A PARALLEL BLOCK-COORDINATE APPROACH FOR PRIMAL-DUAL SPLITTING WITH ARBITRARY RANDOM BLOCK SELECTION

Audrey Repetti, Emilie Chouzenoux, and Jean-Christophe Pesquet

Université Paris-Est, Lab. d’Informatique Gaspard Monge, UMR CNRS 8049, Champs-sur-Marne, 77454 Marne-la-Vallée, France
{first.last}@univ-paris-est.fr

ABSTRACT
The solution of many applied problems relies on finding the minimizer of a sum of smooth and/or nonsmooth convex functions possibly involving linear operators. In the last years, primal-dual methods have shown their efficiency to solve such minimization problems, their main advantage being their ability to deal with linear operators with no need to invert them. However, when the problem size becomes increasingly large, the implementation of these algorithms can be complicated, due to memory limitation issues. A simple way to overcome this difficulty consists of splitting the original numerous variables into blocks of smaller dimension, corresponding to the available memory, and to process them separately. In this paper we propose a random block-coordinate primal-dual algorithm, converging almost surely to a solution to the considered minimization problem. Moreover, an application to large-size 3D mesh denoising is provided to show the numerical efficiency of our method.

Index Terms— convex optimization, nonsmooth optimization, primal-dual algorithm, stochastic algorithm, parallel algorithm, random block-coordinate approach, proximity operator, mesh denoising.

1. INTRODUCTION
In the last years, the curse of dimensionality has become a prevailing concern in a number of application areas, e.g. large-scale graph processing, 3D imaging, or machine learning using huge databases. Not only the number of data to handle is becoming increasingly large, but there is also a need for fast processing methods. Many of the problems of interest are actually related to optimization tasks involving cost functions which are often nonsmooth so as to leverage sparsity assumptions on the sought variables. In this context, two basic ideas can be exploited in order to lower the memory requirements and computational cost in high-dimensional optimization methods.

THIS WORK WAS SUPPORTED BY THE AGENCE NATIONALE DE LA RECHERCHE UNDER GRANT ANR GRAPHIP.
randomly, and the computations can be performed in a parallel manner by adjusting the computation load to the number of available processors. Our approach is grounded on recent developments concerning primal-dual convex optimization methods [9–13] as well as on novel convergence results on stochastic fixed point algorithms [14].

This paper is organized as follows. Section 2 gives the formulation of the considered optimization problem and introduces our notation. Section 3 describes the proposed random block-coordinate primal-dual proximal splitting algorithm, discusses its merits with respect to existing methods, and states its convergence properties. Section 4 illustrates the good performance of our method on a 3D mesh denoising problem. Finally, some conclusions are drawn in Section 5.

2. PROBLEM FORMULATION

In this paper, we consider a quite general form of convex optimization problems which can be expressed as

\[
\min_{x_1 \in \mathbb{H}_1, \ldots, x_p \in \mathbb{H}_p} \sum_{j=1}^{p} h_j(x_j) + \sum_{k=1}^{q} (g_k \square l_k) \left( \sum_{j=1}^{p} L_{k,j} x_j \right)
\]

where, for every \( j \in \{1, \ldots, p\} \), \( H_j \) is a separable real Hilbert space, \( h_j : H_j \to [-\infty, +\infty] \) is a convex differentiable function with a Lipschitzian gradient, for every \( k \in \{1, \ldots, q\} \), \( G_k \) is a separable real Hilbert space, \( g_k \in \Gamma_0(G_k) \), \( l_k \in \Gamma_0(G_k) \) is assumed to be strongly convex, and \( L_{k,j} \) is a linear bounded operator from \( H_j \) to \( G_k \) satisfying

\[
\begin{align*}
\mathbb{I}_k &= \left\{ j' \in \{1, \ldots, p\} \mid L_{k,j'} \neq 0 \right\} \neq \emptyset, \\
\mathbb{I}_{k,j}^\prime &= \left\{ k' \in \{1, \ldots, q\} \mid L_{k',j} \neq 0 \right\} \neq \emptyset.
\end{align*}
\]

We recall that the inf-convolution of two functions \( g : H \to [-\infty, +\infty] \) and \( l : H \to [-\infty, +\infty] \) is defined as \( g \square l : y \mapsto \inf_{z \in H} \{ g(z) + l(y - z) \} \). The identity element of the inf-convolution is the indicator function of \( \{0\} \), denoted by \( \delta_{\{0\}} \).

