A weighted stochastic gradient descent algorithm for image reconstruction in 3D computed tomography

Davood Karimi[1], Rabab Ward[1] , and Nancy Ford[2]

[1] University of British Columbia/Department of Electrical and Computer Engineering, Vancouver, Canada
[2] University of British Columbia /Faculty of Dentistry, Vancouver, Canada

Abstract— **We present and evaluate an algorithm for image reconstruction from a small number of projections in 3D x-ray computed tomography (CT). The proposed algorithm is similar to the class of projected gradient methods. Because each iteration of these algorithms for large 3D CT reconstruction is very computationally demanding, our goal is to devise an algorithm with fast convergence. To achieve this goal, in the proposed algorithm the gradient descent for reducing the measurement misfit term is carried out using a stochastic gradient descent iteration and the gradient directions are weighted using weights suggested by parallel coordinate descent. To further improve the speed of the algorithm, at each iteration we minimize the cost function on the subspace spanned by the direction of the current projected gradient and several previous update directions. We apply the proposed algorithm on simulated and real cone-beam projections and compare it with a well-known accelerated projected gradient algorithm, Monotone Fast Iterative Shrinkage-Thresholding Algorithm (MFISTA). Evaluations show that the rate of convergence of the proposed algorithm is superior to that of MFISTA.**

Keywords— **Computed tomography, stochastic gradient descent, parallel coordinate descent, projected gradient.**

I. Introduction

Statistical and iterative reconstruction methods for x-ray computed tomography (CT) have received renewed interest in recent years. The majority of algorithms proposed for 3D CT in recent years are based on the class of projected gradient methods. Each iteration of these algorithms involves a gradient step to reduce the measurement misfit term followed by a proximal operator for the regularization term, which is usually the total variation. Accelerated versions of these algorithms, such as FISTA, Nesterov's method, and ADMM have also been proposed for 3D CT [1-3].

In this study, we propose a new algorithm that is different from the basic gradient projection scheme in several ways. For reducing the measurement misfit term, we suggest a parallel coordinate descent update that will lead to a weighted gradient descent step. To improve the convergence speed of the algorithm, instead of performing full gradient descent steps, we perform stochastic gradient descent steps with diminishing step size. After applying the proximal operator for the TV regularization, this can be used as the new estimate of the image. However, in the spirit of methods such as the method of conjugate gradients, we will use this direction along with the directions of several previous image updates to define a subspace over which the cost function is approximately minimized in each iteration. We apply the proposed algorithm on simulated and real CT data and compare it with monotone version of FISTA (MFISTA) [1].

II. Materials and Methods

We denote the unknown image by $x \in \mathbb{R}^n$ and the projection measurements by $y \in \mathbb{R}^m$ and the system matrix with A. Our goal is to recover an estimate of the unknown image by solving the following unconstrained problem:

$$\hat{x} = \underset{x}{\operatorname{argmin}} \frac{1}{2} \|Ax - y\|_2^2 + \lambda\, TV(x) \qquad (1)$$

where the first term is the squared norm of measurement misfit and the second term, the total variation (TV) of the image, reflects the prior assumption that the unknown image is piece-wise constant.

Assuming that x^k is our current estimate, if we want to minimize the measurement misfit term with respect to its i^{th} coordinate, the exact solution will be:

$$x^{k+1}[i] = x^k[i] + \frac{A_i^T (y - A\,x^k)}{\|A_i\|_2^2} \qquad (2)$$

where A_i denotes the i^{th} column of A. While a straight-forward implementation of this algorithm is possible in theory, it will be impractical for 3D CT because of the very large size of the problem and because fast forward and back-projection algorithms process all measurements in one projection view at once. Therefore, we suggest the following parallel coordinate descent iteration:

$$x^{k+1} = \Psi_\lambda\left(x^k + W^{-1} A^T (y - Ax)\right) \qquad (3)$$

where $W \in \mathbb{R}^n$ is a diagonal matrix whose diagonal elements are norms of the columns of A and $\Psi_\lambda(.)$ is the proximal operator for TV, i.e.:

$$\Psi_\lambda(x) = \underset{u}{\operatorname{argmin}} \left(\frac{1}{2}\|x - u\|_2^2 + \lambda\, TV(u)\right) \qquad (4)$$

