A Parallel Block-Coordinate Approach for Primal-Dual Splitting with Arbitrary Random Block Selection

Jean-Christophe Pesquet

Laboratoire d'Informatique Gaspard Monge - CNRS
Univ. Paris-Est Marne-la-Vallée, France

EUSIPCO 2015 – Nice

UPEM UNIVERSITÉ PARIS-EST MARNE-LA-VALLÉE

GASPARD - MONGE LABORATOIRE D'INFORMATIQUE

In collaboration with

A. Repetti

E. Chouzenoux

Variational formulation

OBJECTIVE FUNCTION:

Find a solution to the convex optimization problem

$$\underset{\mathbf{x}\in\mathsf{H}}{\text{minimize}} \ \Phi(\mathbf{x})$$

where

- H: signal space (separable real Hilbert space),
- $\Phi \in \Gamma_0(\mathsf{H})$: class of convex lower-semicontinuous functions from H to $]-\infty, +\infty]$ with a nonempty domain.

Variational formulation

OBJECTIVE FUNCTION:

Find a solution to the convex optimization problem

$$\underset{\mathbf{x}\in \mathsf{H}}{\text{minimize}} \;\; \Phi(\mathbf{x})$$

where

- H: signal space (separable real Hilbert space),
- $\Phi \in \Gamma_0(\mathsf{H})$: class of convex lower-semicontinuous functions from H to $]-\infty, +\infty]$ with a nonempty domain.

In the context of large scale problems, how to find an optimization algorithm able to deliver a reliable numerical solution in a reasonable time, with low memory requirement?

Fundamental Tools in Convex Analysis

The inf-convolution of $f: H \to]-\infty, +\infty]$ and $g: H \to]-\infty, +\infty]$ is

$$f \,\square\, g: H \to [-\infty, +\infty]: x \mapsto \inf_{y \in H} f(y) + g(x-y).$$

PARTICULAR CASE: $f \,\square\, \iota_{\{0\}} = f$,

where, for $C \subset H$,

$$(\forall x \in H) \quad \iota_C(x) = \begin{cases} 0 & \text{if } x \in C, \\ +\infty & \text{otherwise.} \end{cases}$$

Fundamental Tools in Convex Analysis

The **inf-convolution** of $f\colon \mathcal{H} \to]-\infty, +\infty]$ and $g\colon \mathcal{H} \to]-\infty, +\infty]$ is

$$f \,\square\, g\colon \mathcal{H} \to [-\infty, +\infty]\colon x \mapsto \inf_{y \in \mathcal{H}} f(y) + g(x - y).$$

The **conjugate** of $f\colon \mathcal{H} \to]-\infty, +\infty]$ is $f^*\colon \mathcal{H} \to [-\infty, +\infty]$ such that

$$(\forall u \in \mathcal{H}) \qquad f^*(u) = \sup_{x \in \mathcal{H}} \big(\langle x \mid u \rangle - f(x) \big).$$

Fundamental Tools in Convex Analysis

The **inf-convolution** of $f\colon \mathcal{H} \to]-\infty,+\infty]$ and $g\colon \mathcal{H} \to]-\infty,+\infty]$ is

$$f \,\square\, g\colon \mathcal{H} \to [-\infty,+\infty]\colon x \mapsto \inf_{y\in\mathcal{H}} f(y) + g(x-y).$$

The **conjugate** of $f\colon \mathcal{H} \to]-\infty,+\infty]$ is $f^*\colon \mathcal{H} \to [-\infty,+\infty]$ such that

$$(\forall u \in \mathcal{H}) \qquad f^*(u) = \sup_{x\in\mathcal{H}} \big(\langle x \mid u\rangle - f(x) \big).$$

Let $f \in \Gamma_0(\mathcal{H})$. Let $U\colon \mathcal{H} \to \mathcal{H}$ be a strongly positive self-adjoint linear operator.
The **proximity operator** $\mathrm{prox}_f^U(x)$ of f at $x \in \mathcal{H}$ relative to the metric induced by U is the unique vector $\widehat{y} \in \mathcal{H}$ such that

$$f(\widehat{y}) + \frac{1}{2}\|\widehat{y} - x\|_U^2 = \inf_{y\in\mathcal{H}} f(y) + \frac{1}{2}\langle y - x \mid U(y-x)\rangle.$$

Parallel proximal primal-dual problem

PRIMAL PROBLEM

We want to $\underset{\mathsf{x}\in\mathsf{H}}{\text{minimize}} \;\; \mathsf{h}(\mathsf{x}) + \sum_{k=1}^{q} (\mathsf{g}_k \,\Box\, \mathsf{l}_k)(\mathsf{L}_k \mathsf{x}).$

DUAL PROBLEM

We want to $\underset{\mathsf{v}_1\in\mathsf{G}_1,\ldots,\mathsf{v}_q\in\mathsf{G}_q}{\text{minimize}} \;\; \mathsf{h}^*\Big(-\sum_{k=1}^{q} \mathsf{L}_k^*\mathsf{v}_k\Big) + \sum_{k=1}^{q} \mathsf{g}_k^*(\mathsf{v}_k) + \mathsf{l}_k^*(\mathsf{v}_k).$

- $\mathsf{h}\colon \mathsf{H} \to \mathbb{R}$ convex, μ-Lipschitz differentiable function with $\mu \in]0,+\infty[$
- $\mathsf{g}_k \in \Gamma_0(\mathsf{G}_k)$ with G_k separable real Hilbert space
- $\mathsf{l}_k \in \Gamma_0(\mathsf{G}_k)$ ν_k-strongly convex with $\nu_k \in]0,+\infty[$
  $\Leftrightarrow \mathsf{l}_k^* \in \Gamma_0(\mathsf{G})$ ν_k-Lipschitz differentiable
- $\mathsf{L}_k\colon \mathsf{H} \to \mathsf{G}_k$ linear and bounded.

