Convergent Incremental Optimization Transfer Algorithms: Application to Tomography

Sangtae Ahn, *Student Member, IEEE*, Jeffrey A. Fessler, *Senior Member, IEEE*, Doron Blatt, *Student Member, IEEE*, and Alfred O. Hero, *Fellow, IEEE*

Abstract

No *convergent* ordered subsets (OS) type image reconstruction algorithms for transmission tomography have been proposed to date. In contrast, in emission tomography, there are two known families of convergent OS algorithms: methods that use relaxation parameters (Ahn and Fessler, 2004), and methods based on the incremental expectation maximization (EM) approach (Hsiao *et al.*, 2002). This paper generalizes the incremental EM approach by introducing a general framework that we call "incremental optimization transfer." Like incremental EM methods, the proposed algorithms accelerate convergence speeds and ensure global convergence (to a stationary point) under mild regularity conditions without requiring inconvenient relaxation parameters. The general optimization transfer framework enables the use of a very broad family of non-EM surrogate functions. In particular, this paper provides the first *convergent* OS-type algorithm for transmission tomography. The general approach is applicable to both monoenergetic and polyenergetic transmission scans as well as to other image reconstruction problems. We propose a particular incremental optimization transfer method for (nonconcave) penalized-likelihood (PL) transmission image reconstruction by using separable paraboloidal surrogates (SPS). Results show that the new "transmission incremental optimization transfer (TRIOT)" algorithm is faster than nonincremental ordinary SPS and even OS-SPS yet is convergent.

Index Terms

Statistical image reconstruction, maximum likelihood estimation, penalized-likelihood estimation, incremental optimization transfer, transmission tomography

This work was supported in part by the National Institutes of Health (NIH) under Grants CA-60711 and CA-87634, by the Department of Energy (DOE) under Grant DE-FG02-87ER60561, and by a Rackham Predoctoral Fellowship.

The authors are with the Electrical Engineering and Computer Science Department, University of Michigan, Ann Arbor, MI 48109-2122 USA (e-mail: sangtaea@umich.edu, fessler@umich.edu, dblatt@eecs.umich.edu, hero@eecs.umich.edu).

I. Introduction

Ordered subsets (OS) algorithms, also known as block iterative or incremental gradient methods, have been very popular in the medical imaging community for tomographic image reconstruction due to their fast convergence rates [1]–[10]. The incremental gradient type algorithms are also found in convex programming [11]–[14]. The ordered subsets (or incremental) idea is to perform the update iteration incrementally by sequentially (or sometimes randomly [12], [13]) using a subset of the data. Row-action methods [15] including algebraic reconstruction techniques (ART) [16], [17] can also be viewed as OS type algorithms in which each subset corresponds to a single measurement.

The OS algorithms apply successfully to problems where an objective function of interest is a sum of a large number of component functions. Because of the assumed statistical independence of tomographic data, such sums arise in statistical tomographic reconstruction problems including penalized-likelihood (PL) [equivalently, maximum *a posteriori* (MAP)] or maximum likelihood (ML) reconstruction. Typically, the OS methods decompose the sum of component functions into several subobjective functions, each corresponding to a subset of the projection views, and then update the image estimate by using, in a specified cyclic order, the gradient of a subobjective function as an approximate gradient of the objective function.

If the subset gradients are suitably balanced, then the gradient approximation can be quite reasonable when the iterates are far from a maximizer. Thus OS methods initially accelerate convergence in the sense that less computation is required to achieve nearly the same level of objective increase as with non-OS methods. However, ordinary (unrelaxed) OS algorithms such as OS-EM [1], RBI-EM [3], and OS-SPS (or OSTR in a context of transmission tomography) [6] generally do not converge to an optimal solution but rather approach a suboptimal limit cycle that consists of as many points as there are subsets. In fact, due to their subset-dependent scaling (or preconditioning) matrices [8], OS-EM and RBI-EM in their original forms [1], [3] usually do not converge to the optimal point even if relaxed.

Convergence to an optimal solution is important for any algorithm for optimization problems, particularly in medical applications where reliability and stability are essential. For PL (or MAP) reconstruction, the convergence issue is perhaps more critical than for ML for which one often does not run algorithms to convergence. For example, the image shown in Fig. 5(c), which corresponds to one point of a limit cycle generated by an OS algorithm, looks noticeably different from the PL solution image shown in Fig. 5(b) (see Section IV for details). It is desirable to achieve both fast initial convergence rates (typical of OS algorithms) and global convergence. There have been three known families of convergent incremental

(or OS type) algorithms: methods that use relaxation parameters, methods based on the incremental EM approach, and incremental aggregated gradient (IAG) methods.

Relaxation parameters are used widely to render OS algorithms convergent [2], [4], [5], [7]–[9], [11]–[13], [18]–[20]. Suitably relaxed algorithms can be shown to converge to an optimal solution under certain regularity conditions[1] [8]. However, since relaxation parameters should be scheduled to converge to zero for global convergence, relaxed OS algorithms have slow asymptotic convergence rates. Also, inappropriately chosen (*e.g.*, too rapidly decreasing) relaxation parameters could make initial convergence rates even worse than those of non-OS algorithms. On the other hand, overly large relaxation parameters can lead to unstable or divergent behavior. Finding good relaxation parameters (in terms of convergence rates) may require some experimentation and trial-and-error; as a rule of thumb, for properly scaled OS algorithms such as modified BSREM and relaxed OS-SPS, one should initialize the relaxation parameter near unity and decrease it gradually as convergence to a limit cycle nears [8]. One may optimize a few initial relaxation parameters by training when a training set is available for a particular task [2], [17]. Or one could use the dynamic stepsize rule in [12], [13], but that method needs to compute the objective value at every update, which is computationally expensive in tomographic reconstruction problems. Alternatively, to achieve convergence, one could decrease the number of subsets as iterations proceed or could use hybrid methods that combine OS and non-OS algorithms [22]. However, the schedule for decreasing the number of subsets or the parameters for the hybrid algorithms are as inconvenient to determine as relaxation parameters for relaxed OS algorithms.

Incremental EM algorithms do not require user-specified relaxation parameters [23]. They are convergent yet faster than ordinary EM algorithms although slower initially than nonconvergent OS-EM type algorithms [24]–[26]. Such incremental EM algorithms have been applied to emission tomography [10], [24], [26], [27].

Recently, Blatt *et al.* proposed a convergent incremental gradient method, called incremental aggregated gradient (IAG), that does not require relaxation parameters [28]. The IAG method computes a single subset gradient for each update but aggregates it with the stored subset gradients that were computed in previous iterations. The use of the aggregated gradient to approximate the full gradient of the objective function leads to convergence. Similarly, as discussed below, the use of the sum of surrogate functions (rather than a single surrogate function) to approximate a minorizing function yields convergent algorithms.

[1]One of these conditions being the (strict) concavity of the objective function excludes the nonconcave transmission problem [21].

In this paper we generalize the incremental EM algorithms by introducing an approach called "incremental optimization transfer"; this is akin to the generalization of the EM algorithms [29] by the optimization transfer principles [30]. In fact, the broad family of "incremental optimization transfer algorithms" includes the ordinary optimization transfer algorithms (*e.g.*, EM), also referred to as MM (minorize-maximize or majorize-minimize) algorithms in [31], as a special case where the objective function consists of only one subobjective function.

In the incremental optimization transfer approach, for *each* subobjective function, we define an augmented vector that has the same size as the parameter vector to be estimated. The augmented vector plays a role as an expansion point at which a minorizing surrogate function is defined for the subobjective function (see Section II for details). The sum of the surrogate functions defines an augmented objective that is a function of the parameter vector and the augmented vectors. With surrogate functions satisfying usual minorization conditions [21], [30], a solution to the problem of maximizing the original objective can be found by maximizing the augmented objective instead. Applying a block coordinate ascent approach to the augmented problem leads to a new class of "incremental optimization transfer algorithms." By using the block coordinate ascent approach, incremental optimization transfer algorithms are monotonic in the augmented objective though not necessarily in the original objective; nevertheless, global convergence is ensured under mild regularity conditions. Incremental optimization transfer algorithms show faster convergence rates than their nonincremental counterparts like EM [23], [24], [26].

Incremental optimization transfer is a general framework in which one can develop many different algorithms by using a very broad family of application-dependent surrogate functions. These methods are particularly useful for large-scale problems where the objective function is expressed as a sum of several subobjective functions. In this paper, we focus on PL image reconstruction for transmission tomography, which is a challenging nonconcave maximization problem. We propose a particular incremental optimization transfer algorithm that uses separable paraboloidal surrogates (SPS) [6]. Such quadratic surrogates simplify the maximization. In contrast, the standard EM surrogates for transmission tomography do not have a closed-form maximizer in the "M-step" [32].

The proposed "transmission incremental optimization transfer (TRIOT)" algorithm is convergent yet converges faster than ordinary SPS [6]; it can be further accelerated by the enhancement method in [33] or by initializing through a few iterations of OS-SPS (see Section III for details). It is parallelizable, and the nonnegativity constraint is naturally enforced. In addition, it is easily implemented for system models that use factored system matrices [34], [35] whereas pixel-grouped coordinate ascent based methods require column access of the system matrix [36]–[39].

Section II describes the incremental optimization transfer algorithms in a general framework and discusses their convergence properties. Section III develops incremental optimization transfer algorithms for transmission tomography, and addresses acceleration methods. Section IV provides simulation and real PET data results, and Section V gives conclusions.