The inner update in (3) (i.e. the update prior to the application of the proximal operator) is in the form of a weighted gradient descent. The algorithm can be made significantly faster by replacing this step with a stochastic gradient descent

D.A. Jaffray (ed.), *World Congress on Medical Physics and Biomedical Engineering, June 7-12, 2015, Toronto, Canada*,
IFMBE Proceedings 51, DOI: 10.1007/978-3-319-19387-8_18

algorithm [5,5]. To explain this idea, let us note that the measurement misfit term can be written as:

$$F(x) = \frac{1}{2}\|Ax - y\|_2^2 = \sum_{i=1}^{n} \frac{1}{2}\|A_i x - y_i\|_2^2 = \sum_{i=1}^{n} f_i(x)$$

where n is total number of projection views, y_i is the vector of measurements in the i^{th} projection, and A_i is the submatrix of A containing rows that correspond to the i^{th} projection. This form of a cost function is very conducive to stochastic gradient descent. A full gradient descent iteration for this cost function will have this form:

$$x^{k+1} = x^k + \gamma_k A^T(y - Ax)$$

which is the same as the inner update in (3) save the multiplication by the diagonal matrix W. A stochastic gradient descent step will be as follows:

$$x^{k+1} = x^k + \gamma_k A_{i_k}^T(y_{i_k} - A_{i_k}x)$$

where i_k is a randomly selected index from among the set $\{1, \dots, n\}$ and γ_k is the step size.

Therefore, at each iteration of the algorithm, instead of performing a full gradient step, we perform n stochastic gradient descent steps, where n is the number of projection views. The order of projection views is chosen randomly in each iteration and each projection view is used exactly once in each iteration. Stochastic gradient descents usually exhibit fast convergence in the initial iterations but their convergence rate deteriorates as the algorithm makes progress [6]. This is because the direction of a stochastic gradient is equal to the direction of the true (full) gradient only in expectation and the variance can be quite high. Therefore, it is common to use diminishing step sizes. We use a rule of the form $\gamma_k = \gamma_0/(1 + \beta\gamma_0 k)$, where γ_0 is the initial step size, k is an iteration number, and β is a decay parameter [6]. We used a value of $\beta = 10$ which we found empirically. As for the initial value, γ_0, we used different values for different projection views. Specifically, γ_{0_i} was selected to be inversely proportional to largest eigenvalue of the corresponding submatrix A_i, which is the Lipschitz constant of the gradient of the measurement misfit term associated with the corresponding projection view (i.e., $f_i(x)$).

The iteration in (3) is in the form of forward-backward splitting algorithms. These algorithms are known to have a slow convergence rate. To improve their speed, several algorithms have been proposed. In most of these algorithms (e.g. [1,3]), the speedup is achieved by exploiting the directions of previous updates. In other words, x^{k+1} is computed not only based on x^k, but also x^{k-1}. In sequential subspace optimization, proposed in [7], directions of several previous updates are exploited and superior convergence rates are reported. This idea is very similar in essence to the method of conjugate gradients applied to quadratic functions. Following this idea, at every iteration of the algorithm we minimize the cost function in (1) over the subspace spanned by the direction suggested by (3) as well as the directions of K previous updates. Formally:

$$x^{k+1} = x^k + \sum_{i=1}^{K+1} \hat{\alpha}_i^k d_i^k$$

where

$$\begin{cases} d_1^k = x^k - \Psi_\lambda\left(x^k + W^{-1}A^T(y - Ax)\right) \\ d_i^k = x^{k-i+1} - x^{k-i} \qquad i = 2{:}\,K+1 \end{cases}$$

and the coefficients $\hat{\alpha}_i^k$ are chosen to minimize the cost function:

$$\hat{\alpha}_i^k = \underset{\alpha}{\operatorname{argmin}} \frac{1}{2}\left\|A\left(x^k + \sum_{i=1}^{K+1}\alpha_i^k d_i^k\right) - y\right\|_2^2 + \lambda\, TV\left(x^k + \sum_{i=1}^{K+1}\alpha_i^k d_i^k\right)$$