Then, if \( I = \{0\} \), \( g \square I \) reduces to \( g \).

In the remainder of the paper, it is assumed that Problem (1) has at least one solution and that one of the following two technical conditions holds:

- \( \forall j \in \{1, \ldots, p\} \) (\( \forall k \in \{1, \ldots, q\} \), \( (x_j)_{1 \leq j \leq p} \mapsto \sum_{j=1}^{p} L_{k,j} x_j \) is surjective.
- \( \forall k \in \{1, \ldots, q\} \) either \( g_k \) or \( l_k \) is real-valued.

The dual problem associated to Problem (1) is then given by

\[
\begin{align*}
\min_{v_k \in G_1, \ldots, v_q \in G_q} \sum_{j=1}^{p} h_j^* \left( - \sum_{k=1}^{q} L_{k,j}^* v_k \right) \\
+ \sum_{k=1}^{q} (g_k^*(v_k) + l_k^*(v_k)).
\end{align*}
\]

This dual problem involves the Fenchel-Legendre conjugate functions of \((h_j)_{1 \leq j \leq p}, (g_k)_{1 \leq k \leq q}\) and \((l_k)_{1 \leq k \leq q}\), here denoted by a * symbol. We recall that the conjugate \( f^* \) of a proper lower-continuous convex function \( f \) is defined as \( (\forall x \in H) \ f^*(x) = \sup_{y \in H} \{ \langle y, x \rangle - f(y) \} \).

Subsequently, we denote by \( F \) (resp. \( F^* \)) the set of solutions to Problem (1) (resp. (4)).

3. BLOCK-COORDINATE PRIMAL-DUAL ALGORITHM

3.1. Proposed Algorithm

We propose a primal-dual proximal splitting approach for solving Problem (1). Our method is based on a randomized block-coordinate strategy, described in Algorithm 1. For conciseness, in the remainder of this work, \( H = H_1 \oplus \cdots \oplus H_p \) denotes the Hilbert direct sum of \( H_j \)s, \( 1 \leq j \leq p \).

A generic element of \( H \) is denoted by \( x = (x_j)_{1 \leq j \leq p} \) with, for every \( j \in \{1, \ldots, p\} \), \( x_j \in H_j \).

Similarly, we define \( G = G_1 \oplus \cdots \oplus G_q \) with generic element \( v = (v_k)_{1 \leq k \leq q} \) where, for every \( k \in \{1, \ldots, q\} \), \( v_k \in G_k \).

Moreover, \( \Omega = \{0,1\}^{p+q} \setminus \{0\} \) is the set of nonzero binary vectors of length \( p + q \).

Finally, the proximity operator of a proper lower-continuous function \( f \) defined on a Hilbert space \( H \), relative to the metric induced by some strongly positive self-adjoint bounded linear operator \( U : H \to H \), is defined as

\[
\text{prox}_U^f : H \to H : x \mapsto \arg\min_{y \in H} f(y) + \frac{1}{2} \langle x - y, U(x - y) \rangle,
\]

where \( \langle \cdot, \cdot \rangle \) is the scalar product endowing \( H \). We refer the reader to [15] for further mathematical details.

We now discuss the main advantages offered by the proposed algorithm:

(i) In order to accelerate the convergence speed of the method, we modify the underlying space metrics in a manner similar to [16, 17], by introducing preconditioning linear operators \((W_j)_{1 \leq j \leq p}\) and \((U_k)_{1 \leq k \leq q}\).

These operators are assumed to be bounded, self-adjoint and strongly positive.

(ii) A random block alternating strategy over indices \( j \in \{1, \ldots, p\} \) and \( k \in \{1, \ldots, q\} \) is adopted in order to take advantage of the block structure of the problem. More precisely, at each iteration \( n \) of Algorithm 1, \((\varepsilon_{j,n})_{1 \leq j \leq p}\) (resp. \((\varepsilon_{p+k,n})_{1 \leq k \leq q}\)) are Boolean variables signaling the primal variables \((x_j)_{1 \leq j \leq p}\) (resp. dual variables \((v_k)_{1 \leq k \leq q}\)) that are activated. In addition, at iteration \( n \), for every \( j \in \{1, \ldots, p\} \), the Boolean \( \eta_{j,n} \) is chosen such that \( s_{j,n} \) and \( y_{j,n} \) are updated only if there exists some \( k \in \{1, \ldots, q\} \) such that both \( \varepsilon_{p+k,n} \) and \( L_{k,j} \) are nonzero.

