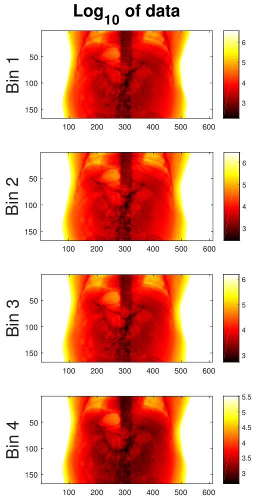
An ADMM algorithm for constrained material decomposition in spectral CT

May, 23th, 2018

CREATIS INSA INSTITUT NATIONAL DES SCIENCES APPLIQUÉES LYON PRIMES UNIVERSITE DE LYON ED EEA

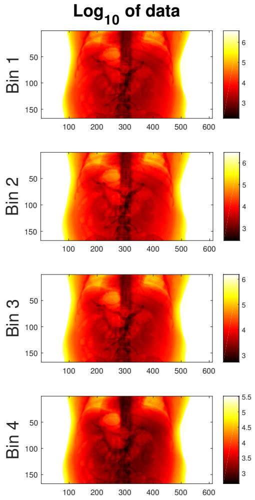
Spectral CT

Spectral CT records energy-dependant data

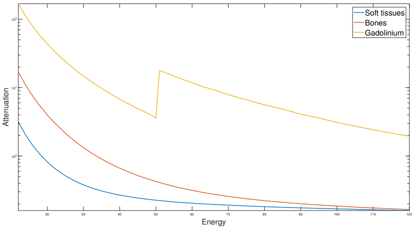


Spectral CT

Spectral CT records energy-dependant data

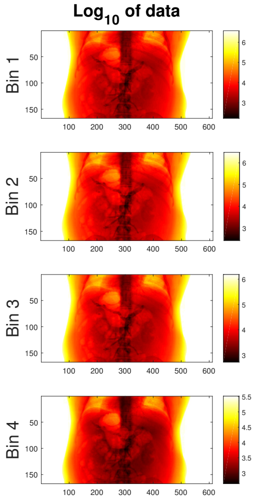


Attenuation coefficients depend on energy

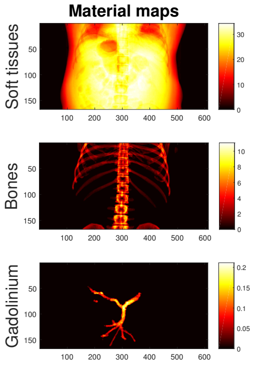


Spectral CT

Spectral CT records energy-dependant data

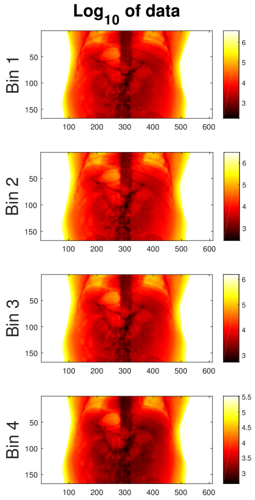


Allow decomposing materials



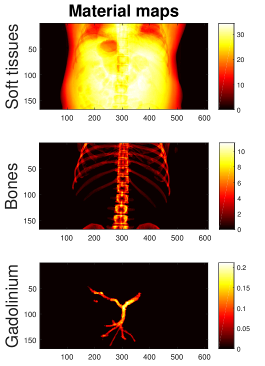
Spectral CT

Spectral CT records energy-dependant data



Solve the non-convex $\longrightarrow$ nonlinear problem $\mathbf{s}=\mathcal{F}(\mathbf{a})$

Allow decomposing materials

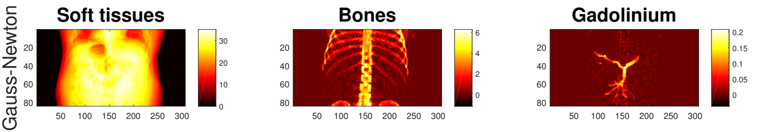


Introduction

Previous work

A Gauss-Newton algorithm was developed [1]

+ Fast convergence
+ Better decomposition error than first order algorithms
+ Spatially regularized
- Can have negative values