II. Incremental Optimization Transfer

A. Incremental Optimization Transfer Algorithms

Most objective functions of interest in image reconstruction can be expressed as a sum of subobjective functions:[2]

$$\Phi(\boldsymbol{x}) = \sum_{m=1}^{M} \Phi_m(\boldsymbol{x}), \tag{1}$$

where $\Phi_m : \mathcal{X} \subset \mathbb{R}^p \to \mathbb{R}$ is a continuously differentiable function whose domain $\mathcal{X}$ is a nonempty, convex and closed set. We consider the following optimization problem:

$$\text{maximize } \Phi(\boldsymbol{x}) \text{ subject to } \boldsymbol{x} \in \mathcal{X}. \tag{2}$$

Since usually there exists no closed-form solution to the above problem, one must apply iterative algorithms. Assume that for each subobjective function Φ_m, we find a surrogate function $\phi_m : \mathcal{X}^2 \subset \mathbb{R}^p \times \mathbb{R}^p \to \mathbb{R}$ that is easier to maximize than Φ_m and that satisfies the following usual minorization conditions [21], [41]:

$$\begin{aligned} \phi_m(\boldsymbol{x};\boldsymbol{x}) &= \Phi_m(\boldsymbol{x}), \quad \forall \boldsymbol{x} \in \mathcal{X} \\ \phi_m(\boldsymbol{x};\bar{\boldsymbol{x}}) &\le \Phi_m(\boldsymbol{x}), \quad \forall \boldsymbol{x}, \bar{\boldsymbol{x}} \in \mathcal{X}, \end{aligned} \tag{3}$$

where $\mathcal{X}^n$ denotes the n-ary Cartesian product over the set $\mathcal{X}$. It follows from the above conditions that

$$\Phi_m(\boldsymbol{x}) - \Phi_m(\bar{\boldsymbol{x}}) \ge \phi_m(\boldsymbol{x};\bar{\boldsymbol{x}}) - \phi_m(\bar{\boldsymbol{x}};\bar{\boldsymbol{x}}), \quad \forall \boldsymbol{x}, \bar{\boldsymbol{x}} \in \mathcal{X}.$$

In other words, choosing $\boldsymbol{x}$ such that $\phi_m(\boldsymbol{x};\bar{\boldsymbol{x}}) \ge \phi_m(\bar{\boldsymbol{x}};\bar{\boldsymbol{x}})$ ensures that $\Phi_m(\boldsymbol{x}) \ge \Phi_m(\bar{\boldsymbol{x}})$. Define the following "divergence" function:

$$\mathsf{D}_m(\boldsymbol{x} \,\|\, \bar{\boldsymbol{x}}) \stackrel{\triangle}{=} \Phi_m(\boldsymbol{x}) - \phi_m(\boldsymbol{x};\bar{\boldsymbol{x}}).$$

[2]Such functions are said to be *additive-separable* in [11]; and to be *partially separable* [40] when each $\Phi_m(\boldsymbol{x})$ is a function of fewer components of $\boldsymbol{x} \in \mathbb{R}^p$ than p.

Then by (3), we have the following properties:[3]

$$\mathsf{D}_m(\boldsymbol{x} \,\|\, \bar{\boldsymbol{x}}) \geq 0 \text{ and } \mathsf{D}_m(\boldsymbol{x} \,\|\, \boldsymbol{x}) = 0. \tag{4}$$

Now we define the following "augmented" objective function:

$$\begin{aligned} F(\boldsymbol{x}; \bar{\boldsymbol{x}}_1, \ldots, \bar{\boldsymbol{x}}_M) &= \Phi(\boldsymbol{x}) - \sum_{m=1}^{M} \mathsf{D}_m(\boldsymbol{x} \,\|\, \bar{\boldsymbol{x}}_m) && (5) \\ &= \sum_{m=1}^{M} \phi_m(\boldsymbol{x}; \bar{\boldsymbol{x}}_m). && (6) \end{aligned}$$

Since

$$\min_{(\bar{\boldsymbol{x}}_1, \ldots, \bar{\boldsymbol{x}}_M) \in \mathcal{X}^M} \sum_{m=1}^{M} \mathsf{D}_m(\boldsymbol{x} \,\|\, \bar{\boldsymbol{x}}_m) = 0, \qquad \forall \boldsymbol{x} \in \mathcal{X},$$

that is,

$$\max_{(\bar{\boldsymbol{x}}_1, \ldots, \bar{\boldsymbol{x}}_M) \in \mathcal{X}^M} F(\boldsymbol{x}; \bar{\boldsymbol{x}}_1, \ldots, \bar{\boldsymbol{x}}_M) = \Phi(\boldsymbol{x}), \qquad \forall \boldsymbol{x} \in \mathcal{X},$$

one can rewrite the optimization problem (2) equivalently as follows:

$$\begin{aligned} &\text{maximize } F(\boldsymbol{x}; \bar{\boldsymbol{x}}_1, \ldots, \bar{\boldsymbol{x}}_M) \\ &\text{subject to } (\boldsymbol{x}; \bar{\boldsymbol{x}}_1, \ldots, \bar{\boldsymbol{x}}_M) \in \mathcal{X}^{M+1}, \end{aligned} \tag{7}$$

in a sense that $\boldsymbol{x}^* \in \mathcal{X}$ is an optimal solution of (2) if and only if $(\boldsymbol{x}^*; \bar{\boldsymbol{x}}_1^*, \ldots, \bar{\boldsymbol{x}}_M^*) \in \mathcal{X}^{M+1}$ is an optimal solution of (7) for some $(\bar{\boldsymbol{x}}_1^*, \ldots, \bar{\boldsymbol{x}}_M^*) \in \mathcal{X}^M$. Therefore we can find a solution to problem (2) by maximizing F with respect to $(\boldsymbol{x}; \bar{\boldsymbol{x}}_1, \ldots, \bar{\boldsymbol{x}}_M)$.

By alternating between updating $\boldsymbol{x}$ and one of the $\bar{\boldsymbol{x}}_m$'s, we obtain an "incremental optimization transfer algorithm" outlined in Table I, where we assume that there exists one or possibly more maximizers in (T-1), and "$\arg\max$" denotes one of those maximizers.

The incremental optimization transfer algorithm shown in Table I can be viewed as a block coordinate ascent algorithm for maximizing F with respect to $(\boldsymbol{x}; \bar{\boldsymbol{x}}_1, \ldots, \bar{\boldsymbol{x}}_M)$ [42, p. 270]. It monotonically increases the augmented objective function F, but not necessarily the original objective function Φ [43]. If one has only one subobjective function in (1), that is, $M = 1$, then the incremental optimization transfer algorithm reduces to an ordinary optimization transfer algorithm [30]. The incremental approach ($M > 1$) usually leads to faster convergence rates than nonincremental methods ($M = 1$) [23]. The incremental

[3]When there exists $\breve{\boldsymbol{x}} \neq \bar{\boldsymbol{x}}$ such that $\mathsf{D}_m(\breve{\boldsymbol{x}} \,\|\, \bar{\boldsymbol{x}}) = 0$, using a modified surrogate $\phi_m^{\text{new}}(\boldsymbol{x}; \bar{\boldsymbol{x}}) = \phi_m(\boldsymbol{x}; \bar{\boldsymbol{x}}) - \epsilon \|\boldsymbol{x} - \bar{\boldsymbol{x}}\|^2$ for any fixed $\epsilon > 0$ would lead to the following property: $\mathsf{D}_m^{\text{new}}(\boldsymbol{x} \,\|\, \bar{\boldsymbol{x}}) \geq 0$ where equality holds *if and only if* $\boldsymbol{x} = \bar{\boldsymbol{x}}$. Although this modification might provide a more natural definition of divergence, it is not needed for our convergence proofs. So we allow the less restrictive conditions in (4).

EM algorithms [23], [27] including COSEM [24], [26] are a special case where the surrogates ϕ_m are constructed by EM principles as described in the next subsection.

If one were to maximize just one of the ϕ_m's instead of the sum shown in (6), then one would have ordinary OS type algorithms. Although this greedy approach usually yields faster initial convergence rates than incremental optimization transfer algorithms, the OS type algorithms are not monotonic in F nor in Φ.

For incremental optimization transfer algorithms one must store M vectors $\{\bar{\boldsymbol{x}}_m\}_{m=1}^M$, so one needs more memory compared to ordinary OS algorithms; however, this is not a practical limitation unless M is overly large.

B. Special Case: Incremental EM Algorithms

This section shows that the incremental EM algorithms are a special case of the incremental optimization transfer framework given in the preceding subsection.

For maximum likelihood (ML) estimation, one must maximize a log-likelihood function

$$\Phi(\boldsymbol{x}) = \log f(\boldsymbol{y}; \boldsymbol{x})$$

with respect to parameter $\boldsymbol{x} \in \mathbb{R}^p$ over a feasible set $\mathcal{X} \subset \mathbb{R}^p$ where $\boldsymbol{y} \in \mathbb{R}^N$ denotes a realization of an observable random vector $\boldsymbol{Y}$ with probability distribution $f(\boldsymbol{y}; \boldsymbol{x}^{\mathrm{true}})$, and $\boldsymbol{x}^{\mathrm{true}} \in \mathbb{R}^p$ is the true value of the unknown parameter. Assume that we identify an admissible complete-data[4] random vector $\boldsymbol{Z}$ for $f(\boldsymbol{y}; \boldsymbol{x})$. Then the following EM surrogate function satisfies the minorization conditions in (3) [29]:

$$\phi(\boldsymbol{x}; \bar{\boldsymbol{x}}) \stackrel{\triangle}{=} E[\log f(\boldsymbol{Z}; \boldsymbol{x}) | \boldsymbol{Y} = \boldsymbol{y}; \bar{\boldsymbol{x}}] \tag{8}$$

for all $\bar{\boldsymbol{x}} \in \mathcal{X}$. But in many applications including imaging problems, the observed data is independent so the log-likelihood objective is additive-separable, that is,

$$\Phi(\boldsymbol{x}) = \sum_{m=1}^{M} \Phi_m(\boldsymbol{x}), \qquad \Phi_m(\boldsymbol{x}) = \log f(\boldsymbol{y}_m; \boldsymbol{x}),$$

and the complete data is conditionally independent, so for each $\Phi_m(\boldsymbol{x})$, one can obtain the following EM surrogate:

$$\phi_m(\boldsymbol{x}; \bar{\boldsymbol{x}}) = E[\log f(\boldsymbol{Z}_m; \boldsymbol{x}) | \boldsymbol{Y}_m = \boldsymbol{y}_m; \bar{\boldsymbol{x}}]\,, \tag{9}$$

[4]A random vector $\boldsymbol{Z}$ with probability distribution $f(\boldsymbol{z}; \boldsymbol{x})$ is called an admissible complete-data vector for $f(\boldsymbol{y}; \boldsymbol{x})$ if $f(\boldsymbol{y}, \boldsymbol{z}; \boldsymbol{x}) = f(\boldsymbol{y}|\boldsymbol{z}) f(\boldsymbol{z}; \boldsymbol{x})$ [37], [38]. A special case is that $\boldsymbol{Y}$ is a deterministic function of $\boldsymbol{Z}$.

which also satisfies the minorization conditions in (3) where $\boldsymbol{Y} = (\boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_M)$ and $\boldsymbol{Z} = (\boldsymbol{Z}_1, \ldots, \boldsymbol{Z}_M)$ are some decompositions of the incomplete data and the complete data, respectively. Defining the augmented objective function as in (6) and then alternating between updating $\boldsymbol{x}$ and one of the $\bar{\boldsymbol{x}}_m$'s as in Table I leads to the incremental EM algorithms [23], [27]. The COSEM algorithm [24], [26], a special case of the incremental EM for emission tomography, can be readily derived.

