Monotonic Algorithms for Transmission Tomography

Hakan Erdoğan and Jeffrey A. Fessler

Department of EECS, University of Michigan
4415 EECS Bldg., 1301 Beal Ave. Ann Arbor, MI 48109-2122
email: erdogan@umich.edu, Voice:(734) 647 8390, FAX:(734) 764 8041

Abstract— **We present a framework for designing fast and monotonic algorithms for transmission tomography penalized-likelihood image reconstruction. The new algorithms are based on paraboloidal surrogate functions for the log-likelihood. Due to the form of the log-likelihood function, it is possible to find low curvature surrogate functions that guarantee monotonicity. Unlike previous methods, the proposed surrogate functions lead to monotonic algorithms even for the nonconvex log-likelihood that arises due to background events such as scatter and random coincidences. The gradient and the curvature of the likelihood terms are evaluated only once per iteration. Since the problem is simplified at each iteration, the CPU time is less than that of current algorithms which directly minimize the objective, yet the convergence rate is comparable. The simplicity, monotonicity and speed of the new algorithms are quite attractive. The convergence rates of the algorithms are demonstrated using real and simulated PET transmission scans.**

Keywords— **Maximum likelihood, Image reconstruction, PET, SPECT**

I. Introduction

Attenuation correction is required for quantitatively accurate image reconstruction in emission tomography. The accuracy of this correction is very important in both PET and SPECT [1]. Transmission scans are performed to measure the attenuation characteristics of the object and to determine attenuation correction factors (ACFs) for emission image reconstruction. Conventional smoothing methods for ACF computation are simple and fast, but suboptimal [2], [3]. For low-count transmission scans, statistical reconstruction methods provide lower noise ACFs. However, a drawback of statistical methods is the slow convergence (or possible divergence) of current reconstruction algorithms. This paper describes fast and monotonic algorithms for penalized-likelihood reconstruction of attenuation maps from transmission scan data. These reconstructed attenuation maps can be reprojected to calculate lower noise ACFs for improved emission image reconstruction.

Statistical methods for reconstructing attenuation maps from transmission scans are becoming increasingly important in thorax and whole-body PET imaging, where lower counts and short scan times are typical. 3-D PET systems also require attenuation correction, which can be done by reprojecting 2-D attenuation maps. SPECT systems with transmission sources are becoming increasingly available where statistical algorithms can be efficiently used for attenuation map reconstructions. For low-count transmission scans, the non-statistical FBP reconstruction method systematically overestimates attenuation map coefficients, whereas data-weighted least squares methods (WLS) for transmission reconstruction are systematically negatively biased [4]. By accurate statistical modeling, penalized-likelihood reconstruction of attenuation maps eliminates the systematic bias and yields lower variance relative to linear methods. Hence, we focus on penalized-likelihood image reconstruction rather than WLS in this paper.

There are many reconstruction algorithms based on the Poisson model for transmission measurements. The expectation maximization (EM) algorithm [5], which led to a simple M-step for the emission problem, does not yield a closed form expression for the M-step in the transmission case [6]. Modifications of the transmission ML-EM algorithm [7], [8], [9] as well as algorithms that directly optimize the penalized-likelihood objective [10], [11], [12], [13], [3] have been introduced. Some of these algorithms seem to converge rapidly in the convex case.

However, up to now, no practically realizable monotonic (or convergent) algorithm has been found for the penalized-likelihood problem when the objective is not convex. The negative log-likelihood is nonconvex when there are "background" counts in the data. This is unavoidable in PET and SPECT, due to the accidental coincidences in PET and emission crosstalk[1] in SPECT. The assumption of no background counts may be reasonable in X-ray CT.

In this paper, we present a new algorithm which is guaranteed to be monotonic even when the objective function is nonconvex. This algorithm depends on paraboloidal surrogate functions for the log-likelihood which transform the problem into a simpler quadratic optimization problem at each iteration. The transformed problem at each iteration is similar to a Penalized Weighted Least Squares (PWLS) problem, and thus has a familiar and simple form. This quadratic problem need not be solved exactly; an algorithm that monotonically decreases the surrogate function suffices. Since evaluating the gradient and Hessian of the surrogate function is much less costly, the CPU time per iteration is greatly reduced as compared to algorithms that directly attempt to minimize the objective function, such

This work was supported in part by NIH grants CA-60711 and CA-54362.

[1] Even though different photon energies are used in simultaneous emission/transmission SPECT imaging, some emission events are recorded in the transmission energy window due to Compton scatter and finite energy resolution.

as coordinate descent. Remarkably, the convergence rate is comparable to other direct algorithms. For nonconvex objective functions, monotonicity alone does not guarantee convergence to the global minimizer when local minima exist, but it does ensure that the estimates do not diverge since the likelihood is bounded. Whether the transmission log likelihood has multiple local minima is an open question.

The "surrogate" or "substitute" function idea is not new to the tomographic reconstruction area. EM algorithms can be viewed as providing a surrogate function for the log-likelihood function by means of a statistically more informative "complete" data set which is unobservable [5]. The conditional expectation of the log-likelihood function for this new space is often easier to maximize, having a closed form for the emission case. This statistical construction of surrogate functions is somewhat indirect and seems to yield a limited selection of choices. De Pierro has developed surrogate functions for nonnegative least squares problems based solely on convexity arguments, rather than statistics [14]. Our proposed approach is similar in spirit.

The EM algorithm did not result in a closed form M-step for the transmission case [6], so direct minimization of the objective function became more attractive. Cyclic Newtonian coordinate descent (CD,NR) [11] has been used effectively in transmission tomography. However, coordinate descent based on Newton's iteration for each pixel is not guaranteed to be monotonic. Furthermore, an iteration of Newton-based coordinate descent requires at least M exponentiations and $17M$ floating point operations[2] , where M is the (very large) number of nonzero elements in the system matrix $\boldsymbol{A}$ in (1) below. These exponentiations and floating point operations constitute a significant fraction of the CPU time per iteration. Recently, Zheng *et al.* introduced a "functional substitution" (FS) method [15], [16] which is proven to be monotonic for transmission scans with no background counts ($r_i = 0$ in (1) below). Like coordinate descent, FS algorithm cyclically updates the coordinates of the image vector, *i.e.* the attenuation map values for each pixel. However, instead of minimizing the original complex objective function with respect to each parameter, FS algorithm minimizes a 1-D parabolic surrogate function. The minimization of the surrogate is guaranteed to monotonically decrease the original objective function if the derivative of the negative log-likelihood is concave (which is true when $r_i = 0$) [15], [16]. On the other hand, the FS algorithm requires at least $2M$ exponentiations and $17M$ floating point operations[3] per iteration, which means that the guarantee of monotonicity comes at a price of significantly increased computation time per iteration for that method. Furthermore, the FS algorithm is not monotonic in the nonconvex case of interest in PET and SPECT, where $r_i \neq 0$.

De Pierro [17] has used a surrogate function for the penalty part of the penalized-likelihood problem for convex penalties. The surrogate function idea was also used in several algorithms which update a group of pixel values at a time instead of sequential update of each pixel. Examples of these types of algorithms are the convex algorithm of [18] which updates all pixels simultaneously and the grouped coordinate ascent (GCA) algorithm of [12], [15] which updates a subset of pixels at a time. The surrogate functions used in these algorithms were obtained using De Pierro's convexity trick [17] to form a separable function that is easier to minimize than the non-separable original objective function. The convergence rates per iteration decrease due to the higher curvature of these surrogate functions, but these algorithms require less computation per iteration as compared to single coordinate descent [11] and are parallelizable. Furthermore, it is trivial to impose the nonnegativity constraint with an additively separable surrogate function [12].

In this paper, we propose to use a global surrogate function for the original objective function. This global surrogate function is *not* separable, but has a simple quadratic form. The method is based on finding 1-D parabolic functions that are tangent to and lie above each of the terms in the log-likelihood, similar to Huber's method for robust linear regression [19]. Whereas Huber considered strictly convex cost functions, we extend the method to derive provably monotonic algorithms even for nonconvex negative log-likelihood functions. Remarkably, these algorithms require *less* CPU time to converge than the fastest algorithm introduced before (GCA of [12]) and as an additional advantage, they are proven to be monotonic. We call the new approach to image reconstruction the "Paraboloidal Surrogates" (PS) method.

In the rest of this paper, we describe the problem, develop the new algorithm, and present representative performance results on real PET transmission data.

II. The Problem

The measurements in a photon-limited application such as PET or SPECT are modeled appropriately as Poisson random variables. In transmission tomography, the means of the prompt coincidences are related exponentially to the projections (or line integrals) of the attenuation map through Beer's Law [6]. In addition, the measurements are contaminated by extra "background" counts due mostly to random coincidences and scatter in PET and emission crosstalk in SPECT. Thus, it is realistic to assume the following model:

$$y_i \sim \text{Poisson}\{b_i e^{-[\boldsymbol{A}\mu]_i} + r_i\},\ i = 1, \ldots, N, \tag{1}$$

where N is the number of measurements, μ_j is the average linear attenuation coefficient in voxel j for $j = 1, \ldots, p$, and p denotes the number of voxels. The notation $[\boldsymbol{A}\mu]_i = \sum_{j=1}^{p} a_{ij}\mu_j$ represents the ith "line integral" of the attenuation map μ, and $\boldsymbol{A} = \{a_{ij}\}$ is the $N \times p$ system matrix. We assume that $\{b_i\}$, $\{r_i\}$ and $\{a_{ij}\}$ are known nonnegative

[2] This can be reduced to $9M$ floating point operations if the denominator terms are precomputed similar to section III-F in this paper.

[3] Precomputation of the denominator terms in FSCD would destroy monotonicity.

constants[4], where r_i is the mean number of background events, b_i is the blank scan factor, and y_i represents the number of transmission events counted by the ith detector (or detector pair in PET).

We seek to find a statistical estimate of the attenuation map μ which "agrees" with the data and is anatomically reasonable. For this purpose, a natural approach is to use a likelihood-based estimation strategy. The log-likelihood function for the independent transmission data is:

$$L(\mu) = \sum_{i=1}^{N} \left\{ y_i \log(b_i e^{-[\boldsymbol{A}\mu]_i} + r_i) - (b_i e^{-[\boldsymbol{A}\mu]_i} + r_i) \right\},$$

ignoring constant terms. The log-likelihood depends on the parameter vector μ through only its projections $[\boldsymbol{A}\mu]_i$ and can be expressed in the following form:

$$-L(\mu) = \sum_{i=1}^{N} h_i([\boldsymbol{A}\mu]_i), \tag{2}$$

where the contribution of the ith measurement to the negative log-likelihood is given by:

$$h_i(l) \overset{\triangle}{=} (b_i e^{-l} + r_i) - y_i \log(b_i e^{-l} + r_i). \tag{3}$$

The proposed algorithm exploits the additive form of (2). Directly minimizing $-L(\mu)$ (maximum likelihood) results in a very noisy estimate $\hat{\mu}$ due to the ill-posed nature of the problem. However, it is well known that the attenuation map in the body consists of approximately locally homogeneous regions. This property has formed the basis of many segmentation methods for transmission scans [22]. Rather than applying hard segmentation, we add to the negative log-likelihood a penalty term which encourages piecewise smoothness in the image, resulting in the penalized-likelihood image reconstruction formulation as given below:

$$\hat{\mu} = \arg\min_{\mu \geq 0} \Phi(\mu), \quad \Phi(\mu) = -L(\mu) + \beta R(\mu). \tag{4}$$

Our goal is to develop an algorithm for finding the minimizing $\hat{\mu}$ with minimal CPU time.

We consider roughness penalties $R(\mu)$ that can be expressed in the following very general form [23], [17]:

$$R(\mu) = \sum_{k=1}^{K} \psi_k([\boldsymbol{C}\mu]_k), \tag{5}$$

where the ψ_k's are potential functions acting as a norm on the "soft constraints" $\boldsymbol{C}\mu \approx 0$ and K is the number of such constraints. The functions ψ_k we consider are convex, symmetric, nonnegative, differentiable and satisfy some more conditions that are listed in Section III-C. The β in equation (4) is a parameter which controls the level of smoothness in the final reconstructed image. For more explanation of the penalty function, see [23].

The objective function defined in (4) is not convex when there are nonzero background counts ($r_i \neq 0$) in the data. In this realistic case, there is no guarantee that there is a single global minimum. However, some practical algorithms exist that seem to work very well, yet none of them are proven to be monotonic. In this paper we introduce an algorithm that is monotonic even when Φ is not convex. The new approach is based on successive paraboloidal surrogate functions and will be explained in the rest of the paper.

III. Paraboloidal Surrogates Algorithms

The penalized-likelihood objective function $\Phi(\mu)$ has a complex form that precludes analytical minimization. Thus, iterative methods are necessary for minimizing $\Phi(\mu)$. Our approach uses the optimization transfer idea proposed by De Pierro [14], [17], summarized as follows. Let μ^n be the attenuation map estimate after the nth iteration. We would like to find a "surrogate" function[5] $\phi(\mu; \mu^n)$ which is easier to minimize or to monotonically decrease than $\Phi(\mu)$. This approach transforms the optimization problem into a simpler problem at each iteration, as illustrated in Figure 1. The following "monotonicity" condition on the surrogate function is sufficient to ensure that the iterates $\{\mu^n\}$ monotonically decrease Φ:

$$\Phi(\mu) - \Phi(\mu^n) \leq \phi(\mu; \mu^n) - \phi(\mu^n; \mu^n), \ \forall \mu \geq 0. \tag{6}$$

We restrict ourselves to differentiable surrogate functions, for which the following conditions are sufficient[6] to ensure (6):

$$\begin{array}{ll} 1. & \phi(\mu^n; \mu^n) = \Phi(\mu^n) \\ 2. & \left.\dfrac{\partial \phi}{\partial \mu_j}(\mu; \mu^n)\right|_{\mu=\mu^n} = \left.\dfrac{\partial \Phi}{\partial \mu_j}(\mu)\right|_{\mu=\mu^n}, j = 1, \ldots, p \\ 3. & \phi(\mu; \mu^n) \geq \Phi(\mu) \text{ for } \mu \geq 0. \end{array} \tag{7}$$

Figure 1 illustrates two surrogate functions that are tangent to the original objective at the current iterate and lie above it for all feasible values of the parameters.

