A Block Parallel Majorize-Minimize Memory Gradient Algorithm

Emilie Chouzenoux, LIGM, UPEM

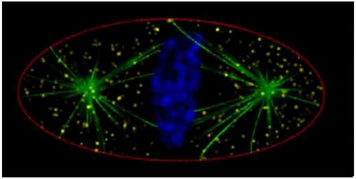
(joint work with *Sara Cadoni, Jean-Christophe Pesquet and Caroline Chaux)*

Séminaire Parisien des Mathématiques Appliquées à l'Imagerie

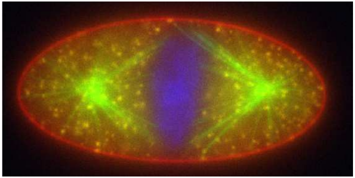
Institut Henri Poincaré

3 November 2016

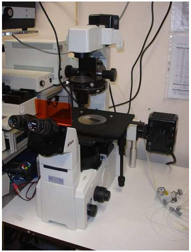
Inverse problems and large scale optimization



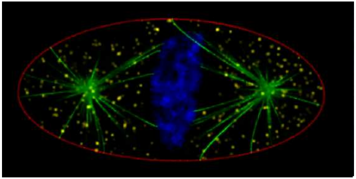
Original image



Degraded image

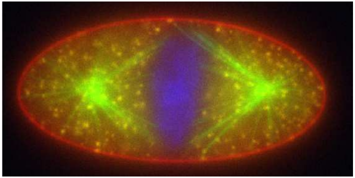


Inverse problems and large scale optimization



Original image

$\overline{\boldsymbol{x}} \in \mathbb{R}^N$



Degraded image

$\boldsymbol{y} = \mathcal{D}(\boldsymbol{H}\overline{\boldsymbol{x}}) \in \mathbb{R}^M$

- $\boldsymbol{H} \in \mathbb{R}^{M \times N}$: matrix associated with the degradation operator.
- $\mathcal{D}\colon \mathbb{R}^M \to \mathbb{R}^M$: noise degradation.

How to find a good estimate of $\overline{\boldsymbol{x}}$ from the observations $\boldsymbol{y}$ and the model $\boldsymbol{H}$ in the context of large scale processing?

Inverse problems and large scale optimization

Variational approach:

An image estimate $\hat{\boldsymbol{x}} \in \mathbb{R}^N$ is generated by minimizing

$$(\forall \boldsymbol{x} \in \mathbb{R}^N) \quad F(\boldsymbol{x}) = \sum_{s=1}^{S} f_s(\boldsymbol{L}_s \boldsymbol{x})$$

with $f_s : \mathbb{R}^{P_s} \to \mathbb{R}$, $\boldsymbol{L}_s \in \mathbb{R}^{P_s \times N}, P_s > 0$.

In the context of maximum a posteriori estimation :

- $\boldsymbol{L}_1$: Degradation operator, i.e. $\boldsymbol{H}$;
- f_1: Data fidelity (e.g. least squares);
- $(f_s)_{2 \leqslant s \leqslant S}$: Regularization functions on some linear transforms $(\boldsymbol{L}_s)_{2 \leqslant s \leqslant S}$ of the sought solution.

$\rightarrow$ Often no closed form expression or solution expensive to compute (especially in large scale context).

▶ **Need for an efficient iterative minimization strategy !**

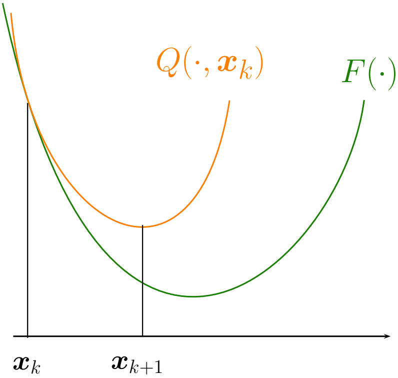
Outline

* **Majorize-Minimize Memory Gradient algorithm**
  - Majorize-Minimize principle
  - Subspace acceleration
  - Convergence theorem
* **Block parallel 3MG algorithm**
  - Block alternating 3MG
  - Block separable majorant
  - Practical implementation
  - Convergence theorem
* **Application to 3D deconvolution**
  - Variational approach
  - Parallel implementation
  - Numerical results