In some applications, using surrogates other than (8) or (9) can lead to more convenient implementation (*e.g.*, see Section III-B).

C. Convergence Properties

Since incremental optimization transfer algorithms monotonically increase the augmented objective F, the sequence of augmented objective values converges to some value in the usual case where F has an upper bound. However, the question of whether the algorithms really converge to a maximizer of (2) is addressed next.

Define a *solution set* as the collection of stationary points of (2):

$$\Gamma \stackrel{\triangle}{=} \{\boldsymbol{x}^* \in \mathcal{X} : \nabla\Phi(\boldsymbol{x}^*)'(\boldsymbol{x} - \boldsymbol{x}^*) \leq 0, \quad \forall \boldsymbol{x} \in \mathcal{X}\}, \tag{10}$$

where $'$ denotes matrix or vector transpose, and we assume $\Gamma \neq \emptyset$. Each element of the solution set Γ satisfies the first-order necessary condition for a local maximizer of Φ over $\mathcal{X}$ [42, p. 194]. We want algorithms to converge to some point in Γ. If the objective function Φ is concave, then Γ is the set of (possibly multiple) global maximizers of Φ over $\mathcal{X}$ [42, p. 194]. If Φ is strictly concave, then Γ is the singleton of a unique global maximizer [42, p. 685]. On the other hand, for a nonconcave objective function Φ (as in Section III), the solution set Γ could contain local maximizers and even local minimizers. It is difficult to guarantee finding a global maximizer of a nonconcave objective function that may have multiple local maxima. However, the hope is that, with an initial point reasonably close to a global maximizer, the iterates generated by a monotonic algorithm will approach the global maximizer (see [39] for discussion about convergence to a globally optimal point).

In Appendix A, we show that every limit point[5] of the sequence generated by an incremental optimization transfer algorithm is an element of the solution set Γ of stationary points regardless of initial

[5]Recall the distinction between a limit and a limit point. A point $\breve{\boldsymbol{x}}$ is called *a limit* of a sequence $\{\boldsymbol{x}^n\}$ if $\forall \epsilon > 0$, $\exists N$ such that $\forall n > N$, $\|\breve{\boldsymbol{x}} - \boldsymbol{x}^n\| < \epsilon$. On the other hand, a point $\bar{\boldsymbol{x}}$ is called *a limit point* of a sequence $\{\boldsymbol{x}^n\}$ if $\forall \epsilon > 0$, $\forall N$, $\exists n > N$ such that $\|\bar{\boldsymbol{x}} - \boldsymbol{x}^n\| < \epsilon$, in other words, if there exists a subsequence $\{\boldsymbol{x}^{n_k}\}$ whose limit is $\bar{\boldsymbol{x}}$.

points[6] when the following general sufficient conditions hold: (i) each Φ_m and $\phi_m(\cdot;\cdot)$ is continuously differentiable, (ii) the iterates are bounded (*e.g.*, $\mathcal{X}$ is a bounded set), (iii) the surrogates ϕ_m satisfy the minorization conditions in (3), (iv) the gradients of Φ_m and $\phi_m(\cdot;\bar{\boldsymbol{x}})$ match at $\bar{\boldsymbol{x}}$ (see Condition 2 in Appendix A), and (v) the maximizer in (T-1) is defined uniquely (*e.g.*, $\phi_m(\cdot;\bar{\boldsymbol{x}}_m)$ is strictly concave). Consequently, if the objective function Φ is strictly concave, then the algorithm converges to the global maximizer. For a nonconcave objective function Φ, if the points in Γ are isolated, the algorithm will still converge to some stationary point in Γ that we hope is a global maximizer or at least a local maximizer (see Appendix A). It is an open question whether optimization transfer algorithms converge to nonisolated stationary points (see [39] for a discussion of this issue).

Appendix B analyzes the asymptotic local convergence rate of the incremental optimization transfer algorithms, and provides an illustrative one-parameter example for a comparison of the convergence rates of incremental and nonincremental algorithms.

III. Application to Transmission Tomography

In this section we develop a particular incremental optimization transfer algorithm for transmission tomographic reconstruction. We use quadratic surrogates [6], [21] rather than EM surrogates in (9) because the standard complete-data proposed in [32] for transmission tomography does not yield a closed-form M-step [46]. Using quadratic surrogates is not limited to the transmission case [47]–[49]; the incremental optimization transfer algorithms using quadratic surrogates developed in this section are easily extended to other applications including emission tomography.

A. Problem

We assume the following Poisson statistical model for (monoenergetic) transmission measurements:

$$y_i \sim \text{Poisson}\left\{ b_i e^{-[\boldsymbol{A}\boldsymbol{x}]_i} + r_i \right\}, \quad i = 1, \ldots, N \tag{11}$$

where y_i denotes the transmission measurement of the ith detector, b_i denotes the blank scan counts of the ith detector, r_i denotes the mean number of background counts, and $[\boldsymbol{A}\boldsymbol{x}]_i = \sum_{j=1}^{p} a_{ij} x_j$ represents the ith line integral of the attenuation map in which x_j is the unknown attenuation coefficient in the jth pixel, $\boldsymbol{A} = \{a_{ij}\}$ is the system matrix, and N and p are the number of detectors and pixels, respectively.

[6]Some authors define *global convergence* as the property that limit points of the sequence generated by an algorithm are stationary points of the problem [44, p. 228] or that limits are stationary points [45, p. 312], irrespective of starting points. We adopt the former convention here.

We assume that $\{b_i\}$, $\{a_{ij}\}$, and $\{r_i\}$ are known nonnegative constants. We focus on penalized-likelihood (PL), also known as maximum *a posteriori* (MAP), estimation for the attenuation map reconstruction. Our goal is to compute a PL estimate $\hat{\boldsymbol{x}}^{\mathrm{PL}}$ which is defined by

$$\hat{\boldsymbol{x}}^{\mathrm{PL}} = \arg\max_{\boldsymbol{x}\in\mathcal{X}} \Phi(\boldsymbol{x}), \quad \Phi(\boldsymbol{x}) = L(\boldsymbol{x}) - \beta R(\boldsymbol{x}) \tag{12}$$

where the objective function Φ, which can be nonconcave when $r_i \neq 0$ [21], includes the log-likelihood

$$\begin{aligned} L(\boldsymbol{x}) &= \sum_{i=1}^{N} h_i([\boldsymbol{A}\boldsymbol{x}]_i) \\ h_i(l) &= y_i \log(b_i e^{-l} + r_i) - (b_i e^{-l} + r_i) \end{aligned}$$

and a roughness penalty

$$R(\boldsymbol{x}) = \frac{1}{2}\sum_{j=1}^{p}\sum_{k\in\mathcal{N}_j} w_{jk}\psi(x_j - x_k). \tag{13}$$

The box constraint set is defined by

$$\mathcal{X} = \{\boldsymbol{x} \in \mathbb{R}^p : 0 \leq x_j \leq U, \quad \forall j\}. \tag{14}$$

In the box constraint set in (14), the nonnegativity restriction is imposed on physical grounds, and the upper bound $U > 0$ is set by the user to be a value that is larger than the maximum attenuation coefficient conceivable for the object being scanned. The reason for using the box constraint rather than the usual nonnegativity constraint is that the convergence proofs in Appendix A need the iterates to be bounded. However, imposing upper bounds is not overly restrictive in a sense that one can choose a physically meaningful upper bound for attenuation coefficients, and the image estimate $\hat{\boldsymbol{x}}$ is unlikely to be affected by U if one chooses an arbitrarily large U. In practice, if the upper bound happens to be hit by some iterate, then the user could re-run the algorithm with a larger bound.

In the penalty function (13), the function ψ is a symmetric and convex potential function, $\mathcal{N}_j$ represents a neighborhood of the jth pixel, β is a regularization parameter that controls the smoothness in reconstructed images, and w_{jk} are weights (ordinarily, $w_{jk} = 1$ for horizontal and vertical neighboring pixels, and $w_{jk} = 1/\sqrt{2}$ for diagonal neighboring pixels). We assume the potential function ψ satisfies some conditions given in [21], [50, p. 184]. We used the following edge-preserving nonquadratic potential function in our PL reconstruction results [51]:

$$\psi(t) = \delta^2[|t/\delta| - \log(1 + |t/\delta|)] \tag{15}$$

for some $\delta > 0$. We assume that appropriate β and δ are prespecified.

B. Transmission Incremental Optimization Transfer (TRIOT)

We decompose the objective function Φ into the following subobjective functions:

$$\Phi_m(\boldsymbol{x}) = \sum_{i \in S_m} h_i([\boldsymbol{A}\boldsymbol{x}]_i) - \frac{\beta}{M} R(\boldsymbol{x}), \quad m = 1, \cdots, M,$$

where $\{S_m\}_{m=1}^M$ is a partition of $\{1, \ldots, N\}$. We use the usual subsets corresponding to downsampled projection angles [1]. Consider the following *separable* quadratic surrogate ϕ_m for the subobjective function Φ_m:

$$\phi_m(\boldsymbol{x}; \bar{\boldsymbol{x}}) = \Phi_m(\bar{\boldsymbol{x}}) + \nabla \Phi_m(\bar{\boldsymbol{x}})'(\boldsymbol{x} - \bar{\boldsymbol{x}}) - \frac{1}{2}(\boldsymbol{x} - \bar{\boldsymbol{x}})' \breve{\boldsymbol{C}}_m(\bar{\boldsymbol{x}})(\boldsymbol{x} - \bar{\boldsymbol{x}}) \tag{16}$$

with

$$\breve{\boldsymbol{C}}_m(\boldsymbol{x}) = \text{diag}_j\{\breve{c}_{mj}(\boldsymbol{x})\} \tag{17}$$

where $\breve{c}_{mj}(\cdot) > 0$ and $\text{diag}\{\cdot\}$ denotes a diagonal matrix appropriately formed. The surrogates ϕ_m in (16) satisfy Conditions 2 and 3 in Appendix A.

