

## Research Article

# ADMM-EM Method for $L_1$ -Norm Regularized Weighted Least Squares PET Reconstruction

Yueyang Teng,<sup>1</sup> Hang Sun,<sup>1</sup> Chen Guo,<sup>2</sup> and Yan Kang<sup>1</sup>

<sup>1</sup>Sino-Dutch Biomedical and Information Engineering School, Northeastern University, Shenyang 110004, China

<sup>2</sup>Shenyang Branch of Philips Healthcare (Suzhou) Co., Ltd., Shenyang 110004, China

Correspondence should be addressed to Yan Kang; kangy@neusoft.com

Received 8 December 2015; Accepted 26 September 2016

Academic Editor: Chuangyin Dang

Copyright © 2016 Yueyang Teng et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The  $L_1$ -norm regularization is usually used in positron emission tomography (PET) reconstruction to suppress noise artifacts while preserving edges. The alternating direction method of multipliers (ADMM) is proven to be effective for solving this problem. It sequentially updates the additional variables, image pixels, and Lagrangian multipliers. Difficulties lie in obtaining a nonnegative update of the image. And classic ADMM requires updating the image by greedy iteration to minimize the cost function, which is computationally expensive. In this paper, we consider a specific application of ADMM to the  $L_1$ -norm regularized weighted least squares PET reconstruction problem. Main contribution is derivation of a new approach to iteratively and monotonically update the image while self-constraining in the nonnegativity region and the absence of a predetermined step size. We give a rigorous convergence proof on the quadratic subproblem of the ADMM algorithm considered in the paper. A simplified version is also developed by replacing the minima of the image-related cost function by one iteration that only decreases it. The experimental results show that the proposed algorithm with greedy iterations provides a faster convergence than other commonly used methods. Furthermore, the simplified version gives a comparable reconstructed result with far lower computational costs.

## 1. Introduction

Positron emission tomography (PET) is an important imaging tool in modern medicine and provides noninvasive quantification of the biochemical and biological processes inside living subjects. Several reconstruction methods have been developed and applied in clinical practice. These methods can be roughly divided into two categories: analytical methods and iterative methods. Iterative methods have attracted more attention because they generally model imaging physics better and are more capable of suppressing noise artifacts than analytical methods. A basic target of PET reconstruction is to solve a system of the following form:

$$Y = PX + S, \quad (1)$$

where  $X$  represents the biochemical activity distribution inside a subject,  $Y$  denotes the measured projections,  $S$  is scatter and random events, and  $P$  is a predetermined system matrix, where  $X \in R^{N \times 1}$ ,  $Y \in R^{M \times 1}$ ,  $S \in R^{M \times 1}$ , and  $P \in R^{M \times N}$ . All of these are fully nonnegative.

A classical approach is to select  $X$  such that the weighted least squares (WLS) error [1] between  $Y$  and  $PX + S$  is minimized:

$$\min_{X \geq 0} F(X) = \|PX + S - Y\|_{\Sigma}^2. \quad (2)$$

$\Sigma$  is a diagonal matrix with the diagonal element  $\Sigma_{ii} = 1/\sigma_i$ , where  $\sigma_i$  is the variance of the  $i$ th measurement. In theory, the values in  $Y$  are larger than those in  $S$ ; however, in practical applications, the latter one could exceed the former one when estimating  $S$ .

Although an iterative algorithm can more effectively suppress noise propagation than conventional filtered back-projection, it may fail in special cases, such as increased amount of noise, sparse projections, or low dose (which results in high noise or poor SNR). In fact, the problem of reconstructing PET image data is an ill-posed inverse problem. Over the past twenty years, efforts have been made to solve these problems by employing regularization techniques [2–8]. A general method is to introduce *a priori* knowledge

to constrain the solution space, which can be expressed as a regularization (or penalization) on the reconstructed image to reflect information on the properties of acceptable images. Tikhonov regularization [9, 10] is a popular method that generally leads to a unique solution. There are many methods for solving such a quadratic programming problem [1, 11]. However, they exhibit a weak ability to preserve the edges while smoothing the interior of the image.

Another viable regularization method is the  $L_1$ -norm regularization [12–16], in which one seeks to find the solution of the following optimization problem:

$$\min_{X \geq 0} F(X) + \beta \|RX\|_1, \quad (3)$$

where  $\beta > 0$  serves as a penalty parameter and  $R$  is a linear operator (e.g., gradient operator and orthonormal transformation). The penalty parameter  $\beta$  controls the trade-off between data fidelity and resolution (image smoothness). Several linear operators  $R$  have been proposed, such as the first- or second-order derivative or wavelet basis. Note that  $R$  could include negative elements. The underlying philosophy in working with  $L_1$ -norm regularization is that most images have a sparse representation [17].

In recent years, the subgradient-based method [18] has been developed for solving convex and nonconvex optimization problems; this method takes a subgradient related surrogate function at each step to obtain the update. Another method is the alternating direction method of multipliers (ADMM) [19, 20]. ADMM decomposes the original problem into three subproblems, and then it sequentially solves these subproblems at each iteration. For medical imaging, ADMM will distributively minimize the augmented Lagrangian-related function to solve for the additional and primal variables (pixels), and then it updates the dual variables, which are associated with a coupling constraint.

One can reformulate the optimization problem in (3) by imposing the extra constraint  $V = RX$ , which leads to the following optimization problem:

$$\begin{aligned} \min \quad & F(X) + \beta \|V\|_1 \\ \text{s.t.} \quad & X \geq 0, \\ & V = RX. \end{aligned} \quad (4)$$

Note that (3) and (4) have the same solution. The scaled augmented Lagrangian function [19] is introduced to overcome these difficulties, and it is defined as follows:

$$\begin{aligned} L(X, V, \mu) = F(X) + \beta \|V\|_1 + \left(\frac{\rho}{2}\right) \|RX - V + \mu\|_2^2 \\ - \left(\frac{\rho}{2}\right) \|\mu\|_2^2, \end{aligned} \quad (5)$$

where  $\mu$  is the dual variable or Lagrange multiplier and  $\rho > 0$  is the penalty parameter. When  $\rho = 0$ , the augmented Lagrangian can be reduced to the unaugmented (common) version.

By applying distribution optimization for  $V$  and  $X$  and dual ascent to  $\mu$ , a unified framework can be introduced to solve the  $L_1$ -norm regularized WLS reconstruction

problems. For the  $V$ -update, one exploits the separability  $L(X^t, V, \mu^t)$  in  $V$ , that is,  $L(X^t, V, \mu^t) = \sum_{i=1}^M L_i(X^t, V_i, \mu^t)$ , to solve for each  $V_i$  independently. A solution can be found in [17], which is the well-known shrinkage method. To update  $\mu$ , a simple gradient-ascent method can be used. When not considering the details of the  $X$ -update, the ADMM's framework can be formulated [19] as follows.

*Algorithm 1* (ADMM general framework). One has

$$\begin{aligned} (1) \quad & V_i^{t+1} = (|(RX^t + \mu^t)_i| - \beta/\rho)_+ \text{sgn}[(RX^t + \mu^t)_i], \quad i = 1, \dots, M; \\ (2) \quad & X^{t+1} = \arg \min_{X \geq 0} L(X, V^{t+1}, \mu^t); \\ (3) \quad & \mu^{t+1} = \mu^t + RX^{t+1} - V^{t+1}. \end{aligned}$$

Here, the  $X$ -update is a difficult problem that minimizes the following function with the nonnegativity constraint:

$$L(X, V^{t+1}, \mu^t) = F(X) + \left(\frac{\rho}{2}\right) \|RX - V^{t+1} + \mu^t\|_2^2. \quad (6)$$

When  $2P^T P + \rho R^T R$  is invertible, we may obtain a unique global solution, which has been used in [21, 22].

$$\begin{aligned} X \\ = (2P^T P + \rho R^T R)^{-1} [2P^T (Y - S) + \rho R^T (V^{t+1} - \mu^t)] \\ X = \max(0, X), \end{aligned} \quad (7)$$

where the truncation below zero is necessary for constraining the solution to the nonnegative space. This formulation is feasible in theory if the regularization process guarantees the nonsingularity of the matrix.

A viable method is the gradient-based method. The steepest descent method is perhaps the simplest technique to implement, which takes the negative gradient as the descent direction:

$$\begin{aligned} X^{t+1} = X^t - \alpha(t) \nabla L(X^t, V^{t+1}, \mu^t) \\ X^{t+1} = \max(X^{t+1}, 0), \end{aligned} \quad (8)$$

where the superscript  $t$  denotes the  $t$ th iteration and  $\alpha(t)$  is the step size. Of course, negative pixel values still need to be truncated. The convergence of the gradient-based methods depends on the choice of step size, which is problematic for practical implementations.

Let  $\alpha(t) = \alpha$  be a constant, where  $0 < \alpha < 2/\|\nabla^2 L_X(X^t, V^{t+1}, \mu^t)\|$  ( $\|\cdot\|$  denotes the maximum eigenvalue); then (8) becomes the projected Landweber method [23]:

$$\begin{aligned} X^{t+1} = X^t - \alpha \nabla L(X^t, V^{t+1}, \mu^t) \\ X^{t+1} = \max(X^{t+1}, 0). \end{aligned} \quad (9)$$

The conjugate gradient method is a popular approach, which is often implemented as an iterative algorithm applicable to sparse systems for large-scale problems.

