Optimization considerations for regularizations of inverse and learning problems

Hugo Raguet[1]

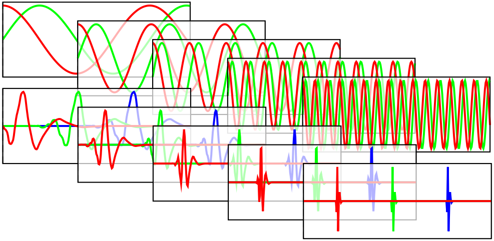
Statistics seminar at LIRMM, Montpellier
April 11, 2018

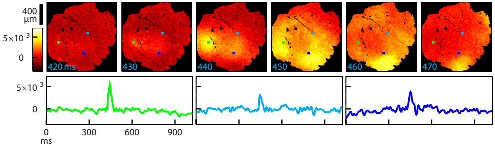
[1] hugo.raguet@gmail.com

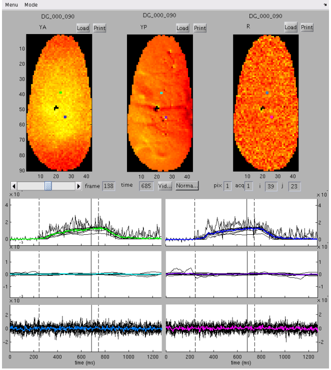
Let me introduce myself briefly

Ph.D. at Paris-Dauphine University

structured sparse modeling for neuroimaging







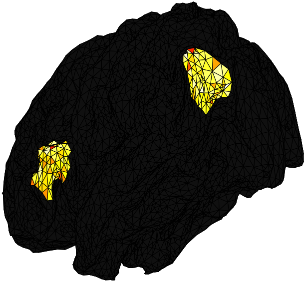
Let me introduce myself briefly

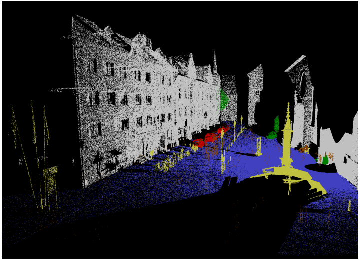
Ph.D. at Paris-Dauphine University

structured sparse modeling for neuroimaging

Lecturer at Aix-Marseille University

optimization for signal and learning on graphs





Let me introduce myself briefly

Ph.D. at Paris-Dauphine University

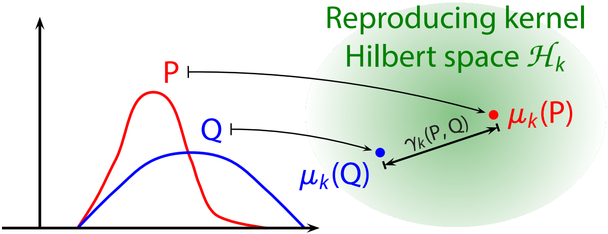
structured sparse modeling for neuroimaging

Lecturer at Aix-Marseille University

optimization for signal and learning on graphs

Postdoc at French Commission for Atomic Energy

dependence measures for sensitivity analysis



Some Motivation

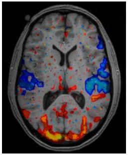
Proximal Splitting

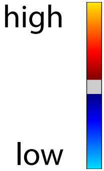
Variants and Accelerations

Cut-pursuit Algorithm

An Example in functional MRI

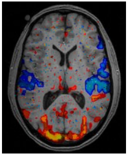
Observing the brain at work

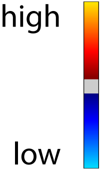




An Example in functional MRI

Observing the brain at work

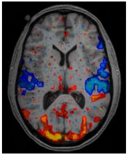


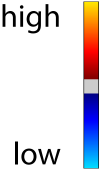


$x^{(1)} \in \mathbb{R}^V$

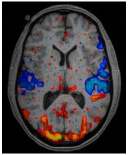
An Example in functional MRI

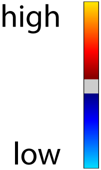
Observing the brain at work





$x^{(1)} \in \mathbb{R}^V$

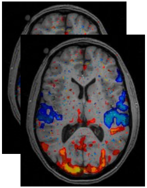


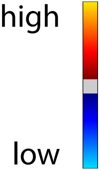


$x^{(2)} \in \mathbb{R}^V$

An Example in functional MRI

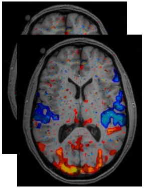
Observing the brain at work

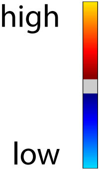




$x^{(1)} \in \mathbb{R}^V$

$x^{(3)} \in \mathbb{R}^V$



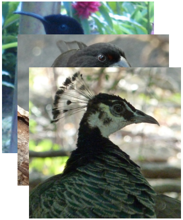


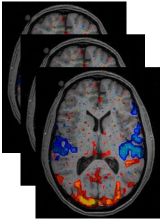
$x^{(2)} \in \mathbb{R}^V$

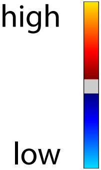
$x^{(5)} \in \mathbb{R}^V$

An Example in functional MRI

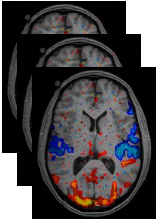
Observing the brain at work

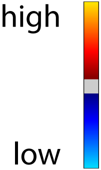






$x^{(1)} \in \mathbb{R}^V$
$x^{(3)} \in \mathbb{R}^V$
$x^{(4)} \in \mathbb{R}^V$

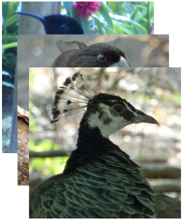


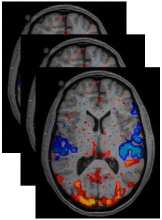


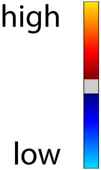
$x^{(2)} \in \mathbb{R}^V$
$x^{(5)} \in \mathbb{R}^V$
$x^{(6)} \in \mathbb{R}^V$

An Example in functional MRI

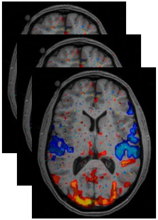
A binary logistic classification problem

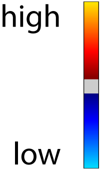






$$x^{(1)} \in \mathrm{I\!R}^V$$
$$x^{(3)} \in \mathrm{I\!R}^V$$
$$x^{(4)} \in \mathrm{I\!R}^V$$
$$c^{(n)} = +1$$





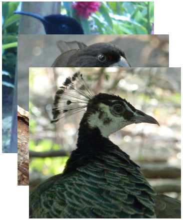
$$x^{(2)} \in \mathrm{I\!R}^V$$
$$x^{(5)} \in \mathrm{I\!R}^V$$
$$x^{(6)} \in \mathrm{I\!R}^V$$
$$c^{(n)} = -1$$

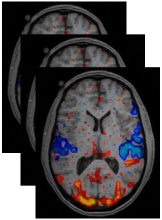
An Example in functional MRI

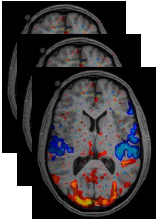
A binary logistic classification problem

for $n \in \{1, \dots, N\}$,

$$c^{(n)} = \operatorname{sign}(\langle w, x^{(n)} \rangle)$$

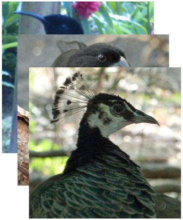


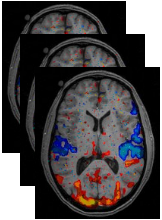




An Example in functional MRI

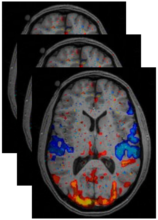
A binary logistic classification problem

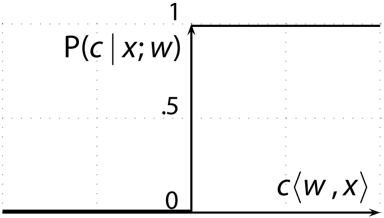




for $n \in \{1, \dots, N\}$,

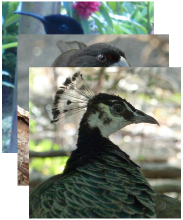
$$c^{(n)} = \operatorname{sign}(\langle w, x^{(n)} \rangle)$$

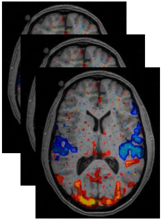




An Example in functional MRI

A binary logistic classification problem

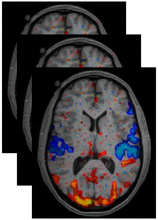


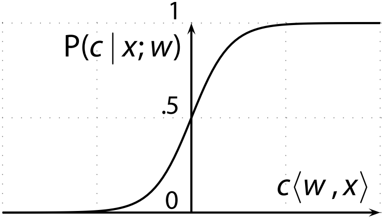


for $n \in \{1, \dots, N\}$,

$$P\left(c^{(n)} \mid x^{(n)}; w\right) = \sigma\left(c^{(n)} \langle w, x^{(n)} \rangle\right)$$

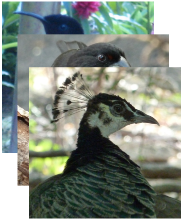
$$\sigma : t \mapsto 1 / \left(1 + \exp(-t)\right)$$