Majorize-Minimize Memory Gradient algorithm

Majorize-Minimize principle

1. **Find a tractable surrogate for** $F \rightsquigarrow$ **M**ajorization step



Majorize-Minimize principle

1. **Find a tractable surrogate for F $\rightsquigarrow$** Majorization step

$\rightsquigarrow$ Quadratic tangent majorant of F at $\boldsymbol{x}_k$

$$(\forall \boldsymbol{x} \in \mathbb{R}^N) \quad Q(\boldsymbol{x}, \boldsymbol{x}_k) = F(\boldsymbol{x}_k) + \nabla F(\boldsymbol{x}_k)^\top (\boldsymbol{x} - \boldsymbol{x}_k) + \frac{1}{2}(\boldsymbol{x} - \boldsymbol{x}_k)^\top \boldsymbol{A}(\boldsymbol{x}_k)(\boldsymbol{x} - \boldsymbol{x}_k)$$

where, for every $\boldsymbol{x} \in \mathbb{R}^N$, $\boldsymbol{A}(\boldsymbol{x}) \in \mathbb{R}^{N \times N}$ is a symmetric definite positive matrix such that

$$(\forall \boldsymbol{x} \in \mathbb{R}^N) \quad Q(\boldsymbol{x}, \boldsymbol{x}_k) \geqslant F(\boldsymbol{x}).$$

$\ast$ Several methods available to construct matrix $\boldsymbol{A}(\boldsymbol{x})$ in the context of inverse problems in image processing.

Subspace acceleration

2. Minimize in a subspace $\rightsquigarrow$ Minimization step

$$(\forall k \in \mathbb{N}^*) \qquad \boldsymbol{x}_{k+1} \in \underset{\boldsymbol{x} \in \operatorname{ran} \boldsymbol{D}_k}{\operatorname{Argmin}} \; Q(\boldsymbol{x}, \boldsymbol{x}_k),$$

with $\boldsymbol{D}_k \in \mathbb{R}^{N \times M_k}$.

- $\operatorname{ran} \boldsymbol{D}_k = \mathbb{R}^N \Rightarrow$ half-quadratic algorithm.
- M_k small $\Rightarrow$ low-complexity per iteration.

Memory-Gradient subspace:

$$\boldsymbol{D}_k = \begin{cases} [-\nabla F(\boldsymbol{x}_k), \boldsymbol{x}_k - \boldsymbol{x}_{k-1}] & \text{if } k \geqslant 1 \\ -\nabla F(\boldsymbol{x}_0) & \text{if } k = 0 \end{cases}$$

$\rightsquigarrow$ **3MG algorithm**

(similar ideas in NLCG, L-BFGS, TWIST, FISTA, ...)

3MG algorithm

Initialize $\boldsymbol{x}_0 \in \mathbb{R}^N$
For $k = 0, 1, 2, \ldots$
  Compute $\nabla F(\boldsymbol{x}_k)$
  If $k = 0$
    $\boldsymbol{D}_k = -\nabla F(\boldsymbol{x}_0)$
  Else
    $\boldsymbol{D}_k = [-\nabla F(\boldsymbol{x}_k), \boldsymbol{x}_k - \boldsymbol{x}_{k-1}]$
  $\boldsymbol{S}_k = \boldsymbol{D}_k^\top \boldsymbol{A}(\boldsymbol{x}_k)\boldsymbol{D}_k$
  $\boldsymbol{u}_k = \boldsymbol{S}_k^\dagger \boldsymbol{D}_k^\top \nabla F(\boldsymbol{x}_k)$
  $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \boldsymbol{D}_k \boldsymbol{u}_k$

⇝ Low computational cost since $\boldsymbol{S}_k$ is of dimension $M_k \times M_k$, with $M_k \in \{1, 2\}$.