However, for large-scale problems such as PET reconstruction, the inverse matrix method becomes very costly from a computational perspective. Furthermore, the “pure” steepest descent method and the “pure” conjugate gradient method do not meet the nonnegativity constraint, and consequently, negative pixel values need to be truncated. The truncation, however, leads to a divergent modification. The projected Landweber method, however, is able to meet the nonnegativity, but its convergence cannot be proven theoretically. In addition, the projected algorithms also destroy the monotonic decreasing properties of the decomposed cost function.

In fact, ADMM is a framework of distribution optimization, which is not limited to PET reconstruction with the nonnegativity constraints. With different constraints and distributed variables, ADMM can be applicable in many other fields, such as multiple-block convex programming [24], ADMM for tomography with nonlocal regularizers [25], linear classification [26], and optimal power flow problems [27].

In this paper, we consider a specific application of ADMM to the  $L_1$ -norm regularized WLS reconstruction problems. Here, we do not change the framework of ADMM; rather, we develop a new algorithm for PET image reconstruction. The proposed approach is applicable to several medical image reconstruction problems, such as TV regularized [13] and wavelet regularized [14] image reconstruction. We focus on a key subproblem: the  $X$ -update. A multiplicative update rule is derived to iteratively and monotonically (in the sense of decreasing cost function) update the pixel values. Similar to the EM algorithm [28, 29], the pixel-update algorithm also intrinsically satisfies the automatic satisfaction of the nonnegativity constraint without the need for an adjustable step size. We provide a rigorous convergence proof for the proposed  $X$ -update, which shows that the algorithm will iteratively pursue a single global optimum. The  $X$ -update that optimizes the subproblem inevitably leads to high computational costs, and we can replace it by a single iteration algorithm to decrease the decomposed cost function of the subproblem, which is an often used strategy in distributed optimization. The experimental results demonstrate that the proposed algorithm (with greedy reconstruction of the pixels) provides better performance compared to those of other commonly used methods with respect to image qualification and convergence speed. The results also show that the simplified version provides a comparable reconstructed result but at a considerably lower computational cost compared to the existing methods.

## 2. Methodology

For notational simplicity, we define  $C = -V^{t+1} + \mu^t$  as a constant vector, and then we define the cost function in (6) as follows:

$$\Phi(X) = L(X, V^{t+1}, \mu^t) = F(X) + \left(\frac{\rho}{2}\right)G(X), \quad (10)$$

$$\text{where: } G(X) = \|RX + C\|_2^2.$$

We will solve this optimization problem using a modified EM-type algorithm. As mentioned in many articles [30–33], a surrogate function, as defined below, is useful in algorithm derivation and convergence proof.

*Definition 2* (surrogate). The function  $\psi(X | X^t)$  is a surrogate of  $\Psi(X)$  at  $X^t$  (fixed) if  $\psi(X^t | X^t) = \Psi(X^t)$  and  $\psi(X | X^t) \geq \Psi(X)$ .

Clearly,  $\Psi(X)$  is decreasing under the update  $X^{t+1} = \min_X \psi(X | X^t)$  because of

$$\Psi(X^{t+1}) \leq \psi(X^{t+1} | X^t) \leq \psi(X^t | X^t) = \Psi(X^t). \quad (11)$$

There are two important properties for the surrogate: additivity and transitivity. For the former, the sum of two surrogates is a surrogate of the sum of two original functions. For the latter, the surrogate of the surrogate of a function is a surrogate of this function. Following these properties, we will construct the surrogates for  $F(X)$  and  $G(X)$ .

*2.1. Surrogate for  $F(X)$ .* We construct a surrogate  $f(X | X^t)$  by the convexity. Let

$$\begin{aligned} \lambda_{i*} &= \frac{S_i}{(PX^t + S)_i}, \\ \lambda_{ij} &= \frac{P_{ij}X_j^t}{(PX^t + S)_i} \end{aligned} \quad (12)$$

that satisfy  $\lambda_{i*}, \lambda_{ij} \geq 0$  and  $\lambda_{i*} + \sum_{j=1}^N \lambda_{ij} = 1$ . They can be the convex combination coefficients such that

$$\begin{aligned} f(X | X^t) &= \sum_{i=1}^M \sigma_i \left[ \lambda_{i*} \left( \frac{S_i}{\lambda_{i*}} - Y_i \right)^2 + \sum_{j=1}^N \lambda_{ij} \left( \frac{P_{ij}X_j}{\lambda_{ij}} - Y_i \right)^2 \right]. \end{aligned} \quad (13)$$

It can be verified that  $f(X^t | X^t) = F(X^t)$ . When considering Jensen’s inequality and the convex combination coefficients  $\lambda_{ij}$ , then  $f(X | X^t) \geq F(X)$  is proven by the following inequality:

$$\begin{aligned} \lambda_{i*} \left( \frac{S_i}{\lambda_{i*}} - Y_i \right)^2 + \sum_{j=1}^N \lambda_{ij} \left( \frac{P_{ij}X_j}{\lambda_{ij}} - Y_i \right)^2 & \\ \geq [(PX)_i - Y_i + S_i]^2. \end{aligned} \quad (14)$$

A similar derivation can be found in [34]. Note that  $\lambda_{ij}^*$  only relates to the constant term with regard to  $X$  and can therefore be ignored when minimizing the function  $f(X | X^t)$ .

*2.2. Surrogate for  $G(X)$ .* Since there may be some negative values in the matrix  $R$ , it is difficult to directly construct a surrogate as above. Some previous works, such as [35, 36], are unable to solve the problem because they fail to guarantee nonnegativity during the iterations, for which we provided

a counterexample in [11]. We will utilize an intermediate surrogate to solve the problem. Let  $R = \bar{R} - \hat{R}$  and  $C = \bar{C} - \hat{C}$ , where  $\bar{R}, \hat{R}, \bar{C}$ , and  $\hat{C}$  are matrices or vectors with nonnegative entries. Subsequently, we can construct a surrogate for  $G(X)$  at  $X^t$ .

$$g_{\text{mid}}(X | X^t) = \bar{g}(X | X^t) + \hat{g}(X | X^t), \quad (15)$$

where

$$\begin{aligned} \bar{g}(X | X^t) &= \frac{1}{2} \left\| 2(\bar{R}X + \bar{C}) - (\bar{R}X^t + \bar{C}) - (\hat{R}X^t + \hat{C}) \right\|_2^2 \\ \hat{g}(X | X^t) &= \frac{1}{2} \left\| 2(\hat{R}X + \hat{C}) - (\bar{R}X^t + \bar{C}) - (\hat{R}X^t + \hat{C}) \right\|_2^2. \end{aligned} \quad (16)$$

It can be verified that  $g_{\text{mid}}(X^t | X^t) = G(X^t)$ . By the convexity of  $G(X)$ , we view 1/2 as the combination coefficients, leading to

$$\begin{aligned} g_{\text{mid}}(X | X^t) &= \frac{1}{2} \left\| 2(\bar{R}X + \bar{C}) - (\bar{R}X^t + \bar{C}) - (\hat{R}X^t + \hat{C}) \right\|_2^2 \\ &\quad + \frac{1}{2} \left\| -2(\hat{R}X + \hat{C}) + (\bar{R}X^t + \bar{C}) + (\hat{R}X^t + \hat{C}) \right\|_2^2 \\ &\geq G(X). \end{aligned} \quad (17)$$

Following the same process as in Section 2.1, we can construct surrogates  $\bar{g}(X | X^t)$  and  $\hat{g}(X | X^t)$  for  $\bar{g}(X | X^t)$  and  $\hat{g}(X | X^t)$ , respectively. Let

$$\begin{aligned} \bar{\lambda}_{ij} &= \frac{\bar{R}_{ij}X_j^t}{(\bar{R}X^t + \bar{C})_i} \\ \hat{\lambda}_{ij} &= \frac{\hat{R}_{ij}X_j^t}{(\hat{R}X^t + \hat{C})_i} \end{aligned} \quad (18)$$

$$X_j^{t+1} = X_j^t \frac{[A_1 + (\rho/2)A_2]_j}{[A_3 + \rho A_4]_j}, \quad (22)$$

$$\text{where } A_1 = P^T \Sigma Y, \quad A_2 = (\bar{R} + \hat{R})^T (\bar{R} + \hat{R}) X^t + (\bar{R} + \hat{R})^T (\bar{C} + \hat{C}), \quad A_3 = P^T \Sigma (PX^t + S), \quad A_4 = (\bar{R}^T \bar{R} + \hat{R}^T \hat{R}) X^t + \bar{R}^T \bar{C} + \hat{R}^T \hat{C}.$$

The update rule results from the current image multiplied by a factor and is flexible and easy to implement. The derivation process can also be explained in terms of EM optimization: when considering the surrogate as the minimal conditional expectation, its minimization is equivalent to the maximization of the conditional expectation. The algorithm shows two important properties: the iterations

then

$$\begin{aligned} \bar{\bar{g}}(X | X^t) &= \frac{1}{2} \sum_{i=1}^M \sum_{j=1}^N \bar{\lambda}_{ij} \left[ \frac{2\bar{R}_{ij}X_j}{\bar{\lambda}_{ij}} - ((\bar{R} + \hat{R}) X^t - \bar{C} - \hat{C})_j \right]^2 \\ \hat{\hat{g}}(X | X^t) &= \frac{1}{2} \sum_{i=1}^M \sum_{j=1}^N \hat{\lambda}_{ij} \left[ \frac{2\hat{R}_{ij}X_j}{\hat{\lambda}_{ij}} - ((\bar{R} + \hat{R}) X^t - \bar{C} - \hat{C})_j \right]^2. \end{aligned} \quad (19)$$

