TWO DISTRIBUTED ALGORITHMS FOR THE DECONVOLUTION OF LARGE RADIO-INTERFEROMETRIC MULTISPECTRAL IMAGES

Céline Meillier, Pascal Bianchi, Walid Hachem

LTCI, CNRS, Telecom Paristech, Université Paris-Saclay, 75013, Paris, France

ABSTRACT

We address in this paper the deconvolution issue for radio-interferometric multispectral images. Whereas this problem has been widely explored in the recent literature for single images, a few algorithms are able to reconstruct multispectral images (three-dimensional images) [1], [2]. We propose in this paper two new distributed algorithms based on the optimization methods ADMM and projected gradient (PG) for the reconstruction of radio-interferometric multispectral images. We present an original distributed architecture and a comparison of their performance on a quasi-real data cube.

Index Terms— ADMM, deconvolution, distributed optimization, projected gradient, radio-interferometry, multispectral images.

1. INTRODUCTION

With the advent of new generations of radio interferometers such as the Low Frequency Array (LOFAR) and the Square Kilometer Array (SKA), a large amount of multispectral images will be produced in the next few years. These new interferometers have a very large field of view (millions of pixels) with a high spectral resolution (hundreds of frequency bands in the radio wave domain). These massive observed data are corrupted by the noise and by the instrument response. A challenging point is the design of deconvolution algorithms that are able to deal with the large size of the observations. A good deal of the recent research is focused on distributed optimization algorithms aiming to solve the deconvolution issue in two cases:

• Large two-dimensional monochromatic (i.e., with only one spectral band) image deconvolution with different approaches: let us cite the so-called PURIFY algorithm [3] based on the Simultaneous Direction Method of Multipliers (SDMM), [4] where the authors propose in particular two new scalable splitting algorithms for image reconstruction, or [5] where a compressive sensing approach is proposed.

• The extension to the multispectral (3D) images bearing two spatial dimensions and a spectral one. The third dimension obviously increases the size of the deconvolution problem. The authors of [2] propose a method based on the Alternating Direction Method of Multipliers (ADMM) that is known amongst other things for its ability to be distributed [6].

This work starts with the regularized optimization problem described in [2] and recalled in Section 2. In Section 3, a simpler version of ADMM than in [2] is proposed by removing one Lagrange multiplier vector. Concurrently, we also propose to solve the optimization problem by resorting to the projected gradient method (PG) in the dual space (Section 4). Distributed implementations for both these algorithms are proposed in Section 5. This distributed architecture differs from the classical master/slave one by the fact that none of the nodes stores the entire multispectral image (in classical architecture, the global image is stored on the master node). Thanks to a minimal amount of data exchange between the nodes, the required memory and the computational cost supported by a single node are decreased. The performances of the two algorithms are finally compared in Section 6.

2. PROBLEM FORMULATION

Assuming that the observations are made on \( L \) frequency bands, the 3D data cube can be seen as a collection of \( L \) monochromatic images of \( N \) pixels each. Similar to [2], the image observed at the frequency \( \nu_l \in \{\nu_1, \cdots, \nu_L\} \), also called the “dirty image” at \( \nu_l \), is given by the equation

\[
y_l = H_l x_l + n_l \in \mathbb{R}^N
\]

where \( x_l \in \mathbb{R}^N \) is the “true” image vector, \( n_l \) is the noise vector and \( H_l \) is a convolution matrix representing the so-called Point Spread Function (PSF) of the radio-interferometer. Stacking the \( L \) dirty images in the vector \( y = [y_1^T, \cdots, y_L^T]^T \in \mathbb{R}^M \) where \( M = N \times L \), and denoting respectively as \( \| \cdot \|_2 \) and \( \| \cdot \|_1 \) the Euclidean and the \( \ell_1 \) norms, the optimization problem is written

\[
\min_{x} \frac{1}{2} \| y - Hx \|_2^2 + \frac{\mu}{2} \| x \|_2^2 + \tau g(x) + \| Wx \|_1,
\]

