HAL open science

Local-mean preserving post-processing step for non-negativity enforcement in PET imaging: application to 90 Y-PET

Maël Millardet, Said Moussaoui, Diana Mateus, Jérôme Idier, Thomas Carlier

To cite this version:

Maël Millardet, Said Moussaoui, Diana Mateus, Jérôme Idier, Thomas Carlier. Local-mean preserving post-processing step for non-negativity enforcement in PET imaging: application to 90 Y-PET. IEEE Transactions on Medical Imaging, 2020, 39, pp.3725 - 3736. 10.1109/tmi.2020.3003428 . hal-02565204

HAL Id: hal-02565204

https://hal.science/hal-02565204v1

Submitted on 18 Nov 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TMI.2020.3003428

Local-mean preserving post-processing step for non-negativity enforcement in PET imaging: application to ^{90}Y-PET.

Maël Millardet, Saïd Moussaoui, Diana Mateus, Jérôme Idier, and Thomas Carlier

***Abstract*—In a low-statistics PET imaging context, the positive bias in regions of low activity is a burning issue. To overcome this problem, algorithms without the built-in non-negativity constraint may be used. They allow negative voxels in the image to reduce, or even to cancel the bias. However, such algorithms increase the variance and are difficult to interpret since the resulting images contain negative activities, which do not hold a physical meaning when dealing with radioactive concentration. In this paper, a post-processing approach is proposed to remove these negative values while preserving the local mean activities. Its original idea is to transfer the value of each voxel with negative activity to its direct neighbors under the constraint of preserving the local means of the image. In that respect, the proposed approach is formalized as a linear programming problem with a specific symmetric structure, which makes it solvable in a very efficient way by a dual-simplex-like iterative algorithm. The relevance of the proposed approach is discussed on simulated and on experimental data. Acquired data from an yttrium-90 phantom show that on images produced by a non-constrained algorithm, a much lower variance in the cold area is obtained after the post-processing step, at the price of a slightly increased bias. More specifically, when compared with the classical OSEM algorithm, images are improved, both in terms of bias and of variance.**

***Index Terms*—Image restoration (noise and artifact reduction), linear programming, local mean, non-negativity constraint enforcement, nuclear imaging (PET), optimization.**

I. Introduction

POSITRON Emission Tomography (PET) has now a major impact on the management of patients in oncology. Its ability to image interactions at picomolar concentrations makes it very attractive for revealing molecular processes. Furthermore, PET is a native quantitative imaging modality [1]. However, several phenomena may counteract the quantitative precision in the reconstructed images, like scatter, random or multiple coincidences [2].

This is especially true when the number of coincidences collected is small, such as in brain dynamic PET where the time frames are often very short or may have only limited amount of data due to activity decay [3]. The number of prompt events recorded may be also reduced even further than in dynamic PET if radio-pharmaceuticals with small branching ratio are used. In particular, micro-spheres of yttrium-90 (^{90}Y) are indicated for the treatment of primary and secondary liver cancers [4]. However, ^{90}Y is a β^--emitter, and its decay produces positrons with a branching ratio of only 3.2×10^{-5} [5]. The data are therefore very noisy and the reconstruction of ^{90}Y-PET images is possible, but challenging [6], [7]. In low-activity regions (called "cold" areas), the noise, coupled with the built-in non-negativity constraint of most of the algorithms used today, such as the classical MLEM (maximum-likelihood expectation maximization) [8], [9], and OSEM (ordered-subset expectation maximization) [10], [11], induce a positive bias, *i.e.*, an over-estimation of the activity in these regions [12]. To tackle this issue, modified algorithms allowing negative values in the image have been introduced [3], [13]–[15]. Such algorithms do reduce the positive bias (or even cancel it out), but at the cost of a variance increase and a loss of physical meaning for the voxel values. This is a major issue for dosimetric studies because dose/volume histograms become unrealistic. Here, we address the difficult problem of reducing both the bias and the variance while making the image non-negative.

Previous works have been focused on finding new algorithms that require non-negativity in the data space while being unconstrained in the image space [15], [16]. Such methods partially solve the physical inconsistencies, but negative voxels may still be present in the image. This work proposes a post-processing strategy that can be used in addition to any unconstrained, or partially constrained, algorithm, by producing a non-negative image.

A possible post-processing strategy would be to make a simple orthogonal projection of the reconstructed image on the set of non-negative images, *i.e.*, setting negative intensity values to zero, while keeping the positive ones unchanged.

This work has been supported in part by the European Regional Development Fund, the Pays de la Loire region on the Connect Talent scheme MILCOM (Multi-modal Imaging and Learning for Computational-based Medicine), Nantes Métropole (Convention 2017-10470), the French National Agency for Research called "Investissements d'Avenir" IRON Labex n° ANR-11-LABX-0018-01 and the French program Infrastructure d'avenir en Biologie Santé ANR-11-INBS-0006 (France Life Imaging).

M. Millardet, S. Moussaoui, D. Mateus and J. Idier are with the LS2N, CNRS UMR 6004, Nantes, France and with École centrale de Nantes, Nantes, France (e-mail: mael.millardet@ls2n.fr).

M. Millardet and T. Carlier are with the CRCINA, INSERM UMR 1232, Nantes, France.

T. Carlier is with the Nuclear Medicine Department, University Hospital, Nantes, France.

This paper has supplementary downloadable material available at http://ieeexplore.ieee.org., provided by the authors.

Copyright (c) 2019 IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending a request to pubs-permissions@ieee.org.

© 2020 IEEE

Copyright (c) 2020 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TMI.2020.3003428

However, such a solution presents several drawbacks. First, the average activity is increased, leading to a positive bias even higher than what is achievable using a constrained algorithm. Second, this does not take into consideration the meaning of negative values. Indeed, the negative values are not completely insignificant: they do not have any physical meaning at the voxel level but are important to reduce the mean activity of their surrounding area.

In this paper, we propose a nontrivial post-processing approach that transfers all negative values to neighboring voxels to modify the image only locally, and with a minimal impact in terms of statistical properties at the local scale. To achieve such a goal, one classical framework would be to use the formalism of optimal transport introduced by Gaspard Monge in 1781 [17], and revisited by Leonid Kantorovich in 1942 [18]. Its original idea is to transfer a defined amount of mass in a defined amount of holes with minimal cost. Optimal transport classically deals with the case where the amount of mass exactly fits the holes. This formalism is more focused on the way to arrive at a final solution than the intrinsic final solution. However, it has been extended to the case where the initial and final distributions do not have the same mass [17], [19]. The main drawback of optimal transport applied to the spread of PET image negative values is that the transfer may be asymmetric, leading to a spurious spread of information. This happens because the path taken by the negative values would be in this case arbitrary. Another difficulty to use the formalism of optimal transport is that it would lead in general to several possible final images. The approach followed in this paper is to pre-define a symmetric *voxel spread function* describing which proportion of the value coming from a voxel will be distributed to each neighbor. The solution is the one corresponding to the minimal transfer following such a symmetric rule.

The contribution of this paper is firstly to formalize this post-processing strategy of negative voxel removal, and secondly to implement and test the method. We cast the problem in a linear programming framework [20], and we propose an adapted version of the dual simplex algorithm to solve the optimization problem. Besides, some of its mathematical properties are discussed. Finally, the performances of the non-negativity enforcement post-processing step (NNEPPS) have been evaluated on acquired ^{90}Y-PET data.

The rest of this paper is organized as follows. Section II gives the mathematical formulation of the negative voxel removal problem. Section III proposes a variant of the simplex algorithm allowing to compute the desired final image. The post-processing approach is then illustrated in Section IV. The impact of the proposed approach on real PET images is studied in Section V. Finally, discussions and conclusions are drawn in Section VI.

II. Problem statement

A. Presentation of the idea

The goal of the proposed post-processing step is to obtain a non-negative image by a minimal spread of the negative values over the positive voxels. Every voxel value is allowed to increase according to a *transfer coefficient* assigned to this voxel, while the values of its neighors decrease, such that the local mean is preserved.

This post-processing step comes with guarantees of the existence and uniqueness of a solution under the minimal assumption that the global mean of the initial image is non-negative[1]. The proposed algorithm finds the solution in a finite number of iterations, in practice around 15 for full PET images of 4×10^6 voxels, for a total time comprised between 1 and 10 minutes on a standard multicore CPU. Fig. 1 illustrates the effect of the NNEPPS on a simple 2D example.

In the rest of this section and in the next one, we go deeper into the mathematical formulation of the problem, and into the proposed solution.

B. Requirements

Let us consider a PET image $\boldsymbol{X}$ having negative values, obtained from an unconstrained reconstruction algorithm, and $\boldsymbol{x} \in \mathbb{R}^n$ a column vector version of $\boldsymbol{X}$ (*e.g.*, based on the lexicographic order). The addressed problem is the suppression of the negative values from $\boldsymbol{X}$ to get a modified image $\boldsymbol{Y}$ corresponding to a vector $\boldsymbol{y} \in (\mathbb{R}^+)^n$. As discussed above, a simple orthogonal projection of $\boldsymbol{x}$ onto $(\mathbb{R}^+)^n$ is not satisfactory. In this paper, an alternative approach is sought.

An additional requirement is the preservation of the global mean, *i.e.*, the mean of the post-processed image $\boldsymbol{Y}$ should be equal to the mean of $\boldsymbol{X}$. A natural post-processing solution could be to consider the orthogonal projection of $\boldsymbol{x}$ onto the convex set of non-negative vectors with the same mean as $\boldsymbol{x}$. Unfortunately, the latter would not respect the local character of intensity transfers. For example, if $\boldsymbol{x}$ is an image with only one negative voxel, the latter will be set to 0 while all the other voxels will be decreased by the same amount.

Therefore, we have to introduce the notion of neighborhood between voxels. A natural idea is then to minimize the "quantity of transfer" between neighboring voxels, instead of the difference between the final and the initial image. For this purpose, we can formulate the problem as $\min \|\boldsymbol{\alpha}\|$, such that $\boldsymbol{y} = \boldsymbol{x} + \boldsymbol{H}\boldsymbol{\alpha}$, where $\boldsymbol{\alpha}$ represents a map of the elementary transfers, $\|\cdot\|$ is a norm whose choice will be discussed later (in Section II-C3), and the operator $\boldsymbol{H}$ encodes the influence of the transfers on the image. Let us remark that the preservation of the global mean is automatically fulfilled in this formulation as long as it involves only transfers between voxels, *i.e.*, each line of $\boldsymbol{H}$ sums to zero.