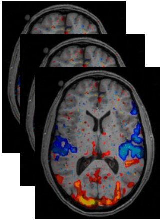




An Example in functional MRI

A binary logistic classification problem





for $n \in \{1, \dots, N\}$,

$$P(c^{(n)} \mid x^{(n)}; w) = \sigma(c^{(n)} \langle w, x^{(n)} \rangle)$$

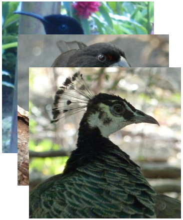
$$\sigma : t \mapsto 1/(1 + \exp(-t))$$

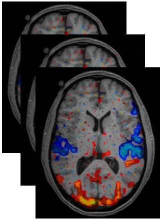
Maximize log-likelihood

Find $w \in \arg\max_{\mathbb{R}^V} \sum_n \log P(c^{(n)} \mid x^{(n)}; w)$

An Example in functional MRI

A binary logistic classification problem





for $n \in \{1, \dots, N\}$,

$$P(c^{(n)} \mid x^{(n)}; w) = \sigma(c^{(n)} \langle w, x^{(n)} \rangle)$$

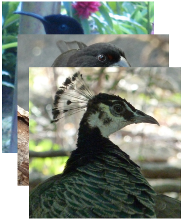
$$\sigma : t \mapsto 1/(1 + \exp(-t))$$

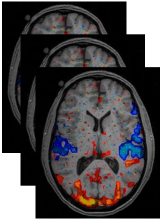
Maximize log-likelihood

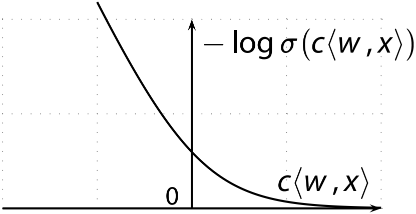
Find $w \in \arg\min_{\mathbb{R}^V} \sum_n -\log \sigma(c^{(n)} \langle w, x^{(n)} \rangle)$

An Example in functional MRI

A binary logistic classification problem







Maximize log-likelihood

Find $w \in \arg\min_{\mathbb{R}^V} \sum_n -\log\sigma\left(c^{(n)}\langle w, x^{(n)}\rangle\right)$

Optimization

Simple, smooth and convex

Maximize log-likelihood

Find $w \in \arg\min_{\mathbb{R}^V} F : w \mapsto \sum_n -\log \sigma\left(c^{(n)} \langle w, x^{(n)} \rangle\right)$

Optimization

Simple, smooth and convex

Maximize log-likelihood

Find $w \in \arg\min_{\mathbb{R}^V} F : w \mapsto \sum_n -\log \sigma\left(c^{(n)} \langle w, x^{(n)} \rangle\right)$

- First order: $$(\nabla F)_v = \sum_n -c^{(n)} w_v x_v^{(n)} \left(1 - \sigma\left(c^{(n)} \langle w, x^{(n)} \rangle\right)\right)$$

Optimization

Simple, smooth and convex

Maximize log-likelihood

Find $w \in \arg\min_{\mathbb{R}^V} F : w \mapsto \sum_n -\log \sigma(c^{(n)} \langle w, x^{(n)} \rangle)$

- First order: $(\nabla F)_v = \sum_n -c^{(n)} w_v x_v^{(n)} (1 - \sigma(c^{(n)} \langle w, x^{(n)} \rangle))$
- Second order: $(\nabla^2 F)_{uv} = \sum_n (c^{(n)})^2 w_u w_v x_u^{(n)} x_v^{(n)} \sigma (1 - \sigma)$

Optimization

Simple, smooth and convex

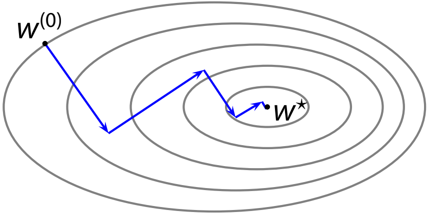
Maximize log-likelihood

Find $w \in \arg\min_{\mathbb{R}^V} F : w \mapsto \sum_n -\log \sigma(c^{(n)} \langle w, x^{(n)} \rangle)$

- First order: $$(\nabla F)_v = \sum_n -c^{(n)} w_v x_v^{(n)} (1 - \sigma(c^{(n)} \langle w, x^{(n)} \rangle))$$
- Second order: $$(\nabla^2 F)_{uv} = \sum_n (c^{(n)})^2 w_u w_v x_u^{(n)} x_v^{(n)} \sigma(1 - \sigma)$$

Gradient descent

$$w^{(k+1)} = w^{(k)} - \gamma \nabla F(w^{(k)})$$



Optimization

Simple, smooth and convex

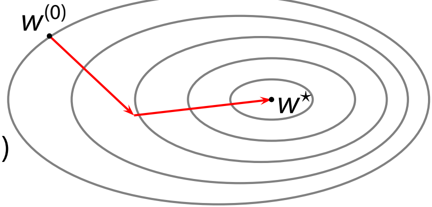
Maximize log-likelihood

Find $w \in \arg\min_{\mathbb{R}^V} F : w \mapsto \sum_n -\log\sigma(c^{(n)}\langle w, x^{(n)}\rangle)$

- First order: $(\nabla F)_v = \sum_n -c^{(n)} w_v x_v^{(n)} (1 - \sigma(c^{(n)}\langle w, x^{(n)}\rangle))$
- Second order: $(\nabla^2 F)_{uv} = \sum_n (c^{(n)})^2 w_u w_v x_u^{(n)} x_v^{(n)} \sigma(1-\sigma)$

(Quasi-)Newton method

$w^{(k+1)} = w^{(k)} - \gamma(\nabla^2 F)^{-1} \nabla F(w^{(k)})$

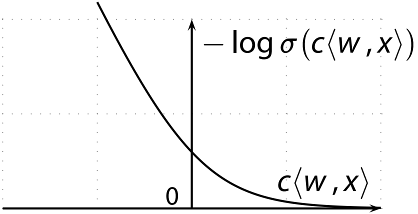


Regularization

Stability and prior knowledge

$$\log\left(1+\exp\left(-c^{(n)}\langle w, x^{(n)}\rangle\right)\right)$$

$$N \ll V; \qquad \|w^\star\| \to +\infty$$

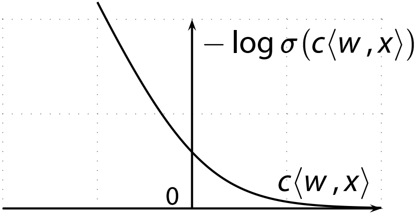


Regularization

Stability and prior knowledge

$$\log\Big(1 + \exp\big(-c^{(n)}\langle w, x^{(n)}\rangle\big)\Big)$$

$$N \ll V; \qquad \|w^\star\| \to +\infty$$



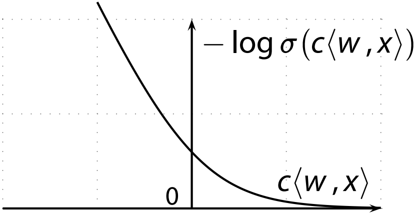
- "Ridge" $\qquad F(w) = \ell(w) + \lambda\|w\|^2$

Regularization

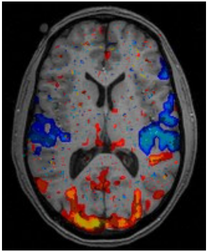
Stability and prior knowledge

$$\log\Big(1+\exp\big(-c^{(n)}\langle w, x^{(n)}\rangle\big)\Big)$$

$$N \ll V; \qquad \|w^\star\| \to +\infty$$



- "Ridge" $\qquad F(w) = \ell(w) + \lambda\|w\|^2$

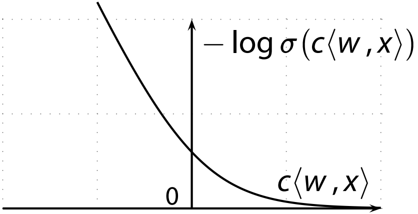


Regularization

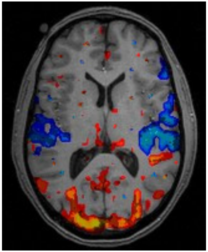
Stability and prior knowledge

$$\log\left(1 + \exp(-c^{(n)}\langle w, x^{(n)}\rangle)\right)$$

$$N \ll V; \qquad \|w^\star\| \to +\infty$$



- "Ridge" $\quad F(w) = \ell(w) + \lambda\|w\|^2$
- "LASSO" $\quad F(w) = \ell(w) + \lambda\sum_v |w_v|$

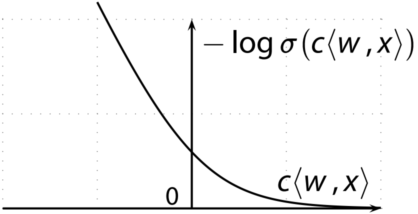


Regularization

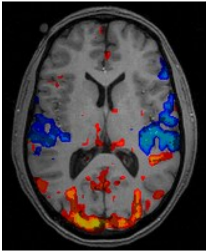
Stability and prior knowledge

$$\log\left(1 + \exp\left(-c^{(n)}\langle w, x^{(n)}\rangle\right)\right)$$

$$N \ll V; \qquad \|w^\star\| \to +\infty$$



- "Ridge" $F(w) = \ell(w) + \lambda\|w\|^2$
- "LASSO" $F(w) = \ell(w) + \lambda\sum_v |w_v|$
- "Group LASSO" $F(w) = \ell(w) + \lambda\sum_b \|w_b\|$

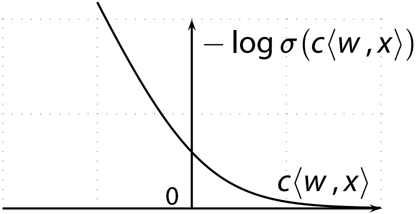


Regularization

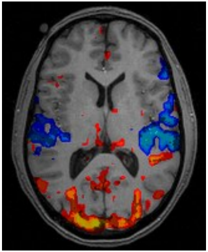
Stability and prior knowledge

$$\log\left(1 + \exp\left(-c^{(n)}\langle w, x^{(n)}\rangle\right)\right)$$

$$N \ll V; \qquad \|w^\star\| \to +\infty$$



- "Ridge" $F(w) = \ell(w) + \lambda \|w\|^2$
- "LASSO" $F(w) = \ell(w) + \lambda \sum_v |w_v|$
- "Group LASSO" $F(w) = \ell(w) + \lambda \sum_b \|w_b\|$
- "Total variation" $F(w) = \ell(w) + \lambda \sum_b \|Dw_b\|$