Parallel proximal primal-dual problem

PRIMAL PROBLEM

We want to $\underset{x \in H}{\text{minimize}} \; h(x) + \sum_{k=1}^{q} (g_k \,\square\, l_k)(L_k x).$

DUAL PROBLEM

We want to $\underset{v_1 \in G_1, \dots, v_q \in G_q}{\text{minimize}} \; h^*\Big(-\sum_{k=1}^{q} L_k^* v_k\Big) + \sum_{k=1}^{q} g_k^*(v_k) + l_k^*(v_k).$

DIFFICULTIES:

- ⋆ Large-size optimization problem
- ⋆ Functions g_k often nonsmooth
(indicator functions of constraint sets, sparsity measures,...).
- ⋆ Linear operators required by standard optimization methods (e.g. ADMM) difficult to invert due to the form of operators L_k
(e.g. weighted incidence matrices of graphs).

Parallel proximal primal-dual algorithm

for $n = 0, 1, \ldots$

$$\mathsf{s}_n \simeq \mathsf{x}_n - \mathsf{W}\nabla \mathsf{h}(\mathsf{x}_n)$$

$$\mathsf{y}_n = \mathsf{s}_n - \mathsf{W}\sum_{k=1}^{q} \mathsf{L}_k^* \mathsf{v}_{k,n}$$

for $k = 1, \ldots, q$

$$\mathsf{u}_{k,n} \simeq \mathrm{prox}_{\mathsf{g}_k^*}^{\mathsf{U}_k^{-1}}\Big(\mathsf{v}_{k,n} + \mathsf{U}_k\big(\mathsf{L}_k \mathsf{y}_n - \nabla \mathsf{l}_k^*(\mathsf{v}_{k,n})\big)\Big)$$

$$\mathsf{v}_{k,n+1} = \mathsf{v}_{k,n} + \lambda_n(\mathsf{u}_{k,n} - \mathsf{v}_{k,n})$$

$$\mathsf{p}_n = \mathsf{s}_n - \mathsf{W}\sum_{k=1}^{q} \mathsf{L}_k^* \mathsf{u}_{k,n}$$

$$\mathsf{x}_{n+1} = \mathsf{x}_n + \lambda_n(\mathsf{p}_n - \mathsf{x}_n).$$

Parallel proximal primal-dual algorithm

for $n = 0, 1, \dots$

$$\mathsf{s}_n \simeq \mathsf{x}_n - \mathsf{W}\nabla \mathsf{h}(\mathsf{x}_n)$$

$$\mathsf{y}_n = \mathsf{s}_n - \mathsf{W}\sum_{k=1}^{q} \mathsf{L}_k^* \mathsf{v}_{k,n}$$

for $k = 1, \dots, q$

$$\mathsf{u}_{k,n} \simeq \operatorname{prox}_{\mathsf{g}_k^*}^{\mathsf{U}_k^{-1}} \Big(\mathsf{v}_{k,n} + \mathsf{U}_k \big(\mathsf{L}_k \mathsf{y}_n - \nabla \mathsf{l}_k^*(\mathsf{v}_{k,n}) \big) \Big)$$

$$\mathsf{v}_{k,n+1} = \mathsf{v}_{k,n} + \lambda_n (\mathsf{u}_{k,n} - \mathsf{v}_{k,n})$$

$$\mathsf{p}_n = \mathsf{s}_n - \mathsf{W}\sum_{k=1}^{q} \mathsf{L}_k^* \mathsf{u}_{k,n}$$

$$\mathsf{x}_{n+1} = \mathsf{x}_n + \lambda_n (\mathsf{p}_n - \mathsf{x}_n).$$

Parallel proximal primal-dual algorithm

for $n = 0, 1, \ldots$

$$\mathsf{s}_n \simeq \mathsf{x}_n - \mathsf{W}\nabla \mathsf{h}(\mathsf{x}_n)$$

$$\mathsf{y}_n = \mathsf{s}_n - \mathsf{W}\sum_{k=1}^{q} \mathsf{L}_k^* \mathsf{v}_{k,n}$$

for $k = 1, \ldots, q$

$$\mathsf{u}_{k,n} \simeq \mathrm{prox}_{\mathsf{g}_k^*}^{\mathsf{U}_k^{-1}}\Big(\mathsf{v}_{k,n} + \mathsf{U}_k\big(\mathsf{L}_k \mathsf{y}_n - \nabla \mathsf{l}_k^*(\mathsf{v}_{k,n})\big)\Big)$$

$$\mathsf{v}_{k,n+1} = \mathsf{v}_{k,n} + \lambda_n(\mathsf{u}_{k,n} - \mathsf{v}_{k,n})$$

$$\mathsf{p}_n = \mathsf{s}_n - \mathsf{W}\sum_{k=1}^{q} \mathsf{L}_k^* \mathsf{u}_{k,n}$$

$$\mathsf{x}_{n+1} = \mathsf{x}_n + \lambda_n(\mathsf{p}_n - \mathsf{x}_n).$$

where

- $\mathsf{W} \colon \mathsf{H} \to \mathsf{H}$ and $(\forall k \in \{1, \ldots, q\})$ $\mathsf{U}_k \colon \mathsf{G}_k \to \mathsf{G}_k$ strongly positive self-adjoint bounded linear operators such that

  $$\min\left\{\mu^{-1}\|\mathsf{W}\|^{-1}, \nu^{-1}\left(1 - \sum_{k=1}^{q}\|\mathsf{U}_k^{1/2}\mathsf{L}_k\mathsf{W}^{1/2}\|^2\right)\right\} > 1/2$$

  with $\nu = \max\{\nu_1\|\mathsf{U}_1\|, \ldots, \nu_q\|\mathsf{U}_q\|\}$.