(iii) For every \( j \in \{1, \ldots, p\} \) and \( k \in \{1, \ldots, q\} \), the random variables \( (a_{j,n})_{n \in \mathbb{N}}, (b_{k,n})_{n \in \mathbb{N}} \) and \( (c_{k,n})_{n \in \mathbb{N}} \) model
some possible errors arising in the computation of \( \nabla h_j \), \( \text{prox} U^{-1}_\varepsilon \), and \( \nabla g_i \), respectively. The algorithm is robust to such stochastic errors as soon as they fulfilled, almost surely (a.s.),

\[
\sum_{n \in \mathbb{N}} \sqrt{\mathbb{E} \left[ \left\| a_n \right\|^2 \right]} X_n < +\infty, \quad \sum_{n \in \mathbb{N}} \sqrt{\mathbb{E} \left[ \left\| b_n \right\|^2 \right]} X_n < +\infty,
\]

\[
\sum_{n \in \mathbb{N}} \sqrt{\mathbb{E} \left[ \left\| c_n \right\|^2 \right]} X_n < +\infty, \text{ with } X_n = (x_n, v_n)(0 \leq n \leq \infty).
\]

**Algorithm 1: Randomized Block Primal-Dual Algorithm.**

**Initialization:** Let \((\lambda_n)_{n \in \mathbb{N}}\) be a sequence in \([0, 1]\) such that \(\inf_{n \in \mathbb{N}} \lambda_n > 0\).

Let \(x_0, (a_n)_{n \in \mathbb{N}}\) be \(H\)-valued random variables, and let \(v_0, (b_n)_{n \in \mathbb{N}}\), and \((c_n)_{n \in \mathbb{N}}\) be \(G\)-valued random variables.

**Iterations:**

for \(n = 0, 1, \ldots\)

Select randomly a vector \(\varepsilon_n = (\varepsilon_{1,n})_{1 \leq i \leq p + q} \in \mathcal{D}\)

for \(j = 1, \ldots, p\)

\[
\begin{align*}
\eta_j,n &= n_{p+1,k}, \quad k \in \mathbb{N}^+ \\
\sigma_j,n &= \eta_j,n, x_{j,n} - W_j \left( \nabla h_j(x_{j,n}) + a_{j,n} \right) \\
y_j,n &= \eta_j,n \left( s_{j,n} - W_j \left( \sum_{k \in \mathcal{L}} L_{k,j} x_{k,n} \right) \right)
\end{align*}
\]

for \(k = 1, \ldots, q\)

\[
\begin{align*}
u_k,n &= \varepsilon_{p+k,j} \left( \text{prox} U^{-1}_\varepsilon \left( v_k,n + U_k \left( \sum_{j \in \mathcal{L}} L_{k,j} y_{j,n} \right) - U_k(\nabla g_k(v_k,n) + c_k,n) \right) \right) + b_k,n \\
v_{k,n+1} &= v_{k,n} + \lambda_n \varepsilon_{p+k,j} (u_{k,n} - v_{k,n})
\end{align*}
\]

for \(j = 1, \ldots, p\)

\[
\begin{align*}
p_j,n &= \varepsilon_{n,j} \left( s_{j,n} - W_j \left( \sum_{k \in \mathcal{L}} L_{k,j} u_k,n \right) \right) \\
x_{j,n+1} &= x_{j,n} + \lambda_n \varepsilon_{j,n} (p_j,n - x_{j,n}).
\end{align*}
\]

3.2. Link with existing works

It can be noticed that, when \(p = 1\) and \((\forall n \in \mathbb{N}) \mathbb{P}[\varepsilon_{1,n} = 1] = 1\), Algorithm 1 is related to a number of existing primal-dual algorithms:

(i) The deterministic algorithms proposed in [18, 19] are recovered when \(q = 1, (\forall n \in \mathbb{N}) \mathbb{P}[\varepsilon_{2,n} = 1] = 1, H_1\), and \(G_1\) are finite dimensional spaces, \(l_1 = \ell(0)\), \(W_1 = \mathcal{P}r\) and \(U = \mathcal{P}r\) with \((r, p) \in [0, +\infty]^2, \) with no relaxation \((\lambda_n \equiv 1)\) or a constant one \((\lambda_n \equiv \lambda < 1)\) is performed, and in the absence of errors.