To make ϕ_m additionally satisfy the minorization conditions in (3), one has at least two choices for $\breve{c}_{mj}$: "optimum curvature" (OC) and "maximum curvature" (MC). Those curvatures $\breve{c}_{mj}$ have the following form:

$$\breve{c}_{mj}(\boldsymbol{x}) = \max\left\{ \sum_{i \in S_m} a_{ij} a_i c_i([\boldsymbol{A}\boldsymbol{x}]_i) + \frac{2\beta}{M} \sum_{k \in \mathcal{N}_j} w_{jk} \omega_\psi(x_j - x_k),\, \epsilon \right\} \tag{18}$$

for some small value $\epsilon > 0$ where $a_i \stackrel{\triangle}{=} \sum_{j=1}^p a_{ij}$ and $\omega_\psi(t) \stackrel{\triangle}{=} \dot{\psi}(t)/t$. The functionals $c_i(\cdot)$ are defined as follows. For OC, we define

$$c_i^{\text{OC}}(l) \stackrel{\triangle}{=} \begin{cases} \left[-2 \dfrac{h_i(0) - h_i(l) + \dot{h}_i(l) \cdot l}{l^2} \right]_+, & l > 0 \\ \left[-\ddot{h}_i(0) \right]_+, & l = 0, \end{cases} \tag{19}$$

and for MC,

$$c_i^{\text{MC}}(l) \stackrel{\triangle}{=} \left[-\ddot{h}_i(0) \right]_+, \tag{20}$$

where $[x]_+ = \max\{x, 0\}$. Detailed derivations of (18)–(20) can be found in [21]. On the right side in (18), the first term corresponds to the curvature of the surrogate for the log-likelihood part, and the second term for the penalty part. The optimum curvature c_i^{OC} in (19) is the lowest curvature that a 1D quadratic surrogate function for a marginal log-likelihood $h_i(l)$ can have in projection domain (l) while satisfying the minorization conditions. A low curvature of a surrogate implies a wide paraboloid and, consequently, a large stepsize, that is, fast convergence [21]. However, one needs an "extra" backprojection for computing

the first term in (18). On the other hand, the maximum curvature c_i^{MC} is a constant independent of $\boldsymbol{x}$, and thus the first term in (18) can be precomputed and stored. But c^{MC} is larger than c^{OC} and consequently leads to smaller stepsizes. We leave the second term in (18) as a function of $\boldsymbol{x}$ even for MC since its computation is usually cheap compared to projection and backprojection operations unless M is too large.

The augmented objective function F defined in (6) with (16) is readily maximized with respect to $\boldsymbol{x}$ over the box constraint $\mathcal{X}$ as follows:

$$\hat{\boldsymbol{x}} = \mathcal{P}_{\mathcal{X}} \left(\left[\sum_{m=1}^{M} \breve{\boldsymbol{C}}_m(\bar{\boldsymbol{x}}_m) \right]^{-1} \sum_{m=1}^{M} \left[\breve{\boldsymbol{C}}_m(\bar{\boldsymbol{x}}_m) \bar{\boldsymbol{x}}_m + \nabla \Phi_m(\bar{\boldsymbol{x}}_m) \right] \right) \tag{21}$$

where $\mathcal{P}_{\mathcal{X}}(\boldsymbol{x})$ is the orthogonal projection of $\boldsymbol{x} \in \mathbb{R}^p$ onto $\mathcal{X}$ and is easily computed componentwise as follows: $[\mathcal{P}_{\mathcal{X}}(\boldsymbol{x})]_j = \mathrm{median}\{0, x_j, U\}$ for all j. Using (21) in the step (T-1) leads to a new "transmission incremental optimization transfer (TRIOT)" algorithm, which is outlined in Table II. When $M = 1$, then TRIOT reduces to ordinary SPS [6]. The TRIOT update begins after $n_{\mathrm{iter}}^{\mathrm{OS}}$ (≥ 1) iteration(s) of OS-SPS [6] (see the next subsection for OS-SPS in detail). The strategy to switch from OS-SPS to TRIOT is discussed in Section III-D. Running initially (at least) one iteration of OS-SPS is more effective than initializing all $\bar{\boldsymbol{x}}_m$'s to be the same image (*e.g.*, a FBP or uniform image) because both cases require nearly the same computation (note one needs to compute partial gradients $\nabla \Phi_m(\bar{\boldsymbol{x}}_m)$ and curvatures for all m to perform the TRIOT update) yet one can take advantage of fast initial convergence rates of OS-SPS.

In Table II, a TRIOT using MC in (20), we call TRIOT-MC[7], is shown; however, OC in (19) can be easily included. The two steps (T-1) and (T-2) in Table I are combined in Table II. In (T-5), one can avoid the sum $\sum_{l=1}^{M}$ at every subiteration by maintaining that sum as a state vector that is updated incrementally as in [24], [26], [33]. And one could slightly modify the algorithm to perform (T-5) more than one time at every subiteration so that one additionally updates the surrogate for the penalty part with fixing the surrogate for the likelihood part as in [21]. One iteration, indexed by n, of TRIOT-MC requires one projection and one backprojection operation while TRIOT-OC needs an extra backprojection [see (18) and (19)].

The discussion and proofs for global convergence given in Section II-C and Appendix A apply to TRIOT. When $r_i = 0$ for all i, under mild conditions,[8] since the PL objective for transmission tomography is strictly concave, the algorithm converges to the optimal solution [52]. In the case $r_i \neq 0$, the objective

[7]The second part denotes a specific curvature used (*e.g.*, SPS-OC).

[8]The potential function ψ is strictly convex, and $\boldsymbol{A}'\boldsymbol{y} \neq \boldsymbol{0}$.

function is not necessarily concave [21], and we have a weaker conclusion that every limit point of a sequence generated by TRIOT is a stationary point. However, in our practical experience, we obtained the same limit in all experiments with different initializations, suggesting that suboptimal local maxima are rare, or are far from reasonable starting images.

C. OS-SPS

Since we use OS-SPS in initializing and accelerating TRIOT, we briefly review OS-SPS [6] for completeness. For each subiteration, indexed by m, maximizing the mth subobjective $\phi_m(\cdot; \bar{\boldsymbol{x}}_m)$ in (16) instead of the augmented objective $F(\cdot; \bar{\boldsymbol{x}}_1, \ldots, \bar{\boldsymbol{x}}_M)$ in (6) leads to the following OS-SPS update:

$$\bar{\boldsymbol{x}}^{\text{new}}_{(m \bmod M)+1} = \mathcal{P}_{\mathcal{X}}\left(\bar{\boldsymbol{x}}_m + \left[\breve{\boldsymbol{C}}_m(\bar{\boldsymbol{x}}_m) \right]^{-1} \nabla \Phi_m(\bar{\boldsymbol{x}}_m) \right) \tag{22}$$

for $m = 1, \ldots, M$ where $\breve{\boldsymbol{C}}_m(\cdot)$ is based on (18). This greedy approach does not ensure monotonicity, in neither the augmented objective nor the PL objective, so we need not insist that the curvatures satisfy the minorization conditions. A natural choice for $c_i(\cdot)$ is the Newton's curvature $-\ddot{h}_i(\cdot)$; this can be approximated as follows:

$$\begin{aligned} -\ddot{h}_i(l) \approx c_i^{\text{PC}} &\triangleq -\ddot{h}_i\left(\arg\max_{\tilde{l} \geq 0} h_i(\tilde{l}) \right) \\ &= \begin{cases} \dfrac{(y_i - r_i)^2}{y_i}, & y_i > r_i \\ 0, & \text{otherwise.} \end{cases} \end{aligned} \tag{23}$$

This choice is called "precomputed curvature (PC)" [6], [21]. For OS-SPS, the following subset-independent preconditioning matrix using PC is usually used in place of $\breve{\boldsymbol{C}}_m(\bar{\boldsymbol{x}}_m)$ in (22):

$$\begin{aligned} \breve{\boldsymbol{C}}^{\text{PC}}(\boldsymbol{x}) &= \text{diag}_j\left\{ \breve{c}_j^{\text{PC}}(\boldsymbol{x}) \right\} \\ \breve{c}_j^{\text{PC}}(\boldsymbol{x}) &= \max\left\{ \frac{1}{M} \sum_{i=1}^{N} a_{ij} a_i c_i^{\text{PC}} + \frac{2\beta}{M} \sum_{k \in \mathcal{N}_j} w_{jk} \omega_\psi(x_j - x_k),\, \epsilon \right\} \end{aligned} \tag{24}$$

where c_i^{PC} is given in (23). The first term on the right side in (24) can be precomputed and stored like the maximum curvatures (MC). The benefit of using PC is that it leads to faster convergence rates than MC since $c_i^{\text{MC}} \geq c_i^{\text{PC}}$. The update for OS-SPS is shown in (T-4) in Table II.

The OS-SPS shows very fast initial convergence rates but becomes eventually stuck at a limit cycle. Using more subsets leads to a faster initial convergence rate but causes the points in the limit cycle to be farther from the optimal solution.

It is worth noting that, for each update, OS-SPS uses the gradient and curvature for only one subobjective function at the previous subiterate in (22) whereas TRIOT uses the gradients and curvatures for all

subobjective functions at previous M subiterates respectively in (21). When the number of subobjective functions is $M = 1$, then both OS-SPS and TRIOT reduce to SPS.

D. Acceleration

TRIOT-OC/MC is convergent yet faster than nonincremental ordinary SPS [6], but it is still slower initially than OS-SPS which is not convergent unless relaxed. Here we discuss methods to accelerate TRIOT.

1) Switch from OS-SPS to TRIOT: It is a popular idea to switch from a nonconvergent yet initially fast OS type algorithm to a convergent non-OS algorithm at some point to take advantage of both fast initial convergence rates of OS methods and global convergence of non-OS methods.

We observed that it is very effective to switch to TRIOT from OS-SPS at the point where the OS-SPS algorithm nearly gets to a limit cycle; even one single subiteration of TRIOT moves the iterate from the limit cycle very close to the optimal solution. The reason is as follows: a group of the points in the limit cycle would be roughly centered around the optimal point and the update for TRIOT includes a weighted average of the points [see the first term on the right side in (21) or (T-5)].

To obtain further insight into this property, consider a simple unconstrained quadratic problem where the objective function and the subobjective functions are

$$\Phi(\boldsymbol{x}) = -\frac{1}{2}\boldsymbol{x}'\boldsymbol{Q}\boldsymbol{x} + \boldsymbol{b}'\boldsymbol{x}, \qquad \Phi_m(\boldsymbol{x}) = -\frac{1}{2}\boldsymbol{x}'\boldsymbol{Q}_m\boldsymbol{x} + \boldsymbol{b}_m'\boldsymbol{x}$$

for $m = 1, \cdots, M$ where $\sum_{m=1}^{M} \boldsymbol{Q}_m = \boldsymbol{Q}$ and $\sum_{m=1}^{M} \boldsymbol{b}_m = \boldsymbol{b}$. Assume that each surrogate function $\phi_m(\boldsymbol{x}; \bar{\boldsymbol{x}})$ is equal to its corresponding subobjective $\Phi_m(\boldsymbol{x})$ so it has a closed-form maximizer $\hat{\boldsymbol{x}}_m = \boldsymbol{Q}_m^{-1}\boldsymbol{b}_m$ where we assume each $\boldsymbol{Q}_m$ is invertible. Then the OS approach will generate a limit cycle that consists of those $\hat{\boldsymbol{x}}_1, \ldots, \hat{\boldsymbol{x}}_M$. Now applying just one iteration of the incremental optimization transfer method as in (21) leads to

$$\begin{aligned} \hat{\boldsymbol{x}} = \left(\sum_{m=1}^{M} \boldsymbol{Q}_m\right)^{-1} \sum_{m=1}^{M} \boldsymbol{Q}_m \hat{\boldsymbol{x}}_m &= \boldsymbol{Q}^{-1} \sum_{m=1}^{M} \boldsymbol{Q}_m (\boldsymbol{Q}_m^{-1}\boldsymbol{b}_m) \\ &= \boldsymbol{Q}^{-1}\boldsymbol{b}, \end{aligned}$$

which is the maximizer of the original objective Φ [the second term on the right side in (21) equals zero]. This example suggests that the built-in averaging operation in TRIOT helps iterates escape from a limit cycle, generated by nonconvergent OS algorithms, towards the optimal solution.