Note that  $\bar{\lambda}_{ij}^*$  and  $\hat{\lambda}_{ij}^*$  are relative to the constant terms and can safely be ignored. Now, we can obtain a surrogate for  $G(X)$  at  $X^t$ .

$$g(X | X^t) = \bar{\bar{g}}(X | X^t) + \hat{\hat{g}}(X | X^t). \quad (20)$$

**2.3. Multiplicative Update Rule.** We minimize  $\phi(X | X^t) = f(X | X^t) + (\rho/2)g(X | X^t)$  to obtain a new iteration. Taking the partial derivatives for  $f(X | X^t)$ ,  $\bar{g}(X | X^t)$  and  $\hat{g}(X | X^t)$  leads to

$$\begin{aligned} \frac{\partial f(X | X^t)}{\partial X_j} &= 2 \frac{[P^T \Sigma (PX^t + S)]_j}{X_j^t} X_j - 2(P^T \Sigma Y)_j \\ \frac{\partial \bar{g}(X | X^t)}{\partial X_j} &= 4 \frac{[\bar{R}^T (\bar{R}X^t + \bar{C})]_j}{X_j^t} X_j \\ &\quad - 2 \left\{ \bar{R}^T [(\bar{R} + \hat{R}) X^t + \bar{C} + \hat{C}] \right\}_j \\ \frac{\partial \hat{g}(X | X^t)}{\partial X_j} &= 4 \frac{[\hat{R}^T (\hat{R}X^t + \hat{C})]_j}{X_j^t} X_j \\ &\quad - 2 \left\{ \hat{R}^T [(\bar{R} + \hat{R}) X^t + \bar{C} + \hat{C}] \right\}_j. \end{aligned} \quad (21)$$

Solving the one-dimensional equations  $\partial \phi(X | X^t) / \partial X_j = 0$  leads to a multiplicative update rule, which is the main result in this paper.

are positive if the initial estimate is positive, and the cost function monotonically decreases. For this derivation process, the key step is to replace the minimization of the cost function by minimizing at each iteration the surrogate whose variables are separable. Moreover, the minimization of the surrogate ensures that the cost function decreases.

**2.4. Specific ADMM and Simplified Version.** Now, we can present a specific ADMM that is flexible and convenient for PET reconstruction.

*Algorithm 3* (specific ADMM). Given  $\beta > 0$  and  $\rho > 0$ , then

- (1)  $V_i^{t+1} = (|(RX^t + \mu^t)_i| - \beta/\rho)_+ \text{sgn}[(RX^t + \mu^t)_i]$ ,  $i = 1, \dots, M$ ;
- (2) iteratively update  $X$  by (22) until some stop rule is satisfied;
- (3)  $\mu^{t+1} = \mu^t + RX^{t+1} - V^{t+1}$ .

Note that, in Algorithm 3, ADMM requires greedy iterations to obtain the optimal solution with respect to  $X$ , which is an expensive operation. In general, an update that decreases the primal cost function is sufficient for use in practical applications; thus, we will provide the following simplified algorithm.

*Algorithm 4* (simplified ADMM). Given  $\beta > 0$  and  $\rho > 0$ , then

- (1)  $V_i^{t+1} = (|(RX^t + \mu^t)_i| - \beta/\rho)_+ \text{sgn}[(RX^t + \mu^t)_i]$ ,  $i = 1, \dots, M$ ;
- (2) update  $X$  by (22) with only one iteration;
- (3)  $\mu^{t+1} = \mu^t + RX^{t+1} - V^{t+1}$ .

### 3. Convergence

There are many convergence proofs for both constrained and unconstrained ADMM [19, 20]. Therefore, we only need to limit ourselves to discussing the convergence of the  $X$ -update. We will theoretically prove that the iteration sequence will converge to a global solution if we use it to pursue an accurate solution without considering the computational cost. In the appendix, we will prove that update (22) can iteratively and monotonically minimize the cost function (10) while observing the nonnegativity constraint along the lines of [35–39].

### 4. Experiments

**4.1. Simulated Head Phantom Data.** A simulated head phantom with  $128 \times 128$  pixels (pixel width of 4 mm), as shown in Figure 1, is used in the following experiments. This phantom is modified to meet the needs of PET simulation because the original one is a CT phantom. There are many advantages to using simulated phantoms, including prior knowledge of the pixel values and the ability to control noise. The total detection counts are approximately  $5 \times 10^5$ .

An anisotropic TV regularization is used to test the algorithmic performance as follows:

$$G(X) = \sum_{j=1}^N (|X_j - X_{j,\text{right}}| + |X_j - X_{j,\text{below}}|), \quad (23)$$

where  $X_{j,\text{right}}$  and  $X_{j,\text{below}}$ , respectively, represent pixels to the right and below  $X_j$ .

We compare the performance of the proposed method with De Pierro's ISRA (image space reconstruction method) [34] and the PWLS-EM algorithm with a quadratic smoothing regularization [11]. These methods are selected to demonstrate the difference between regularized and nonregularized reconstruction algorithms. Moreover, the difference between  $L_1$ -norm regularization and squared  $L_2$ -norm regularization is examined. We also use several methods to pursue  $X$ -update, including the projected Landweber and the conjugate gradient methods. The projected Landweber method uses  $\alpha = 1/[\|P\|_1 \|P\|_\infty + \beta \|R\|_1 \|R\|_\infty]$  [40], where  $\|\cdot\|_1$  and  $\|\cdot\|_\infty$  denote the 1- and  $\infty$ -norms of a matrix, respectively. The code of the conjugate gradient method comes from [41], which is slightly modified to meet our criteria.

In the following, the results of the ADMM-type algorithms are named ADMM- $x$ - $y$ - $z$ , where  $x$  represents the  $X$ -update method, which can be EM (proposed), PL (projected Landweber), and CG (conjugate gradient);  $y$  denotes the number of outer loops; and  $z$  is the number of inner loops. For example, ADMM-EM-400-120 refers to using the proposed method to update  $X$  with 120 inner iterations and update all of  $V$ ,  $X$ , and  $\mu$  with 400 outer iterations. In the unambiguous case, we will omit  $x$ ,  $y$ ,  $z$ , or all of them for notational simplicity. The experiments are performed on a HP Compaq PC with a 3.00 GHz Core i5 CPU and 4 GB of memory. The algorithms are implemented in MATLAB 7.0. All of the algorithms are initiated using the same uniform image for a fair comparison.

The system matrix is obtained using the ‘‘angle of view’’ method [28]. The diagonal matrix  $\Sigma$  is computed using Fessler's ‘‘data-plugin’’ technique [1].

Using the system matrix, we project the phantom on the sinogram with 128 radial bins (bin size of 4 mm) and 128 angular views evenly spaced over  $\pi$ . The noisy projections are obtained by Fessler's pseudorandom formulation.

$$y_i = c_i \text{Poisson} \{c_i^{-1} (y_i^* + a_i y_i^*)\} - c_i \text{Poisson} \{c_i^{-1} a_i y_i^*\}, \quad (24)$$

where  $y_i^*$  is the noise-free projection,  $a_i = 30\%$  simulates the contribution of random events, and  $c_i$  is the  $i$ th detector efficiency. We select  $c_i = 1$  and thus ignore the influence of the detector efficiency. Furthermore, we ignore  $S$  in (2) during the simulation process.

Mean absolute error (MAE) can be used to measure the proximity of the reconstructed image to the true image. The MAE value is calculated by taking the average of the absolute difference between the reconstructed pixel values and the real ones over the entire image. The best algorithm will provide the lowest MAE value.

$$\text{MAE}(t) = \frac{1}{N} \|X^t - X^{\text{True}}\|_1. \quad (25)$$

The following criterion is applied to stop the iteration process:

$$\chi(t) = \frac{\|X^{t+1} - X^t\|^2}{\|X^t\|^2} < \epsilon, \quad (26)$$

where  $\epsilon$  is a difference tolerance.





FIGURE 1: Shepp-Logan phantom with  $128 \times 128$  grids: (a) Shepp-Logan phantom, (b) background, (c) low-activity ROI, and (d) high-activity ROI.

Contrast and variability are typically used to evaluate image quality. We compute these parameters using the method of NEMA [42]. Denote  $\text{mean}_{\Omega}(X)$  and  $\text{std}_{\Omega}(X)$  as the mean and standard deviation of the image  $X$  on the region  $\Omega$ ; then

$$\begin{aligned} \text{Contrast} &= \frac{\text{mean}_{\Omega_1}(X) / \text{mean}_{\Omega_2}(X)}{\text{mean}_{\Omega_1}(X^{\text{True}}) / \text{mean}_{\Omega_2}(X^{\text{True}})} \\ \text{Variability} &= \frac{\text{std}_{\Omega_1}(X)}{\text{mean}_{\Omega_1}(X)}, \end{aligned} \quad (27)$$

where  $\Omega_1$  represents the ROI (region of interest) and  $\Omega_2$  denotes the background.

