Accelerating ordered-subsets X-ray CT image reconstruction using the linearized augmented Lagrangian framework

Hung Nien and Jeffrey A. Fessler

Department of Electrical Engineering and Computer Science
University of Michigan, Ann Arbor, MI

ABSTRACT

The augmented Lagrangian (AL) optimization method has drawn more attention recently in imaging applications due to its decomposable structure for composite cost functions and empirical fast convergence rate under weak conditions. However, for problems, e.g., X-ray computed tomography (CT) image reconstruction, where the inner least-squares problem is challenging, the AL method can be slow due to its iterative inner updates. In this paper, using a linearized AL framework, we propose an ordered-subsets (OS) accelerable linearized AL method, OS-LALM, for solving penalized weighted least-squares (PWLS) X-ray CT image reconstruction problems. To further accelerate the proposed algorithm, we also propose a deterministic downward continuation approach for fast convergence without additional parameter tuning. Experimental results show that the proposed algorithm significantly accelerates the "convergence" of X-ray CT image reconstruction with negligible overhead and exhibits excellent gradient error tolerance when using many subsets for OS acceleration.

1. INTRODUCTION

The augmented Lagrangian (AL) method (including its alternating direction variants)[1–5] is a powerful technique for solving regularized inverse problems using variable splitting. In X-ray computed tomography (CT) image reconstruction, the AL method is used to decompose the original CT problem into several easier and better-conditioned inner minimization problems so that one can accelerate convergence using appropriate preconditioners.[6,7] Experimental results showed that the acceleration is significant in 2D CT;[6] however, in 3D CT, due to different geometries, it is more difficult to construct a good preconditioner for the inner least-squares problem, and one has yet to achieve the same acceleration as in 2D CT. In comparison, the ordered-subsets (OS) algorithm[8] is a gradient method with a diagonal preconditioner/majorizer that uses somewhat conservative step sizes but is more easily applicable to different geometries. Furthermore, by grouping the projections into M ordered subsets that satisfy the "subset balance condition" and updating the image incrementally using the M subset gradients, the OS algorithm effectively runs M times as many image updates per outer iteration as the standard gradient descent method, thus leading to M times acceleration at least in early iterations. In this paper, we focus on solving regularized (weighted) least-squares problems using a linearized variant of the AL method[9–12] and show that the linearized AL method, in this case, can be intepreted as a gradient method and hence, is feasible for OS acceleration.

2. METHODS

2.1 OS-LALM: an OS-accelerable splitting-based algorithm

Consider a regularized least-squares image reconstruction problem:

$$\hat{\mathbf{x}} \in \arg \min_{\mathbf{x}} \left\{ \Psi(\mathbf{x}) \triangleq \tfrac{1}{2} \left\| \mathbf{y} - \mathbf{A}\mathbf{x} \right\|_2^2 + h(\mathbf{x}) \right\} , \tag{1}$$

where $\mathbf{A}$ is the system matrix, $\mathbf{y}$ is the noisy measurement, and h is a convex (and possibly non-smooth) regularization term. Let $\ell(\mathbf{x})$ denote the quadratic data-fitting term. We further assume that ℓ is suitable for

Hung Nien: hungnien@umich.edu, 734-353-0850

Jeffrey A. Fessler: fessler@umich.edu, 734-763-1434

OS acceleration; that is, ℓ can be decomposed into M smaller quadratic functions $\ell_1, \ldots, \ell_M$ that satisfy the "subset balance condition":[8]

$$\nabla\ell(\mathbf{x}) \approx M\nabla\ell_1(\mathbf{x}) \approx \cdots \approx M\nabla\ell_M(\mathbf{x}) . \tag{2}$$

Instead of solving (1) using the (proximal) gradient method or its OS variants, we consider solving an equivalent constrained minimization problem:[13]

$$(\hat{\mathbf{x}}, \hat{\mathbf{u}}) \in \arg\min_{\mathbf{x}, \mathbf{u}} \left\{ \tfrac{1}{2} \|\mathbf{u}\|_2^2 + h(\mathbf{x}) \right\} \text{ s.t. } \mathbf{u} = \mathbf{A}\mathbf{x} - \mathbf{y} \tag{3}$$

using the alternating direction AL method that alternatingly minimizes the scaled augmented Lagrangian:

$$\mathcal{L}_{\mathrm{A}}(\mathbf{x}, \mathbf{u}, \mathbf{d}; \rho) \triangleq \tfrac{1}{2} \|\mathbf{u}\|_2^2 + h(\mathbf{x}) + \tfrac{\rho}{2} \|\mathbf{A}\mathbf{x} - \mathbf{y} - \mathbf{u} - \mathbf{d}\|_2^2 \tag{4}$$