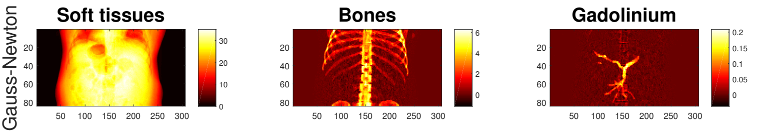


Introduction

Previous work

A Gauss-Newton algorithm was developed [1]

+ Fast convergence
+ Better decomposition error than first order algorithms
+ Spatially regularized
− Can have negative values



Constrained algorithms

Positivity constraints were added for materials decomposition

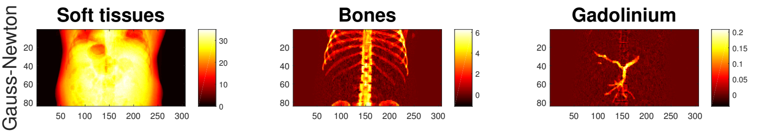
- Long and Fessler 2014 [2]
- Noh and Fessler 2009 [3]
- Sidky and Pan 2014 [4]
- Ding and Long 2017 [5]

Introduction

Previous work

A Gauss-Newton algorithm was developed [1]

+ Fast convergence
+ Better decomposition error than first order algorithms
+ Spatially regularized
− Can have negative values



Constrained algorithms

Positivity constraints were added for materials decomposition

- Long and Fessler 2014 [2]
- Noh and Fessler 2009 [3]
- Sidky and Pan 2014 [4]

} First oder algorithms in projection and image domain

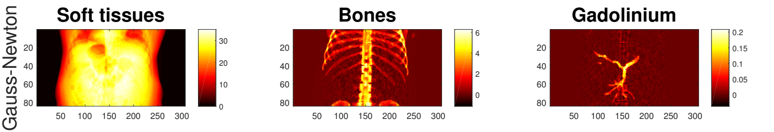
- Ding and Long 2017 [5]

Introduction

Previous work

A Gauss-Newton algorithm was developed [1]

+ Fast convergence
+ Better decomposition error than first order algorithms
+ Spatially regularized
− Can have negative values



Constrained algorithms

Positivity constraints were added for materials decomposition

- Long and Fessler 2014 [2]
- Noh and Fessler 2009 [3]
- Sidky and Pan 2014 [4]

} First oder algorithms in projection and image domain

- Ding and Long 2017 [5] } Second order algorithm in image domain

Proposed method

Positivity / Inequality

- We propose to enforce the positivity of the materials maps

Proposed method

Positivity / Inequality

- We propose to enforce the positivity of the materials maps

Equality

- Quantity of injected marker is known
- Should be retrieved in the decompositions

Proposed method

Positivity / Inequality

- We propose to enforce the positivity of the materials maps

Equality

- Quantity of injected marker is known
- Should be retrieved in the decompositions

Algorithm

- Use of an alternating direction method of multipliers (ADMM) : update of $\mathbf{a}$ with a second order algorithm

Forward problem

Denoting $\mathbf{s} \in \mathbb{R}^{IP}$ the data vector and $\mathbf{a} \in \mathbb{R}^{MP}$ the materials maps, we have :

$$\mathbf{s} = [s_{1,1}, \ldots, s_{I,1}, \ldots, s_{I,P}]^{\mathrm{T}}$$

$$\mathbf{a} = [a_{1,1},\ \ldots,\ a_{m,1},\ \ldots,\ a_{M,P}]^{\mathrm{T}}$$

Forward problem

Denoting $\mathbf{s} \in \mathbb{R}^{IP}$ the data vector and $\mathbf{a} \in \mathbb{R}^{MP}$ the materials maps, we have :

$$\mathbf{s} = [s_{1,1}, \ldots, s_{I,1}, \ldots, s_{I,P}]^{\mathrm{T}}$$

$$\mathbf{a} = [a_{1,1},\ \ldots,\ a_{m,1},\ \ldots,\ a_{M,P}]^{\mathrm{T}}$$

Using the standard spectral CT forward model is :

$$\mathbf{s}_{i,p} = \int_{\mathbf{R}} n_i^0(E) \exp\left(-\sum_{m=1}^{M} a_{m,p}\tau_m(E)\right) \mathrm{d}E$$

with $n_i^0(E)$ the effective spectrum, $\tau_m(E)$ a function representing the material attenuation and $a_{m,p}$ the projected mass of material m on pixel p.