The running time of the algorithms is easily influenced by many factors, for example, coding level and running environment. Therefore, we also compare the computational complexity for the sake of fairness. Here, we use a simple and feasible method by counting the number of multiplications of the system matrix and any vector because the TV matrix ( $R$ ) is very sparse, and thus the computation load can be ignored. In fact, for all of ADMM-PL, ADMM-CG, and ADMM-EM,

each inner loop ( $X$ -update) requires only two multiplications of the system matrix and a vector; consequently, the computation complexity can be further simplified to the total number of inner loops, which equals the multiplications of the number of outer loops and inner loops for each  $X$ -update. In the following, we will take it to indicate the computational complexity.

In Figure 2, we find a suitable penalty parameter for PWLS-EM by comparing the change of the MAE at the 400th iteration with respect to  $\beta$ . We also pursue the optimal  $\beta$  and  $\rho$  for ADMM-EM by comparing the change of MAE at the 400th outer iteration and the 120th inner iteration. Since the  $X$ -update by EM can be proven in theory to converge to a single global optimum, ADMM-EM may be viewed as a golden standard. For PWLS-EM, the optimal MAE values are obtained at  $\beta = 10^{-4}$ . For ADMM-EM, the optimal penalty parameter is obtained at  $(\beta, \rho) = (10^{-2}, 10^{-4})$ . In addition, the minimal MAE values for PWLS-EM and ADMM-EM are 41.91 and 13.05, respectively. This phenomenon shows a clear advantage of  $L_1$ -norm regularization compared to Tikhonov-type regularization.

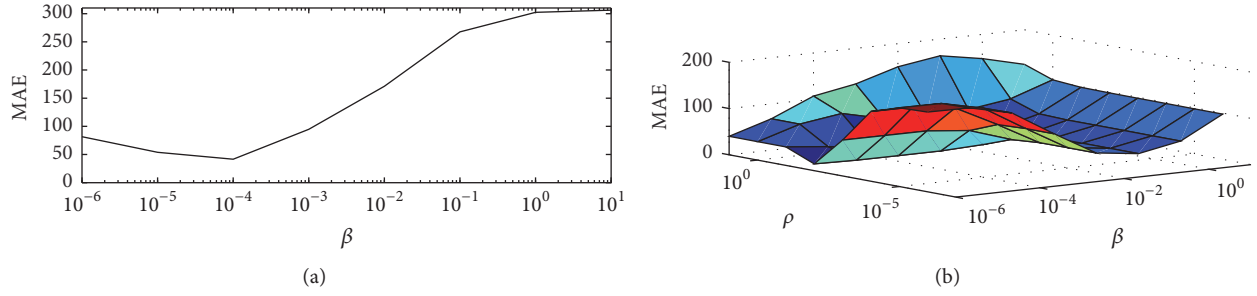


FIGURE 2: MAE values generated by the greedy enumeration strategy: (a) PWLS-EM with 400 iterations and (b) ADMM-EM with 400 outer iterations and 120 inner iterations. The global minimum MAE values correspond to the optimal penalty parameters. Note that both the  $x$ - and  $y$ -axes are log-scale to provide a clear visualization. PWLS-EM obtains the minimal MAE value (41.91) at  $\beta = 10^{-4}$ , and ADMM-EM obtains the minimal value (13.05) at  $(\beta, \rho) = (10^{-2}, 10^{-4})$ .

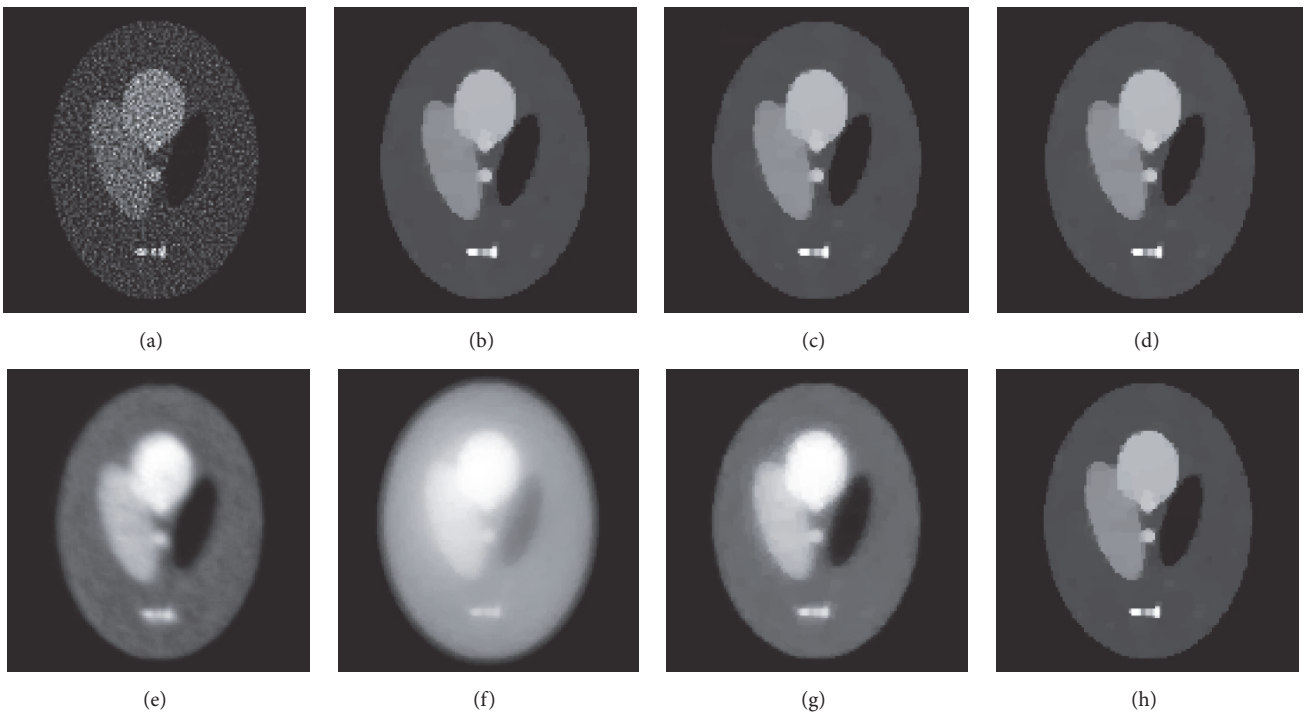


FIGURE 3: Reconstructed images by (a) ISRA with 400 iterations, (b) ADMM-PL-400-120, (c) ADMM-CG-400-120, (d) ADMM-EM-400-120, (e) PWLS-EM with 400 iterations, (f) ADMM-PL-400-1, (g) ADMM-CG-400-1, and (h) ADMM-EM-400-1.

Below, we focus on several special algorithms, including ISRA, PWLS-EM ( $\beta = 10^{-4}$ ), and ADMM ( $\beta = 10^{-2}$  and  $\rho = 10^{-4}$ ). Figure 3 shows the reconstructions corresponding to a greedy outer and inner iteration. As shown in this figure, ISRA's image suffers from serious noise artifacts, and PWLS-EM fails to preserve the edges. ADMM-PL-400-1 and ADMM-CG-400-1 provide an obvious blurred reconstruction. The remaining images exhibit smooth interiors and sharp edges, which are desirable results. These algorithms provide almost identical reconstructed results. In fact, since the subproblem to update the pixel values has a strictly convex cost function, every convergent algorithm will converge to the same result with a greedy iteration number. This is the reason for why ADMM-PL-400-120, ADMM-CG-400-120, and ADMM-EM-400-120 provide similar results. This

experiment also indicates that EM is more suitable for the simplified ADMM algorithm than PL and CG. We believe that the reason is because the EM- $X$ -update can ensure the monotonic decrease of the augmented Lagrangian function; however, the projected Landweber and CG have no such characteristic.

Figure 4 presents a comparison of the MAE curves and cost function with increasing iteration numbers for ADMM-PL, ADMM-CG, and ADMM-EM, in which the curves of ISRA and PWLS-EM disappear because they clearly fail to preserve edges and suppress noise artifacts. As shown, for both curves, ADMM-PL-1 presents larger MAE and cost function values and ADMM-CG-1 provides the next-worse curve, demonstrating the consistent conclusion that ADMM-PL and ADMM-CG are ill-suited for the proposed simplified