Parallel proximal primal-dual algorithm

for $n = 0, 1, \ldots$

$$\left\lfloor \begin{array}{l} \mathsf{s}_n \simeq \mathsf{x}_n - \mathsf{W}\nabla \mathsf{h}(\mathsf{x}_n) \\ \mathsf{y}_n = \mathsf{s}_n - \mathsf{W}\sum_{k=1}^{q} \mathsf{L}_k^* \mathsf{v}_{k,n} \\ \text{for } k = 1, \ldots, q \\ \left\lfloor \begin{array}{l} \mathsf{u}_{k,n} \simeq \mathrm{prox}_{\mathsf{g}_k^*}^{\mathsf{U}_k^{-1}}\Big(\mathsf{v}_{k,n} + \mathsf{U}_k\big(\mathsf{L}_k \mathsf{y}_n - \nabla \mathsf{l}_k^*(\mathsf{v}_{k,n})\big)\Big) \\ \mathsf{v}_{k,n+1} = \mathsf{v}_{k,n} + \lambda_n(\mathsf{u}_{k,n} - \mathsf{v}_{k,n}) \end{array} \right. \\ \mathsf{p}_n = \mathsf{s}_n - \mathsf{W}\sum_{k=1}^{q} \mathsf{L}_k^* \mathsf{u}_{k,n} \\ \mathsf{x}_{n+1} = \mathsf{x}_n + \lambda_n(\mathsf{p}_n - \mathsf{x}_n). \end{array} \right.$$

where

- $(\forall n \in \mathbb{N})\ \lambda_n \in \,]0,1]$ such that $\inf_{n\in\mathbb{N}} \lambda_n > 0$.

Parallel proximal primal-dual algorithm

for $n = 0, 1, \ldots$

$$
\left\lfloor
\begin{array}{l}
\mathsf{s}_n \simeq \mathsf{x}_n - \mathsf{W}\nabla \mathsf{h}(\mathsf{x}_n) \\
\mathsf{y}_n = \mathsf{s}_n - \mathsf{W} \sum_{k=1}^{q} \mathsf{L}_k^* \mathsf{v}_{k,n} \\
\text{for } k = 1, \ldots, q \\
\left\lfloor
\begin{array}{l}
\mathsf{u}_{k,n} \simeq \mathrm{prox}_{\mathsf{g}_k^*}^{\mathsf{U}_k^{-1}} \Big(\mathsf{v}_{k,n} + \mathsf{U}_k\big(\mathsf{L}_k \mathsf{y}_n - \nabla \mathsf{l}_k^*(\mathsf{v}_{k,n})\big)\Big) \\
\mathsf{v}_{k,n+1} = \mathsf{v}_{k,n} + \lambda_n(\mathsf{u}_{k,n} - \mathsf{v}_{k,n})
\end{array}
\right. \\
\mathsf{p}_n = \mathsf{s}_n - \mathsf{W} \sum_{k=1}^{q} \mathsf{L}_k^* \mathsf{u}_{k,n} \\
\mathsf{x}_{n+1} = \mathsf{x}_n + \lambda_n(\mathsf{p}_n - \mathsf{x}_n).
\end{array}
\right.
$$

Assume that there exists $\overline{\mathsf{x}} \in \mathsf{H}$ such that

$$0 \in \nabla \mathsf{h}(\overline{\mathsf{x}}) + \sum_{k=1}^{q} \mathsf{L}_k^* (\partial \mathsf{g}_k \,\Box\, \partial \mathsf{l}_k)(\mathsf{L}_k \overline{\mathsf{x}}).$$

We have:

- ⋆ $\mathsf{x}_n \rightharpoonup \widehat{\mathsf{x}}$ where $\widehat{\mathsf{x}}$ is a solution to the primal problem
- ⋆ $(\forall k \in \{1, \ldots, q\})\ \mathsf{v}_{k,n} \rightharpoonup \widehat{\mathsf{v}}_k$ where $(\widehat{\mathsf{v}}_k)_{1 \leqslant k \leqslant q}$ is a solution to the dual problem.

Proximal primal-dual algorithm

ADVANTAGES:

- No linear operator inversion.
- Use of proximable or/and differentiable functions.
- Less restrictive convergence conditions than other primal-dual algorithms.

DISADVANTAGES: At each iteration,

- all the dual variables are updated in parallel,
- it is necessary to update the full primal variable.

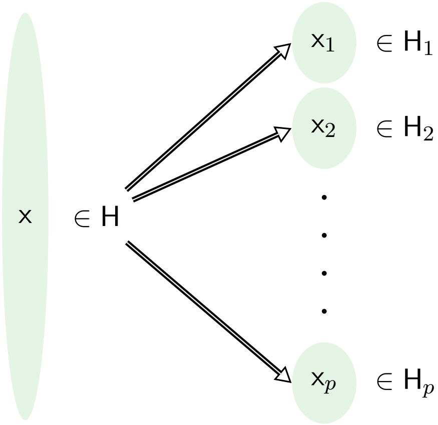
Proximal primal-dual algorithm

Bibliographical remarks:

- Pioneering work in the 1950's: Arrow-Hurwicz method.
- Methods based on Forward-Backward iteration
  - type I: [Vu - 2013][Condat - 2013]
    (extensions of [Esser *et al.* - 2010][Chambolle and Pock - 2011])
  - type II : [Combettes *et al.* - 2014]
    (extensions of [Loris and Verhoeven - 2011][Chen *et al.* - 2014])
- Methods based on Forward-Backward-Forward iteration
  [Combettes and Pesquet - 2012] [Boţ and Hendrich - 2014]
- Projection based methods
  [Alotaibi *et al.* - 2013]
- ...

Improvement via block alternation

- Idea: split variable.



$$\mathsf{H} = \times_{j=1}^{p} \mathsf{H}_j$$

$\mathsf{H}_1, \ldots, \mathsf{H}_p$ are real separable Hilbert spaces

Improvement via block alternation

► Assumption: h is an additively block separable function.

$$\mathsf{h}\left(\mathbf{x}\right) = \mathsf{h}\begin{pmatrix} \mathsf{x}_1 \\ \mathsf{x}_2 \\ \vdots \\ \vdots \\ \mathsf{x}_p \end{pmatrix} = \sum_{j=1}^{p} \mathsf{h}_j(\mathsf{x}_j)$$

$(\forall j \in \{1, \ldots, p\})$ h_j convex and μ_j-Lipschitz differentiable with $\mu_j \in \,]0, +\infty[$.

Block-coordinate strategy

- ⋆ At each iteration $n \in \mathbb{N}$, update only a subset of components (∼ Gauss-Seidel methods).

ADVANTAGES:

- ⋆ Reduced computational cost at each iteration.
- ⋆ Reduced memory requirement.
- ⋆ More flexibility.