The EM algorithm [6] provides a statistical method for constructing surrogate functions $\phi(\mu; \mu^n)$ satisfying the above conditions. However, in the transmission tomography problem, the natural EM surrogate is difficult to minimize and leads to slow convergence. In this paper, we construct a simpler surrogate using ordinary calculus rather than statistical techniques.

The log-likelihood function (2) has a certain kind of dependence on the parameters μ, namely through their projections $\boldsymbol{A}\mu$. The negative log-likelihood is the sum of individual functions h_i, each of which depends on a single

[4] The assumption that the background counts r_i are known nonnegative constants is an approximation. In PET, we estimate the r_i's by smoothing the delayed coincidences from the transmission scan [20]. Alternatively, one can use time scaled delayed coincidences from a blank scan (which are less noisy due to longer scan times) as the r_i factors [21] or use Bayesian estimation techniques to estimate r_i's from delayed coincidences [3], [20].

[5] We use the notation $\phi(\mu; \mu^n)$ to emphasize that the surrogate is a function of μ once μ^n is fixed and it changes for each μ^n, following the Q function notation of the EM algorithm [5].

[6] The second condition follows from the other two conditions for differentiable surrogate functions.

projection only. We can exploit this form of the likelihood function by selecting a 1-D surrogate function for each of the one-dimensional h_i functions in the projection (l) domain. The overall sum of these individual 1-D functions will be an appropriate surrogate for the likelihood part of the objective.

Let $l_i^n = [\boldsymbol{A}\mu^n]_i$ denote the estimate of the ith line integral of the attenuation coefficient at the nth iteration. We choose the following quadratic form for the surrogate functions q_i:

$$q_i(l; l_i^n) \triangleq h_i(l_i^n) + \dot{h}_i(l_i^n)(l - l_i^n) + \frac{1}{2}c_i(l_i^n)(l - l_i^n)^2, \quad (8)$$

where $c_i(l_i^n)$ is the curvature of the parabola q_i and $\dot{h}$ denotes first derivative of h. This construction ensures that $q_i(l_i^n; l_i^n) = h_i(l_i^n)$ and $\dot{q}_i(l_i^n; l_i^n) = \dot{h}_i(l_i^n)$ similar to (7). To ensure monotonicity, we must choose the curvatures to satisfy the following inequality at each iteration:

$$h_i(l) \leq q_i(l; l_i^n), \quad \text{for } l \geq 0. \quad (9)$$

After determining the parabolas, one can easily verify that the following function is a global surrogate function for the objective $\Phi(\mu)$ which satisfies the properties in (7):

$$\phi(\mu; \mu^n) = Q(\mu; \mu^n) + \beta R(\mu), \quad (10)$$

where

$$\begin{aligned} Q(\mu; \mu^n) &\triangleq \sum_{i=1}^{N} q_i([\boldsymbol{A}\mu]_i; l_i^n) &(11)\\ &= \Phi(\mu^n) + \boldsymbol{d}_h(l^n)'\boldsymbol{A}(\mu - \mu^n) \\ &\quad + \frac{1}{2}(\mu - \mu^n)'\boldsymbol{A}'\boldsymbol{D}(c_i(l_i^n))\boldsymbol{A}(\mu - \mu^n), &(12) \end{aligned}$$

where the column vector $\boldsymbol{d}_h(l^n) \triangleq \left[\dot{h}_i(l_i^n)\right]_{i=1}^{N}$, $\boldsymbol{x}'$ denotes the transpose of $\boldsymbol{x}$, and $\boldsymbol{D}(c_i(l_i^n))$ is the $N \times N$ diagonal matrix with diagonal entries $c_i(l_i^n)$ for $i = 1, \ldots, N$.

The surrogate function $\phi(\mu; \mu^n)$ in (10) consists of the sum of a paraboloid (*i.e.* a quadratic form) and the convex penalty term. An algorithm that decreases the function ϕ will also monotonically decrease the objective function if the inequality in (9) holds. The general paraboloidal surrogates (PS) method can be outlined as follows:

```
for each iteration $n$
    determine $c_i(l_i^n)$ and consequently $\phi(\mu; \mu^n)$
    find a $\mu^{n+1} \geq 0$ that decreases (or minimizes) $\phi(\mu; \mu^n)$
end.
```

The key design choices in the general method outlined above are:
1. The different ways of choosing the curvatures $c_i(l_i^n)$'s which would satisfy (9).
2. The algorithm to monotonically decrease $\phi(\mu; \mu^n)$ defined in (10) for $\mu \geq 0$.
Each combination of choices leads to a different algorithm, as we elaborate in the following sections.

A. Maximum Curvature

A natural choice for $c_i(l_i^n)$ is the maximum second derivative in the feasible region for the projections. This "maximum curvature" ensures that (9) holds, which follows from the generalized mean value theorem for twice differentiable functions (page 228, [24]). The feasible region for the projections is $[0, \infty)$ due to the nonnegativity constraint. Hence, the choice

$$c_i(l_i^n) = \max_{l \in [0,\infty)} \{\ddot{h}_i(l)\} \quad (13)$$

is guaranteed to satisfy (9). We show in Appendix A that the closed form expression for $c_i(l_i^n)$ is:

$$c_i(l_i^n) = \left[\left(1 - \frac{y_i r_i}{(b_i + r_i)^2}\right) b_i\right]_+ \quad (14)$$

where $[x]_+ = x$ for $x > 0$ and zero otherwise. Thus, it is trivial to compute the $c_i(l_i^n)$ terms in this case. The choice (14) for the curvature $c_i(l_i^n)$ does not depend on the iteration n, so it is a constant. We refer to this choice as the "maximum curvature" (PS,M,CD).

Having specified the curvatures $\{c_i(l_i^n)\}$, the paraboloidal surrogate $Q(\mu; \mu^n)$ in (12) is now fully determined. Next we need an algorithm that decreases or minimizes the surrogate function $\phi(\mu; \mu^n)$.

B. Algorithms for Minimizing the Paraboloidal Surrogate

In the absence of the nonnegativity constraint, in principle one could minimize the surrogate function $\phi(\mu; \mu^n)$ over μ by zeroing its gradient. The column gradient of $\phi(\mu; \mu^n)$ with respect to μ is given by

$$\begin{aligned} \nabla_\mu \phi(\mu; \mu^n) &= \boldsymbol{A}'\boldsymbol{d}_h(l^n) \\ &\quad + \boldsymbol{A}'\boldsymbol{D}(c_i(l_i^n))\boldsymbol{A}(\mu - \mu^n) + \beta \nabla R(\mu) (15) \end{aligned}$$

If $R(\mu)$ is a quadratic form, *i.e.* $R(\mu) = \frac{1}{2}\mu'\boldsymbol{R}\mu$, then we can analytically zero the gradient, yielding the iteration:

$$\mu^{n+1} = \mu^n - [\boldsymbol{A}'\boldsymbol{D}(c_i(l_i^n))\boldsymbol{A} + \beta\boldsymbol{R}]^{-1}\nabla_\mu \Phi(\mu^n). \quad (16)$$

There are three problems with the above iteration. It does not enforce the nonnegativity constraint, the matrix inverse is impractical to compute exactly, and it is limited to quadratic penalty functions. To overcome these limitations, we instead apply a monotonic coordinate descent iteration to decrease $\phi(\mu; \mu^n)$.

C. Coordinate Descent Applied to the Surrogate Function

To apply coordinate descent to monotonically decrease the surrogate function $\phi(\mu; \mu^n)$, we need a quadratic function that majorizes the function $\phi(\mu; \mu^n)$ at each pixel. We treat the likelihood part and the penalty part separately. Let $\hat{Q}_j^n(\mu_j) \triangleq Q([\hat{\mu}_1, \ldots, \hat{\mu}_{j-1}, \mu_j, \hat{\mu}_{j+1}, \ldots, \hat{\mu}_p]; \mu^n)$ and $\hat{R}_j^o(\mu_j) \triangleq R([\hat{\mu}_1, \ldots, \hat{\mu}_{j-1}, \mu_j, \hat{\mu}_{j+1}, \ldots, \hat{\mu}_p])$, where $\hat{\mu}$ denotes the current estimate of the parameter μ. Then we must select curvatures d_j^n and $\hat{p}_j$ that satisfy the following:

$$\hat{Q}_j^n(\mu_j) = Q(\hat{\mu};\mu^n) + \dot{Q}_j^n(\hat{\mu})(\mu_j - \hat{\mu}_j) + \frac{1}{2}d_j^n(\mu_j - \hat{\mu}_j)^2 \tag{17}$$

$$\begin{aligned}\hat{R}_j^{\mathrm{o}}(\mu_j) \leq & \hat{R}_j(\mu_j) \triangleq R(\hat{\mu}) \\ & +\dot{R}_j(\hat{\mu})(\mu_j - \hat{\mu}_j) + \frac{1}{2}\hat{p}_j(\mu_j - \hat{\mu}_j)^2,\ \forall \mu_j \geq 0,\end{aligned} \tag{18}$$

where $\hat{Q}_j^n(\mu_j)$ and $\hat{R}_j^{\mathrm{o}}(\mu_j)$ are treated as functions of μ_j only. Equality is achievable in (17) since the likelihood surrogate $\hat{Q}_j^n(\mu_j)$ is quadratic. For the penalty part $\hat{R}_j^{\mathrm{o}}(\mu_j)$, we must find a quadratic function $\hat{R}_j(\mu_j)$ that lies above it, by appropriate choice of $\hat{p}_j$ as considered below.

The derivative of the likelihood surrogate parabola at $\hat{\mu}_j$ is (from (11))

$$\dot{Q}_j^n(\hat{\mu}) \triangleq \frac{\partial}{\partial \mu_j}\hat{Q}_j^n(\mu_j)\Big|_{\mu_j=\hat{\mu}_j} = \sum_{i=1}^N a_{ij}\dot{q}_i(\hat{l}_i),$$

where from (8)

$$\dot{q}_i(\hat{l}_i) = \dot{h}_i(l_i^n) + c_i(l_i^n)(\hat{l}_i - l_i^n), \tag{19}$$

where $\hat{l}_i = \sum_{i=1}^N a_{ij}\hat{\mu}_j$, and

$$\dot{h}_i(l) = \left(\frac{y_i}{b_i e^{-l} + r_i} - 1\right) b_i e^{-l}. \tag{20}$$

From (8) and (11), the curvature of the parabola $\hat{Q}_j^n(\mu_j)$ is obviously:

$$d_j^n \triangleq \sum_{i=1}^N a_{ij}^2 c_i(l_i^n). \tag{21}$$

From (5), the derivative of the penalty part at $\hat{\mu}_j$ is

$$\dot{R}_j(\hat{\mu}) \triangleq \frac{\partial}{\partial \mu_j}\hat{R}_j^{\mathrm{o}}(\mu_j)\Big|_{\mu_j=\hat{\mu}_j} = \sum_{k=1}^K c_{kj}\dot{\psi}_k([\boldsymbol{C}\hat{\mu}]_k).$$

We must obtain a parabolic surrogate $\hat{R}_j(\mu_j)$ that satisfies (18). We assume the potential functions $\psi_k(\cdot)$ satisfy the following conditions:

- ψ is symmetric
- ψ is everywhere differentiable (and therefore continuous)
- $\dot{\psi}(t) = d/dt\,\psi(t)$ is non-decreasing (and hence ψ is convex)
- $\omega_\psi(t) \triangleq \dot{\psi}(t)/t$ is non-increasing for $t \geq 0$
- $\omega_\psi(0) = \lim_{t\to 0}\dot{\psi}(t)/t$ is finite and nonzero *i.e.* $0 < \omega_\psi(0) < \infty$.

In the context of robust regression, Huber showed (Lemma 8.3 on page 184 in [19], also [23]) that for potential functions ψ_k that satisfy the conditions above, we can find a parabola $\hat{\psi}_k(t)$ that lies above $\psi_k(t)$, $\forall t \in \mathbb{R}$. This parabola $\hat{\psi}_k(t)$ is tangent to the potential function at the current point $\hat{t}_k \triangleq [\boldsymbol{C}\hat{\mu}]_k$ and at $-\hat{t}_k$ and has the curvature $\omega_{\psi_k}(\hat{t}_k)$ where $\omega_\psi(\cdot)$ was defined above. The surrogate parabola is given by:

$$\hat{\psi}_k(t) = \psi_k(\hat{t}_k) + \dot{\psi}(\hat{t}_k)(t - \hat{t}_k) + \frac{1}{2}\omega_{\psi_k}(\hat{t}_k)(t - \hat{t}_k)^2,$$

and is illustrated in Figure 2. Thus, the following is a surrogate parabola for the penalty part of the objective function:

$$\hat{R}_j(\mu_j) = \sum_{k=1}^K \hat{\psi}_k([\boldsymbol{C}\mu]_k)\Big|_{\mu_m=\hat{\mu}_m,\forall m\neq j}. \tag{22}$$

The curvature of the parabola $\hat{R}_j(\mu_j)$ is:

$$\hat{p}_j \triangleq \sum_{k=1}^K c_{kj}^2\omega_{\psi_k}([\boldsymbol{C}\hat{\mu}]_k). \tag{23}$$

Combining the above surrogate parabolas (17) and (22), the minimization step of the coordinate descent for pixel j is simply:

$$\begin{aligned}\hat{\mu}_j^{\text{new}} &= \arg\min_{\mu_j\geq 0}\hat{Q}_j^n(\mu_j) + \beta\hat{R}_j(\mu_j) \\ &= \left[\hat{\mu}_j - \frac{\dot{Q}_j^n(\hat{\mu}) + \beta\dot{R}_j(\hat{\mu})}{d_j^n + \beta\hat{p}_j}\right]_+.\end{aligned} \tag{24}$$

This is an update that monotonically decreases the value of $\phi(\cdot;\mu^n)$ and consequently the value of $\Phi(\cdot)$. One iteration is finished when all pixels are updated via (24) in a sequential order. We usually update the paraboloidal surrogate function after one iteration of coordinate descent (CD), but one could also perform more than one CD iteration per surrogate. We call this method the Paraboloidal Surrogates Coordinate Descent (PSCD) method.