C. Mathematical formulation of the problem

In this part, we examine several possibilities of local transfer structure though the specification of matrix $\boldsymbol{H}$, and we go deeper into the formulation of the problem.

1) Properties of $\boldsymbol{H}$: One first possibility is to allow direct transfers between pairs of neighboring voxels. If we choose to minimize the L1 norm of $\boldsymbol{\alpha}$, we then fall into the framework of optimal transport, which presents several drawbacks, as discussed in the introduction.

[1] It is necessary because the global mean is preserved during the NNEPPS.

Copyright (c) 2020 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TMI.2020.3003428

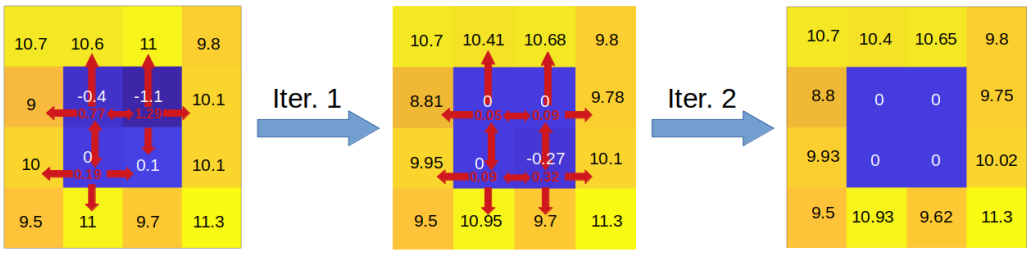


Fig. 1. Simple 2D example of the effect of The NNEPPS. In the initial image (left), three pixels are negative. Their intensities are transferred to their neighbors during the first iteration (transfer coefficients depicted in red). A new negative pixel appears (middle image), and is added to the set of pixels that needs to be subsequently set to 0. After the second iteration (right image), no new negative pixel appears, and the algorithm therefore stops.

The second possibility, which is chosen here, is to impose an isotropic spread of the negative values. Therefore, the weight vector $\boldsymbol{\alpha}$ has the same size as $\boldsymbol{x}$ and $\boldsymbol{y}$, each coefficient of $\boldsymbol{\alpha}$ representing the quantity that the corresponding voxel transfers to its direct neighbors in a symmetric, predefined way. This possibility answers the hunch that the intensity of a negative voxel should be uniformly transferred to its neighbors.

For the sake of clarity, we have grouped the properties that should be fulfilled by matrix $\boldsymbol{H}$ into two classes. The first one contains five properties **M1**-**M5**. They are **mandatory** in the sense that the mathematical validity of our contribution depends on them. We also consider three **optional** requirements. **O1** will be assumed throughout the article, but its role is purely a matter of technical simplicity, and properties extend easily if it is not fulfilled. **O2** and **O3** correspond to natural choices, as discussed above, although they are not necessary conditions in the strict sense. Finally, let us note that we have postponed the presentation of **M5** at the end of the list, because it plays a specific role that deserves additional comments.

M1 Every line of $\boldsymbol{H}$ has a null sum (as transfers preserve the overall activity).

M2 The diagonal of $\boldsymbol{H}$ is uniformly positive (by convention, a positive α_i corresponds to an increase of voxel i).

M3 $\boldsymbol{H}$ is non-positive elsewhere (A node can *decrease* the value of neighbors by transferring its own value, the *increase* of a voxel is associated to a *decrease* of its neighbors).

M4 $\boldsymbol{H}$ is symmetric *i.e.*, $H_{ij} = H_{ji}$ (the off-diagonal coefficients represent proximity between voxels: H_{ij} represents the amount of transfer from voxel i to voxel j when i transfers its value, which should be the same as the transfer from j to i when j transfers its value.)

Remark. *Assumptions **M1**-**M4** impose the matrix to be a Laplacian matrix, associated with a particular non-oriented graph corresponding to the PET image, where the nodes are the voxels, and a transfer coefficient is assigned to each edge. For an introduction on graphs and Laplacian matrices, see [21], [22].*

As a consequence of being a Laplacian matrix, $\boldsymbol{H}$ is a positive semi-definite matrix of rank $n-1$, every principal sub-matrix of $\boldsymbol{H}$ is invertible, and its inverse has no negative coefficients [23].

O1 $\boldsymbol{H}$ corresponds to a connected image, *i.e.*, given any two voxels, it is possible to link them with a path that crosses only edges where $H_{ij} \neq 0$.

O2 It is natural that elementary transfers only involve direct neighbors, in order for the post-processing to be as local as possible. The inclusion of second and third-order neighbors[2] may be a matter of choice.

O3 A simple choice for $\boldsymbol{H}$ is that it results from a spatially invariant and isotropic elementary mask, *i.e.*, a spatially symmetric convolution operator. The convolution kernel can be given by $(-0.5, 1, -0.5)$ for a 1D image, or

$$\begin{pmatrix} 0 & -0.25 & 0 \\ -0.25 & 1 & -0.25 \\ 0 & -0.25 & 0 \end{pmatrix} \tag{1}$$

in 2D, for instance.

2) Additional requirement on $\boldsymbol{\alpha}$: One goal of the NNEPPS is to correct the physical inconsistencies of unconstrained algorithms by giving back to each voxel its true meaning: negative values are meaningful only to correct the too large values of neighboring voxels, and negative voxels should give back this value to their neighbors. However, the problem is asymmetric, and there is no reason for positive voxels to spread a potentially "too large" value. For this reason, we further impose the following mandatory requirement:

M5 Every component of $\boldsymbol{\alpha}$ is non-negative (which is denoted $\boldsymbol{\alpha} \geq \mathbf{0}_n$).

Numerical tests have confirmed that the relaxation of the latter constraint generates post-processed images with many artifacts.

3) Choice of the norm: The post-processing procedure to remove non-negative values can be formulated as:

$$\min_{\boldsymbol{\alpha}} \|\boldsymbol{\alpha}\| \quad \text{such that} \tag{2a}$$

$$\begin{cases} \boldsymbol{\alpha} \geq \mathbf{0}_n \\ \boldsymbol{y} = \boldsymbol{x} + \boldsymbol{H}\boldsymbol{\alpha} \geq \mathbf{0}_n \end{cases} \tag{2b}$$

The following Proposition, proved in Appendix A, defines the notion of "minimal" $\boldsymbol{\alpha}$, independently of the chosen norm[3].

Proposition 1. *If $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are two transfer maps satisfying conditions (2b), the vector $\boldsymbol{\gamma}$ defined by $\forall i, \gamma_i = \min(\alpha_i, \beta_i)$ also satisfies (2b).*

[2]Voxels that share only an edge or a corner.

[3]Proposition 1 can be extended to an infinite number of transfer maps by noticing that (2) only involves closed subsets and continuous operations with respect to $\boldsymbol{\alpha}$.

Copyright (c) 2020 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TMI.2020.3003428

Proposition 1 has the following direct consequence.

Proposition 2. *(2a) is equivalent to* $\min_{\boldsymbol{\alpha}} d(\boldsymbol{\alpha})$*, for any scalar, coordinatewise increasing function* d*. In particular, all* L_p *norms,* $p \in]0, \infty[$*, induce equivalent problems.*

In the rest of the paper, the solution to (2) is called the *minimal transfer map*.

D. Solution existence and uniqueness

The properties of existence and uniqueness of the NNEPPS are discussed below. Proofs can be found in Appendices B and C, respectively.

1) Existence:

Proposition 3. *A solution to Problem (2) exists if and only if the mean value of the initial image* $\boldsymbol{X}$ *is non-negative.*

2) Uniqueness:

Proposition 4. *If a solution to Problem (2) exists, it is unique.*

III. Linear programming formulation and its resolution using an improved dual-simplex

The previous problem can be reformulated as a linear programming problem (Section III-A). The particular structure of this problem allows the dual simplex algorithm to be particularly efficient, especially, many iterations can be performed in parallel, and the entering indexes can be chosen in such a way that they are truly in the basic index set of the solution.

A. Linear programming formulation

The post-processing problem may be formulated as the minimization of $\mathbf{1}_n^T \boldsymbol{\alpha}$ while satisfying[4]:

$$\begin{cases} \boldsymbol{\alpha} \geq \mathbf{0}_n \\ \boldsymbol{y} = \boldsymbol{x} + \boldsymbol{H}\boldsymbol{\alpha} \geq \mathbf{0}_n \end{cases} \tag{3}$$

This problem can be re-written in the standard form of linear programming in the following way [20]: minimize $\boldsymbol{c}^T \boldsymbol{z}$ over $\boldsymbol{z}$ subject to: $\begin{cases} \boldsymbol{A}\boldsymbol{z} = -\boldsymbol{x} \\ \boldsymbol{z} \geq \mathbf{0}_{2n} \end{cases}$ where $\boldsymbol{c} = \begin{pmatrix} \mathbf{0}_n \\ \mathbf{1}_n \end{pmatrix}$, $\boldsymbol{z} = \begin{pmatrix} \boldsymbol{y} \\ \boldsymbol{\alpha} \end{pmatrix}$ and $\boldsymbol{A} = \begin{pmatrix} -\boldsymbol{I}_n & \boldsymbol{H} \end{pmatrix}$

As imposed by some authors for the standard formulation of linear programming, note that here $\boldsymbol{A}$ has strictly more columns than rows (respectively $2n$ and n) and is full row rank.

