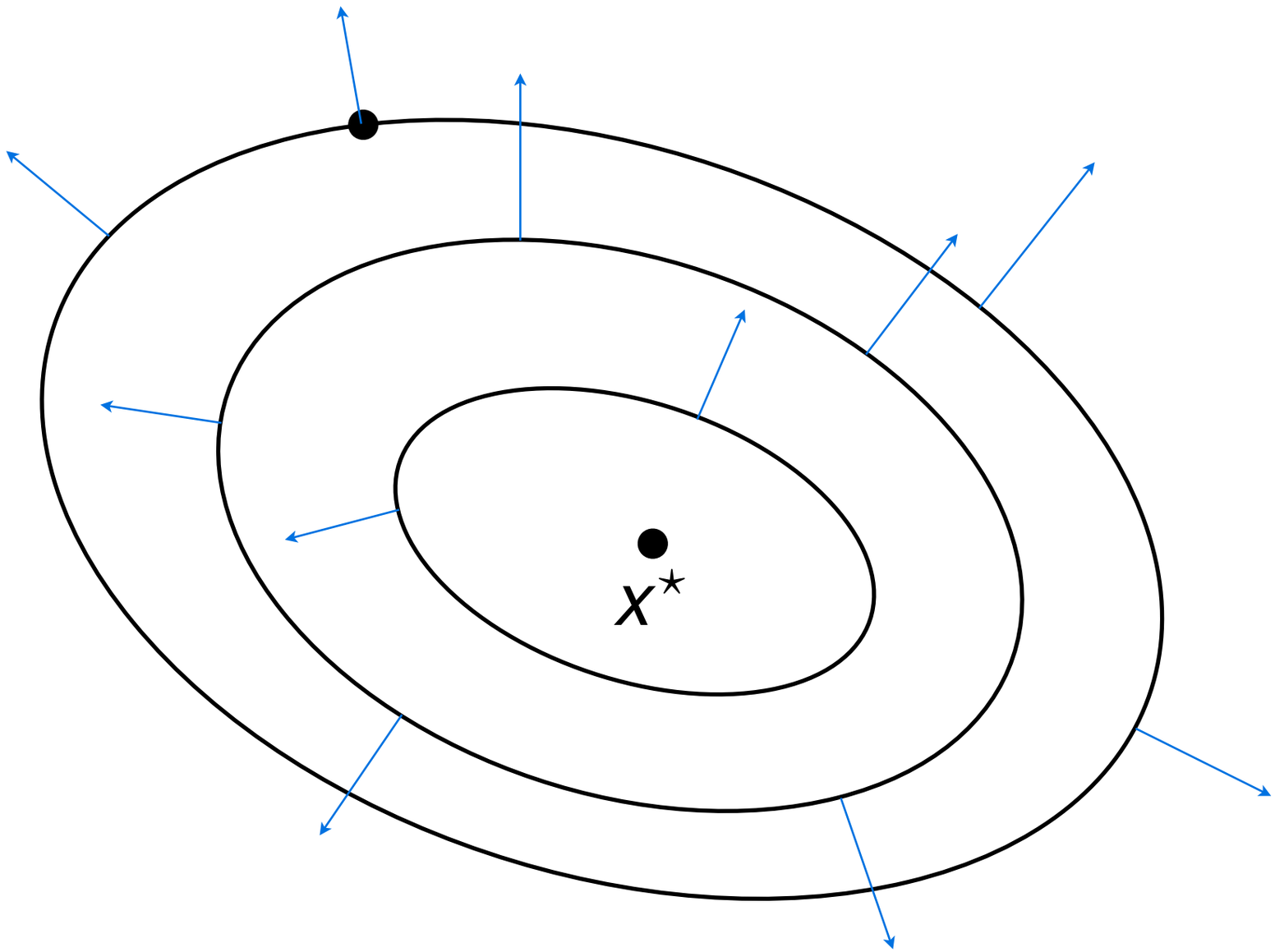
$$\forall e \in \mathcal{X},\ f(x+e) = f(x) + \langle e, \nabla f(x) \rangle + o(\|e\|)$$