Inverse problem

- We propose to solve :

$$\min_{\mathbf{a}} \mathcal{C}(\mathbf{a}, \mathbf{s}) \quad \text{s.t} \quad \begin{cases} \mathbf{a} \geq \mathbf{0} \\ \sum_p a_{m,p} = c_m \end{cases} \tag{1}$$

with c_m is the quantity of mth material.

- We chose

$$\mathcal{C}(\mathbf{a}, \mathbf{s}) = \mathcal{D}(\mathbf{a}, \mathbf{s}) + \alpha_{\mathrm{R}} \mathcal{R}(\mathbf{a})$$

where

$$\mathcal{D}(\mathbf{a}, \mathbf{s}) = ||\mathbf{s} - \mathcal{F}(\mathbf{a})||^2_{\mathbf{W}}$$
$$\mathcal{R}(\mathbf{a}) = ||\Delta \mathbf{a}_{\text{soft}}||^2_2 + ||\nabla \mathbf{a}_{\text{bone}}||_1 + ||\nabla \mathbf{a}_{\text{Gd}}||_1$$

ADMM

The Lagrangian function is minimized by an ADMM algorithm :

$$\begin{aligned}\mathcal{L}(\mathbf{a},\mathbf{b},\boldsymbol{\alpha}_{\mathrm{I}},\alpha_{\mathrm{E}},\mathbf{s}) = {} & \mathcal{D}(\mathbf{a},\mathbf{s}) + \alpha_{\mathrm{R}}\mathcal{R}(\mathbf{a}) \\ & + \mathcal{H}_{\mathrm{E}}(\mathbf{a},\alpha_{\mathrm{E}}) + \mathcal{G}_{\mathrm{E}}(\mathbf{a}) \\ & + \mathcal{H}_{\mathrm{I}}(\mathbf{a},\mathbf{b},\boldsymbol{\alpha}_{\mathrm{I}}) + \mathcal{G}_{\mathrm{I}}(\mathbf{a},\mathbf{b}) + \mathbf{1}(\mathbf{b})\end{aligned}$$

ADMM

The Lagrangian function is minimized by an ADMM algorithm :

$$\begin{aligned}\mathcal{L}(\mathbf{a},\mathbf{b},\boldsymbol{\alpha}_{\mathrm{I}},\alpha_{\mathrm{E}},\mathbf{s}) &= \mathcal{D}(\mathbf{a},\mathbf{s}) + \alpha_{\mathrm{R}}\mathcal{R}(\mathbf{a}) \\ &+ \mathcal{H}_{\mathrm{E}}(\mathbf{a},\alpha_{\mathrm{E}}) + \mathcal{G}_{\mathrm{E}}(\mathbf{a}) \\ &+ \mathcal{H}_{\mathrm{I}}(\mathbf{a},\mathbf{b},\boldsymbol{\alpha}_{\mathrm{I}}) + \mathcal{G}_{\mathrm{I}}(\mathbf{a},\mathbf{b}) + \mathbf{1}(\mathbf{b})\end{aligned}$$

where

Equality : $\sum_p a_{m,p} = c_m$

$$\mathcal{H}_{\mathrm{E}}(\mathbf{a},\alpha_{\mathrm{E}}) = \alpha_{\mathrm{E}}\left(\sum_{p=1}^{P}\frac{a_{\mathrm{gd},p}}{c_{\mathrm{gd}}} - 1\right)$$

$$\mathcal{G}_{\mathrm{E}}(\mathbf{a}) = \frac{\beta_{\mathrm{E}}}{2}\left(\sum_{p=1}^{P}\frac{a_{\mathrm{gd},p}}{c_{\mathrm{gd}}} - 1\right)^2$$

with $\alpha_{\mathrm{E}} \in \mathbf{R}$ and $\beta_{\mathrm{E}} \in \mathbf{R}$.