Each linear programming problem comes with a related dual problem [20]. From the standard form of the primal problem, we can infer that its associate dual problem is to maximize $-\boldsymbol{x}^T \boldsymbol{\pi}$ while satisfying $\boldsymbol{A}^T \boldsymbol{\pi} \leq \boldsymbol{c}$. This can be written in its standard form of linear programming as follows:

$$\text{minimize } \boldsymbol{b}^T \boldsymbol{s} \text{ subject to: } \begin{cases} \boldsymbol{G}\boldsymbol{s} = \mathbf{1}_n \\ \boldsymbol{s} \geq \mathbf{0}_{2n} \end{cases}$$

[4]As seen in Proposition 2, the minimization of $\boldsymbol{\alpha}$ is independent of the norm. For instance, the problem could have been formulated equivalently with the L2-norm. The L1-norm is chosen here to express the problem as a linear program.

with $\boldsymbol{b} = \begin{pmatrix} \boldsymbol{x} \\ \mathbf{0}_n \end{pmatrix}$, $\boldsymbol{s} = \begin{pmatrix} \boldsymbol{\pi} \\ \boldsymbol{t} \end{pmatrix}$, and $\boldsymbol{G} = \begin{pmatrix} \boldsymbol{H} & \boldsymbol{I}_n \end{pmatrix}$. $\boldsymbol{\pi}$, $\boldsymbol{t}$ and $\boldsymbol{s}$ are dual variables of size n, n and $2n$ respectively.

Note that again, $\boldsymbol{G}$ has strictly more columns than rows (respectively $2n$ and n) and is full row rank.

The Karush-Kuhn-Tucker (KKT) optimality conditions for the post-processing problem read:

$$\begin{cases} \boldsymbol{y} = \boldsymbol{x} + \boldsymbol{H}\boldsymbol{\alpha} & \text{(4a)} \\ \boldsymbol{t} = \mathbf{1}_n - \boldsymbol{H}\boldsymbol{\pi} & \text{(4b)} \\ \boldsymbol{s} \geq \mathbf{0}_{2n} & \text{(4c)} \\ \boldsymbol{z} \geq \mathbf{0}_{2n} & \text{(4d)} \\ \forall i \in [\![1, n]\!],\ \pi_i y_i = 0 & \text{(4e)} \\ \forall i \in [\![1, n]\!], t_i \alpha_i = 0 & \text{(4f)} \end{cases}$$

A pair $(\boldsymbol{x}, \boldsymbol{s})$ is a solution to the problem if and only if the KKT conditions are fulfilled.

The next section contains the outlines of the simplex algorithm applied to the dual of the post-processing problem. It will be shown later that this algorithm has particularly important properties in this case.

B. The dual-simplex algorithm

If a linear program has solutions, at least one of these solutions belongs to a category called *basic feasible point* [20].

A feasible point $\boldsymbol{s}$ (*i.e.*, a point satisfying $\boldsymbol{G}\boldsymbol{s} = \mathbf{1}_n$ and $\boldsymbol{s} \geq \mathbf{0}_{2n}$) is a basic feasible point if it has at most n nonzero components and if there exists a subset $\mathcal{B} \in [\![1, 2n]\!]$ — called the *basic index set* — verifying the three following properties [20]:

- $\mathcal{B}$ contains exactly n indexes
- $i \notin \mathcal{B} \implies s_i = 0$
- The $n \times n$ matrix $\boldsymbol{B} = \boldsymbol{G}_{[\![1,n]\!],\mathcal{B}}$ is invertible.

Basic feasible points are actually the vertices of the feasible polytope. We have already shown in Propositions 3 and 4 that if the mean activity is non-negative, which is actually the only interesting case, the post-processing problem has a unique solution, which is therefore a basic feasible point.

The simplex algorithm [24]–[26] is the most widely used solver for linear programming problems [20], [27]. This algorithm works by generating iteratively a sequence of *basic feasible points*. One simplex iteration is summarized in Algorithm 1.

Although the simplex algorithm usually requires at most $2n$ or $3n$ iterations to converge, its worst case has an exponential complexity [28]. This drawback has motivated the search for new algorithms that compete with the simplex method, especially on large scale problems [29], [30]. In the next section, an improved version of the simplex algorithm is proposed by exploiting the properties of the post-processing problem.

C. Resolution using an improved dual simplex algorithm

The size of the problems studied in this article is large (a PET image typically has 10^7 voxels). However, the simplex algorithm applied to the dual problem (simply called the dual-simplex algorithm [27], [31]), holds in this case very strong

Copyright (c) 2020 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TMI.2020.3003428

Algorithm 1 One iteration of the simplex algorithm

Given a basic index set $\mathcal{B}^{(k)}$ and its complement $\mathcal{N}^{(k)} = [\![1, 2n]\!] \setminus \mathcal{B}^{(k)}$
Set $\boldsymbol{s}^{(k)}_{\mathcal{N}^{(k)}} = \mathbf{0}_n$, and compute $\boldsymbol{s}^{(k)}_{\mathcal{B}^{(k)}}$ from (4b).
Set $\boldsymbol{z}^{(k)}_{\mathcal{B}^{(k)}} = \mathbf{0}_n$, and compute $\boldsymbol{z}^{(k)}_{\mathcal{N}^{(k)}}$ from (4a).
if $\boldsymbol{z} \geq \mathbf{0}_{2n}$ **then** declare success and **return** $\boldsymbol{z}$
else
    Choose $q \in \mathcal{N}^{(k)}$ such that $z_q < 0$ (the entering index).
    Increase s_q while maintaining $\boldsymbol{s}_{\mathcal{N}^{(k)} \setminus \{q\}} = \mathbf{0}_{n-1}$ and condition (4b), until one component p of $\boldsymbol{s}_{\mathcal{B}^{(k)}}$ is driven to 0 (*i.e.*, the leaving index).
    Set $\mathcal{B}^{(k+1)} = \mathcal{B}^{(k)} \cup \{q\} \setminus \{p\}$
end if

properties, presented in Theorem 1, which makes it highly efficient.

Theorem 1. *Problem (2) can be solved efficiently by a dual simplex algorithm, in a number of iterations bounded by n.*

Theorem 1 is implied by the following two propositions, which are proved in Appendices D and E respectively.

Proposition 5. *If the post-processing problem has a solution (see Proposition 3), the dual-simplex algorithm reaches this solution.*

Proposition 6. *Starting from $\boldsymbol{y}^{(0)} = \boldsymbol{x}$, in every dual-simplex iteration, all the possible entering indexes (all i for which $y_i^{(k)} < 0$) are indeed in the inactive set of the solution* i.e., *all possible choices are correct choices. On top of that, the leaving index associated to an entering one is known.*

Proof of Theorem 1: Proposition 5 shows that when a solution exists (*i.e.*, when the mean value of the image is non-negative, *cf.* Proposition 3), which is easy to check, the dual-simplex algorithm indeed reaches this solution after a finite number of iterations. Proposition 6 implies that one change in the basic index set is always correct. This means that even if the basic index set of the solution is completely different from the initial one, at most n iterations are necessary. Proposition 6 also implies that many changes in the basic index set can be done simultaneously since all the possible entering indexes can be included at once, and their associated leaving indexes are known. ■

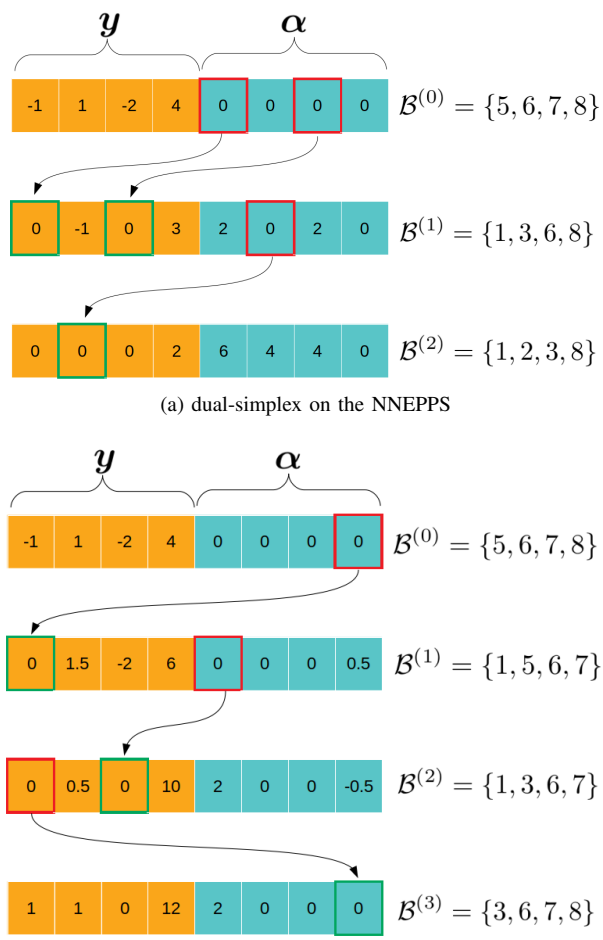
Following the steps of Algorithm 1, and taking into account the previous properties, the computation of $\boldsymbol{s}^{(k)}$ is not needed since the entering and leaving indexes are known from z. The algorithm can then be written as in Algorithm 2.

Fig. 2 gives a schematic view of the differences between the dual simplex applied to the NNEPPS (2a), and another linear program without any particular structure. In Fig. 2a, the entering index is always in the "$\boldsymbol{y}$ part" of $\boldsymbol{z}$, and the leaving index is the corresponding one in the $\boldsymbol{\alpha}$ part. Every entering index truly belongs to the final basis. Iterations are condensed. On the contrary, for a normal run of the dual-simplex (shown in Fig. 2b), only one index enters the basis at each iteration. This index can be anywhere, and there is no general rule to find

Algorithm 2 Summary of the algorithm

Initialize $\boldsymbol{y}^{(0)} = \boldsymbol{x}$, $k = 0$
while $\boldsymbol{y}^{(k)}$ contains negative voxels **do**
    $k = k + 1$
    create the set $\mathcal{Z}^{(k)}$ of all the non-positive voxels of $\boldsymbol{y}^{(k-1)}$
    create $\mathcal{P}^{(k)}$, the complement of $\mathcal{Z}^{(k)}$
    $\boldsymbol{\alpha}^{(k)}_{\mathcal{Z}^{(k)}} = -(\boldsymbol{H}_{\mathcal{Z}^{(k)}\mathcal{Z}^{(k)}})^{-1}\boldsymbol{x}_{\mathcal{Z}^{(k)}}$
    $\boldsymbol{\alpha}^{(k)}_{\mathcal{P}^{(k)}} = 0$
    $\boldsymbol{y}^{(k)} = \boldsymbol{x} + \boldsymbol{H}\boldsymbol{\alpha}^{(k)}$
end while
    return $\boldsymbol{y} = \boldsymbol{y}^{(k)}$

the corresponding leaving index. The choice of the entering index is not always optimal: during the first iteration, index 1 enters the basis but is later removed.