with respect to $\mathbf{x}$ and $\mathbf{u}$, followed by a gradient ascent of $\mathbf{d}$, yielding the following AL iterates:

$$\begin{cases} \mathbf{x}^{(k+1)} \in \arg\min_{\mathbf{x}} \left\{ h(\mathbf{x}) + \tfrac{\rho}{2} \|\mathbf{A}\mathbf{x} - \mathbf{y} - \mathbf{u}^{(k)} - \mathbf{d}^{(k)}\|_2^2 \right\} \\ \mathbf{u}^{(k+1)} \in \arg\min_{\mathbf{u}} \left\{ \tfrac{1}{2} \|\mathbf{u}\|_2^2 + \tfrac{\rho}{2} \|\mathbf{A}\mathbf{x}^{(k+1)} - \mathbf{y} - \mathbf{u} - \mathbf{d}^{(k)}\|_2^2 \right\} \\ \mathbf{d}^{(k+1)} = \mathbf{d}^{(k)} - \left(\mathbf{A}\mathbf{x}^{(k+1)} - \mathbf{y} \right) + \mathbf{u}^{(k+1)} , \end{cases} \tag{5}$$

where $\mathbf{d}$ is the scaled Lagrange multiplier of the split variable $\mathbf{u}$, and $\rho > 0$ is the corresponding AL penalty parameter. We choose this particular split (3) because $\mathbf{u}$ (a split variable for the residual) is almost the gradient of ℓ ($\nabla\ell(\mathbf{x}) = \mathbf{A}'(\mathbf{A}\mathbf{x} - \mathbf{y})$), i.e, a back-projection of the residual. Note that for any algorithm that can be accelerated using the approximation (2), all the update iterates should depend only on the gradient of ℓ, just like any gradient method!

Before continuing the derivation of our proposed algorithm, let's take a closer look at (5). Since ℓ is quadratic, the $\mathbf{u}$-update in (5) is linear and has the following simple closed-form solution:

$$\mathbf{u}^{(k+1)} = \tfrac{\rho}{\rho+1} \left(\left(\mathbf{A}\mathbf{x}^{(k+1)} - \mathbf{y} \right) - \mathbf{d}^{(k)} \right) . \tag{6}$$

Furthermore, combining (6) with the $\mathbf{d}$-update in (5) yields the identity:

$$\mathbf{u}^{(k+1)} = -\rho\mathbf{d}^{(k+1)} \tag{7}$$

if we initialize $\mathbf{d}$ as $\mathbf{d}^{(0)} = -\rho^{-1}\mathbf{u}^{(0)}$. Substituting (6) and (7) into (5) leads to the *simplified* AL iterates:

$$\begin{cases} \mathbf{x}^{(k+1)} \in \arg\min_{\mathbf{x}} \left\{ \rho^{-1}h(\mathbf{x}) + \tfrac{1}{2} \|\mathbf{A}\mathbf{x} - \mathbf{y} + \left(\rho^{-1} - 1 \right) \mathbf{u}^{(k)}\|_2^2 \right\} \\ \mathbf{u}^{(k+1)} = \tfrac{\rho}{\rho+1} \left(\mathbf{A}\mathbf{x}^{(k+1)} - \mathbf{y} \right) + \tfrac{1}{\rho+1}\mathbf{u}^{(k)} . \end{cases} \tag{8}$$

If we "solve" the inner minimization problem using the OS algorithm, it becomes the AL-OS algorithm.[13]

As mentioned in Section 1, the main problem of the AL method is the non-trivial inner update, i.e., the $\mathbf{x}$-update in (8) that involves the Hessian $\mathbf{A}'\mathbf{A}$. To overcome this problem, we propose to further "linearize" the quadratic term (i.e., the quadratic penalty of the scaled augmented Lagrangian) in the inner minimization problem with respect to $\mathbf{x}^{(k)}$ and add an additional quadratic proximal penalty, leading to a type of the linearized AL method.[9–12] Specifically, we majorize the quadratic term in (8) by

$$\tfrac{1}{2} \|\mathbf{A}\mathbf{x} - \mathbf{y} + \left(\rho^{-1} - 1 \right) \mathbf{u}^{(k)}\|_2^2 \leq C_k + \mathbf{x}'\mathbf{A}' \left(\mathbf{A}\mathbf{x}^{(k)} - \mathbf{y} + \left(\rho^{-1} - 1 \right) \mathbf{u}^{(k)} \right) + \tfrac{L}{2} \|\mathbf{x} - \mathbf{x}^{(k)}\|_2^2 , \tag{9}$$