Some Motivation

Proximal Splitting

Variants and Accelerations

Cut-pursuit Algorithm

Proximal Point Algorithm

Fixed-point algorithm for nonsmooth optimization

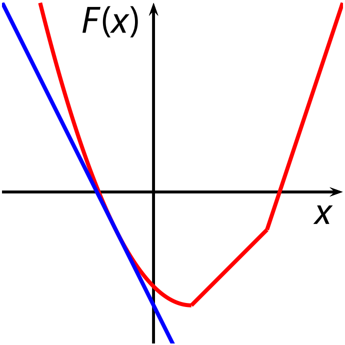
- Gradient and subgradient:

$$\nabla F(x) = u \quad \overset{\text{def}}{\Longleftrightarrow} \quad \forall\, y,\ F(y) = F(x) + \langle u \,|\, y - x \rangle + o(\|y - x\|)$$

- First-order optimality:
  - $0 = \nabla F(x^\star)$

- Fixed point equation:
  - $x^\star = x^\star - \gamma \nabla F(x^\star)$

- Algorithm:
  - $x^{(k+1)} = (\mathrm{Id} - \gamma \nabla F) x^{(k)}$



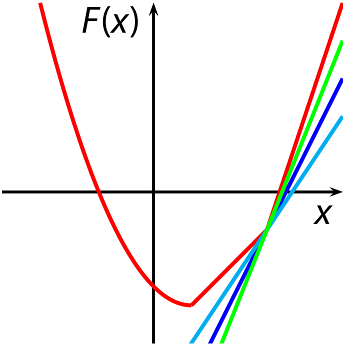
Proximal Point Algorithm

Fixed-point algorithm for nonsmooth optimization

- Gradient and subgradient:

$$\nabla F(x) = u \quad \stackrel{\text{def}}{\Longleftrightarrow} \quad \forall\, y,\ F(y) = F(x) + \langle u \,|\, y - x \rangle + o(\|y - x\|)$$
$$u \in \partial F(x) \quad \stackrel{\text{def}}{\Longleftrightarrow} \quad \forall\, y,\ F(y) \geq F(x) + \langle u \,|\, y - x \rangle$$

- First-order optimality:
  - $0 = \nabla F(x^\star)$
- Fixed point equation:
  - $x^\star = x^\star - \gamma \nabla F(x^\star)$
- Algorithm:
  - $x^{(k+1)} = (\mathrm{Id} - \gamma \nabla F) x^{(k)}$



Proximal Point Algorithm

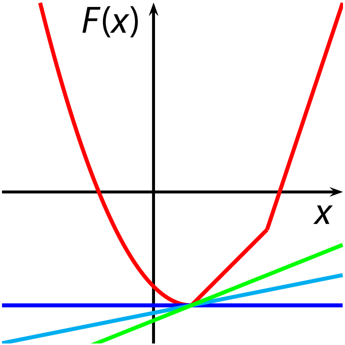
Fixed-point algorithm for nonsmooth optimization

- Gradient and subgradient:

$$\nabla F(x) = u \quad \stackrel{\text{def}}{\Longleftrightarrow} \quad \forall\, y,\ F(y) = F(x) + \langle u \mid y - x \rangle + o(\|y - x\|)$$

$$u \in \partial F(x) \quad \stackrel{\text{def}}{\Longleftrightarrow} \quad \forall\, y,\ F(y) \geq F(x) + \langle u \mid y - x \rangle$$

- First-order optimality:
  - $0 = \nabla F(x^\star)$
  - $0 \in \partial F(x^\star)$
- Fixed point equation:
  - $x^\star = x^\star - \gamma \nabla F(x^\star)$
  - $x^\star \in x^\star - \gamma \partial F(x^\star)$
- Algorithm:
  - $x^{(k+1)} = (\mathrm{Id} - \gamma \nabla F) x^{(k)}$
  - $x^{(k+1)} \in (\mathrm{Id} - \partial F) x^{(k)}$



Proximal Point Algorithm

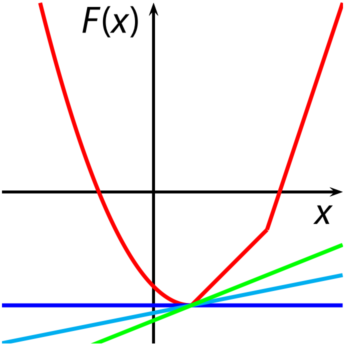
Fixed-point algorithm for nonsmooth optimization

- Gradient and subgradient:

$$\nabla F(x) = u \quad \overset{\text{def}}{\Longleftrightarrow} \quad \forall\, y,\ F(y) = F(x) + \langle u \,|\, y - x \rangle + o(\|y - x\|)$$

$$u \in \partial F(x) \quad \overset{\text{def}}{\Longleftrightarrow} \quad \forall\, y,\ F(y) \geq F(x) + \langle u \,|\, y - x \rangle$$

- First-order optimality:
  - $0 = \nabla F(x^\star)$
  - $0 \in \partial F(x^\star)$
- Fixed point equation:
  - $x^\star = x^\star - \gamma \nabla F(x^\star)$
  - $x^\star + \gamma \partial F(x^\star) \ni x^\star$
- Algorithm:
  - $x^{(k+1)} = (\mathrm{Id} - \gamma \nabla F) x^{(k)}$
  - $x^{(k+1)} = (\mathrm{Id} + \gamma \partial F)^{-1} x^{(k)}$



Proximal Point Algorithm

Fixed-point algorithm for nonsmooth optimization

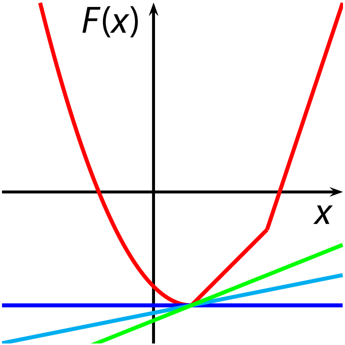
- Gradient and subgradient:

$$\nabla F(x) = u \quad \overset{\text{def}}{\Longleftrightarrow} \quad \forall\, y,\ F(y) = F(x) + \langle u \,|\, y - x \rangle + o(\|y - x\|)$$

$$u \in \partial F(x) \quad \overset{\text{def}}{\Longleftrightarrow} \quad \forall\, y,\ F(y) \geq F(x) + \langle u \,|\, y - x \rangle$$

- First-order optimality:
  - $0 = \nabla F(x^\star)$
  - $0 \in \partial F(x^\star)$
- Fixed point equation:
  - $x^\star = x^\star - \gamma \nabla F(x^\star)$
  - $x^\star + \gamma \partial F(x^\star) \ni x^\star$
- Algorithm:
  - $x^{(k+1)} = (\mathrm{Id} - \gamma \nabla F) x^{(k)}$
  - $x^{(k+1)} = (\mathrm{Id} + \gamma \partial F)^{-1} x^{(k)}$

$$= \arg\min_x \tfrac{1}{2} \|x^{(k)} - x\|^2 + \gamma F(x)$$



Proximal Point Algorithm

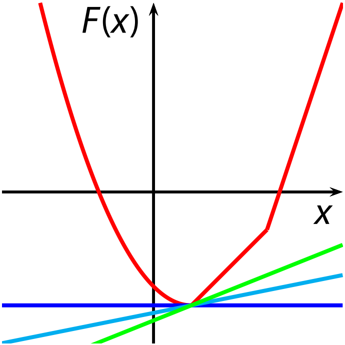
Fixed-point algorithm for nonsmooth optimization

- Gradient and subgradient:

$$\nabla F(x) = u \overset{\text{def}}{\iff} \forall\, y,\ F(y) = F(x) + \langle u \,|\, y - x \rangle + o(\|y - x\|)$$

$$u \in \partial F(x) \overset{\text{def}}{\iff} \forall\, y,\ F(y) \geq F(x) + \langle u \,|\, y - x \rangle$$

- First-order optimality:
  - $0 = \nabla F(x^\star)$
  - $0 \in \partial F(x^\star)$
- Fixed point equation:
  - $x^\star = x^\star - \gamma \nabla F(x^\star)$
  - $x^\star + \gamma \partial F(x^\star) \ni x^\star$
- Algorithm:
  - $x^{(k+1)} = (\mathrm{Id} - \gamma \nabla F) x^{(k)}$
  - $x^{(k+1)} = (\mathrm{Id} + \gamma \partial F)^{-1} x^{(k)}$



$$= \arg\min_x \tfrac{1}{2}\|x^{(k)} - x\|^2 + \gamma F(x) = \mathrm{prox}_{\gamma F}(x^{(k)})$$

Proximal Splitting Algorithms

Primal algorithms

$F = f + g$, where:

- f smooth (Lipschitz-continuous gradient)
- g simple (proximity operator easy to compute)

Proximal Splitting Algorithms

Primal algorithms

$F = f + g$, where:

- f smooth (Lipschitz-continuous gradient)
- g simple (proximity operator easy to compute)

$$0 \quad \in \quad \partial F(x^\star)$$

Proximal Splitting Algorithms

Primal algorithms

$F = f + g$, where:

- f smooth (Lipschitz-continuous gradient)
- g simple (proximity operator easy to compute)

$$\begin{aligned} 0 &\in \partial F(x^\star) \\ 0 &\in (\nabla f + \partial g)x^\star \end{aligned}$$

Proximal Splitting Algorithms

Primal algorithms

$F = f + g$, where:

- f smooth (Lipschitz-continuous gradient)
- g simple (proximity operator easy to compute)

$$\begin{aligned} 0 &\in \partial F(x^\star) \\ 0 &\in (\nabla f + \partial g)x^\star \\ -\nabla f(x^\star) &\in \partial g(x^\star) \end{aligned}$$

Proximal Splitting Algorithms

Primal algorithms

$F = f + g$, where:

- f smooth (Lipschitz-continuous gradient)
- g simple (proximity operator easy to compute)

$$\begin{aligned}
0 &\in \partial F(x^\star) \\
0 &\in (\nabla f + \partial g)x^\star \\
-\nabla f(x^\star) &\in \partial g(x^\star) \\
(\mathrm{Id} - \nabla f)x^\star &\in (\mathrm{Id} + \partial g)x^\star
\end{aligned}$$

Proximal Splitting Algorithms

Primal algorithms

$F = f + g$, where:

- f smooth (Lipschitz-continuous gradient)
- g simple (proximity operator easy to compute)

$$
\begin{aligned}
0 &\in \partial F(x^\star) \\
0 &\in (\nabla f + \partial g)x^\star \\
-\nabla f(x^\star) &\in \partial g(x^\star) \\
(\mathrm{Id} - \nabla f)x^\star &\in (\mathrm{Id} + \partial g)x^\star \\
(\mathrm{Id} + \partial g)^{-1}(\mathrm{Id} - \nabla f)x^\star &= x^\star
\end{aligned}
$$

Proximal Splitting Algorithms

Primal algorithms

$F = f + g$, where:

- f smooth (Lipschitz-continuous gradient)
- g simple (proximity operator easy to compute)

$$
\begin{aligned}
0 &\in \partial F(x^\star) \\
0 &\in (\nabla f + \partial g)x^\star \\
-\nabla f(x^\star) &\in \partial g(x^\star) \\
(\mathrm{Id} - \nabla f)x^\star &\in (\mathrm{Id} + \partial g)x^\star \\
(\mathrm{Id} + \partial g)^{-1}(\mathrm{Id} - \nabla f)x^\star &= x^\star
\end{aligned}
$$

Forward-Backward Splitting Algorithm

$$x^{(k+1)} = \mathrm{prox}_{\gamma g}(x^{(k)} - \gamma \nabla f(x^{(k)})).$$

Proximal Splitting Algorithms

Primal algorithms

$F = f + g$ Forward-Backward (Lions and Mercier, 1979)

$$x^{(k+1)} = \text{prox}_{\gamma g}(x^{(k)} - \gamma \nabla f(x^{(k)})).$$

$F = g + h$, g and h are simple

Proximal Splitting Algorithms

Primal algorithms

$F = f + g$ **Forward-Backward** (Lions and Mercier, 1979)

$$x^{(k+1)} = \mathrm{prox}_{\gamma g}(x^{(k)} - \gamma \nabla f(x^{(k)})).$$

$F = g + h$, g and h are simple $\quad \mathrm{rprox} \overset{\mathrm{def}}{=} 2\,\mathrm{prox} - \mathrm{Id}$

$F = g + h$ **Douglas–Rachford Splitting Algorithm**

$$y^{(k+1)} = \tfrac{1}{2}\,\mathrm{rprox}_{\gamma g}(\mathrm{rprox}_{\gamma h}(y^{(k)})) + \tfrac{1}{2} y^{(k)}; \qquad x^{(k+1)} = \mathrm{prox}_{\gamma h}(y^{(k+1)})$$

Proximal Splitting Algorithms

Primal algorithms

$F = f + g$ Forward-Backward (Lions and Mercier, 1979)

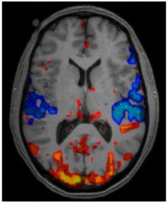
$$x^{(k+1)} = \text{prox}_{\gamma g}(x^{(k)} - \gamma \nabla f(x^{(k)})).$$

$F = g + h$ Douglas–Rachford (Lions and Mercier, 1979)

$$y^{(k+1)} = \tfrac{1}{2}\,\text{rprox}_{\gamma g}(\text{rprox}_{\gamma h}(y^{(k)})) + \tfrac{1}{2}y^{(k)}; \qquad x^{(k+1)} = \text{prox}_{\gamma h}(y^{(k+1)})$$

$\sum_b \|w_b\|$

$\sum_b \|Dw_b\|$



Proximal Splitting Algorithms

Primal algorithms

$F = f + g$ **Forward-Backward** (Lions and Mercier, 1979)

$$x^{(k+1)} = \text{prox}_{\gamma g}(x^{(k)} - \gamma \nabla f(x^{(k)})).$$

$F = g + h$ **Douglas–Rachford** (Lions and Mercier, 1979)

$$y^{(k+1)} = \frac{1}{2}\,\text{rprox}_{\gamma g}(\text{rprox}_{\gamma h}(y^{(k)})) + \frac{1}{2}y^{(k)}; \qquad x^{(k+1)} = \text{prox}_{\gamma h}(y^{(k+1)})$$

$F = \sum_i g_i$, each g_i is simple

$\min_x F(x) = \min_{x_i} \sum_i g_i(x_i)$ subject to $\forall i, j,\ x_i = x_j$

$\min_x F(x) = \min_{\boldsymbol{x}}\ \boldsymbol{g} + \boldsymbol{\iota_V}$

Proximal Splitting Algorithms

Primal algorithms

$F = f + g$ **Forward-Backward** (Lions and Mercier, 1979)

$$x^{(k+1)} = \text{prox}_{\gamma g}(x^{(k)} - \gamma \nabla f(x^{(k)})).$$

$F = g + h$ **Douglas–Rachford** (Lions and Mercier, 1979)

$$y^{(k+1)} = \tfrac{1}{2}\,\text{rprox}_{\gamma g}(\text{rprox}_{\gamma h}(y^{(k)})) + \tfrac{1}{2} y^{(k)}; \qquad x^{(k+1)} = \text{prox}_{\gamma h}(y^{(k+1)})$$

$F = \sum_i g_i$ **D.–R. on Product Space** (Spingarn, 1983)

$$\forall i,\ y_i^{(k+1)} = y_i^{(k)} + \text{prox}_{\frac{\gamma}{w_i} g_i}(2x^{(k)} - y_i^{(k)}) - x^{(k)}; \quad x^{(k+1)} = \sum_i w_i y_i^{(k+1)}$$

Proximal Splitting Algorithms

Primal algorithms

$F = f + g$ **Forward-Backward** (Lions and Mercier, 1979)

$$x^{(k+1)} = \operatorname{prox}_{\gamma g}(x^{(k)} - \gamma \nabla f(x^{(k)})).$$

$F = g + h$ **Douglas–Rachford** (Lions and Mercier, 1979)

$$y^{(k+1)} = \tfrac{1}{2}\operatorname{rprox}_{\gamma g}(\operatorname{rprox}_{\gamma h}(y^{(k)})) + \tfrac{1}{2}y^{(k)}; \qquad x^{(k+1)} = \operatorname{prox}_{\gamma h}(y^{(k+1)})$$

$F = \sum_i g_i$ **D.–R. on Product Space** (Spingarn, 1983)

$$\forall i,\ y_i^{(k+1)} = y_i^{(k)} + \operatorname{prox}_{\frac{\gamma}{w_i} g_i}(2x^{(k)} - y_i^{(k)}) - x^{(k)}; \quad x^{(k+1)} = \textstyle\sum_i w_i y_i^{(k+1)}$$

$F = f + \sum_i g_i$, f is smooth, each g_i is simple

Proximal Splitting Algorithms

Primal algorithms

$F = f + g$ **Forward-Backward** (Lions and Mercier, 1979)

$$x^{(k+1)} = \mathrm{prox}_{\gamma g}(x^{(k)} - \gamma \nabla f(x^{(k)})).$$

$F = g + h$ **Douglas–Rachford** (Lions and Mercier, 1979)

$$y^{(k+1)} = \tfrac{1}{2}\,\mathrm{rprox}_{\gamma g}(\mathrm{rprox}_{\gamma h}(y^{(k)})) + \tfrac{1}{2} y^{(k)}; \qquad x^{(k+1)} = \mathrm{prox}_{\gamma h}(y^{(k+1)})$$

$F = \sum_i g_i$ **D.–R. on Product Space** (Spingarn, 1983)

$$\forall i,\ y_i^{(k+1)} = y_i^{(k)} + \mathrm{prox}_{\frac{\gamma}{w_i} g_i}(2x^{(k)} - y_i^{(k)}) - x^{(k)}; \quad x^{(k+1)} = \textstyle\sum_i w_i y_i^{(k+1)}$$

$F = f + \sum_i g_i$ **Generalized F.-B.** (Raguet et al., 2013)

$$\forall i,\ y_i^{(k+1)} = y_i^{(k)} + \mathrm{prox}_{\frac{\gamma}{w_i} g_i}(2x^{(k)} - y_i^{(k)} - \gamma \nabla f(x^{(k)})) - x^{(k)};$$

$$x^{(k+1)} = \textstyle\sum_i w_i y_i^{(k+1)}$$

Proximal Splitting Algorithms

Primal algorithms

$F = f + g$ Forward-Backward (Lions and Mercier, 1979)

$F = g + h$ Douglas–Rachford (Lions and Mercier, 1979)

$F = \sum_i g_i$ D.–R. on Product Space (Spingarn, 1983)

$F = f + \sum_i g_i$ Generalized F.-B. (Raguet et al., 2013)

what about $g \circ L$, g simple, L bounded linear operator?

Proximal Splitting Algorithms

Primal algorithms

$F = f + g$ Forward-Backward (Lions and Mercier, 1979)

$F = g + h$ Douglas–Rachford (Lions and Mercier, 1979)

$F = \sum_i g_i$ D.–R. on Product Space (Spingarn, 1983)

$F = f + \sum_i g_i$ Generalized F.-B. (Raguet et al., 2013)

what about $g \circ L$, g simple, L bounded linear operator?

''tight frame''

$$\forall y \in \operatorname{ran} L,\ LL^{*}y = \nu y$$

Proximal Splitting Algorithms

Primal algorithms

$F = f + g$ Forward-Backward (Lions and Mercier, 1979)

$F = g + h$ Douglas–Rachford (Lions and Mercier, 1979)

$F = \sum_i g_i$ D.–R. on Product Space (Spingarn, 1983)

$F = f + \sum_i g_i$ Generalized F.-B. (Raguet et al., 2013)

what about $g \circ L$, g simple, L bounded linear operator?