(a) dual-simplex on the NNEPPS

(b) dual simplex on another problem

Fig. 2. Optimization of the vector $\boldsymbol{z}$ using the dual-simplex algorithm for the NNEPPS (2a) and for another linear program without any particular structure (2b). The basis $\mathcal{B}$ is shown at each iteration. Entering indexes are framed in green, and leaving indexes in red.

D. Practical implementation details

1) Improvement of the algorithm (initialization step): Using PET images (matrix size $200 \times 200 \times 109$), approximately

Copyright (c) 2020 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TMI.2020.3003428

Algorithm 3 Initialization step

Given a number of voxels N chosen as stopping threshold:
while less than N new negative voxels were found **do**
  for i in every voxel of the image **do**
    if $y_i < 0$ **then**
      Increase the number of negative voxels found
      Set $y_i = 0$ and decrease y_j for all j neighbors of i such that the mean is preserved.
    end if
  end for
end while
return $\boldsymbol{y}$

TABLE I
TIME EVALUATION OF THE INITIALIZATION STEP

Tolerance	without init.		with init.		
	time	#iter	init. time	total time	#iter
10^{-3}	2'16"	14	13"	1'19"	7
10^{-6}	7'3"	15	40"	3'52"	6
2×10^{-16}	21'40"	15	1'33"	10'17"	5

15 condensed iterations were needed for the algorithm to converge. But actually, a better starting point $\boldsymbol{y}^{(0)}$ can be found quickly by following Algorithm 3. From this new starting point, the algorithm converges faster to the same output image. Table I compares the computing time and the number of iterations (#iter) needed without and with the initialization step (init.), for different levels of relative residual error upper bound on the conjugate gradient solver (Tolerance). This tolerance parameter corresponds well to the relative precision of the final image. It determines the amount of time needed per iteration. The stopping criterion of the initialization step has been optimized for each tolerance level.

2) Implementation: To test the algorithm, a C++ program has been developed. The linear systems are solved using the conjugate gradient method [32]–[34]. This method is an iterative linear solver that is well suited for sparse matrix, and especially large symmetric problems like ours. This algorithm allows the user to obtain fast approximate solutions, with a chosen precision level. The use of an approximate solution is justified here because of the limited number of post-processing iterations, and the fact that the computed voxel values are not used in later iterations: the only goal of the iterations of the NNEPPS is to find which voxels are null in the final image.

3) Choice of the Laplacian matrix $\boldsymbol{H}$: In Section II-C1, some requirements on $\boldsymbol{H}$ have been formulated, as well as good practices. However, there is still some leeway for the choice of the neighborhood and the off-diagonal coefficients. The precise effect of such choices still has to be studied in detail and is let for a future work. Here, we restrict ourselves to the first-order neighborhood (so that each pixel has four neighbors in 2D, and each voxel has six in 3D). In our 2D simulations, the pixels were square, and the coefficients were evenly distributed between neighbors so that each was assigned a coefficient of 0.25. For 3D real PET images, voxels were cuboids of size $4\,\mathrm{mm} \times 4\,\mathrm{mm} \times 2\,\mathrm{mm}$. Coefficient values near 0.28 and 0.11 were respectively assigned to the two neighbors in the axial direction and to the four ones in the lateral direction. Such values have been derived from theoretical considerations and from numerical simulations, reported as a supplementary material. Boundaries of the image were naturally considered by preserving the same coefficients for the existing neighbors and adapting the diagonal coefficient to be the opposite sum of the neighbors coefficients. For instance, for a 1D image of 4 pixels, $\boldsymbol{H}$ would be:

$$\begin{pmatrix} 0.5 & -0.5 & 0 & 0 \\ -0.5 & 1 & -0.5 & 0 \\ 0 & -0.5 & 1 & -0.5 \\ 0 & 0 & -0.5 & 0.5 \end{pmatrix} \tag{5}$$

IV. NUMERICAL ILLUSTRATIONS

The efficiency of the proposed post-processing approach is firstly analyzed on a simulated image before its application to a real data-set from ^{90}Y-PET imaging.

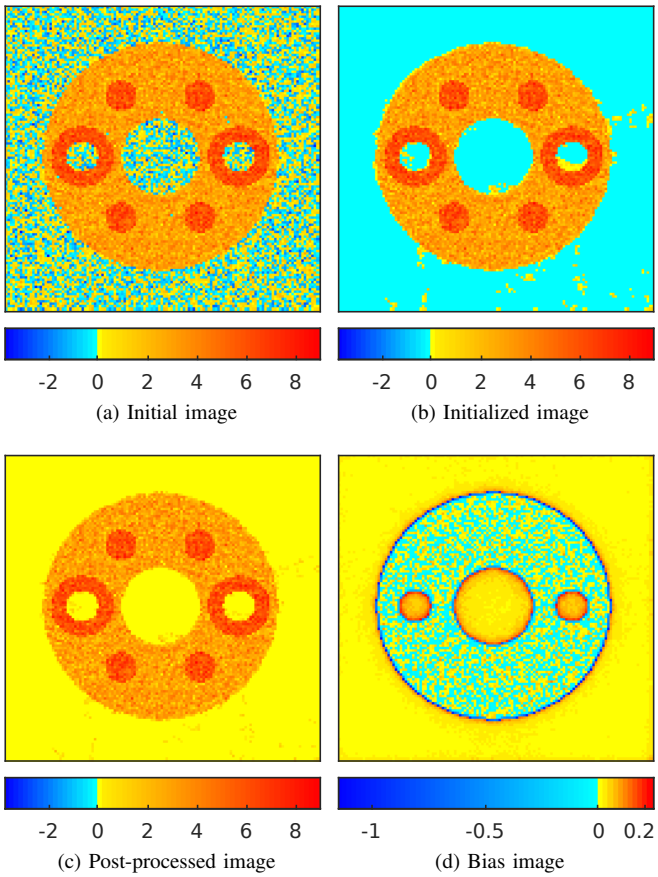
A. Example on a phantom

Fig. 3a shows a simple image with distinct areas, taking their mean intensity level among three values: 0, 3 and 6. Moreover, the intensity in each area is perturbed by an additive Gaussian noise of zero mean and unit standard deviation, to simulate reconstruction errors. One can notice in the resulting image after the NNEPPS (Fig. 3c) that the main modifications concern the zero value area. The intensities of initially negative pixels are set to zero and their surrounding pixels are either canceled or attenuated thanks to the local mean preserving constraint. This effect can be seen more clearly in Fig. 4 that displays the histograms of both images (raw and post-processed images using the proposed algorithm). One can notice that negative pixels are set to zero while high-intensity pixels remain unchanged (areas of theoretical mean value (TMV) 3 and 6). Fig. 3b shows the initialized image[5]. It is close to the post-processed image but appears noisier and contains many negative values (the minimum voxel value being -0.0072). Note that we have chosen a specific colormap to make a sharp difference between positive and negative pixels in Fig. 3. Table II displays the characteristics of the different areas. The variance (Var.) of the post-processed image is drastically decreased, while the other characteristics are roughly unchanged. Results are also compared to an image obtained by a simple projection of the rough image on the set of non-negative images. Fig. 3d shows the systematic bias created by the NNEPPS (averaged over 1000 realizations). One can see that the small positive bias introduced in the cold area by the non-negativity constraint (roughly 0.02) is compensated by a negative bias at the boundary between non-zero and zero regions (roughly -0.6 for the outer boundary). However, this bias is very localized at the boundary pixels. This effect will be further examined from experimental data in Section V, and perspectives of refinements will be discussed in Section VI regarding this point.

[5]The Initialization step has been stopped after 100 whole paths over the image has led to a relative number of new voxels of less than 10^{-6}.

Copyright (c) 2020 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TMI.2020.3003428



(a) Initial image (b) Initialized image

(c) Post-processed image (d) Bias image

Fig. 3. Different images of the same phantom. (a) shows the initial image. The different parts are filled by a random process following a Gaussian distribution of standard deviation 1, with a mean of 0, 3, or 6 depending on the different areas. (b) shows the image after the initialization step. (c) shows the image after the NNEPPS. Regions with a mean activity of 0 are cleaned compared to the initial image, while the other areas remain unchanged. (d) shows the bias introduced by the NNEPPS, averaged over 1000 realizations.

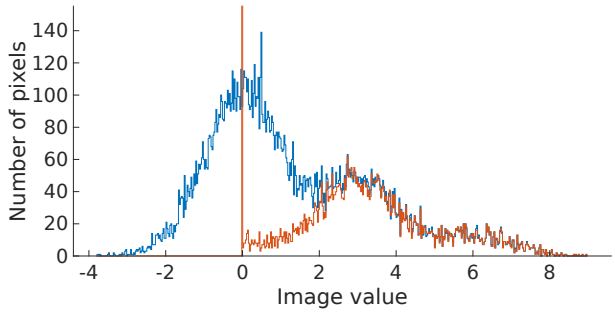


Fig. 4. Histogram representation of the two images of Fig. 3. The rough image is in blue, the post-processed one in red. In the post-processed image, 10 079 pixels belong to the bin containing 0.

B. Comparison between CPLEX and the proposed algorithm

Since the problem is a linear programming problem, one could wonder how classical linear programming techniques relate to the proposed algorithm.

Comparison of the proposed algorithm and classical linear programming algorithms have been performed and are displayed in Fig. 5.

The computations have been performed on a 64-bit Ubuntu 18.04.2 LTS with a RAM of 7.5 Go, the code was multi-threaded on 8 Intel Core i7-8550U CPU @ 1.80 GHz. An identical precision level is reached for each final image. The proposed algorithm is much faster than CPLEX, regardless of the chosen algorithm, and the computation time is not far from being linear with respect to the number of voxels. In comparison, the best CPLEX algorithm (the primal simplex) follows a power law with an exponent between 3 and 4. Given an image containing $200 \times 200 \times 109$ voxels and a precision of 10^{-3}, the proposed algorithm with a precision of 10^{-3} takes around 2 minutes. CPLEX is much too slow to be used: by extrapolation of the above curve, the computation time needed can be estimated to 80 years.