(ii) The deterministic fully parallel algorithm proposed in [9, Prop. 4.3] is recovered when \((\forall n \in \mathbb{N}) (\forall k \in \{1, \ldots, q\}) \mathbb{P}[\varepsilon_{p+k,n} = 1] = 1, \) and the errors \((a_n)_{n \in \mathbb{N}}, (b_n)_{n \in \mathbb{N}}\) and \((c_n)_{n \in \mathbb{N}}\) are deterministic and summable.

(iii) A different stochastic block primal-dual algorithm was recently proposed in [20] (see also [21] for some particular case) but this algorithm requires more restrictive convergence conditions than those provided in the next section.

Note that random block-coordinate algorithms already have been proposed in the literature. However, they are limited to the standard forward-backward algorithm [1, 14], or its dual version [22].

3.3. Convergence result

The following theorem, deduced from [23], guarantees that the sequence \((x_n, v_n)_{n \in \mathbb{N}}\) asymptotically provides a pair of primal-dual solutions to the problems (1) - (4).

**Theorem 3.1.** Suppose that the following assumptions hold:

(i) \((\varepsilon_n)_{n \in \mathbb{N}}\) are identically distributed \(\mathcal{D}\)-valued random variables and, for every \(n \in \mathbb{N}, \varepsilon_n\) and \(X_n\) are independent. In addition, \((\forall j \in \{1, \ldots, p\}) \mathbb{P}[\varepsilon_{j,0} = 1] > 0, \) and \((\forall k \in \{1, \ldots, q\}) \varepsilon_{p+k,n} = \max \{\varepsilon_{j,n}, j \in \mathcal{L}_k\}.

(ii) For every \(j \in \{1, \ldots, p\}\) and for every \(k \in \{1, \ldots, q\}\), \(h_j \in W_1^{1/2} \) and \(l_k \in U_1^{1/2}\) have Lipschitzian gradients with positive constants \(\mu_{j,1}^{-1} \) and \(\nu_k^{-1}\), respectively. Moreover.

\[
\min \left\{ \mu, \nu \left( 1 - \left\| U^{1/2}W^{1/2} \right\| \right) \right\} > 1/2.
\]

where \(L\) : \((x_j, 1)_{1 \leq j \leq p} \mapsto (\sum_{j=1}^{p} L_{k,j} x_j)_{1 \leq k \leq q}\) \(U\) : \((v_k)_{1 \leq k \leq q} \mapsto (U_1 v_1, \ldots, U_1 v_q)\), \(W\) : \((x_j)_{1 \leq j \leq p} \mapsto (W_1 x_1, \ldots, W_1 x_p)\), \(\mu = \min \{\mu_1, \ldots, \mu_p\}\), and \(\nu = \min \{\nu_1, \ldots, \nu_q\}.

Then, \((x_n)_{n \in \mathbb{N}}\) converges weakly a.s. to an \(F\)-valued random variable, and \((v_n)_{n \in \mathbb{N}}\) converges weakly a.s. to an \(F\)-valued random variable.

Note that in the case when, for every \(k \in \{1, \ldots, q\}\), \(l_k = \ell(0)\), Condition (5) simplifies into

\[
\left\| U^{1/2}W^{1/2} \right\| < 1 \quad \text{and} \quad \mu > 1/2.
\]

Finally, let us emphasize that the assumption we make on \((\varepsilon_n)_{n \in \mathbb{N}}\) is weak, so that our method benefits from a high flexibility on the way the blocks can be selected. Basically, almost any arbitrary stationary random sampling strategy fits into our framework.

4. APPLICATION TO 3D MESH DENOISING

4.1. Problem statement

We focus on the inverse problem, consisting of restoring the spatial positions \(x = (x_1, x_2, x_3)_{1 \leq j \leq p} \in \mathbb{R}^{p \times 3}\) of the nodes of a large 3D mesh from measurements \(z = (z^X, z^Y, z^Z)\) corrupted with an i.i.d. noise. The topology of the observed mesh \(x\) is assumed to be similar to the one of \(\hat{x}\) (i.e. both meshes have the same adjacency matrix).

We propose to define the estimate \(\hat{x} \in \mathbb{R}^{p \times 3}\) of \(\hat{x}\) as a solution to Problem (1) [24], where:}
variable constants satisfying Condition (6).