$$\Psi(x) = \sum_{m=1}^{M} g_m(L_m x)$$

$$\nabla \Psi(x) = \sum_{m=1}^{M} L_m^* \nabla g_m(L_m x)$$

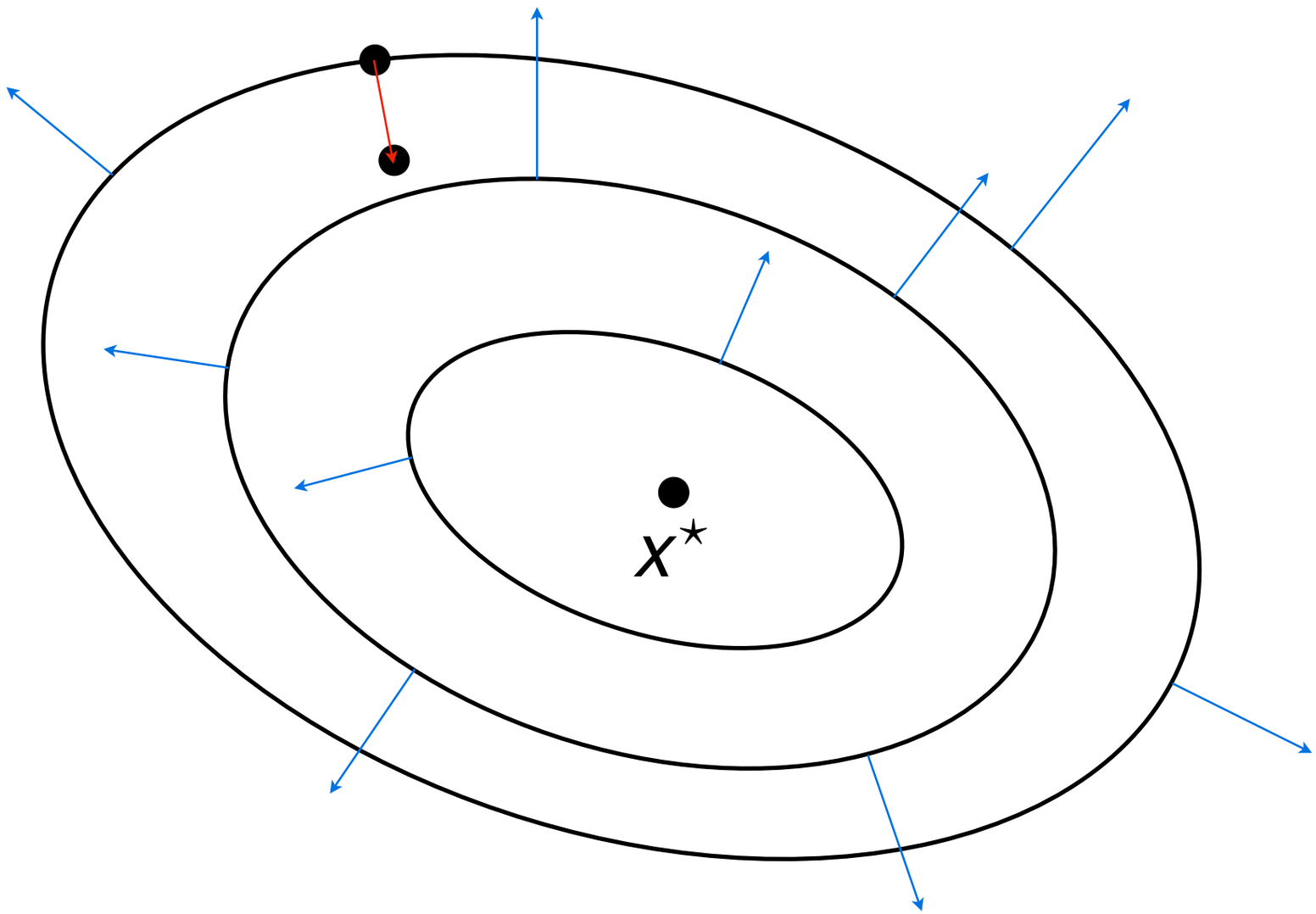
Gradient descent

$$x^{k+1} = x^k - \gamma \nabla \Psi(x^k)$$



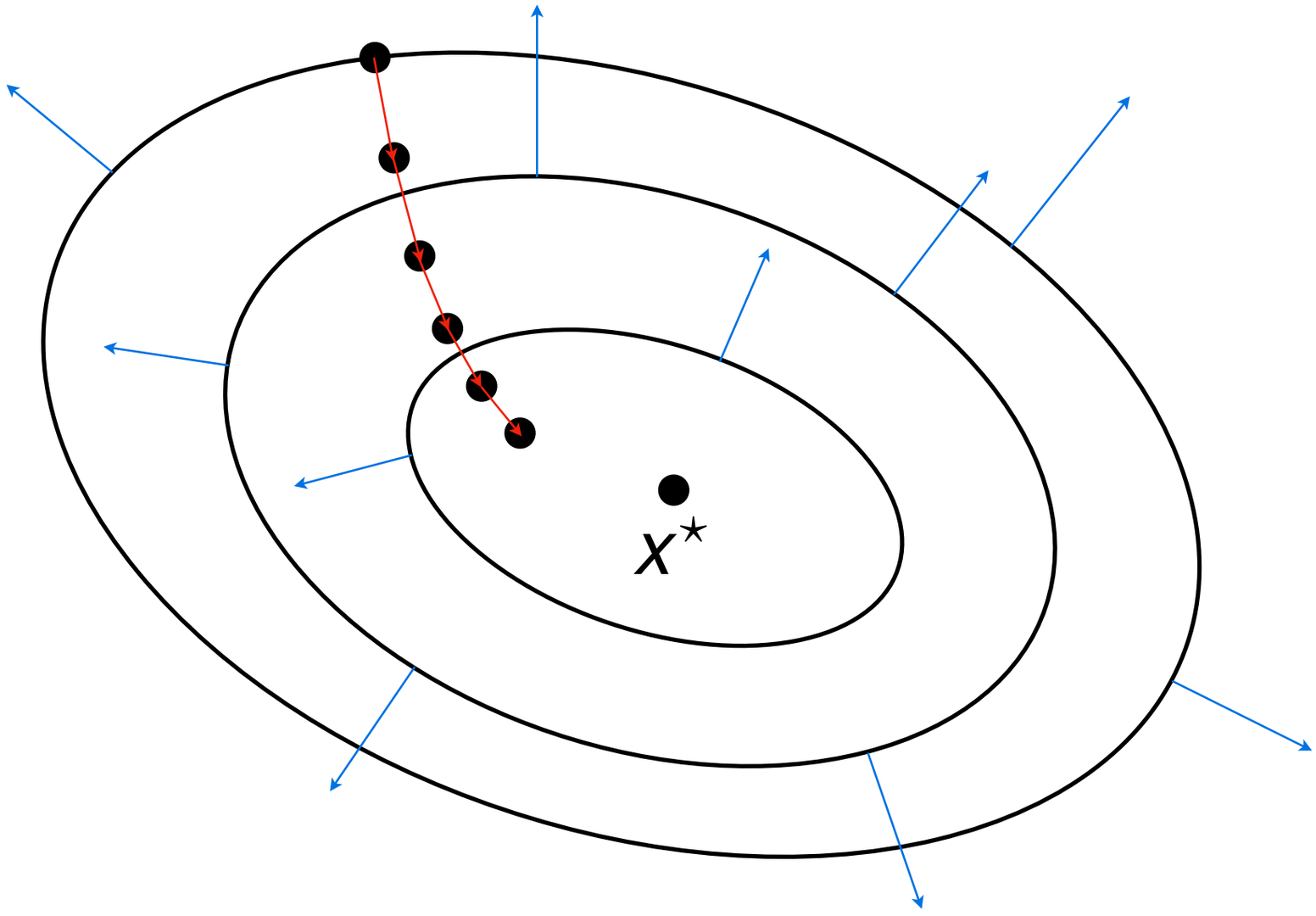
Gradient descent

$$x^{k+1} = x^k - \gamma \nabla \Psi(x^k)$$

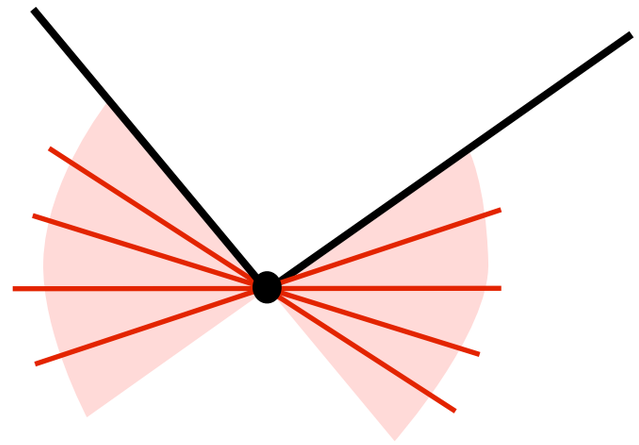


Gradient descent

$$x^{k+1} = x^k - \gamma \nabla \Psi(x^k)$$



The subdifferential

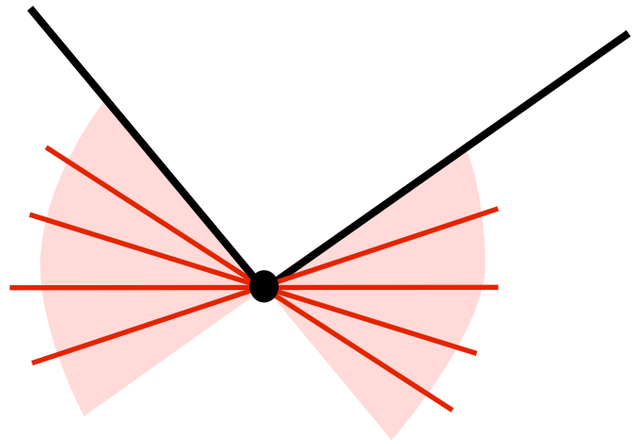


$$\partial f : \mathcal{X} \to 2^{\mathcal{X}}$$

$$x \mapsto \left\{ u \in \mathcal{X} \ : \ \forall y \in \mathcal{X},\ f(x) + \langle y - x, u \rangle \leq f(y) \right\}$$

☞ $\partial f(x)$ is the set of gradients of the affine minorants of f at x

The subdifferential

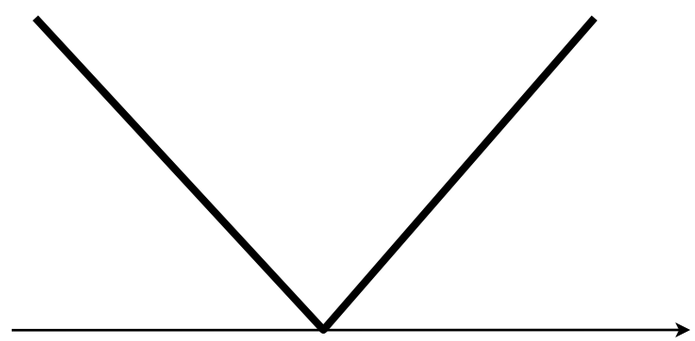


$$\partial f : \mathcal{X} \to 2^{\mathcal{X}}$$

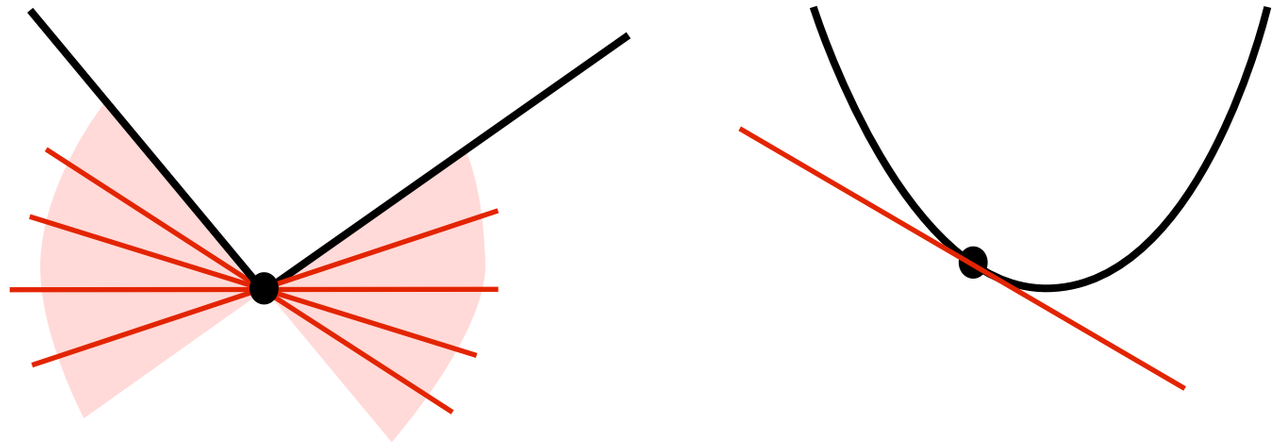
$$x \mapsto \left\{ u \in \mathcal{X} \ : \ \forall y \in \mathcal{X},\ f(x) + \langle y - x, u \rangle \leq f(y) \right\}$$

Example: $f = |\cdot|$

☞ $\partial f(0) = [-1, 1]$



The subdifferential



$$\partial f : \mathcal{X} \to 2^{\mathcal{X}}$$

$$x \mapsto \left\{ u \in \mathcal{X} \ : \ \forall y \in \mathcal{X},\ f(x) + \langle y - x, u \rangle \leq f(y) \right\}$$

f is convex and smooth at $x \to \partial f(x) = \{\nabla f(x)\}$.