However, in the early iterations, when OS-SPS is still far from the limit cycle and is making progress towards the optimal point, TRIOT is usually slower than OS-SPS due to the averaging of the past

subiterates because the incremental optimization transfer approach updates the surrogates incrementally, that is, conservatively to ensure monotonicity. So it is desirable to get to a limit cycle quickly using OS-SPS with *many* subsets and then switch to TRIOT. In a 2D reconstruction case in Section IV, the use of 64 subsets is sufficient to reach a limit cycle within a couple of iterations.

2) Precomputed Curvatures: Forgoing monotonicity (in the augmented objective) and accordingly provable convergence, one can use for TRIOT the "precomputed curvatures (PC)" in (23). TRIOT-PC is faster than provably convergent TRIOT-OC/MC. It is an open question whether TRIOT-PC converges to an optimal solution. However, in our experiments, TRIOT-PC yielded the same limit as convergent algorithms like SPS-OC within numerical precision!

3) Enhanced Incremental Optimization Transfer Algorithms: Hsiao *et al.* proposed E-COSEM, an accelerated version of COSEM [33]. The idea is to choose for each update a convex combination of an initially fast yet nonconvergent OS algorithm and a convergent incremental optimization transfer algorithm such that the combination both ensures monotonicity in the augmented objective and is as close to the OS algorithm as possible. This approach often accelerates incremental optimization transfer algorithms without destroying the monotonicity in the augmented objective.

IV. Results

To assess the performance of the proposed algorithms, we performed 2D attenuation map reconstructions from real PET data.

We acquired PET data using a Siemens/CTI ECAT EXACT 921 PET scanner with rotating rod transmission sources [53]. We used an anthropomorphic thorax phantom (Data Spectrum, Chapel Hill, NC). The sinogram had 160 radial bins and 192 angles, and the reconstructed images were 128×128 with 4.2 mm pixels. The system geometry was approximated with 3.375 mm wide strip integrals and 3.375 mm ray spacing; the system matrix was generated using ASPIRE [54]. The total counts amounted to 9.2×10^5. We used the edge-preserving nonquadratic penalty (15) with $\delta = 4 \times 10^{-4}$ mm^{-1} and $\beta = 2^{18.5}$, chosen by visual inspection. A uniform image was used as a starting image. The results obtained by using a FBP reconstruction as a starting image were similar and are not shown here.

Images were reconstructed using SPS-MC/PC, OS-SPS, and TRIOT-MC/PC. For OS-SPS and TRIOT algorithms, we used 16 subsets (a moderate number) and 64 subsets (a little larger number than usual). For SPS and TRIOT, the performance (objective value or distance from the optimal image) with the optimum curvature (OC) in (19), that requires an extra backprojection per iteration, was between those with MC

and PC; and the results with OC are not shown here. And, for TRIOT algorithms, the enhancement method described in Section III-D yielded only minor improvements in this study (not shown here).

Fig. 1 shows normalized Φ difference versus iteration number for different algorithms using 16 subsets. The normalized Φ difference is defined as $(\Phi(\hat{\boldsymbol{x}}^{\mathrm{PL}}) - \Phi(\hat{\boldsymbol{x}}^n))/(\Phi(\hat{\boldsymbol{x}}^{\mathrm{PL}}) - \Phi(\hat{\boldsymbol{x}}^0))$ where $\hat{\boldsymbol{x}}^{\mathrm{PL}}$ is a maximizer of the PL objective; a small value means the image is closer to the optimal image $\hat{\boldsymbol{x}}^{\mathrm{PL}}$. The optimal image $\hat{\boldsymbol{x}}^{\mathrm{PL}}$ [shown in Fig. 5(b)] was estimated by 30 iterations of OS-SPS with 16 subsets followed by 800 iterations of the SPS-OC algorithm that is monotonic and convergent (to a stationary point). As described in Section III-B, TRIOT algorithms were initialized by running one iteration of OS-SPS. So were the SPS algorithms for a fair comparison. Although OS-SPS showed a fast initial convergence rate, it became stuck at a suboptimal point whereas other methods continued to improve in terms of objective values. The TRIOT algorithms were outperformed by other algorithms in early iterations since the built-in averaging in TRIOT slows down convergence, as discussed in Section III-D, when a limit cycle has not reached yet. However, TRIOT-MC and TRIOT-PC eventually outrun SPS-MC and SPS-PC, respectively. Although global convergence is not provably ensured for TRIOT-PC, the limit of TRIOT-PC (say, obtained by 1000 iterations) was the same as that of SPS-OC (obtained similarly) within numerical precision (not shown here), which suggests TRIOT-PC has desirable convergence properties.

To investigate the performance of TRIOT algorithms after OS-SPS reaches a limit cycle, we performed 6 iterations of OS-SPS, which is sufficient to get close to a limit cycle, and then applied TRIOT (and SPS as well). Fig. 2 shows that TRIOT yielded considerable improvement at iteration 6 where TRIOT was first applied. TRIOT-MC and TRIOT-PC converge faster than SPS-MC and SPS-PC, respectively, which are similarly initialized by 6 iterations of OS-SPS. This shows that it is effective to switch from OS-SPS to TRIOT, as described in Section III-D, when OS-SPS almost reaches a limit cycle. However, it is inconvenient to predict how many iterations are required for OS-SPS to arrive at a limit cycle.

Fig. 3 shows normalized Φ difference versus iteration number when 64 subsets are used. As the number of subsets increased to 64, the initial convergence rate of OS-SPS became faster (even a couple of iterations led to a limit cycle) but OS-SPS stagnated at a worse image. Meanwhile, the TRIOT algorithms were quite effective even though they used only a couple of iterations of OS-SPS as initialization, and they outperformed the SPS algorithms initialized similarly. In light of the effectiveness of the built-in averaging in TRIOT, to make SPS a stronger competitor, prior to switching to SPS (at iteration 2), we averaged the 64 previous subiterates that approximately comprise the limit cycle. As shown in Fig. 3, this averaging yielded significant improvements for SPS algorithms. However, convergence rates of TRIOT were still faster than those of SPS with such averaging. Fig. 4 shows a similar trend when the results are viewed in

terms of the normalized distance from the optimal image, $\|\hat{\boldsymbol{x}}^n - \hat{\boldsymbol{x}}^{\mathrm{PL}}\|/\|\hat{\boldsymbol{x}}^{\mathrm{PL}}\|$, versus iteration number.

Fig. 5(c) shows the image to which OS-SPS with 64 subsets converged. It represents one point of the limit cycle generated by the OS-SPS, and looks visually different from the true PL optimal image in Fig. 5(b). In contrast, the TRIOT-PC initialized by 2 iterations of OS-SPS yielded, with 18 iterations, the image in Fig. 5(d) which is nearly indistinguishable from the optimal image in Fig. 5(b).

V. Conclusion

We presented a broad family of incremental optimization transfer algorithms by generalizing the incremental EM family. The incremental optimization transfer algorithms usually show faster convergence rates than ordinary optimization transfer methods like EM, but they are globally convergent.

We also developed a particular incremental optimization transfer algorithm for transmission tomography by using separable quadratic surrogates: TRIOT algorithms. We found that it is very effective to switch from OS-SPS to TRIOT when OS-SPS nearly reaches a limit cycle. When reasonably many subsets are used, as few as one or two iteration(s) of OS-SPS can be sufficient to get close to a limit cycle (although it would depend on the degree of regularization and the size of the problem). This switching strategy is more convenient than relaxed OS algorithms that require determining relaxation parameters. Also, TRIOT is preferable to reducing the number of subsets with iteration since the consistent data flow in OS-SPS and TRIOT could be beneficial and it would be inconvenient to determine an optimal schedule for reducing the number of subsets. The switching idea is also found in [55].

One iteration of TRIOT-MC/PC or OS-SPS requires computing one projection and one backprojection plus the penalty related gradients and curvatures (the use of OC needs an extra backprojection); so the computational cost is almost the same as classic ML-EM except for the contribution of the penalty part. As the number of subsets increases, computation per iteration also increases due to the penalty part being updated for each subiteration. Although the computational contribution of the penalty function is usually small compared to projection/backprojection particularly for a large-scale problem like 3D, further investigation could help reduce this computation, *e.g.*, by subsetizing the penalty part.

In our 2D reconstruction from real PET data, with 64 subsets, it was very effective to switch from OS-SPS to TRIOT-PC after 2 iterations of OS-SPS. This switching strategy seems robust since we obtained similar results (not shown here) from a 2D simulation study using a different digital phantom. Although the TRIOT-PC was numerically found to be convergent, if one really wants provable convergence, one could switch to TRIOT-MC or OC at some point.

APPENDIX A
GLOBAL CONVERGENCE PROOF

In this appendix we prove the convergence of the incremental optimization transfer algorithm given in Table I. Define $\boldsymbol{z} \stackrel{\triangle}{=} (\boldsymbol{x}; \bar{\boldsymbol{x}}_1, \ldots, \bar{\boldsymbol{x}}_M) \in \mathcal{X}^{M+1}$, and define a mapping $\mathcal{M} : \mathcal{X}^{M+1} \to \mathcal{X}^{M+1}$ such that $\mathcal{M}(\boldsymbol{z}^n) = \boldsymbol{z}^{n+1}$ where $\boldsymbol{z}^{n+1} = (\boldsymbol{x}^{n+1}; \bar{\boldsymbol{x}}_1^{n+1}, \ldots, \bar{\boldsymbol{x}}_M^{n+1})$ is computed by (T-1)–(T-3) for $\boldsymbol{z}^n = (\boldsymbol{x}^n; \bar{\boldsymbol{x}}_1^n, \ldots, \bar{\boldsymbol{x}}_M^n)$. Suppose that the algorithm generates a sequence $\{\boldsymbol{z}^n\}$ (or a sequence $\{\boldsymbol{x}^n\}$ by taking the first component of $\boldsymbol{z}^n$), given some initial point $\boldsymbol{z}^0 \in \mathcal{X}^{M+1}$. Define an augmented solution set as follows:

$$\Lambda \stackrel{\triangle}{=} \{\boldsymbol{z} = (\boldsymbol{x}; \boldsymbol{x}, \ldots, \boldsymbol{x}) \in \mathcal{X}^{M+1} : \boldsymbol{x} \in \Gamma\} \tag{25}$$

where Γ is defined in (10). We impose the following assumptions.