In our simulations, we simply take \( \tau \) to the metric induced by applying the Moreau decomposition theorem \([27]\). Moreover the proximity operators of functions \( g \) are positive for updating variable \( u_k \). Since, for every \( j \in \{1, \ldots, q\} \), \( g_j \) identifies with a separable sum of \( \ell_2 \) norms, \( \text{prox}_{g_j} \) has an explicit form \([4]\), from which the expression of \( \text{prox}_{g_j} \) can be deduced by applying the Moreau decomposition theorem \([27]\). Moreover the proximity operators of functions \( g \) are positive for updating variable \( u_k \). Since, for every \( j \in \{1, \ldots, q\} \), \( g_j \) identifies with a separable sum of \( \ell_2 \) norms, \( \text{prox}_{g_j} \) has an explicit form \([4]\), from which the expression of \( \text{prox}_{g_j} \) can be deduced by applying the Moreau decomposition theorem \([27]\).

Finally, when one deals with a very large amount of data, memory requirement becomes a more limiting factor than the computational time. This implies that only a limited number of variables can be handled at each iteration of the algorithm. At each iteration \( n \in \mathbb{N} \), the activation variables \( (\xi_{j,n})_{1 \leq j \leq p} \) must thus be chosen so as to satisfy some budget constraint, which can be expressed as \( \sum_{j=1}^{p} \xi_{j,n} = r \), with \( r \leq p \). To this aim, we propose to divide the considered mesh into \( p/r \) non-overlapping blocks, and to set \( (\xi_{j,n})_{1 \leq j \leq p, n \in \mathbb{N}} \) so as to activate a single block chosen randomly at every iteration.

4.3. Numerical results

To show the performance of the proposed method, we consider a large 3D mesh, namely the standard Dragon available at \http://graphics.stanford.edu/data/3Dscanrep/\ with \( p = 100250 \) nodes and 301207 edges, displayed in Figure 1(a). The positions of the original mesh are corrupted through a Gaussian mixture noise model. The resulting observed mesh \( \mathbf{z} \) is displayed in Figure 1(b). The restored mesh obtained using Algorithm 1, with \( \lambda_n \equiv 1 \), is represented in Figure 1(c). We choose the regularization parameters \( (\beta, \delta) \) so as to minimize the Mean Square Error (MSE). For comparison, Figure 1(d) displays the estimated mesh obtained using Laplacian smoothing \([28]\).

Figure 2 illustrates the time necessary to reach the stopping criterion \( \|x_n - \hat{x}\| < 10^{-3} \|\hat{x}\| \), where \( \hat{x} \) has been pre-computed using a large number of iterations, as a function of the number of blocks \( p/r \). The memory space required for running the algorithm in Matlab R2013a is also reported. As one could expect, the best reconstruction time (equal to 5 s.) is obtained for \( r = p \), i.e. with a deterministic strategy. The choice of smaller blocks leads to a decrease of the occupied memory, at the expense of a decrease of the convergence speed. A major advantage of our method is that it allows the user to select the random sampling strategy leading to a minimal reconstruction time, without exceeding the available memory on its own computer architecture.

![Fig. 1.](image-url) (a) Original mesh \( \mathbf{x} \), (b) noisy mesh \( \mathbf{z} \) with MSE = 2.89 \times 10^{-6}, and reconstructed meshes \( \mathbf{x} \) using (c) Algorithm 1 with MSE = 8.09 \times 10^{-8} and (d) Laplacian smoothing with MSE = 5.23 \times 10^{-7}.
In this paper, we proposed a random block-coordinate version of a primal-dual algorithm for solving minimization problems involving convex (possibly non differentiable) functions and linear operators, whose convergence is established. Our approach generalizes existing primal-dual methods, by allowing to update randomly, at each iteration, only a subset of the components of the iterates. This is particularly useful for the resolution of large scale problems with limited allocated memory. The good performance of our method has been illustrated through an application on a large 3D mesh denoising problem. As an extension of our method, it could be interesting to combine the proposed algorithm to a variable metric strategy [27] in order to accelerate its convergence.

5. CONCLUSION

In this paper, we proposed a random block-coordinate version of a primal-dual algorithm for solving minimization problems involving convex (possibly non differentiable) functions and linear operators, whose convergence is established. Our approach generalizes existing primal-dual methods, by allowing to update randomly, at each iteration, only a subset of the components of the iterates. This is particularly useful for the resolution of large scale problems with limited allocated memory. The good performance of our method has been illustrated through an application on a large 3D mesh denoising problem. As an extension of our method, it could be interesting to combine the proposed algorithm to a variable metric strategy [27] in order to accelerate its convergence.

REFERENCES