where C_k is a constant that depends only on $\mathbf{x}^{(k)}$ and $\mathbf{u}^{(k)}$, L is the spectral radius of $\mathbf{A}'\mathbf{A}$, and the equality holds if $\mathbf{x} = \mathbf{x}^{(k)}$. The majorization (9) removes the entanglement of $\mathbf{x}$ introduced by the system matrix $\mathbf{A}$ and leads to the following simplified *linearized* AL iterates:

$$\begin{cases} \mathbf{x}^{(k+1)} \in \mathrm{prox}_{(\rho^{-1}t)h} \left(\mathbf{x}^{(k)} - t\mathbf{A}' \left(\mathbf{A}\mathbf{x}^{(k)} - \mathbf{y} + \left(\rho^{-1} - 1 \right) \mathbf{u}^{(k)} \right) \right) \\ \mathbf{u}^{(k+1)} = \tfrac{\rho}{\rho+1} \left(\mathbf{A}\mathbf{x}^{(k+1)} - \mathbf{y} \right) + \tfrac{1}{\rho+1}\mathbf{u}^{(k)} , \end{cases} \tag{10}$$

where $t \triangleq 1/L$ serves as a step size, and $\mathrm{prox}_\phi(\mathbf{z})$ denotes the proximal mapping of ϕ defined as

$$\mathrm{prox}_\phi(\mathbf{z}) \triangleq \arg\min_{\mathbf{x}} \left\{ \phi(\mathbf{x}) + \tfrac{1}{2} \left\| \mathbf{x} - \mathbf{z} \right\|_2^2 \right\} . \tag{11}$$

It is trivial to generalize L to any positive semi-definite matrix $\mathbf{G} \succeq \mathbf{A}'\mathbf{A}$, e.g., the diagonal majorizer $\mathbf{G}_{\mathrm{diag}} \triangleq \mathrm{diag}\{|\mathbf{A}|'|\mathbf{A}|\mathbf{1}\}$ used in most OS-based algorithms.[8,14,15] When $\mathbf{G} = \mathbf{A}'\mathbf{A}$, the linearized AL method reverts to the standard AL method (8). Now, the net computational complexity of updates in (10) reduces to one multiplication by $\mathbf{A}$, one multiplication by $\mathbf{A}'$, and one proximal mapping of h that usually can be solved non-iteratively or solved iteratively without using $\mathbf{A}$ or $\mathbf{A}'$. Note that the linearized AL method (also known as the *split inexact Uzawa method* in the image processing literature[16–18]) converges for any $\rho > 0$, even with inexact updates.[9–12,19] Therefore, the simplified linearized AL method (10) is a convergent algorithm for any $\rho > 0$.

To enable OS acceleration, observe that the $\mathbf{x}$-update in (10) depends only on $\nabla\ell(\mathbf{x}^{(k)}) = \mathbf{A}'(\mathbf{A}\mathbf{x}^{(k)} - \mathbf{y})$ and $\mathbf{A}'\mathbf{u}^{(k)}$, where the latter is a back-projection of the split residual. Let $\mathbf{g} \triangleq \mathbf{A}'\mathbf{u}$ denote the *split gradient* of ℓ and introduce an auxiliary variable $\mathbf{s}$. We rewrite (10) as the following *OS-accelerable* linearized AL iterates:

$$\begin{cases} \mathbf{s}^{(k+1)} = \rho \nabla\ell(\mathbf{x}^{(k)}) + (1 - \rho) \mathbf{g}^{(k)} \\ \mathbf{x}^{(k+1)} \in \mathrm{prox}_{(\rho^{-1}t)h}(\mathbf{x}^{(k)} - (\rho^{-1}t) \mathbf{s}^{(k+1)}) \\ \mathbf{g}^{(k+1)} = \frac{\rho}{\rho+1} \nabla\ell(\mathbf{x}^{(k+1)}) + \frac{1}{\rho+1} \mathbf{g}^{(k)} . \end{cases} \tag{12}$$

The net computational complexity of (12) per iteration becomes one gradient evaluation of ℓ and one proximal mapping of h. We can intepret (12) as a generalized proximal gradient descent of Ψ with step size $\rho^{-1}t$ and search direction $\mathbf{s}^{(k+1)}$, a linear average of the gradient and the split gradient of ℓ. A smaller ρ leads to a larger step size. When $\rho = 1$, (12) is simply the proximal gradient method or the iterative shrinkage/thresholding algorithm (ISTA).[20] In other words, by using the linearized AL framework, we can arbitrarily increase the step size of the proximal gradient method by decreasing ρ, thanks to the simple ρ-dependent correction in (12)!