The PSCD algorithm with the curvatures obtained from (14) is outlined in Table I. In this table, the algorithm flow is given for the general case where $c_i(l_i^n)$ may change at each iteration. However, the curvatures $c_i(l_i^n)$ given in (34) in Table I are constant throughout the iterations. If one uses fixed $c_i(l_i^n)$ values which do not depend on n as in (34), then the d_j^n terms can be precomputed and the algorithm should be reorganized to take this computational advantage into account.

Another computational advantage of curvatures that do not depend on the iterations is as follows. If we define $\tilde{q}_i = \dot{q}_i/\sqrt{c_i}$ and $\tilde{a}_{ij} = a_{ij}\sqrt{c_i}$, then the update in (37) will be simplified to:

$$\tilde{q}_i := \tilde{q}_i + \tilde{a}_{ij}(\mu_j^{\text{work}} - \hat{\mu}_j),$$

which decreases the computation time devoted to back and forward projections per iteration by about 20% for implementations using precomputed system matrices. The equations (36) and (38) should also be modified to use the new variables. We have not implemented this faster version for this paper.

The algorithm in Table I requires roughly double the floating point operations required for one forward and one backprojection per iteration. The gradient of the original log-likelihood with respect to the projections $\left\{\dot{h}_i(l_i^n)\right\}_{i=1}^N$ and the curvature terms $c_i(l_i^n)$ are computed only once per iteration[7]. The gradient of the surrogate paraboloid uses $\dot{q}_i$ terms which can be updated easily as shown in (37) in the algorithm. This implementation does not update the projections $\hat{l}_i$ after each pixel update since they are only needed in the outer loop (33). The projections are computed in (38) after updating all pixels. The update (38) requires $c_i^n > 0$ to work. In (35), we constrain the curvature value to some small value $\epsilon > 0$ (which obviously does not hurt monotonicity) so that (38) can be evaluated for all $i = 1, \ldots, N$. However, ϵ should not be very small since it will cause undesirable numerical precision errors. Storage requirements are also modest for the proposed algorithm. A single copy of the image and four sinograms for $\hat{l}_i$, c_i, $\dot{h}_i$ and $\dot{q}_i$ need to be stored in addition to data vectors y_i, b_i, r_i.

In the following, we discuss the convergence rate of the algorithm, which provides motivation for obtaining better curvatures.

D. Convergence and Convergence Rate

In the absence of background events, *i.e.* when $r_i = 0$, the penalized-likelihood objective Φ is convex and our proposed PSCD algorithm is globally convergent. This is a fairly straightforward consequence of the proof in [25] for convergence of SAGE, so we omit the details.

However when $r_i \neq 0$, little can be said about global convergence due to the possibility that there are multiple minima or a continuous region of minima. Our practical experience suggests that local minima are either unlikely to be present, or are quite far from reasonable starting images, since all experiments with multiple initializations of the algorithm yielded the same limit within numerical precision. The PSCD algorithm is monotonic even with the nonconvex objective function. One can easily show that every fixed point of the algorithm is a stationary point of the objective function and vice versa. Thus, it is comforting to know that the algorithm will converge to a local minimum and will not blow up.

The convergence rate of the proposed algorithm with the "maximum curvature" choice is suboptimal. The curvatures $c_i(l_i^n)$ are too conservative and the paraboloids are unnecessarily narrow. Intuitively, one can deduce that smaller $c_i(l_i^n)$ values will result in faster convergence. The reason for this is that the lower the curvature, the wider the paraboloid and the bigger the step size as can be seen in Fig. 1. To verify this intuition, we analyze the convergence rate of the algorithm. For simplicity, we assume that a quadratic penalty is used in the reconstruction and that the surrogate function $\phi(\mu; \mu^n)$ (10) is minimized exactly.

Let $\hat{\mu}$ be the unconstrained minimizer of the original objective function. At step n, by zeroing the gradient of (10), we get the simple Newton-like update in (16). By Taylor series, for $\mu^n \approx \hat{\mu}$, we can approximate the gradient of the objective function as: $\nabla\Phi(\mu^n) \approx \boldsymbol{H}(\hat{\mu})(\mu^n - \hat{\mu})$, where $\boldsymbol{H}(\hat{\mu})$ is the Hessian of Φ at $\hat{\mu}$. Define $\boldsymbol{N}(c) = \boldsymbol{A'}\boldsymbol{D}(c_i)\boldsymbol{A} + \beta\boldsymbol{R}$, then from (16):

$$\begin{aligned} \mu^{n+1} - \hat{\mu} &\approx \mu^n - \hat{\mu} - [\boldsymbol{N}(c)]^{-1}\boldsymbol{H}(\hat{\mu})(\mu^n - \hat{\mu}) \\ &= (\boldsymbol{I} - [\boldsymbol{N}(c)]^{-1}\boldsymbol{H}(\hat{\mu}))(\mu^n - \hat{\mu}). \end{aligned} \tag{25}$$

This equation describes how the convergence rate of the proposed algorithm is affected by different c_i choices. We use the results from [26] to evaluate the convergence rate. Let $\boldsymbol{N}(c^1)$ and $\boldsymbol{N}(c^2)$ be two matrices corresponding to curvature vectors c^1 and c^2 respectively with $c_i^1 < c_i^2, \forall i$. Then obviously $\boldsymbol{N}(c^2) - \boldsymbol{N}(c^1)$ is positive definite and it follows from Lemma 1 in [26] that the algorithm corresponding to c^1 has a lower root-convergence factor and thus converges faster than the algorithm corresponding to c^2.

Therefore, to optimize the convergence rate, we would like the $c_i(l_i^n)$ values to be as small as possible while still satisfying (9). The optimal choice for the curvatures is the solution to the following constrained optimization problem for each i:

$$c_i(l_i^n) = \min\left\{c \geq 0 : h_i(l) \leq h_i(l_i^n) + \dot{h}_i(l_i^n)(l - l_i^n) + \frac{1}{2}c(l - l_i^n)^2 \quad \forall l \geq\right. \tag{26}$$

This choice yields the fastest convergence rate while still guaranteeing monotonicity. In the following section, we discuss the solution to (26).

E. Optimum Curvature

The curvature that satisfies (26) is not trivial to find for general functions $h_i(\cdot)$. However, the marginal negative log-likelihood functions for each projection (h_i defined in (3)) in transmission tomography have some nice properties. We show the following in Appendix B. The parabola that is:

1. tangent to h_i at the current projection l_i^n, and
2. intersects h_i at $l = 0$,

is guaranteed to lie above $h_i(l)$ $\forall l \geq 0$. This claim is true only when the curvature $c_i(l_i^n)$ of q_i is nonnegative. If the curvature obtained by the above procedure is negative, then we set $c_i(l_i^n)$ to zero[8]. When $c_i(l_i^n) = 0$, the q_i function is the line which is tangent to the h_i curve at the current projection value l_i^n.

The curvature of the parabola described above is[9] :

$$c_i(l_i^n) = \begin{cases} \left[2\dfrac{h_i(0) - h_i(l_i^n) + \dot{h}_i(l_i^n)(l_i^n)}{(l_i^n)^2}\right]_+, & l_i^n > 0, \\ \left[\ddot{h}_i(0)\right]_+, & l_i^n = 0. \end{cases} \tag{27}$$

[7] In contrast to PSCD algorithm, when coordinate descent (CD,NR) is applied to the original objective function, new gradients and curvatures must be computed after each pixel is updated. These computations involve expensive exponentiations and floating point operations which increase the CPU time required for original coordinate descent.

[8] In fact, any nonnegative $c_i(l_i^n)$ will ensure monotonicity, hence the ϵ in (35).

[9] When l_i^n is nonzero but small, due to numerical precision, (27) might turn out to be extremely large during computation. If $c_i(l_i^n) > \left[\ddot{h}_i(0)\right]_+$ (which theoretically should not happen but practically happens due to limited precision), then we set $c_i(l_i^n)$ to be equal to the maximum second derivative $\left[\ddot{h}_i(0)\right]_+$ which eliminates the problem.

We prove in Appendix B that this curvature is the optimum curvature that satisfies (26). The nonnegativity constraint plays an important role in the proof. If nonnegativity is not enforced, the projections at an iteration may go negative and the curvature (27) will not guarantee monotonicity anymore. Fig. 3 illustrates this surrogate parabola with the "optimum curvature" (27). In Table I, the curvature computation in (34) should be changed to (27) to implement PSCD method with the optimum curvature (PS,O,CD).

F. Precomputed Curvature

By relaxing the monotonicity requirement, we can develop faster yet "almost always" monotonic algorithms. We can do this by choosing curvatures $c_i(l_i^n)$ in equation (8) such that $\dot{h}_i(l) = \dot{q}_i(l; l_i^n)$, but $h_i(l) \approx q_i(l; l_i^n)$, rather than requiring the inequality (9). In this case, the paraboloids are quadratic "approximations" to the log-likelihood function at each iteration. A reasonable choice for the curvatures is:

$$c_i = \ddot{h}_i \left(\log \frac{b_i}{y_i - r_i} \right) = (y_i - r_i)^2 / y_i. \tag{28}$$

The value $l_i^{\min} = \log(\frac{b_i}{y_i - r_i})$ is the point that minimizes the h_i function. These curvatures c_i in (28) are close approximations to the second derivative of h_i functions at the projection values $\boldsymbol{A}\hat{\mu}$ where $\hat{\mu}$ is the solution to the penalized-likelihood problem [12]. This is called the "fast denominator" approach in [12], since it features a one-time precomputed approximation to the curvature that is left unchanged during the iterations so that the denominator terms d_j^n (21) can be computed prior to iteration (similar to "maximum curvature" in equation (14)). Computational benefits for iteration independent curvatures as summarized in Section III-C can be utilized. This approximation works well because we usually start the iterations with an FBP image μ^0 where projections $\boldsymbol{A}\mu^0$ are usually close to $l^{\min}$. Nevertheless, unlike with (27) monotonicity is not guaranteed with (28).

The PS method with the curvature (28) yields faster convergence than the other PS algorithms presented above. This method is related to the PWLS image reconstruction method [11], [27], but instead of making a one-time quadratic approximation to the log-likelihood function, the approximation is renewed at each iteration. Although the curvature of the paraboloid remains same, the gradient is changed to match the gradient of the original objective function at the current iterate. The nonnegativity constraint does not play an important role for the derivation, and this curvature may be used for algorithms where nonnegativity is not enforced. We refer to this curvature as "precomputed curvature" (PS,P,CD).

IV. Results

To assess the effectiveness and speed of the new PS algorithms, we present results using real PET data. We acquired a 15-hour blank scan (b_i's) and a 12-min transmission scan data (y_i's) using a Siemens/CTI ECAT EXACT 921 PET scanner with rotating rod transmission sources [28]. The phantom used was an anthropomorphic thorax phantom (Data Spectrum, Chapel Hill, NC). Delayed coincidence sinograms were collected separately in each scan. The blank and transmission scan delayed-coincidence sinograms were shown to be numerically close[10] [21], so we used a time-scaled version of blank scan delayed coincidences as the r_i factors with no other processing. The projection space was 160 radial bins and 192 angles, and the reconstructed images were 128 × 128 with 4.2 mm pixels. The system matrix $\{a_{ij}\}$ was computed by using 3.375 mm wide strip integrals with 3.375 mm spacing, which roughly approximates the system geometry [4].

We performed reconstructions of the phantom by FBP as well as various penalized-likelihood methods. For the penalty term in PL reconstructions, we used the following function:

$$R(\mu) = \frac{1}{2} \sum_{j=1}^{p} \sum_{k \in \mathcal{N}_j} w_{jk} \psi(\mu_j - \mu_k)$$

which is a special case of (5). Here w_{jk} is normally equal to 1 for horizontal and vertical neighbors and $1/\sqrt{2}$ for diagonal neighbors. We used the modified w_{jk}'s described in [29] to achieve more uniform resolution. For the potential function, we used one of the edge-preserving nonquadratic cost functions that was introduced in [30]

$$\psi(x) = \delta^2 \left[|x/\delta| - \log(1 + |x/\delta|) \right].$$

This function acts like a quadratic penalty for small differences in neighboring pixels and is close to absolute value function for differences greater than δ. This nonquadratic function penalizes sharp edges less than quadratic functions. We used $\delta = 0.004$ cm^{-1} chosen by visual inspection. In the final reconstructed image, the horizontal and vertical neighbor differences are less than this δ in homogeneous regions (90% of all differences) which makes the curved part of the penalty effective in those regions. However at edges, for which the differences are greater than δ, this penalty penalizes less than the quadratic one.

