Improve Path Seeking Accuracy for Iterative Reconstruction Using the Karush-Kuhn-Tucker Conditions

Meng Wu, Andreas Maier, Qiao Yang, and Rebecca Fahrig

***Abstract*—Iterative reconstruction (IR) techniques have demonstrated many advantages in X-ray CT reconstruction. The statistical iterative reconstruction approach is often modeled as an optimization problem including a data fitting function and a penalty function. The tuning parameter value that regulates the strength of the penalty function is critical for achieving good reconstruction results. However, appropriate tuning parameter values that are suitable for the scan protocols and imaging tasks are often difficult to choose. In this work, we describe path seeking algorithms that are capable of generating a series of IR images with different strengths of the penalty function. We add an optimization step in the path seeking algorithm to improve the accuracy. Numerical simulation results have shown the proposed true path seeking algorithm has 20% to 30% less errors than the approximate path seeking algorithm [1].**

***Index Terms*—CT, iterative reconstruction, path seeking, KKT conditions**

I. Introduction

Iterative reconstruction (IR) methods for 3D CT offer numerous advantages such as the potential for improved image quality and reduced dose, as compared to the conventional methods such as filtered back-projection (FBP) [2], [3]. The commonly used maximum likelihood (ML) problem in iterative reconstruction is often ill-posed and generates incorrect reconstructions, especially when the number of projections is small, or the data is very noisy [4]. A common remedy is to add a regularization to the ML problem and formulate it as a penalized maximum likelihood problem (PML) with

$$\mu = \underset{\mu \geq 0}{\operatorname{argmax}} \Psi(\mu) - \beta \mathrm{R}(\mu), \tag{1}$$

where $\Psi(\mu)$ is the log-likelihood function, $\mathrm{R}(\mu)$ is the penalty function (also known as regularization function), and β is the tuning parameter that regulates the strength of the penalty function. In this study, we consider the penalized weighted least-squares (PWLS) algorithm

$$\mu = \underset{\mu \geq 0}{\operatorname{argmin}} \frac{1}{2} \sum_{i}^{I} w_i ([\mathbf{A}\mu]_i - l_i)^2 + \beta \mathrm{R}(\mu) \tag{2}$$

where $\mathbf{A}$ denotes the system matrix for the data acquisition geometry, l denotes the logged normalized projection, and w is the least-squares weight.

M. Wu and R. Fahrig are with the Department of Radiology, Stanford University, USA e-mail:mengwu@stanford.edu.

A. Maier and Q. Yang are with Pattern Recognition Lab, Friedrich-Alexander University of Erlangen-Nrnberg, Germany.

This work is supported by Erlangen Graduate School in Advanced Optical Technologies.

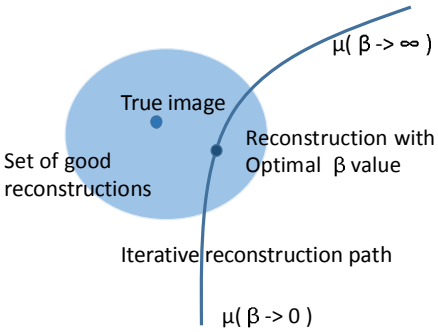

Fig. 1: Reconstruction path of the penalized maximum likelihood iterative method.

The regularization function $\mathrm{R}(\mu)$ often penalizes the differences in the values of neighboring pixels [3]–[6], or produces uniform point spread function [7]. Different tuning parameter β values generate different reconstructed images (solutions to slightly different optimization problems). In fact, the values of the tuning parameter β ($0 \leq \beta \leq \infty$) in Eqn. (1) produce a series of reconstructions $\mu(\beta)$. The value of the tuning parameter is critical to the reconstruction results. For example, if β is too small, the regularization is not strong enough to suppress noise and artifacts; if β is too big, the image exhibits patchy behavior [3]. Figure 1 shows the relationship between the PML reconstruction and the true image with different tuning parameter values.