''tight frame''

$$\operatorname{prox}_{g \circ L}(x) = x + \tfrac{1}{\nu} L^* \left(\operatorname{prox}_{\nu g} - \operatorname{Id} \right) L x$$

Proximal Splitting Algorithms

Primal algorithms

$F = f + g$ Forward-Backward (Lions and Mercier, 1979)

$F = g + h$ Douglas–Rachford (Lions and Mercier, 1979)

$F = \sum_i g_i$ D.–R. on Product Space (Spingarn, 1983)

$F = f + \sum_i g_i$ Generalized F.-B. (Raguet et al., 2013)

what about $g \circ L$, g simple, L bounded linear operator?

''tight frame'' ''split''

$$g \circ L = \sum_i g_i \circ L_i, \quad g_i \text{ simple, } L_i \text{ tight frame}$$

Proximal Splitting Algorithms

Primal algorithms

$F = f + g$ Forward-Backward (Lions and Mercier, 1979)

$F = g + h$ Douglas–Rachford (Lions and Mercier, 1979)

$F = \sum_i g_i$ D.–R. on Product Space (Spingarn, 1983)

$F = f + \sum_i g_i$ Generalized F.-B. (Raguet et al., 2013)

what about $g \circ L$, g simple, L bounded linear operator?

''tight frame'' ''split'' ''augment space''

$$\min_x g(Lx) = \min_{x,y} g(y) \text{ subject to } Lx = y$$

Proximal Splitting Algorithms

Primal algorithms

$F = f + g$ Forward-Backward (Lions and Mercier, 1979)

$F = g + h$ Douglas–Rachford (Lions and Mercier, 1979)

$F = \sum_i g_i$ D.–R. on Product Space (Spingarn, 1983)

$F = f + \sum_i g_i$ Generalized F.-B. (Raguet et al., 2013)

what about $g \circ L$, g simple, L bounded linear operator?

''tight frame'' ''split'' ''augment space''

$$\min_x g(Lx) = \min_{x,y} g(y) + \iota_{\{(x,y)\,|\,Lx=y\}}(x, y)$$

Proximal Splitting Algorithms

Primal algorithms

$F = f + g$ Forward-Backward (Lions and Mercier, 1979)

$F = g + h$ Douglas–Rachford (Lions and Mercier, 1979)

$F = \sum_i g_i$ D.–R. on Product Space (Spingarn, 1983)

$F = f + \sum_i g_i$ Generalized F.-B. (Raguet et al., 2013)

what about $g \circ L$, g simple, L bounded linear operator?

"tight frame" "split" "augment space"

$\operatorname{proj}_{\{(x,y) \mid Lx=y\}}$ involves $(\mathrm{Id} + L^*L)^{-1}$ or $(\mathrm{Id} + LL^*)^{-1}$

Proximal Splitting Algorithms

Primal algorithms

$F = f + g$ Forward-Backward (Lions and Mercier, 1979)

$F = g + h$ Douglas–Rachford (Lions and Mercier, 1979)

$F = \sum_i g_i$ D.–R. on Product Space (Spingarn, 1983)

$F = f + \sum_i g_i$ Generalized F.-B. (Raguet et al., 2013)

what about $g \circ L$, g simple, L bounded linear operator?

''tight frame'' ''split'' ''augment space''

otherwise: primal-dual algorithm

Proximal Splitting Algorithms

Primal-dual algorithms

Canonical form: $F = g \circ L + h$, g, h simple, L linear operator
Split as $\min_{x,y} g(y) + h(x)$ subject to $y = Lx$

Proximal Splitting Algorithms

Primal-dual algorithms

Canonical form: $F = g \circ L + h$, g, h simple, L linear operator

Split as $\min_{x,y} g(y) + h(x)$ subject to $y = Lx$

Alternating-Direction Method of Multipliers? (Gabay and Mercier, 1976)

$$x^{(k+1)} = \arg\min_x \frac{1}{\rho} h(x) + \frac{\rho}{2} \| Lx - (\rho y^{(k)} - \lambda^{(k)}) \|^2$$

$$y^{(k+1)} = \arg\min_y \frac{1}{\rho} g(y) + \frac{1}{2} \| y - (\rho L x^{(k)} + \lambda^{(k)}) \|^2$$

$$\lambda^{(k+1)} = \lambda^{(k)} + \rho \left(L x^{(k+1)} - y^{(k+1)} \right)$$

Proximal Splitting Algorithms

Primal-dual algorithms

Canonical form: $F = g \circ L + h$, $\quad g, h$ simple, $\quad L$ linear operator
Split as $\quad \min_{x,y} g(y) + h(x) \quad$ subject to $y = Lx$

Alternating-Direction Method of Multipliers? (Gabay and Mercier, 1976)

$$x^{(k+1)} = \arg\min_x \tfrac{1}{\rho} h(x) + \tfrac{\rho}{2} \|Lx - (\rho y^{(k)} - \lambda^{(k)})\|^2$$
$$y^{(k+1)} = \arg\min_y \tfrac{1}{\rho} g(y) + \tfrac{1}{2} \|y - (\rho L x^{(k)} + \lambda^{(k)})\|^2$$
$$\lambda^{(k+1)} = \lambda^{(k)} + \rho\big(Lx^{(k+1)} - y^{(k+1)}\big)$$

- update on x
  - well defined only for L injective

Proximal Splitting Algorithms

Primal-dual algorithms

Canonical form: $F = g \circ L + h$, g, h simple, L linear operator

Split as $\min_{x,y} g(y) + h(x)$ subject to $y = Lx$

Alternating-Direction Method of Multipliers? (Gabay and Mercier, 1976)

$$x^{(k+1)} = \arg\min_x \tfrac{1}{\rho} h(x) + \tfrac{\rho}{2} \| Lx - (\rho y^{(k)} - \lambda^{(k)}) \|^2$$
$$y^{(k+1)} = \arg\min_y \tfrac{1}{\rho} g(y) + \tfrac{1}{2} \| y - (\rho L x^{(k)} + \lambda^{(k)}) \|^2$$
$$\lambda^{(k+1)} = \lambda^{(k)} + \rho \left(L x^{(k+1)} - y^{(k+1)} \right)$$

- update on x
  - well defined only for L injective
  - more complicated than $\mathrm{prox}_{\frac{1}{\rho} h}$

Proximal Splitting Algorithms

Primal-dual algorithms

Canonical form: $F = g \circ L + h$, $\quad g, h$ simple, $\quad L$ linear operator

Split as $\quad \min_{x,y} g(y) + h(x) \quad$ subject to $y = Lx$

Alternating-Direction Method of Multipliers? (Gabay and Mercier, 1976)

$$x^{(k+1)} = \arg\min_x \tfrac{1}{\rho} h(x) + \tfrac{\rho}{2} \|Lx - (\rho y^{(k)} - \lambda^{(k)})\|^2$$
$$y^{(k+1)} = \arg\min_y \tfrac{1}{\rho} g(y) + \tfrac{1}{2} \|y - (\rho L x^{(k)} + \lambda^{(k)})\|^2$$
$$\lambda^{(k+1)} = \lambda^{(k)} + \rho\big(Lx^{(k+1)} - y^{(k+1)}\big)$$

- update on x
  - well defined only for L injective
  - more complicated than $\mathrm{prox}_{\frac{1}{\rho}h}$
- require storing both y and λ

Proximal Splitting Algorithms

Primal-dual algorithms

Canonical form: $F = g \circ L + h$, g, h simple, L linear operator

Split as $\min_{x,y} g(y) + h(x)$ subject to $y = Lx$

ADMM? (Gabay and Mercier, 1976)

$F = g \circ L + h$ Primal-Dual of Chambolle and Pock (2011)

or more generally, $F = \sum_i g_i \circ L_i$

Proximal Splitting Algorithms

Primal-dual algorithms

Canonical form: $F = g \circ L + h$, g, h simple, L linear operator
Split as $\min_{x,y} g(y) + h(x)$ subject to $y = Lx$

ADMM? (Gabay and Mercier, 1976)

$F = g \circ L + h$ Primal-Dual of Chambolle and Pock (2011)
or more generally, $F = \sum_i g_i \circ L_i$

And if f is smooth but not simple?

Proximal Splitting Algorithms

Primal-dual algorithms

Canonical form: $F = g \circ L + h$, $\quad g, h$ simple, $\quad L$ linear operator

Split as $\quad \min_{x,y} g(y) + h(x) \quad$ subject to $y = Lx$

ADMM? (Gabay and Mercier, 1976)

$F = g \circ L + h$ Primal-Dual of Chambolle and Pock (2011)

or more generally, $\quad F = \sum_i g_i \circ L_i$

And if f is smooth but not simple?