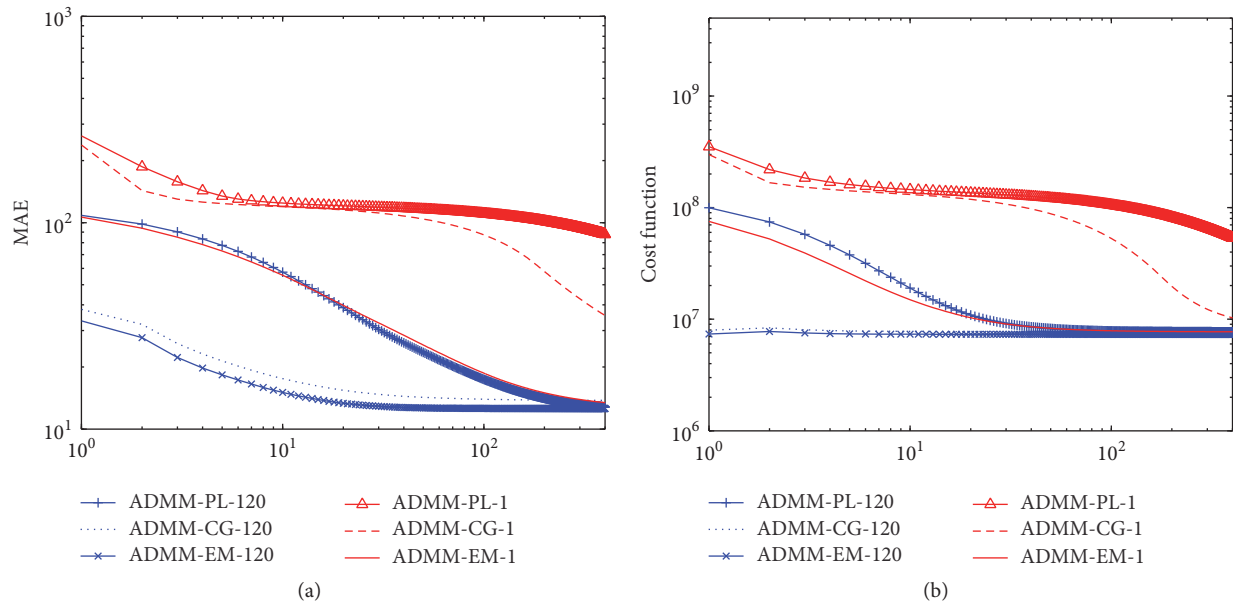


FIGURE 4: MAE and cost function versus the number of outer loops, where the greedy versions of ADMM use 120 iterations for the inner loop to update image pixels: (a) MAE and (b) cost function.

strategy. ADMM-EM-120 exhibits the fastest convergence rate; however, it is only slightly superior to ADMM-CG-120. ADMM-PL-120 and ADMM-EM-1 present similar curves. In conclusion, ADMM-EM-120 and ADMM-CG-120 perform better than ADMM-PL-120 and ADMM-EM-1 when only judging these curves, but eventually, all algorithms will approach the same target with increasing iteration numbers.

Note that ADMM-EM has been proven in theory to converge to a unique global solution of the corresponding subproblem with a greedy  $X$ -update. This means that when ignoring rounding error, our algorithm can arrive at any desired accuracy given a large enough iteration count. Thus, in these experiments, ADMM-PL and ADMM-CG will also converge to that solution if they are indeed convergent. However, to date, there is no evidence to show the convergence when negative values are truncated during the iteration process.

Table 1 presents the comparison of algorithmic performance, including MAE, cost function, contrast, variability, running time (second), and computational complexity. We only compare the greedy versions of ADMM-PL, ADMM-CG, and ADMM-EM with the simplified version of ADMM-EM because the simplified versions of ADMM-PL and ADMM-CG failed to obtain an acceptable image. Our findings show that either the greedy or simplified version of ADMM-EM always provides the best results. In addition, the greedy versions of ADMM with  $X$ -update by PL, CG, and EM impose a similar time cost and have the same computational complexity, but the simplified version of ADMM-EM requires the least number of iterations.

Figure 5 differs from Figure 4 because the former fixes the number of inner iterations and the latter fixes the number of outer iterations. This figure also presents a comparison of MAE curves and cost function. As shown, ADMM-EM always presents the best result, ADMM-CG provides the next

best result, and ADMM-PL shows the worst result; however, all of them will be consistent after sufficient inner iterations. We can also observe that ADMM-EM's result does not actually depend on the number of inner iterations, which demonstrates a clear advantage of the simplified version over the other two.

The above experiments provide a comparison while fixing the number of outer or inner iterations. In practical applications, we generally do not require this many iterations to obtain an acceptable image. In Table 2, (26) will be used as a stopping condition for both cases; then, we obtain the acceptable images with  $\epsilon = 10^{-8}$ . As shown, when fixing the number of inner iterations, the greedy ADMM-EM requires a lower number of outer iterations, whereas the simplified ADMM-EM requires more outer iterations. When fixing the outer iteration count, ADMM-EM still requires the lowest number of inner iterations, while ADMM-PL requires the most.

Table 3 also presents a comparison of algorithmic performance; however, the number of inner and outer iterations use those suggested by Table 2 to meet the stop rule  $\epsilon = 10^{-8}$ . ADMM-EM-68-11 is also included in the comparison. All presets present comparable evaluation parameters, but the simplified ADMM-EM requires very little running time and computational complexity. We observe that either the greedy or simplified version of ADMM-EM still provides the best results. In addition, the greedy version of ADMM-PL requires the largest amount of time and computational cost. The greedy versions of ADMM-EM and ADMM-CG have a similar computational cost, but the simplified version of ADMM-EM requires less computational cost. In fact, ADMM-EM-256-1 consumes 42.98% of the running time of ADMM-CG-71-11 and 9.37% of that of ADMM-PL-292-20, which corresponds to 32.78% and 4.38% of computation complexity, respectively.



TABLE 1: Algorithmic performance including mean absolute error (MAE), cost function (Fun.), contrast (LC and HC denote that of low and high ROI, resp.), variability (Var.), running time (Time, units of seconds), and computational complexity (Comp.), in which we use [ ] to represent the best value. All experiments are executed twenty times to obtain an average value.

ADMM-	MAE	Fun. ( $10^6$ )	LC (%)	HC (%)	Var. (%)	Time	Comp.
PL-50-10	82.40	44.55	71.74	33.85	12.56	6.64	500
CG-50-10	16.00	7.70	97.48	78.06	11.26	10.79	500
EM-50-10	[13.73]	[7.67]	98.33	[83.44]	[10.91]	6.71	500
EM-50-1	25.69	8.24	[98.54]	74.93	13.66	[1.37]	[50]
PL-100-20	42.88	12.21	89.98	48.17	15.62	25.35	2000
CG-100-20	14.26	[7.66]	98.01	83.18	10.73	33.56	2000
EM-100-20	[13.12]	[7.66]	98.15	[84.09]	[10.69]	25.27	2000
EM-100-1	18.98	7.84	[98.61]	81.17	12.27	[2.73]	[100]
PL-200-40	20.47	7.99	96.74	68.93	12.65	99.47	8000
CG-200-40	14.22	7.65	98.24	83.82	10.73	115.17	8000
EM-200-40	[13.03]	[7.64]	98.32	[84.35]	[10.68]	97.62	8000
EM-200-1	15.30	7.69	[98.63]	83.77	11.34	[5.42]	[200]
PL-300-80	14.43	7.68	97.98	81.03	11.13	295.93	24000
CG-300-80	14.20	7.65	98.24	84.30	10.69	318.08	24000
EM-300-80	[12.96]	[7.64]	98.31	[84.78]	[10.64]	288.76	24000
EM-300-1	14.23	7.68	[98.54]	84.66	11.02	[8.14]	[300]
PL-400-120	13.23	7.65	98.26	83.85	10.87	590.77	48000
CG-400-120	14.23	[7.64]	98.19	83.84	10.82	617.80	48000
EM-400-120	[13.05]	[7.64]	98.26	84.28	[10.78]	615.27	48000
EM-400-1	13.15	7.66	[98.42]	[84.30]	10.88	[10.84]	[400]

TABLE 2: Inner and outer iteration numbers to meet  $\epsilon = 10^{-8}$  in (26). For Case 1, we fix 120 inner iterations for the greedy ADMM algorithms; for Case 2, we fix 400 outer iterations.

Algorithm	Iteration number
Case 1: given inner iteration number	Outer iteration number
ADMM-PL-120	292
ADMM-CG-120	71
ADMM-EM-120	68
ADMM-EM-1	256
Case 2: given outer iteration number	Inner iteration number
ADMM-PL-400	20
ADMM-CG-400	11
ADMM-EM-400	1

4.2. *PET Clinical Data.* Real clinical brain projections, which were obtained using Positron’s mPower scanner with  $10 \mu\text{Ci}$  FDG preinjected into a patient, were also used for evaluation purposes. The acquisition lasted for 10 minutes, and 700 M counts in 61 slices were obtained. The normalization was performed by measurements obtained from a calibration scan with a  $1 \mu\text{Ci}$   $\text{Ge}^{68}$  rotating rod source. The attenuation coefficients of the attenuation media were computed from a transmission scan of a  $5 \mu\text{Ci}$   $\text{Ge}^{68}$  rod source. The raw data were 128 radial bins and 128 angles (bin size 4 mm), which are shown in Figure 6. The ADMM-type algorithms with  $(\beta, \rho) = (10^{-2}, 10^{-4})$  obtain the best reconstruction as above,

which we also use on the real clinical data. We also executed the proposed iteration numbers in Table 3 for the algorithms.

Figure 7 shows the reconstructions of the patient’s projection data. For real clinical data, it is difficult to quantitatively evaluate the algorithms. The reconstructed grid is  $128 \times 128$ , with 4 mm pixels. This figure illustrates that the two proposed algorithms produce sharper images with greater contrast than those achieved with other algorithms. Moreover, PL-292-20 and CG-71-11 do not clearly resolve the ellipse. Among them, PL-292-20 results in a seriously degraded image, where the boundaries of the different regions are obscure. The large difference is due to the lower number of iterations. These experimental results further demonstrate that ADMM-EM leads to a superior result than the others by providing a fast convergence.