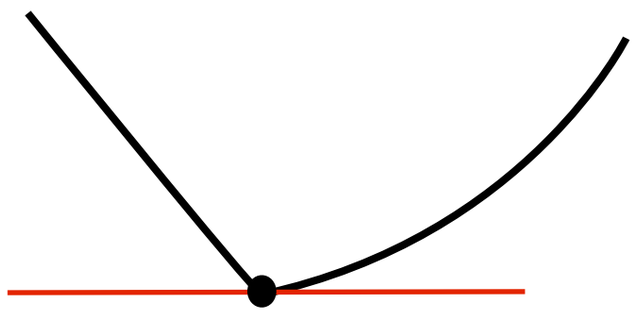
Fermat's rule

$$x^\star \in \arg\min f$$

$$\Leftrightarrow$$

$$0 \in \partial f(x^\star)$$

Pierre de Fermat, 1601-1665



First-order conditions

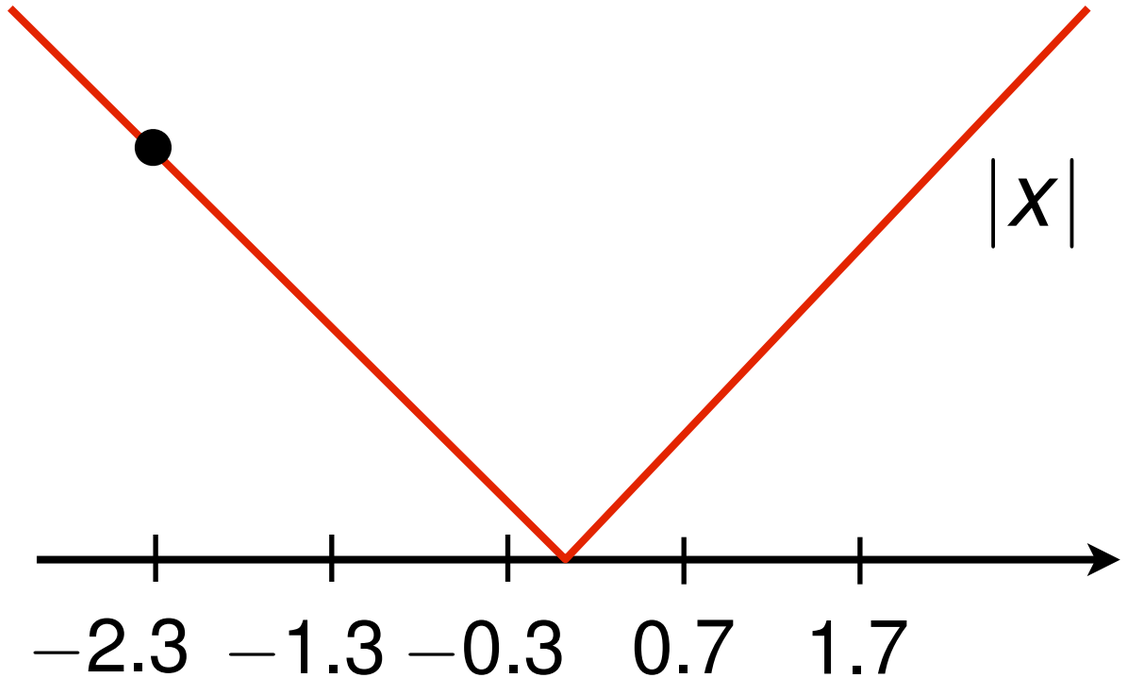
$$\text{Find } x^\star \in \operatorname*{arg\,min}_{x \in \mathcal{X}} \ \Psi(x) = \sum_{m=1}^{M} g_m(L_m x)$$

$$\Uparrow$$

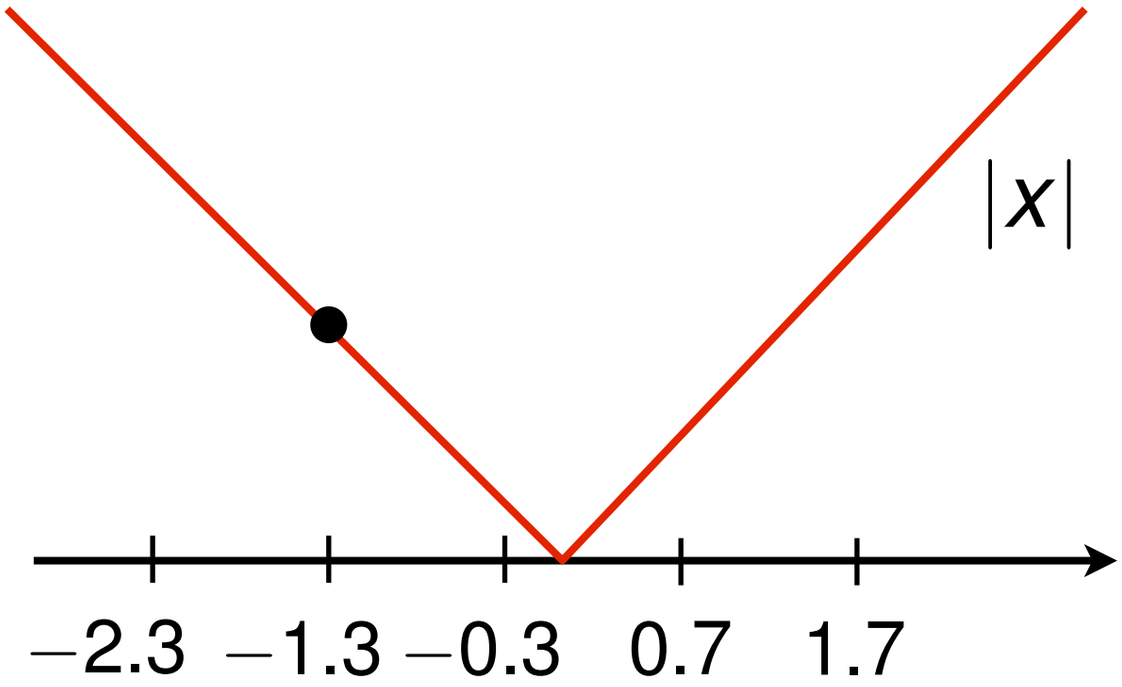
find $x^\star \in \mathcal{X}$ such that

$$0 \in \sum_{m=1}^{M} L_m^* \partial g_m(L_m x^\star)$$

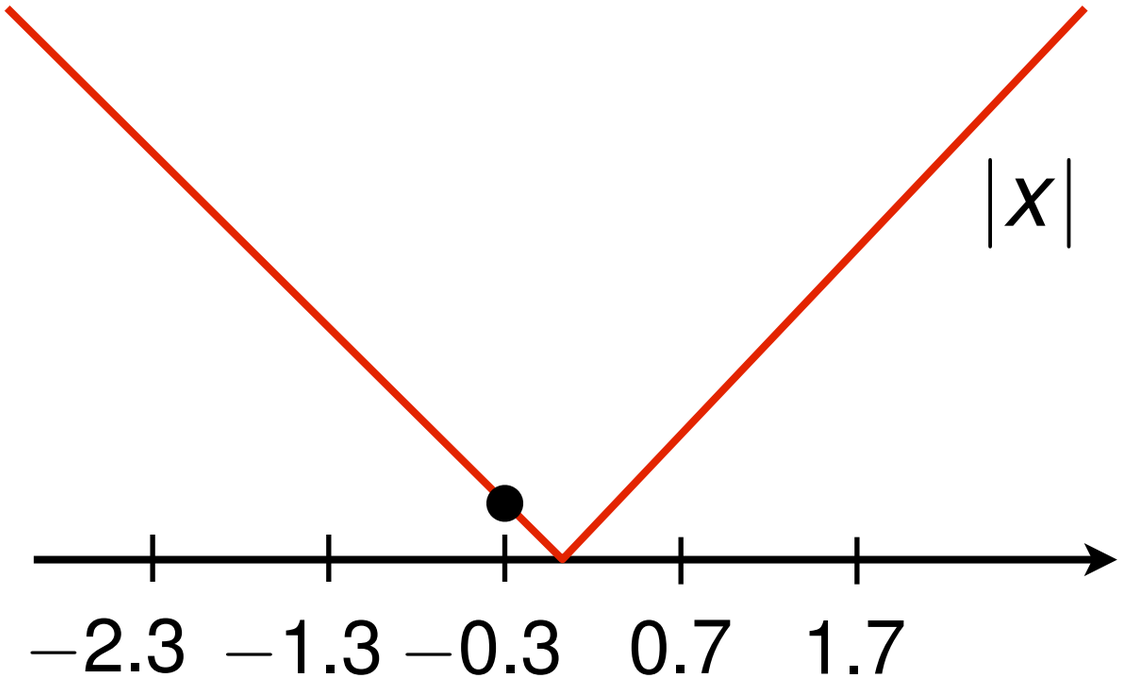
Nonsmooth minimization?