This minimization is not easy because TV(x) is non-smooth. The approach followed in this study was to sequentially minimize the cost function with respect to individual α_is using golden section search. In our experience, the values of α_is did not change drastically between successive iterations of the algorithm. Therefore, α_is can be initialized to the values found in the previous iteration and a search performed in a small neighborhood around these values. This way, a single sweep through α_is, starting with α_1, was enough to find $\hat{\alpha}_i$s to good accuracy. We used $K = 3$ in this study.

In order to evaluate the proposed algorithm, we applied it on a set of simulated data and two sets of real cone-beam projections. The simulated data consisted of 30 projections with uniform angular spacing between 0° and 180° from a 3D Shepp-Logan phantom. A phantom of size $256 \times 256 \times 256$ voxels and a flat detector of 360×360 pixels were considered. The incident photon count was considered to be $N_0 = 10^4$. The real cone-beam sinograms were acquired using a Gamma Medica eXplore CT 120 micro--CT scanner. The imaged objects included a phantom, designed in [8] for comprehensive evaluation of the performance of micro-CT scanners, and a dead mouse. The sinograms were of size 875×568. The tube voltage, tube current, and exposure times were, respectively, 80 kV, 32 mA, and 8 ms for the scan of the phantom, and 50 kV, 63 mA, and 100 ms for the mouse scan. The size of the reconstructed image was $880 \times 880 \times 650$ voxels, with isotropic voxels of $0.1 \times 0.1 \times 0.1$ mm^3 in size.

The scan of the phantom consisted of 220 projections at 0.877°, whereas the scan of the mouse consisted of 260 projections at 0.742° intervals. We used the full set of scans to reconstruct a high-quality image by using a filtered-backprojection followed by 15 iterations of the SART algorithm. We will use these high-quality image as the reference image or "true" image for the physical phantom and the mouse. To evaluate the proposed algorithm, we used 110 projections of

the phantom and 122 projections of the mouse for reconstruction with the proposed algorithm and with MFISTA.

III. Results

For the simulated data from the Shepp-Logan phantom, Fig. 1 shows the Root-mean-square of the reconstruction error (RMSE), where reconstruction error is defined as the difference between the reconstructed and true images, for the proposed algorithm and MFISTA for up to 40 iterations. Both algorithms were initialized with a filtered-backprojection reconstruction. In Fig. 2, we show a typical profile from the images of the phantom reconstructed with the proposed algorithm and MFISTA.

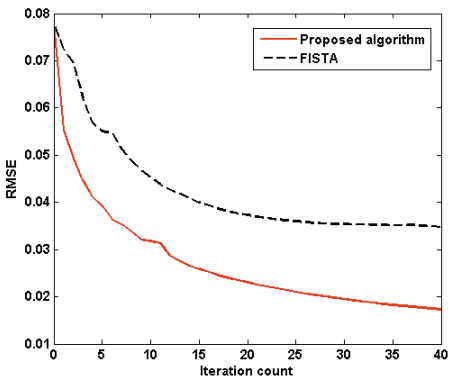

Fig. 1 Change in the RMS of the reconstruction error for the Shepp-Logan phantom reconstructed with the proposed algorithm and MFISTA.

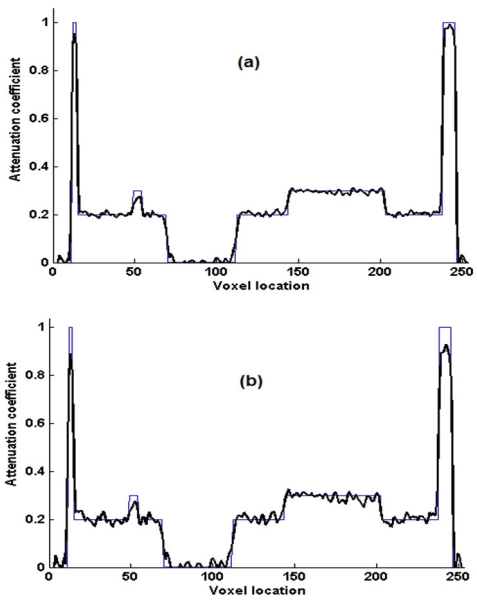

Fig. 2 Typical profiles of the Shepp-Logan phantom reconstructed using the proposed algorithm (a) and MFISTA (b).

