gipsa-lab

An introduction to proximal splitting algorithms for large-scale convex optimization

Laurent Condat,

CNRS research fellow

GIPSA-lab, Grenoble

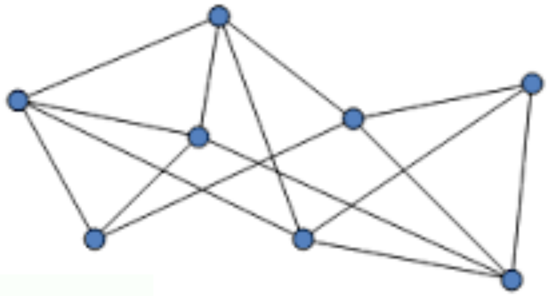
Lyon, July 11, 2017

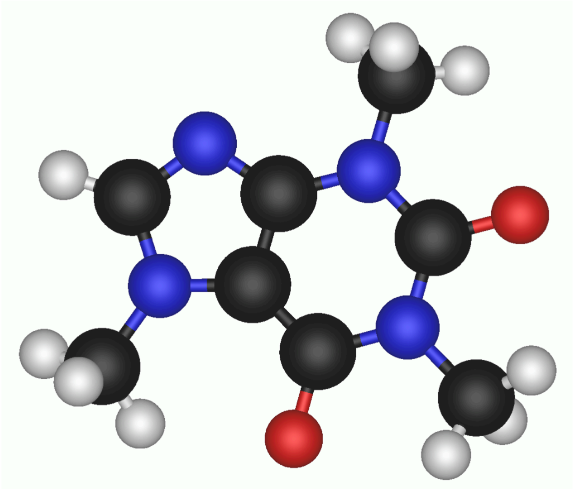
Setting

We place ourselves in a real Hilbert space $\mathcal{X}$

☞ We are looking for an object $\tilde{x} \in \mathcal{X}$, for instance,

$$\begin{bmatrix} \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \end{bmatrix}$$





Goal

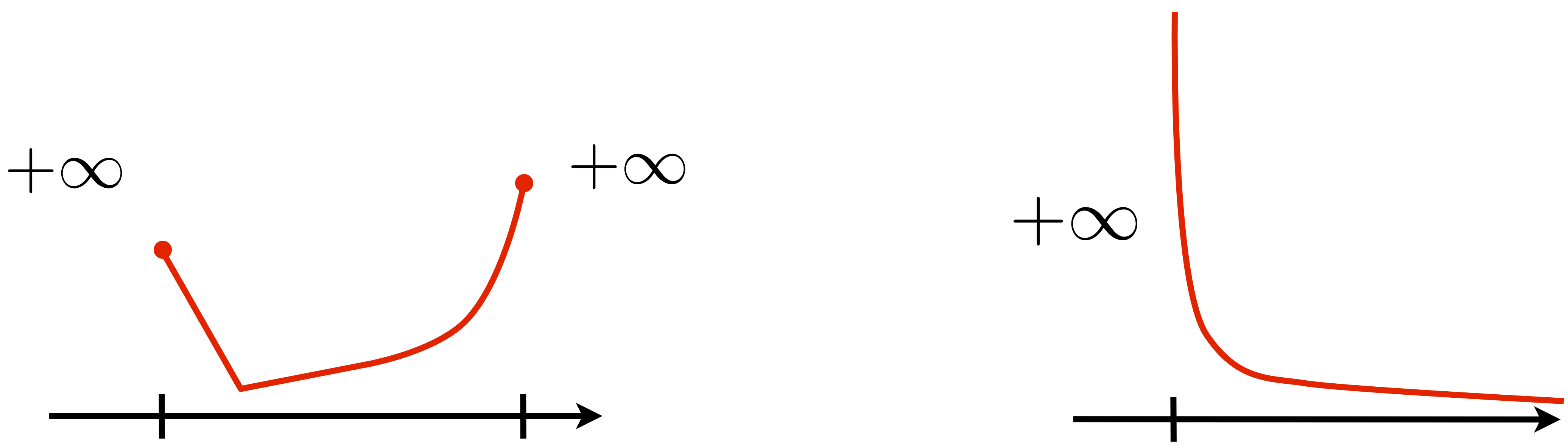
Find $\tilde{x} \in \underset{x \in \mathcal{X}}{\arg\min}\, \Psi(x)$

for a convex, lower semicontinuous, (cost) function $\Psi : \mathcal{X} \to \mathbb{R} \cup \{+\infty\}$

Goal

Find $\tilde{x} \in \underset{x \in \mathcal{X}}{\arg\min}\, \Psi(x)$

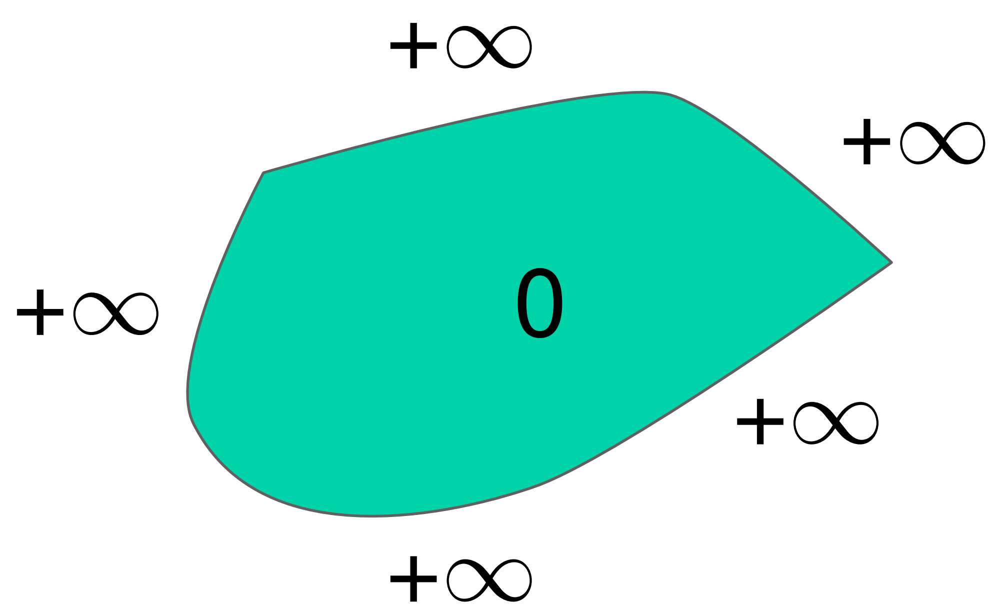
for a convex, lower semicontinuous, (cost) function $\Psi : \mathcal{X} \to \mathbb{R} \cup \{+\infty\}$



Indicator functions

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

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



$\imath_\Omega$ is convex.