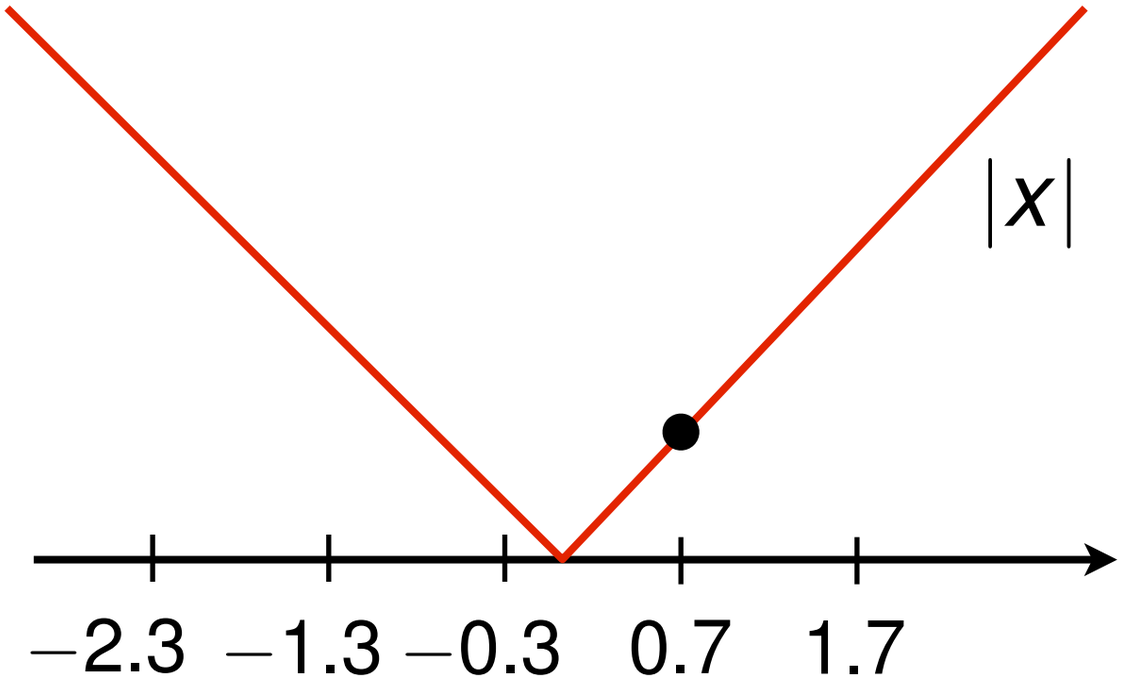
Nonsmooth minimization?



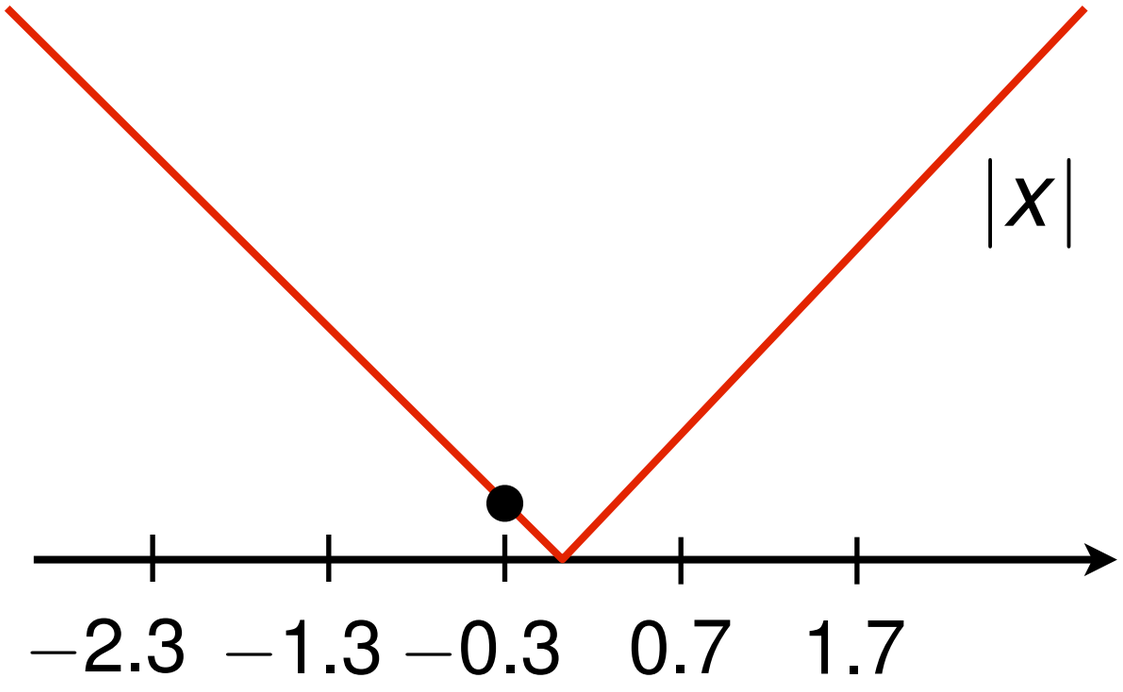
Nonsmooth minimization?



Nonsmooth minimization?

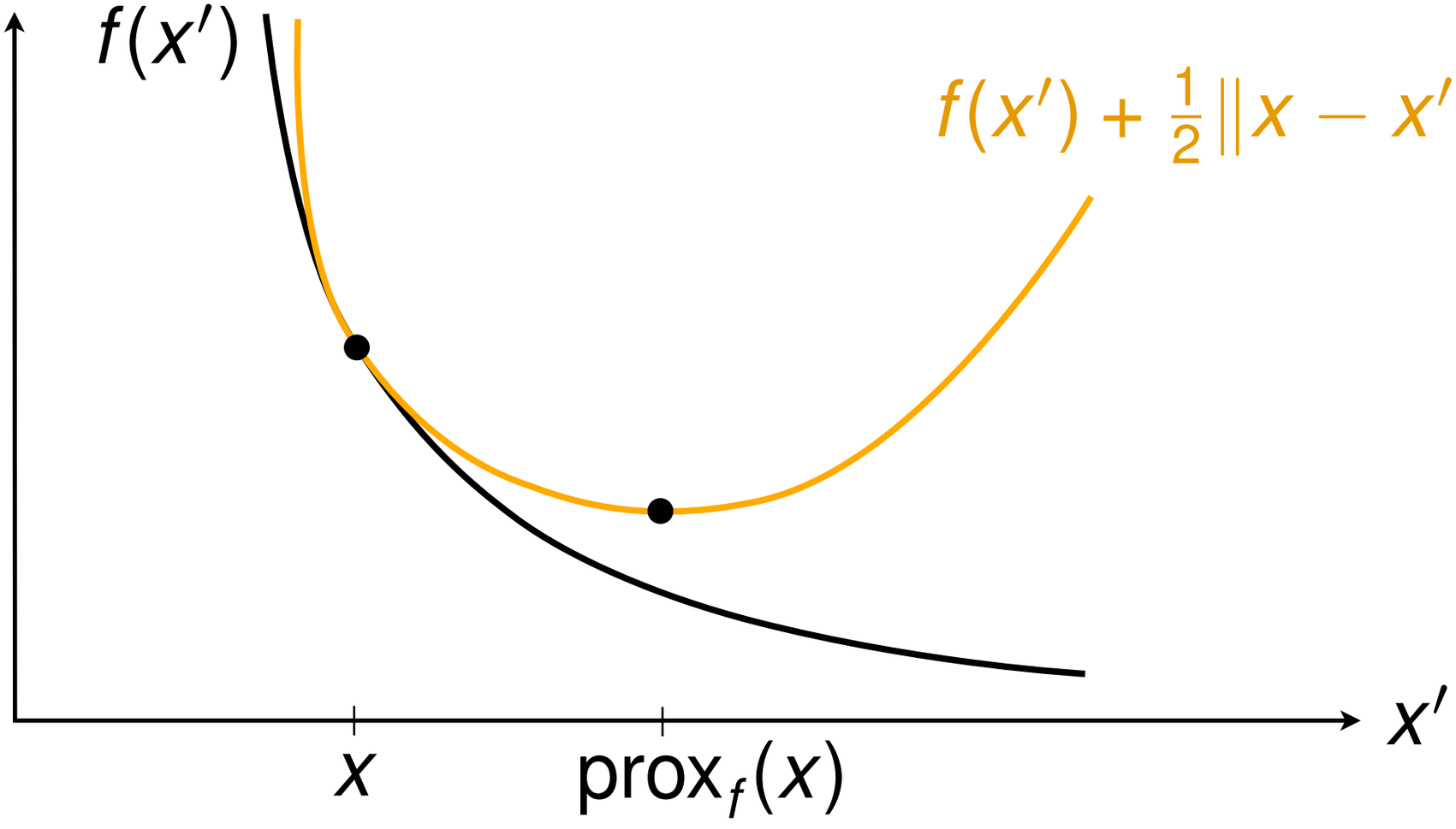


Nonsmooth minimization?