ADMM

The Lagrangian function is minimized by an ADMM algorithm :

$$\begin{aligned}\mathcal{L}(\mathbf{a}, \mathbf{b}, \boldsymbol{\alpha}_{\mathrm{I}}, \alpha_{\mathrm{E}}, \mathbf{s}) = &\mathcal{D}(\mathbf{a}, \mathbf{s}) + \alpha_{\mathrm{R}}\mathcal{R}(\mathbf{a}) \\ &+ \mathcal{H}_{\mathrm{E}}(\mathbf{a}, \alpha_{\mathrm{E}}) + \mathcal{G}_{\mathrm{E}}(\mathbf{a}) \\ &+ \mathcal{H}_{\mathrm{I}}(\mathbf{a}, \mathbf{b}, \boldsymbol{\alpha}_{\mathrm{I}}) + \mathcal{G}_{\mathrm{I}}(\mathbf{a}, \mathbf{b}) + \mathbf{1}(\mathbf{b})\end{aligned}$$

where

Inequality : $\mathbf{a} \geq 0$

Inequalities need an auxiliary variable, $\mathbf{b}$, such that the constraint becomes $\mathbf{a} = \mathbf{b}$ and $\mathbf{b} \geq \mathbf{0}$.

$$\begin{aligned}\mathcal{H}_{\mathrm{I}}(\mathbf{a}, \mathbf{b}, \boldsymbol{\alpha}_{\mathrm{I}}) &= \boldsymbol{\alpha}_{\mathrm{I}}^{\mathrm{T}}(\mathbf{b} - \mathbf{a}) \\ \mathcal{G}_{\mathrm{I}}(\mathbf{a}, \mathbf{b}) &= \frac{\beta_{\mathrm{I}}}{2}||\mathbf{b} - \mathbf{a}||_2^2\end{aligned}$$

and

$$\mathbf{1}(\mathbf{b}) = \begin{cases} 0, & \text{if } \mathbf{b} \geq 0 \\ \infty, & \text{otherwise}\end{cases}$$

with $\alpha_{\mathrm{I}} \in \mathbf{R}^{MP}$ and $\beta_{\mathrm{I}} \in \mathbf{R}$.

Minimization scheme

- Minimization of $\mathcal{L}(\mathbf{a}, \mathbf{b}, \boldsymbol{\alpha}_{\mathrm{I}}, \alpha_{\mathrm{E}}, \mathbf{s})$ is done by finding the saddle point [6], through alternating updates :

Update	Algorithm used
$\mathbf{a}^{\ell+1} \in \underset{\mathbf{a}}{\operatorname{argmin}}\, \mathcal{L}(\mathbf{a}, \mathbf{b}^{\ell}, \alpha_{\mathrm{E}}^{\ell}, \boldsymbol{\alpha}_{\mathrm{I}}^{\ell})$	Iterative Gauss-Newton
$\mathbf{b}^{\ell+1} \in \underset{\mathbf{b}}{\operatorname{argmin}}\, \mathcal{L}(\mathbf{a}^{\ell+1}, \mathbf{b}, \alpha_{\mathrm{E}}^{\ell}, \boldsymbol{\alpha}_{\mathrm{I}}^{\ell})$	Proximal algorithm
$\alpha_{\mathrm{E}}^{\ell+1} \in \underset{\alpha_{\mathrm{E}}}{\operatorname{argmax}}\, \mathcal{L}(\mathbf{a}^{\ell+1}, \mathbf{b}^{\ell+1}, \alpha_{\mathrm{E}}, \boldsymbol{\alpha}_{\mathrm{I}}^{\ell})$	Ascent gradient
$\boldsymbol{\alpha}_{\mathrm{I}}^{\ell+1} \in \underset{\boldsymbol{\alpha}_{\mathrm{I}}}{\operatorname{argmax}}\, \mathcal{L}(\mathbf{a}^{\ell+1}, \mathbf{b}^{\ell+1}, \alpha_{\mathrm{E}}^{\ell+1}, \boldsymbol{\alpha}_{\mathrm{I}})$	Ascent gradient

- Moreover, at each iteration ℓ, Lagrangian parameters are increasing

$$\beta_{\mathrm{E}}^{\ell+1} = \omega \beta_{\mathrm{E}}^{\ell}$$
$$\beta_{\mathrm{I}}^{\ell+1} = \omega \beta_{\mathrm{I}}^{\ell}$$

- Until they reach the maximum allow value $\beta_{\max}$.