From the derivation of the proposed method, we make the following observation: the EM-type X-update ensures that the cost function decreases at each iteration. However, it is difficult for the other algorithms to meet this condition, particularly for lower numbers of iterations. Thus, ADMM-EM with one inner loop can achieve a good result. Certainly, while running many inner loops, all of the algorithms will converge to the same solution; thus, they present some similar images.

## 5. Conclusion

We present a special application of ADMM to PET image reconstruction. Specifically, a new update rule is developed to iteratively update the pixel values, which exhibits desirable

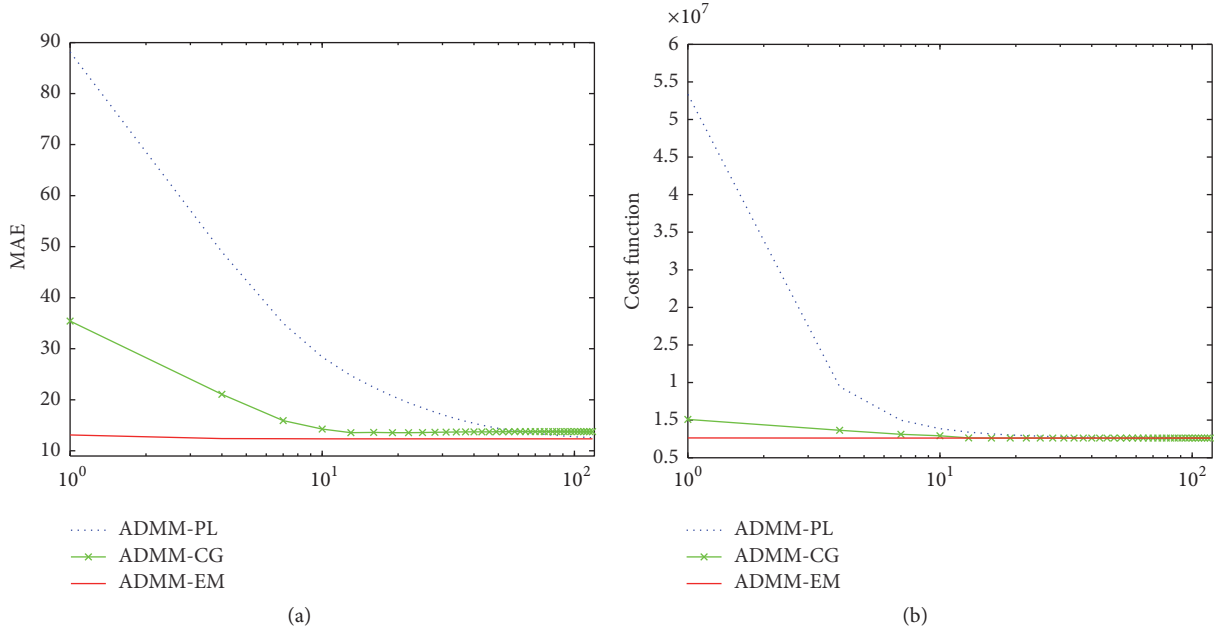


FIGURE 5: MAE and cost function versus the number of inner loops, where the number of outer loops is fixed to 400 for a sufficient convergence: (a) MAE and (b) cost function. Note that the samples of the iteration number of inner loop are collected every 3 iterations during the experiment.

TABLE 3: Comparison of algorithmic performance similar to Table 1. We still use [ ] to represent the best value, and all experiments are still executed twenty times to obtain an average value. Note that ADMM-EM-68-11 uses the same inner iteration count as ADMM-CG.

ADMM-	MAE	Fun. ( $10^6$ )	LC (%)	HC (%)	Var. (%)	Time	Comp.
PL-292-20	23.31	8.31	96.11	64.37	13.25	74.17	5840
CG-71-11	14.73	7.70	97.85	81.21	10.78	16.17	781
EM-68-11	[13.21]	[7.67]	98.42	[84.52]	[10.61]	9.94	748
EM-256-1	14.45	7.68	[98.62]	84.33	11.00	[6.95]	[256]



FIGURE 6: Real clinical brain projections with 128 bins and 128 angles that are acquired from a mPower scanner.

properties, including monotonic decrease of the cost function, self-constraining to the feasible region, and no need to impose a step size. Such properties allow us to implement a simplified version of ADMM, requiring considerably less

time and computational overhead. We provide a rigorous theoretical global convergence proof for the update step. The simulation results demonstrate that the proposed greedy algorithm provides a stabler and faster convergence with similar computational cost as ADMM-PL and ADMM-CG. The results also indicate that the proposed simplified algorithm obtains a similar image quality while imposing lower computational costs.

For our simplified algorithm, a theoretical convergence proof cannot be provided; rather, we use the experimental method to demonstrate the convergence. Proving the convergence for the simplified algorithm will be the focus of future work.

## Appendix

In theory, the Karush-Kuhn-Tucker (KKT) point will be a global solution if the cost function is convex. It is easy to see the convexity of  $\Phi(X)$ . By Theorem 2.19 in [43], the KKT conditions of (10) are as follows:

$$\frac{\partial \Phi(X)}{\partial X_j} = 0 \quad \text{if } X_j > 0 \quad (\text{A.1})$$

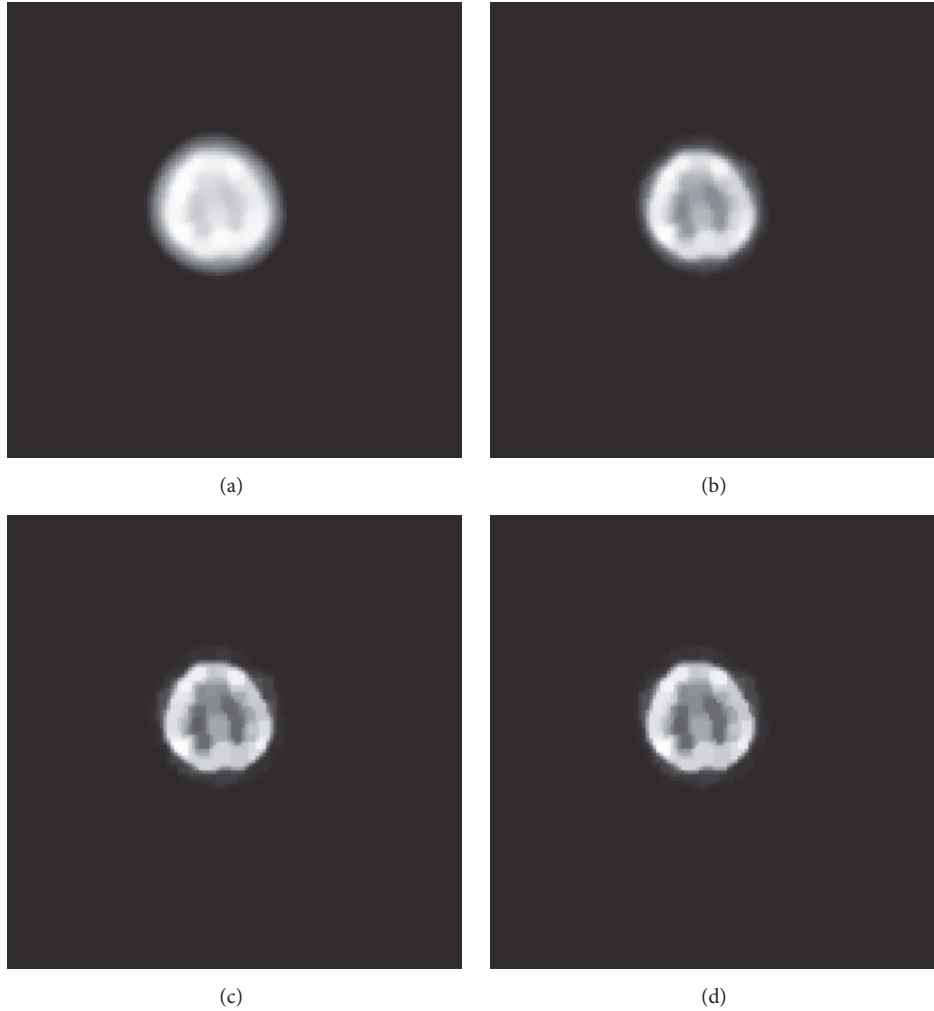


FIGURE 7: The reconstructed results of real clinical brain PET data are, respectively, obtained by (a) PL-292-20, (b) CG-71-11, (c) EM-68-11, and (d) EM-256-1. Every result is scaled according to its own minimum and maximum.

$$\frac{\partial \Phi(X)}{\partial X_j} \geq 0 \quad \text{if } X_j = 0. \quad (\text{A.2})$$

In the following, we use a number of reasonable assumptions.

*Assumption A.1.* For the iteration sequence  $\{X^t\}$ , we assume that

- (1) the algorithm starts from a positive image;
- (2)  $(P^T \Sigma P)_{jj} > 0$  for all  $j$ ;
- (3)  $\Phi$  is a strictly convex function.

The first assumption forces the iterations to be positive, but the limit may be zero. The second condition is reasonable because  $(P^T \Sigma P)_{jj} = 0$  suggests that  $P_{ij} = 0$  for any  $i$ . Thus, the equation  $Y = PX$  is irrelevant to  $X_j$ , and then  $X_j$  in  $X$  is removable. For the third requirement, one of the main reasons to add the regularization is to enforce the convexity.

First, we provide four useful lemmas.