$F = f + g \circ L + h$ Primal-Dual of Condat (2013); Vũ (2013)

or more generally, $\quad F = f + \sum_i g_i \circ L_i$

Proximal Splitting Algorithms

Summary

$F = f + g$ **Forward-Backward** (Lions and Mercier, 1979)

a.k.a proximal gradient algorithm

$F = g + h$ **Douglas–Rachford** (Lions and Mercier, 1979)

$F = \sum_i g_i$ **D.–R. on Product Space** (Spingarn, 1983)

a.k.a Parallel Proximal Algorithm

$F = f + \sum_i g_i$ **Generalized F.-B.** (Raguet et al., 2013)

a.k.a Forward-Douglas–Rachford

$F = g \circ L + h$ **Primal-Dual of** Chambolle and Pock (2011)

a.k.a Primal-Dual Hybrid Gradient

$F = f + g \circ L + h$ **Primal-Dual of** Condat (2013); Vũ (2013)

a.k.a Forward-Backward Primal-Dual

Proximal Splitting Algorithms

Overrelaxation and Inertial Forces

All Methods

- $y^{(k+1)} = Tx^{(k)}$
- $x^{(k+1)} = y^{(k+1)} + \alpha_k(y^{(k+1)} - y^{(k)})$

Acceleration observed in practice (Iutzeler and Hendrickx, 2018)

$F = f + g$ Forward-Backward

Theoretical acceleration on functional values $F(x^{(k)}) - F(x^\star)$ (Beck and Teboulle, 2009)

Proximal Splitting Algorithms

Metric Conditioning

$F = f + g$ Forward-Backward

Variable metric forward-backward (Chen and Rockafellar, 1997)

Quasi-Newton forward-backward (Becker and Fadili, 2012)

$F = f + \sum_i g_i$ Generalized Forward-Backward

$$\forall i,\ y_i^{(k+1)} = y_i^{(k)} + \operatorname{prox}_{\frac{\gamma}{w_i} g_i}\left(2x^{(k)} - y_i^{(k)} - \gamma \nabla f(x^{(k)})\right) - x^{(k)};$$

$$x^{(k+1)} = \sum_i w_i y_i^{(k+1)}$$

Proximal Splitting Algorithms

Metric Conditioning

$F = f + g$ Forward-Backward

Variable metric forward-backward (Chen and Rockafellar, 1997)
Quasi-Newton forward-backward (Becker and Fadili, 2012)

$F = f + \sum_i g_i$ Generalized Forward-Backward

$\forall i,\ y_i^{(k+1)} = y_i^{(k)} + \operatorname{prox}_{g_i}^{\Gamma^{-1}W_i}\left(2x^{(k)} - y_i^{(k)} - \Gamma\nabla f(x^{(k)})\right) - x^{(k)};$
$x^{(k+1)} = \sum_i W_i y_i^{(k+1)}$

- Γ approximate "$(\nabla^2 F^{-1})$"
- $\sum_i W_i = \mathrm{Id}$, but W_i might be only semidefinite
- $\operatorname{prox}_{g_i}^{\Gamma^{-1}W_i}$ might be computable when $\operatorname{prox}_{g_i}$ is not

Proximal Splitting Algorithms

Metric Conditioning

$F = f + g$ Forward-Backward

Variable metric forward-backward (Chen and Rockafellar, 1997)
Quasi-Newton forward-backward (Becker and Fadili, 2012)

$F = f + \sum_i g_i$ Generalized Forward-Backward

$$\forall i,\ y_i^{(k+1)} = y_i^{(k)} + \mathrm{prox}_{g_i}^{\Gamma^{-1} W_i}\big(2x^{(k)} - y_i^{(k)} - \Gamma \nabla f(x^{(k)})\big) - x^{(k)};$$

$$x^{(k+1)} = \textstyle\sum_i W_i y_i^{(k+1)}$$

(Raguet and Landrieu, 2015)

$F = g \circ L + h$ Primal-Dual Hybrid Gradient

Preconditioning on L (Pock and Chambolle, 2011)

$F = f + g \circ L + h$ Forward-Backward Primal-Dual

Preconditioning on both L and "$\nabla^2 f$" (Lorenz and Pock, 2015)

Proximal Splitting Algorithms

Stochastic and distributed versions

Douglas–Rachford and ADMM

Seminal work of Iutzeler et al. (2013)

All Methods

Fall within the scope of stochastic fixed point algorithms (Combettes and Pesquet, 2015)

Special case of Forward-Douglas–Rachford

Replace ∇f by a random variable G
Typical convergence conditions:

- $\mathrm{E}\left[G^{(k)} \,\middle|\, X^{(1)}, \ldots, X^{(k)}\right] = \nabla f(X^{(n)})$ a.s.
- $\sum_k \mathrm{E}\left[\left\|G^{(k)} - \nabla f(X^{(n)})\right\|^2 \,\middle|\, X^{(1)}, \ldots, X^{(k)}\right] < +\infty$ a.s.

(Cevher et al., 2016)

Proximal Splitting Algorithms

Nonconvex cases

$F = f + g$ Forward-Backward

Any function nonconvex (Attouch et al., 2013)
f smooth, g convex (Ochs et al., 2014; Chouzenoux et al., 2014)

$F = g \circ L + h$ Primal-Dual Hybrid Gradient

g semiconvex, h strongly convex (Möllenhoff et al., 2015)
h smooth, L surjective (with ADMM, Li and Pong, 2015)

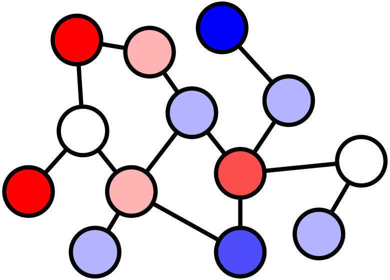
But actually my classification of proximal algorithms is not anymore relevant in absence of convexity

Cut-pursuit Algorithm

Enhancing proximal algorithm with combinatorial optimization

$G = (V, E)$ $\qquad F\colon (x_v)_{v \in V} \mapsto f(x) + \sum_{v \in V} g_v(x_v) + \sum_{(u,v) \in E} w_{(u,v)} |x_u - x_v|$

f smooth; $\quad g$ separable

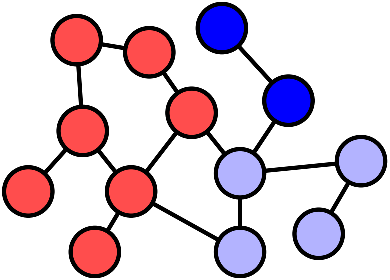


Cut-pursuit Algorithm

Enhancing proximal algorithm with combinatorial optimization

$G = (V, E)$ $\quad F\colon (x_v)_{v \in V} \mapsto f(x) + \sum_{v \in V} g_v(x_v) + \sum_{(u,v) \in E} w_{(u,v)} |x_u - x_v|$

f smooth; $\quad g$ separable

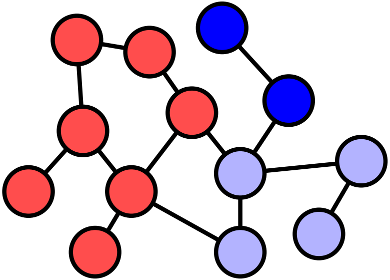


Cut-pursuit Algorithm

Enhancing proximal algorithm with combinatorial optimization

$G = (V, E) \qquad F\colon (x_v)_{v\in V} \mapsto f(x) + \sum_{v\in V} g_v(x_v) + \sum_{(u,v)\in E} w_{(u,v)} |x_u - x_v|$

f smooth; g separable



Typical proximal algorithm:

- GFB (preconditioning)
- PDHG (if prox_f available)
- PDFB (use ∇f)

Visit the entire graph at each iteration!

Cut-pursuit Algorithm

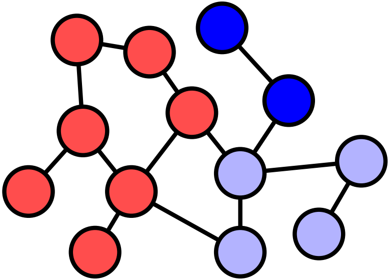
Enhancing proximal algorithm with combinatorial optimization

$$G = (V, E) \qquad F\colon (x_v)_{v\in V} \mapsto f(x) + \sum_{v\in V} g_v(x_v) + \sum_{(u,v)\in E} w_{(u,v)} |x_u - x_v|$$

f smooth; g separable

Typical proximal algorithm:

- GFB (preconditioning)
- PDHG (if prox_f available)
- PDFB (use ∇f)



Visit the entire graph at each iteration!

Use the fact that the solution has few constant components:

- block coordinate
- ''working set'' (Landrieu and Obozinski, 2017)

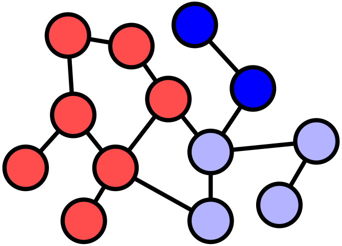
Cut-pursuit

Working set approach

$G = (V, E)$ $\quad F\colon (x_v)_{v \in V} \mapsto f(x) + \sum_{v \in V} g_v(x_v) + \sum_{(u,v) \in E} w_{(u,v)} |x_u - x_v|$

f smooth; $\quad g$ separable

$\mathcal{V}$ partition of V; $\quad x = \sum_{U \in \mathcal{V}} \xi_U 1_U$



Cut-pursuit

Working set approach

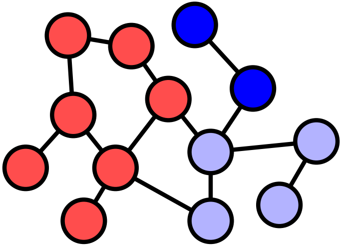
$$G = (V, E) \qquad F\colon (x_v)_{v \in V} \mapsto f(x) + \sum_{v \in V} g_v(x_v) + \sum_{(u,v) \in E} w_{(u,v)} |x_u - x_v|$$

f smooth; g separable

$\mathcal{V}$ partition of V; $x = \sum_{U \in \mathcal{V}} \xi_U 1_U$

$F^{(\mathcal{V})}\colon (\xi_U)_{U \in \mathcal{V}} \mapsto F(\sum_{U \in \mathcal{V}} \xi_U 1_U)$

$$= f\Big(\sum_{U \in \mathcal{V}} \xi_U 1_U\Big) + \sum_{U \in \mathcal{V}} \sum_{v \in U} g_v(\xi_U) + \sum_{(U,U') \in \mathcal{E}} \sum_{(u,v) \in E \cap U \times U'} w_{(u,v)} |\xi_U - \xi'_U|$$



Cut-pursuit

Working set approach

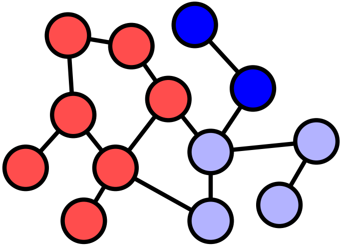
$G = (V, E) \qquad F\colon (x_v)_{v\in V} \mapsto f(x) + \sum_{v\in V} g_v(x_v) + \sum_{(u,v)\in E} w_{(u,v)}|x_u - x_v|$

f smooth; $\quad g$ separable

$\mathcal{V}$ partition of V; $\quad x = \sum_{U\in\mathcal{V}} \xi_U 1_U$

$F^{(\mathcal{V})}\colon (\xi_U)_{U\in\mathcal{V}} \mapsto F(\sum_{U\in\mathcal{V}} \xi_U 1_U)$