The PS algorithms described throughout this section are named using the following format: PS,$\mathcal{C}$,CD. PS stands for paraboloidal surrogates as the general framework for the algorithms and CD stands for coordinate descent applied to the surrogate function. The letter $\mathcal{C}$ in the format represents the curvature type $c_i(l_i^n)$. The types are: "M", "O" and "P" for maximum second derivative curvature (14), optimum curvature (27) and precomputed curvature (28) respectively. The other algorithms we used for comparison in this section are as follows. LBFGS: a constrained Quasi-Newton algorithm [31], CD,P: coordinate descent with precomputed denominators and CD,NR: coordinate descent with Newton-Raphson denominators [11], [12] applied to objective function, GD,P: grouped descent with precomputed denominators [12].

[10] This is due to the fact that singles rate is mostly affected by transmission rods.

Fig. 4 shows images reconstructed by FBP and statistical methods from a 12 minute scan. For comparison, an FBP reconstruction of a 7 hour scan is also shown. Qualitatively, the statistical reconstruction looks better than the FBP image, having less noise and more uniform homogeneous regions. However, our focus here is not the image quality but the amount of time it takes the algorithms to converge to the minimizer image. Nevertheless, improved emission image quality is our ultimate goal. Statistical methods for transmission reconstruction yield better ACFs as compared to conventional methods and result in better emission images. Our goal here is to speed-up and stabilize statistical methods to make them usable routinely in clinic.

Fig. 5 shows that the proposed PSCD algorithms decreased Φ almost as much per iteration as the coordinate descent algorithm applied to Φ directly. This result is important because it shows that the surrogate paraboloids (especially with the optimum curvature) closely approximate the original log-likelihood. More importantly, in Fig. 6 the PSCD algorithms are seen to be much faster than coordinate descent in terms of the actual CPU time[11]. One of the main overhead costs in coordinate descent is the computation of the log-likelihood gradient term after each pixel change [12]. In PSCD algorithm, the gradient of the surrogate function ($\dot{q}_i$'s) can be computed (updated) by a single multiplication (19). The "maximum curvature" method introduced in Section III-A precomputes the denominator terms (d_j^n) for the likelihood part since $c_i(l_i^n)$'s do not depend on the iterations. However, these $c_i(l_i^n)$'s are much larger than the optimal curvatures, so more iterations are required for PS,M,CD than PS,O,CD to converge.

We also compared the PSCD algorithms to the general purpose constrained Quasi-Newton algorithm (LBFGS) [31] in Figures 5 and 6. Although the LBFGS algorithm takes about 25% less CPU time (0.88 seconds) per iteration than PSCD algorithms, it did not converge as fast as the proposed algorithms. This shows that the algorithms such as PSCD which are tailored to our specific problem converge faster than the general purpose Quasi-Newton method.

In Fig. 7, we consider the fastest previous algorithm we know of (*i.e.* GD with 3×3 groups with precomputed denominator [12]) and compare it to the fastest PS algorithms. The PSCD with "precomputed curvatures" (PS,P,CD) (introduced in Section III-F) requires slightly less CPU time than GD,P to converge. Although the PS,P,CD algorithm is not provably monotonic, it is a reasonable approximation and we did not observe any nonmonotonicity in our practical experience when initializing with an FBP image. The monotonic PS,O,CD method is shown in this plot as a baseline for comparison with Fig. 6.

In Fig. 8, we present the results of a transmission scan simulation with zero background counts ($r_i = 0$) and compare the monotonic PSCD algorithm with the functional substitution (FS) method of Zheng *et al.*[15], [16]. The FS algorithm is proven to be monotonic when $r_i = 0$ in which case h_i is convex. However, the FSCD method requires considerably more computation per iteration than both CD and PSCD. The plot in Figure 8 shows that FSCD requires more CPU time than PSCD.

Table II compares the number of iterations and CPU seconds required to minimize the objective function by each method. The CPU times[12], floating point operations and memory accesses (of order M only) per iteration are also tabulated, where M is the number of nonzero entries in system matrix $\boldsymbol{A}$. For comparison purposes, a single forward and backprojection requires about 0.78 CPU seconds. The CD and FS methods are significantly different from our proposed PSCD methods in the following respect. In our methods, the $\dot{q}_i$ terms are kept updated for all i outside the projection loop in (37). In contrast, both CD and FS require $\dot{h}_i$ terms within the backprojection loop, and these change with every pixel update so they must be computed on the fly within the backprojection loop. Thus that backprojection must access $y_i, b_i, r_i, \hat{l}_i$ and the system matrix within the loop, and perform quite a few floating point operations (including the exponentiations) with them. Not only is there inherently more floating point operations required for CD and FS, we suspect that the need to nonsequentially access parts of four sinogram-sized arrays, in addition to the system matrix, significantly degrades the ability of the CPU to pipeline operations. This leads to the dramatic differences in CPU time between PSCD and CD methods.

If a monotonic algorithm is required, the PSCD algorithm with the optimal curvature (PS,O,CD) is the fastest algorithm. The other algorithms are not guaranteed to be monotonic except PSCD with maximum curvature. Although PS,M,CD algorithm consumes less CPU time per iteration, it takes longer to converge since the curvatures result in an unnecessarily narrow surrogate function which causes small step sizes.

Among the nonmonotonic algorithms, another PS method, PSCD with precomputed curvatures (PS,P,CD) is the fastest. It converged in about 15 seconds with the real data used. The CPU time per iteration is the same as PS,M,CD since they both precompute the denominator (d_j^n) terms. Since the curvatures are smaller, this method decreases the objective very rapidly, nevertheless it is not guaranteed to be monotonic. However, as with the CD and GD with precomputed denominators [12], we have never observed any nonmonotonicity in practical applications with iterations started with an FBP image. The FSCD and CD algorithms consume a lot of CPU cycles per iteration and they are much slower than the proposed algorithms. The GD,P algorithm lowers the CPU requirements by decreasing the number of exponentiations, but it does not decrease the objective function as much per iteration as coordinate descent. Thus, it is also slightly slower than

[11] All CPU times are recorded on a DEC 600 5-333 MHz workstation with compiler optimization enabled.

[12] The CPU times are computed on a DEC 600 5-333 MHz. We also compiled the code on a SUN Ultra 2 computer and got similar CPU time ratios for the algorithms. However, the ratios could differ on another architecture or with another compiler due to cache size and pipelining differences.

the PS,P,CD algorithm. This Table shows that PSCD algorithms are preferable for both monotonic and nonmonotonic transmission image reconstructions.

V. Conclusion

We have introduced a new class of algorithms for minimizing penalized-likelihood objective functions for transmission tomography. The algorithms are shown to be monotonic even with the nonconvex objective function. In the nonconvex case, there is no proof that these algorithms will find the global minimum but at least the algorithms will monotonically decrease the objective function towards a local minimum. Practical experience suggests there are rarely multiple minima in this problem, but there is no proof. In the strictly convex case, the proposed algorithms are guaranteed to converge to the global minimum by a proof similar to that in [32].

Convergence is very important for algorithms for any optimization problem, particularly in medical applications. The PSCD algorithm is globally convergent when there are no background counts. Even when there are background counts, the new algorithm is guaranteed to monotonically decrease the objective function making the algorithm stable. Previous algorithms could not guarantee that property without expensive line searches. The robustness, stability and speed of the new algorithm renders it usable in routine clinical studies. Such use should increase the emission image quality as compared to conventional methods which use linear processing and FBP for reconstruction. Further "acceleration" is possible by ordered subsets [33], albeit without guaranteed monotonicity.

The algorithms we introduced are simple, easy to understand, and fast. The simplicity in part is due to the additive form of (2), which is a direct consequence of independent measurements. Since the emission tomography log-likelihood has a very similar form due to independence of measurements, it is possible to apply the paraboloidal surrogates idea to the emission case as well to get faster algorithms [34].

It is possible to parallelize the PS algorithms by applying either grouped descent (GD) [12], [13] algorithm to the surrogate function, or by parallelizing the projection and backprojection operators [35] for each pixel. However, in a serial computer we found that PS method with GD update (PSGD) was not faster than the PSCD algorithm. This is due to the fact that the gradient updates in PSCD algorithm consume much less CPU time than the gradient evaluations in the original CD algorithm which require expensive exponentiations and floating point operations. Hence, grouped descent did not reduce the CPU time per iteration as much in PS method as in the direct method.

In our opinion, the PS,O,CD algorithm supersedes all of our previous methods [4], [18], [12], and is our recommended algorithm for penalized-likelihood transmission tomography. The PS,P,CD algorithm is a faster but nonmonotonic alternative which can be used for noncritical applications. A possible compromise would be to run a few iterations of PS,O,CD algorithm and then fix the curvatures and denominator terms (d_j^n) for the rest of the iterations to save computation time. Alternatively, one can run PS,P,CD algorithm and check the objective function $\Phi(\mu)$ after each iteration to verify that it has decreased. If the objective does not decrease (happens very rarely), then PS,O,CD algorithm can be applied to the previous iterate to ensure monotonicity. For medical purposes, we believe that a monotonic algorithm should be used to reduce the risk of diagnostic errors due to erroneous reconstructions. Fortunately, with the new proposed methods, monotonicity can be assured with only a minor increase in CPU time (17.2 versus 15.1 CPU seconds).

VI. Appendix A

We prove in this appendix that the maximum second derivative of $h_i(l)$ for $l \geq 0$ is given by (14). We drop the subscript i for simplicity.

The form of the h functions is critical in the following. The second and third derivatives of the function h in (3) are:

$$\ddot{h}(l) = \left(1 - \frac{yr}{(be^{-l}+r)^2}\right) be^{-l}, \tag{29}$$

$$h^{(3)}(l) = \left(yr\left[\frac{-be^{-l}+r}{(be^{-l}+r)^3}\right] - 1\right) be^{-l}. \tag{30}$$

We assume $b > 0$, $y \geq 0$, and $r \geq 0$ throughout these appendices. First, we prove two lemmas about properties of these h functions. These lemmas are used for the proofs in Appendix B as well.

Lemma 1: The following are equivalent for $h(l)$ defined in (3):

- (E1) $r = 0$ or $r \geq y$,
- (E2) h is strictly convex,
- (E3) $\dot{h}$ is strictly concave,
- (E4) $\dot{h}$ is monotonically increasing,
- (E5) $\ddot{h}$ is monotonically decreasing.

Proof: Since h is three times continuously differentiable, h is strictly convex if and only if $\ddot{h} > 0$ and $\dot{h}$ is strictly concave if and only if $h^{(3)} < 0$. Clearly, $\ddot{h} > 0$ if and only if $\dot{h}$ is monotonically increasing. So, (E2) $\iff$ (E4). For similar reasons (E3) $\iff$ (E5).

If $r = 0$ or $r \geq y$, then $yr < (be^{-l}+r)^2$, so from (29) $\ddot{h}(l) \geq 0$, $\forall l$. Thus, (E1) $\Rightarrow$ (E2).

To prove (E1) $\Rightarrow$ (E3), from (30), it suffices to show that $(be^{-l}+r)^3 > yr(-be^{-l}+r)$. But this is trivial since $r^3 \geq yr^2$ under the conditions (E1).

To prove the opposite, if $r \neq 0$ and $y > r$, then one can easily show that $\ddot{h}(l)$ and $-h^{(3)}(l)$ can take negative values for sufficiently large l considering (29) and (30). So, (E2) $\Rightarrow$ (E1) and (E3) $\Rightarrow$ (E1). ■

Lemma 2: When $y > r$ and $r \neq 0$, the nonconvex function $\dot{h}$ has the following properties:

- (P1) $\dot{h}$ is continuously differentiable,
- (P2) $\dot{h}$ has exactly one critical point l^*, *i.e.* $\ddot{h}(l^*) = 0$ and l^* is a local maximizer of $\dot{h}(l)$,
- (P3) $\dot{h}$ is strictly concave and monotone increasing for $l < l^*$,
- (P4) $\dot{h}$ is monotone decreasing for $l > l^*$,

- (P5) $\ddot{h}$ has exactly one critical point l^z, *i.e.* $h^{(3)}(l^z) = 0$ and l^z is a local minimizer of $\ddot{h}(l)$.

Proof: (P1) is obvious from (20) and (29).

In the nonconvex case, the equation $\ddot{h}(l) = 0$ has exactly one solution in $\mathbb{R}$, $l^* = \log\left(\frac{b}{\sqrt{yr} - r}\right)$. Since $h^{(3)}(l^*) = -2\frac{(\sqrt{yr} - r)^2}{\sqrt{yr}} < 0$, l^* is a local maximum, proving (P2).

Solutions to the equation $h^{(3)}(l) = 0$ are the roots of a cubic polynomial in the variable $t = be^{-l}$ which has only one real solution. The real root is negative when h is convex resulting no solution for l. But, in the nonconvex case the real root is positive and results in exactly one solution $l^z = \log(b/(a/3 - yr/a - r))$ where $a = \sqrt[3]{27yr^2 + 3\sqrt{3y^3r^3 + 81y^2r^4}}$. So, $\ddot{h}(l)$ has exactly one critical point. We have shown above that $h^{(3)}(l^*) < 0$ and one can easily see that $h^{(3)}(l) \approx \left(\frac{y}{r} - 1\right) be^{-l} > 0$ for large l. Thus $l^z > l^*$ and $h^{(3)}(l) < 0$ for $l < l^z$. So, $\ddot{h}(l)$ is monotonically decreasing for $l < l^z$. Also for $l > l^z$, $h^{(3)}(l) > 0$ and $\ddot{h}(l)$ is monotonically increasing. This proves that l^z is a local minimum for $\ddot{h}(l)$. Hence, (P5) is proven.