Indicator functions

Minimization subject to hard constraints:

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

Goal

☞ We consider the problem

$$\text{Find } \tilde{x} \in \arg\min_{x \in \mathcal{X}} \left\{ \Psi(x) = \sum_{i=1}^{m} g_i(L_i x) \right\}$$

where

- the $\mathcal{U}_i$ and $\mathcal{X}$ are real Hilbert spaces,
- the functions g_i are convex, from $\mathcal{U}_i$ to $\mathbb{R} \cup \{+\infty\}$, lower-semicontinuous,
- the L_i are linear operators from $\mathcal{X}$ to $\mathcal{U}_i$.

Goal

☞ We consider the problem

$$\text{Find } \tilde{x} \in \arg\min_{x \in \mathcal{X}} \left\{ \Psi(x) = \sum_{i=1}^{m} g_i(L_i x) \right\}$$

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

Goal

☞ We consider the problem

$$\text{Find } \tilde{x} \in \arg\min_{x \in \mathcal{X}} \left\{ \Psi(x) = \sum_{i=1}^{m} g_i(L_i x) \right\}$$

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

☞ no implicit operation (inner loop or linear system to solve)

only fast operations in $O(N)$ or $O(N \log N)$, with N = dim.

typically, $N \sim 10^6$–10^9

first-order method (no use of Hessian or so)

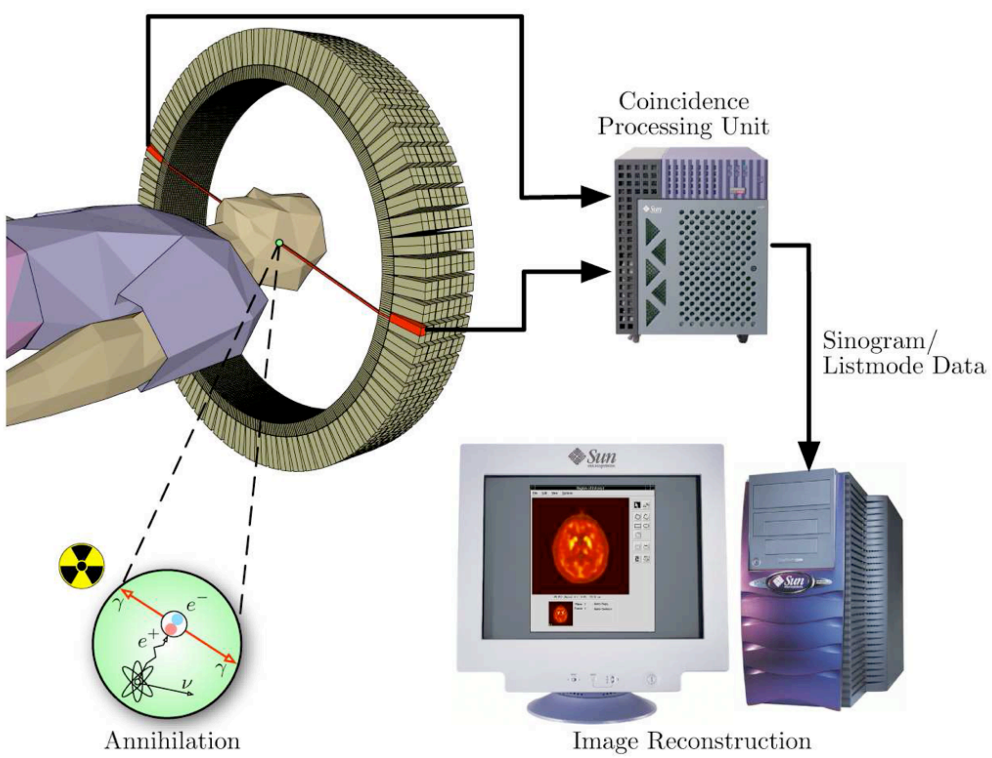
Motivation: inverse problems

Given partial and noisy observations / measurements

$$y \approx Ax^\sharp$$

estimate the unknown image $x^\sharp$ by solving

$$\text{Find} \quad \tilde{x} \in \underset{x \in \mathbb{R}^{N_1 \times N_2}}{\arg\min} \left\{ d(Ax, y) + \imath_\Omega(x) + r(x) \right\}$$



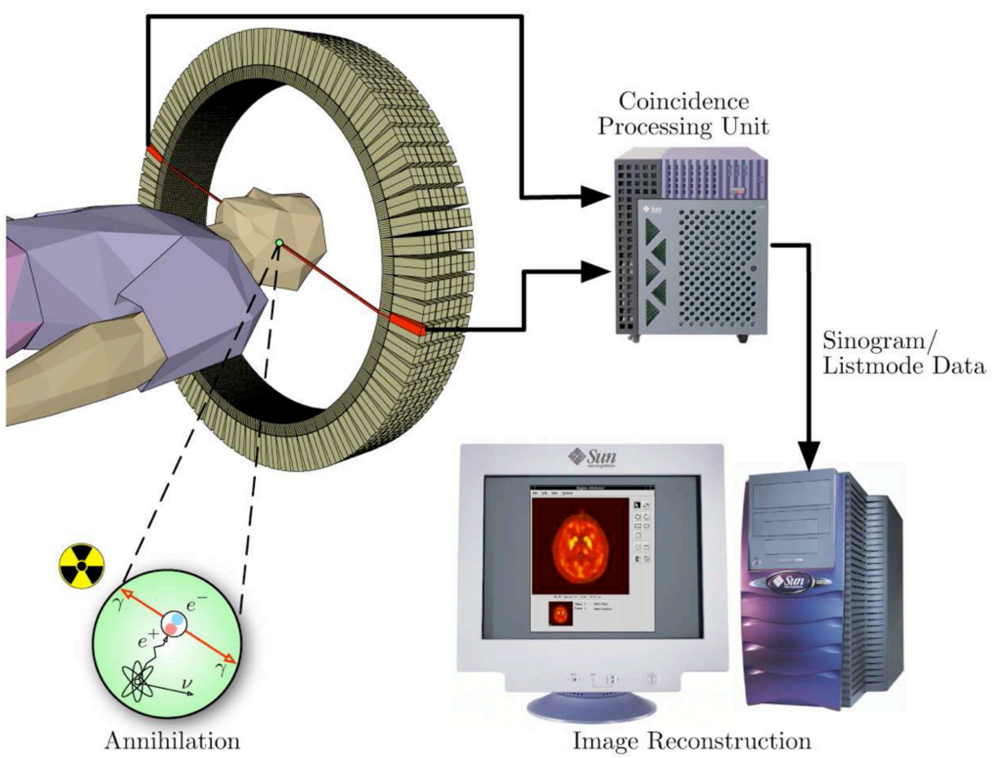
Motivation: inverse problems

Given partial and noisy observations / measurements

$$y \approx Ax^\sharp$$

estimate the unknown image $x^\sharp$ by solving

$$\text{Find} \quad \tilde{x} \in \underset{x \in \mathbb{R}^{N_1 \times N_2}}{\arg\min} \left\{ \frac{1}{2} \| Ax - y \|_2^2 + \imath_\Omega(x) + \lambda \, \mathrm{TV}(x) \right\}$$