**Lemma A.2.** *The set of accumulation points of a bounded sequence  $\{Z^t\}$  with  $\{\|Z^{t+1} - Z^t\|\} \rightarrow 0$  is connected and compact.*

*Proof.* See Theorem 28.1, Ostrowski [44].  $\square$

**Lemma A.3.** *The iteration sequence  $\{X^t\}$  is bounded.*

*Proof.* We have  $\{X^t\} \subset \{X : \Phi(X) \leq \Phi(X^0)\} \subset \{X : F(X) \leq \Phi(X^0)\}$ ; therefore, if  $F$  has bounded level sets,  $\{X^t\}$  is bounded. The boundedness of the level sets of  $F$  can be proven by contradiction. Otherwise, for a given real value  $\widehat{Z}$ , there must exist a sequence  $\{Z^t\}$  that satisfies  $F(Z^t) \leq \widehat{Z}$  and an element  $\{Z_j^t\}$  that approaches  $+\infty$ . Considering that  $P_{ij} \geq 0$  and  $\sum_{i=1}^N P_{ij}^2 > 0$  (Assumption A.1) for all  $j$ , then it implies that  $F(Z^t) \rightarrow +\infty$ , which is a contradiction.  $\square$

**Lemma A.4.** *The sequence  $\{\|X^t - X^{t+1}\|\} \rightarrow 0$ . Furthermore, if a subsequence  $\{X^{t_s}\} \rightarrow X^*$ , then  $\{X^{t_s+1}\} \rightarrow X^*$  as well.*

*Proof.* (1) Take into account that

$$\begin{aligned} & \nabla_{jj}^2 \phi(X | X^t) \\ &= 2 \frac{[P^T \Sigma (PX^t + S)]_j}{X_j^t} \\ &+ 2\rho \left[ \frac{[\bar{R}^T (\bar{R}X^t + \bar{C})]_j}{X_j^t} + \frac{[\hat{R}^T (\hat{R}X^t + \hat{C})]_j}{X_j^t} \right] \quad (\text{A.3}) \\ &\geq 2 \frac{(P^T \Sigma PX^t)_j}{X_j^t} \geq 2 (P^T \Sigma P)_{jj}. \end{aligned}$$

Let  $\gamma = \min_j \{(P^T \Sigma P)_{jj}\}$ . Since  $\nabla \phi(X^{t+1} | X^t) = 0$ , then

$$\begin{aligned} \Phi(X^t) - \Phi(X^{t+1}) &\geq \phi(X^t | X^t) - \phi(X^{t+1} | X^t) \\ &= \frac{1}{2} (X^t - X^{t+1})^T \nabla^2 \phi(X^{t+1} | X^t) (X^t - X^{t+1}) \quad (\text{A.4}) \\ &\geq \gamma \|X^t - X^{t+1}\|^2. \end{aligned}$$

Because  $\{\Phi(X^t)\}$  monotonically decreases and it is bounded from below, then  $\{\Phi(X^t) - \Phi(X^{t+1})\} \rightarrow 0$ ; thus,  $\{\|X^t - X^{t+1}\|\} \rightarrow 0$ .

(2) By contradiction, if  $\{X^{t_s+1}\}$  diverges, then it must have a convergent subsequence  $\{X^{t_{ss}+1}\} \rightarrow X^{**} \neq X^*$  because of the boundedness by Lemma A.3. Let  $\epsilon_0 = \|X^* - X^{**}\| > 0$ . Consider the two convergent subsequences  $\{X^{t_{ss}}\}$  and  $\{X^{t_{ss}+1}\}$ ; then, there must be a positive integer  $T$  to make  $\|X^{t_{ss}} - X^*\| < \epsilon_0/4$  and  $\|X^{t_{ss}+1} - X^{**}\| < \epsilon_0/4$  when  $t_{s_s} > T$ . By the triangle inequality, we can obtain the contradictive result to  $\{\|X^t - X^{t+1}\|\} \rightarrow 0$  as

$$\begin{aligned} & \|X^{t_{ss}} - X^{t_{ss}+1}\| + \|X^{t_{ss}} - X^*\| + \|X^{t_{ss}+1} - X^{**}\| \\ &\geq \|X^* - X^{**}\| \implies \quad (\text{A.5}) \\ &\|X^{t_{ss}} - X^{t_{ss}+1}\| > \frac{\epsilon_0}{2}. \end{aligned}$$

□

**Lemma A.5.** *At each iteration, one knows that*

$$\begin{aligned} X_j^{t+1} &= X_j^t - \alpha_j [\nabla \Phi(X^t)]_j, \\ &\text{where } \alpha_j = \frac{X_j^t}{2(A_3)_j + 2\rho(A_4)_j}. \quad (\text{A.6}) \end{aligned}$$

The derivation process is omitted due to its simplicity. By the three theorems below, the global convergence will be proven.

**Theorem A.6.** *Let  $\{X^{t_s}\} \rightarrow X^*$  be any convergent subsequence; then  $X^*$  meets the first KKT condition (A.1).*

*Proof.* When  $X_j^* > 0$ , by (21), it can be verified that

$$\begin{aligned} \left. \frac{\partial f(X | X^*)}{\partial X_j} \right|_{X^*} &= 2 [P^T (PX^* - Y + S)]_j \\ \left. \frac{\partial \bar{g}(X | X^*)}{\partial X_j} \right|_{X^*} &= 2\bar{R}^T (\bar{R}X^* + \bar{C}) \\ &\quad - 2\bar{R}^T (\hat{R}X^* + \hat{C}) \quad (\text{A.7}) \\ \left. \frac{\partial \hat{g}(X | X^*)}{\partial X_j} \right|_{X^*} &= 2\hat{R}^T (\hat{R}X^* + \hat{C}) \\ &\quad - 2\hat{R}^T (\bar{R}X^* + \bar{C}). \end{aligned}$$

Then  $\partial \phi(X | X^*) / \partial X_j |_{X^*} = \partial \Phi(X^*) / \partial X_j$ .

We consider that  $\partial \phi(X | X^{t_s}) / \partial X_j |_{X^{t_s+1}} = 0$ . Because  $\{X^{t_s}\} \rightarrow X^*$  and  $\{X^{t_s+1}\} \rightarrow X^*$  (Lemma A.4), then

$$\left. \frac{\partial \phi(X | X^*)}{\partial X_j} \right|_{X^*} = \lim_{t_s \rightarrow +\infty} \left. \frac{\partial \phi(X | X^{t_s})}{\partial X_j} \right|_{X^{t_s+1}} = 0. \quad (\text{A.8})$$

□

**Theorem A.7.** *The entire sequence  $\{X^t\}$  converges.*

*Proof.* According to Lemmas A.2, A.3, and A.4, the set of accumulation points of  $\{X^t\}$  is connected and compact. If we can prove that the number of accumulation points is finite, then the desired result follows because a finite set can be connected only if it consists of a single point [38].

To prove the existence of a finite number of accumulation points, we consider any accumulation point  $X^*$ . Given an integer set  $\Omega = \{1, 2, \dots, T\}$ , where  $T$  is the total number of components of  $X$ , then  $\Omega^* = \{j : X_j^* = 0\}$  is a subset of  $\Omega$ . Let  $\Phi_{\Omega^*}$  be the restrictions of  $\Phi$  to the set  $\{X : X_j = 0, j \in \Omega^*\}$ , which is a strictly convex function of the reduced variables. It follows that  $\Phi_{\Omega^*}$  has a unique minimum (Theorem A.6:  $\partial \Phi(X^*) / \partial X_j = 0$  if  $X_j^* > 0$ ). It means that an accumulation point must correspond to a subset of  $\Omega$ . The number of subsets of  $\Omega$  is finite; thus, the number of accumulation points is also finite. □

In Theorem A.6, we prove that every accumulation point meets the first KKT condition, by which the full sequence convergence is provided in Theorem A.7. Naturally, the limit of  $\{X^t\}$  satisfies the first KKT condition. In the following, we will show that the second KKT condition is also satisfied.

**Theorem A.8.** *The limit  $X^*$  of  $\{X^t\}$  meets the second KKT condition (A.2).*

*Proof.* When  $X_j^* = 0$ , by contradiction, we assume that there is  $X_j^* = 0$  satisfying  $[\nabla \Phi(X^*)]_j < 0$ . Since  $\{X^t\} \rightarrow X^*$ , there

exists  $\epsilon < 0$  and a positive integer  $\mathcal{T}$  such that  $[\nabla\Phi(X^t)]_j < \epsilon$  for  $t > \mathcal{T}$ ; then

$$X_j^{t+1} - X_j^t = -\alpha_j [\nabla\Phi(X^t)]_j > -\alpha_j \epsilon > 0 \quad (\text{A.9})$$

(by Lemma A.5).

Thus, we can obtain that  $X_j^{t+1} > X_j^t$ , which is a contradiction to  $\{X_j^t\} \rightarrow 0$ .  $\square$

## Competing Interests

The authors declare that they have no competing interests.

## Acknowledgments

This work was supported by the National Natural Science Foundations of China (61302013 and 61372014), Science and Technology Plan of Liaoning Province of China (2014305001), and Fundamental Research Funds for the Central Universities of China (N141008001).