$= f\Big(\sum_{U\in\mathcal{V}} \xi_U 1_U\Big) + \sum_{U\in\mathcal{V}} \sum_{v\in U} g_v(\xi_U) + \sum_{(U,U')\in\mathcal{E}} \sum_{(u,v)\in E\cap U\times U'} w_{(u,v)}|\xi_U - \xi'_U|$



Cut-pursuit

Working set approach

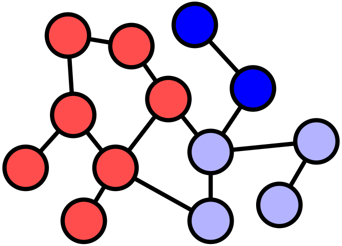
$G = (V, E) \qquad F\colon (x_v)_{v\in V} \mapsto f(x) + \sum_{v\in V} g_v(x_v) + \sum_{(u,v)\in E} w_{(u,v)} |x_u - x_v|$

f smooth; $\quad g$ separable

$\mathcal{V}$ partition of V; $\quad x = \sum_{U\in\mathcal{V}} \xi_U 1_U$

$F^{(\mathcal{V})}\colon (\xi_U)_{U\in\mathcal{V}} \mapsto F(\sum_{U\in\mathcal{V}} \xi_U 1_U)$

$= f\Big(\sum_{U\in\mathcal{V}} \xi_U 1_U\Big) + \sum_{U\in\mathcal{V}} \sum_{v\in U} g_v(\xi_U) + \sum_{(U,U')\in\mathcal{E}} \sum_{(u,v)\in E\cap U\times U'} w_{(u,v)} |\xi_U - \xi'_U|$



Cut-pursuit

Working set approach

$G = (V, E) \qquad F\colon (x_v)_{v\in V} \mapsto f(x) + \sum_{v\in V} g_v(x_v) + \sum_{(u,v)\in E} w_{(u,v)} |x_u - x_v|$

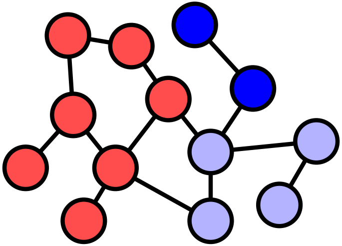
f smooth; g separable

$\mathcal{V}$ partition of V; $x = \sum_{U\in\mathcal{V}} \xi_U 1_U \qquad \mathcal{G} = (\mathcal{V}, \mathcal{E})$

$F^{(\mathcal{V})}\colon (\xi_U)_{U\in\mathcal{V}} \mapsto F(\sum_{U\in\mathcal{V}} \xi_U 1_U)$

$= f\Big(\sum_{U\in\mathcal{V}} \xi_U 1_U\Big) + \sum_{U\in\mathcal{V}} \sum_{v\in U} g_v(\xi_U) + \sum_{(U,U')\in\mathcal{E}} \sum_{(u,v)\in E\cap U\times U'} w_{(u,v)} |\xi_U - \xi'_U|$

find $\xi^{(\mathcal{V})} \in \arg\min F^{(\mathcal{V})}$
efficient with proximal algorithm
(if correctly conditioned)



Cut-pursuit

Working set approach

$G = (V, E) \qquad F\colon (x_v)_{v\in V} \mapsto f(x) + \sum_{v\in V} g_v(x_v) + \sum_{(u,v)\in E} w_{(u,v)} |x_u - x_v|$

f smooth; g separable

$\mathcal{V}$ partition of V; $\quad x = \sum_{U\in\mathcal{V}} \xi_U 1_U \qquad \mathcal{G} = (\mathcal{V}, \mathcal{E})$

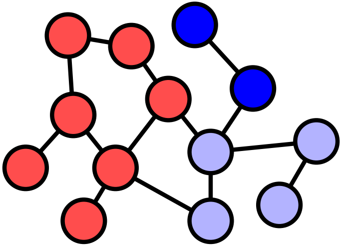
$F^{(\mathcal{V})}\colon (\xi_U)_{U\in\mathcal{V}} \mapsto F(\sum_{U\in\mathcal{V}} \xi_U 1_U)$

$= f\Big(\sum_{U\in\mathcal{V}} \xi_U 1_U\Big) + \sum_{U\in\mathcal{V}} \sum_{v\in U} g_v(\xi_U) + \sum_{(U,U')\in\mathcal{E}} \sum_{(u,v)\in E\cap U\times U'} w_{(u,v)} |\xi_U - \xi'_U|$

find $\xi^{(\mathcal{V})} \in \arg\min F^{(\mathcal{V})}$
efficient with proximal algorithm
(if correctly conditioned)

Algorithmic scheme:

1. solve reduced problem
2. refine partition $\mathcal{V}$



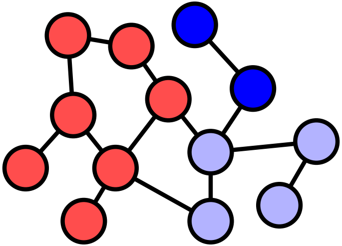
Cut-pursuit

Refining the partition

$$F : (x_v)_{v \in V} \mapsto \quad f(x) \quad + \sum_{v \in V} g_v(x_v) \quad + \sum_{(u,v) \in E} w_{(u,v)} |x_u - x_v|$$

$$F'(x, d)$$

Steepest descent direction? $\arg\min_{d \in \mathbb{R}^V} F'(x, d)$



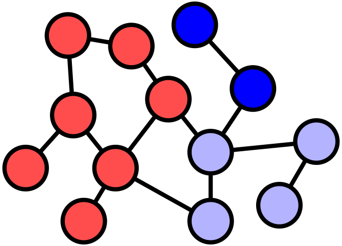
Cut-pursuit

Refining the partition

$$F : (x_v)_{v \in V} \mapsto \quad f(x) \quad + \sum_{v \in V} g_v(x_v) \quad + \sum_{(u,v) \in E} w_{(u,v)} |x_u - x_v|$$

$$F'(x, d) \qquad \nabla_v f(x) d_v$$

Steepest descent direction? $\arg\min_{d \in \mathbb{R}^V} F'(x, d)$



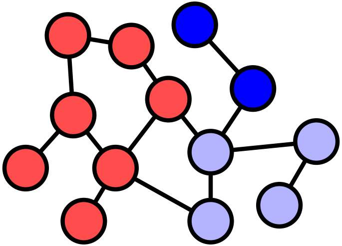
Cut-pursuit

Refining the partition

$$F : (x_v)_{v \in V} \mapsto \quad f(x) \quad + \sum_{v \in V} g_v(x_v) \quad + \sum_{(u,v) \in E} w_{(u,v)} |x_u - x_v|$$

$$F'(x, d) \qquad \nabla_v f(x) d_v \qquad \begin{matrix} g'_v(x_v, +1) d_v \\ g'_v(x_v, -1) d_v \end{matrix}$$

Steepest descent direction? $\arg\min_{d \in \mathbb{R}^V} F'(x, d)$



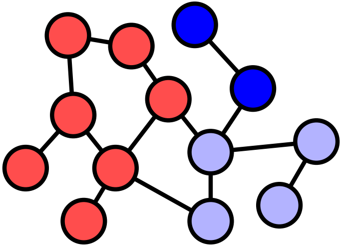
Cut-pursuit

Refining the partition

$$F : (x_v)_{v \in V} \mapsto \quad f(x) \quad + \textstyle\sum_{v \in V} g_v(x_v) \quad + \textstyle\sum_{(u,v) \in E} w_{(u,v)} |x_u - x_v|$$

$$F'(x,d) \qquad \nabla_v f(x) d_v \qquad \begin{matrix} g'_v(x_v, +1) d_v \\ g'_v(x_v, -1) d_v \end{matrix} \qquad \begin{matrix} w_{(u,v)} \operatorname{sign}(x_v - x_u) d_v \\ w_{(u,v)} |d_u - d_v| \end{matrix}$$

Steepest descent direction? $\underset{d \in \mathbb{R}^V}{\arg\min}\, F'(x,d)$



Cut-pursuit

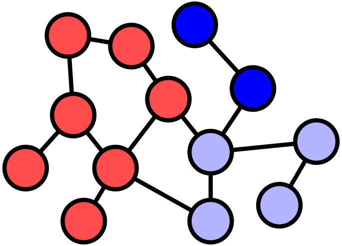
Refining the partition

$$F : (x_v)_{v \in V} \mapsto \quad f(x) \quad + \sum_{v \in V} g_v(x_v) \quad + \sum_{(u,v) \in E} w_{(u,v)} |x_u - x_v|$$

$$F'(x, d) \qquad \nabla_v f(x) d_v \qquad \begin{matrix} g'_v(x_v, +1) d_v \\ g'_v(x_v, -1) d_v \end{matrix} \qquad \begin{matrix} w_{(u,v)} \operatorname{sign}(x_v - x_u) d_v \\ w_{(u,v)} |d_u - d_v| \end{matrix}$$

Steepest descent direction? $\underset{d \in \mathbb{R}^V}{\arg\min} F'(x, d)$

$$\sum_{\substack{v \in V \\ d_v > 0}} \delta_v^+(x) d_v + \sum_{\substack{v \in V \\ d_v < 0}} \delta_v^-(x) d_v + \sum_{(u,v) \in E_=^{(x)}} w_{(u,v)} |d_u - d_v|$$



Cut-pursuit

Refining the partition

$$F : (x_v)_{v\in V} \mapsto \quad f(x) \quad + \sum_{v\in V} g_v(x_v) \quad + \sum_{(u,v)\in E} w_{(u,v)} |x_u - x_v|$$