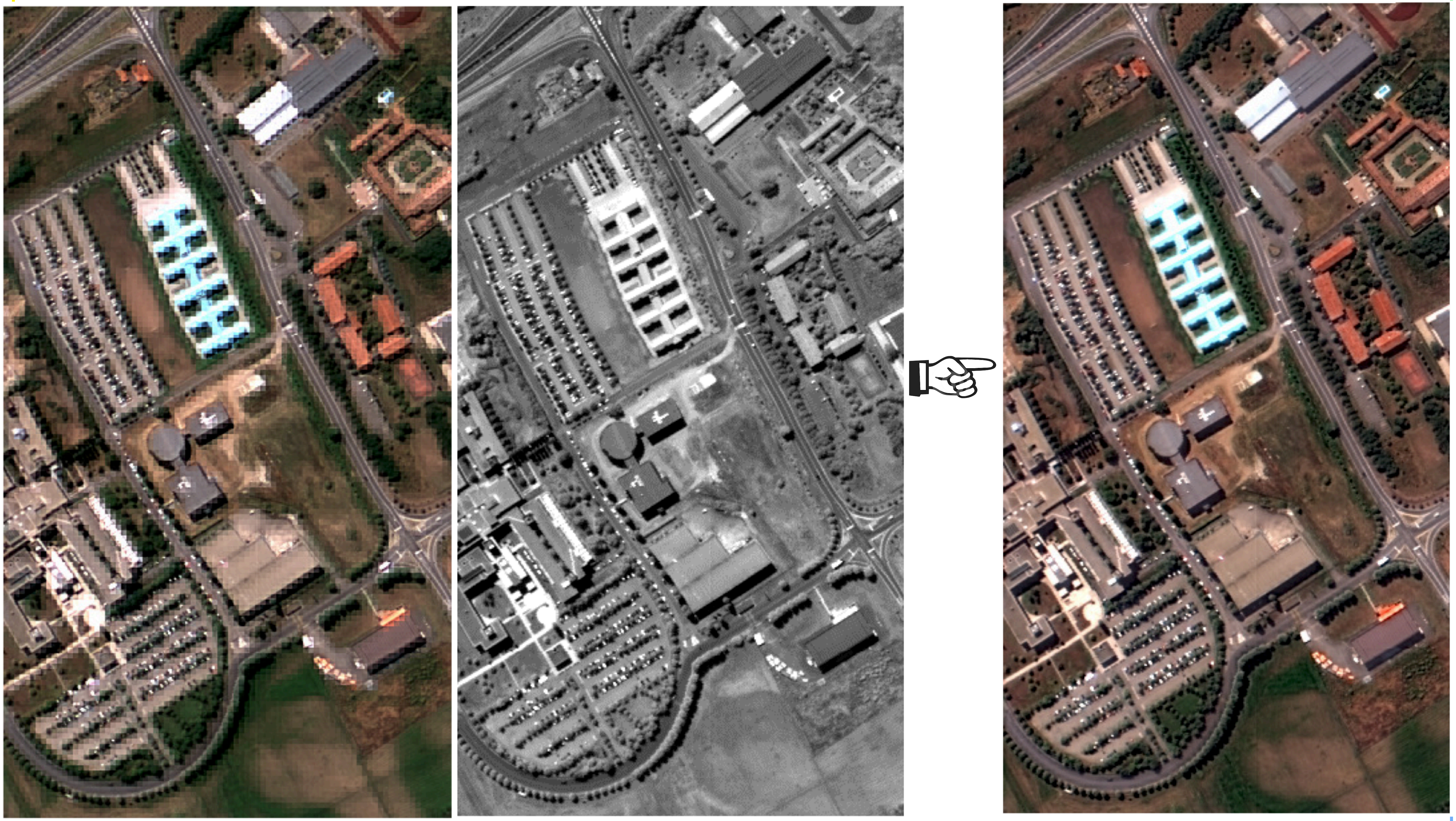
Example 1: deblurring

$$\text{Find} \quad \hat{x} \in \underset{x \in \mathbb{R}^{N_1 \times N_2}}{\arg\min} \left\{ \frac{1}{2} \|Ax - y\|_2^2 + \imath_\Omega(x) + \lambda \, \mathrm{TV}(x) \right\}$$

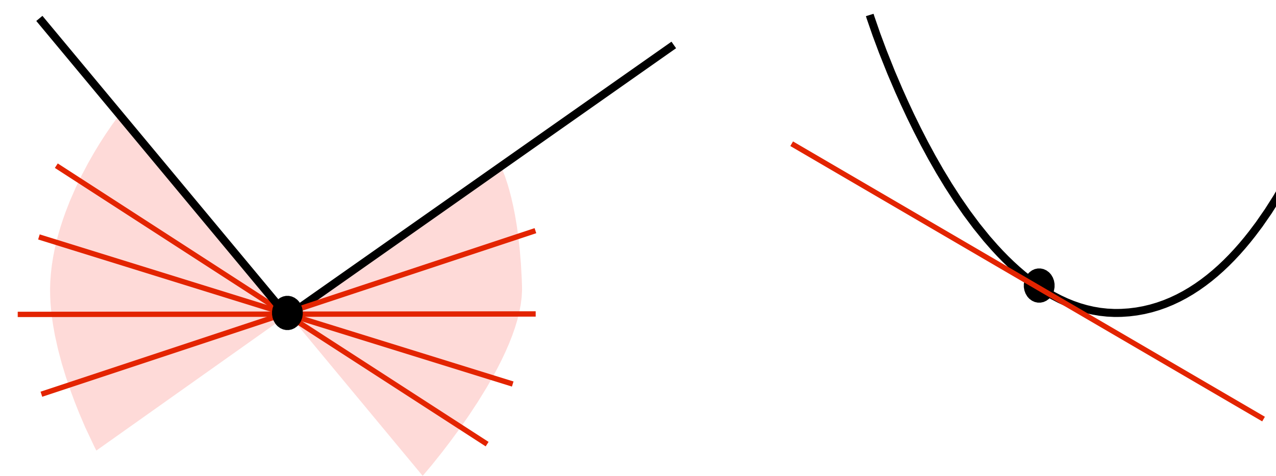
A: convolution

Example 1: deblurring

Ex. 2: Multispectral pansharpening/fusion



The subdifferential



$$\partial f : \mathcal{X} \to 2^{\mathcal{X}}$$
$$x \mapsto \big\{u \in \mathcal{X}\ :\ \forall y \in \mathcal{X},\ f(x) + \langle y - x, u\rangle \leq f(y)\big\}$$

$\to \partial f(x)$ is the set of gradients (slopes) of the affine minorants of f at x.