Parameters

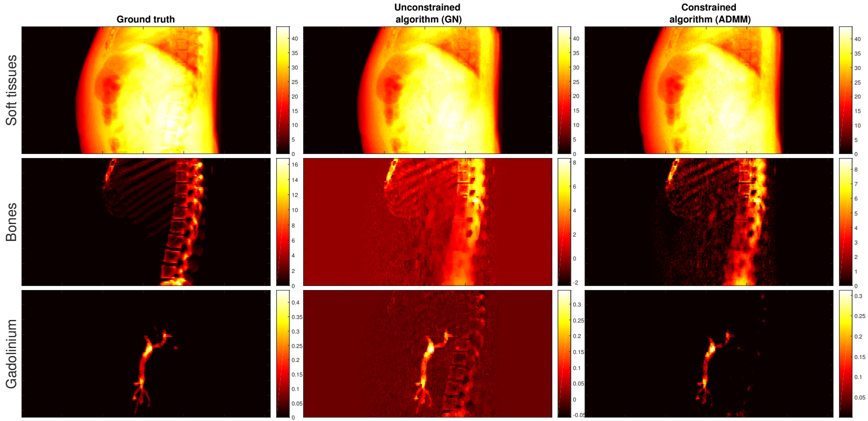
Phantom

- Thorax : 84×306 pixels
- 3 materials : soft tissues, bones and gadolinium (portal vein)
- 4 energy bins

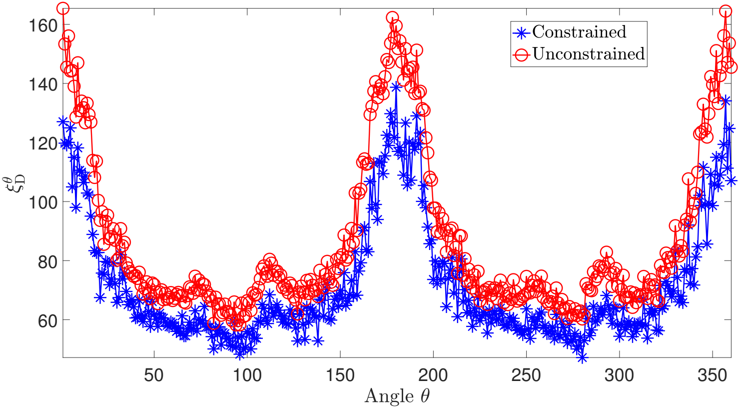
Algorithm

- $\mathbf{a}^0 = 0\text{g.cm}^{-2}$
- $\alpha_{\mathrm{R}} = 1$, $\alpha_{\mathrm{E}}^0 = 0$, $\boldsymbol{\alpha}_{\mathrm{I}}^0 = 0$
- $\beta_{\mathrm{E}}^0 = 10^0$, $\beta_{\mathrm{I}}^0 = 10^{-2}$, $\omega = 1.5$
- $\beta_{\max} = 10^{10}$
- Stopping criteria
  - The inner loop stops if
    - the relative decrease of the cost function is too low (10^{-3})
    - it made more than 30 iterations
  - The outer loop stops if
    - the equality is respected : $\left| \left(\sum_p^P \frac{a_{\mathrm{gd,p}}^\ell}{c_{\mathrm{gd}}} - 1 \right) \right| < 10^{-3}$
    - the inequality is respected : $||\mathbf{a}^\ell - \mathbf{b}^\ell|| < 10^{-3}$
- Decomposition for $\theta = [0^\circ, 359^\circ]$

Projection domain



Projection domain



Evaluation of the decomposition with the ℓ_2-norm :

$$\xi_{\mathrm{D}}^{\theta} = \sum_{m}^{M} ||\mathbf{a}_m^{\theta} - \mathbf{a}_m^{\mathrm{truth},\theta}||_2$$