To the best of our knowledge, there is no perfect way of deciding the value of β that would lead to the best reconstruction for clinical use. Because of the high computational load of IR, directly computing multiple solutions ($\mu(\beta)$) via numerical optimization would not be suitable for practical use. Wu et. al. proposed an approximate path seeking algorithm (APS) to efficiently compute a series of IR images $\mu(\beta)$ with different strength of regularization [1]. However, the APS algorithm assumes that each of the newly computed images are on or near the reconstruction path. Numerical errors and acceleration techniques could cause accumulated errors in the path seeking. The accumulated errors in the approximate path seeking become more severe if the reconstruction path is ”long”. Because each image on the path is a solution in the series of the optimization problems, one can estimate the current tuning parameter value in the middle of the path seeking using the Karush-Kuhn-Tucker (KKT) conditions [9].

An additional optimization step can be then carried out to ensure the computed path images are on the reconstruction path. In this paper, we first review the approximate path seeking (APS) algorithm, and then describe a true path seeking (TPS) to improve the accuracy.

II. Methods

A. Approximate path seeking (APS)

Friedman proposed a fast generalized path seeking (GPS) algorithm that produces solutions closely approximating the path of the constrained regression problems [8]. The GPS algorithm uses the ratio of the gradients to update one regression variable each time, which is accurate and suitable for a small regression problem. However, the number of the variables in the image reconstruction problem is identical to the number of voxels. Considering the heavy computational load of calculating the gradient of the likelihood function $\Psi(\mu)$, we proposed to update a fraction of pixels each time instead of one pixel. Moreover, a path direction is introduced to improve the path seeking efficiency, which is defined as

$$d_j = \mu_j(\beta_2) - \mu_j(\beta_1), \tag{3}$$

where $\mu_j(\beta_2)$ and $\mu_j(\beta_1)$ are two solutions on the path with $\beta_1 < \beta_2$. When β_1 and β_2 are close, we assume the partial derivative of the penalty function follows

$$\frac{\partial \mathrm{R}(\mu(\beta))}{\partial \mu_j} \cdot d_j > 0 \tag{4}$$

for all $\beta \in [\beta_1, \beta_2]$. This is equivalent to saying the path of each pixel is locally monotonic in β. Let us define the negative gradient of the weighted least-squares and penalty function as

$$\begin{aligned} g_j(\mu) &= -\frac{\partial \Psi(\mu)}{\partial \mu_j} = -\left[\mathbf{A}'\mathrm{diag}\{w\}(\mathbf{A}\mu - l)\right]_j, \\ h_j(\mu) &= -\frac{\partial \mathrm{R}(\mu)}{\partial \mu_j}. \end{aligned} \tag{5}$$

The **approximate path seeking** algorithm is

Set up initial conditions:
Reconstruct two images $\mu(\beta_1)$ and $\mu(\beta_2)$ for selected path range $[\beta_1, \beta_2]$
$\mu = \mu(\beta_1)$
Loop {
1) $d = \mu(\beta_2) - \mu$
2) Compute $\lambda_j = h_j(\mu)/|g_j(\mu)|$
3) Find $S = \{j | h_j(\mu) \cdot g_j(\mu) > 0\}$
4) If (S is not empty) {
5) $\mu_j = \mu_j + \Delta v \cdot \mathrm{sign}(\lambda_j)$ for all $j \in S$
6) } Else {
7) If ($\lambda_j \cdot d_j < 0$), then $\lambda_j = 0$
8) Find t such that $\mathbf{P}\{|\lambda_j| \geq t\} \leq p$
9) If ($|\lambda_j| \geq t$), $\mu_j = \mu_j + \Delta v \cdot \mathrm{sign}(\lambda_j)$
10) }
} Until $\|\mu - \mu(\beta_2)\|$ stops decreasing.