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

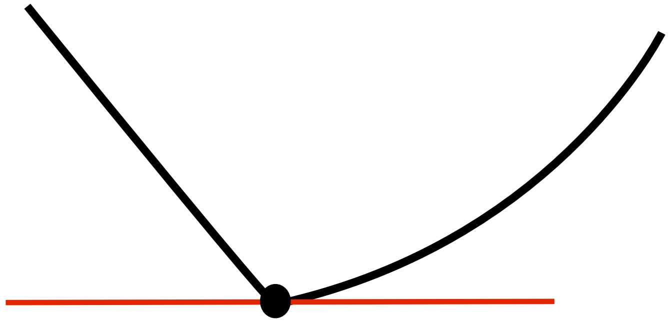
Fermat's rule

$$\underset{x \in \mathcal{X}}{\text{minimize}} \;\; f(x)$$

$$\equiv$$

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

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



Pierre de Fermat, 1601-1665

Fermat's rule

$$\underset{x \in \mathcal{X}}{\text{minimize}} \;\; f(x)$$

$$\equiv$$

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

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

Pierre de Fermat, 1601-1665

With mild hypotheses,

$$\partial f(x) + \partial g(x) = \partial (f+g)(x)$$

Iterative algorithms

Principle: design an algorithm which iterates

$x^{(n+1)} = T(x^{(n)})$

for some operator $T : \mathcal{X} \to \mathcal{X}$

Convergence: $\exists \tilde{x}$ such that $\|x^{(n)} - \tilde{x}\| \to 0$

Iterative algorithms

Principle: design an algorithm which iterates

$x^{(n+1)} = T(x^{(n)})$

for some **nonexpansive** operator $T : \mathcal{X} \to \mathcal{X}$

i.e. $\forall x, y,\ \|T(x) - T(y)\| \leq \|x - y\|$,

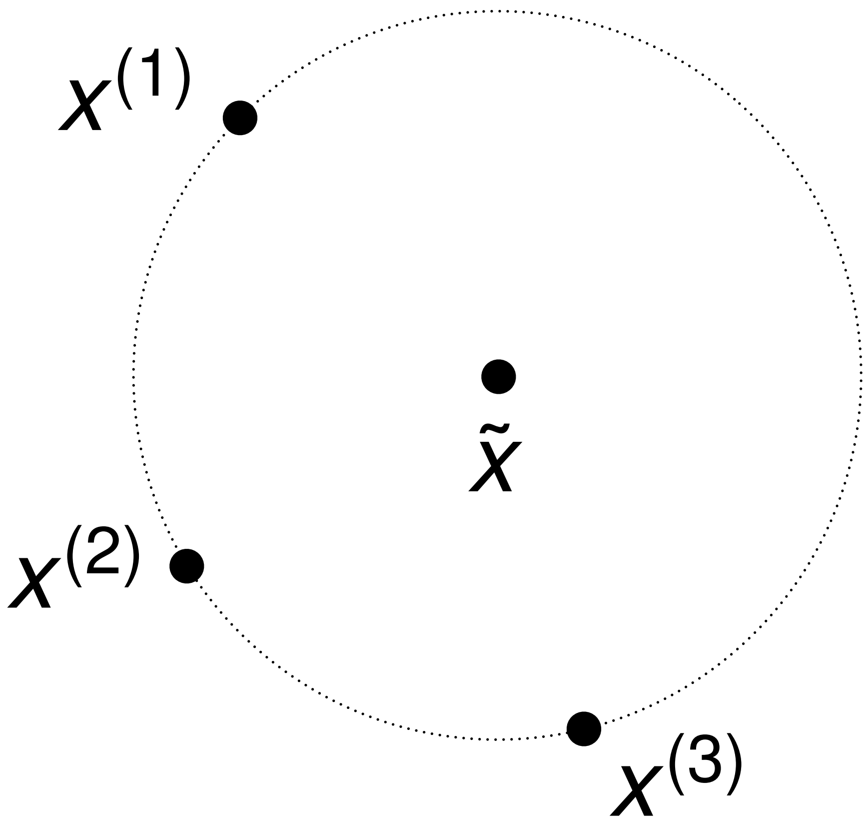
and such that every **fixed point** of T, i.e. $T(\tilde{x}) = \tilde{x}$,

is a solution.

Iterative algorithms

But nonexpansiveness is not sufficient :

for instance, $\mathcal{X} = \mathbb{R}^2$ and T is a rotation.

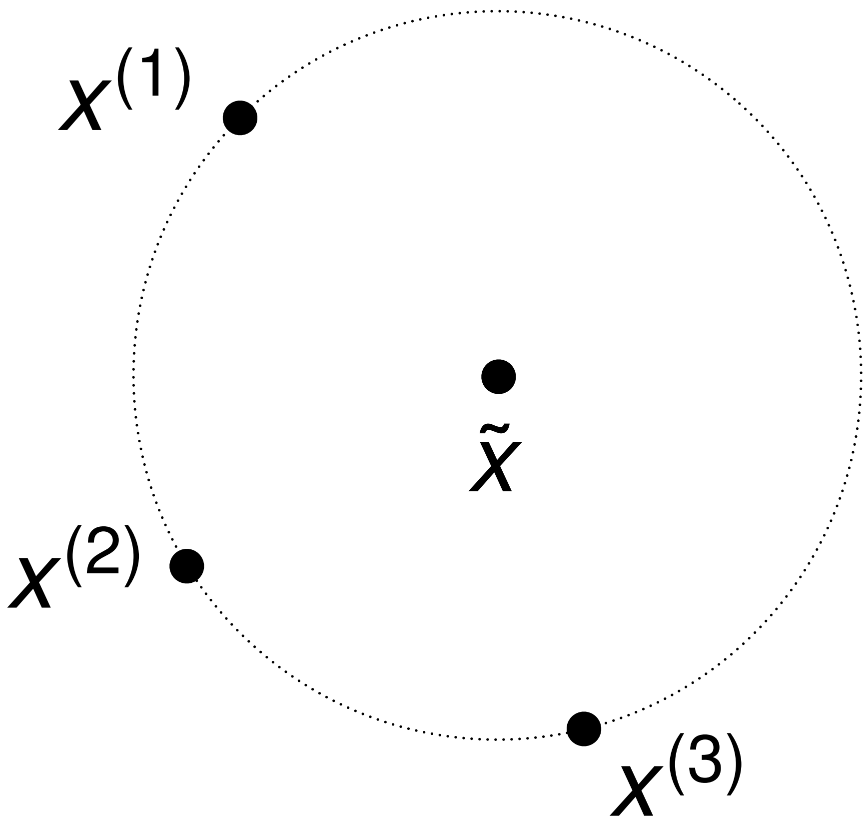


Iterative algorithms

But nonexpansiveness is not sufficient :