To use OS, we replace $\nabla\ell$ with $M\nabla\ell_m$ in (12) using the approximation (2) and incrementally perform (12) for M times as one outer iteration, leading to the proposed algorithm with OS acceleration (OS-LALM):

$$\begin{cases} \mathbf{s}^{(k,m+1)} = \rho M \nabla\ell_m(\mathbf{x}^{(k,m)}) + (1 - \rho) \mathbf{g}^{(k,m)} \\ \mathbf{x}^{(k,m+1)} \in \mathrm{prox}_{(\rho^{-1}t)h}(\mathbf{x}^{(k,m)} - (\rho^{-1}t) \mathbf{s}^{(k,m+1)}) \\ \mathbf{g}^{(k,m+1)} = \frac{\rho}{\rho+1} M \nabla\ell_{m+1}(\mathbf{x}^{(k,m+1)}) + \frac{1}{\rho+1} \mathbf{g}^{(k,m)} \end{cases} \tag{13}$$

with boundary conditions $\mathbf{c}^{(k,M+1)} = \mathbf{c}^{(k+1)} = \mathbf{c}^{(k+1,1)}$ for $\mathbf{c} \in \{\mathbf{s}, \mathbf{x}, \mathbf{g}\}$ and $\ell_{M+1} = \ell_1$. Like typical OS-based algorithms, the proposed algorithm (13) is convergent when $M = 1$, i.e., (12), but is not guaranteed to be convergent for $M > 1$. When $M > 1$, updates generated by OS-based algorithms enter a "limit cycle" in which updates stop approaching the optimum. More importantly, some noise-like artifacts might be observed in the reconstructed image when using too many subsets for OS acceleration. Section 3 investigates how M affects the stabilities of different OS-based algorithms.

2.2 Deterministic downward continuation

One drawback of the AL method with a fixed AL penalty parameter ρ is the difficulty of finding the value that provides the fastest convergence. Intuitively, a smaller ρ is better because we can have a larger step size. However, when the step size is too large, one can encounter overshoots and oscillations that slow down the convergence rate at first and when we approach the optimum. Thus, an iteration-dependent ρ_k, i.e., a continuation approach, is desirable. For example, the classic continuation approach suggests increasing ρ as the algorithm proceeds so that the previous update can serve as a warm start for the following worse-conditioned but more penalized inner minimization problem.[21] In this paper, unlike classic continuation approaches, we consider a *downward continuation* approach. The intuition is that, for a fixed ρ, the step length $\left\| \mathbf{x}^{(k+1)} - \mathbf{x}^{(k)} \right\|$ is typically a decreasing sequence because the gradient norm vanishes as we approach the optimum, and an increasing sequence ρ_k (i.e., a diminishing step size) would aggravate the shrinkage of step length and slow down the convergence rate. In contrast, a decreasing sequence ρ_k can compensate the shrinkage and accelerate convergence.

Intuitively, ρ_k cannot decrease too fast; otherwise, the soaring step size might make the algorithm unstable or even divergent. For instance, ρ_k decreasing as $O(1/k)$ provides great acceleration while still guarantees convergence (for the one-subset case) in practice; however, ρ_k decreasing as $O(1/k^2)$ seems to be too aggressive and might not lead to convergence (or a small limit cycle when using OS for acceleration) although the convergence rate is very fast in early iterations. In this paper, we considered a decreasing sequence:

$$\rho_l = \begin{cases} 1 & , \text{ if } l = 0 \\ \frac{\pi}{l+1}\sqrt{1 - \left(\frac{\pi}{2l+2}\right)^2} & , \text{ otherwise}, \end{cases} \tag{14}$$

where l is a counter that starts from zero and increases by one. Let

$$\xi(k) \triangleq \left(\mathbf{g}^{(k)} - \nabla\ell\big(\mathbf{x}^{(k+1)}\big)\right)'\left(\nabla\ell\big(\mathbf{x}^{(k+1)}\big) - \nabla\ell\big(\mathbf{x}^{(k)}\big)\right) \tag{15}$$

denote a *restart indicator* for the one-subset version (12) of our proposed algorithm and reset l to zero whenever $\xi(k) > 0$. For the M-subset version (13), we simply replace the gradients with the gradient approximations and check the restart condition every inner iteration. This approach works well in practice for CT. We leave its detailed derivation and discussion in the upcoming work.[19] Note that ρ_l does not change for different $\mathbf{A}$'s. The adaptive restart condition takes care of the dependence on $\mathbf{A}$. That is why we call this approach the *deterministic* downward continuation approach.