TABLE II
CHARACTERISTICS OF THE DIFFERENT AREAS

TMV	Rough image		Initialized image		Image after NNEPPS		Projected image	
	Mean	Var.	Mean	Var.	Mean	Var.	Mean	Var.
0	−0.018	1.00	0.0043	0.035	0.015	0.015	0.39	0.33
3	3.00	1.00	2.96	1.04	2.94	1.09	3.00	1.00
6	6.04	0.95	6.02	0.98	6.02	0.98	6.04	0.95

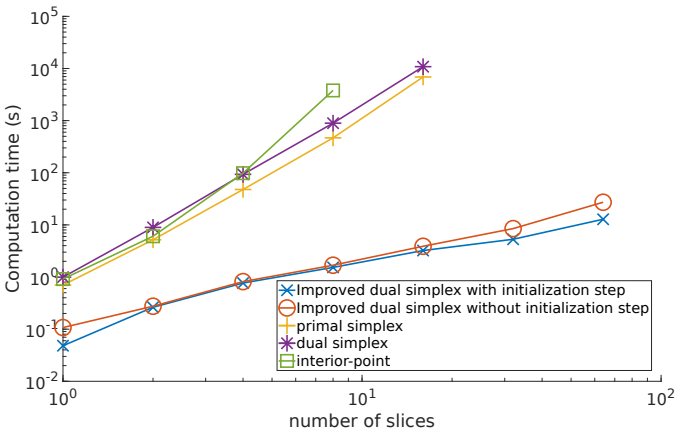


Fig. 5. Comparison of the time needed to post-process an example of image of size $100 \times 100 \times n$. Real PET images are typically of size $200 \times 200 \times 109$

V. Experimental results

To assess the impact of the NNEPPS in the context of PET image analysis, tests have been achieved on images produced by the AML algorithm [3], which allows negative values in the reconstructed image. These tests compare some figures of merit extracted from the image after the NNEPPS (AML+NNEPPS) with the original one (AML), and with an image produced by an algorithm with a built-in non-negativity constraint (OSEM). Reconstructions have been performed with the CASToR software [35].

A. Experimental set-up

In this work, every reconstruction algorithm has been used with 21 subsets. The phantom used for this study is the NEMA IEC.

Copyright (c) 2020 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TMI.2020.3003428

TABLE III
SUMMARY OF THE FOUR ACQUISITIONS DONE ON THE PHANTOM NEMA IEC IN 2013.

Phantom name	Acquisition day	Prompt number $\times 10^6$, % of IEC0	Random fraction	Scatter fraction	Trues / Prompts
IEC0	0	5.7 (100)	76 %	32 %	16 %
IEC1	3	3.5 (62)	82 %	30 %	13 %
IEC2	5	2.8 (50)	86 %	31 %	9.4 %
IEC3	7	2.4 (43)	90 %	34 %	6.4 %

Four acquisitions of the same phantom were performed during 7 days [7], [36]. On day 0, the six spheres of the phantom were filled with a solution of ^{90}Y with an activity concentration of $2.9\,\mathrm{MBq\,mL^{-1}}$. The background was filled with a solution of $^{90}YCl_3$ mixed with DTPA to prevent any sticking to the phantom walls and with a concentration 7.5 times lower than in the spheres. The central cylinder was isolated from any kind of activity. Acquisitions of $1800\,\mathrm{s}$ have then been performed[6] on days 0, 3, 5 and 7. The numbers of prompt coincidences collected for each acquisition are presented in table III.

B. Images

Fig. 6 shows the reconstruction of IEC1. Qualitatively, it can be reported a better "detectability" of the central cold cylinder in AML+NNEPPS while preserving the signal elsewhere.

Fig. 6 also shows that using OSEM reconstruction, the positive bias is clearly visible in the cold cylinder. Using AML, negative values are introduced, so the positive bias is suppressed (or reduced), but the cold areas become very noisy. After the NNEPPS, a fully null area becomes visible.

Fig. 7 shows clinical images acquired on a patient treated with ^{90}Y micro-spheres. The positive bias is difficult to highlight since the ground truth is not known for the patient. However, one can observe the cleaning of the background noise performed by the NNEPPS. The injected activity was $1.71\,\mathrm{GBq}$ at the time of imaging. The field of view (FOV) activity was $1.65\,\mathrm{GBq}$ for OSEM, and $1.52\,\mathrm{GBq}$ for AML and AML+NNEPPS (the NNEPPS does not affect the FOV activity).

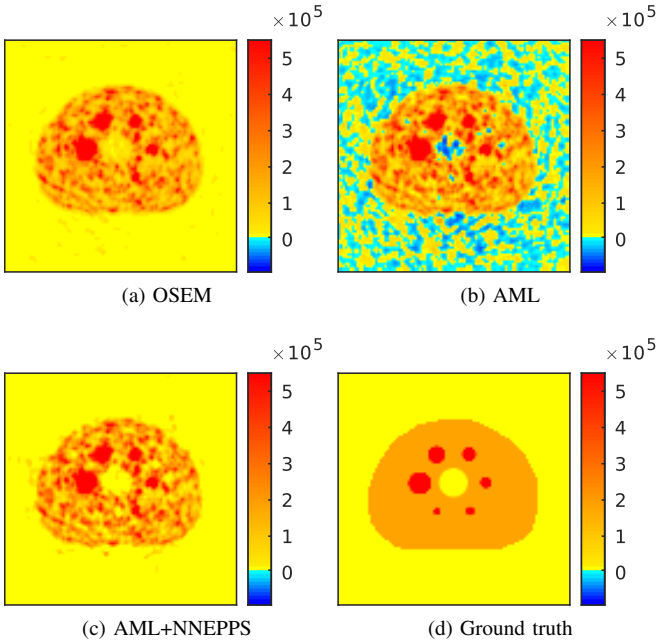
C. Figures of merit

Fig. 8 compares the mean activity as a function of the root-mean-square error (RMSE) for AML without and with the NNEPPS, at different iteration numbers, and different values of A (which is the parameter of AML indicating the lowest possible value of the image).

Two observations may be derived:

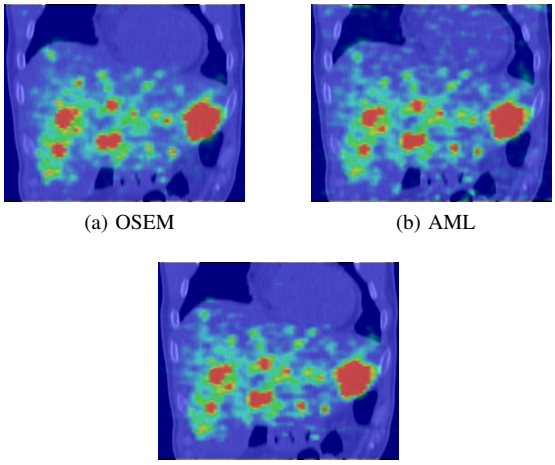
1) The approach allows one to decrease the RMSE, which is roughly divided by two when compared with the image before the NNEPPS. However, this decrease depends on several parameters: for OSEM, the NNEPPS does not affect the image (and thus the RMSE) since the image is already non-negative. For IEC3, with $A = -10^6\,\mathrm{Bq\,mL^{-1}}$, Fig. 8b shows that after three iterations, the RMSE is equal to $2.0 \times 10^5\,\mathrm{Bq\,mL^{-1}}$ for AML, but decreases to $4.3 \times 10^4\,\mathrm{Bq\,mL^{-1}}$ after the NNEPPS, which makes a reduction by a factor of 4.6.
2) The mean activity of the cold cylinder remains approximately unchanged by the NNEPPS until it reaches a limit (of approximately $2.5 \times 10^4\,\mathrm{Bq\,mL^{-1}}$ for IEC1). For example, for $A = -2 \times 10^5\,\mathrm{Bq\,mL^{-1}}$, the activity at the first AML iteration is around $3.4 \times 10^4\,\mathrm{Bq\,mL^{-1}}$

[6]Same duration as the clinical routine for the ^{90}Y-PET scans



(a) OSEM (b) AML (c) AML+NNEPPS (d) Ground truth

Fig. 6. Reconstruction of a phantom with OSEM, AML and AML after the NNEPPS (AML+NNEPPS). In AML+NNEPPS, the cold area at the center is much clearer.

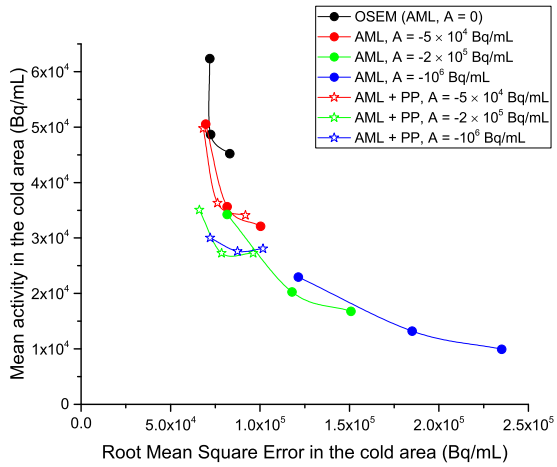


(a) OSEM (b) AML (c) AML+NNEPPS

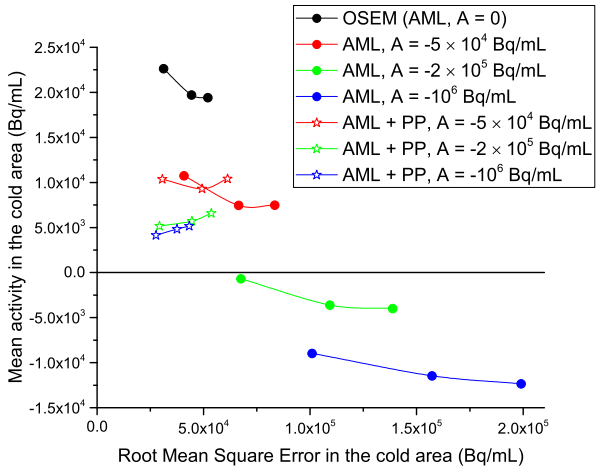
Fig. 7. Reconstruction of a ^{90}Y patient with OSEM, AML and AML after the NNEPPS (AML+NNEPPS). In AML+NNEPPS, the background noise introduced by AML is removed. The images display jointly the PET images and the CT scans. The patient was treated for liver localizations of a neuroendocrine tumor. Activity injected: 2190 MBq (1709 MBq at the time of imaging). Shunt: 4 %. Number of prompts: 3.1×10^6. Random fraction: 81 %.