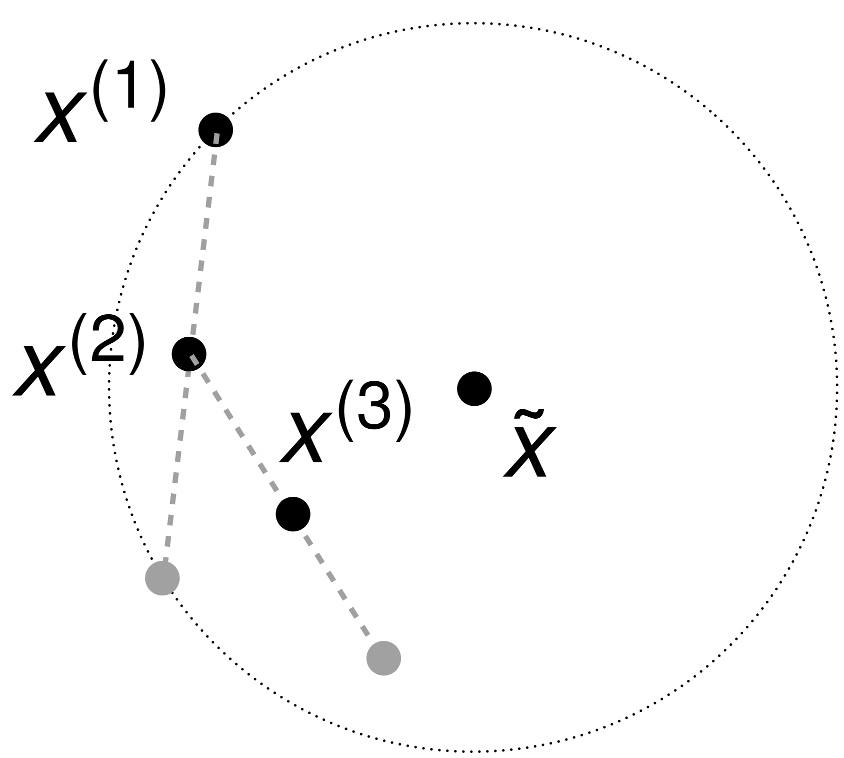
for instance, $\mathcal{X} = \mathbb{R}^2$ and T is a rotation.

☞ we need a bit more than nonexpansiveness.



Iterative algorithms

Definition: T is α-averaged if
$T = \alpha T' + (1-\alpha)\mathrm{Id}$,
for some $0 < \alpha < 1$ and
nonexpansive op. T'.



Iterative algorithms

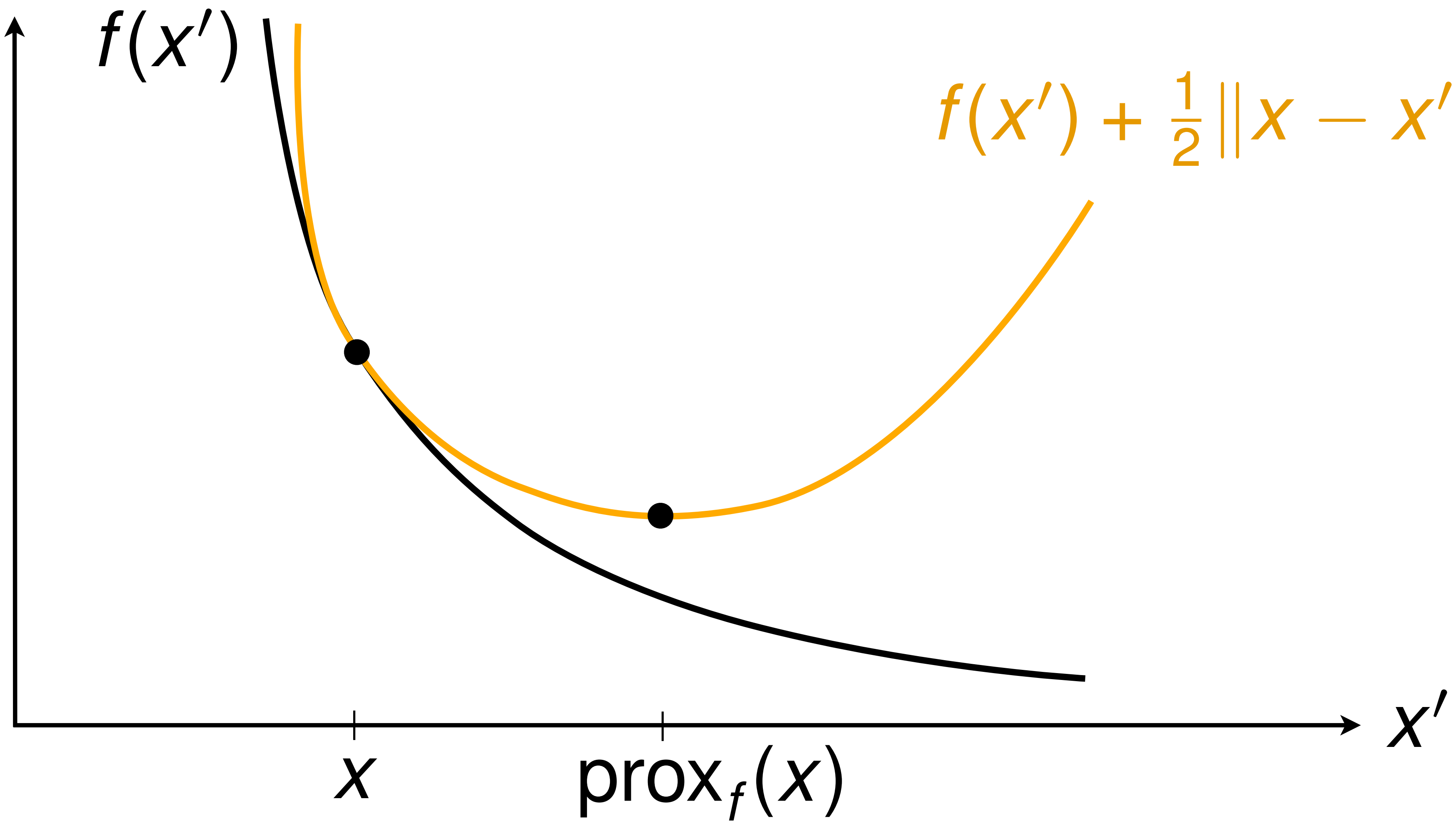
Theorem (Krasnoselskii–Mann)
If T is $\alpha-$averaged, for some $0 < \alpha < 1$,
and a fixed point of T exists,
then the iteration $x^{(n+1)} = Tx^{(n)}$
converges to some fixed point $\tilde{x}$ of T.