where the first term is the objective function, the second is a Tikhonov regularization term controlled by the parameter
\(\mu_c > 0\), the third is a positivity constraint (since we are recovering sky brightnesses) where \(\tau_{2,\infty}(x) = \infty\) if one of the elements of \(x\) at least is negative, and 0 otherwise, and the last term is a sparsity term. Here, the sparsity is induced in the multiresolution (wavelet) domain for each monochromatic image, and in the Discrete Cosine Transform (DCT) domain at each multifrequency pixel. Specifically,

\[
W = \begin{pmatrix}
\mu_s \mathbf{W}_s & \\
- \mu_v \mathbf{W}_v & 0
\end{pmatrix} \in \mathbb{R}^{m_s M \times M}
\]

where \(\mu_s, \mu_v > 0\) are regularization parameters, and

\[
\mathbf{W}_s = \begin{pmatrix}
\mathbf{W}_s & \\
& \ddots & \\
& & \mathbf{W}_s
\end{pmatrix} \in \mathbb{R}^{m_s N \times N}
\]

is a block-diagonal matrix where each block \(\mathbf{W}_s \in \mathbb{R}^{m_s N \times N}\), acting on a monochromatic image, is the concatenation of \(m_s\) orthogonal wavelet bases. Similar to \([2]\) and \([3]\), we identify \(\mathbf{W}_s\) with a dictionary consisting in the concatenation of the first eight Daubechies wavelet bases \((m_s = 8)\). Finally,

\[
\mathbf{W}_v = \begin{pmatrix}
\mathbf{W}_v & \\
& \ddots & \\
& & \mathbf{W}_v
\end{pmatrix} \mathbf{P} \in \mathbb{R}^{M \times M}
\]

where \(\mathbf{P}\) is the permutation matrix that rearranges the elements of the vector \(x\) pixel by pixel, each of these pixels being represented by a vector of \(L\) frequencies, and where each block \(\mathbf{W}_v\) is a \(L \times L\) matrix representing the DCT. Note that \(\mathbf{W}_s^T \mathbf{W}_s = m_s \mathbf{I}_M\) and \(\mathbf{W}_v^T \mathbf{W}_v = \mathbf{I}_M\). It is moreover obvious that \(\|\mathbf{W}x\|_1 = \mu_s \|\mathbf{W}_s x\|_1 + \mu_v \|\mathbf{W}_v x\|_1\).

### 3. ADMM DESCRIPTION

In order to solve Problem (2), we reformulate it as follows:

\[
\min_x f(x) + g(z) \quad \text{subject to:} \quad \mathbf{A}x + \mathbf{B}z = \mathbf{0}
\]

where

\[
f(x) = (1/2)\|y - \mathbf{H}x\|_2^2 + (\mu_c/2)\|x\|_2^2,
\]

\[
g(z) = \tau_{2,\infty}(p) + \mu_s \|t\|_1 + \mu_v \|v\|_1
\]

where

\[
\mathbf{z}^T = (\mathbf{p}^T, \mathbf{t}^T, \mathbf{v}^T) \in \mathbb{R}^M \times \mathbb{R}^{m_s M} \times \mathbb{R}^M,
\]

\[
\mathbf{A} = \begin{pmatrix}
\mathbf{I}_M \\
\mathbf{W}_s \\
\mathbf{W}_v
\end{pmatrix}, \quad \text{and } \mathbf{B} = \begin{pmatrix}
\mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{0} & -\mathbf{I}_{m_s M} & \mathbf{0} \\
\mathbf{0} & \mathbf{0} & -\mathbf{I}_M
\end{pmatrix}.
\]