The proximity operator

$$\text{prox}_f : \mathcal{X} \to \mathcal{X} : x \mapsto \underset{x' \in \mathcal{X}}{\arg\min} \left(f(x') + \tfrac{1}{2}\|x - x'\|^2 \right)$$



The proximity operator

$$\text{prox}_f : \mathcal{X} \to \mathcal{X} : x \mapsto \underset{x' \in \mathcal{X}}{\arg\min} \left(f(x') + \tfrac{1}{2} \|x - x'\|^2 \right)$$

$$\text{prox}_f = (\partial f + \text{Id})^{-1}$$

The proximity operator

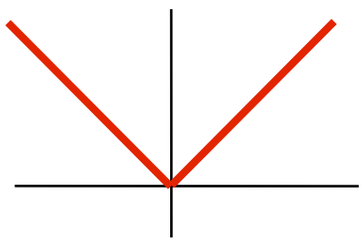
$$\text{prox}_f : \mathcal{X} \to \mathcal{X} : x \mapsto \underset{x' \in \mathcal{X}}{\arg\min} \left(f(x') + \tfrac{1}{2} \|x - x'\|^2 \right)$$

$$\text{prox}_f = (\partial f + \text{Id})^{-1}$$

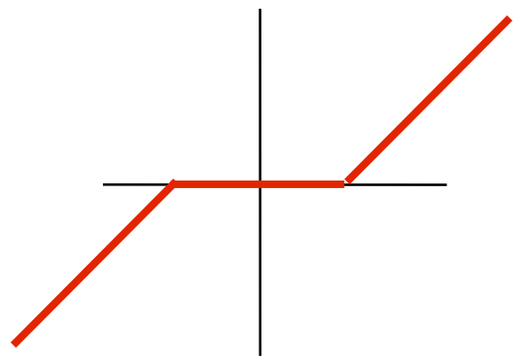
$$y = \text{prox}_f(x) \quad \equiv \quad y \in x - \partial f(y)$$

The proximity operator

$$\text{prox}_f : \mathcal{X} \to \mathcal{X} : x \mapsto \arg\min_{x' \in \mathcal{X}} \left(f(x') + \tfrac{1}{2} \|x - x'\|^2 \right)$$



$f(x) = |x|$

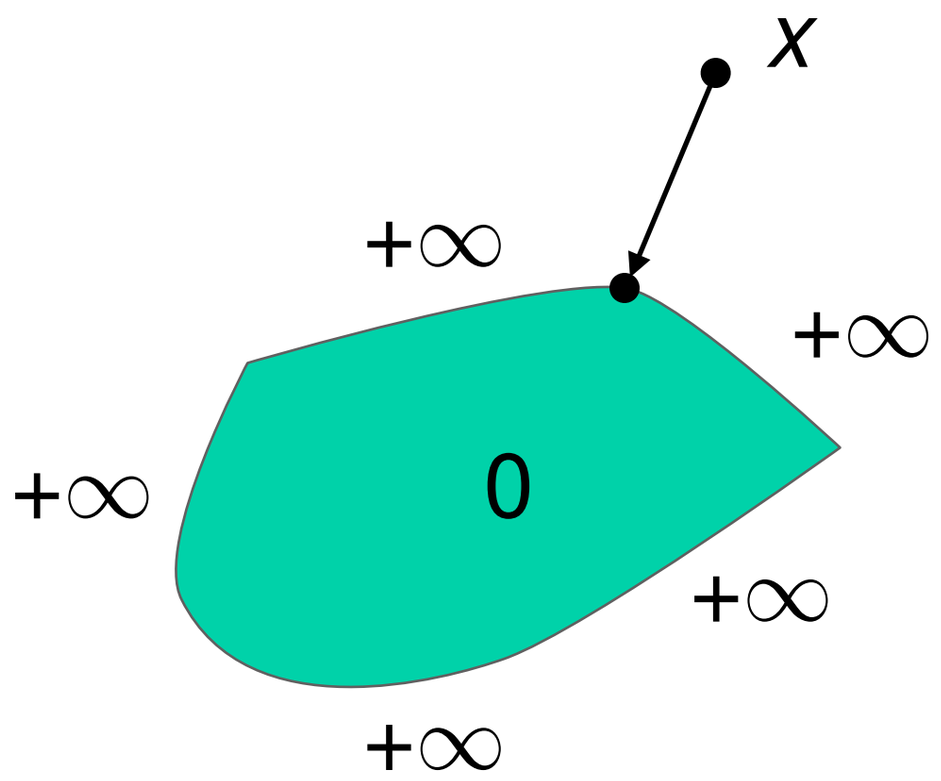


$$\text{prox}_f(x) = \text{sgn}(x) \max(|x| - 1, 0)$$

The proximity operator

$$\text{prox}_f : \mathcal{X} \to \mathcal{X} : x \mapsto \underset{x' \in \mathcal{X}}{\arg\min} \left(f(x') + \tfrac{1}{2} \|x - x'\|^2 \right)$$

$$\text{prox}_{I_\Omega} = \text{proj}_\Omega$$



The proximity operator

Exact, finite time, algorithms are available to compute the proximity operator of:

- $\|X\|_*$ $\rightarrow$ SVD, $O(N^3)$
- 1-D TV $\rightarrow$ taut-string alg., $O(N)$
- 2-D anisotropic TV $\rightarrow$ graph cuts
- proj. onto the simplex $\rightarrow$ $O(N)$

...

The proximity operator

Exact, finite time, algorithms are available to compute the proximity operator of:

- $\|X\|_*$ $\rightarrow$ SVD, $O(N^3)$
- 1-D TV $\rightarrow$ taut-string alg., $O(N)$
- 2-D anisotropic TV $\rightarrow$ graph cuts
- proj. onto the simplex $\rightarrow$ $O(N)$

...

LC, "A direct algorithm for 1-D total variation...", 2013

LC, "Fast projection onto the simplex...", 2016

Pustelnik, LC, "Proximity operator of a sum...", 2017

Proximal splitting algorithms

$$\text{Find } x^\star \in \operatorname*{arg\,min}_{x \in \mathcal{X}} \sum_{m=1}^{M} g_m(L_m x)$$

No easy form of $\text{prox}_{g_1+g_2}$ or $\text{prox}_{g \circ L}$

Proximal splitting algorithms

$$\text{Find } x^\star \in \operatorname*{arg\,min}_{x \in \mathcal{X}} \sum_{m=1}^{M} g_m(L_m x)$$

We want full splitting, with individual activation of L_m, L_m^*, the gradient or proximity operator of g_m.

Proximal splitting algorithms

$$\text{Find } x^\star \in \operatorname*{arg\,min}_{x \in \mathcal{X}} \left(f(x) + \sum_{m=1}^{M} g_m(L_m x) + h(x) \right)$$

with:

- h smooth with β-Lipschitz continuous gradient
  $\rightarrow$ calls to ∇h
- simple functions f and $g_m \rightarrow$ calls to $\text{prox}_{\gamma f}$ and $\text{prox}_{\tau_m g_m}$
- calls to L_m, L_m^*

Product space trick

$$\text{Find } x^\star \in \operatorname*{arg\,min}_{x \in \mathcal{X}} \Big(f(x) + g(Lx) + h(x) \Big)$$

$$\boldsymbol{g}(\boldsymbol{u}) = \sum_{m=1}^{M} g_m(u_m)$$

$$\boldsymbol{L}x = (L_1 x, \dots, L_m x)$$

$$\boldsymbol{g}(\boldsymbol{L}x) = \sum_{m=1}^{M} g_m(L_m x)$$

Minimization of 3 functions

$$\text{Find } x^\star \in \operatorname*{arg\,min}_{x \in \mathcal{X}} \Big(f(x) + g(Lx) + h(x) \Big)$$

$$\text{prox}_{\gamma f}, \quad \text{prox}_{\tau g}, \quad \nabla h, \quad L, \quad L^*$$

Proximal splitting algorithms

minimize $f + g \circ L + h$

1979

$f + h$ ☞ forward-backward

$f + g$ ☞ Douglas-Rachford / ADMM

Proximal splitting algorithms

minimize $f + g \circ L + h$

	$f + h$	☞	forward-backward
1979	$f + g$	☞	Douglas-Rachford / ADMM
2011	$f + g \circ L$	☞	Chambolle-Pock

Proximal splitting algorithms

minimize $f + g \circ L + h$

	$f + h$	☞	forward-backward
1979	$f + g$	☞	Douglas-Rachford / ADMM
2011	$f + g \circ L$	☞	Chambolle-Pock
2011	$g \circ L + h$	☞	PAPC

Proximal splitting algorithms

minimize $f + g \circ L + h$

	$f + h$	☞	forward-backward
1979	$f + g$	☞	Douglas-Rachford / ADMM
2011	$f + g \circ L$	☞	Chambolle-Pock
2011	$g \circ L + h$	☞	PAPC
2013	$f + g \circ L + h$	☞	Condat, Vu

LC, "A primal-dual splitting method for convex optimization...", 2013

Vu, "A splitting algorithm for dual monotone inclusions...", 2013

Proximal splitting algorithms

minimize $f + g \circ L + h$

	$f + h$	☞	forward-backward
1979	$f + g$	☞	Douglas-Rachford / ADMM
2011	$f + g \circ L$	☞	Chambolle-Pock
2011	$g \circ L + h$	☞	PAPC
2013	$f + g \circ L + h$	☞	Condat, Vu
2017	$f + g + h$	☞	Davis-Yin

Proximal splitting algorithms

minimize $f + g \circ L + h$

	$f + h$	☞	forward-backward
1979	$f + g$	☞	Douglas-Rachford / ADMM
2011	$f + g \circ L$	☞	Chambolle-Pock
2011	$g \circ L + h$	☞	PAPC
2013	$f + g \circ L + h$	☞	Condat, Vu
2017	$f + g + h$	☞	Davis-Yin
2018	$f + g \circ L + h$	☞	PD3O