The proximity operator

$$\mathrm{prox}_f : \mathcal{H} \to \mathcal{H} : x \mapsto \operatorname*{arg\,min}_{x' \in \mathcal{H}} f(x') + \frac{1}{2}\|x - x'\|^2$$

The proximity operator

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



The proximity operator

$$\mathrm{prox}_f : \mathcal{H} \to \mathcal{H} : x \mapsto \underset{x' \in \mathcal{H}}{\arg\min}\, f(x') + \frac{1}{2}\|x - x'\|^2$$

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

The proximity operator

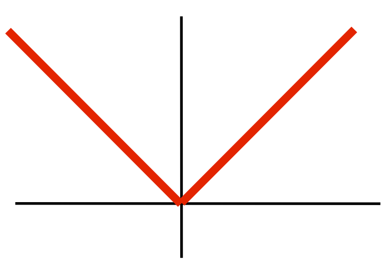
$$\mathrm{prox}_f : \mathcal{H} \rightarrow \mathcal{H} : x \mapsto \underset{x' \in \mathcal{H}}{\arg\min}\, f(x') + \frac{1}{2}\|x - x'\|^2$$

$$\mathrm{prox}_{\imath_\Omega} = P_\Omega$$

The proximity operator

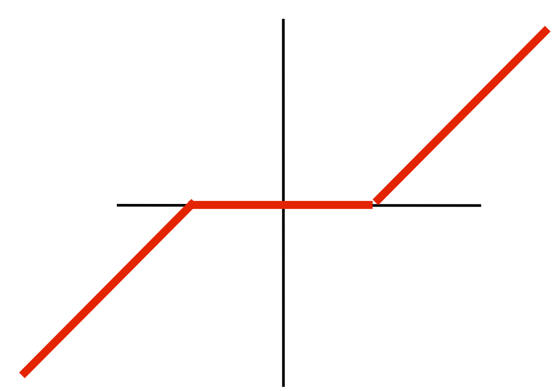
$$\operatorname{prox}_f : \mathcal{H} \to \mathcal{H} : x \mapsto \underset{x' \in \mathcal{H}}{\arg\min}\, f(x') + \tfrac{1}{2}\|x - x'\|^2$$

$$\operatorname{prox}_{\imath_\Omega} = P_\Omega$$



$$f(x) = |x|$$

☞



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

The proximity operator

$g(x)$	$\mathrm{prox}_{\tau g}(x)$	$\nabla g(x)$
0	x	0
$\imath_\Omega(x)$	$P_\Omega(x)$	
$\imath_{(\mathbb{R}_+)^N}(x)$	$\big(\max(x_n,0)\big)_{n=1}^N$	
$\\|x\\|_1 = \sum_{n=1}^N x_n$	$\big(\mathrm{sgn}(x_n)\max(\|x_n\| - \tau, 0)\big)_{n=1}^N$	
$\imath_{\{x'\ ;\ Ax'=y\}}(x)$	$x + A^\dagger(y - Ax)$	
$\frac{1}{2}\\|Ax - y\\|^2$	$(\mathrm{Id} + \tau A^* A)^{-1}(x + \tau A^* y)$	$A^*(Ax - y)$
$\langle Ax, y\rangle = \langle x, A^* y\rangle$	$x - \tau A^* y$	$A^* y$
$\frac{1}{2}\langle Ax, x\rangle$	$(\mathrm{Id} + \tau A)^{-1} x$	Ax