To prove (P3), we have to show $h^{(3)}(l) < 0$ and $\ddot{h}(l) > 0$ for $l < l^*$. But, as we found above $l^* < l^z$ and $h^{(3)}(l) < 0$ for $l < l^z$. Also, $\ddot{h}(l) > 0$ for $l < l^*$ since l^* is the only critical point and local maximizer of $\dot{h}$ due to (P2). So, (P3) is also proven.

The function $\ddot{h}(l)$ has exactly one zero crossing l^* from (P2) which is a local maximizer of $\dot{h}$. Then, $\ddot{h}(l)$ has to be always negative for $l > l^*$ proving (P4). To verify, one can easily see that, $\ddot{h}(l) \approx \left(1 - \frac{y}{r}\right) be^{-l} < 0$ for large l values. So $\ddot{h}(l) < 0 \; \forall l > l^*$. ■

The following result follows from (E5) of Lemma 1 for the convex case and from (P5) of Lemma 2 for the nonconvex case.

Corollary 1: The maximum value for $\ddot{h}$ in the region $[0, \infty)$ is achieved at the end points, *i.e.*

$$\begin{aligned} c_i(l_i^n) &= \max_{l \in [0,\infty)} \{\ddot{h}(l)\} \\ &= \max\{\ddot{h}(\infty), \ddot{h}(0)\} \\ &= \left[\ddot{h}(0)\right]_+, \\ &= \left[\left(1 - \frac{yr}{(b+r)^2}\right) b\right]_+. \end{aligned}$$

The result follows since $\lim_{l\to\infty} \ddot{h}(l) = 0$.

VII. Appendix B

In this appendix, we prove that the curvature defined in (27) is the optimum curvature that satisfies (26), which in turn implies from (25) that the choice (27) yields the fastest convergence rate. We first prove two lemmas about strictly concave functions.

Lemma 3: A one-dimensional line $l(x) = ax + b$ can intersect a strictly concave (or strictly convex) function $f(x)$ at most twice.

Proof: Suppose $l(x_i) = f(x_i)$ at points $x_1 < x_2 < x_3$. Then since $f(x)$ is strictly concave, $f(x) > l(x)$ for $x \in (x_1, x_3)$, which contradicts the initial assumption that $f(x_2) = l(x_2)$. ■

Lemma 4: Let $f(x)$ be a one-dimensional strictly concave function, and let $l(x) = ax + b$ be a line that intersects $f(x)$ at the two points $x_1 < x_2$. Then

$$f(x) < l(x) \text{ for } x \in (-\infty, x_1) \cup (x_2, \infty).$$

Proof: Suppose there exists an $x_3 > x_2$ such that $f(x_3) \geq l(x_3)$. Consider the new line $m(x)$ that intersects $f(x)$ at x_1 and x_3. Since $m(x_1) = l(x_1)$ and $m(x_3) = f(x_3) \geq l(x_3)$, it follows from the affine form of $l(x)$ and $m(x)$ that $m(x_2) \geq l(x_2) = f(x_2)$, which contradicts the assumption that $f(x)$ is strictly concave. The case $x_3 < x_1$ is similar. ■

For simplicity in this appendix, we drop the subscript i and the dependence on n for the variables. Let $h(l)$ be the marginal negative log-likelihood function defined in (3) with derivatives presented in (20), (29) and (30) and let $q(l)$ be the parabolic surrogate function defined in (8) with the "optimum curvature" c defined in (27). We use l^c to denote the current projection value l_i^n. The reader may visualize the following proofs by considering the plots of $\dot{h}$ and $\dot{q}$ functions shown in Fig. 3.

We define the difference function by:

$$\delta(l) \overset{\triangle}{=} q(l) - h(l). \tag{31}$$

To show that $q(l) \geq h(l)$ for $l \geq 0$ as required by (9), it suffices to show that $\delta(l) \geq 0$. When $l^c = 0$, it is obvious from Appendix A that $\delta(l) \geq 0$. Thus we focus on the case $l^c > 0$ in the following.

Lemma 5: The following conditions are sufficient to ensure $\delta(l) \geq 0, \; \forall l \in [0, \infty)$.

- (C1) $\delta(0) \geq 0$ and $\delta(l^c) = 0$,
- (C2) $\dot{\delta}(l) \geq 0$ for $l \geq l^c$, and
- (C3) either
  - (C31) $\dot{\delta}(l) < 0, \; \forall l \in [0, l^c)$, or
  - (C32) $\exists l^p \in [0, l^c)$ such that $\dot{\delta}(l) \geq 0$ for $l \in [0, l^p]$ and $\dot{\delta}(l) \leq 0$ for $l \in (l^p, l^c]$.

Proof: Since $\delta(l^c) = 0$

$$\delta(l) = \int_{l^c}^{l} \dot{\delta}(t)dt. \tag{32}$$

- Case $l \geq l^c$. The integrand in (32) is nonnegative due to (C2), so $\delta(l) \geq 0$.
- Case $l \in [0, l^c]$. If (C31) is true, then $\delta(l) = \delta(l^c) - \int_l^{l^c} \dot{\delta}(t)dt \geq \delta(l^c) = 0$.

If (C32) holds and $l \in [0, l^p]$, then $\delta(l) = \delta(0) + \int_0^l \dot{\delta}(t)dt \geq \delta(0) \geq 0$ by (C1). Likewise if (C32) holds and $l \in (l^p, l^c]$, then $\delta(l) = \delta(l^c) - \int_l^{l^c} \dot{\delta}(t)dt \geq \delta(l^c) = 0$ again by (C1). Hence, $\delta(l) \geq 0 \; \forall l \geq 0$ under the above conditions. ■

We now establish the conditions of Lemma 5. (C1) follows directly from the definition (26), so we focus on (C2) and (C3) below. We first treat the case where $h(l)$ is strictly convex.

Lemma 6: If $h(l)$ is a convex function and $\dot{h}(l)$ is concave for $l \geq 0$, then the difference function $\delta(l)$ in (31) with the curvature c defined in (27) satisfies conditions (C2) and (C32) in Lemma 5. Furthermore, $c > 0$.

Proof: It is trivial to show that the conditions (E2) through (E5) of Lemma 1 hold in this case for $l \geq 0$. First we prove $c > 0$. Suppose $c = 0$, so $\dot{q}$ is a constant. Since $\dot{h}(l)$ is increasing by (E4) in Lemma 1 and $\dot{q}(l^c) = \dot{h}(l^c)$, it is obvious that $\dot{q}(l) > \dot{h}(l)$, $\forall l \in [0, l^c)$, so $\delta(0) = -\int_0^{l^c} \dot{\delta}(t)dt < 0$ contradicting (C1). So, $c > 0$ in this case and $\delta(0) = 0$ by design.

To prove (C32), consider $\dot{h}$. The line $\dot{q}$ cannot intersect the strictly concave $\dot{h}$ at more than two points due to Lemma 3. We know that $\dot{\delta}(l^c) = 0$, thus l^c is an intersection point. We have $\delta(0) = 0$ and $\delta(l^c) = 0$ by definition. From mean value theorem, there must be another intersection point $l^p \in [0, l^c)$ such that $\dot{\delta}(l^p) = 0$. We know by Lemma 3 that there cannot be any additional points where $\dot{\delta}(l) = 0$. $\dot{\delta}(l) < 0$ for $l \in (l^p, l^c)$ due to concavity of $\dot{h}$ and $\dot{\delta}(l) > 0$ for $l \in [0, l^p)$ due to Lemma 4. (C32) is proven.

To prove (C2), apply Lemma 4 to the strictly concave function $\dot{h}$ with two points l^p and l^c as the intersection points of the line with the curve. ■

We now consider the realistic nonconvex case.

Lemma 7: Let $h(l)$ be a nonconvex function with its derivative $\dot{h}$ satisfying properties (P1), (P2) and (P3) in Lemma 2. The difference function $\delta(l)$ defined in (31) with the curvature defined in (27) satisfies (C2) and (C3) in Lemma 5.

Proof:

The reader can refer to Fig. 3 for representative plots of h and its first derivative. Note that in Lemma 2, (P2) $\Rightarrow$ (P4) directly.

Consider these two cases where l^* is defined as in Lemma 2:

- Case $l^c < l^*$.

In this case, by (P3) of Lemma 2, l^c is in a concave increasing region. By Lemma 6, (C32) holds as well as the fact that $c > 0$. To prove (C2), we use property (P4), that $\dot{h}$ is a decreasing function for $l > l^*$. So, since $\dot{q}(l^*) > \dot{h}(l^*)$ (as for (C2) in Lemma 6 again) and $c \geq 0$, $\dot{q}(l) > \dot{h}(l), \forall l \geq l^c$.

- Case $l^c \geq l^*$. Since by (C1), $\delta(l^c) - \delta(0) = \int_0^{l^c} \dot{\delta}(t)dt \leq 0$, $\dot{\delta}(l) = \dot{q}(l) - \dot{h}(l)$ cannot always be nonnegative over the interval $[0, l^c)$. So, either $\dot{q}(l) < \dot{h}(l)$, $\forall l \in [0, l^c)$ or $\dot{q}$ intersects $\dot{h}$ ($\dot{\delta}(l) = 0$) at least once in $[0, l^c)$. If the former case occurs, (C31) holds by definition. If the latter case occurs, then we have to prove that (C32) holds, *i.e.* there is no more than one point at which $\dot{q}$ intersects $\dot{h}$ in $[0, l^c)$. Since $c \geq 0$ and $\dot{h}$ is decreasing in the region $l > l^*$, the intersection point(s) $l^p < l^*$. We cannot apply Lemma 3 here to prove that there is no other intersection point, but we can use Lemma 4 to prove it. Assume there is another intersection point. Then, the function $\dot{q} > \dot{h}$ in the concave region outside the interval between two intersection points by Lemma 4 which implies $\dot{\delta}(l^*) > 0$ and $\dot{\delta}(l) > 0$ for $l > l^*$. But this would contradict the fact that $\dot{\delta}(l^c) = 0$. So, (C32) must hold.

In this case, the fact that $c \geq 0$ is enough to prove (C2), since $\dot{h}$ is decreasing in this region.

■

Theorem 1: Let $h(l)$ be a one-dimensional function that satisfies either of the following:

- (H1) $h(l)$ is strictly convex and $\dot{h}(l)$ is strictly concave in the feasible region $l \geq 0$, or
- (H2) $\dot{h}(l)$ satisfies (P1), (P2) and (P3) of Lemma 2.

Then the curvature defined in (27) satisfies the optimality condition in (26).

Proof: For h functions that satisfy conditions (H1), Lemma 6 with Lemma 5 prove that the curvature (27) satisfies (9) for $l^c > 0$. For h functions satisfying conditions in (H2), Lemma 7 and Lemma 5 similarly prove that the curvature (27) satisfies (9) for $l^c > 0$. The rest of the proof applies to both cases (H1) and (H2). For $l^c = 0$, c in (27) is the maximum second derivative in $[0, \infty)$, and (9) is satisfied by mean value theorem as mentioned in Section III-A.

We need to prove that no other nonnegative curvature less than (27) satisfies (9).

Assume $0 \leq c^* < c$, and let

$$q^*(l) = h(l^c) + \dot{h}(l^c)(l - l^c) + \frac{1}{2}c^*(l - l^c)^2.$$

Obviously c^* can exist only when $c > 0$ since $c = 0$ is the minimum curvature we allow. With $c > 0$, it is obvious from (27) that $q(0) = h(0)$. If $l^c > 0$, this clearly implies that $q^*(0) < q(0) = h(0)$ which shows that c^* cannot satisfy (9). If $l^c = 0$, then a curvature $c^* < c$ would force $\dot{q}$ to lie under $\dot{h}$ for some small values of l. That is, $\exists \epsilon > 0$ such that $q(l) < h(l)$ for $\epsilon > l > 0$. Thus c^* does not satisfy (9) even for $l^c = 0$. ■

Corollary 2: The "optimum curvature" defined in (27) using the marginal negative log-likelihood function $h_i(l)$ defined in (3) for the transmission tomography problem satisfies the optimality condition in (26) for $b_i > 0, y_i \geq 0, r_i \geq 0$.

Proof: The function $h_i(l)$ defined in (3) satisfies the conditions (H1) or (H2) of Theorem 1 depending on the values of y_i and r_i as shown in Lemmas 1 and 2. Hence Theorem 1 is directly applicable to the transmission tomography problem. ■

VIII. Acknowledgement

The authors gratefully acknowledge discussions with Gene Gualtieri and Ken Lange, and thank J. Webster Stayman for implementing LBFGS.

References

[1] S C Huang, E J Hoffman, M E Phelps, and D E Kuhl, "Quantitation in positron emission computed tomography: 2 Effects of inaccurate attenuation correction," *J. Comp. Assisted Tomo.*, vol. 3, no. 6, pp. 804–814, Dec. 1979.

[2] H Erdoğan and J A Fessler, "Statistical image reconstruction methods for simultaneous emission/transmission PET scans," in *Proc. IEEE Nuc. Sci. Symp. Med. Im. Conf.*, 1996, vol. 3, pp. 1579–83.

[3] E U Mumcuoglu, R Leahy, S R Cherry, and Z Zhou, "Fast gradient-based methods for Bayesian reconstruction of transmission and emission PET images," *IEEE Tr. Med. Im.*, vol. 13, no. 3, pp. 687–701, Dec. 1994.

[4] J A Fessler, "Hybrid Poisson/polynomial objective functions for tomographic image reconstruction from transmission scans," *IEEE Tr. Im. Proc.*, vol. 4, no. 10, pp. 1439–50, Oct. 1995.

[5] A P Dempster, N M Laird, and D B Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. Royal Stat. Soc. Ser. B*, vol. 39, no. 1, pp. 1–38, 1977.

[6] K Lange and R Carson, "EM reconstruction algorithms for emission and transmission tomography," *J. Comp. Assisted Tomo.*, vol. 8, no. 2, pp. 306–316, Apr. 1984.