⇝ Complexity reductions possible by taking into account the structures of F and $\boldsymbol{D}_k$.

Convergence theorem

Let assume that:

1. $F : \mathbb{R}^N \to \mathbb{R}$ is a coercive, differentiable function.
2. There exists $(\underline{\nu}, \overline{\nu}) \in]0, +\infty[^2$ such that $(\forall k \in \mathbb{N})$ $\underline{\nu}\,\mathbf{Id} \preceq \boldsymbol{A}(\boldsymbol{x}_k) \preceq \overline{\nu}\,\mathbf{Id}$,

Then, the following hold:

- $\|\nabla F(\boldsymbol{x}_k)\| \to 0$ and $F(\boldsymbol{x}_k) \searrow F(\widehat{\boldsymbol{x}})$ where $\widehat{\boldsymbol{x}}$ is a critical point of F.
- If F is convex, any sequential cluster point of $(\boldsymbol{x}_k)_{k\in\mathbb{N}}$ is a minimizer of F.
- If F is strongly convex, then $(\boldsymbol{x}_k)_{k\in\mathbb{N}}$ converges to the unique (global) minimizer $\widehat{\boldsymbol{x}}$ of F
- If F satisfies the Kurdyka-Łojasiewicz inequality, then the sequence $(\boldsymbol{x}_k)_{k\in\mathbb{N}}$ converges to a critical point of F.

3MG in practical situations

3MG algorithm outperforms state-of-the arts optimization algorithms in many image processing applications.

Problem: **Computational issues with very large-size problems.**

Main reasons:

- High computational time for calculating the gradient direction $\nabla F(\boldsymbol{x}_k)$ and the matrix $\boldsymbol{S}_k = \boldsymbol{D}_k^\top \boldsymbol{A}(\boldsymbol{x}_k)\boldsymbol{D}_k$;
- High storage cost for $\nabla F(\boldsymbol{x}_k)$, $\boldsymbol{D}_k$ and $\boldsymbol{x}_k$.

$\downarrow$

Block parallel approach

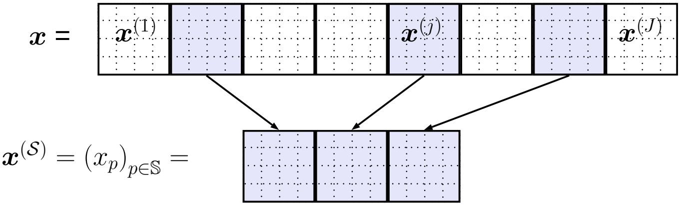
Block parallel 3MG algorithm

Block parallel strategy

The vector of unknowns x is partitioned into **block subsets**.
At each iteration, **some** blocks are updated in **parallel**.

Advantages:

- Control of the memory thanks to the block alternating strategy;
- Reduction of the computational time thanks to the parallel procedure.



Block alternating 3MG

1. **Select a block subset**: Choose a non empty $\mathcal{S}_k \subset \{1,\dots,J\}$.

Block alternating 3MG

1. Select a block subset: Choose a non empty $\mathcal{S}_k \subset \{1, \dots, J\}$.

2. Find a tractable surrogate in this subset:

⇝ Set $\boldsymbol{A}^{(\mathcal{S}_k)}(\boldsymbol{x}_k) = ([\boldsymbol{A}(\boldsymbol{x}_k)]_{p,p})_{p \in \mathbb{S}_k}$. The restriction of F to $\mathcal{S}_k$ is majorized at $\boldsymbol{x}_k$ by

$$(\forall \boldsymbol{v} \in \mathbb{R}^{|\mathbb{S}_k|}) \quad Q^{(\mathcal{S}_k)}(\boldsymbol{v}, \boldsymbol{x}_k) = F(\boldsymbol{x}_k) + \nabla F^{(\mathcal{S}_k)}(\boldsymbol{x}_k)^\top (\boldsymbol{v} - \boldsymbol{x}_k^{(\mathcal{S}_k)}) + \frac{1}{2}(\boldsymbol{v} - \boldsymbol{x}_k^{(\mathcal{S}_k)})^\top \boldsymbol{A}^{(\mathcal{S}_k)}(\boldsymbol{x}_k)(\boldsymbol{v} - \boldsymbol{x}_k^{(\mathcal{S}_k)}).$$