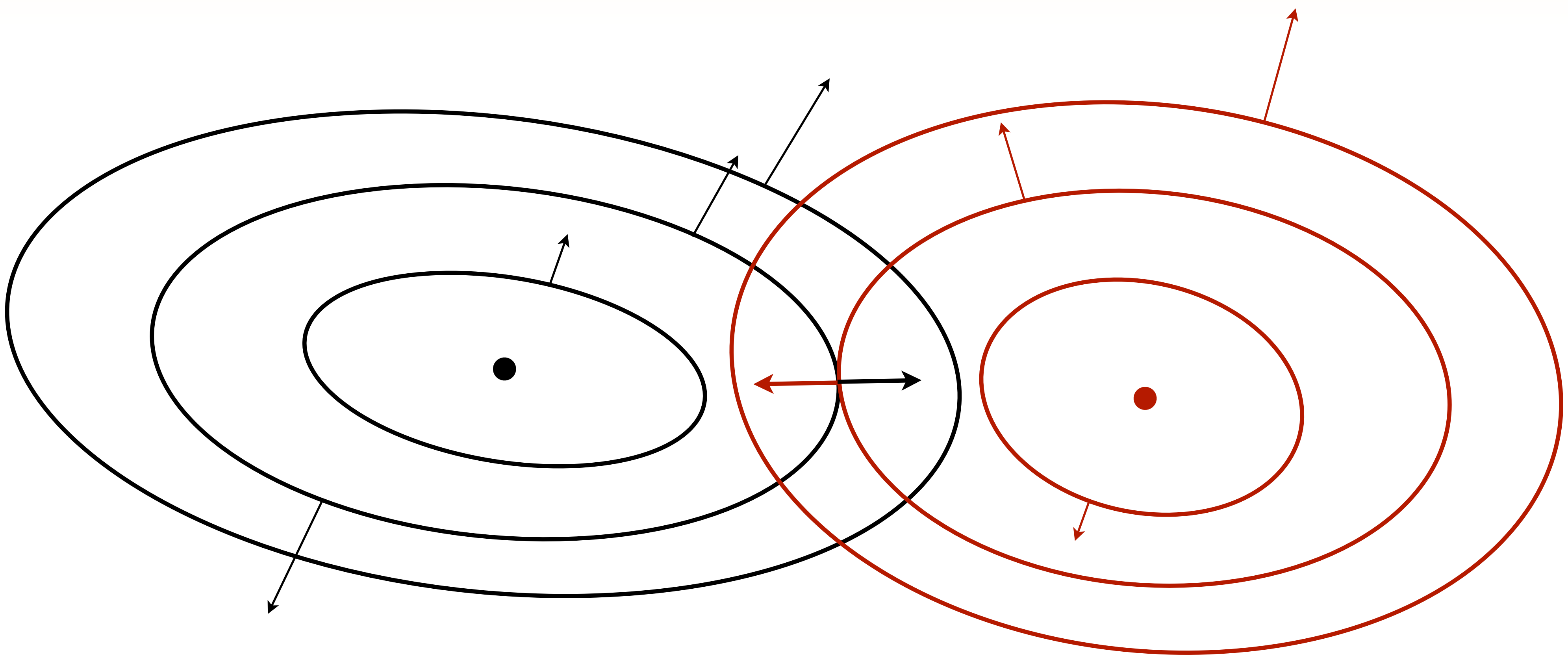
The proximity operator

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 $\rightarrow$ $O(N)$

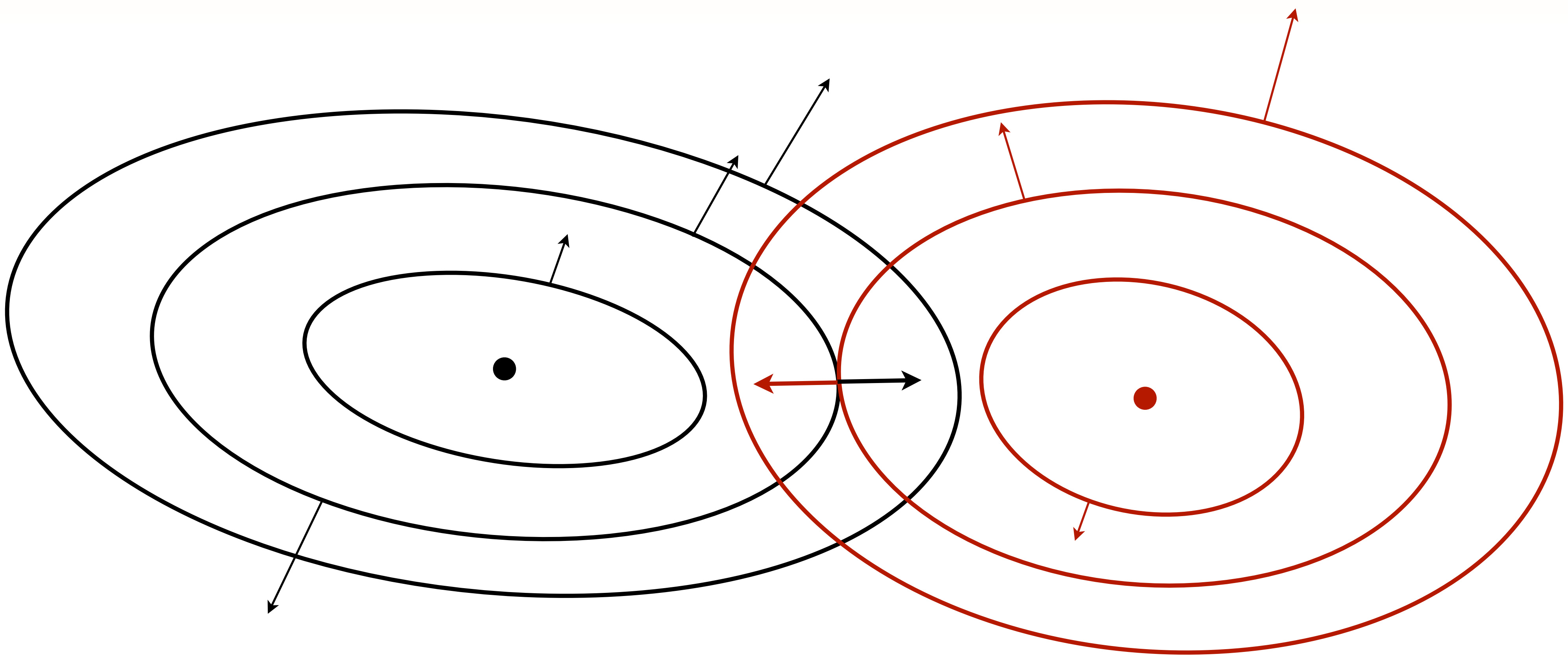
...

Minimization of 2 functions



$$\min\ f + g \ \equiv\ 0 \in \partial f(x) + \partial g(x)$$

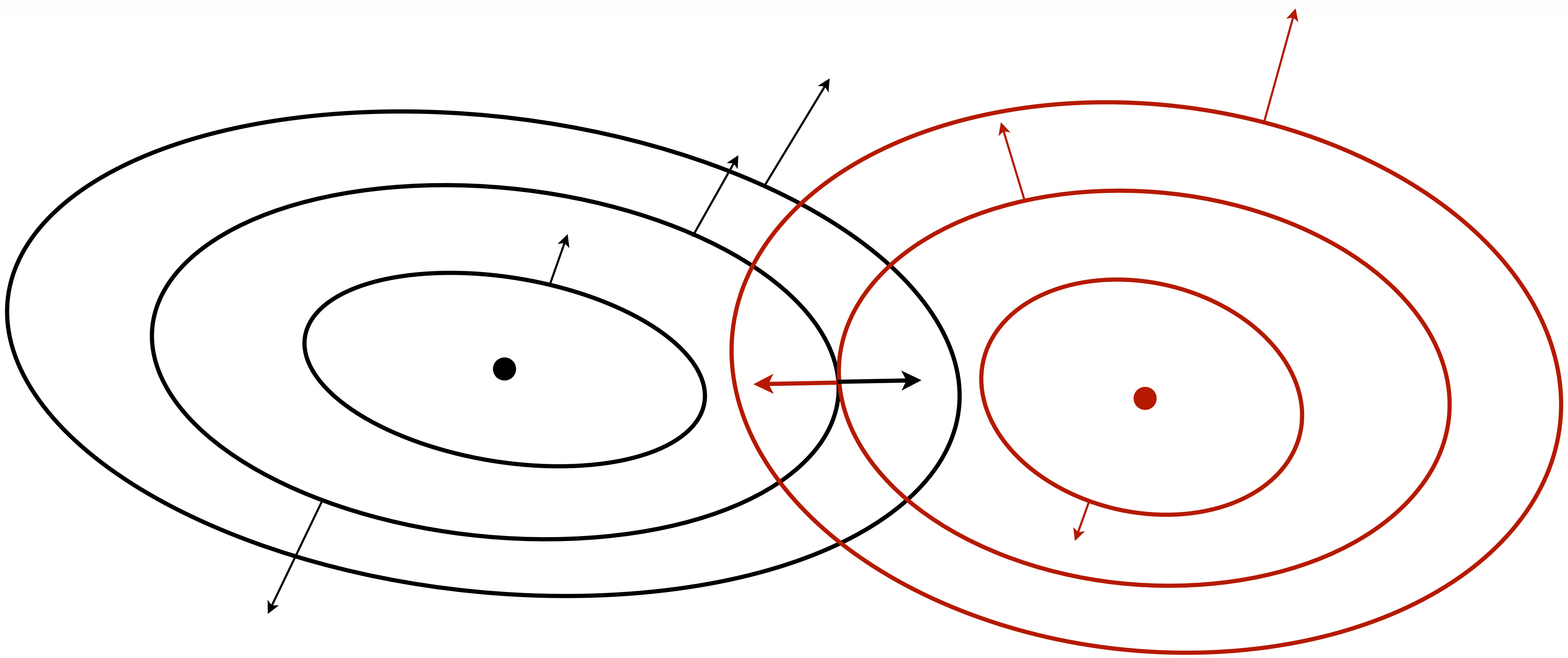
Minimization of 2 functions



$$\min\ f + g \ \equiv\ 0 \in \partial f(x) + \partial g(x)$$

☞ find $(x, u) \in \mathcal{X}^2$ such that $u \in \partial g(x)$ and $-u \in \partial f(x)$.