3. RESULTS

In this section, we evaluated our proposed algorithm using the statistically weighted X-ray CT image reconstruction problem:

$$\hat{\mathbf{x}} \in \arg\min_{\mathbf{x}\in\Omega}\left\{\tfrac{1}{2}\|\mathbf{y} - \mathbf{A}\mathbf{x}\|_{\mathbf{W}}^2 + \mathsf{R}(\mathbf{x})\right\}, \tag{16}$$

where $\mathbf{A}$ is the system matrix of a CT scan, $\mathbf{y}$ is the noisy sinogram, $\mathbf{W}$ is the statistical weighting matrix, R is an edge-preserving regularizer, and Ω denotes the convex set for a box constraint (usually the non-negativity constraint) on $\mathbf{x}$. To solve (16) using the proposed algorithm, we use the following substitution:

$$\begin{cases} \mathbf{A} \leftarrow \mathbf{W}^{1/2}\mathbf{A} \\ \mathbf{y} \leftarrow \mathbf{W}^{1/2}\mathbf{y} \\ h \leftarrow \mathsf{R} + \iota_\Omega, \end{cases} \tag{17}$$

where $\iota_{\mathcal{C}}$ denotes the characteristic function of a convex set $\mathcal{C}$. Thus, the inner minimization problem in (12) and (13) becomes a *constrained denoising problem*. In our implementation, we solve the inner constrained denoising problem using n iterations of the fast iterative shrinkage/thresholding algorithm (FISTA)[22] starting from the previous update as a warm start. Throughout the experiment, we use a diagonal majorizer $\mathbf{G}_{\text{diag}} \triangleq \text{diag}\{\mathbf{A}'\mathbf{W}\mathbf{A}\mathbf{1}\}$ to majorize the quadratic penalty in the scaled augmented Lagrangian.[8] The naming conventions are as follows: **OS-SQS-M** denotes the OS algorithm[8] with M subsets; **OS-Nes83-M** denotes the OS+momentum algorithm[14] based on Nesterov's first fast gradient method[23] with M subsets; **OS-Nes05-M** denotes the OS+momentum algorithm[15] based on Nesterov's third fast gradient method[24] with M subsets; **OS-LALM-M-ρ-n** denotes the proposed algorithm (13) with M subsets, a fixed AL penalty parameter ρ, and n FISTA iterations for solving the inner constrained denoising problem; **OS-LALM-M-c-n** denotes the proposed algorithm (13) using the deterministic downward continuation approach (14) with M subsets and n FISTA iterations for solving the inner constrained denoising problem. When $n = 1$, i.e., with a single gradient descent for the constrained denoising problem, all algorithms listed above have the same computational complexity (one forward/back-projection and M gradient evaluation of the regularization term per iteration), so comparing the convergence rate as a function of iteration is fair. In this experiment, we reconstructed a $512 \times 512 \times 109$ image from a shoulder region helical CT scan, where the sinogram has size $888 \times 32 \times 7146$ and pitch 0.5, provided by GE Healthcare. We tested our proposed algorithm with two different values of M (20 and 40) and three different values of ρ (0.2, 0.1, and 0.05).
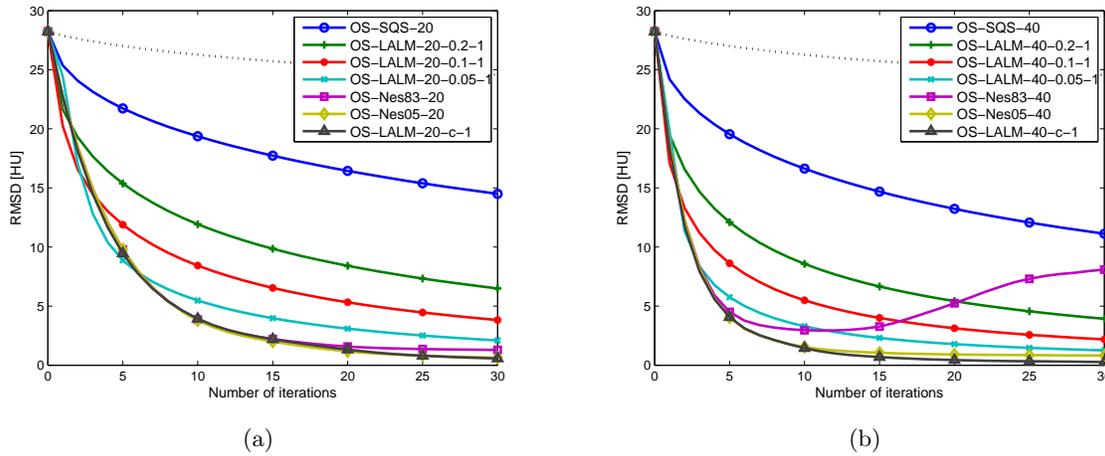
Figure 1: RMS differences between the reconstructed image $\mathbf{x}^{(k)}$ and the reference reconstruction $\mathbf{x}^\star$ as a function of iteration using OS-based algorithms with (a) 20 subsets and (b) 40 subsets, respectively. The dotted lines show the RMS differences using the OS algorithm with one subset as the baseline convergence rate.