The associated augmented Lagrangian for \(\rho > 0\) is

\[
\begin{align*}
\mathcal{L}_\rho(x, z, \gamma) &= f(x) + g(z) \\
&\quad + \gamma_p^T (x - p) + \gamma_t^T (\mathbf{W}_s x - t) + \gamma_v^T (\mathbf{W}_v x - v) \\
&\quad + \frac{\rho}{2} \|x - p\|_2^2 + \frac{\rho}{2} \|\mathbf{W}_sx - t\|_2^2 + \frac{\rho}{2} \|\mathbf{W}_vx - v\|_2^2
\end{align*}
\]

where \(\gamma = [\gamma_p^T, \gamma_t^T, \gamma_v^T]^T \in \mathbb{R}^{(2 + m_s)M}\) is the vector of Lagrange multipliers, decomposed in accordance with the right hand side of (3). Note that the dimension of this vector is smaller than in \([2]\), where four sets of Lagrange multipliers were used instead of three here.

As it is well known, ADMM consists of the following iterations:

\[
\begin{align*}
x^{k+1} &= \arg\min_x \mathcal{L}_\rho(x, z^k, \gamma^k) \\
z^{k+1} &= \arg\min_z \mathcal{L}_\rho(x^{k+1}, z, \gamma^k) \\
\gamma^{k+1} &= \gamma^k + \rho (\mathbf{A}x^{k+1} + \mathbf{B}z^{k+1})
\end{align*}
\]

We now write \(x^k = [x_1^k, \ldots, x_L^k]^T\) where each block is of size \(N\) and thus corresponds to a monochromatic image. We do the same decomposition for \(p^k\) and \(z^k\). Similar decompositions are also done for \(t^k\) and \(v^k\) where the dimensions of the blocks are this time equal to \(m_s N\).

Let us consider the minimization (4). Solving the equation shows that the minimization w.r.t. \(x\) is separable with respect to the frequencies thanks to the block diagonal structure of \(\mathbf{W}_s\) and to the orthogonality of \(\mathbf{W}_v\). After a straightforward computation, we obtain that \(x_{l+1}^k = Q_l^{-1}b_l^k\) for each \(l \in \{1, \ldots, L\}\) where

\[
Q_l = \mathbf{H}_l^T \mathbf{H}_l + (\mu_c + (2 + m_s)\rho) \mathbf{I}_N \quad \text{and} \quad b_l^k = \mathbf{H}_l^T y_l - \gamma_{p,l}^k - \mathbf{W}_s^T (\gamma_{t,l}^k - \rho t_l^k) + \rho \left( p_l^k + \left( \mathbf{p}^l (\mathbf{W}_v^T \gamma_v^k) \right)_l \right)_l.
\]

Here \((\cdot)_l\) denotes the \(l^{th}\) size-\(N\) block of a vector. The computations leading to the expression of \(Q_l\) make use of the isometric nature of \(\mathbf{W}_s\). Observe that the \(Q_l\) can be computed once at the beginning of the algorithm while the vectors \(b_l^k\) need to be computed at every iteration. Finally, since \(\mathbf{H}_l\) is a convolution operator, the equation \(Q_l^{-1}b_l^k\) can be practically approximated by using the Fast Fourier Transform operator.

Minimizations w.r.t. the vectors \(p^k, t^k\) and \(v^k\) in (5) are structurally separable with respect to the frequencies. Writing \(\hat{p}_l^{k+1} = \rho^{-1} \gamma_{p,l}^k + x_{l+1}^k\), we get

\[
\begin{align*}
p_l^{k+1} &= \arg \min_{u \in \mathbb{R}^M} \tau_{2,\infty}(u) + \gamma_{p,l}^k \mathbf{H}_l^T (x_{l+1}^k - u) \\
&+ \frac{\rho}{2} \|x_{l+1}^k - u\|_2^2
\end{align*}
\]

where \(\max\) is taken elementwise.
Writing $t_{k+1} = W_s x_{k+1}^p + \rho^{-1} \gamma_{k,t}$, we also have:

$$t_{k+1} = \arg \min_{u \in \mathbb{R}^{m \times M}} \mu_s \|u\|_1 + \frac{\rho}{2} \|u - t_{k+1}^p\|_2^2$$

$$= \gamma_{k,t} \cdot \max(0, 1 - \frac{\rho^{-1} \mu_s}{\|t_{k+1}^p\|_2})$$

where $\cdot$ is the elementwise product. We recognize here the usual soft thresholding operator.