Primal-dual problem

PRIMAL PROBLEM

$$\underset{\mathsf{x}_1\in\mathsf{H}_1,\ldots,\mathsf{x}_p\in\mathsf{H}_p}{\text{minimize}} \quad \sum_{j=1}^{p}\mathsf{h}_j(\mathsf{x}_j) + \sum_{k=1}^{q}\Big(\mathsf{g}_k \,\square\, \mathsf{l}_k\Big)\Big(\sum_{j=1}^{p}\mathsf{L}_{k,j}\mathsf{x}_j\Big)$$

$(\forall j \in \{1,\ldots,p\})(\forall k \in \{1,\ldots,q\})$

- H_j and G_k real separable Hilbert spaces
- $\mathsf{h}_j\colon \mathsf{H}_j \to \mathbb{R}$ convex, μ_j-Lipschitz differentiable, with $\mu_j \in \,]0,+\infty[$
- $\mathsf{g}_k \in \Gamma_0(\mathsf{G}_k)$
- $\mathsf{l}_k \in \Gamma_0(\mathsf{G}_k)$ ν_k-strongly convex, with $\nu_k \in \,]0,+\infty[$
- $\mathsf{L}_{k,j}\colon \mathsf{H}_j \to \mathsf{G}_k$ is linear and bounded
- $\mathbb{L}_k = \big\{j \in \{1,\ldots,p\} \;\big|\; \mathsf{L}_{k,j} \neq 0\big\} \neq \varnothing$, and $\mathbb{L}_j^* = \big\{k \in \{1,\ldots,q\} \;\big|\; \mathsf{L}_{k,j} \neq 0\big\} \neq \varnothing$.

Primal-dual problem

PRIMAL PROBLEM

$$\underset{\mathsf{x}_1\in\mathsf{H}_1,\ldots,\mathsf{x}_p\in\mathsf{H}_p}{\text{minimize}} \quad \sum_{j=1}^{p} \mathsf{h}_j(\mathsf{x}_j) + \sum_{k=1}^{q} \big(\mathsf{g}_k \,\square\, \mathsf{l}_k\big)\Big(\sum_{j=1}^{p} \mathsf{L}_{k,j}\mathsf{x}_j\Big)$$

DUAL PROBLEM

$$\underset{\mathsf{v}_1\in\mathsf{G}_1,\ldots,\mathsf{v}_q\in\mathsf{G}_q}{\text{minimize}} \quad \sum_{j=1}^{p} \mathsf{h}_j^*\Big(-\sum_{k=1}^{q} \mathsf{L}_{k,j}^*\mathsf{v}_k\Big) + \sum_{k=1}^{q} \big(\mathsf{g}_k^*(\mathsf{v}_k) + \mathsf{l}_k^*(\mathsf{v}_k)\big)$$

- Assume that there exists $(\overline{\mathsf{x}}_1,\ldots,\overline{\mathsf{x}}_p) \in \mathsf{H}_1\times\ldots\times\mathsf{H}_p$ such that

$$(\forall j\in\{1,\ldots,p\}) \quad 0 \in \partial\nabla\mathsf{h}_j(\overline{\mathsf{x}}_j) + \sum_{k=1}^{q} \mathsf{L}_{k,j}^*(\partial\mathsf{g}_k \,\square\, \partial\mathsf{l}_k)\big(\mathsf{L}_{k,j}\overline{\mathsf{x}}_j\big).$$

OBJECTIVE: Let $\mathbf{F}$ and $\mathbf{F}^*$ be the sets of solutions to the primal and dual problems. Find an $\mathbf{F}\times\mathbf{F}^*$-valued random variable $(\widehat{\boldsymbol{x}},\widehat{\boldsymbol{v}})$.

Random block-coordinate proximal primal-dual algorithm

For $n = 0, 1, \ldots$

for $j = 1, \ldots, p$

$$\eta_{j,n} = \max\left\{\varepsilon_{p+k,n} \mid k \in \mathbb{L}_j^*\right\}$$

$$s_{j,n} = \eta_{j,n}\Big(x_{j,n} - \mathsf{W}_j\big(\nabla \mathsf{h}_j(x_{j,n}) + a_{j,n}\big)\Big)$$

$$y_{j,n} = \eta_{j,n}\big(s_{j,n} - \mathsf{W}_j \sum_{k\in\mathbb{L}_j^*} \mathsf{L}_{k,j}^* v_{k,n}\big)$$

for $k = 1, \ldots, q$

$$u_{k,n+1} = \varepsilon_{p+k,n}\Big(\operatorname{prox}_{\mathsf{g}_k^*}^{\mathsf{U}_k^{-1}}\Big(v_{k,n} + \mathsf{U}_k \sum_{j\in\mathbb{L}_k} \mathsf{L}_{k,j} y_{j,n} - \mathsf{U}_k\big(\nabla \mathsf{l}_k^*(v_{k,n}) + c_{k,n}\big)\Big) + b_{k,n}\Big)$$

$$v_{k,n+1} = v_{k,n} + \lambda_n \varepsilon_{p+k,n}(u_{k,n} - v_{k,n})$$

for $j = 1, \ldots, p$

$$p_{j,n+1} = \varepsilon_{j,n}\Big(s_{j,n} - \mathsf{W}_j \sum_{k\in\mathbb{L}_j^*} \mathsf{L}_{k,j}^* v_{k,n}\Big)$$

$$x_{j,n+1} = x_{j,n} + \lambda_n \varepsilon_{j,n}(p_{j,n} - x_{j,n}).$$

where

- $(\varepsilon_n)_{n\in\mathbb{N}} \rightsquigarrow$ binary variables signaling the blocks to be activated