Figure 1 shows the convergence rate curves (RMS differences between the reconstructed image $\mathbf{x}^{(k)}$ and the reference reconstruction $\mathbf{x}^\star$ as a function of iteration) of different OS-based algorithms. As can be seen in Figure 1, the proposed algorithm accelerates the convergence rate of the OS algorithm remarkably by exploiting the linearized AL framework. As mentioned in Section 2.1, a smaller ρ can provide greater acceleration due to the increased step size. However, too large step sizes can cause overshoots in early iterations. For example, the proposed algorithm with $\rho = 0.05$ shows slower convergence rate in first few iterations but decreases more rapidly later. This trade-off can be overcome by using our proposed deterministic downward continuation approach. Furthermore, comparing to OS+momentum algorithms, our proposed algorithm with deterministic downward continuation shows not only the same fast convergence rate for small number of subsets ($M = 20$), but the much better gradient error tolerance when using many subsets ($M = 40$) for OS acceleration. When the number of subsets increases, fewer views in a subset are used to approximate the full gradient of ℓ. This introduces errors to the search directions in OS-based algorithms and makes the algorithms unstable. For instance, the OS+momentum algorithm based on Nesterov's first fast gradient method with 40 subsets (the purple curve in Figure 1(b)) performs so bad that the RMS difference starts increasing after the 10th iteration. The other OS+momentum algorithm works better but still has higher RMS difference in the end. As can be seen in Figure 1, the proposed algorithm with deterministic downward continuation reaches the lowest RMS differences (lower than 1 HU) within only 30 iterations, i.e., 30 forward/back-projection pairs!