## References

- [1] J. A. Fessler, "Penalized weighted least-squares image reconstruction for positron emission tomography," *IEEE Transactions on Medical Imaging*, vol. 13, no. 2, pp. 290–300, 1994.
- [2] J. Zhou, J.-L. Coatrieux, A. Bousse, H. Shu, and L. Luo, "A Bayesian MAP-EM algorithm for PET image reconstruction using wavelet transform," *IEEE Transactions on Nuclear Science*, vol. 54, no. 5, pp. 1660–1669, 2007.
- [3] V. Y. Panin, G. L. Zeng, and G. T. Gullberg, "Total variation regulated EM algorithm," *IEEE Transactions on Nuclear Science*, vol. 46, no. 6, pp. 2202–2210, 1999.
- [4] O. Demirkaya, "Anisotropic diffusion filtering of PET attenuation data to improve emission images," *Physics in Medicine and Biology*, vol. 47, no. 20, pp. N271–N278, 2002.
- [5] C. Riddell, H. Benali, and I. Buvat, "Diffusion regularization for iterative reconstruction in emission tomography," *IEEE Transactions on Nuclear Science*, vol. 51, no. 3, pp. 712–718, 2004.
- [6] S. Ramani and J. A. Fessler, "A splitting-based iterative algorithm for accelerated statistical X-ray CT reconstruction," *IEEE Transactions on Medical Imaging*, vol. 31, no. 3, pp. 677–688, 2012.
- [7] G. Wang and J. Qi, "Penalized likelihood PET image reconstruction using patch-based edge-preserving regularization," *IEEE Transactions on Medical Imaging*, vol. 31, no. 12, pp. 2194–2204, 2012.
- [8] G. Wang and J. Qi, "Edge-preserving PET image reconstruction using trust optimization transfer," *IEEE Transactions on Medical Imaging*, vol. 34, no. 4, pp. 930–939, 2015.
- [9] A. N. Tikhonov, A. V. Goncharsky, V. V. Stepanov, and A. G. Yagola, *Numerical Methods for the Solution of Ill-Posed Problems*, vol. 328 of *Mathematics and Its Applications*, Kluwer Academic Publishers, Boston, Mass, USA, 1995.
- [10] A. N. Tikhonov and V. Y. Arsenin, *Solution of Ill-Posed Problems*, Winston & Sons, Washington, DC, USA, 1997.
- [11] Y. Teng and T. Zhang, "Generalized EM-type reconstruction algorithms for emission tomography," *IEEE Transactions on Medical Imaging*, vol. 31, no. 9, pp. 1724–1733, 2012.
- [12] J. Verhaeghe, D. Van De Ville, I. Khalidov, M. Unser, Y. D'Asseler, and I. Lemahieu, "Reconstruction of dynamic PET data using spatio-temporal wavelet  $L_1$  regularization," in *Proceedings of the 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBS '07)*, pp. 6540–6543, August 2007.
- [13] C. Wang, Z. Hu, P. Shi, and H. Liu, "Low dose PET reconstruction with total variation regularization," in *Proceedings of the 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC '14)*, pp. 1917–1920, Chicago, Ill, USA, August 2014.
- [14] Y. Choi, J.-Y. Koo, and N.-Y. Lee, "Image reconstruction using the wavelet transform for positron emission tomography," *IEEE Transactions on Medical Imaging*, vol. 20, no. 11, pp. 1188–1193, 2001.
- [15] D. J. Lingenfelter, J. A. Fessler, and Z. He, "Sparsity regularization for image reconstruction with poisson data," in *Computational Imaging VII*, vol. 7246 of *Proceedings of SPIE*, San Jose, Calif, USA, January 2009.
- [16] H. Yi, D. Chen, W. Li et al., "Reconstruction algorithms based on  $L_1$ -norm and  $L_2$ -norm for two imaging models of fluorescence molecular tomography: a comparative study," *Journal of Biomedical Optics*, vol. 18, no. 5, Article ID 056013, 2013.
- [17] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM Journal on Imaging Sciences*, vol. 2, no. 1, pp. 183–202, 2009.
- [18] W. Yin, S. Osher, and D. Goldfarb, "Bregman iterative algorithms for  $L_1$ -minimization with applications to compressed sensing," *SIAM Journal on Imaging Sciences*, vol. 1, no. 1, pp. 143–168, 2008.
- [19] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2011.
- [20] J. F. C. Mota, J. M. F. Xavier, P. M. Q. Aguiar, and M. Püschel, "D-ADMM: a communication-efficient distributed algorithm for separable optimization," *IEEE Transactions on Signal Processing*, vol. 61, no. 10, pp. 2718–2723, 2013.
- [21] G. H. Golub, P. C. Hansen, and D. P. O'Leary, "Tikhonov regularization and total least squares," *SIAM Journal on Matrix Analysis and Applications*, vol. 21, no. 1, pp. 185–194, 1999.
- [22] P. D. Hale and A. M. Dietstrey, "Waveform metrology and a quantitative study of regularized deconvolution," in *Proceedings of the IEEE International Instrumentation and Measurement Technology Conference (I2MTC '10)*, pp. 386–391, IEEE, Austin, Tex, USA, May 2010.
- [23] M. Piana and M. Bertero, "Projected landweber method and preconditioning," *Inverse Problems*, vol. 13, no. 2, pp. 441–463, 1997.
- [24] B. S. He and X. M. Yuan, "Block-wise alternating direction method of multipliers for multiple-block convex programming and beyond," <http://www.optimization-online.org>.
- [25] S. Y. Chun, Y. K. Dewaraja, and J. A. Fessler, "Alternating direction method of multiplier for tomography with nonlocal regularizers," *IEEE Transactions on Medical Imaging*, vol. 33, no. 10, pp. 1960–1968, 2014.
- [26] H. H. Wang, Y. Gao, Y. H. Shi, and R. L. Wang, "Group-based alternating direction method of multipliers for distributed linear classification," *IEEE Transactions on Cybernetics*, vol. 46, pp. 1–15, 2016.
- [27] S. You and Q. Y. Peng, "A non-convex alternating direction method of multipliers heuristic for optimal power flow," in



- Proceedings of the IEEE International Conference on Smart Grid Communications (SmartGridComm '14)*, pp. 788–793, Venice, Italy, November 2014.
- [28] L. A. Shepp and Y. Vardi, “Maximum likelihood reconstruction for emission tomography,” *IEEE Transactions on Medical Imaging*, vol. 1, no. 2, pp. 113–122, 1982.
- [29] Y. Vardi, L. A. Shepp, and L. A. Kaufman, “Statistical model for positron emission tomography,” *Journal of the American Statistical Association*, vol. 80, no. 389, pp. 8–20, 1985.
- [30] K. Lange, D. R. Hunter, and I. Yang, “Optimization transfer using surrogate objective functions,” *Journal of Computational and Graphical Statistics*, vol. 9, no. 1, pp. 1–20, 2000.
- [31] D. D. Lee and H. S. Seung, *Algorithms for Non-negative Matrix Factorization*, vol. 13 of *Advances in Neural Information Processing Systems*, 2001.
- [32] Y. Lin and D. D. Lee, “Bayesian regularization and nonnegative deconvolution for room impulse response estimation,” *IEEE Transactions on Signal Processing*, vol. 54, no. 3, pp. 839–847, 2006.
- [33] A. Cichockia, H. Lee, Y. D. Kimb et al., “Non-negative matrix factorization with  $\alpha$ -divergence,” *Pattern Recognition Letters*, vol. 29, no. 9, pp. 1433–1440, 2008.
- [34] A. R. De Pierro, “On the relation between the ISRA and the EM algorithm for positron emission tomography,” *IEEE Transactions on Medical Imaging*, vol. 12, no. 2, pp. 328–333, 1993.
- [35] A. R. De Pierro, “A modified expectation maximization algorithm for penalized likelihood estimation in emission tomography,” *IEEE Transactions on Medical Imaging*, vol. 14, no. 1, pp. 132–137, 1995.
- [36] A. R. De Pierro, “On the convergence of an EM-type algorithm for penalized likelihood estimation in emission tomography,” *IEEE Transactions on Medical Imaging*, vol. 14, no. 4, pp. 762–765, 1995.
- [37] C. F. Wu, “On the convergence properties of the EM algorithm,” *The Annals of Statistics*, vol. 11, no. 1, pp. 95–103, 1983.
- [38] 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, 1984.
- [39] Y. Teng, Y. Zhang, H. Li, and Y. Kang, “A convergent non-negative deconvolution algorithm with Tikhonov regularization,” *Inverse Problems*, vol. 31, no. 3, Article ID 035002, 2015.
- [40] G. H. Golub and C. F. Van Loan, *Matrix Computations*, Johns Hopkins Studies in the Mathematical Sciences, Johns Hopkins University Press, Baltimore, Md, USA, 3rd edition, 1996.
- [41] <http://web.stanford.edu/group/SOL/index.html>.
- [42] National Electrical Manufacturers Association, *Performance Measurements of Positron Emission Tomographs*, NEMA Standard Publication NU 2-2007, National Electrical Manufacturers Association, Arlington, Va, USA, 2007.
- [43] W. I. Zangwill, *Nonlinear programming: a unified approach*, Prentice-Hall, New York, NY, USA, 1969.
- [44] A. P. Ostrowski, *Solution of Equations in Euclidean and Banach Spaces*, Academic Press, New York, NY, USA, 1973.



**Hindawi**  
Submit your manuscripts at  
<http://www.hindawi.com>