Assumption 1: Each Φ_m and $\phi_m(\cdot;\cdot)$ is continuously differentiable on a nonempty, closed, and convex set $\mathcal{X} \subset \mathbb{R}^p$ and $\mathcal{X}^2 \subset \mathbb{R}^p \times \mathbb{R}^p$, respectively.

Assumption 2: The iterates $\{\boldsymbol{z}^n\}$ are bounded where $\boldsymbol{z}^n = (\boldsymbol{x}^n; \bar{\boldsymbol{x}}_1^n, \ldots, \bar{\boldsymbol{x}}_M^n)$.

Assumption 2 is ensured by either of the following sufficient conditions.

Assumption 2′: The feasible set $\mathcal{X}$ is bounded.

Assumption 2″: A level set defined by $\{\boldsymbol{z} \in \mathcal{X}^{M+1} : F(\boldsymbol{z}) \geq F(\boldsymbol{z}^0)\}$ is bounded.

We assume that the surrogates ϕ_m satisfy the following conditions.

Condition 1: The functionals ϕ_m satisfy the minorization conditions in (3).

Condition 2: The following derivatives match for all m and $\boldsymbol{x} \in \mathcal{X}$:

$$\nabla \Phi_m(\boldsymbol{x}) = \nabla^{10} \phi_m(\boldsymbol{x}; \boldsymbol{x}) \tag{26}$$

where ∇^{10} is the column gradient operator with respect to the first argument[9] (see [39] for less restrictive conditions).

Condition 3: There exists a *unique* maximizer in (T-1).

The following is sufficient for Condition 3.

Condition 3′: Each $\phi_m(\cdot; \bar{\boldsymbol{x}}_m)$ is strictly concave for all $\bar{\boldsymbol{x}}_m \in \mathcal{X}$, and there exists a maximizer of $F(\cdot; \bar{\boldsymbol{x}}_1, \ldots, \bar{\boldsymbol{x}}_M)$ over $\mathcal{X}$ for all $\bar{\boldsymbol{x}}_1, \ldots, \bar{\boldsymbol{x}}_M \in \mathcal{X}$.

Using the above assumptions and conditions, we prove a series of lemmas necessary for proving convergence.

[9]For $\boldsymbol{x}$ being an interior point of $\mathcal{X}$, Condition 2 is implied by Condition 1 [56].

Lemma 1: The iterates $\{\boldsymbol{z}^n\}$ generated by (T-1)–(T-3) yield monotonic increases in F, that is, $F(\boldsymbol{z}^{n+1}) \geq F(\boldsymbol{z}^n)$ for all n.

Proof: It follows from the cyclic block coordinate ascent updates in (T-1) and (T-2). ■

Lemma 2: Suppose that $\boldsymbol{z}^* \in \mathcal{X}^{M+1}$ is a fixed point of $\mathcal{M}$, that is, $\mathcal{M}(\boldsymbol{z}^*) = \boldsymbol{z}^*$. Then $\boldsymbol{z}^* \in \Lambda$ where Λ is defined in (25).

Proof: For the fixed point $\boldsymbol{z}^* = (\boldsymbol{x}^*; \bar{\boldsymbol{x}}_1^*, \ldots, \bar{\boldsymbol{x}}_M^*)$, in view of Condition 3, one can show that $\boldsymbol{x}^* = \bar{\boldsymbol{x}}_1^* = \ldots = \bar{\boldsymbol{x}}_M^*$. Since $\boldsymbol{x}^*$ is a maximizer of $\sum_{m=1}^M \phi_m(\cdot; \boldsymbol{x}^*)$ over $\mathcal{X}$, it follows that $\sum_{m=1}^M \nabla^{10}\phi_m(\boldsymbol{x}^*; \boldsymbol{x}^*)'(\boldsymbol{x} - \boldsymbol{x}^*) \leq 0$ for all $\boldsymbol{x} \in \mathcal{X}$ [42, p. 194]. Therefore, by Condition 2, $\nabla\Phi(\boldsymbol{x}^*)'(\boldsymbol{x} - \boldsymbol{x}^*) \leq 0$ for all $\boldsymbol{x} \in \mathcal{X}$, and it follows that $\boldsymbol{x}^* \in \Gamma$. ■

Lemma 3: If $\boldsymbol{z} \notin \Lambda$, then $F(\mathcal{M}(\boldsymbol{z})) > F(\boldsymbol{z})$.

Proof: If $\boldsymbol{z} \notin \Lambda$, then $\boldsymbol{z}$ is not a fixed point of $\mathcal{M}$ by Lemma 2. Combining Condition 3 and Lemma 1 leads to the conclusion. ■

Now we prove the following theorem on the convergence of the incremental optimization transfer algorithm.

Theorem 1: Suppose that $\{\boldsymbol{z}^n\}$ is a sequence generated by (T-1)–(T-3) with $\boldsymbol{z}^0 \in \mathcal{X}^{M+1}$ and that Assumptions 1 and 2 and Conditions 1–3 hold. Then any limit point of $\{\boldsymbol{z}^n\}$ is an element of Λ.

Proof: Following [44, p. 209 and p. 228], one can show that the mapping $\mathcal{M}$ is closed, in other words, $\mathcal{M}$ is continuous. The conclusion then follows from the Zangwill's Convergence Theorem [57, p. 91] with Assumption 2, Lemmas 1 and 3, and the closedness of $\mathcal{M}$. ■

The following corollaries and lemmas also hold when "$\{\boldsymbol{x}^n\}$" is replaced with "$\{\bar{\boldsymbol{x}}_m^n\}$" for all m.

Corollary 1: Suppose $\{\boldsymbol{x}^n\}$ is a sequence obtained by taking the first component from $\boldsymbol{z}^n$ in Theorem 1. Then any limit point of $\{\boldsymbol{x}^n\}$ is an element of Γ.

Proof: Use Theorem 1, Assumption 2, and the definition of Λ in (25). ■

Corollary 2: If Φ is concave, then any limit point of $\{\boldsymbol{x}^n\}$ is a global maximizer of Φ over $\mathcal{X}$. Moreover, if Φ is strictly concave, then $\{\boldsymbol{x}^n\}$ converges to *the* global maximizer of Φ over $\mathcal{X}$.

Proof: Use Corollary 1 and [42, Proposition 2.1.2]. ■

When Φ is not strictly concave, there is no guarantee that the algorithm converges to a limit. However, convergence can be established by additionally assuming that the solution set Γ is discrete.

Lemma 4: Suppose $\{\boldsymbol{x}^n\}$ is a sequence from Corollary 1. Then $\left\|\boldsymbol{x}^{n+1} - \boldsymbol{x}^n\right\| \to 0$.

Proof: It follows from [58, Theorem 3.1] that $\|\boldsymbol{z}^{n+1} - \boldsymbol{z}^n\| \to 0$. Since $\|\boldsymbol{z}^{n+1} - \boldsymbol{z}^n\|^2 = \|\boldsymbol{x}^{n+1} - \boldsymbol{x}^n\|^2 + \sum_{m=1}^M \|\bar{\boldsymbol{x}}_m^{n+1} - \bar{\boldsymbol{x}}_m^n\|^2$, it must be a case that $\left\|\boldsymbol{x}^{n+1} - \boldsymbol{x}^n\right\| \to 0$. ■

Lemma 5: Suppose $\{\boldsymbol{x}^n\}$ is a sequence from Corollary 1. Additionally, suppose that the set Γ is discrete. Then $\{\boldsymbol{x}^n\}$ converges to an element in Γ.

Proof: Let S be a set of limit points of $\{\boldsymbol{x}^n\}$. Then $S \subset \Gamma$ by Theorem 1. But, by Lemma 4, S is connected [59, p. 173]. Since S is both discrete and connected, it is a singleton. ■

The above lemma implies that if stationary points of (2) are isolated, then the algorithm converges to one of them.

APPENDIX B
LOCAL CONVERGENCE RATE ANALYSIS

A. Asymptotic Convergence Rate

We analyze the asymptotic convergence rate of the incremental optimization transfer algorithm given in Table I. As in usual local convergence analysis, we assume that a sequence $\{\bar{\boldsymbol{x}}_m^n\}_{n=1}^{\infty}$ generated by the algorithm converges to an optimal point $\hat{\boldsymbol{x}}$ of (2) for all m, and that every iterate $\bar{\boldsymbol{x}}_m^n$ and the limit $\hat{\boldsymbol{x}}$ lie in the interior of $\mathcal{X}$.