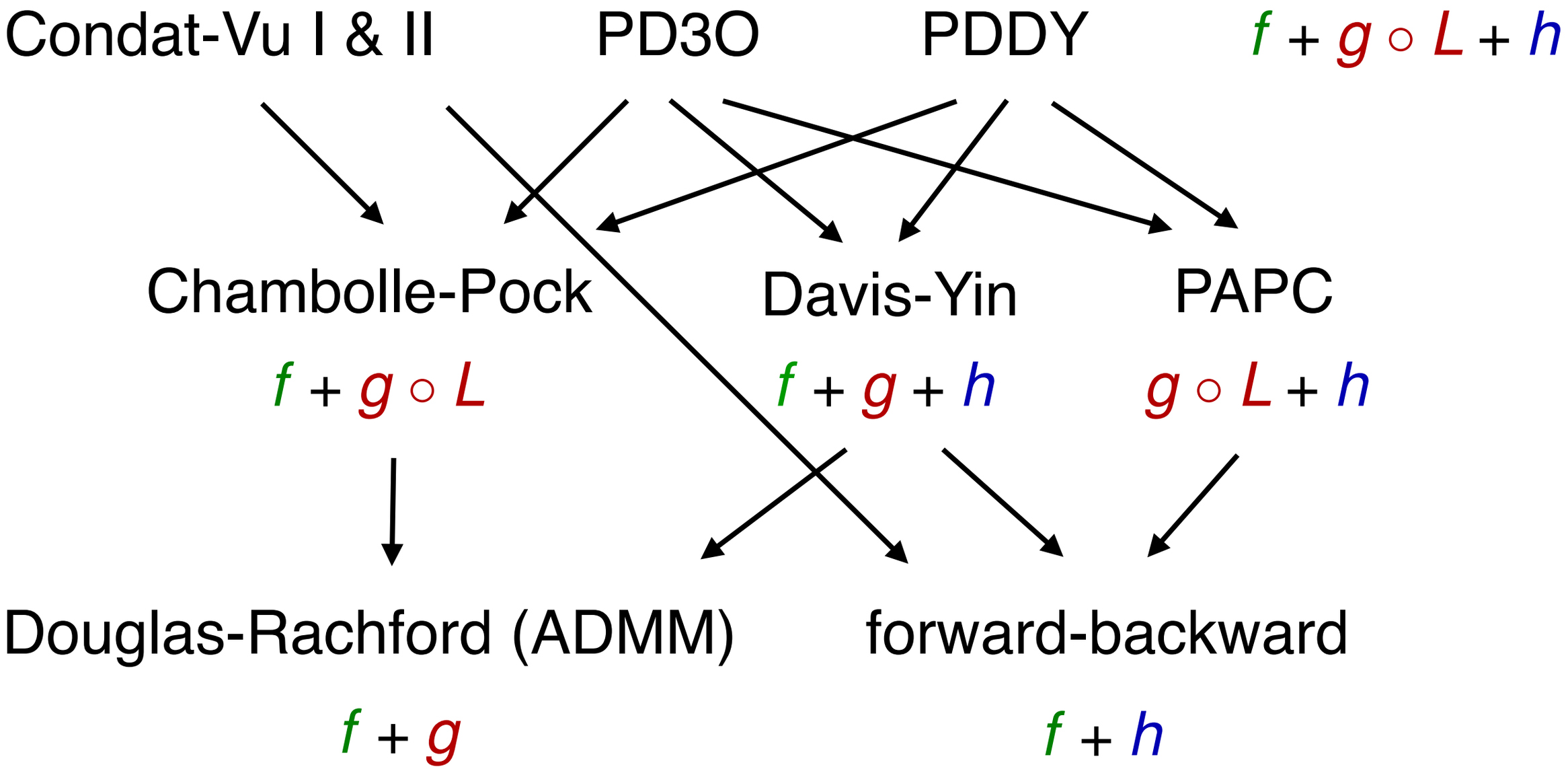
Proximal splitting algorithms

minimize $f + g \circ L + h$

	$f + h$	☞	forward-backward
1979	$f + g$	☞	Douglas-Rachford / ADMM
2011	$f + g \circ L$	☞	Chambolle-Pock
2011	$g \circ L + h$	☞	PAPC
2013	$f + g \circ L + h$	☞	Condat, Vu
2017	$f + g + h$	☞	Davis-Yin
2018	$f + g \circ L + h$	☞	PD3O
2022	$f + g \circ L + h$	☞	PDDY

Salim, LC et al., "Dualize, split, randomize...", 2022

Proximal splitting algorithms



4 primal-dual algorithms

Condat–Vu algorithm form I

minimize $f + g \circ L + h$

$$\begin{cases} x^{k+1} = \operatorname{prox}_{\gamma f}\left(x^k - \gamma \nabla h(x^k) - \gamma L^* u^k\right) \\ u^{k+1} = \operatorname{prox}_{\tau g^*}\left(u^k + \tau L(2x^{k+1} - x^k)\right) \end{cases}$$

Condat–Vu algorithm form II

$$\begin{cases} u^{k+1} = \operatorname{prox}_{\tau g^*}\left(u^k + \tau L x^k\right) \\ x^{k+1} = \operatorname{prox}_{\gamma f}\left(x^k - \gamma \nabla h(x^k) - \gamma L^*(2u^{k+1} - u^k)\right) \end{cases}$$

PD3O algorithm

$$\begin{cases} x^{k+1} = \operatorname{prox}_{\gamma f}\left(x^k - \gamma \nabla h(x^k) - \gamma L^* u^k\right) \\ u^{k+1} = \operatorname{prox}_{\tau g^*}\left(u^k + \tau L(2x^{k+1} - x^k - \gamma \nabla h(x^{k+1}) + \gamma \nabla h(x^k))\right) \end{cases}$$

PDDY algorithm

$$\begin{cases} u^{k+1} = \operatorname{prox}_{\tau g^*}\left(u^k + \tau L x^k\right) \\ x^{k+1} = \operatorname{prox}_{\gamma f}\left(x^k - \gamma \nabla h(x^k - \gamma L^*(u^{k+1} - u^k)) - \gamma L^*(2u^{k+1} - u^k)\right) \end{cases}$$

First-order conditions

$$\text{Find } x^\star \in \operatorname*{arg\,min}_{x \in \mathcal{X}} \Big(f(x) + g(Lx) + h(x) \Big)$$

$$\sim\equiv$$

$$0 \in \nabla h(x^\star) + \partial f(x^\star) + L^* \partial g(Lx^\star)$$

First-order conditions

$$\text{Find } x^\star \in \operatorname*{arg\,min}_{x \in \mathcal{X}} \Big(f(x) + g(Lx) + h(x) \Big)$$

$$\sim\equiv$$

$$0 \in \nabla h(x^\star) + \partial f(x^\star) + L^* \partial g(Lx^\star)$$

$$\equiv$$

$$\begin{cases} 0 \in \nabla h(x^\star) + \partial f(x^\star) + L^* u^\star \\ u^\star \in \partial g(Lx^\star) \end{cases}$$

First-order conditions

$$\text{Find } x^\star \in \operatorname*{arg\,min}_{x \in \mathcal{X}} \Big(f(x) + g(Lx) + h(x) \Big)$$

$$\sim\equiv$$

$$0 \in \nabla h(x^\star) + \partial f(x^\star) + L^* \partial g(Lx^\star)$$

$$\equiv$$

$$\begin{cases} 0 \in \nabla h(x^\star) + \partial f(x^\star) + L^* u^\star \\ 0 \in -Lx^\star + \partial g^*(u^\star) \end{cases}$$

$$(\partial g^* = (\partial g)^{-1})$$

First-order conditions

$$\text{Find } x^\star \in \operatorname*{arg\,min}_{x \in \mathcal{X}} \Big(f(x) + g(Lx) + h(x) \Big)$$

$$\sim\equiv$$

Find $(x^\star, u^\star)$ solution to

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \begin{pmatrix} \nabla h(x) + \partial f(x) + L^* u \\ -Lx + \partial g^*(u) \end{pmatrix}$$

First-order conditions

$$\text{Find } x^\star \in \operatorname*{arg\,min}_{x \in \mathcal{X}} \Big(f(x) + g(Lx) + h(x) \Big)$$

$$\sim\equiv$$

Find $(x^\star, u^\star)$ solution to

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \begin{pmatrix} \nabla h(x) + \partial f(x) + L^* u \\ -Lx + \partial g^*(u) \end{pmatrix}$$

Dual problem:

$$\text{Find } u^\star \in \operatorname*{arg\,min}_{u \in \mathcal{U}} \Big((f + h)^*(-L^* u) + g^*(u) \Big)$$

First-order conditions

$$\text{Find } x^\star \in \operatorname*{arg\,min}_{x \in \mathcal{X}} \Big(f(x) + g(Lx) + h(x) \Big)$$

$$\sim\equiv$$

Find $(x^\star, u^\star)$ solution to

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \begin{pmatrix} \nabla h(x) + \partial f(x) + L^* u \\ -Lx + \partial g^*(u) \end{pmatrix}$$

$$\uparrow$$

monotone

M monotone: $\forall (x, x') \in \mathcal{X}^2$, $v \in Mx$, $v' \in Mx'$, $\langle x - x', v - v' \rangle \geq 0$