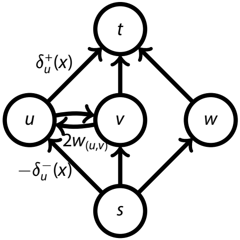
$$F'(x,d) \quad \nabla_v f(x) d_v \quad g'_v(x_v,+1)d_v \quad w_{(u,v)} \operatorname{sign}(x_v - x_u) d_v$$

$$g'_v(x_v,-1)d_v \quad w_{(u,v)} |d_u - d_v|$$

Steepest binary descent direction? $\underset{d\in\{-1,+1\}^V}{\arg\min}\ F'(x,d)$

$$\sum_{\substack{v\in V\\ d_v=+1}} \delta_v^+(x) - \sum_{\substack{v\in V\\ d_v=-1}} \delta_v^-(x) + \sum_{(u,v)\in E_=^{(x)}} w_{(u,v)} |d_u - d_v|$$

Can be solved by a minimal cut in an appropriate flow graph



Cut-pursuit

Refining the partition

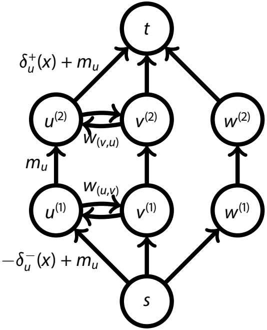
$$F\colon (x_v)_{v\in V} \mapsto \quad f(x) \quad + \textstyle\sum_{v\in V} g_v(x_v) \quad + \textstyle\sum_{(u,v)\in E} w_{(u,v)} |x_u - x_v|$$

$$F'(x,d) \quad \nabla_v f(x) d_v \quad \begin{matrix} g'_v(x_v,+1)d_v \\ g'_v(x_v,-1)d_v \end{matrix} \quad \begin{matrix} w_{(u,v)}\,\mathrm{sign}(x_v - x_u)d_v \\ w_{(u,v)}|d_u - d_v| \end{matrix}$$

Steepest ternary descent direction? $\underset{d\in\{-1,0,+1\}^V}{\arg\min}\ F'(x,d)$

$$\sum_{\substack{v\in V\\ d_v=+1}} \delta_v^+(x) - \sum_{\substack{v\in V\\ d_v=-1}} \delta_v^-(x) + \sum_{(u,v)\in E_=^{(x)}} w_{(u,v)}|d_u - d_v|$$

Can be solved by a minimal cut in an appropriate flow graph



Theorem: this set of descent directions is rich enough to ensure optimality

Cut-pursuit

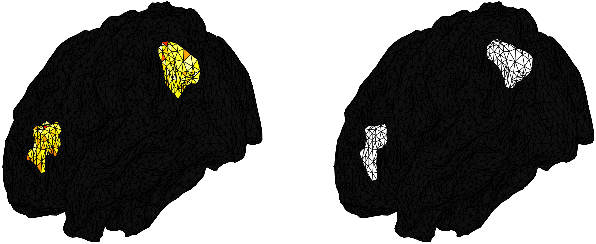
Preliminary results

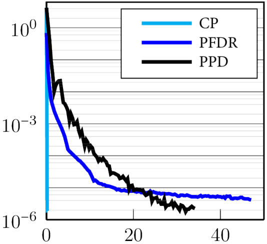
Brain source identification in electroencephalography

$$F\colon x \mapsto \frac{1}{2}\|y - \Phi x\|^2 + \sum_{v \in V} \big(\lambda_v |x_v| + \iota_{\mathbb{R}_+}(x_v)\big) + \sum_{(u,v) \in E} w_{(u,v)} |x_u - x_v|$$

$|V| = 19\,626$

$|E| = 29\,439$





Cut-pursuit

Preliminary results

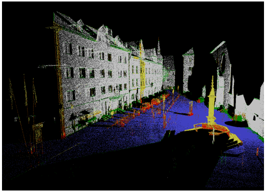
regularization of 3D point cloud classification

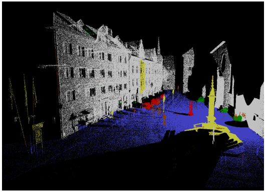
given probabilistic assignment $q \in \mathbb{R}^{V \times K}$

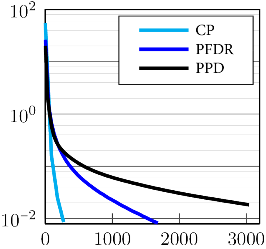
$$F\colon p \mapsto \sum_{v \in V} \mathrm{KL}^{(\beta)}(q_v, p_v) + \sum_{v \in V} \iota_{\triangle_K}(p_v) + \sum_{(u,v) \in E} w_{(u,v)} \|p_u - p_v\|_1$$

$|V| = 3\,000\,111$

$|E| = 17\,206\,938$







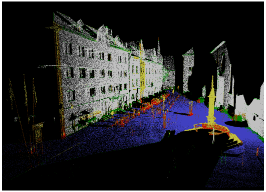
Cut-pursuit

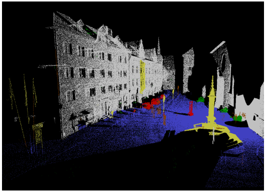
Preliminary results

regularization of 3D point cloud classification
given probabilistic assignment $q \in \mathbb{R}^{V \times K}$

$$F\colon p \mapsto \sum_{v \in V} \mathrm{KL}^{(\beta)}(q_v, p_v) + \sum_{v \in V} \iota_{\triangle_K}(p_v) + \sum_{(u,v) \in E} w_{(u,v)} \|p_u - p_v\|_1$$

$|V| = 3\,000\,111$
$|E| = 17\,206\,938$





Next: parallelize graph cuts along components in $\mathcal{V}$

- almost linear acceleration
- distributed optimization

Integration in ICAR team

Strengths

- continuous methods
- regularization techniques
- convex optimization

Weaknesses

- not (yet) an expert in (deep) learning
- not familiar with ''discrete formulations''

Research interest

- registration and inverse problems for medical imaging
- high-resolution satellite image segmentation
- dependence measures for identifying functional relationship between data with statistical tools

References I

Attouch, H., Bolte, J., and Svaiter, B. F. (2013). Convergence of descent methods for semi-algebraic and tame problems: proximal algorithms, forward-backward splitting, and regularized Gauss–Seidel methods. *Mathematical Programming*, 137(1-2):91–129.

Beck, A. and Teboulle, M. (2009). A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202.

Becker, S. and Fadili, J. (2012). A quasi-Newton proximal splitting method. In *Advances in Neural Information Processing Systems*, pages 2627–2635.

Cevher, V., Vũ, B. C., and Yurtsever, A. (2016). Stochastic forward-Douglas–Rachford splitting for monotone inclusions. Technical report, EPFL.

References II

Chambolle, A. and Pock, T. (2011). A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of Mathematical Imaging and Vision*, 40(1):120–145.

Chen, G. H.-G. and Rockafellar, R. T. (1997). Convergence rates in forward-backward splitting. *SIAM Journal on Optimization*, 7(2):421–444.

Chouzenoux, E., Pesquet, J.-C., and Repetti, A. (2014). Variable metric forward-backward algorithm for minimizing the sum of a differentiable function and a convex function. *Journal of Optimization Theory and Applications*, 162(1):107–132.

Combettes, P. L. and Pesquet, J.-C. (2015). Stochastic quasi-fejér block-coordinate fixed point iterations with random sweeping. *SIAM Journal of Optimization*, 25:1221–1248.

References III

Condat, L. (2013). A primal-dual splitting method for convex optimization involving Lipschitzian, proximable and linear composite terms. *Journal of Optimization Theory and Applications*, 158(2):460–479.

Gabay, D. and Mercier, B. (1976). A dual algorithm for the solution of nonlinear variational problems via finite element approximation. *Computers & Mathematics with Applications*, 2(1):17–40.

Iutzeler, F., Bianchi, P., and Hachem, W. (2013). Asynchronous distributed optimization using a randomized alternating direction method of multipliers. In *IEEE Conference on Decision and Control*.

Iutzeler, F. and Hendrickx, J. M. (2018). A generic online acceleration scheme for optimization algorithms via relaxation and inertia.

References IV

Landrieu, L. and Obozinski, G. (2017). Cut pursuit: Fast algorithms to learn piecewise constant functions on general weighted graphs. *SIAM Journal on Imaging Sciences*, 10(4):1724–1766.

Li, G. and Pong, T. K. (2015). Global convergence of splitting methods for nonconvex composite optimization. *SIAM Journal on Optimization*, 25(4):2434–2460.

Lions, P.-L. and Mercier, B. (1979). Splitting algorithms for the sum of two nonlinear operators. *SIAM Journal on Numerical Analysis*, 16(6):964–979.

Lorenz, D. A. and Pock, T. (2015). An inertial forward-backward algorithm for monotone inclusions. *Journal of Mathematical Imaging and Vision*, 51(2):311–325.

References V

Möllenhoff, T., Strekalovskiy, E., Moeller, M., and Cremers, D. (2015). The primal-dual hybrid gradient method for semiconvex splittings. *SIAM Journal on Imaging Sciences*, 8(2):827–857.

Ochs, P., Chen, Y., Brox, T., and Pock, T. (2014). iPiano: Inertial proximal algorithm for nonconvex optimization. *SIAM Journal on Imaging Sciences*, 7(2):1388–1419.

Pock, T. and Chambolle, A. (2011). Diagonal preconditioning for first order primal-dual algorithms in convex optimization. In *IEEE International Conference on Computer Vision*, pages 1762–1769. IEEE.

Raguet, H., Fadili, J., and Peyré, G. (2013). A generalized forward-backward splitting. *SIAM Journal on Imaging Sciences*, 6(3):1199–1226.

References VI

Raguet, H. and Landrieu, L. (2015). Preconditioning of a generalized forward-backward splitting and application to optimization on graphs. *SIAM Journal on Imaging Sciences*, 8(4):2706–2739.

Spingarn, J. E. (1983). Partial inverse of a monotone operator. *Applied Mathematics and Optimization*, 10(1):247–265.

Vũ, B. C. (2013). A splitting algorithm for dual monotone inclusions involving cocoercive operators. *Advances in Computational Mathematics*, 38(3):667–681.