The update of the vector $v^k$ is done at each multifrequency pixel. Write this time $v^k = [v_1^T, \ldots, v_N^T]^T$ and $\gamma_{v} = [\gamma_{v,1}^T, \ldots, \gamma_{v,1}^T]^T$ where the blocks within these vectors have the size $L$. Let $\tilde{v}_{k+1}^i = (\tilde{W}_p x_{k+1}^p)_i + \rho^{-1} \gamma_{v,i}^k$ where $(\cdot)_i$ is a size-$L$ block, we get:

$$v_{k+1}^i = \arg \min_{u \in \mathbb{R}^L} \mu_v \|u\|_1 + \frac{\rho}{2} \|u - \tilde{v}_{k+1}^i\|_2^2$$

$$= \tilde{v}_{k+1}^i \cdot \max(0, 1 - \frac{\rho^{-1} \mu_v}{\|\tilde{v}_{k+1}^i\|_2})$$

Finally, the inspection of Equation (6) shows that $\gamma_{v}^k$ and $\gamma_{v}^k$ are updated at the level of the monochromatic images while $\gamma_{v}^k$ is updated at the multifrequency pixel level.

4. PROJECTED GRADIENT ON THE DUAL PROBLEM

4.1. Primal and dual problems

In this section, we replace Problem (2) with the problem:

$$\min_{x \in \mathbb{R}^M} f(x) + h(Wx), \quad (8)$$

where we recall that $f(x) = (1/2)\|y - Hx\|_2^2 + (\mu_r/2)\|x\|_2^2$, and where we set $h: \mathbb{R}^{(m+1)M} \rightarrow \mathbb{R}_+$, $u \mapsto \|u\|_1$. Note that the positivity assumption is now absent. In order to solve this problem, we start by writing its dual problem:

$$\min_{\lambda \in \mathbb{R}^{(m+1)M}} f^*(\gamma - W^T \lambda) + h^*(\lambda) \quad (9)$$

where $f^*$ is the Legendre-Fenchel transform of $f$, defined as $f^*(\phi) = \sup_{x} \langle x, \phi \rangle - f(x)$. After a standard calculation, we get:

$$f^*(\phi) = \frac{1}{2} \phi^T \Delta^{-1} \phi + \frac{1}{\Delta} \phi^T (H^T I - H) y$$

where $\Delta = (H^T H + \mu_s I)$ is a block diagonal matrix of size $M \times M$ with Toeplitz blocks, and

$$h^*(\lambda) = \mathbb{1}_{B_\infty}(\lambda)$$

where $B_\infty = \{ \lambda \in \mathbb{R}^{(m+1)M} : \|\lambda\|_\infty \leq 1 \}$ is the unit ball for the $\|\cdot\|_\infty$ norm.

The inspection of $f$ and $h$ shows that the qualification conditions for the duality gap to be zero hold. Moreover, the saddle point $(x^*, \lambda^*)$ satisfies $x^* = \nabla f^*(-W^T \lambda^*)$ where $\nabla f^*$ is the gradient of $f^*$, given by:

$$\nabla f^*(\phi) = \Delta^{-1} (\phi + Hy). \quad (10)$$

4.2. Solving the dual problem using PG

The dual problem (9) can be reformulated as:

$$\min_{x \in \mathbb{R}^{(m+1)M}} f^*(-W^T \lambda)$$

Since $f^*$ is smooth, this problem can be solved with the help of PG (see e.g. [7]). In our context, this algorithm reads:

$$\lambda_{k+1} = P_{\infty} \left( \lambda_k + \rho W \nabla f^*(-W^T \lambda_k) \right)$$