The variable p is the update percentage of the pixels in each iteration, and $\Delta v > 0$ is a small increment value, e.g. 1 - 3 HU. At each step, the pixel-wise ratio of the two gradients indicates the relative effects of the data fitting (likelihood) and penalty functions at current μ. Line 5 updates the pixel if both gradients are in the same direction. Line 7 ensures the algorithm only considers the rest of the pixels that have the same updating direction as d_j. Line 8 selects the pixels that have the largest ratios of the gradients. The selected pixel is then incremented by a small fixed amount (Δv) in the direction of λ_j. Figure 2 shows a 2D illustration of the approximate path seeking algorithm. The green arrows $d_1 \ldots d_4$ are the direction vectors (line 1) toward final path image $\mu(\beta_2)$; the orange arrows are the fixed size updates to the image (line 9). In each iteration, one out of two variables was updated by Δ_v in the same direction as d and λ.

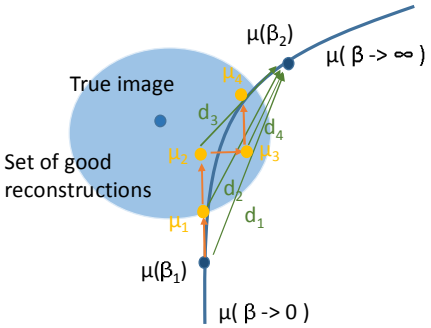

Fig. 2: 2D illustration of the approximate path seeking algorithm. The yellow points ($\mu_1 \ldots \mu_4$) are the path images computed by the approximate path seeking algorithm.

In the previously described APS algorithm, the path image goes from $\mu(\beta_1)$ to $\mu(\beta_2)$. We call it a forward model, because it shows the effect of increasing the strength of the penalty function. The path seeking algorithm can also work in the opposite direction (backward model). We just need to change the initialization to $\mu(\beta_2)$, and use $d_j = \mu_j(\beta_1) - \mu_j$ and $\lambda_j = g_j(\mu)/|h_j(\mu)|$.

B. True path seeking (TPS)

The APS method assumes that the ratios of gradients (λ_j) and path direction (d_j) select correct pixels to update. The accuracy of the APS algorithm relies on that each newly computed image is on or near the reconstruction path so that the APS updates are tracking the effects of the regularization instead of doing optimization. Numerical errors in reconstructions and acceleration techniques (e.g. inappropriate step sizes and order subset method) could cause accumulated errors in the APS. Not surprisingly, when the distance between $\mu(\beta_1)$ and $\mu(\beta_2)$ is large, the accumulated errors will be greater too.

Fortunately, the path images are supposed to be solutions of a series optimization problems, which only have different β values. Thus, if we can estimate the corresponding β value of each path image, additional optimization steps can then be executed to improve the accuracy of the path seeking. If each estimated image is a close solution of the convex optimization problem, the tuning parameter value can be estimated using the Karush-Kuhn-Tucker (KKT) conditions [9]. The KKT

condition for the PWLS problem is

$$\begin{aligned} g_j(\mu^*) + \beta \cdot h_j(\mu^*) - \eta_j^* &= 0 \\ \eta_j^* \cdot \mu_j^* &= 0 \quad \text{for all} \quad j, \\ \eta_j^* &\geq 0 \end{aligned} \tag{6}$$

where η_j^* is the Lagrange multiplier of the non-negative constrains. The KKT condition implies that if $\mu_j^* \neq 0$ then

$$g_j(\mu^*) + \beta \cdot h_j(\mu^*) = 0. \tag{7}$$

Therefore, the tuning parameter value β can be estimated by

$$\beta \approx \text{Median} \left\{ \frac{g_j(\mu^*)}{h_j(\mu^*)}, \quad \forall j : \mu_j > 0 \right\}. \tag{8}$$

Using the median is more numerically robust than using the average, because the $h_j(\mu^*)$ can be a very small number. With a way of estimating β, we propose a **true path seeking (TPS)** algorithm as

---

Set up initial conditions:
Reconstruct two images $\mu(\beta_1)$ and $\mu(\beta_2)$ for selected path range $[\beta_1, \beta_2]$
$\mu = \mu(\beta_1)$
Loop {