Block alternating 3MG

1. **Select a block subset:** Choose a non empty $\mathcal{S}_k \subset \{1, \ldots, J\}$.
2. **Find a tractable surrogate in this subset:**

⇝ Set $\boldsymbol{A}^{(\mathcal{S}_k)}(\boldsymbol{x}_k) = ([\boldsymbol{A}(\boldsymbol{x}_k)]_{p,p})_{p \in \mathbb{S}_k}$. The restriction of F to $\mathcal{S}_k$ is majorized at $\boldsymbol{x}_k$ by

$$(\forall \boldsymbol{v} \in \mathbb{R}^{|\mathbb{S}_k|}) \quad Q^{(\mathcal{S}_k)}(\boldsymbol{v}, \boldsymbol{x}_k) = F(\boldsymbol{x}_k) + \nabla F^{(\mathcal{S}_k)}(\boldsymbol{x}_k)^\top (\boldsymbol{v} - \boldsymbol{x}_k^{(\mathcal{S}_k)}) + \frac{1}{2}(\boldsymbol{v} - \boldsymbol{x}_k^{(\mathcal{S}_k)})^\top \boldsymbol{A}^{(\mathcal{S}_k)}(\boldsymbol{x}_k)(\boldsymbol{v} - \boldsymbol{x}_k^{(\mathcal{S}_k)}).$$

3. **Minimize within the memory gradient subspace**

$$\boldsymbol{x}_{k+1}^{(\mathcal{S}_k)} = \underset{\boldsymbol{v} \in \operatorname{ran} \boldsymbol{D}_k^{(\mathcal{S}_k)}}{\operatorname{Argmin}} \; Q^{(\mathcal{S}_k)}(\boldsymbol{v}, \boldsymbol{x}_k)$$

where

$$(\forall j \in \mathcal{S}_k) \quad \boldsymbol{D}_k^{(j)} = \begin{cases} -\nabla F^{(j)}(\boldsymbol{x}_k) & \text{if} \quad j \notin \bigcup_{\ell=0}^{k-1} \mathcal{S}_\ell, \\ \left[-\nabla F^{(j)}(\boldsymbol{x}_k) \middle| \boldsymbol{x}_k^{(j)} - \boldsymbol{x}_{k-1}^{(j)} \right] & \text{otherwise.} \end{cases}$$

Block alternating 3MG

1. Select a block subset: Choose a non empty $\mathcal{S}_k \subset \{1, \ldots, J\}$.

2. Find a tractable surrogate in this subset:

⇝ Set $\boldsymbol{A}^{(\mathcal{S}_k)}(\boldsymbol{x}_k) = ([\boldsymbol{A}(\boldsymbol{x}_k)]_{p,p})_{p \in \mathbb{S}_k}$. The restriction of F to $\mathcal{S}_k$ is majorized at $\boldsymbol{x}_k$ by

$$
\begin{aligned}
(\forall \boldsymbol{v} \in \mathbb{R}^{|\mathbb{S}_k|}) \quad Q^{(\mathcal{S}_k)}(\boldsymbol{v}, \boldsymbol{x}_k) = F(\boldsymbol{x}_k) + \nabla F^{(\mathcal{S}_k)}(\boldsymbol{x}_k)^\top (\boldsymbol{v} - \boldsymbol{x}_k^{(\mathcal{S}_k)}) \\
+ \frac{1}{2} (\boldsymbol{v} - \boldsymbol{x}_k^{(\mathcal{S}_k)})^\top \boldsymbol{A}^{(\mathcal{S}_k)}(\boldsymbol{x}_k)(\boldsymbol{v} - \boldsymbol{x}_k^{(\mathcal{S}_k)}).
\end{aligned}
$$