Primal-dual 3 operator splitting

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \begin{pmatrix} L^* u \\ -Lx + \partial g^*(u) \end{pmatrix} + \begin{pmatrix} \partial f(x) \\ 0 \end{pmatrix} + \begin{pmatrix} \nabla h(x) \\ 0 \end{pmatrix}$$

Primal-dual 3 operator splitting

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \begin{pmatrix} L^* u \\ -Lx + \partial g^*(u) \end{pmatrix} + \begin{pmatrix} \partial f(x) \\ 0 \end{pmatrix} + \begin{pmatrix} \nabla h(x) \\ 0 \end{pmatrix}$$

or

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \begin{pmatrix} \partial f(x) + L^* u \\ -Lx \end{pmatrix} + \begin{pmatrix} 0 \\ \partial g^*(u) \end{pmatrix} + \begin{pmatrix} \nabla h(x) \\ 0 \end{pmatrix}$$

Primal-dual 3 operator splitting

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \begin{pmatrix} L^* u \\ -Lx + \partial g^*(u) \end{pmatrix} + \begin{pmatrix} \partial f(x) \\ 0 \end{pmatrix} + \begin{pmatrix} \nabla h(x) \\ 0 \end{pmatrix}$$

or

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \begin{pmatrix} \partial f(x) + L^* u \\ -Lx \end{pmatrix} + \begin{pmatrix} 0 \\ \partial g^*(u) \end{pmatrix} + \begin{pmatrix} \nabla h(x) \\ 0 \end{pmatrix}$$

monotone monotone cocoercive

C is ξ-cocoercive: $\forall (x, x') \in \mathcal{X}^2, \xi \|Cx - Cx'\|^2 \leq \langle x - x', Cx - Cx' \rangle$

Primal-dual 3 operator splitting

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \begin{pmatrix} L^* u \\ -Lx + \partial g^*(u) \end{pmatrix} + \begin{pmatrix} \partial f(x) \\ 0 \end{pmatrix} + \begin{pmatrix} \nabla h(x) \\ 0 \end{pmatrix}$$

or

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \begin{pmatrix} \partial f(x) + L^* u \\ -Lx \end{pmatrix} + \begin{pmatrix} 0 \\ \partial g^*(u) \end{pmatrix} + \begin{pmatrix} \nabla h(x) \\ 0 \end{pmatrix}$$

☞ $0 \in A(z^\star) + B(z^\star) + C(z^\star)$

Primal-dual 3 operator splitting

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \begin{pmatrix} L^* u \\ -Lx + \partial g^*(u) \end{pmatrix} + \begin{pmatrix} \partial f(x) \\ 0 \end{pmatrix} + \begin{pmatrix} \nabla h(x) \\ 0 \end{pmatrix}$$

or

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \begin{pmatrix} \partial f(x) + L^* u \\ -Lx \end{pmatrix} + \begin{pmatrix} 0 \\ \partial g^*(u) \end{pmatrix} + \begin{pmatrix} \nabla h(x) \\ 0 \end{pmatrix}$$

☞ $0 \in P^{-1} A(z^\star) + P^{-1} B(z^\star) + P^{-1} C(z^\star)$

for some $P \succ 0$

(M monotone $\Rightarrow$ $P^{-1} M$ monotone w.r.t. $\langle \cdot, P \cdot \rangle$)

Primal-dual 3 operator splitting

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \begin{pmatrix} L^* u \\ -Lx + \partial g^*(u) \end{pmatrix} + \begin{pmatrix} \partial f(x) \\ 0 \end{pmatrix} + \begin{pmatrix} \nabla h(x) \\ 0 \end{pmatrix}$$

or

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \begin{pmatrix} \partial f(x) + L^* u \\ -Lx \end{pmatrix} + \begin{pmatrix} 0 \\ \partial g^*(u) \end{pmatrix} + \begin{pmatrix} \nabla h(x) \\ 0 \end{pmatrix}$$

$$0 \in P^{-1} A(z^\star) + P^{-1} B(z^\star) + P^{-1} C(z^\star)$$

☞ **Davis-Yin splitting algorithm**

$$\left\lfloor \begin{array}{l} z^k = J_{P^{-1}B}(v^k) \\ w^{k+1} = J_{P^{-1}A}\big(2z^k - v^k - P^{-1}C(z^k)\big) \\ v^{k+1} = v^k + w^{k+1} - z^k \end{array} \right. \qquad (J_M = (M + \mathrm{Id})^{-1})$$

PD3O algorithm

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \begin{pmatrix} L^* u \\ -Lx + \partial g^*(u) \end{pmatrix} + \begin{pmatrix} \partial f(x) \\ 0 \end{pmatrix} + \begin{pmatrix} \nabla h(x) \\ 0 \end{pmatrix}$$

☞ Davis-Yin splitting with $P = \begin{pmatrix} \frac{1}{\gamma}\mathrm{Id} & 0 \\ 0 & \frac{1}{\tau}\mathrm{Id} - \gamma L L^* \end{pmatrix}$

$$\equiv$$

PD3O algorithm

$$\left\lfloor \begin{aligned} x^{k+1} &= \mathrm{prox}_{\gamma f}(s^k) \\ u^{k+1} &= \mathrm{prox}_{\tau g^*}\Big(u^k + \tau L\big(2x^{k+1} - s^k - \gamma \nabla h(x^{k+1}) - \gamma L^* u^k\big)\Big) \\ s^{k+1} &= x^{k+1} - \gamma \nabla h(x^{k+1}) - \gamma L^* u^{k+1} \end{aligned} \right.$$

PDDY algorithm

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \begin{pmatrix} L^* u \\ -Lx + \partial g^*(u) \end{pmatrix} + \begin{pmatrix} \partial f(x) \\ 0 \end{pmatrix} + \begin{pmatrix} \nabla h(x) \\ 0 \end{pmatrix}$$

☞ Davis-Yin splitting with $P = \begin{pmatrix} \frac{1}{\gamma}\mathrm{Id} & 0 \\ 0 & \frac{1}{\tau}\mathrm{Id} - \gamma L L^* \end{pmatrix}$ and swapping A and B

$$\equiv$$

PDDY algorithm

$$\left\lfloor \begin{aligned} \hat{x}^k &= \mathrm{prox}_{\gamma f}\left(x^k - \gamma \nabla h(x^k) + \gamma h^k\right) \\ u^{k+1} &= \mathrm{prox}_{\tau g^*}\left(u^k + \tau L \hat{x}^k\right) \\ h^{k+1} &= -L^* u^{k+1} \\ x^{k+1} &= \hat{x}^k + \gamma(h^{k+1} - h^k) \end{aligned} \right.$$

Condat-Vu algorithm

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \begin{pmatrix} \partial f(x) + L^* u \\ -Lx \end{pmatrix} + \begin{pmatrix} 0 \\ \partial g^*(u) \end{pmatrix} + \begin{pmatrix} \nabla h(x) \\ 0 \end{pmatrix}$$

☞ Davis-Yin splitting with $P = \begin{pmatrix} \frac{1}{\gamma}\mathrm{Id} - \tau L^* L & 0 \\ 0 & \frac{1}{\tau}\mathrm{Id} \end{pmatrix}$

$$\equiv$$

Condat-Vu algorithm

$$\left\lfloor \begin{array}{l} x^{k+1} = \mathrm{prox}_{\gamma f}\left(x^k - \gamma \nabla h(x^k) - \gamma L^* u^k\right) \\ u^{k+1} = \mathrm{prox}_{\tau g^*}\left(u^k + \tau L x (2x^{k+1} - x^k)\right) \end{array} \right.$$

Condat-Vu algorithm

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \begin{pmatrix} \partial f(x) + L^* u \\ -Lx \end{pmatrix} + \begin{pmatrix} 0 \\ \partial g^*(u) \end{pmatrix} + \begin{pmatrix} \nabla h(x) \\ 0 \end{pmatrix}$$

Davis-Yin splitting with $P = \begin{pmatrix} \frac{1}{\gamma}\mathrm{Id} - \tau L^* L & 0 \\ 0 & \frac{1}{\tau}\mathrm{Id} \end{pmatrix}$ and swapping A and B

$$\equiv$$

Condat-Vu algorithm form II

$$\left\lfloor \begin{array}{l} u^{k+1} = \mathrm{prox}_{\tau g^*}\left(u^k + \tau L x^k\right) \\ x^{k+1} = \mathrm{prox}_{\gamma f}\left(x^k - \gamma \nabla h(x^k) - \gamma L^*(2u^{k+1} - u^k)\right) \end{array} \right.$$

Summary

nonsmooth functions

large scale

proximal splitting algorithms designed using monotone and fixed-point operator theory

Summary

nonsmooth functions

large scale

proximal splitting algorithms
designed using monotone and fixed-point operator theory

LC et al. "Proximal Splitting Algorithms for Convex Optimization: A Tour of Recent Advances, with New Twists", *SIAM Review*, 2023