Copyright (c) 2020 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TMI.2020.3003428



(a) IEC1, real mean activity: 0



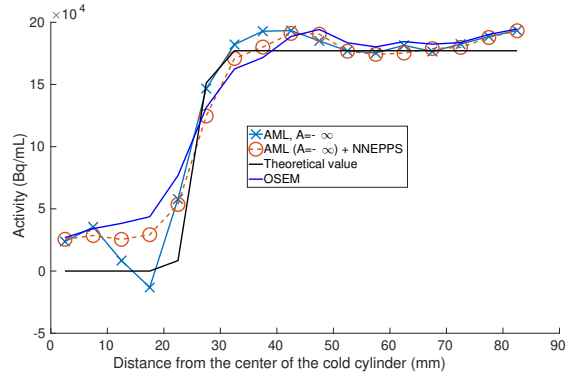
(b) IEC3, real mean activity: 0

Fig. 8. Graphs displaying the mean activity in the cold cylinder as a function of the RMSE in the same area. Tests have been done for different values of the AML parameter A, which is the lower bound of the image [3]. Each point in the same curve represent a different iteration number of AML. PP stands for NNEPPS. The curves for IEC0 and IEC2 are not shown but are close to IEC1, which is representative of the standard behavior of AML and the NNEPPS. IEC3 is an uncommon case on which AML completely underestimates the activity in the cold cylinder.

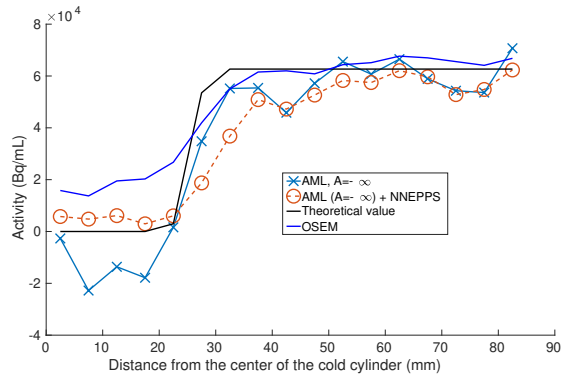
before the NNEPPS and remains the same after. But at the second AML iteration, the activity before the NNEPPS is lower than the limit (around $2.0 \times 10^4\,\mathrm{Bq\,mL^{-1}}$) so the activity after is approximately equal to the limit.

Therefore, the activity is not strictly preserved in the cold cylinder while the NNEPPS is supposed to preserve the local means. A small part of the negative activity has spread to the neighboring warm regions, increasing the activity in the true cold cylinder, which is thus not totally set to 0, even if the initial mean activity is 0. This was unavoidable because of the non-negativity properties of the post-processed image.

The reduction of the activity in the neighboring areas (already observed in Fig. 3d on a numerical example), is illustrated on the phantom data in Fig. 9, which shows the mean activity in a series of concentric hollow cylinders.



(a) IEC1



(b) IEC3

Fig. 9. Graphs displaying the mean activity measured as a function of the distance from the center of the cold cylinder. The theoretical value and OSEM are given for reference. The curves for IEC0 and IEC2 are not shown but are close to IEC1, which is representative of the standard behavior of AML and the NNEPPS. IEC3 is an uncommon case on which AML completely underestimates the activity in the cold cylinder.

The mean activity in the close neighborhood of the cold cylinder is indeed reduced, but this effect is limited and localized. Fig. 9a shows that the induced negative bias is up to 15% in the very close neighborhood of the cold cylinder for IEC1, but it decreases rapidly and completely vanishes at 10 mm to 15 mm away from the boundary. Moreover, in this area, the estimated activity is obtained with an error that is comparable to the one produced by OSEM. Fig. 9b shows an uncommon case, based on very low statistics, where AML completely under-estimates the activity in the cold cylinder. After the NNEPPS, the image is better estimated in the cold cylinder, but the nearby region has to compensate for this increase of activity. This is an unavoidable consequence of the fact that global activity is preserved. Compared to OSEM, the NNEPPS is finally less biased in the cold region, but slightly more in the close neighborhood of the latter.

VI. Discussion and Conclusion

Many currently used iterative reconstruction algorithms, such as OSEM, incorporate a built-in positivity constraint. This constraint is natural from a physical point of view. However, when associated with noise, it induces a significant positive bias in cold areas, which can be problematic when considering quantitative analysis. This problem is especially important in

Copyright (c) 2020 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TMI.2020.3003428

the low-statistics context of yttrium-90. The use of algorithms that are not subject to this positivity constraint may represent a solution, but at the cost of an even higher noise level and a lack of physical meaning for negative values.

Based on considerations on the information carried by negative-valued voxels, we propose a non-negativity enforcement post-processing step (NNEPPS), preserving the mean image activity as locally as possible. The NNEPPS introduces a small positive bias in cold areas, but it is successful in preserving this positive bias at a much lower level than OSEM. This small positive bias has a corresponding small negative bias in areas close to cold ones. The effect of this induced negative bias on small lesions should be evaluated in the future, but here again, this negative bias remains lower than that of OSEM. Let us mention an interesting perspective to further reduce the number of transfers between regions, and thus the created biases, which would consist in extending the post-processing step to an anisotropic spread, in the spirit of anisotropic diffusion [37].

The RMSE in cold areas is also reduced by the NNEPPS to a level comparable to that of OSEM. Qualitatively, the images are also visually easier to interpret after the NNEPPS. In particular, cold areas are much more visible. The method has also shown signs of robustness in performing well on an uncommon case (see Figs. 8b and 9b).

An algorithm executing the NNEPPS is proposed and its source code is available in Supplementary Material. It acts as a condensed simplex on the dual problem, which is justified by the particular nature of the problem. The time required to post-process a whole 3D PET image is about 2 minutes.

Among several perspectives, one would be to include the NNEPPS in the reconstruction process by applying the method after each iteration of an iterative algorithm. However, the convergence properties of such an algorithm, as well as its effect on the final image, have yet to be studied.

Future work could also include a precise study of the type of neighborhood to be considered and the choice of the coefficients of $\boldsymbol{H}$.

Finally, the NNEPPS is not intrinsically limited to PET. Its application could be extended to other imaging modalities, where a non-negativity constraint is imposed from a physical perspective. One could think about MRI modalities, where a positive bias has also been reported [38], [39], or the diffusion with decay [40]–[42]. Moreover, the principle of NNEPPS is neither limited to non-negativity constraints: the constraint could be any threshold, corresponding to a physical extremal value in the image.

Appendix A
Proof of Proposition 1

Proof: It is obvious that $\forall i, \gamma_i \geq 0$. Let i be an arbitrary index and without loss of generality, suppose that $\alpha_i \leq \beta_i$. Then $\gamma_i = \alpha_i$. Since $\boldsymbol{H}$ has a positive diagonal, its other entries being non-positive, the function $\left\{\begin{array}{l} \mathbb{R}^N \to \mathbb{R} \\ \boldsymbol{\alpha} \mapsto (\boldsymbol{x} + \boldsymbol{H}\boldsymbol{\alpha})_i \end{array}\right.$ is increasing relatively to the i-th variable, and decreasing relatively to the others. Since $\gamma_i \geq \alpha_i$ (because $\gamma_i = \alpha_i$), and $\forall j \neq i, \gamma_j \leq \alpha_j$ (because $\forall j, \gamma_j \leq \alpha_j$), we have $(\boldsymbol{x} + \boldsymbol{H}\boldsymbol{\gamma})_i \geq (\boldsymbol{x} + \boldsymbol{H}\boldsymbol{\alpha})_i \geq 0$. ■

Appendix B
Proof of Proposition 3

Proof: If the mean value of the initial image is negative, it is obviously impossible to find an image with uniformly non-negative voxels and a preserved mean.

Let us also show the converse. First, we will show that according to the properties of $\boldsymbol{H}$, it is possible to find an $\boldsymbol{\alpha}$ such that $\boldsymbol{y} \geq \boldsymbol{0}_n$. Then we will create a uniformly positive $\boldsymbol{\alpha}$ from the previous one. As mentioned in the remark of Section II-C1, matrix $\boldsymbol{H}$ is of rank $n-1$, and every principal sub-matrix of $\boldsymbol{H}$ is invertible. Let i be any voxel, and $\mathcal{I} = [\![1, n]\!] \setminus i$. Since $\boldsymbol{H}_{\mathcal{I}\mathcal{I}}$ is invertible, one can find $\tilde{\boldsymbol{\alpha}}$ such that $\boldsymbol{x}_{\mathcal{I}} + \boldsymbol{H}_{\mathcal{I}\mathcal{I}}\tilde{\boldsymbol{\alpha}}_{\mathcal{I}} = \boldsymbol{0}$. Let us set

$$\left\{\begin{array}{l} \tilde{\boldsymbol{\alpha}}_{\mathcal{I}} = -\boldsymbol{H}_{\mathcal{I}\mathcal{I}}^{-1}\boldsymbol{x}_{\mathcal{I}} \\ \tilde{\alpha}_i = 0 \end{array}\right. \tag{6}$$

This gives us the final image $\boldsymbol{y} = \boldsymbol{x} + \boldsymbol{H}\tilde{\boldsymbol{\alpha}}$. Because $\boldsymbol{x}$ and $\boldsymbol{y}$ share the same (positive) mean, and given that $\boldsymbol{y}_{\mathcal{I}} = 0$, the i-th voxel verifies $\boldsymbol{y}_i \geq 0$. So the final image is non-negative. However, the constraint $\tilde{\boldsymbol{\alpha}} \geq 0$ is not necessarily fulfilled by the previous process. Because constant vectors belong to the kernel of $\boldsymbol{H}$, any such vector can be added to $\tilde{\boldsymbol{\alpha}}$ with no change to the final image, and thus $\boldsymbol{\alpha}$ can be made non-negative. The existence of the solution is therefore proved. ■

Appendix C
Proof of Proposition 4

Proof: This is a direct consequence of how the minimal $\boldsymbol{\alpha}$ is defined: its i-th coordinate is the infimum of the i-th coordinates of every map satisfying (2b). ■

Appendix D
Termination of the dual-simplex

The use of a dual simplex is justified by its finite termination, which is proved in this section. Indeed, provided that the linear program is non-degenerate and bounded, the simplex method terminates at a basic feasible point [20]. Let us show the two parts:

Lemma 1. *The dual problem is not degenerate*

Proof: In a degenerate linear program, there exists at least one basic feasible point that has fewer than n non-zero components. Let us show that each feasible point of the dual problem has at least n non-zero components.