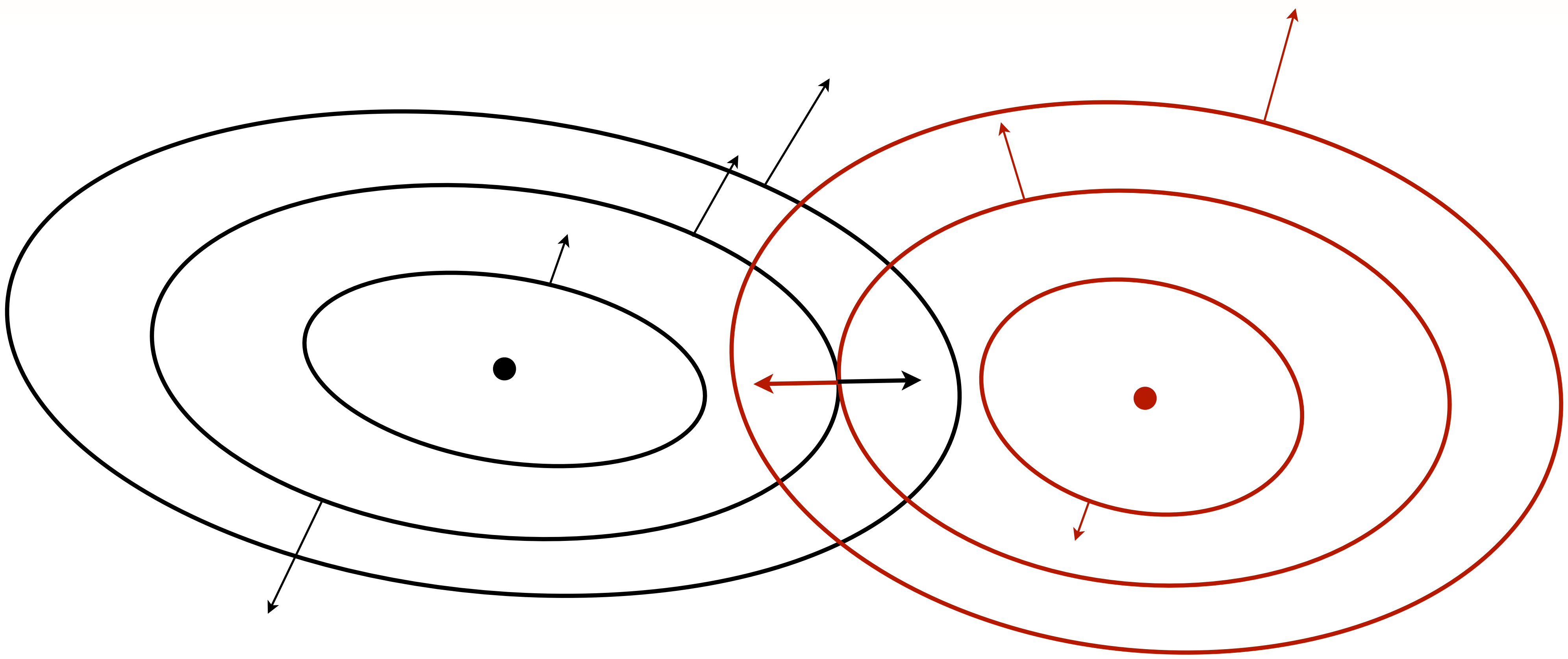
Minimization of 2 functions



$$\min \; f + g \;\; \equiv \;\; 0 \in \partial f(x) + \partial g(x)$$

☞ find $(x, u) \in \mathcal{X}^2$ such that $\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \begin{pmatrix} \partial f(x) + u \\ -x + (\partial g)^{-1}(u) \end{pmatrix}$

Minimization of 2 functions



$$\min\ f + g \ \equiv\ 0 \in \partial f(x) + \partial g(x)$$

☞ find $(x, u) \in \mathcal{X}^2$ such that $\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \begin{pmatrix} \partial f(x) + u \\ -x + (\partial g)^{-1}(u) \end{pmatrix}$

(a monotone inclusion)

Minimization of 2 functions

$$\underset{x \in \mathcal{X}}{\text{minimize}} \;\; f(x) + g(x)$$

☞ design an algorithm such that, $\forall n \in \mathbb{N}$,

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix} \in \begin{pmatrix} \partial f(x^{(n+1)}) + u^{(n+1)} \\ -x^{(n+1)} + \partial g^*(u^{(n+1)}) \end{pmatrix} + \begin{pmatrix} \frac{1}{\tau}\mathrm{Id} & -\mathrm{Id} \\ -\mathrm{Id} & \tau\mathrm{Id} \end{pmatrix} \begin{pmatrix} x^{(n+1)} - x^{(n)} \\ u^{(n+1)} - u^{(n)} \end{pmatrix}$$

Minimization of 2 functions

$$\underset{x\in\mathcal{X}}{\text{minimize}}\ f(x)+g(x)$$

☞ design an algorithm such that, $\forall n \in \mathbb{N}$,

$$\begin{pmatrix}0\\0\end{pmatrix} \in \begin{pmatrix}\partial f(x^{(n+1)}) + u^{(n+1)}\\ -x^{(n+1)} + \partial g^*(u^{(n+1)})\end{pmatrix} + \begin{pmatrix}\frac{1}{\tau}\mathrm{Id} & -\mathrm{Id}\\ -\mathrm{Id} & \tau\mathrm{Id}\end{pmatrix}\begin{pmatrix}x^{(n+1)} - x^{(n)}\\ u^{(n+1)} - u^{(n)}\end{pmatrix}$$

☞ Douglas–Rachford iteration:

$$\left\lfloor \begin{array}{ll} x^{(n+1)} & = \mathrm{prox}_{\tau f}\big(s^{(n)}\big)\\ z^{(n+1)} & = \mathrm{prox}_{\tau g}\big(2x^{(n+1)} - s^{(n)}\big)\\ s^{(n+1)} & = s^{(n)} + z^{(n+1)} - x^{(n+1)} \end{array}\right.$$