Image domain

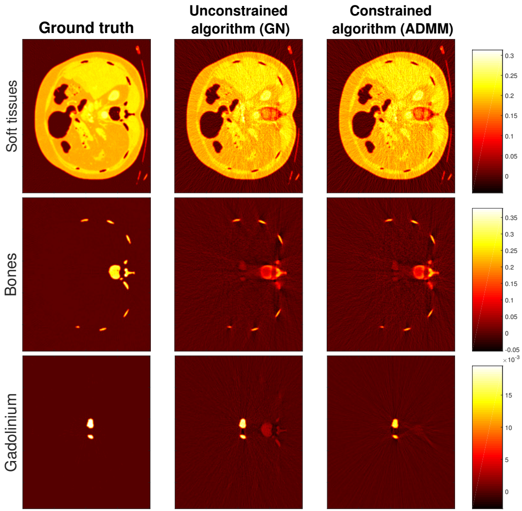
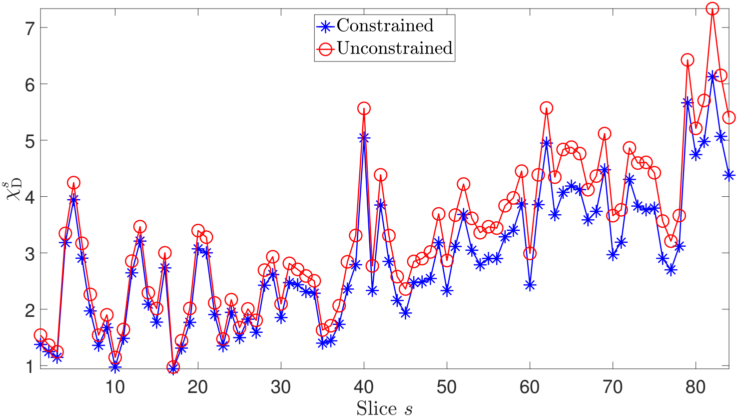


Image domain



Evaluation of the reconstructed materials with the ℓ_2-norm :

$$\chi_{\mathrm{D}}^{z} = \sum_{m}^{M} ||\boldsymbol{\rho}_m^{\mathrm{z}} - \boldsymbol{\rho}_m^{\mathrm{truth,z}}||_2$$

Conclusion

- Equality and inequality constraints give a better decomposition and visualization
- Decrease cross-talk
- However, it needs more iteration

Conclusion

- Equality and inequality constraints give a better decomposition and visualization
- Decrease cross-talk
- However, it needs more iteration

And next :

- Application of this algorithm to real data

Thank you for your attention !

Questions ? :)

References

Nicolas Ducros, Juan Felipe Perez-Juste Abascal, Bruno Sixou, Simon Rit, and Françoise Peyrin,
"Regularization of nonlinear decomposition of spectral x-ray projection images,"
Medical physics, vol. 44, no. 9, 2017.

Yong Long and Jeffrey A Fessler,
"Multi-material decomposition using statistical image reconstruction for spectral ct,"
IEEE transactions on medical imaging, vol. 33, no. 8, pp. 1614–1626, 2014.

Joonki Noh, Jeffrey A Fessler, and Paul E Kinahan,
"Statistical sinogram restoration in dual-energy ct for pet attenuation correction,"
IEEE transactions on medical imaging, vol. 28, no. 11, pp. 1688–1702, 2009.

Xiaochuan Pan, Buxin Chen, Zheng Zhang, Erik Pearson, Emil Sidky, and Xiao Han,
"Optimization-based reconstruction exploiting spectral information in ct,"
in *The Third International Conference on Image Formation in X-Ray Computed Tomography*, 2014, pp. 228–232.

Qiaoqiao Ding, Tianye Niu, Xiaoqun Zhang, and Yong Long,
"Image-domain multi-material decomposition for dual-energy ct based on correlation and sparsity of material images,"
arXiv preprint arXiv :1710.07028, 2017.

Jonathan Eckstein and W Yao,
"Augmented lagrangian and alternating direction methods for convex optimization : A tutorial and some illustrative computational results,"
RUTCOR Research Reports, vol. 32, pp. 3, 2012.