Fig. 3 shows the change in the RMSE of the images of the physical phantom and the mouse for the proposed algorithm and MFISTA. Here, we use the high-quality reference image as the "true" image. Both algorithms were initialized with a filtered-backprojection reconstruction and the RMSE values were normalized with that of the initial estimate so that we can show the curves for the physical phantom and the mouse on the same figure.

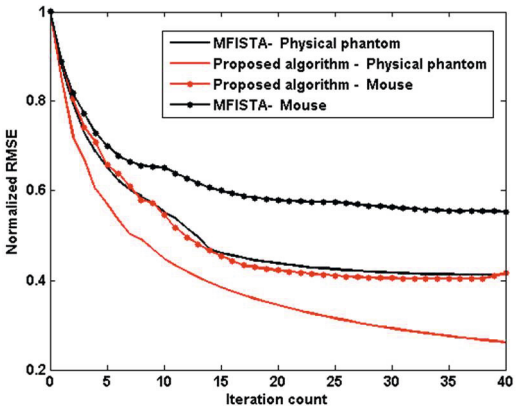

Fig. 3 Change in the normalized RMSE for reconstruction of the images of the physical phantom and the mouse from real projections with the proposed algorithm and MFISTA.

In Fig. 4 we have shown cross sections of the reconstructed phantom at the location of a resolution coil and one-dimensional profiles through it. We also used a uniform polycarbonate plate in the phantom to assess the noise level in the reconstructed images. The signal-to-noise-ratio for the images reconstructed using MFISTA and the proposed algorithm were 22.3 and 23.0 dB, respectively. Fig. 5 shows a selected regions of the images of the mouse reconstructed with the proposed algorithm and MFISTA. As expected from the RMSE plots in Fig. 3, images reconstructed with the proposed algorithm have a noticeably higher quality than those reconstructed with MFISTA.

IV. Conclusions

The proposed algorithm shows a promising performance on simulated and real data. A very important feature of the proposed algorithm is its fast convergence, particularly in the initial iterations. For the simulated data (Fig. 1) and the real data from both the physical phantom and the mouse (Fig. 3), approximately 10 iterations of the proposed algorithm reduces the RMSE to the level that is achieved in 40 iterations of MFISTA. This can be a very desirable behavior in many applications. Reconstruction of CT images from a small number of projections has attracted much attention in recent

years because of its potential for dose reduction. Even though implementation of CT reconstruction algorithms on special-purpose hardware has reduced the computational time, because of the very large size of 3D CT images, forward and back-projection operations are still very computationally demanding and reducing the number of these operations is highly desirable. Our results show that the proposed algorithm is successful in reducing the number of these operations.

A limitation of the proposed algorithm is the need to store several previous update directions and their projections. Each update direction has the size of the reconstructed image and the size of its projection is equal to the size of the projection measurements used in the reconstruction. Considering the large size of 3D CT images and their projections, the memory requirements may be problematic. The required memory grows linearly with the number, K, of previous update directions used. In our experience no additional improvements are achieved beyond K = 3 and the memory requirements for storing three update directions and their projections should not be prohibitive for most applications.