Random block-coordinate proximal primal-dual algorithm

For $n = 0, 1, \ldots$

for $j = 1, \ldots, p$

$$\eta_{j,n} = \max\left\{\varepsilon_{p+k,n} \mid k \in \mathbb{L}_j^*\right\}$$
$$s_{j,n} = \eta_{j,n}\Big(x_{j,n} - \mathsf{W}_j\big(\nabla \mathsf{h}_j(x_{j,n}) + a_{j,n}\big)\Big)$$
$$y_{j,n} = \eta_{j,n}\Big(s_{j,n} - \mathsf{W}_j \sum_{k\in\mathbb{L}_j^*} \mathsf{L}_{k,j}^* v_{k,n}\Big)$$

for $k = 1, \ldots, q$

$$u_{k,n+1} = \varepsilon_{p+k,n}\Big(\operatorname{prox}_{\mathsf{g}_k^*}^{\mathsf{U}_k^{-1}}\Big(v_{k,n} + \mathsf{U}_k \sum_{j\in\mathbb{L}_k} \mathsf{L}_{k,j} y_{j,n} - \mathsf{U}_k\big(\nabla \mathsf{l}_k^*(v_{k,n}) + c_{k,n}\big)\Big) + b_{k,n}\Big)$$
$$v_{k,n+1} = v_{k,n} + \lambda_n \varepsilon_{p+k,n}(u_{k,n} - v_{k,n})$$

for $j = 1, \ldots, p$

$$p_{j,n+1} = \varepsilon_{j,n}\Big(s_{j,n} - \mathsf{W}_j \sum_{k\in\mathbb{L}_j^*} \mathsf{L}_{k,j}^* v_{k,n}\Big)$$
$$x_{j,n+1} = x_{j,n} + \lambda_n \varepsilon_{j,n}(p_{j,n} - x_{j,n}).$$

where

- $(\boldsymbol{\varepsilon}_n)_{n\in\mathbb{N}}$ identically distributed $\mathbb{D}$-valued random variables with $\mathbb{D} = \{0,1\}^{p+q} \smallsetminus \{\mathbf{0}\}$
  $\rightsquigarrow$ binary variables signaling the blocks to be activated
- $\boldsymbol{x}_0$, $(\boldsymbol{a}_n)_{n\in\mathbb{N}}$, and $(\boldsymbol{c}_n)_{n\in\mathbb{N}}$ H-valued random variables, $\boldsymbol{v}_0$, $(\boldsymbol{b}_n)_{n\in\mathbb{N}}$, and $(\boldsymbol{d}_n)_{n\in\mathbb{N}}$ **G**-valued random variables with $\mathbf{G} = \mathsf{G}_1 \times \cdots \times \mathsf{G}_q$
  $\rightsquigarrow$ $(\boldsymbol{a}_n)_{n\in\mathbb{N}}$, $(\boldsymbol{b}_n)_{n\in\mathbb{N}}$ and $(\boldsymbol{c}_n)_{n\in\mathbb{N}}$: error terms

Random block-coordinate proximal primal-dual algorithm

For $n = 0, 1, \ldots$
  for $j = 1, \ldots, p$
    $\eta_{j,n} = \max\left\{\varepsilon_{p+k,n} \mid k \in \mathbb{L}_j^*\right\}$
    $s_{j,n} = \eta_{j,n}\Big(x_{j,n} - \mathsf{W}_j\big(\nabla \mathsf{h}_j(x_{j,n}) + a_{j,n}\big)\Big)$
    $y_{j,n} = \eta_{j,n}\big(s_{j,n} - \mathsf{W}_j \sum_{k \in \mathbb{L}_j^*} \mathsf{L}_{k,j}^* v_{k,n}\big)$
  for $k = 1, \ldots, q$
    $u_{k,n+1} = \varepsilon_{p+k,n}\Big(\operatorname{prox}_{\mathsf{g}_k^*}^{\mathsf{U}_k^{-1}}\Big(v_{k,n} + \mathsf{U}_k \sum_{j \in \mathbb{L}_k} \mathsf{L}_{k,j} y_{j,n} - \mathsf{U}_k\big(\nabla \mathsf{l}_k^*(v_{k,n}) + c_{k,n}\big)\Big) + b_{k,n}\Big)$
    $v_{k,n+1} = v_{k,n} + \lambda_n \varepsilon_{p+k,n}(u_{k,n} - v_{k,n})$
  for $j = 1, \ldots, p$
    $p_{j,n+1} = \varepsilon_{j,n}\Big(s_{j,n} - \mathsf{W}_j \sum_{k \in \mathbb{L}_j^*} \mathsf{L}_{k,j}^* v_{k,n}\Big)$
    $x_{j,n+1} = x_{j,n} + \lambda_n \varepsilon_{j,n}(p_{j,n} - x_{j,n})$.

where

- $(\boldsymbol{\varepsilon}_n)_{n\in\mathbb{N}} \rightsquigarrow$ binary variables signaling the blocks to be activated
- $(\boldsymbol{a}_n)_{n\in\mathbb{N}}$, $(\boldsymbol{b}_n)_{n\in\mathbb{N}}$ and $(\boldsymbol{c}_n)_{n\in\mathbb{N}}$: error terms
- $(\forall j \in \{1, \ldots, p\})$ $\mathsf{W}_j \colon \mathsf{H}_j \to \mathsf{H}_j$ and $(\forall k \in \{1, \ldots, q\})$ $\mathsf{U}_k \colon \mathsf{G}_k \to \mathsf{G}_k$ strongly positive self-adjoint preconditioning linear operators such that

$$\min\left\{\mu^{-1}\|\mathsf{W}\|^{-1}, \nu^{-1}\Big(1 - \sum_{k=1}^{q} \|\mathsf{U}_k^{1/2}\mathsf{L}_k\mathsf{W}^{1/2}\|^2\Big)\right\} > 1/2$$

with $\mu = \max\{\mu_1\|\mathsf{W}_1\|, \ldots, \mu_p\|\mathsf{W}_p\|\}$ and
$\nu = \max\{\nu_1\|\mathsf{U}_1\|, \ldots, \nu_q\|\mathsf{U}_q\|\}$.