Minimization scheme

while *Stopping criteria aren't met for the outer loop* **do**
  while *Stopping criteria aren't met for the inner loop* **do**
    solve $\left(\mathbf{H}^k \delta \mathbf{a}^k = -\mathbf{g}^k\right)$
    $\mathbf{a}^{k+1} = \mathbf{a}^k + \lambda^k \delta \mathbf{a}^k$
    $k = k + 1$
  end
  $\mathbf{a}^{\ell+1} = \mathbf{a}^{k-1}$

- Update of $\mathbf{a}$: Gauss-Newton algorithm

Minimization scheme

while *Stopping criteria aren't met for the outer loop* **do**
  while *Stopping criteria aren't met for the inner loop* **do**
    solve $\left(\mathbf{H}^k \delta \mathbf{a}^k = -\mathbf{g}^k\right)$
    $\mathbf{a}^{k+1} = \mathbf{a}^k + \lambda^k \delta \mathbf{a}^k$
    $k = k + 1$
  end
  $\mathbf{a}^{\ell+1} = \mathbf{a}^{k-1}$
  $\mathbf{b}^{\ell+1} = \Pi(\mathbf{a}^{\ell+1} - \frac{\boldsymbol{\alpha}_{\mathrm{I}}^{\ell}}{\beta_{\mathrm{I}}})$

- Update of **a** : Gauss-Newton algorithm
- Update of **b** : Proximal algorithm

Minimization scheme

while *Stopping criteria aren't met for the outer loop* **do**
  while *Stopping criteria aren't met for the inner loop* **do**
    solve $\left(\mathbf{H}^k \delta \mathbf{a}^k = -\mathbf{g}^k\right)$
    $\mathbf{a}^{k+1} = \mathbf{a}^k + \lambda^k \delta \mathbf{a}^k$
    $k = k + 1$
  end
  $\mathbf{a}^{\ell+1} = \mathbf{a}^{k-1}$
  $\mathbf{b}^{\ell+1} = \Pi(\mathbf{a}^{\ell+1} - \frac{\boldsymbol{\alpha}_{\mathrm{I}}^{\ell}}{\beta_{\mathrm{I}}})$
  $\alpha_{\mathrm{E}}^{\ell+1} = \alpha_{\mathrm{E}}^{\ell} + \beta_{\mathrm{E}} \frac{\mathcal{H}_E(\mathbf{a}_m^{\ell+1})}{\alpha_{\mathrm{E}}^{\ell}}$

- Update of **a** : Gauss-Newton algorithm
- Update of **b** : Proximal algorithm
- Update of α_{E} : Ascent gradient

Minimization scheme

while *Stopping criteria aren't met for the outer loop* **do**
  while *Stopping criteria aren't met for the inner loop* **do**
    solve $\left(\mathbf{H}^k \delta \mathbf{a}^k = -\mathbf{g}^k\right)$
    $\mathbf{a}^{k+1} = \mathbf{a}^k + \lambda^k \delta \mathbf{a}^k$
    $k = k + 1$
  end
  $\mathbf{a}^{\ell+1} = \mathbf{a}^{k-1}$
  $\mathbf{b}^{\ell+1} = \Pi(\mathbf{a}^{\ell+1} - \frac{\boldsymbol{\alpha}_{\mathrm{I}}^{\ell}}{\beta_{\mathrm{I}}})$
  $\alpha_{\mathrm{E}}^{\ell+1} = \alpha_{\mathrm{E}}^{\ell} + \beta_{\mathrm{E}} \frac{\mathcal{H}_E(\mathbf{a}_m^{\ell+1})}{\alpha_{\mathrm{E}}^{\ell}}$
  $\boldsymbol{\alpha}_{\mathrm{I}}^{\ell+1} = \boldsymbol{\alpha}_{\mathrm{I}}^{\ell} + \beta_{\mathrm{I}}(\mathbf{b}^{\ell+1} - \mathbf{a}^{\ell+1})$

- Update of **a** : Gauss-Newton algorithm
- Update of **b** : Proximal algorithm
- Update of α_{E} : Ascent gradient
- Update of $\boldsymbol{\alpha}_{\mathrm{I}}$: Ascent gradient