[7] J M Ollinger, "Maximum likelihood reconstruction of transmission images in emission computed tomography via the EM algorithm," *IEEE Tr. Med. Im.*, vol. 13, no. 1, pp. 89–101, Mar. 1994.

[8] J T Kent and C Wright, "Some suggestions for transmission tomography based on the EM algorithm," in *Stochastic Models, Statistical Methods, and Algorithms in Im. Analysis*, M Piccioni P Barone, A Frigessi, Ed., vol. 74 of *Lecture Notes in Statistics*, pp. 219–232. Springer, New York, 1992.

[9] J A Browne and T J Holmes, "Developments with maximum likelihood X-ray computed tomography," *IEEE Tr. Med. Im.*, vol. 12, no. 2, pp. 40–52, Mar. 1992.

[10] C A Bouman and K Sauer, "A unified approach to statistical tomography using coordinate descent optimization," *IEEE Tr. Im. Proc.*, vol. 5, no. 3, pp. 480–92, Mar. 1996.

[11] K Sauer and C Bouman, "A local update strategy for iterative reconstruction from projections," *IEEE Tr. Sig. Proc.*, vol. 41, no. 2, pp. 534–548, Feb. 1993.

[12] J A Fessler, E P Ficaro, N H Clinthorne, and K Lange, "Grouped-coordinate ascent algorithms for penalized-likelihood transmission image reconstruction," *IEEE Tr. Med. Im.*, vol. 16, no. 2, pp. 166–75, Apr. 1997.

[13] K D Sauer, S Borman, and C A Bouman, "Parallel computation of sequential pixel updates in statistical tomographic reconstruction," in *Proc. IEEE Intl. Conf. on Image Processing*, 1995, vol. 3, pp. 93–6.

[14] A R De Pierro, "On the relation between the ISRA and the EM algorithm for positron emission tomography," *IEEE Tr. Med. Im.*, vol. 12, no. 2, pp. 328–333, June 1993.

[15] J Zheng, S Saquib, K Sauer, and C Bouman, "Functional substitution methods in optimization for Bayesian tomography," *IEEE Tr. Im. Proc.*, Mar. 1997, Submitted to IEEE Tr. Image Proc.

[16] S Saquib, J Zheng, C A Bouman, and K D Sauer, "Provably convergent coordinate descent in statistical tomographic reconstruction," in *Proc. IEEE Intl. Conf. on Image Processing*, 1996, vol. 2, pp. 741–4.

[17] A R De Pierro, "A modified expectation maximization algorithm for penalized likelihood estimation in emission tomography," *IEEE Tr. Med. Im.*, vol. 14, no. 1, pp. 132–137, Mar. 1995.

[18] K Lange and J A Fessler, "Globally convergent algorithms for maximum a posteriori transmission tomography," *IEEE Tr. Im. Proc.*, vol. 4, no. 10, pp. 1430–8, Oct. 1995.

[19] P J Huber, *Robust statistics*, Wiley, New York, 1981.

[20] E Ü Mumcuoğlu, R M Leahy, and S R Cherry, "Bayesian reconstruction of PET images: methodology and performance analysis," *Phys. Med. Biol.*, vol. 41, no. 9, pp. 1777–1807, Sept. 1996.

[21] M Yavuz and J A Fessler, "New statistical models for randoms-precorrected PET scans," in *Information Processing in Medical Im.*, J. Duncan and G. Gindi, Eds., vol. 1230 of *Lecture Notes in Computer Science*, pp. 190–203. Springer Verlag, Berlin, 1997.

[22] S R Meikle, M Dahlbom, S R Cherry, and A Chatziioannou, "Attenuation correction in whole body PET," *J. Nuc. Med. (Abs. Book)*, vol. 33, no. 5, pp. 862, May 1992.

[23] J A Fessler, "Grouped coordinate descent algorithms for robust edge-preserving image restoration," in *Proc. SPIE 3071, Im. Recon. and Restor. II*, 1997, pp. 184–94.

[24] Philippe G Ciarlet, *Introduction to numerical linear algebra and optimisation*, Cambridge, Cambridge, 1982.

[25] J A Fessler and A O Hero, "Penalized maximum-likelihood image reconstruction using space-alternating generalized EM algorithms," *IEEE Tr. Im. Proc.*, vol. 4, no. 10, pp. 1417–29, Oct. 1995.

[26] J A Fessler, N H Clinthorne, and W L Rogers, "On complete data spaces for PET reconstruction algorithms," *IEEE Tr. Nuc. Sci.*, vol. 40, no. 4, pp. 1055–61, Aug. 1993.

[27] J A Fessler, "Penalized weighted least-squares image reconstruction for positron emission tomography," *IEEE Tr. Med. Im.*, vol. 13, no. 2, pp. 290–300, June 1994.

[28] K Wienhard, L Eriksson, S Grootoonk, M Casey, U Pietrzyk, and W D Heiss, "Performance evaluation of a new generation positron scanner ECAT EXACT," *J. Comp. Assisted Tomo.*, vol. 16, no. 5, pp. 804–813, Sept. 1992.

[29] J A Fessler and W L Rogers, "Spatial resolution properties of penalized-likelihood image reconstruction methods: Space-invariant tomographs," *IEEE Tr. Im. Proc.*, vol. 5, no. 9, pp. 1346–58, Sept. 1996.

[30] K Lange, "Convergence of EM image reconstruction algorithms with Gibbs smoothing," *IEEE Tr. Med. Im.*, vol. 9, no. 4, pp. 439–446, Dec. 1990, Corrections, June 1991.

[31] C Zhu, R H Byrd, P Lu, and J Nocedal, "Algorithm 778: L-BFGS-B: Fortran subroutines for large-scale bound-constrained optimization," *ACM Tr. Math. Software*, vol. 23, no. 4, pp. 550–60, Dec. 1997.

[32] J A Fessler and A O Hero, "Space-alternating generalized expectation-maximization algorithm," *IEEE Tr. Sig. Proc.*, vol. 42, no. 10, pp. 2664–77, Oct. 1994.

[33] H Erdoğan, G Gualtieri, and J A Fessler, "An ordered subsets algorithm for transmission tomography," in *Proc. IEEE Nuc. Sci. Symp. Med. Im. Conf.*, 1998, Inadvertently omitted from proceedings. Available from web page.

[34] J A Fessler and H Erdoğan, "A paraboloidal surrogates algorithm for convergent penalized-likelihood emission image reconstruction," in *Proc. IEEE Nuc. Sci. Symp. Med. Im. Conf.*, 1998, vol. 2, pp. 1132–5.

[35] W Niethammer, "A note on the implementation of the successive overrelaxation method for linear complementarity problems," *Numerical Algorithms*, vol. 4, no. 1, pp. 197–200, Jan. 1993.

```
Initialize: $\hat{\mu} = \mathrm{FBP}\{\log(b_i/(y_i - r_i))\}_{i=1}^{N}$ and $\hat{l}_i = \sum_{j=1}^{p} a_{ij}\hat{\mu}_j, \quad \forall i = 1, \ldots, N$
for each iteration $n = 0, \ldots, \mathrm{Niter} - 1$
        $\dot{q}_i = \dot{h}_i = \left( \frac{y_i}{b_i e^{-\hat{l}_i} + r_i} - 1 \right) b_i e^{-\hat{l}_i}, \text{ for } i = 1, \ldots, N \qquad (33)$
        $c_i = \max_{l \geq 0} \ddot{h}_i(l) = \left[ \left( 1 - \frac{y_i r_i}{(b_i + r_i)^2} \right) b_i \right]_+, \text{ for } i = 1, \ldots, N \qquad (34)$
        $c_i := \begin{cases} c_i, & c_i > \epsilon \\ \epsilon, & c_i \leq \epsilon \end{cases} \qquad (35)$
    repeat one or more times
        for $j = 1, \ldots, p$
            $\dot{Q}_j = \sum_{i=1}^{N} a_{ij}\dot{q}_i, \quad d_j = \sum_{i=1}^{N} a_{ij}^2 c_i \qquad (36)$
            $\mu_j^{\mathrm{old}} = \hat{\mu}_j$
            for a couple sub-iterations
                $\hat{\mu}_j := \left[ \hat{\mu}_j - \frac{\dot{Q}_j + d_j(\hat{\mu}_j - \mu_j^{\mathrm{old}}) + \beta \sum_{k=1}^{K} c_{kj} \dot{\psi}\left([\boldsymbol{C}\hat{\mu}]_k\right)}{d_j + \beta \sum_{k=1}^{K} c_{kj}^2 \omega_{\psi_k}\left([\boldsymbol{C}\hat{\mu}]_k\right)} \right]_+$
            end
            $\dot{q}_i := \dot{q}_i + a_{ij} c_i (\hat{\mu}_j - \mu_j^{\mathrm{old}}) \ \forall i \text{ s.t. } a_{ij} \neq 0 \qquad (37)$
        end
    end
        $\hat{l}_i := \hat{l}_i + \frac{\dot{q}_i - \dot{h}_i}{c_i}, \text{ for } i = 1, \ldots, N \qquad (38)$
end
```

TABLE I

Algorithm outline for a paraboloidal surrogates algorithm with coordinate descent (PSCD). The curvature choice shown here is the maximum second derivative.

Real data, $r_i \neq 0$	monotonic		nonmonotonic				
methods	PS,M,CD	PS,O,CD	PS,P,CD	GD,P,3x3	CD,P	CD,NR	FSCD
iters for convergence	18	12	11	14	11	11	11
CPU s for convergence	23.3	17.4	15.1	18.1	44.3	52.3	56.2
CPU s per iteration	1.2	1.3	1.2	1.1	3.8	4.6	4.9
exponentiations per iteration	0	0	0	0	M	M	2M
add/subts per iteration	2M	3M	2M	2M	4M	6M	7M
mult/divs per iteration	3M	5M	3M	2M	5M	11M	10M
nonsequential accesses per backprojection	M	2M	M	M	4M	4M	4M
nonsequential accesses per forward projection	2M	2M	2M	M	M	M	M
system matrix accesses per iteration	2M	2M	2M	2M	2M	2M	2M

TABLE II

Comparison of CPU times, number of iterations to converge, floating point operations and memory accesses for the PS algorithms versus CD, GD and FS methods. Convergence in this table means $\Phi(\mu^0) - \Phi(\mu^n) > 0.999 \left[\Phi(\mu^0) - \Phi(\mu^*)\right]$ where $\Phi(\mu^*)$ is the smallest objective value obtained in 30 iterations among all the methods. The floating point operations and memory accesses only in the order of M are shown for each method.

List of Figures

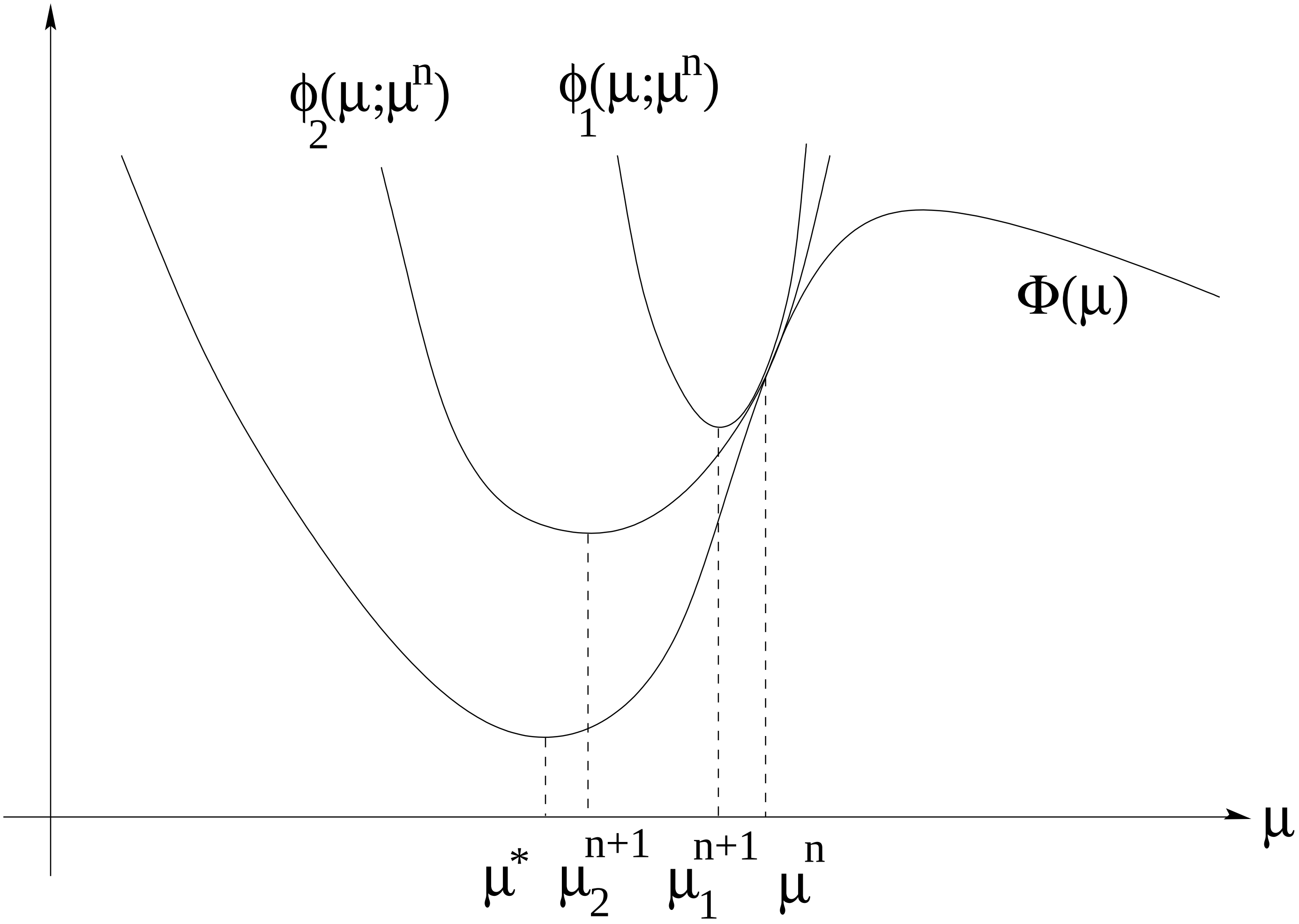

Fig. 1. One-dimensional illustration of the optimization transfer principle. Instead of minimizing $\Phi(\mu)$, we minimize the surrogate function $\phi(\mu;\mu^n)$ at the nth iteration. Here, the surrogate function ϕ_2 has a smaller curvature and is wider than ϕ_1, thus it has a bigger step size and hence faster convergence rate to the local minimum μ^*.