Random block-coordinate proximal primal-dual algorithm

For $n = 0, 1, \ldots$
- for $j = 1, \ldots, p$
  - $\eta_{j,n} = \max\left\{\varepsilon_{p+k,n} \mid k \in \mathbb{L}_j^*\right\}$
  - $s_{j,n} = \eta_{j,n}\Big(x_{j,n} - \mathsf{W}_j\big(\nabla \mathsf{h}_j(x_{j,n}) + a_{j,n}\big)\Big)$
  - $y_{j,n} = \eta_{j,n}\big(s_{j,n} - \mathsf{W}_j \sum_{k\in\mathbb{L}_j^*} \mathsf{L}_{k,j}^* v_{k,n}\big)$
- for $k = 1, \ldots, q$
  - $u_{k,n+1} = \varepsilon_{p+k,n}\Big(\mathrm{prox}_{\mathsf{g}_k^*}^{\mathsf{U}_k^{-1}}\Big(v_{k,n} + \mathsf{U}_k \sum_{j\in\mathbb{L}_k} \mathsf{L}_{k,j} y_{j,n} - \mathsf{U}_k\big(\nabla \mathsf{l}_k^*(v_{k,n}) + c_{k,n}\big)\Big) + b_{k,n}\Big)$
  - $v_{k,n+1} = v_{k,n} + \lambda_n \varepsilon_{p+k,n}(u_{k,n} - v_{k,n})$
- for $j = 1, \ldots, p$
  - $p_{j,n+1} = \varepsilon_{j,n}\Big(s_{j,n} - \mathsf{W}_j \sum_{k\in\mathbb{L}_j^*} \mathsf{L}_{k,j}^* v_{k,n}\Big)$
  - $x_{j,n+1} = x_{j,n} + \lambda_n \varepsilon_{j,n}(p_{j,n} - x_{j,n})$.

where

- $(\boldsymbol{\varepsilon}_n)_{n\in\mathbb{N}} \rightsquigarrow$ binary variables signaling the blocks to be activated
- $(\boldsymbol{a}_n)_{n\in\mathbb{N}}$, $(\boldsymbol{b}_n)_{n\in\mathbb{N}}$ and $(\boldsymbol{c}_n)_{n\in\mathbb{N}}$: error terms
- $(\forall j \in \{1, \ldots, p\})$ $\mathsf{W}_j : \mathsf{H}_j \to \mathsf{H}_j$ and $(\forall k \in \{1, \ldots, q\})$ $\mathsf{U}_k : \mathsf{G}_k \to \mathsf{G}_k$ strongly positive self-adjoint preconditioning linear operators
- $(\forall n \in \mathbb{N})$ $\lambda_n \in]0, 1]$ such that $\inf_{n\in\mathbb{N}} \lambda_n > 0$.

Random block-coordinate proximal primal-dual algorithm

Let $(\Omega, \mathcal{F}, \mathsf{P})$ be the underlying probability space.
Set $(\forall n \in \mathbb{N})\ \mathfrak{X}_n = (x_{n'}, v_{n'})_{0 \leqslant n' \leqslant n}$. Assume that

- $\sum_{n\in\mathbb{N}} \sqrt{\mathsf{E}(\|\boldsymbol{a}_n\|^2 \,|\, \mathfrak{X}_n)} < +\infty$, $\sum_{n\in\mathbb{N}} \sqrt{\mathsf{E}(\|\boldsymbol{b}_n\|^2 \,|\, \mathfrak{X}_n)} < +\infty$, and $\sum_{n\in\mathbb{N}} \sqrt{\mathsf{E}(\|\boldsymbol{c}_n\|^2 \,|\, \mathfrak{X}_n)} < +\infty$ a.s.
- The variables $(\boldsymbol{\varepsilon}_n)_{n\in\mathbb{N}}$ are identically distributed such that $(\forall j \in \{1,\ldots,p\})\ \mathsf{P}[\varepsilon_{j,0} = 1] > 0$.
- For every $n \in \mathbb{N}$, $\boldsymbol{\varepsilon}_n$ and $\mathfrak{X}_n$ are independent.
- For every $k \in \{1,\ldots,q\}$ and $n \in \mathbb{N}$,

$$\varepsilon_{p+k,n} = \max_{1\leqslant j\leqslant p} \left\{ \varepsilon_{j,n} \;\middle|\; j \in \mathbb{L}_k \right\}.$$

- $(\boldsymbol{x}_n)_{n\in\mathbb{N}}$ converges weakly a.s. to an **F**-valued random variable.
- $(\boldsymbol{v}_n)_{n\in\mathbb{N}}$ converges weakly a.s. to an $\mathbf{F}^*$-valued random variable.

Proof: Based on properties of quasi-Fejér stochastic sequences [Combettes and Pesquet – 2014].

Illustration of the random sampling strategy

Variable selection $(\forall n \in \mathbb{N})$

$x_{1,n}$ activated when $\varepsilon_{1,n} = 1$

$x_{2,n}$ activated when $\varepsilon_{2,n} = 1$

$x_{3,n}$ activated when $\varepsilon_{3,n} = 1$

$x_{4,n}$ activated when $\varepsilon_{4,n} = 1$

$x_{5,n}$ activated when $\varepsilon_{5,n} = 1$

$x_{6,n}$ activated when $\varepsilon_{6,n} = 1$

How choosing $(\forall n \in \mathbb{N})$ the variable $\boldsymbol{\varepsilon}_n = (\varepsilon_{1,n}, \ldots, \varepsilon_{6,n})$?

Illustration of the random sampling strategy

Variable selection $(\forall n \in \mathbb{N})$

- $x_{1,n}$ activated when $\varepsilon_{1,n} = 1$
- $x_{2,n}$ activated when $\varepsilon_{2,n} = 1$
- $x_{3,n}$ activated when $\varepsilon_{3,n} = 1$
- $x_{4,n}$ activated when $\varepsilon_{4,n} = 1$
- $x_{5,n}$ activated when $\varepsilon_{5,n} = 1$
- $x_{6,n}$ activated when $\varepsilon_{6,n} = 1$

How choosing $(\forall n \in \mathbb{N})$ the variable $\boldsymbol{\varepsilon}_n = (\varepsilon_{1,n}, \ldots, \varepsilon_{6,n})$?

$\mathsf{P}[\boldsymbol{\varepsilon}_n = (1,1,0,0,0,0)] = 0.1$

Illustration of the random sampling strategy

Variable selection $(\forall n \in \mathbb{N})$

- $x_{1,n}$ activated when $\varepsilon_{1,n} = 1$
- $x_{2,n}$ activated when $\varepsilon_{2,n} = 1$
- $x_{3,n}$ activated when $\varepsilon_{3,n} = 1$
- $x_{4,n}$ activated when $\varepsilon_{4,n} = 1$
- $x_{5,n}$ activated when $\varepsilon_{5,n} = 1$
- $x_{6,n}$ activated when $\varepsilon_{6,n} = 1$

How choosing $(\forall n \in \mathbb{N})$ the variable $\boldsymbol{\varepsilon}_n = (\varepsilon_{1,n}, \ldots, \varepsilon_{6,n})$?