Salim, LC, Mishchenko, Richtárik, "Dualize, split, randomize: Toward Fast Nonsmooth Optimization Algorithms", *JOTA*, 2022

Convergence rates

Theorem – PD3O, with g continuous around $Lx^\star$:

$$\Psi(x^k) - \Psi(x^\star) = o(1/\sqrt{k})$$

Theorem – accelerated PD3O and PDDY
when h or f strongly convex, with varying stepsizes:

$$\|x^k - x^\star\|^2 = O\left(1/k^2\right)$$

Theorem – linear convergence of PD3O and PDDY
when h or f strongly convex and g smooth:

$$\|x^k - x^\star\|^2 \leq (1-\rho)^k c_0$$

LC, Malinovsky, Richtárik, "Distributed Proximal Splitting Algorithms with Rates and Acceleration", 2022

PDDY algorithm

PDDY

input: initial points $x^0 \in \mathcal{X}$, $u^0 \in \mathcal{U}$;
stepsizes $\gamma > 0$, $\tau > 0$
$h^0 := -L^* u^0$
for $k = 0, 1, \ldots$ **do**
$\hat{x}^k := \operatorname{prox}_{\gamma f}\left(x^k - \gamma \nabla h(x^k) + \gamma h^k\right)$
$u^{k+1} := \operatorname{prox}_{\tau g^*}\left(u^k + \tau L \hat{x}^k\right)$
$h^{k+1} := -L^* u^{k+1}$
$x^{k+1} := \hat{x}^t + \gamma(h^{k+1} - h^k)$
end for

PDDY algorithm

PDDY

input: initial points $x^0 \in \mathcal{X}$, $u^0 \in \mathcal{U}$;
stepsizes $\gamma > 0$, $\tau > 0$
$h^0 := -L^* u^0$
for $k = 0, 1, \ldots$ **do**
$\hat{x}^k := \operatorname{prox}_{\gamma f}\left(x^k - \gamma \nabla h(x^k) + \gamma h^k\right)$
$u^{k+1} := \operatorname{prox}_{\tau g^*}\left(u^k + \tau L \hat{x}^k\right)$
$h^{k+1} := -L^* u^{k+1}$
$x^{k+1} := \hat{x}^t + \gamma(h^{k+1} - h^k)$
end for

$\operatorname{prox}_{\tau g^*}$ can be costly

RandProx algorithm

RandProx

input: initial points $x^0 \in \mathcal{X}$, $u^0 \in \mathcal{U}$;
stepsizes $\gamma > 0$, $\tau > 0$; $\omega \geq 0$
$h^0 := -L^* u^0$
for $k = 0, 1, \ldots$ **do**
$\hat{x}^k := \text{prox}_{\gamma f}\left(x^k - \gamma \nabla h(x^k) + \gamma h^k\right)$
$u^{k+1} := u^k + \frac{1}{1+\omega} \mathcal{R}^k \left(\text{prox}_{\tau g^*}(u^k + \tau L \hat{x}^k) - u^k\right)$
$h^{k+1} := -L^* u^{k+1}$
$x^{k+1} := \hat{x}^k + \gamma(1+\omega)(h^{k+1} - h^k)$
end for

LC, Richtárik, "RandProx: Primal-Dual Optimization Algorithms with Randomized Proximal Updates", ICLR, 2023

RandProx algorithm

RandProx

input: initial points $x^0 \in \mathcal{X}$, $u^0 \in \mathcal{U}$;
stepsizes $\gamma > 0$, $\tau > 0$; $\omega \geq 0$
$h^0 := -L^* u^0$
for $k = 0, 1, \ldots$ **do**
$\hat{x}^k := \text{prox}_{\gamma f}\left(x^k - \gamma \nabla h(x^k) + \gamma h^k\right)$
$u^{k+1} := u^k + \frac{1}{1+\omega} \mathcal{R}^k\left(\text{prox}_{\tau g^*}(u^k + \tau L \hat{x}^k) - u^k\right)$
$h^{k+1} := -L^* u^{k+1}$
$x^{k+1} := \hat{x}^k + \gamma(1+\omega)(h^{k+1} - h^k)$
end for

$$\mathbb{E}\left[\mathcal{R}(r^k)\right] = r^k \quad \text{and} \quad \mathbb{E}\left[\left\|\mathcal{R}(r^k) - r^k\right\|^2\right] \leq \omega \left\|r^k\right\|^2$$

Linear convergence

Theorem 1. Suppose that $\mu_h > 0$ *or* $\mu_f > 0$, *and* $\mu_{g^*} > 0$. *For suitable* γ *and* τ, $\forall k \geq 0$,

$$\mathbb{E}\left[\Psi^k\right] \leq c^k \Psi^0, \text{ where}$$

$$\Psi^k = \frac{1}{\gamma}\left\|x^k - x^\star\right\|^2 + (1+\omega)\left(\frac{1}{\tau} + 2\mu_{g^*}\right)\left\|u^k - u^\star\right\|^2,$$

$$c \coloneqq \max\left(\frac{(1-\gamma\mu_h)^2}{1+\gamma\mu_f}, 1 - \frac{2\tau\mu_{g^*}}{(1+\omega)(1+2\tau\mu_{g^*})}\right)$$

Examples

RandProx-skip

input: initial points $x^0 \in \mathcal{X}$, $u^0 \in \mathcal{U}$;
stepsizes $\gamma > 0$, $\tau > 0$; $p \in (0, 1]$
$h^0 := -L^* u^0$
for $k = 0, 1, \ldots$ **do**
  Flip a coin θ^k = (1 with probability p, 0 else)
  if $\theta^k = 1$ **then**
    $\hat{x}^k := \mathrm{prox}_{\gamma f}\left(x^k - \gamma \nabla h(x^k) + \gamma h^k\right)$
    $u^{k+1} := \mathrm{prox}_{\tau g^*}(u^k + \tau L \hat{x}^k)$
    $h^{k+1} := -L^* u^{k+1}$
    $x^{k+1} := \hat{x}^k + \frac{\gamma}{p}(h^{k+1} - h^k)$
  else
    $x^{k+1} := \mathrm{prox}_{\gamma f}\left(x^k - \gamma \nabla h(x^k) + \gamma h^k\right)$
    $u^{k+1} := u^k$, $h^{k+1} := h^k$
  end if
end for

$$\mathcal{R}^t : r^t \mapsto \begin{cases} \frac{1}{p} r^t & \text{with prob } p \\ 0 & \text{with prob } 1-p \end{cases}$$

Examples

$$\min \; h + f + \sum_{m=1}^{M} g_m$$

RandProx-minibatch

input: initial points $x^0 \in \mathcal{X}$, $(u_i^0)_{m=1}^M \in \mathcal{X}^M$;
stepsize $\gamma > 0$; $s \in \{1, \dots, M\}$
$h^0 := -\sum_{m=1}^{M} u_m^0$
for $k = 0, 1, \dots$ **do**
  $\hat{x}^k := \mathrm{prox}_{\gamma f}\left(x^k - \gamma \nabla h(x^k) + \gamma h^k\right)$
  pick $\Omega^k \subset \{1, \dots, M\}$ of size s unif. at random
  for $m \in \Omega^k$ **do**
    $u_m^{k+1} := \mathrm{prox}_{\frac{1}{\gamma M} g_m^*}(u_m^k + \frac{1}{\gamma M}\hat{x}^k)$
  end for
  for $m \in \{1, \dots, M\} \setminus \Omega^k$ **do**
    $u_m^{k+1} := u_m^k$
  end for
  $h^{k+1} := -\sum_{m=1}^{M} u_m^{k+1}$
  $x^{k+1} := \hat{x}^k + \frac{\gamma M}{s}(h^{k+1} - h^t)$
end for

$\mathcal{R}^k$: sampling

Examples

RandProx-FL

input: initial estimates $(x_i^0)_{i=1}^n \in \mathcal{X}^n$,
$(u_i^0)_{i=1}^n \in \mathcal{X}^n$ such that $\sum_{i=1}^n u_i^0 = 0$;
stepsize $\gamma > 0$; $\omega \geq 0$
for $k = 0, 1, \ldots$ **do**
  for $i = 1, \ldots, n$ at nodes in parallel **do**
    $\hat{x}_i^k := x_i^k - \gamma \nabla h_i(x_i^k) - \gamma u_i^k$
    $a_i^k := \mathcal{R}^k(\hat{x}_i^k)$
    // send compressed vector a_i^k to master
  end for
  $a^k := \frac{1}{n} \sum_{i=1}^n a_i^k$ // aggregation at master
  // broadcast a^k to all nodes
  for $i = 1, \ldots, n$ at nodes in parallel **do**
    $d_i^k := a_i^k - a^k$
    $u_i^{k+1} := u_i^k + \frac{1}{\gamma(1+\omega)^2} d_i^k$
    $x_i^{k+1} := \hat{x}_i^k - \frac{1}{1+\omega} d_i^k$
  end for
end for

$\min \sum_{i=1}^n h_i$

$\mathcal{R}^k$: compression

Conclusion

nonsmooth large-scale optimization

☞ proximal splitting algorithms

Perspectives

- acceleration
- different metric / Bregman distances
- cheaper random/inexact operations
- application to nonconvex optimization