If a point is feasible, $\boldsymbol{H}\boldsymbol{\pi} + \boldsymbol{t} = \boldsymbol{1}_n$. But $\pi_i = 0$ implies that $(\boldsymbol{H}\boldsymbol{\pi})_i \leq 0$, and therefore $\boldsymbol{t}_i \neq 0$. So, to be feasible, a point $\begin{pmatrix} \boldsymbol{\pi} \\ \boldsymbol{t} \end{pmatrix}$ must have at least n non-zero components.

The dual problem is not degenerate. ■

Lemma 2. *If the mean of the initial value is non-negative, the dual problem is bounded.*

Proof: This is a direct consequence of the feasibility of the primal problem in this case (see Proposition 3). ■

Copyright (c) 2020 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TMI.2020.3003428

The dual-simplex algorithm, therefore, terminates if the mean of the initial image is non-negative.

Appendix E
The choice of the entering index is always correct

We choose a natural starting point in the sense that $\boldsymbol{y}^{(0)} = \boldsymbol{x}$, that is, we choose $\mathcal{B}^{(0)} = [\![n+1, 2n]\!]$. Let us first verify that this leads to a basic feasible point:

This choice corresponds to $\boldsymbol{\pi} = \mathbf{0}_n$ and $\boldsymbol{t} = \mathbf{1}_n$.

- $\boldsymbol{s}$ is then a feasible point of the dual problem presented in Section III-A, with at most (actually exactly) n non-zero components.
- $\mathcal{B}^{(0)}$ is a subset of $[\![1, 2n]\!]$ such that:
  - $\mathcal{B}^{(0)}$ contains n indexes
  - $i \notin \mathcal{B}^{(0)} \implies z_i = 0$
  - The $n \times n$ matrix $\boldsymbol{B}^{(0)} = \boldsymbol{G}_{[\![1,2n]\!],\mathcal{B}}$ (so $\boldsymbol{B}^{(0)} = \boldsymbol{I}_n$) is invertible.

The initial point is, therefore, a basic feasible point of the dual problem. The point here is that the entering index q must always be chosen such that $z_q = \begin{pmatrix} \boldsymbol{y} \\ \boldsymbol{\alpha} \end{pmatrix}_q < 0$.

The following two lemmata will be shown by induction.

Lemma 3. *$\boldsymbol{\alpha}$ always remains non-negative, and therefore the entering index always has to be chosen in $[\![1, n]\!]$*

Lemma 4. *If the chosen entering index is q, the leaving index is $n+q$.*

Lemmata 3 and 4 imply that the possible entering indexes are always in $[\![1, n]\!]$, and the leaving indexes in $[\![n+1, 2n]\!]$. Therefore, it is known that any choice of the entering index will be an index that will never leave and is indeed in the basic index set of the solution. On top of that, since the leaving indexes associated with the entering ones are known, it is possible to condense the iterations efficiently.

Proof of Lemmata 3 and 4: The two lemmata will be shown in a single demonstration by induction.

The base case is already done since in the initial state, $\boldsymbol{\alpha}^{(0)} = \mathbf{0}_n$.

Inductive step: since $\boldsymbol{\alpha}^{(k)} \geq 0$, the entering index has to be chosen in $[\![1, n]\!]$, and such that $y_q^{(k)} < 0$. A simplex iteration would consist in the increase of $\pi_q^{(k)}$ while maintaining $\boldsymbol{G}\boldsymbol{s}^{(k+1)} = \mathbf{1}_n$, and $\boldsymbol{z}^{(k+1)}_{\mathcal{N}^{(k)}\setminus q} = \mathbf{0}_{n-1}$. If $\mathcal{Z}^{(k)} = [\![1, n]\!] \cap \mathcal{B}^{(k)}$, and $\mathcal{P}^{(k)} = [\![1, n]\!] \setminus \mathcal{B}^{(k)}$, from the induction hypothesis of Lemma 4, $\mathcal{B}^{(k)} = \mathcal{Z}^{(k)} \cup (n + \mathcal{P}^{(k)})$, implying that $\boldsymbol{\pi}^{(k)}_{\mathcal{P}^{(k)}} = \mathbf{0}$ and $\boldsymbol{t}^{(k)}_{\mathcal{Z}^{(k)}} = \mathbf{0}$. Therefore:

$$\boldsymbol{H}_{\mathcal{Z}^{(k)}\mathcal{Z}^{(k)}} \boldsymbol{\pi}^{(k+1)}_{\mathcal{Z}^{(k)}} + \boldsymbol{H}_{\mathcal{Z}^{(k)}q} \pi_q^{(k+1)} = \mathbf{1} = \boldsymbol{H}_{\mathcal{Z}^{(k)}\mathcal{Z}^{(k)}} \boldsymbol{\pi}^{(k)}_{\mathcal{Z}^{(k)}}$$

$$\begin{aligned} \boldsymbol{H}_{\mathcal{P}^{(k)}\mathcal{Z}^{(k)}} \boldsymbol{\pi}^{(k+1)}_{\mathcal{Z}^{(k)}} + \boldsymbol{H}_{\mathcal{P}^{(k)}q} \pi_q^{(k+1)} + \boldsymbol{t}^{(k+1)}_{\mathcal{P}^{(k)}} &= \mathbf{1} \\ &= \boldsymbol{H}_{\mathcal{P}^{(k)}\mathcal{Z}^{(k)}} \boldsymbol{\pi}^{(k)}_{\mathcal{Z}^{(k)}} + \boldsymbol{t}^{(k)}_{\mathcal{P}^{(k)}} \end{aligned}$$

which leads to:

$$\begin{aligned} \boldsymbol{\pi}^{(k+1)}_{\mathcal{Z}^{(k)}} &= \boldsymbol{\pi}^{(k)}_{\mathcal{Z}^{(k)}} - (\boldsymbol{H}_{\mathcal{Z}^{(k)}\mathcal{Z}^{(k)}})^{-1} \boldsymbol{H}_{\mathcal{Z}^{(k)}q} \pi_q^{(k+1)} \\ \boldsymbol{t}^{(k+1)}_{\mathcal{P}^{(k)}} &= \boldsymbol{t}^{(k)}_{\mathcal{P}^{(k)}} + \big(\boldsymbol{H}_{\mathcal{P}^{(k)}\mathcal{Z}^{(k)}} (\boldsymbol{H}_{\mathcal{Z}^{(k)}\mathcal{Z}^{(k)}})^{-1} \\ &\qquad \boldsymbol{H}_{\mathcal{Z}^{(k)}q} - \boldsymbol{H}_{\mathcal{P}^{(k)}q}\big) \pi_q^{(k+1)} \end{aligned}$$

Moreover, according to Remark II-C1, all the coefficients of $(\boldsymbol{H}_{\mathcal{Z}^{(k)}\mathcal{Z}^{(k)}})^{-1}$ are non-negative, and $\boldsymbol{H}_{\mathcal{Z}^{(k)}q} \leq 0$ (since q is outside $\mathcal{Z}^{(k)}$). This implies that all the components of $\boldsymbol{\alpha}^{(k+1)}$ are increasing with $\pi_q^{(k+1)}$. In the same way, we have: $\boldsymbol{H}_{\mathcal{P}^{(k)}\mathcal{Z}^{(k)}} \leq 0$, $(\boldsymbol{H}_{\mathcal{Z}^{(k)}\mathcal{Z}^{(k)}})^{-1} \geq 0$, $\boldsymbol{H}_{\mathcal{Z}^{(k)}q} \leq 0$. We finally have $(\boldsymbol{H}_{\mathcal{P}^{(k)}q})_{i \neq q} \leq 0$ and $(\boldsymbol{H}_{\mathcal{P}^{(k)}q})_q = H_{qq} \geq 0$. Therefore all the components of $\boldsymbol{t}_{\mathcal{P}^{(k)}}$ are increasing, except the q-th component, on which we can *a priori* infer nothing. But since the dual-simplex terminates (see Appendix III-C), one index must be leaded to 1. It is therefore necessarily $n+q$.

Now, according to equation (4a); and since $\boldsymbol{y}^{(k+1)}_{\mathcal{Z}^{(k+1)}} = \mathbf{0}$, and $\boldsymbol{\alpha}^{(k+1)}_{\mathcal{P}^{(k+1)}} = \mathbf{0}$; $\mathbf{0} = \boldsymbol{x}_{\mathcal{Z}^{(k+1)}} + \boldsymbol{H}_{\mathcal{Z}^{(k+1)}\mathcal{Z}^{(k+1)}} \boldsymbol{\alpha}^{k+1}_{\mathcal{Z}^{(k+1)}}$, which leads to $\boldsymbol{\alpha}^{k+1}_{\mathcal{Z}^{(k+1)}} = -(\boldsymbol{H}_{\mathcal{Z}^{(k+1)}\mathcal{Z}^{(k+1)}})^{-1} \boldsymbol{x}_{\mathcal{Z}^{(k+1)}}$. In the same way, and since $\mathcal{P}^{(k+1)} \subset \mathcal{P}^{(k)}$;

$$\boldsymbol{y}^{(k)}_{\mathcal{Z}^{(k+1)}} = x_{\mathcal{Z}^{(k+1)}} + \boldsymbol{H}_{\mathcal{Z}^{(k+1)}\mathcal{Z}^{(k+1)}} \boldsymbol{\alpha}^{k}_{\mathcal{Z}^{(k+1)}}$$

Coupled with the above form of $\boldsymbol{\alpha}^{k}_{\mathcal{Z}^{(k+1)}}$, we obtain

$$\boldsymbol{\alpha}^{(k+1)}_{\mathcal{Z}^{(k+1)}} - \boldsymbol{\alpha}^{(k)}_{\mathcal{Z}^{(k+1)}} = -(\boldsymbol{H}_{\mathcal{Z}^{(k+1)}\mathcal{Z}^{(k+1)}})^{-1} \boldsymbol{y}^{(k)}_{\mathcal{Z}^{(k+1)}}$$

Moreover, $\boldsymbol{y}^{(k)}_{\mathcal{Z}^{(k+1)}} \leq \mathbf{0}$ (because the entering index is chosen among negative voxels). And therefore, $\boldsymbol{\alpha}^{(k+1)}_{\mathcal{Z}^{(k+1)}} - \boldsymbol{\alpha}^{(k)}_{\mathcal{Z}^{(k+1)}} \geq \mathbf{0}$. So, $\boldsymbol{\alpha}^{(k+1)} \geq \mathbf{0}_n$. ■

Acknowledgment

The authors would like to thank Simon Stute for his help concerning the reconstruction part and Clément Bailly for his medical contribution.