Consider the following first-order Taylor's expansion of $\nabla^{10}\phi_m(\cdot;\bar{\boldsymbol{x}}_m^n)$ with respect to the first argument about $\bar{\boldsymbol{x}}_m^n$:

$$\nabla^{10}\phi_m(\boldsymbol{x};\bar{\boldsymbol{x}}_m^n) \approx \nabla^{10}\phi_m(\bar{\boldsymbol{x}}_m^n;\bar{\boldsymbol{x}}_m^n) + \nabla^{20}\phi_m(\bar{\boldsymbol{x}}_m^n;\bar{\boldsymbol{x}}_m^n)(\boldsymbol{x}-\bar{\boldsymbol{x}}_m^n) \tag{27}$$

where ∇^{20} is the Hessian operator with respect to the first argument. The first term on the right hand side can be further approximated as

$$\begin{aligned}\nabla^{10}\phi_m(\bar{\boldsymbol{x}}_m^n;\bar{\boldsymbol{x}}_m^n) &= \nabla\Phi_m(\bar{\boldsymbol{x}}_m^n) \\ &\approx \nabla\Phi_m(\hat{\boldsymbol{x}}) + \nabla^2\Phi_m(\hat{\boldsymbol{x}})(\bar{\boldsymbol{x}}_m^n - \hat{\boldsymbol{x}}) \end{aligned} \tag{28}$$

where the equality is due to (26). Because of the assumption of $\hat{\boldsymbol{x}}$ and $\bar{\boldsymbol{x}}_1^{n+1}$ being in the interior of $\mathcal{X}$, and the construction of $\bar{\boldsymbol{x}}_1^{n+1} = \arg\max_{\boldsymbol{x}\in\mathcal{X}} F(\boldsymbol{x};\bar{\boldsymbol{x}}_1^n,\cdots,\bar{\boldsymbol{x}}_M^n)$ [see (T-1) and (T-2)], it follows that

$$\sum_{m=1}^{M}\nabla\Phi_m(\hat{\boldsymbol{x}}) = \nabla\Phi(\hat{\boldsymbol{x}}) = \mathbf{0}$$

$$\sum_{m=1}^{M}\nabla^{10}\phi_m(\bar{\boldsymbol{x}}_1^{n+1};\bar{\boldsymbol{x}}_m^n) = \mathbf{0}.$$

Now combining (27) and (28) yields the following approximation:

$$\boldsymbol{e}_1^{n+1} \approx \left[\sum_{m=1}^{M}\nabla^{20}\phi_m(\bar{\boldsymbol{x}}_m^n;\bar{\boldsymbol{x}}_m^n)\right]^{-1}\sum_{m=1}^{M}\left[\nabla^{20}\phi_m(\bar{\boldsymbol{x}}_m^n;\bar{\boldsymbol{x}}_m^n) - \nabla^2\Phi_m(\hat{\boldsymbol{x}})\right]\boldsymbol{e}_m^n$$

where $\boldsymbol{e}_m^n \stackrel{\triangle}{=} \bar{\boldsymbol{x}}_m^n - \hat{\boldsymbol{x}}$ for all m and n. Similarly, one can obtain the following approximation for all m:

$$\boldsymbol{e}_m^{n+1} \approx \left[\sum_{k=1}^{m-1} \nabla^{20}\phi_k(\bar{\boldsymbol{x}}_k^{n+1};\bar{\boldsymbol{x}}_k^{n+1}) + \sum_{k=m}^{M} \nabla^{20}\phi_k(\bar{\boldsymbol{x}}_k^{n};\bar{\boldsymbol{x}}_k^{n})\right]^{-1} \cdot \left(\sum_{k=1}^{m-1}\left[\nabla^{20}\phi_k(\bar{\boldsymbol{x}}_k^{n+1};\bar{\boldsymbol{x}}_k^{n+1}) - \nabla^2\Phi_k(\hat{\boldsymbol{x}})\right]\boldsymbol{e}_k^{n+1} + \sum_{k=m}^{M}\left[\nabla^{20}\phi_k(\bar{\boldsymbol{x}}_k^{n};\bar{\boldsymbol{x}}_k^{n}) - \nabla^2\Phi_k(\hat{\boldsymbol{x}})\right]\boldsymbol{e}_k^{n}\right). \quad (29)$$

Assuming that $\nabla^{20}\phi_m(\cdot;\cdot)$ is continuous, it will converge to $\nabla^{20}\phi_m(\hat{\boldsymbol{x}};\hat{\boldsymbol{x}})$ as $\lim_{n\to\infty}\bar{\boldsymbol{x}}_m^n = \hat{\boldsymbol{x}}$. For notational convenience, define $\boldsymbol{D}_m \stackrel{\triangle}{=} \nabla^{20}\phi_m(\hat{\boldsymbol{x}};\hat{\boldsymbol{x}})$, $\boldsymbol{H}_m \stackrel{\triangle}{=} \nabla^2\Phi_m(\hat{\boldsymbol{x}})$, and $\boldsymbol{T}_m \stackrel{\triangle}{=} (\sum_{k=1}^{M}\boldsymbol{D}_k)^{-1}(\boldsymbol{D}_m - \boldsymbol{H}_m)$ for all m. Then one can write the *asymptotic* approximation of (29) in matrix form as follows:

$$\boldsymbol{\mathcal{E}}^{n+1} \approx (\boldsymbol{I}_{pM} - \boldsymbol{\Gamma}_l)^{-1}\boldsymbol{\Gamma}_u\boldsymbol{\mathcal{E}}^n$$

where $\boldsymbol{\mathcal{E}}^n \stackrel{\triangle}{=} [(\boldsymbol{e}_1^n)', \cdots (\boldsymbol{e}_M^n)']'$ is a $pM \times 1$ column vector, $\boldsymbol{I}_k$ is a $k \times k$ identity matrix, and

$$\boldsymbol{\Gamma}_l = \begin{bmatrix} \boldsymbol{0} & \boldsymbol{0} & \cdots & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{T}_1 & \boldsymbol{0} & \cdots & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{T}_1 & \boldsymbol{T}_2 & \cdots & \boldsymbol{0} & \boldsymbol{0} \\ \vdots & \vdots & & \vdots & \vdots \\ \boldsymbol{T}_1 & \boldsymbol{T}_2 & \cdots & \boldsymbol{T}_{M-1} & \boldsymbol{0} \end{bmatrix} \quad (30)$$

$$\boldsymbol{\Gamma}_u = \begin{bmatrix} \boldsymbol{T}_1 & \boldsymbol{T}_2 & \cdots & \boldsymbol{T}_{M-1} & \boldsymbol{T}_M \\ \boldsymbol{0} & \boldsymbol{T}_2 & \cdots & \boldsymbol{T}_{M-1} & \boldsymbol{T}_M \\ \vdots & \vdots & & \vdots & \vdots \\ \boldsymbol{0} & \boldsymbol{0} & \cdots & \boldsymbol{T}_{M-1} & \boldsymbol{T}_M \\ \boldsymbol{0} & \boldsymbol{0} & \cdots & \boldsymbol{0} & \boldsymbol{T}_M \end{bmatrix} \quad (31)$$

with $\boldsymbol{0}$ being a $p \times p$ zero matrix. Thus, the root-convergence factor [60, p. 288] of the sequence $\{[(\bar{\boldsymbol{x}}_1^n)', \cdots, (\bar{\boldsymbol{x}}_M^n)']'\}_{n=1}^{\infty}$ for the incremental optimization transfer algorithm is given by the spectral radius

$$\rho_M = \rho\big((\boldsymbol{I}_{pM} - \boldsymbol{\Gamma}_l)^{-1}\boldsymbol{\Gamma}_u\big) \quad (32)$$

where $\rho(\cdot)$ denotes spectral radius. One can show that the root-convergence factor of the sequence $\{\bar{\boldsymbol{x}}_m^n\}_{n=1}^{\infty}$ is also governed by the above spectral radius for all m. For ordinary optimization transfer algorithms, that is, when $M = 1$, the spectral radius (32) reduces to

$$\rho_1 = \rho\big(\boldsymbol{I}_p - [\nabla^{20}\phi(\hat{\boldsymbol{x}};\hat{\boldsymbol{x}})]^{-1}\nabla^2\Phi(\hat{\boldsymbol{x}})\big), \quad (33)$$

as is well known [30]. To compare ρ_1 and ρ_M for $M > 1$, we provide an illustrative example in the following subsection.

B. One-Parameter Example

We consider a simple one-parameter transmission problem. Suppose the measurement model is:

$$y_i \sim \text{Poisson}\left\{b_i e^{-ax}\right\}, \qquad i = 1, \cdots, N$$

where $a > 0$ and $b_i > 0$, $\forall i$. Assuming $\sum_{i=1}^{N} y_i > 0$, the ML estimate is given by

$$\hat{x} = \left[\frac{1}{a} \log \frac{\sum_{i=1}^{N} b_i}{\sum_{i=1}^{N} y_i}\right]_+$$

where $[x]_+ = \max\{x, 0\}$. Assuming $\hat{x} > 0$, which is very likely for high SNR data, the root-convergence factor ρ_M for an incremental optimization transfer algorithm, TRIOT-MC (see Section III-B for details), is given by (32) with substituting

$$\boldsymbol{T}_m = \frac{\sum_{i \in S_m} b_i}{\sum_{i=1}^{N} b_i} \left(1 - \frac{\sum_{i=1}^{N} y_i}{\sum_{i=1}^{N} b_i}\right), \qquad m = 1, \cdots, M \tag{34}$$

in (30) and (31), where $\{S_m\}_{m=1}^{M}$ is a partition of $\{1, \ldots, N\}$. Fig. 6 shows the mean root-convergence factor $E[\rho_M]$ as a function of the number M of subsets for an example where $x^{\text{true}} = 0.7$, $N = 128$, $a = 1$, and b_i was simulated using pseudorandom uniform variates with mean of 0.5. The mean was approximately computed by replacing y_i in (34) with its mean $b_i e^{-ax^{\text{true}}}$; this approximation is reasonably accurate for high SNR. For example, for $M = 1$, that is, for a nonincremental algorithm, ordinary SPS-MC, the mean of the root-convergence factor is given by

$$E[\rho_1] \approx 1 - e^{-ax^{\text{true}}}.$$

As shown in Fig. 6, for this one-parameter example, the asymptotic convergence rates of incremental optimization transfer algorithms ($M > 1$) are faster than that of the nonincremental one ($M = 1$), and the convergence rate of the incremental one becomes faster as the number M of subsets increases.

REFERENCES

[1] H. M. Hudson and R. S. Larkin, "Accelerated image reconstruction using ordered subsets of projection data," *IEEE Trans. Med. Imag.*, vol. 13, no. 4, pp. 601–609, Dec. 1994.

[2] J. A. Browne and A. R. De Pierro, "A row-action alternative to the EM algorithm for maximizing likelihoods in emission tomography," *IEEE Trans. Med. Imag.*, vol. 15, no. 5, pp. 687–699, Oct. 1996.

[3] C. L. Byrne, "Accelerating the EMML algorithm and related iterative algorithms by rescaled block-iterative methods," *IEEE Trans. Image Processing*, vol. 7, no. 1, pp. 100–109, Jan. 1998.

[4] H. Kudo, H. Nakazawa, and T. Saito, "Convergent block-iterative method for general convex cost functions," in *Proc. of the 1999 Int. Mtg. Fully 3D Im. Recon. in Rad. Nuc. Med.*, 1999, pp. 247–250.

[5] H. Kudo, H. Nakazawa, and T. Saito, "Block-gradient method for image reconstruction in emission tomography," *Trans. IEICE*, vol. J83-D-II, no. 1, pp. 63–73, Jan. 2000, In Japanese.

[6] H. Erdoğan and J. A. Fessler, "Ordered subsets algorithms for transmission tomography," *Phys. Med. Biol.*, vol. 44, no. 11, pp. 2835–2851, Nov. 1999.

[7] A. R. De Pierro and M. E. B. Yamagishi, "Fast EM-like methods for maximum 'a posteriori' estimates in emission tomography," *IEEE Trans. Med. Imag.*, vol. 20, no. 4, pp. 280–288, Apr. 2001.