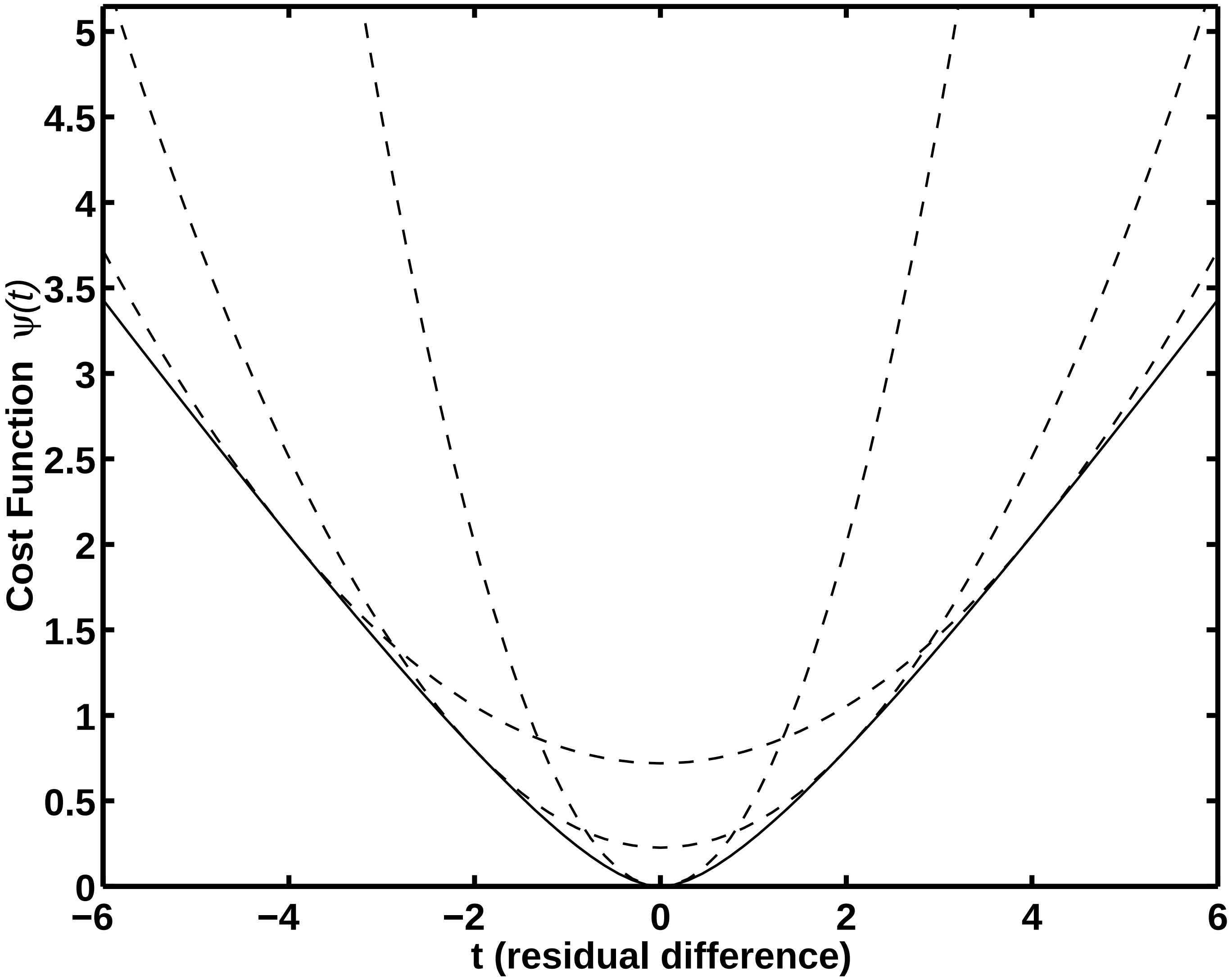

Fig. 2. Illustration of the tangent parabolas lying above a potential function.

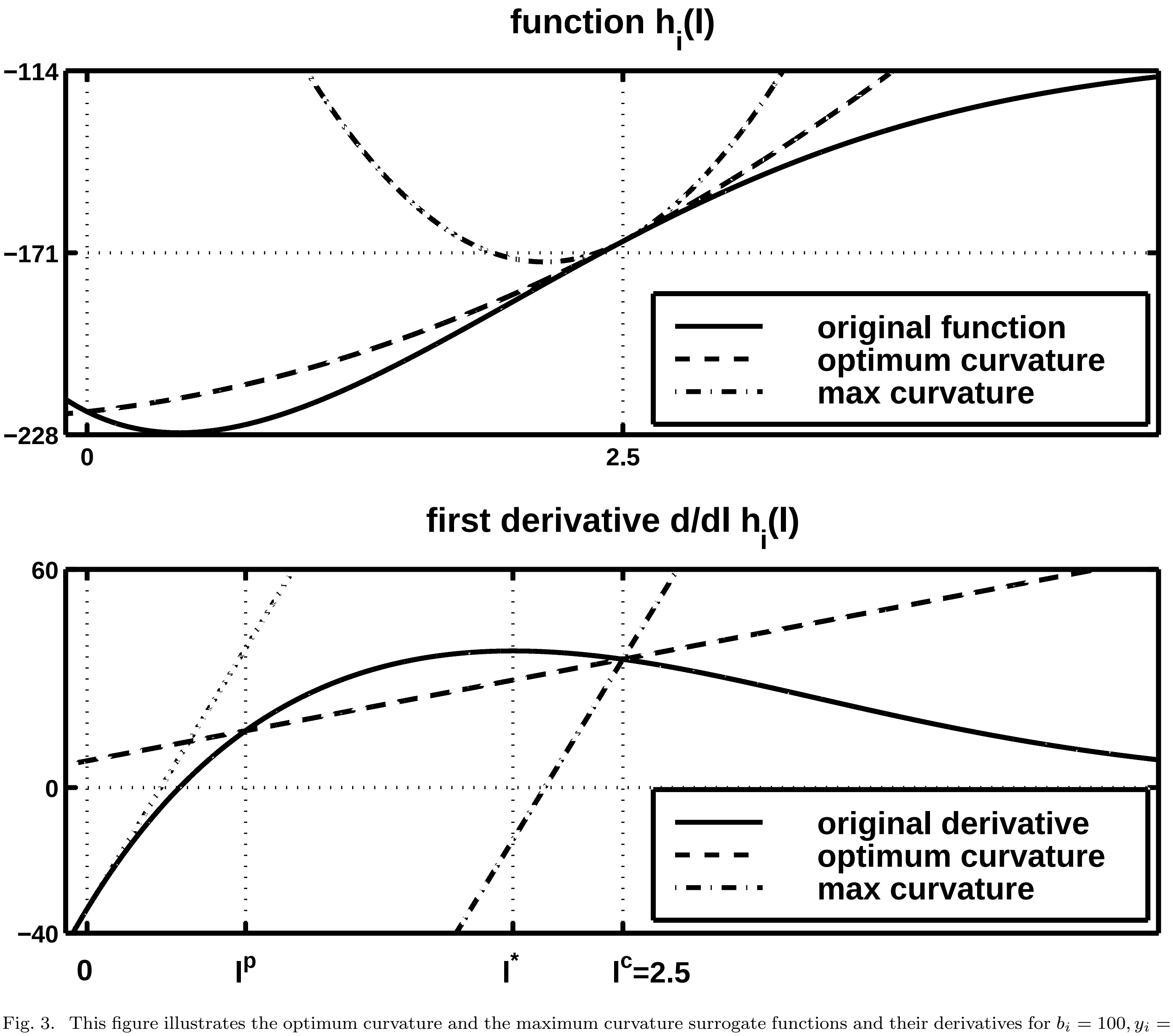

Fig. 3. This figure illustrates the optimum curvature and the maximum curvature surrogate functions and their derivatives for $b_i = 100, y_i = 70, r_i = 5$, and $l_i^n = 2.5$.

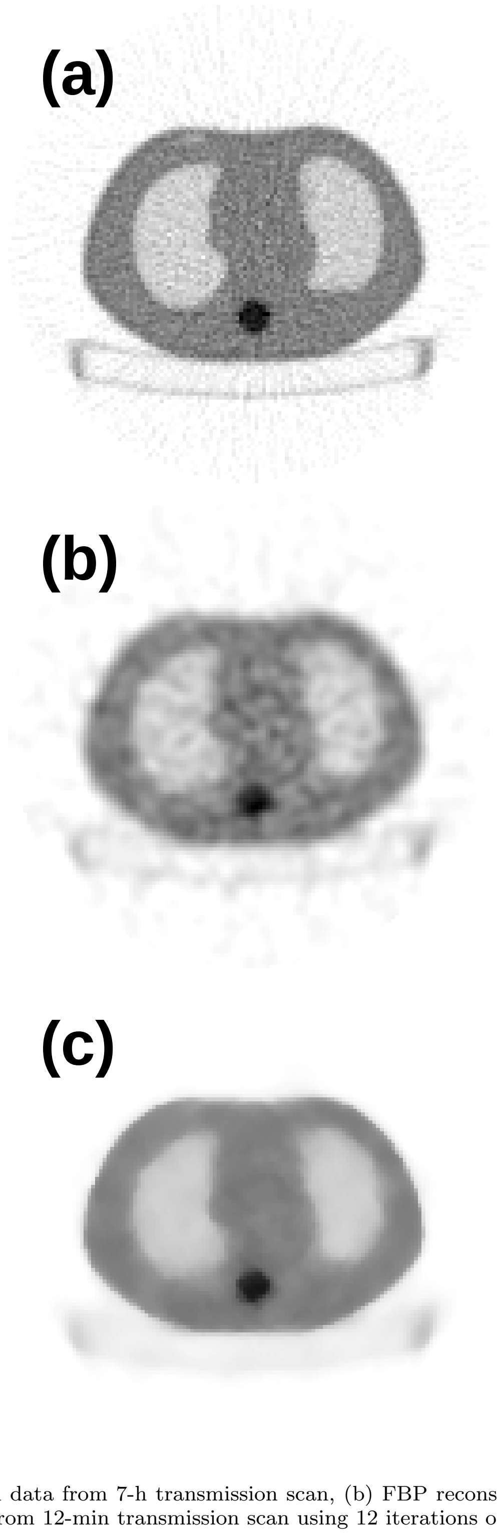

Fig. 4. (a) FBP reconstruction of phantom data from 7-h transmission scan, (b) FBP reconstruction from 12-min transmission scan, and (c) Penalized-likelihood reconstruction from 12-min transmission scan using 12 iterations of the “optimum curvature” PSCD algorithm.

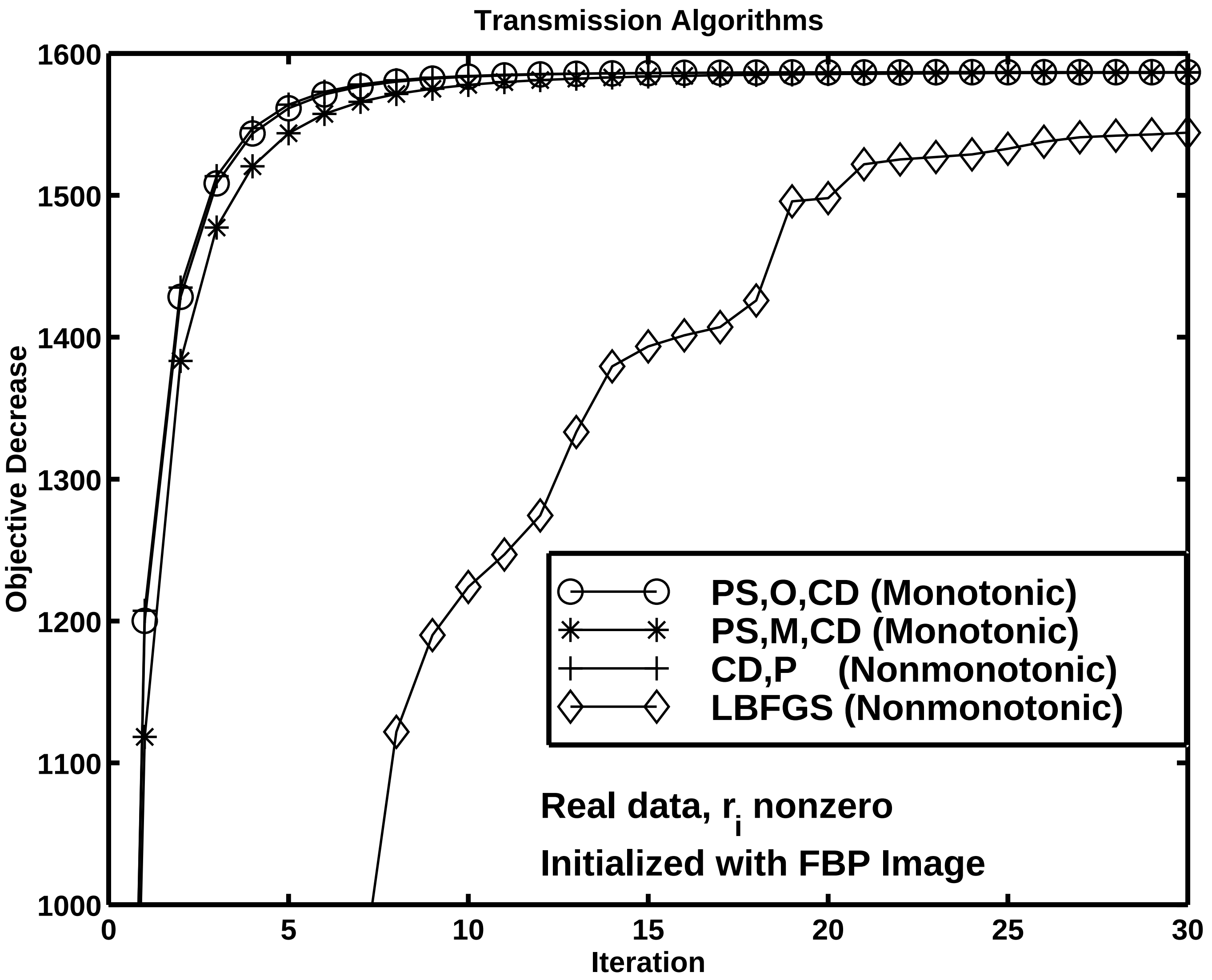

Fig. 5. Comparison of objective function decrease $\Phi(\mu^0) - \Phi(\mu^n)$ versus iteration number n of monotonic PS methods with coordinate descent and LBFGS methods for real phantom data.