1) Estimate β using Eqn. (8)
2) $u = u - \alpha \left(g(\mu) + \beta \cdot h(\mu) \right).$
3) $d = \mu(\beta_2) - \mu$
4) Compute $\lambda_j = h_j(\mu) / |g_j(\mu)|$
5) Find $S = \{ j | h_j(\mu) \cdot g_j(\mu) > 0 \}$
6) If (S is not empty) {
7) $\quad \mu_j = \mu_j + \Delta v \cdot \text{sign}(\lambda_j)$ for all $j \in S$
8) } Else {
9) $\quad$ If ($\lambda_j \cdot d_j < 0$), then $\lambda_j = 0$
10) $\quad$ Find t such that $\mathbf{P}\{|\lambda_j| \geq t\} \leq p$
11) $\quad$ If ($|\lambda_j| \geq t$), $\mu_j = \mu_j + \Delta v \cdot \text{sign}(\lambda_j)$
12) }

} Until $\|\mu - \mu(\beta_2)\|$ stops decreasing.

Lines 1-2 are the additional minimization step in the TPS algorithm to draw μ closer to the reconstruction path. The minimization step does not need to be executed for many iterations because the image is already very close to the path when step size is small. Figure 3 shows a 2D illustration of the TPS algorithm. The image from the previous path seeking step (orange points) are corrected by the minimization step (red arrows) to more accurate path images (red points) before the path seeking.

Moreover, computing the gradient $g(\mu)$ requires forward and backward projection that has high computation cost. The gradients computed in line 2 may be reused in line 4. Reusing the gradient reduces the computational cost, but the accuracy of the path seeking step may be affected. We called the algorithm using the newly re-computed gradients as TPS-1 and using previously computed gradient as TPS-2.

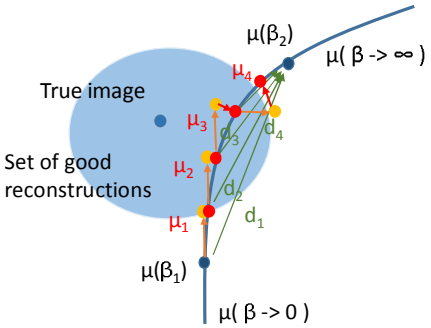

Fig. 3: 2D illustration of the true path seeking algorithm. The red points ($\mu_1 \ldots \mu_4$) are the path images computed by the true path seeking algorithm.

III. Simulations

A thorax XCAT phantom is used to simulate projections for a Siemens SOMATOM 32 row clinical CT geometry in 3D. We reconstructed a 512 × 512 × 32 image from an undersampled chest axial CT scan. The size of the original sinogram is 768 × 32 × 721 (half scan), and we uniformly undersampled the number of projection views from 721 to 91. We used the total variation (TV, i.e., the L1 norm of the first-order derivative) as the penalty function R, which has shown many good properties in the sparse view reconstruction. The unattenuated scan factor is 1×10^5 photon counts / pixel. The direct optimization solutions of the PWLS problem for 20 β values log spaced from 10 to 200 were achieved using the ordered-subsets linearized augmented Lagrangian method [10] with 100 iterations.

The update percentage p was set to 20%, with step size Δv equal to 1 HU for both the APS and the proposed TPS algorithms. Ordered subsets of 3 were used to approximate the gradient g_j in Eqn. (5) with a smaller number of projections. A total of 40 path images with equal L_1 distances between $\mu(\beta = 10)$ and $\mu(\beta = 200)$ were computed and stored using both forward and backward path seeking models. The accuracy of the path seeking algorithms was evaluated by comparing the reconstructions using numerical optimization and the path seeking.