Minimization scheme

while *Stopping criteria aren't met for the outer loop* **do**

  while *Stopping criteria aren't met for the inner loop* **do**

    solve $\left(\mathbf{H}^k \delta \mathbf{a}^k = -\mathbf{g}^k\right)$

    $\mathbf{a}^{k+1} = \mathbf{a}^k + \lambda^k \delta \mathbf{a}^k$

    $k = k + 1$

  end

  $\mathbf{a}^{\ell+1} = \mathbf{a}^{k-1}$

  $\mathbf{b}^{\ell+1} = \Pi(\mathbf{a}^{\ell+1} - \frac{\boldsymbol{\alpha}_{\mathrm{I}}^{\ell}}{\beta_{\mathrm{I}}})$

  $\alpha_{\mathrm{E}}^{\ell+1} = \alpha_{\mathrm{E}}^{\ell} + \beta_{\mathrm{E}} \frac{\mathcal{H}_E(\mathbf{a}_m^{\ell+1})}{\alpha_{\mathrm{E}}^{\ell}}$

  $\boldsymbol{\alpha}_{\mathrm{I}}^{\ell+1} = \boldsymbol{\alpha}_{\mathrm{I}}^{\ell} + \beta_{\mathrm{I}}(\mathbf{b}^{\ell+1} - \mathbf{a}^{\ell+1})$

  $\beta_{\mathrm{E}}^{\ell+1} = \omega \beta_{\mathrm{E}}^{\ell}$

  $\beta_{\mathrm{I}}^{\ell+1} = \omega \beta_{\mathrm{I}}^{\ell}$

- Update of $\mathbf{a}$: Gauss-Newton algorithm
- Update of $\mathbf{b}$: Proximal algorithm
- Update of α_{E} : Ascent gradient
- Update of $\boldsymbol{\alpha}_{\mathrm{I}}$: Ascent gradient
- Update of the βs

Minimization scheme

while *Stopping criteria aren't met for the outer loop* **do**
  while *Stopping criteria aren't met for the inner loop* **do**
    solve $\left(\mathbf{H}^k \delta \mathbf{a}^k = -\mathbf{g}^k\right)$
    $\mathbf{a}^{k+1} = \mathbf{a}^k + \lambda^k \delta \mathbf{a}^k$
    $k = k + 1$
  end
  $\mathbf{a}^{\ell+1} = \mathbf{a}^{k-1}$
  $\mathbf{b}^{\ell+1} = \Pi(\mathbf{a}^{\ell+1} - \frac{\boldsymbol{\alpha}_{\mathrm{I}}^{\ell}}{\beta_{\mathrm{I}}})$
  $\alpha_{\mathrm{E}}^{\ell+1} = \alpha_{\mathrm{E}}^{\ell} + \beta_{\mathrm{E}} \frac{\mathcal{H}_E(\mathbf{a}_m^{\ell+1})}{\alpha_{\mathrm{E}}^{\ell}}$
  $\boldsymbol{\alpha}_{\mathrm{I}}^{\ell+1} = \boldsymbol{\alpha}_{\mathrm{I}}^{\ell} + \beta_{\mathrm{I}}(\mathbf{b}^{\ell+1} - \mathbf{a}^{\ell+1})$
  $\beta_{\mathrm{E}}^{\ell+1} = \omega \beta_{\mathrm{E}}^{\ell}$
  $\beta_{\mathrm{I}}^{\ell+1} = \omega \beta_{\mathrm{I}}^{\ell}$
  if $\beta_{\mathrm{E}}^{\ell+1} > \beta_{\max}$ **then**
    $\beta_{\mathrm{E}}^{\ell+1} = \beta_{\max}$
  end
  if $\beta_{\mathrm{I}}^{\ell+1} > \beta_{\max}$ **then**
    $\beta_{\mathrm{I}}^{\ell+1} = \beta_{\max}$
  end
  $\ell = \ell + 1$
end

- Update of **a** : Gauss-Newton algorithm
- Update of **b** : Proximal algorithm
- Update of α_{E} : Ascent gradient
- Update of $\boldsymbol{\alpha}_{\mathrm{I}}$: Ascent gradient
- Update of the βs
- Make sure that the values aren't too high