References

[1] T. Carlier and C. Bailly, "State-Of-The-Art and Recent Advances in Quantification for Therapeutic Follow-Up in Oncology Using PET," *Frontiers in Medicine*, vol. 2, Mar 2015. [Online]. Available: http://journal.frontiersin.org/Article/10.3389/fmed.2015.00018/abstract

[2] D. L. Bailey, D. W. Townsend, P. E. Valk, and M. N. Maisey, *Positron emission tomography*. Springer, 2005.

[3] K. van Slambrouck *et al.*, "Bias reduction for low-statistics PET: Maximum likelihood reconstruction with a modified poisson distribution," *IEEE Trans. Med. Imag.*, vol. 34, no. 1, pp. 126–136, Jan. 2015.

[4] R. Salem, V. Mazzaferro, and B. Sangro, "Yttrium 90 radioembolization for the treatment of hepatocellular carcinoma: Biological lessons, current challenges, and clinical perspectives," *Hepatology*, vol. 58, no. 6, pp. 2188–2197, 2013.

[5] R. G. Selwyn, R. J. Nickles, B. Thomadsen, L. A. DeWerd, and J. A. Micka, "A new internal pair production branching ratio of 90Y: The development of a non-destructive assay for 90Y and 90Sr," *Applied Radiation and Isotopes*, vol. 65, no. 3, pp. 318–327, 2007.

[6] R. J. Nickles *et al.*, "Assaying and PET Imaging of Ytrrium-90: 1>>34ppm>0," in *IEEE Symposium Conference Record Nuclear Science 2004.*, vol. 6, Oct 2004, pp. 3412–3414.

[7] T. Carlier, K. P. Willowson, E. Fourkal, D. L. Bailey, M. Doss, and M. Conti, "^{90}Y-PET imaging : Exploring limitations and accuracy under conditions of low counts and high random fraction." *Med. Phys.*, vol. 42, no. 7, pp. 4295–4309, 2015.

[8] L. A. Shepp and Y. Vardi, "Maximum likelihood reconstruction for emission tomography," *IEEE Trans. Med. Imag.*, vol. 1, no. 2, pp. 113–122, Oct. 1982.

[9] K. Lange and R. Carson, "EM reconstruction algorithms for emission and transmission tomography," *Journal of Computer Assisted Tomography*, vol. 8, no. 2, pp. 306–316, Apr. 1984.

[10] H. M. Hudson, B. F. Hutton, and R. Larkin, "Accelerated EM reconstruction using ordered subsets," *The Journal of Nuclear Medicine*, vol. 33, p. 960, 1992.

[11] H. M. Hudson and R. Larkin, "Accelerated image reconstruction using ordered subsets of projection data," *IEEE Trans. Med. Imag.*, vol. 13, pp. 601–609, Aug. 1994.

Copyright (c) 2020 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.

This is the author's version of an article that has been published in this journal. Changes were made to this version by the publisher prior to publication.
The final version of record is available at http://dx.doi.org/10.1109/TMI.2020.3003428

[12] M. D. Walker *et al.*, "Bias in iterative reconstruction of low-statistics PET data: benefits of a resolution model," *Physics in Medicine and Biology*, vol. 56, no. 4, pp. 931–949, jan 2011. [Online]. Available: https://doi.org/10.1088%2F0031-9155%2F56%2F4%2F004

[13] C. Byrne, "Iterative algorithms for deblurring and deconvolution with constraints," *Inverse Problems*, vol. 14, no. 6, pp. 1455–1467, 1998.

[14] J. Nuyts, S. Stroobants, P. Dupont, S. Vleugels, P. Flamen, and L. Mortelmans, "Reducing loss of image quality because of the attenuation artifact in uncorrected PET whole-body images," *J. Nucl. Med.*, vol. 43, no. 8, pp. 1054–1062, 2002.

[15] H. Lim, Y. K. Dewaraja, and J. A. Fessler, "A PET reconstruction formulation that enforces non-negativity in projection space for bias reduction in y-90 imaging," *Phys. Med. Biol.*, vol. 63, no. 3, pp. 35–42, Feb. 2018.

[16] A. Bousse *et al.*, "PET Reconstruction with non-Negativity Constraint in Projection Space: Optimization Through Hypo-Convergence," *IEEE Transactions on Medical Imaging*, 2019.

[17] G. Monge, "Mémoire sur la théorie des déblais et des remblais." *Histoire de l'Académie Royale des Sciences*, pp. 666–704, 1781.

[18] L. V. Kantorovitch, "On translation of mass," *Proceedings of the USSR Academy of Sciences*, no. 37, pp. 199–201, 1942.

[19] G. Peyré and M. Cuturi, "Computational optimal transport," *Foundations and Trends in Machine Learning*, vol. 11, no. 5-6, pp. 355–607, 2019.

[20] J. Nocedal and S. J. Wright, *Numerical Optimization*. Springer, 1999.

[21] F. R. Chung and F. C. Graham, *Spectral graph theory*. American Mathematical Soc., 1997, no. 92.

[22] E. W. Weisstein. Laplacian matrix. [Online]. Available: http://mathworld.wolfram.com/LaplacianMatrix.html

[23] U. Miekkala, "Graph properties for splitting with grounded laplacian matrices," *BIT Numerical Mathematics*, vol. 33, no. 3, pp. 485–495, Sep 1993. [Online]. Available: https://doi.org/10.1007/BF01990530

[24] G. B. Dantzig, "Programming in a linear structure." *U. S. Air Force Comptroller, USAF, Washington, D.C.*, 1948.

[25] ——, "Maximization of a linear function of variables subject to linear inequalities." *Activity Analysis of Production and Allocation, New York*, p. 339–347, 1951.

[26] G. B. Dantzig, A. Orden, and P. Wolfe, "The generalized simplex method for minimizing a linear form under linear inequality restraints." *Pacific Journal of Mathematics*, no. 5, p. 183–195, 1955.

[27] A. Koberstein, "The dual simplex method, techniques for a fast and stable implementation," Ph.D. dissertation, Fakultät für Wirtschaftswissenschaften der Universität Paderborn, November 2005.

[28] V. Klee and G. J. Minty, "How good is the simplex algorithm?" *in Inequalities, O. Shisha, ed., Academic Press, New York*, pp. 159–175, 1972.

[29] L. G. Khachiyan, "A polynomial algorithm in linear programming," *Soviet Mathematics Doklady*, no. 20, pp. 191–194, 1979.

[30] N. Karmarkar, "A new polynomial-time algorithm for linear programming," *Combinatorics*, no. 4, pp. 373–395, 1984.

[31] C. E. Lemke, "The dual method of solving the linear programming problem." *Naval Research Logistics Quarterly*, no. 1, p. 36–47, 1954.

[32] M. R. Hestenes and E. Stiefel, "Methods of conjugate gradients for solving linear systems," *Journal of Research of the National Bureau of Standards*, vol. 49, no. 6, pp. 409–436, Dec. 1952.

[33] K. A. Atkinson, *An introduction to numerical analysis*, 2nd ed. John Wiley and Sons, 1988.

[34] M. Avriel, *Nonlinear Programming: Analysis and Methods*. Dover Publishing, 2003.

[35] T. Merlin *et al.*, "CASToR: a generic data organization and processing code framework for multi-modal and multi-dimensional tomographic reconstruction," *Physics in Medicine & Biology*, vol. 63, no. 18, p. 185005, sep 2018.

[36] K. P. Willowson, M. Tapner, and D. L. Bailey, "A multicentre comparison of quantitative ^{90}Y PET/CT for dosimetric purposes after radioembolization with resin microspheres," *Eur J Nucl Med Mol Imaging*, no. 42, pp. 1202–1222, 2015.

[37] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 7, pp. 629–639, July 1990.

[38] H. Gudbjartsson and S. Patz, "The Rician distribution of noisy MRI data," *Magnetic resonance in medicine*, vol. 34, no. 6, pp. 910–914, 1995.

[39] A. Andersen and J. Kirsch, "Analysis of noise in phase contrast MR imaging," *Medical Physics*, vol. 23, no. 6, pp. 857–869, 1996.

[40] H. Nagarajan and K. Nakshatrala, "Enforcing the non-negativity constraint and maximum principles for diffusion with decay on general computational grids," *International Journal for Numerical Methods in Fluids*, vol. 67, no. 7, pp. 820–847, 2011.

[41] K. Nakshatrala and A. J. Valocchi, "Non-negative mixed finite element formulations for a tensorial diffusion equation," *Journal of Computational Physics*, vol. 228, no. 18, pp. 6726–6752, 2009.

[42] K. Nakshatrala, H. Nagarajan, and M. Shabouei, "A numerical methodology for enforcing maximum principles and the non-negative constraint for transient diffusion equations," *Communications in Computational Physics*, vol. 19, no. 1, pp. 53–93, 2016.

Copyright (c) 2020 IEEE. Personal use is permitted. For any other purposes, permission must be obtained from the IEEE by emailing pubs-permissions@ieee.org.