3. Minimize within the memory gradient subspace

$$
\boldsymbol{x}_{k+1}^{(\mathcal{S}_k)} = \underset{\boldsymbol{v} \in \operatorname{ran} \boldsymbol{D}_k^{(\mathcal{S}_k)}}{\operatorname{Argmin}} \quad Q^{(\mathcal{S}_k)}(\boldsymbol{v}, \boldsymbol{x}_k)
$$

where

$$
(\forall j \in \mathcal{S}_k) \quad \boldsymbol{D}_k^{(j)} = \begin{cases} -\nabla F^{(j)}(\boldsymbol{x}_k) & \text{if} \quad j \notin \bigcup_{\ell=0}^{k-1} \mathcal{S}_\ell, \\ \left[-\nabla F^{(j)}(\boldsymbol{x}_k) | \boldsymbol{x}_k^{(j)} - \boldsymbol{x}_{k-1}^{(j)} \right] & \text{otherwise.} \end{cases}
$$

Problem: Matrices $\boldsymbol{A}^{(\mathcal{S})}$ do not have any block diagonal structure

$\Longrightarrow$ Difficult to perform **Step 3** in parallel !

Block separable majorant matrix

Let us assume that:

$$(\forall \boldsymbol{x} \in \mathbb{R}^N) \quad \boldsymbol{A}(\boldsymbol{x}) = \sum_{s=1}^{S} \boldsymbol{L}_s^\top \operatorname{Diag}\{\boldsymbol{\omega}_s(\boldsymbol{L}_s\boldsymbol{x})\}\, \boldsymbol{L}_s,$$

Let $\mathcal{S} \subset \{1,\dots,J\}$ non empty. Then,

$$(\forall \boldsymbol{x} \in \mathbb{R}^N) \quad \boldsymbol{A}^{(\mathcal{S})}(\boldsymbol{x}) \preceq \boldsymbol{B}^{(\mathcal{S})}(\boldsymbol{x}) = \operatorname{BDiag}\left\{\left(\boldsymbol{B}^{(j)}(\boldsymbol{x})\right)_{j\in\mathcal{S}}\right\},$$

where, for every $j \in \mathcal{S}$, matrix $\boldsymbol{B}^{(j)}(\boldsymbol{x}) \in \mathbb{R}^{N_j \times N_j}$ is given by:

$$\boldsymbol{B}^{(j)}(\boldsymbol{x}) = \sum_{s=1}^{S} \left((\boldsymbol{L}_s^{(j)})^\top \operatorname{Diag}\{\boldsymbol{b}_s(\boldsymbol{L}_s\boldsymbol{x})\}\, \boldsymbol{L}_s^{(j)} \right),$$

with, for every $s \in \{1,\dots,S\}$ and $p \in \{1,\dots,P_s\}$,

$$[\boldsymbol{b}_s(\boldsymbol{L}_s\boldsymbol{x})]_p = [\boldsymbol{\omega}_s(\boldsymbol{L}_s\boldsymbol{x})]_p [|\boldsymbol{L}_s^{(\mathcal{S})}|\mathbf{1}_{|\mathbb{S}|}]_p / [|\boldsymbol{L}_s^{(j)}|\mathbf{1}_{N_j}]_p.$$

Proof: Rely on Jensen's inequality.

BP3MG Algorithm

Initialize $\boldsymbol{x}_0 \in \mathbb{R}^N$

For $k = 0, 1, 2, \ldots$

Select $\mathcal{S}_k \subset \{1, \ldots, J\}$ s.t. $|\mathcal{S}_k| = C$

Parfor $j \in \mathcal{S}_k$

Compute $\nabla F^{(j)}(\boldsymbol{x}_k)$

Compute $\boldsymbol{B}_k^{(j)}(\boldsymbol{x}_k)$

Construct $\boldsymbol{D}_k^{(j)}$

$$\boldsymbol{u}_k^{(j)} = -\left((\boldsymbol{D}_k^{(j)})^\top \boldsymbol{B}_k^{(j)}(\boldsymbol{x}_k)\boldsymbol{D}_k^{(j)}\right)^\dagger (\boldsymbol{D}_k^{(j)})^\top \nabla F^{(j)}(\boldsymbol{x}_k)$$

$$\boldsymbol{x}_{k+1}^{(j)} = \boldsymbol{x}_k^{(j)} + \boldsymbol{D}_k^{(j)}\boldsymbol{u}_k^{(j)}$$

Set , for every $j \in \{1, \ldots, J\} \setminus \mathcal{S}_k$, $\boldsymbol{x}_{k+1}^{(j)} = \boldsymbol{x}_k^{(j)}$.