Figure 2 shows the cropped images from the central transaxial plane of the initial FBP image $\mathbf{x}^{(0)}$, the reconstructed images at the 30th iteration $\mathbf{x}^{(30)}$ using different OS-based algorithms, and the reference reconstruction $\mathbf{x}^\star$. As can be seen in Figure 2, the reconstructed images using OS+momentum algorithms and our proposed algorithm look sharp and less noisy, while other reconstructed images using the OS algorithm exhibit residual streak artifacts. To see the difference between OS+momentum algorithms and our proposed algorithm, Figure 3 shows the difference image, i.e., $\mathbf{x}^{(30)} - \mathbf{x}^\star$, for these OS-based algorithms. We can see that the difference images look more uniform and less structured for our proposed algorithm; however, the OS+momentum results exhibit high frequency structured noise and strong ripples, especially when using many subsets for OS acceleration. This demonstrates the better gradient error tolerance of our proposed algorithm when OS is used, probably due to the way we compute the search direction. As can be seen in (13), our proposed algorithm computes the search direction as an average of all previous gradient approximations using *all* subsets, while the OS+momentum algorithms compute their search directions using only the *current* subset. The average process in our proposed algorithm might mitigate the gradient error accumulation and provide better and more stable reconstructions.
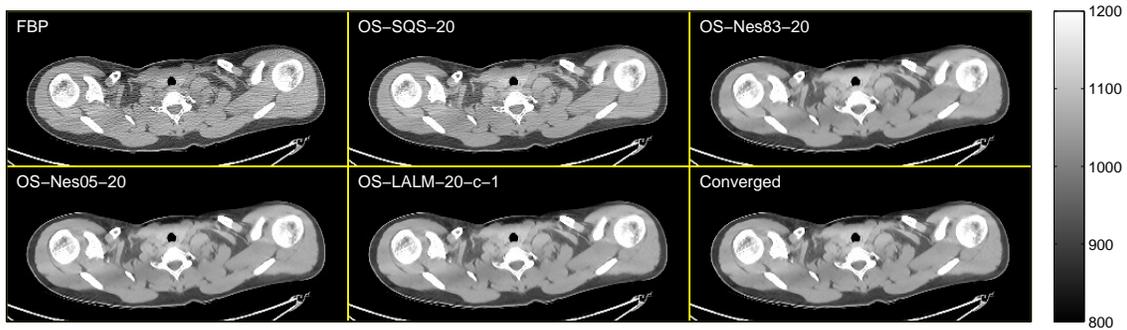
(a)
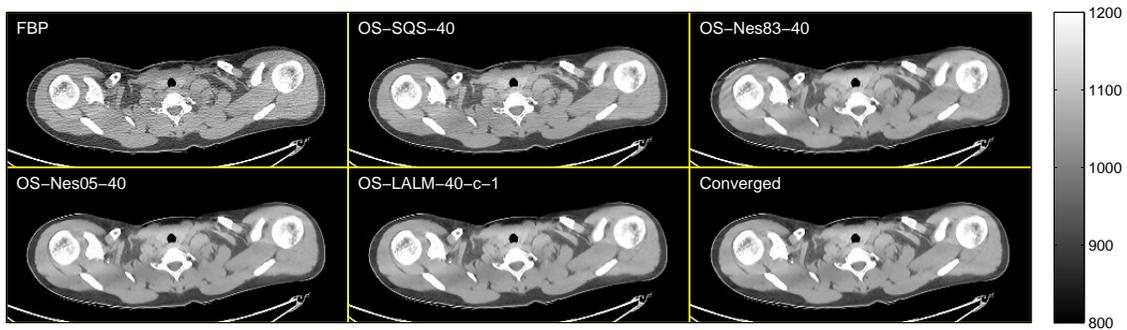
(b)

Figure 2: Cropped images (displayed from 800 to 1200 HU) from the central transaxial plane of the reconstructed image at the 30th iteration $\mathbf{x}^{(30)}$ using OS-based algorithms with (a) 20 subsets and (b) 40 subsets, respectively.
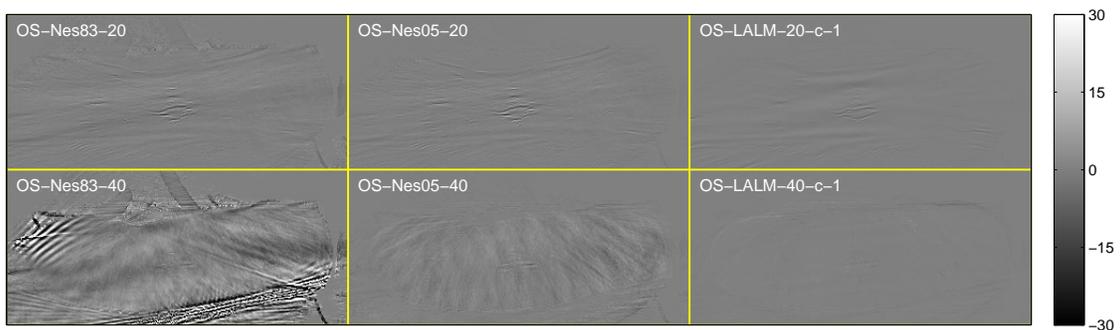
Figure 3: Cropped images (displayed from -30 to 30 HU) from the central transaxial plane of the difference image at the 30th iteration $\mathbf{x}^{(30)} - \mathbf{x}^{\star}$ using OS-based algorithms with 20 and 40 subsets.

4. CONCLUSION

In this paper, we proposed an OS-accelerable splitting-based algorithm, OS-LALM, for solving penalized weighted least-squares X-ray CT image reconstruction problems using a linearized AL framework. To further accelerate the proposed algorithm, we also proposed a deterministic downward continuation approach that avoids tedious parameter tuning for fast convergence. Experimental results showed that our proposed algorithm significantly accelerates the "convergence" of X-ray CT image reconstruction with negligible overhead and greatly reduces the OS artifacts in the reconstructed image when using many subsets for OS acceleration.

ACKNOWLEDGEMENTS

REFERENCES

[1] M. R. Hestenes. Multiplier and gradient methods. *J. Optim. Theory Appl.*, 4(5):303–20, November 1969.