IV. Results

Figure 4 shows five frames in the path of the PWLS method using the direct optimization and the TPS2 forward method. Both approaches provided sequences of images from noisy reconstructions with view undersampling artifacts to over blurred images. The root-mean-squared-difference (RMSE) and mean-absolute-difference (MAD) between the first ($\mu(\beta = 10)$) and the last ($\mu(\beta = 200)$) images are 44 HU and 15.5 HU, respectively. Figure 5 shows eight frames of 6 cm × 6 cm region-of-interest (ROI) of the path images using direct optimization and the proposed path seeking (forward) methods. The path images generated by the path seeking algorithms are similar to the direct optimization results. The path images using the path seeking algorithms are more evenly spread between the first and last images. There are several

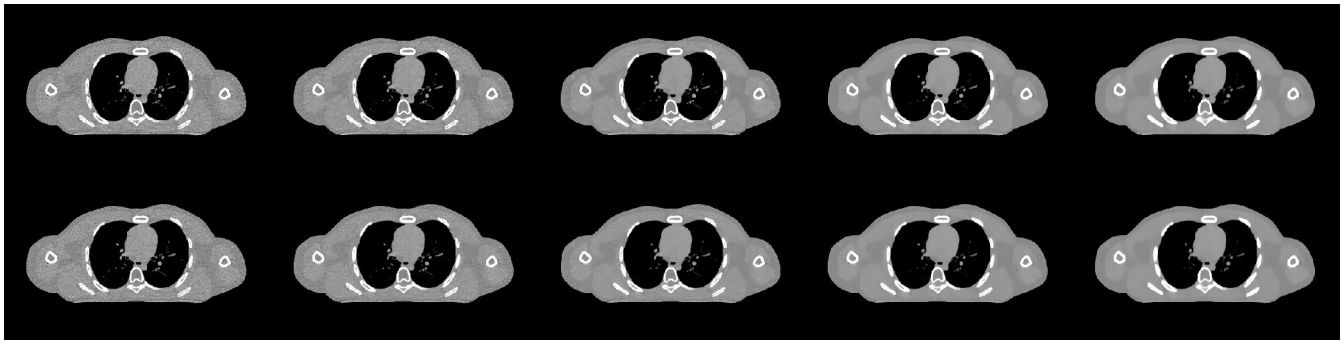

Fig. 4: Five example images in the PWLS reconstructions path computed using the direct optimization (top) and the proposed TPS2 algorithm (bottom). The display window is [-400, 400] HU.

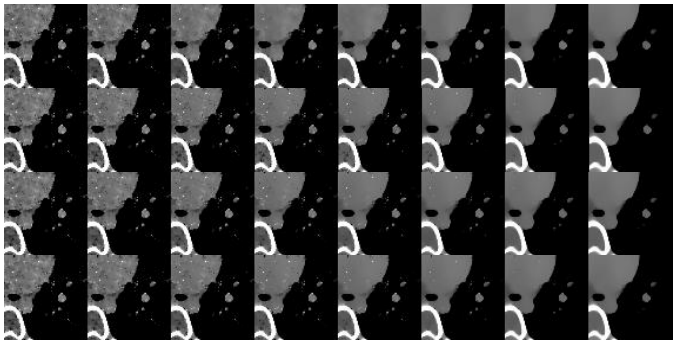

Fig. 5: Sequences of path images of a 6 cm $\times$ 6 cm ROI using the direct optimization and the proposed path seeking (forward model) algorithms. From top to bottom: direct optimization, APS, TPS1 and TPS2. The display window is [-100, 300] HU.

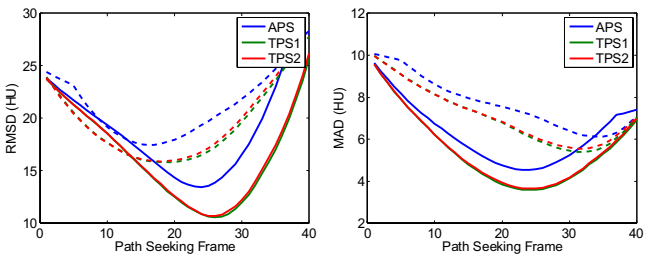

Fig. 6: The RMSD and MAD measurements of the entire path images generated by the proposed method compared to a directly solved PWLS images. The solid lines are forward model, and the dashed lines are backward model.

pixels in the image using the path seeking algorithms that are updated slower than the rest of the image, because the total path lengths of those pixels are too big for the small step size. The TPS images haves smaller errors than the APS images because the additional optimization step corrects those pixels too. We expect that the errors due to the step size become smaller when using larger number of projections.