Share $\left(\boldsymbol{x}_{k+1}^{(j)}\right)_{j \in \mathcal{S}_k}$ between all cores.

Practical implementation

In practice, it is usually not necessary to send the full vector $\boldsymbol{x}_{k+1}$ to all the cores, at each iteration k.

- The update of the j-th block only require the knowledge of the current iterate at indices

$$\mathcal{N}_j = \bigcup_{s=1}^{S} \{n \in \{1,\dots,N\} | (\exists p \in \mathcal{P}_{s,j}) \ [\boldsymbol{L}_s]_{p,n} \neq 0\},$$

where $\mathcal{P}_{s,j} = \{p \in \{1,\dots,P_s\} \, | (\exists i \in \mathbb{J}_j) \ [\boldsymbol{L}_s]_{p,i} \neq 0\}$.

$*$ The cardinality of $\mathcal{N}_j$ is usually very small with respect to N.

Example: $S = 1$ and $\boldsymbol{L}_1$ is a discrete gradient operator with one pixel neighborhood $\Rightarrow |\mathcal{N}_j| = 3$.

Convergence theorem (ongoing work)

Let assume that:

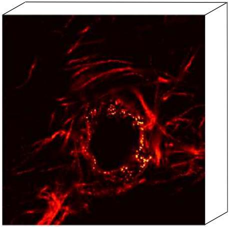
1. $F : \mathbb{R}^N \to \mathbb{R}$ is a coercive, differentiable function.
2. There exists a constant $K \geqslant J$ such that, for every $k \in \mathbb{N}$, $\{1, \dots, J\} \subset \bigcup_{\ell=k}^{k+K-1} \mathcal{S}_\ell$.
3. There exists $(\underline{\nu}, \overline{\nu}) \in]0, +\infty[^2$ such that $(\forall k \in \mathbb{N})$ $\underline{\nu}\,\mathbf{Id} \preceq \boldsymbol{B}^{(\mathcal{S}_k)}(\boldsymbol{x}_k) \preceq \overline{\nu}\,\mathbf{Id}$,

Then, the following hold:

- $\|\nabla F(\boldsymbol{x}_k)\| \to 0$ and $F(\boldsymbol{x}_k) \searrow F(\widehat{\boldsymbol{x}})$ where $\widehat{\boldsymbol{x}}$ is a critical point of F.
- If F is convex, any sequential cluster point of $(\boldsymbol{x}_k)_{k\in\mathbb{N}}$ is a minimizer of F.
- If F is strongly convex, then $(\boldsymbol{x}_k)_{k\in\mathbb{N}}$ converges to the unique (global) minimizer $\widehat{\boldsymbol{x}}$ of F

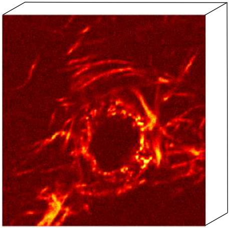
Application to 3D image deconvolution

Problem statement



Original 3D image
$\overline{\boldsymbol{x}} \in \mathbb{R}^N$

Degradations
$\boldsymbol{H} \in \mathbb{R}^{N \times N}$, $\boldsymbol{b} \in \mathbb{R}^N$



Measured 3D image
$\boldsymbol{y} = \boldsymbol{H}\overline{\boldsymbol{x}} + \boldsymbol{b}$

- $\boldsymbol{H}$: 3D convolution operator representing depth-variant 3D Gaussian blur (kernel size $5 \times 5 \times 11$). For each depth $z \in \{1, \ldots, N_{\mathrm{Z}}\}$, different variance and rotation parameters.
- $\boldsymbol{b}$: additive Gaussian i.i.d. zero-mean noise.