$\mathrm{P}[\boldsymbol{\varepsilon}_n = (1,1,0,0,0,0)] = 0.1$

$\mathrm{P}[\boldsymbol{\varepsilon}_n = (1,0,1,0,0,0)] = 0.2$

Illustration of the random sampling strategy

Variable selection $(\forall n \in \mathbb{N})$

- $x_{1,n}$ activated when $\varepsilon_{1,n} = 1$
- $x_{2,n}$ activated when $\varepsilon_{2,n} = 1$
- $x_{3,n}$ activated when $\varepsilon_{3,n} = 1$
- $x_{4,n}$ activated when $\varepsilon_{4,n} = 1$
- $x_{5,n}$ activated when $\varepsilon_{5,n} = 1$
- $x_{6,n}$ activated when $\varepsilon_{6,n} = 1$

How choosing $(\forall n \in \mathbb{N})$ the variable $\boldsymbol{\varepsilon}_n = (\varepsilon_{1,n}, \ldots, \varepsilon_{6,n})$?

$\mathsf{P}[\boldsymbol{\varepsilon}_n = (1,1,0,0,0,0)] = 0.1$

$\mathsf{P}[\boldsymbol{\varepsilon}_n = (1,0,1,0,0,0)] = 0.2$

$\mathsf{P}[\boldsymbol{\varepsilon}_n = (1,0,0,1,1,0)] = 0.2$

Illustration of the random sampling strategy

Variable selection $(\forall n \in \mathbb{N})$

- $x_{1,n}$ activated when $\varepsilon_{1,n} = 1$
- $x_{2,n}$ activated when $\varepsilon_{2,n} = 1$
- $x_{3,n}$ activated when $\varepsilon_{3,n} = 1$
- $x_{4,n}$ activated when $\varepsilon_{4,n} = 1$
- $x_{5,n}$ activated when $\varepsilon_{5,n} = 1$
- $x_{6,n}$ activated when $\varepsilon_{6,n} = 1$

How choosing $(\forall n \in \mathbb{N})$ the variable $\boldsymbol{\varepsilon}_n = (\varepsilon_{1,n}, \ldots, \varepsilon_{6,n})$?

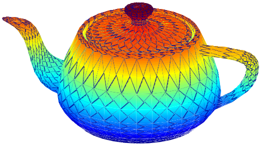
$\mathrm{P}[\boldsymbol{\varepsilon}_n = (1,1,0,0,0,0)] = 0.1$

$\mathrm{P}[\boldsymbol{\varepsilon}_n = (1,0,1,0,0,0)] = 0.2$

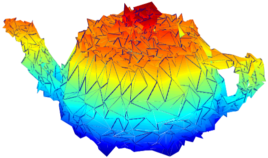
$\mathrm{P}[\boldsymbol{\varepsilon}_n = (1,0,0,1,1,0)] = 0.2$

$\mathrm{P}[\boldsymbol{\varepsilon}_n = (0,1,1,1,1,1)] = 0.5$

Application: 3D mesh denoising



Original mesh $\overline{\mathbf{x}}$



Observed mesh $\mathbf{z}$

Undirected nonreflexive graph

OBJECTIVE: Estimate $\overline{\mathbf{x}} = (\overline{\mathsf{x}}_i)_{1 \leqslant i \leqslant M}$ from noisy observations $\mathbf{z} = (\mathsf{z}_i)_{1 \leqslant i \leqslant M}$ where, for every $i \in \{1, \ldots, M\}$, $\overline{\mathsf{x}}_i \in \mathbb{R}^3$ is the vector of 3D coordinates of the i-th vertex of a mesh

⋆ $\mathsf{H} = (\mathbb{R}^3)^M$

Application: 3D mesh denoising

OBJECTIVE: Estimate $\overline{\mathbf{x}} = (\overline{\mathsf{x}}_i)_{1 \leqslant i \leqslant M}$ from noisy observations $\mathbf{z} = (\mathsf{z}_i)_{1 \leqslant i \leqslant M}$ where, for every $i \in \{1, \ldots, M\}$, $\overline{\mathsf{x}}_i \in \mathbb{R}^3$ is the vector of 3D coordinates of the i-th vertex of a mesh

COST FUNCTION:

$$\Phi(\mathbf{x}) = \sum_{j=1}^{M} \psi_j(\mathsf{x}_j - \mathsf{z}_j) + \iota_{\mathsf{C}_j}(\mathsf{x}_j) + \eta_j \|(\mathsf{x}_j - \mathsf{x}_i)_{i \in \mathcal{N}_j}\|_{1,2}$$

where $(\forall j \in \{1, \ldots, M\})$,

- ⋆ $\psi_j \colon \mathbb{R}^3 \to \mathbb{R}$: $\ell_2 - \ell_1$ Huber function
  - robust data fidelity measure
  - convex, Lipschitz differentiable function
- ⋆ C_j: nonempty convex subset of $\mathbb{R}^3$
- ⋆ $\mathcal{N}_j$: neighborhood of j-th vertex
- ⋆ $(\eta_j)_{1 \leqslant j \leqslant M}$: nonnegative regularization constants.