$$= P_{\infty} \left( \lambda_k - \rho W \Delta^{-1}(W^T \lambda_k - Hy) \right)$$

where $\rho > 0$ and $P_{\infty}$ is the projection operator on $B_\infty$, being the proximity operator of $h^*$. At the last iteration $k_{\text{max}}$, the 3D cube $\bar{x}_{k_{\text{max}}}^m$ is recovered according to the equation:

$$x_{k_{\text{max}}}^m = \nabla f^*(-W^T \lambda_{k_{\text{max}}}) = \Delta^{-1} (H^T y - W^T \lambda_{k_{\text{max}}})$$

In order to provide a distributed implementation of this algorithm, we write $\lambda^k = [\lambda_1^T, \lambda_2^T]^T$ where $\lambda_1^k \in \mathbb{R}^{m \times M}$ is processed at the level of the monochromatic images and $\lambda_2^k \in \mathbb{R}^M$ is processed at the level of the multifrequency pixels. Details are provided in the next section.

5. DISTRIBUTED ARCHITECTURE

Implementation of deconvolution algorithms on a distributed architecture is needed for memory charge reasons; SKA multispectral data cubes are expected to be 80 tera bytes.

5.1. Structure of the cluster

From the 2D + 1D structure of the multispectral data and the spatial and spectral sparsity constraints of the minimization problems (2) and (8), the variables $x, p, t, \gamma_{v}, \gamma_{t}, \lambda_1$ (resp. $v, \gamma_{v}, \gamma_{p}$) can be evaluated only on monochromatic images (resp. on pixels). All of these calculations can be done with parallel programming w.r.t. the frequencies (resp. the pixels). We use a cluster of machines that we divide into two groups: one for the calculations w.r.t. the frequencies (group A), the other for the calculations w.r.t. the pixels (group B). Figure 1 illustrates the two groups of nodes architecture and exchanges between nodes. For the sake of simplicity, in figure 1, we assume there are as many nodes in group A (resp. in group B) as frequency bands (resp. pixels) in the multispectral image. Note that in real implementation, each node is in charge of several pixels or pixels, depending on the capacity of the cluster and the image size.

5.2. Distributed implementation of ADMM and PG

Algorithm 1 summarizes the distribution of the alternated updating steps of the primal and dual variables described in section 3 for the ADMM algorithm. All the calculations are distributed on the two groups of nodes architecture introduced in the previous paragraph w.r.t. the frequencies (resp. w.r.t.
Algorithm 1 Distributed ADMM algorithm

Initialize $x$, $p$, $t$, $v$, $\gamma_p$, $\gamma_t$, $\gamma_v$

while $\delta_x \geq 10^{-5}$ do

for nodes in group A do

Evaluate $b_i^{k+1}$

Solve $Q_i x_i^{k+1} = b_i^{k+1}$

Send $x_i^{k+1}$ to nodes of group B

end for

MPI synchronization barrier

for nodes in group A do

$p_i^{k+1} = \max(0, p_i^{k+1})$

$t_i^{k+1} = \max(b_i^{k+1}, 0, 1 - \frac{\rho^{-1} p_i^{k+1}}{|\nabla f_i|})$

$\gamma_{p,i}^{k+1} = \gamma_{p,i}^{k} + \rho p_i^{k+1}$

$\gamma_{t,i}^{k+1} = \gamma_{t,i}^{k} + \rho (t_i^{k+1} - b_i^{k+1})$

end for

MPI synchronization barrier

for nodes in group B do

$v_i^{k+1} = \max(0, 1 - \frac{\rho^{-1} \mu_v}{|\nabla f_i|})$

$\gamma_{v,i}^{k+1} = \gamma_{v,i}^{k} + \rho (v_i^{k+1} - b_i^{k+1})$

Send $v_i^{k+1}$ and $\gamma_{v,i}^{k+1}$ to nodes of group A

end for

MPI synchronization barrier

$k = k + 1$

end while

return $x^{k-1}$

6. RESULTS AND CONCLUSION

The two algorithms are tested on a quasi-real multispectral image of size $256 \times 256 \times 100$ pixels. We use the radio emission image from an HII region of the M31 galaxy (figure 2 on the left) as the reference image. We build a multispectral image by applying a sine wave spectrum to each pixel of the image. The dirty image (figure 2 on the right) is obtained by convolution with a 2D Gaussian PSF and corrupted by a white Gaussian noise whose signal-to-noise ratio (SNR) is from 15dB to 25dB according to the frequencies.