Variational approach

Objective function

$$(\forall x \in \mathbb{R}^N) \quad F(\boldsymbol{x}) = \frac{1}{2}\|\boldsymbol{H}\boldsymbol{x} - \boldsymbol{y}\|^2 + R(\boldsymbol{x})$$

⇝ Hybrid penalization term $R = R_1 + R_2 + R_3$:

- $R_1(\boldsymbol{x}) = \eta \sum_{n=1}^{N} d^2_{[x_{\min}, x_{\max}]}(\boldsymbol{x}_n)$
- $R_2(\boldsymbol{x}) = \lambda \sum_{n=1}^{N} \sqrt{([\boldsymbol{V}^{\mathsf{X}}\boldsymbol{x}]_n)^2 + ([\boldsymbol{V}^{\mathsf{Y}}\boldsymbol{x}]_n)^2 + \delta^2}$
- $R_3(\boldsymbol{x}) = \kappa \sum_{n=1}^{N} ([\boldsymbol{V}^{\mathsf{Z}}\boldsymbol{x}])^2$

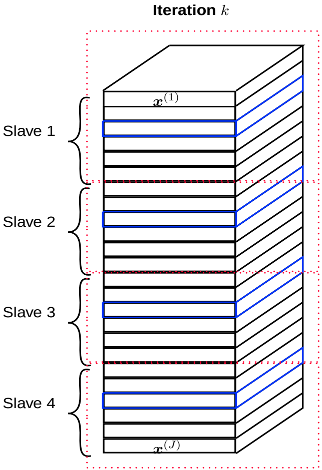
- $(\eta, \lambda, \delta, \kappa) \in (0, +\infty)^4$: regularization parameters;
- $[x_{\min}, x_{\max}]$: range of pixel intensity values; d_C: distance to set C;
- $\boldsymbol{V}^{\mathsf{X}}$,$\boldsymbol{V}^{\mathsf{Y}}$,$\boldsymbol{V}^{\mathsf{Z}} \in \mathbb{R}^{N \times N}$: discrete gradients along X,Y and Z.

Parallel implementation

- **Blocks:** N_Z slices of the 3D volume.
- Message Passing Interface command SPMD of MATLAB®
- Master-Slave implementation:
  - ➜ **1 master core:**
    Main loop of the algorithm.
  - ➜ $\overline{C}$ **slave cores:**
    Perform their tasks simultaneously.

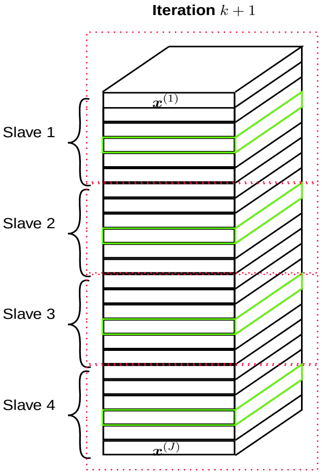
Parallel implementation

- **Blocks:** N_Z slices of the 3D volume.
- Message Passing Interface command SPMD of MATLAB®
- Master-Slave implementation:
  - ➜ **1 master core:**
    Main loop of the algorithm.
  - ➜ $\overline{C}$ **slave cores:**
    Perform their tasks simultaneously.

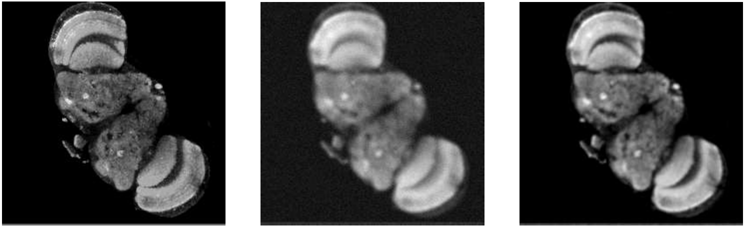


Parallel implementation

- **Blocks:** N_Z slices of the 3D volume.
- Message Passing Interface command SPMD of MATLAB®
- Master-Slave implementation:
  - ➜ **1 master core:** Main loop of the algorithm.
  - ➜ $\overline{C}$ **slave cores:** Perform their tasks simultaneously.