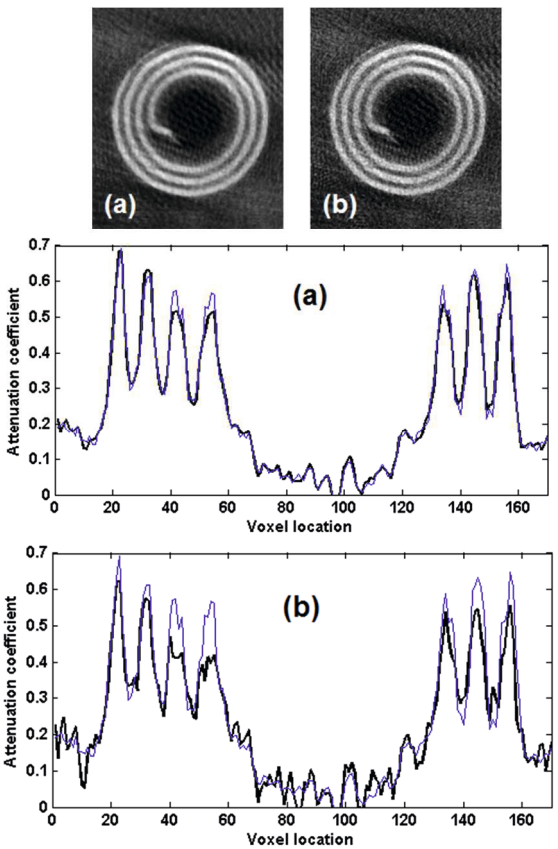

Fig. 4 Typical 2D slices and 1D profiles through a resolution coil in the image of physical phantom reconstructed with the proposed algorithm (a) and MFISTA (b). The thin blue lines in 1D profiles show the profile for the reference (i.e. "true") image.

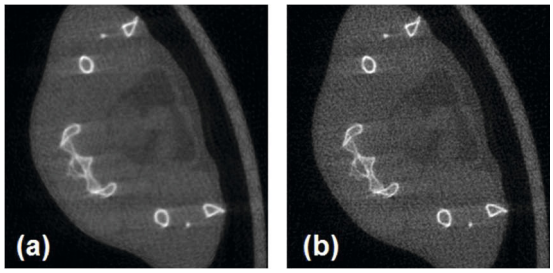

Fig. 5 Typical slices from the image of the mouse reconstructed with the proposed algorithm (a) and MFISTA (b).

Acknowledgment

Micro-CT imaging was performed in the Centre for High-Throughput Phenogenomics at the University of British Columbia, a facility supported by the Canada Foundation for Innovation, British Columbia Knowledge Development Foundation, and the UBC Faculty of Dentistry.

Conflict of Interest

The authors declare that they have no conflict of interest".

References

1. Beck A, Teboulle M (2009) A fast iterative shrinkage-thresholding algorithm for linear inverse problems. SIAM J Img Sci 2:183–202.
2. Nesterov Y (2007) Gradient methods for minimizing composite objective function. Universite Catholique de Louvain, Tech. Rep. CCIT 559.
3. Bioucas-Dias J, Figueiredo M (2007) A new twist: Two-step iterative shrinkage/thresholding algorithms for image restoration. IEEE Transactions on Image Processing 16: 2992–3004.
4. Le Roux N, Schmidt M, Bach F (2012) A stochastic gradient method with an exponential convergence rate for finite training sets, in Adv Neur Inf Prc Sys, pp. 2663-2671.
5. Defazio A, Bach F, Lacoste-Julien S (2014) Saga: A fast incremental gradient method with support for non-strongly convex composite objectives, in Adv Neur Inf Prc Sys, pp. 1646–1654.
6. Bottou L, Stochastic (2012) Gradient descent tricks, in Neural Networks: Tricks of the Trade, Lecture Notes in Computer Science, Montavon G, Orr G, Mller K, Eds. Springer Berlin Heidelberg 7700: 421–436.
7. Elad M, Matalon B, Zibulevsky M (2007) Coordinate and subspace optimization methods for linear least squares with non-quadratic regularization. Applied and Computational Harmonic Analysis 23: 346 – 367.
8. Du L, Umoh J, Nikolov H, Pollmann S, Lee T, Holdsworth D (2007) A quality assurance phantom for the performance evaluation of volumetric micro-CT systems, Phy Med Bio 52: 7087-7108.