Quantitive evaluations using RMSD and MAD are shown in Figure 6. The 40 path images that are computed by the path seeking methods were compared to one directly solved PWLS image at $\beta = 50$ (the middle image in Figure 4 first row). The closest path seeking images has RMSD around 10 HU and MAD about 4 HU. The RMSD is much larger than the MAD because of the large errors in several pixels that are also noticeable in Figure 5. The minimum differences using the TPS to the direct optimization result are 20% to 30% smaller than using the APS method in both RMSD and the MAD. There is no big difference in accuracy between the TPS-1 and TPS-2 method. Because the TPS-2 algorithm requires only half times of computing the gradients compare to the TPS1 algorithm, the TPS-2 version is much faster than the TPS-1 version. The forward model is more accurate than the backward model except when β is very large.

V. Conclusion

In this work, we proposed to add an optimization step in the path seeking algorithm to improve the accuracy. The tuning parameter of the path images were estimated using the KKT condition of the convex optimization problem. Numerical simulation results have shown the proposed TPS algorithm has 20% to 30% less errors than the APS algorithm.

References

[1] M. Wu, Q. Yang, A. Maier, and R. Fahrig, "Approximate Path Seeking for Statistical Iterative Reconstruction," in *Proc. of SPIE Med. Imaging*, pp. 9412–46, 2015 (To be appeared).

[2] J.-B. Thibault, K. D. Sauer, C. A. Bouman, and J. Hsieh, "A three-dimensional statistical approach to improved image quality for multislice helical CT," *Med. Phys.*, vol. 34, no. 11, p. 4526, 2007.

[3] J. Tang, B. E. Nett, and G.-H. Chen, "Performance comparison between total variation (TV)-based compressed sensing and statistical iterative reconstruction algorithms.," *Phys. Med. Biol.*, vol. 54, no. 19, pp. 5781–5804, 2009.

[4] I. A. Elbakri and J. A. Fessler, "Statistical image reconstruction for polyenergetic X-ray computed tomography," *IEEE Trans. Med. Imaging*, vol. 21, pp. 89–99, Feb. 2002.

[5] J. Bian, J. H. Siewerdsen, X. Han, E. Y. Sidky, J. L. Prince, C. a. Pelizzari, and X. Pan, "Evaluation of sparse-view reconstruction from flat-panel-detector cone-beam CT.," *Phys. Med. Biol.*, vol. 55, pp. 6575–99, Nov. 2010.

[6] L. Pfister and Y. Bresler, "Model-based iterative tomographic reconstruction with adaptive sparsifying transforms," in *Proc. of SPIE Med. Imaging* (C. A. Bouman and K. D. Sauer, eds.), vol. 9020, p. 90200H, Mar. 2014.

[7] J. W. Stayman and J. A. Fessler, "Regularization for uniform spatial resolution properties in penalized-likelihood image reconstruction.," *IEEE Trans. Med. Imaging*, vol. 19, pp. 601–15, June 2000.

[8] J. Friedman, "Fast sparse regression and classification," *Int. J. Forecast.*, vol. 1, 2012.

[9] S. Boyd and L. Vandenberghe, *Convex Optimization*, vol. 25. Cambridge University Press, 2010.

[10] H. Nien and J. A. Fessler, "Fast Splitting-Based Ordered-Subsets X-Ray CT Image Reconstruction," in *Proc. of the third int. conf. on image formation in X-ray CT*, pp. 291–294, 2014.