[2] M. J. D. Powell. A method for nonlinear constraints in minimization problems. In R. Fletcher, editor, *Optimization*, pages 283–98. Academic Press, New York, 1969.

[3] R. Glowinski and A. Marrocco. Sur lapproximation par elements nis dordre un, et la resolution par penalisation-dualite dune classe de problemes de dirichlet nonlineaires, rev. francaise daut. *Inf. Rech. Oper.*, R-2:41–76, 1975.

[4] D. Gabay and B. Mercier. A dual algorithm for the solution of nonlinear variational problems via finite-element approximations. *Comput. Math. Appl.*, 2(1):17–40, 1976.

[5] J. Eckstein and D. P. Bertsekas. On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators. *Mathematical Programming*, 55(1-3):293–318, April 1992.

[6] S. Ramani and J. A. Fessler. A splitting-based iterative algorithm for accelerated statistical X-ray CT reconstruction. *IEEE Trans. Med. Imag.*, 31(3):677–88, March 2012.

[7] M. G. McGaffin, S. Ramani, and J. A. Fessler. Reduced memory augmented Lagrangian algorithm for 3D iterative X-ray CT image reconstruction. In *Proc. SPIE 8313 Medical Imaging 2012: Phys. Med. Im.*, page 831327, 2012.

[8] H. Erdoğan and J. A. Fessler. Ordered subsets algorithms for transmission tomography. *Phys. Med. Biol.*, 44(11):2835–51, November 1999.

[9] Z. Lin, R. Liu, and Z. Su. Linearized alternating direction method with adaptive penalty for low-rank representation. In *Adv. in Neural Info. Proc. Sys.*, pages 612–20, 2011.

[10] X. Wang and X. Yuan. The linearized alternating direction method of multipliers for Dantzig selector. *SIAM J. Sci. Comput.*, 34(5):A2792–A2811, 2012.

[11] J. Yang and X. Yuan. Linearized augmented Lagrangian and alternating direction methods for nuclear norm minimization. *Math. Comp.*, 82(281):301–29, January 2013.

[12] Y. Xiao, S. Wu, and D. Li. Splitting and linearizing augmented Lagrangian algorithm for subspace recovery from corrupted observations. *Adv. Comput. Math.*, 38(4):837–58, May 2013.

[13] H. Nien and J. A. Fessler. Combining augmented Lagrangian method with ordered subsets for X-ray CT image reconstruction. In *Proc. Intl. Mtg. on Fully 3D Image Recon. in Rad. and Nuc. Med*, pages 280–3, 2013.

[14] D. Kim, S. Ramani, and J. A. Fessler. Ordered subsets with momentum for accelerated X-ray CT image reconstruction. In *Proc. IEEE Conf. Acoust. Speech Sig. Proc.*, pages 920–3, 2013.

[15] D. Kim, S. Ramani, and J. A. Fessler. Accelerating X-ray CT ordered subsets image reconstruction with Nesterov's first-order methods. In *Proc. Intl. Mtg. on Fully 3D Image Recon. in Rad. and Nuc. Med*, pages 22–5, 2013.

[16] E. Esser, X. Zhang, and T. Chan. A general framework for a class of first order primal-dual algorithms for convex optimization in imaging science. *SIAM J. Imaging Sci.*, 3(4):1015–46, 2010.

[17] X. Zhang, M. Burger, X. Bresson, and S. Osher. Bregmanized nonlocal regularization for deconvolution and sparse reconstruction. *SIAM J. Imaging Sci.*, 3(3):253–76, 2010.

[18] X. Zhang, M. Burger, and S. Osher. A unified primal-dual algorithm framework based on Bregman iteration. *Journal of Scientific Computing*, 46(1):20–46, 2011.

[19] H. Nien and J. A. Fessler. Fast X-ray CT image reconstruction using the linearized augmented Lagrangian method with ordered subsets. In preparation.

[20] I. Daubechies, M. Defrise, and C. D. Mol. An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. *Comm. Pure Appl. Math.*, 57(11):1413–57, November 2004.

[21] D. P. Bertsekas. *Nonlinear programming*. Athena Scientific, Belmont, 2 edition, 1999.

[22] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM J. Imaging Sci.*, 2(1):183–202, 2009.

[23] Y. Nesterov. A method of solving a convex programming problem with convergence rate $O(1/k^2)$. *Soviet Math. Dokl.*, 27(2):372–76, 1983.

[24] Y. Nesterov. Smooth minimization of non-smooth functions. *Mathematical Programming*, 103(1):127–52, May 2005.