![Fig. 1: Illustration of the distributed architecture. Nodes of group A (orange blocks) do parallel calculation w.r.t. the frequencies, while nodes of group B (blue blocks) do parallel calculation w.r.t. the pixels.](image)

![Fig. 2: Left: reference image of the M31 galaxy. Right: Dirty image at a given frequency (SNR = 25dB).](image)
Algorithm 2 Distributed PG for dual problem

Initialization

for nodes in group A do
  Eval. $\Delta^{-1}H^Ty$
  Send $\Delta^{-1}H^Ty$ to B
  Evaluate and save $\mu_sW_s\Delta^{-1}H^Ty$
end for

for nodes in group B do
  Receive $\Delta^{-1}H^Ty$ from A
  Evaluate and save $\mu\nu\tilde{W}_\nu\Delta^{-1}H^Ty$
end for

while $\delta^k_s \geq 10^{-5}$ do
  for nodes in group A do
    Eval. $\mu_s\tilde{W}_s^k\lambda^k_s$
  end for

  for nodes in group B do
    Eval. $\mu\nu\tilde{W}_\nu^k\lambda^k_\nu$
    Send $\mu\nu\tilde{W}_\nu^k(\lambda^k_\nu)$ to A
  end for

  MPI synchronization barrier

  for nodes in group A do
    Receive $\mu\nu\tilde{W}_\nu^k\lambda^k_\nu$ from B
    Eval. $\Delta^{-1}\left(\mu_s\tilde{W}_s^k\lambda^k_s + \mu\nu\tilde{W}_\nu^k\lambda^k_\nu\right)$
    Send $\Delta^{-1}\left(\mu_s\tilde{W}_s^k\lambda^k_s + \mu\nu\tilde{W}_\nu^k\lambda^k_\nu\right)$ to B
    Eval. $\theta_s = \mu_s\tilde{W}_s\Delta^{-1}\left(\mu_s\tilde{W}_s^k\lambda^k_s + \mu\nu\tilde{W}_\nu^k\lambda^k_\nu\right)$
    $\lambda^k_{s+1} \leftarrow P_{\infty}(\lambda^k_s - \rho(\theta_s - \mu_s\tilde{W}_s\Delta^{-1}H^Ty))$
  end for

  for nodes in group B do
    Receive $\Delta^{-1}\left(\mu_s\tilde{W}_s^k\lambda^k_s + \mu\nu\tilde{W}_\nu^k\lambda^k_\nu\right)$ from A
    Eval. $\theta_\nu = \mu\nu\tilde{W}_\nu\Delta^{-1}\left(\mu_s\tilde{W}_s^k\lambda^k_s + \mu\nu\tilde{W}_\nu^k\lambda^k_\nu\right)$
    $\lambda^k_{\nu+1} \leftarrow P_{\infty}(\lambda^k_\nu - \rho(\theta_\nu - \mu\nu\tilde{W}_\nu\Delta^{-1}H^Ty))$
  end for

  $k = k+1$
end while

return $\lambda^{k-1}$

This helped to test the algorithms before applying them to real data of size $(2048 \times 2048 \times 256)$ that should be soon available. We also would like to modify the minimization problem formulation to try a smoother way for imposing the positivity on the reconstructed image.

7. ACKNOWLEDGEMENT

We thank Maxime Flauder for useful suggestions regarding the distributed implementation.

REFERENCES