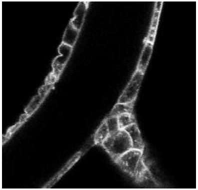
Restoration results: `FlyBrain`



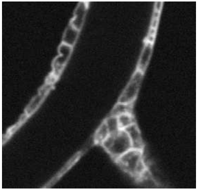
(a) Original (b) Degraded (c) Restored

Images corresponding to slice $z = 18$ of the 3D volume `FlyBrain` ($256 \times 256 \times 48$). Initial SNR 13.42 dB. Final SNR 16.98 dB.

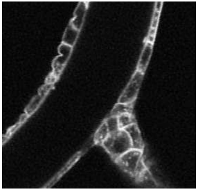
Restoration results: Tube



(a) Original



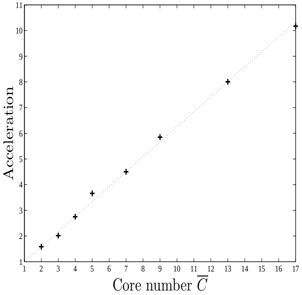
(b) Degraded



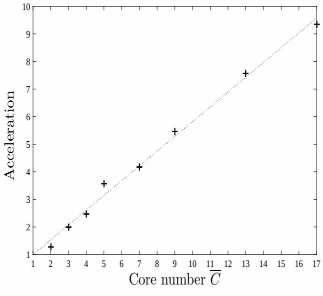
(c) Restored

Images corresponding to slice $z = 31$ of the 3D volume Tube ($284 \times 280 \times 48$). Initial SNR 11.53 dB. Final SNR 14.47 dB.

Acceleration



(a) `FlyBrain`



(b) `Tube`

Ratio between the computation time for one core and the computation time for $\overline{C}$ cores ($+$) with linear fitting ($\cdots$).

Conclusion

The *Block Parallel Majorize-Minimize Memory Gradient (BP3MG) Algorithm* handles smooth optimization problems of very large dimension.

- ✓ Reduced complexity / memory requirement.
- ✓ High efficiency in the context of 3D image restoration.
- ✓ Great potential for parallelization.

$\rightsquigarrow$ Future work will involve implementation in other languages.

Some references

S. Cadoni, E. Chouzenoux, J.-C. Pesquet, C. Chaux.
A Block Parallel Majorize-Minimize Memory Gradient Algorithm.
In Proc. IEEE Conf. Image Process. (ICIP 2016), pp. 3194–3198, Phoenix, AZ, 25-28 Sep 2016. Best paper award finalist.

E. Chouzenoux, J. Idier, S. Moussaoui
A Majorize-Minimize strategy for subspace optimization applied to image restoration
in *IEEE Trans. Image Process.*, vol.20, no.18, pp. 1517-1528, 2011.

E. Chouzenoux,A. Jezierska, J.-C. Pesquet, H. Talbot
A Majorize-Minimize Subspace Approach for ℓ_2-ℓ_0 Image Regularization
in *SIAM J. Imag. Sci.*, vol.6, no.1, pp. 563-591,2013.

E. Chouzenoux, L. Lamassé, S. Anthoine, C. Chaux, A. Jaouen, I. Vanzetta, F. Debarbieux
Approche variationnelle pour la déconvolution rapide de données 3D en microscopie biphotonique
in *Actes du 25e colloque GRETSI*, Lyon, France, 8-11 septembre 2015.

A. Florescu, E. Chouzenoux, J.-C. Pesquet, P. Ciuciu and S. Ciochina.
A Majorize-Minimize Memory Gradient Method for Complex-Valued Inverse Problems
in *Signal Processing*, Vol. 103, pp. 285-295, 2014.