Application: 3D mesh denoising

OBJECTIVE: Estimate $\overline{\mathsf{x}} = (\overline{\mathsf{x}}_i)_{1\leqslant i\leqslant M}$ from noisy observations $\mathsf{z} = (\mathsf{z}_i)_{1\leqslant i\leqslant M}$ where, for every $i \in \{1,\ldots,M\}$, $\overline{\mathsf{x}}_i \in \mathbb{R}^3$ is the vector of 3D coordinates of the i-th vertex of a mesh

COST FUNCTION:

$$\Phi(\mathsf{x}) = \sum_{j=1}^{M} \psi_j(\mathsf{x}_j - \mathsf{z}_j) + \iota_{C_j}(\mathsf{x}_j) + \eta_j \|(\mathsf{x}_j - \mathsf{x}_i)_{i\in\mathcal{N}_j}\|_{1,2}$$

IMPLEMENTATION DETAILS: a block $\equiv$ a vertex

$p = M$, $q = 2M$

- ⋆ $(\forall j \in \{1,\ldots,M\})\ \mathsf{h}_j = \psi_j(\cdot - \mathsf{z}_j)$

$(\forall k \in \{1,\ldots,M\})(\forall \mathsf{x} \in (\mathbb{R}^3)^M)$

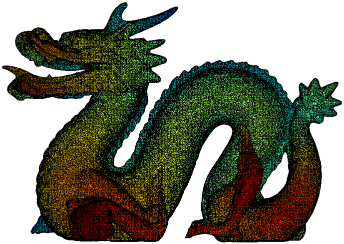
- ⋆ $\mathsf{g}_k(\mathsf{L}_k\mathsf{x}) = \|(\mathsf{x}_k - \mathsf{x}_i)_{i\in\mathcal{N}_k}\|_{1,2}$
- ⋆ $\mathsf{g}_{M+k}(\mathsf{L}_{M+k}\mathsf{x}) = \iota_{C_k}(\mathsf{x}_k)$
- ⋆ $\mathsf{l}_k = \iota_{\{0\}}$

Simulation results

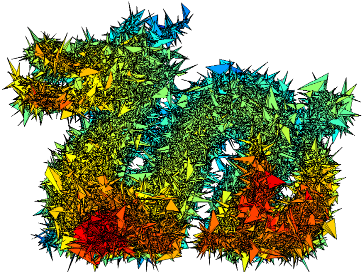
- positions of the original mesh are corrupted through an additive i.i.d. zero-mean Gaussian mixture noise model.
- a limited number r of variables can be handled at each iteration, where

$$\sum_{j=1}^{p} \varepsilon_{j,n} = r \leqslant p.$$

- mesh decomposed into p/r non-overlapping sets.

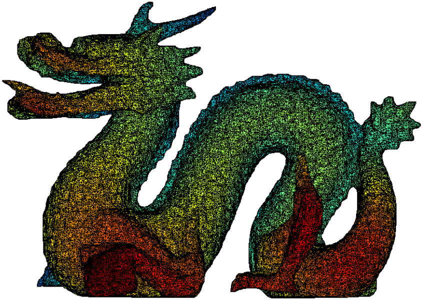


Original mesh, $M = 100250$.



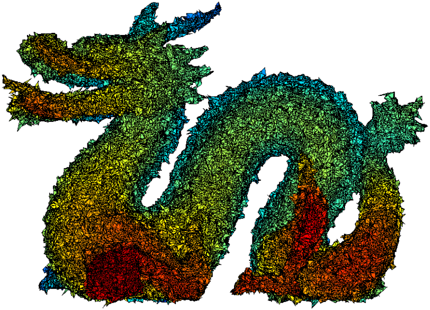
Noisy mesh, MSE = 2.89×10^{-6}.

Simulation results



Proposed reconstruction

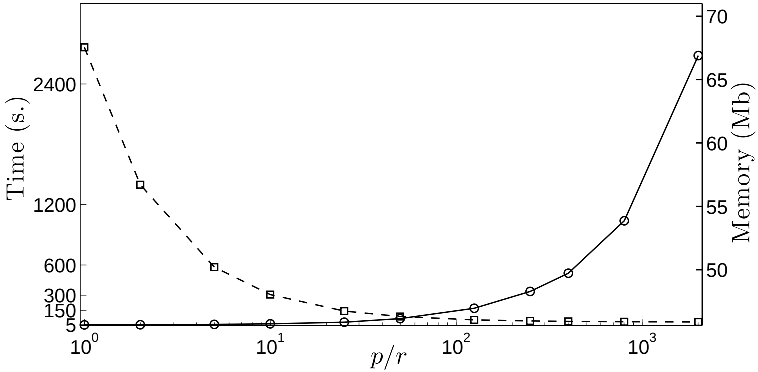
MSE = 8.09×10^{-8}



Laplacian smoothing

MSE = 5.23×10^{-7}

Complexity



- dashed line: required memory
- continuous line: reconstruction time

Conclusion

- No linear operator inversion.
- Flexibility in the random activation of primal/dual components.
- Existing parallel proximal primal-dual algorithms recovered when $p = 1$ and $\varepsilon_n \equiv (1, \dots, 1)$.
- Possibility to address other graph processing problems than denoising.
  [Couprie et al.,2013]
- Available extensions: asynchronous distributed algorithms (stochastic, primal-dual, proximal, defined on a hypergraph).

Some references

P. L. Combettes and J.-C. Pesquet
Proximal splitting methods in signal processing
in *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*, H. H. Bauschke, R. Burachik, P. L. Combettes, V. Elser, D. R. Luke, and H. Wolkowicz editors. Springer-Verlag, New York, pp. 185-212, 2011.

C. Couprie, L. Grady, L. Najman, J.-C. Pesquet, and H. Talbot
Dual constrained TV-based regularization on graphs
SIAM Journal on Imaging Sciences, vol. 6, no 3, pp. 1246-1273, 2013.

P. L. Combettes, L. Condat, J.-C. Pesquet, and B. C. Vũ
A forward-backward view of some primal-dual optimization methods in image recovery
IEEE International Conference on Image Processing (ICIP 2014), pp. 4141-4145, Paris, France, Oct. 27-30, 2014.

P. Combettes and J.-C Pesquet
Stochastic quasi-Fejér block-coordinate fixed point iterations with random sweeping
SIAM Journal on Optimization, vol. 25, no. 2, pp. 1221-1248, July 2015.

N. Komodakis and J.-C. Pesquet
Playing with duality: An overview of recent primal-dual approaches for solving large-scale optimization problems
to appear in *IEEE Signal Processing Magazine*, 2015.

J.-C. Pesquet and A. Repetti
A class of randomized primal-dual algorithms for distributed optimization
to appear in *Journal of Nonlinear and Convex Analysis*, 2015.

A. Repetti, E. Chouzenoux and J.-C. Pesquet
A random block-coordinate primal-dual proximal algorithm with application to 3D mesh denoising
IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2015), pp. 3561-3565, South Brisbane, Australia, 19-24 April 2015.