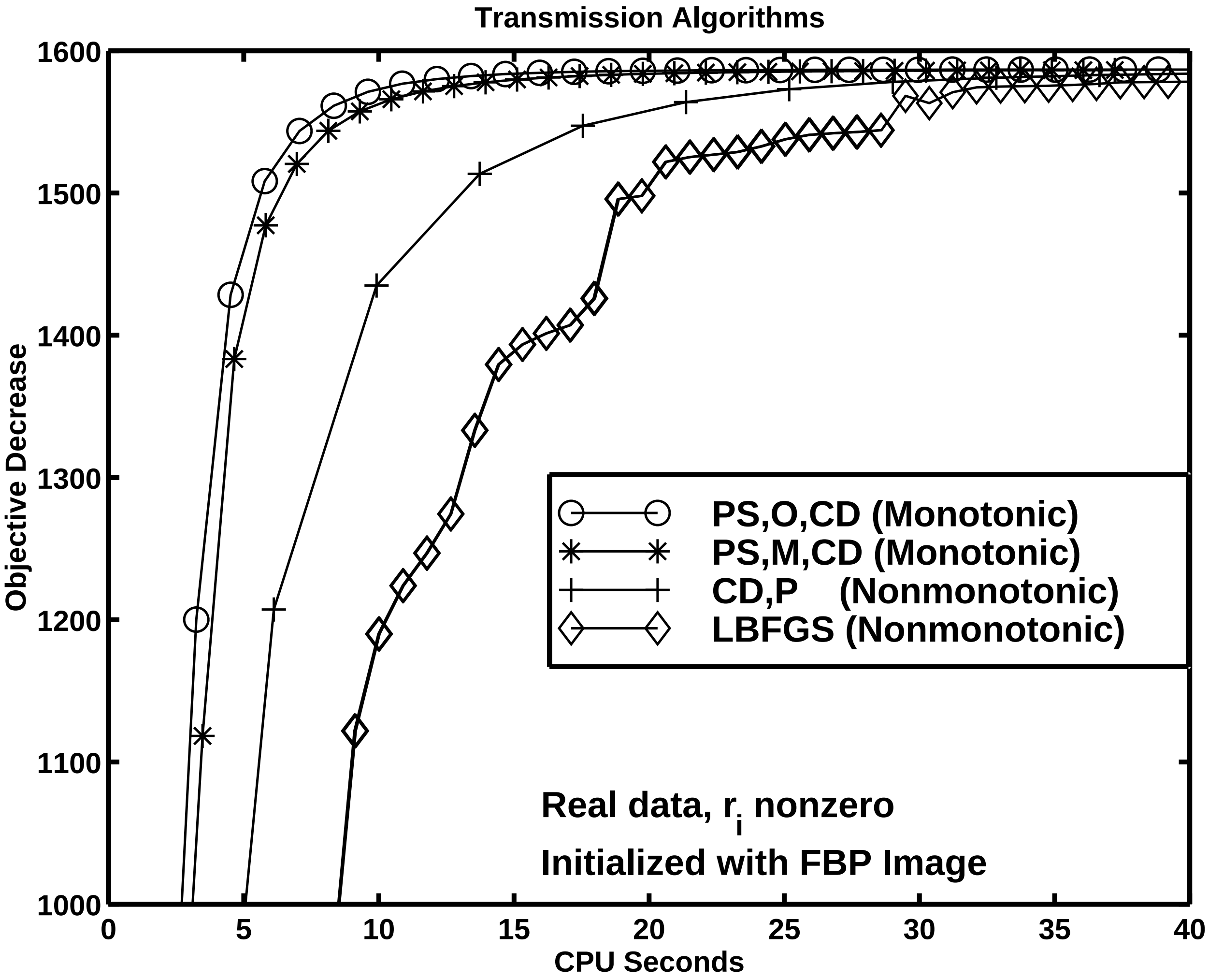

Fig. 6. Same as Figure 5, but x-axis is CPU seconds on a DEC AlphaStation 600 5-333 MHz.

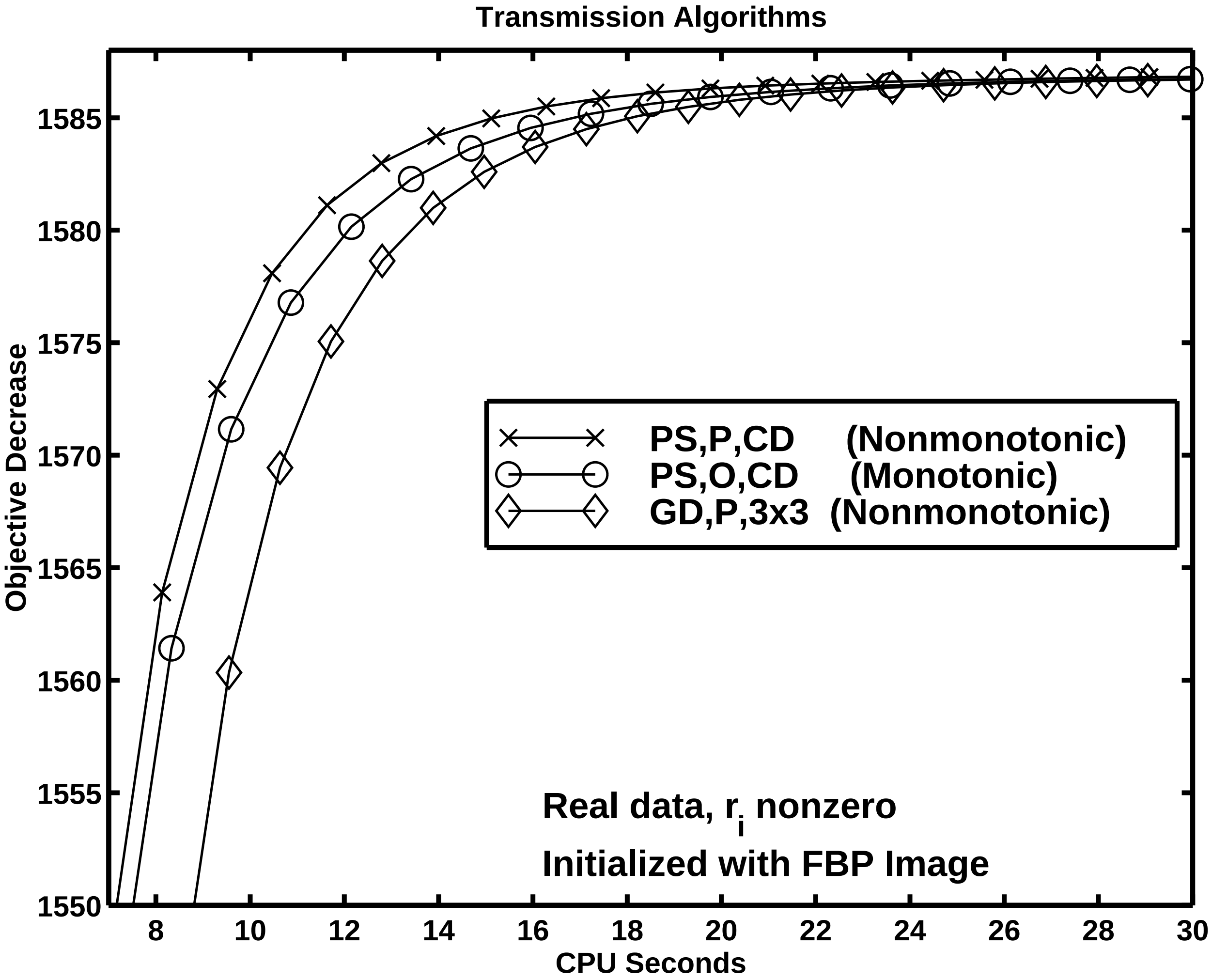

Fig. 7. Comparison of the speed of the proposed PS algorithms with the fastest algorithm that was introduced before: grouped coordinate descent with 3x3 groups.

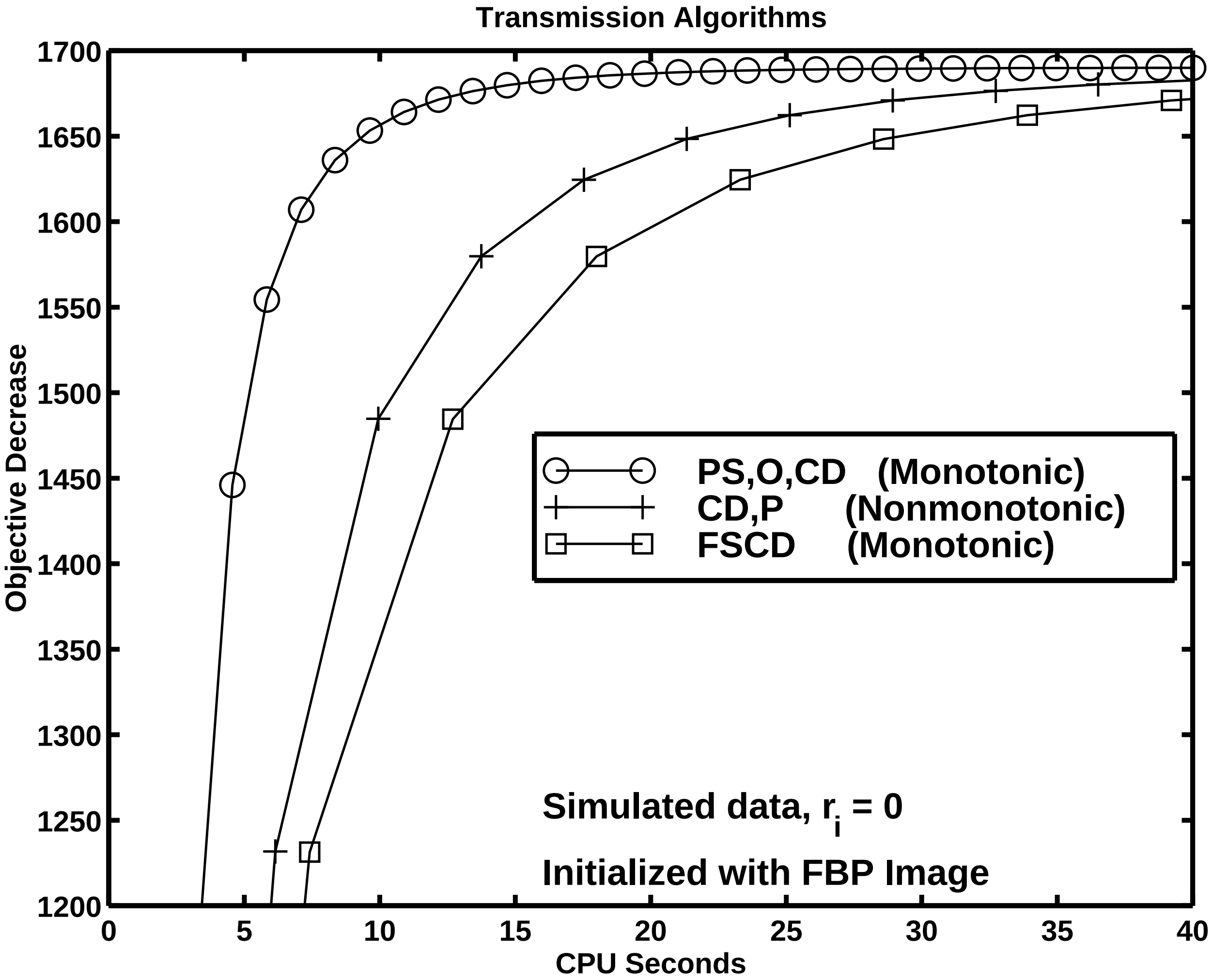

Fig. 8. Comparison of objective function decrease $\Phi(\mu^0) - \Phi(\mu^n)$ versus CPU time of monotonic PS and FS methods with coordinate descent. Note $r_i = 0$ in this simulation.