[8] S. Ahn and J. A. Fessler, "Globally convergent image reconstruction for emission tomography using relaxed ordered subsets algorithms," *IEEE Trans. Med. Imag.*, vol. 22, no. 3, pp. 613–626, May 2003.

[9] Q. Li, E. Asma, and R. M. Leahy, "A fast fully 4D incremental gradient reconstruction algorithm for list mode PET data," in *Proc. IEEE Intl. Symp. Biomedical Imaging*, 2004, pp. 555–558.

[10] P. Khurd, I. T. Hsiao, A. Rangarajan, and G. Gindi, "A globally convergent regularized ordered-subset EM algorithm for list-mode reconstruction," *IEEE Tr. Nuc. Sci.*, vol. 51, no. 3, pp. 719–725, June 2004.

[11] V. M. Kibardin, "Decomposition into functions in the minimization problem," *Automat. Remote Control*, vol. 40, pp. 1311–1323, 1980, Translation of *Avtomatika i Telemekhanika*, vol. 9, pp. 66–79, Sept. 1979.

[12] A. Nedić and D. P. Bertsekas, "Incremental subgradient methods for nondifferentiable optimization," *SIAM J. Optim.*, vol. 12, no. 1, pp. 109–138, 2001.

[13] A. Nedić and D. Bertsekas, "Convergence rate of incremental subgradient algorithms," in *Stochastic Optimization: Algorithms and Applications*, S. Uryasev and P. M. Pardalos, Eds., pp. 263–304. Kluwer, New York, 2000.

[14] K. C. Kiwiel, "Convergence of approximate and incremental subgradient methods for convex optimization," *SIAM J. Optim.*, vol. 14, no. 3, pp. 807–840, 2004.

[15] Y. Censor, "Row-action methods for huge and sparse systems and their applications," *SIAM Review*, vol. 23, no. 4, pp. 444–466, Oct. 1981.

[16] R. Gordon, R. Bender, and G. T. Herman, "Algebraic reconstruction techniques (ART) for the three-dimensional electron microscopy and X-ray photography," *J. Theor. Biol.*, vol. 29, pp. 471–481, 1970.

[17] G. T. Herman and L. B. Meyer, "Algebraic reconstruction techniques can be made computationally efficient," *IEEE Trans. Med. Imag.*, vol. 12, no. 3, pp. 600–609, Sept. 1993.

[18] Y. Censor, P. P. B. Eggermont, and D. Gordon, "Strong underrelaxation in Kaczmarz's method for inconsistent systems," *Numerische Mathematik*, vol. 41, pp. 83–92, 1983.

[19] Y. Censor, D. Gordon, and R. Gordon, "Component averaging: An efficient iterative parallel algorithm for large and sparse unstructured problems," *Parallel Computing*, vol. 27, no. 6, pp. 777–808, May 2001.

[20] S. Sotthivirat and J. A. Fessler, "Relaxed ordered-subsets algorithm for penalized-likelihood image restoration," *J. Opt. Soc. Amer. A*, vol. 20, no. 3, pp. 439–449, Mar. 2003.

[21] H. Erdoğan and J. A. Fessler, "Monotonic algorithms for transmission tomography," *IEEE Trans. Med. Imag.*, vol. 18, no. 9, pp. 801–814, Sept. 1999.

[22] D. P. Bertsekas, "A new class of incremental gradient methods for least squares problems," *SIAM J. Optim.*, vol. 7, no. 4, pp. 913–926, Nov. 1997.

[23] R. Neal and G. E. Hinton, "A view of the EM algorithm that justifies incremental, sparse and other variants," in *Learning in Graphical Models*, M. I. Jordan, Ed., pp. 255–268. Kluwer, Dordrencht, 1998.

[24] I. T. Hsiao, A. Rangarajan, and G. Gindi, "A provably convergent OS-EM like reconstruction algorithm for emission tomography," in *Proc. SPIE 4684, Medical Imaging 2002: Image Proc.*, 2002, pp. 10–19.

[25] A. Gunawardana and W. Byrne, "Convergence of EM variants," Tech. Rep. CLSP Research Note No. 32, ECE Dept., Johns Hopkins University, Feb. 1999.

[26] I. Hsiao, A. Rangarajan, and G. Gindi, "A new convergent MAP reconstruction algorithm for emission tomography using ordered subsets and separable surrogates," in *Proc. IEEE Intl. Symp. Biomedical Imaging*, 2002, pp. 409–412.

[27] A. J. R. Gunawardana, *The information geometry of EM variants for speech and image processing*, Ph.D. thesis, Johns Hopkins University, Baltimore, MD., 2001.

[28] D. Blatt, A. Hero, and H. Gauchman, "An incremental gradient method that converges with a constant step size," *SIAM J. Optim.*, 2004, Submitted.

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

[30] K. Lange, D. R. Hunter, and I. Yang, "Optimization transfer using surrogate objective functions," *J. Computational and Graphical Stat.*, vol. 9, no. 1, pp. 1–20, Mar. 2000.

[31] D. R. Hunter and K. Lange, "Rejoinder to discussion of "Optimization transfer using surrogate objective functions"," *J. Computational and Graphical Stat.*, vol. 9, no. 1, pp. 53–59, Mar. 2000.

[32] K. Lange and R. Carson, "EM reconstruction algorithms for emission and transmission tomography," *J. Comput. Assist. Tomogr.*, vol. 8, no. 2, pp. 306–316, Apr. 1984.

[33] I. T. Hsiao, A. Rangarajan, P. Khurd, and G. Gindi, "An accelerated convergent ordered subsets algorithm for emission tomography," *Phys. Med. Biol.*, vol. 49, no. 11, pp. 2145–2156, June 2004.

[34] J. Qi, R. M. Leahy, S. R. Cherry, A. Chatziioannou, and T. H. Farquhar, "High resolution 3D Bayesian image reconstruction using the microPET small-animal scanner," *Phys. Med. Biol.*, vol. 43, no. 4, pp. 1001–14, Apr. 1998.

[35] R. M. Leahy and J. Qi, "Statistical approaches in quantitative positron emission tomography," *Statistics and Computing*, vol. 10, no. 2, pp. 147–165, Apr. 2000.

[36] 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.

[37] J. A. Fessler and A. O. Hero, "Space-alternating generalized expectation-maximization algorithm," *IEEE Tr. Sig. Proc.*, vol. 42, no. 10, pp. 2664–2677, Oct. 1994.

[38] J. A. Fessler and A. O. Hero, "Penalized maximum-likelihood image reconstruction using space-alternating generalized EM algorithms," *IEEE Trans. Image Processing*, vol. 4, no. 10, pp. 1417–1429, Oct. 1995.

[39] M. W. Jacobson and J. A. Fessler, "Properties of optimization transfer algorithms on convex feasible sets," *SIAM J. Optim.*, 2003, Submitted.

[40] J. Nocedal, "Large scale unconstrained optimization," in *The State of the Art in Numerical Analysis*, I. S. Duff and G. A. Watson, Eds., pp. 311–338. Clarendon Press, Oxford, 1997.

[41] D. R. Hunter and K. Lange, "A tutorial on MM algorithms," *The American Statistician*, vol. 58, no. 1, pp. 30–37, Feb. 2004.

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

[43] W. Byrne and A. Gunawardana, "Comments on "Efficient training algorithms for HMMs using incremental estimation"," *IEEE Trans. Speech Audio Processing*, vol. 8, no. 6, pp. 751–754, Nov. 2000.

[44] D. G. Luenberger, *Linear and nonlinear programming*, Addison-Wesley, Massachusetts, 2 edition, 1984.

[45] M. S. Bazaraa, H. D. Sherali, and C. M. Shetty, *Nonlinear programming: Theory and algorithms*, Wiley, New York, 1993.

[46] J. A. Fessler, "Statistical image reconstruction methods for transmission tomography," in *Handbook of Medical Imaging*,

Volume 2. Medical Image Processing and Analysis, M. Sonka and J. Michael Fitzpatrick, Eds., pp. 1–70. SPIE, Bellingham, 2000.

[47] J. A. Fessler and H. Erdoğan, "A paraboloidal surrogates algorithm for convergent penalized-likelihood emission image reconstruction," in *Proc. IEEE Nuclear Science Symp. Medical Imaging Conf.*, 1998, vol. 2, pp. 1132–1135.

[48] S. Ahn and J. A. Fessler, "Emission image reconstruction for randoms-precorrected PET allowing negative sinogram values," *IEEE Trans. Med. Imag.*, vol. 23, no. 5, pp. 501–601, May 2004.

[49] S. Sotthivirat and J. A. Fessler, "Penalized-likelihood image reconstruction for digital holography," *J. Opt. Soc. Am. A*, vol. 21, no. 5, pp. 737–50, May 2004.

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

[51] K. Lange, "Convergence of EM image reconstruction algorithms with Gibbs smoothing," *IEEE Trans. Med. Imag.*, vol. 9, no. 4, pp. 439–446, Dec. 1990, Corrections, T-MI, 10:2(288), June 1991.

[52] K. Lange and J. A. Fessler, "Globally convergent algorithms for maximum a posteriori transmission tomography," *IEEE Trans. Image Processing*, vol. 4, no. 10, pp. 1430–1438, Oct. 1995.

[53] 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. Comput. Assist. Tomogr.*, vol. 16, no. 5, pp. 804–813, Sept. 1992.

[54] J. A. Fessler, "ASPIRE 3.0 user's guide: A sparse iterative reconstruction library," Tech. Rep. 293, Comm. and Sign. Proc. Lab., Dept. of EECS, Univ. of Michigan, Ann Arbor, MI, 48109-2122, July 1995, Available from `http://www.eecs.umich.edu/~fessler`.

[55] A. Rahmim, M. Lenox, A. J. Reader, C. Michel, Z. Burbar, T. J. Ruth, and V. Sossi, "Statistical list-mode image reconstruction for the high resolution research tomograph," *Phys. Med. Biol.*, vol. 49, pp. 4239–4258, Aug. 2004.

[56] S. Ahn, *Convergent algorithms for statistical image reconstruction in emission tomography*, Ph.D. thesis, Univ. of Michigan, Ann Arbor, MI, 48109-2122, Ann Arbor, MI., 2004.

[57] W. Zangwill, *Nonlinear programming, a unified approach*, Prentice-Hall, NJ, 1969.

[58] R. R. Meyer, "Sufficient conditions for the convergence of monotonic mathematical programming algorithms," *J. Comput. System. Sci.*, vol. 12, no. 1, pp. 108–121, 1976.

[59] A. M. Ostrowski, *Solution of equations in Euclidean and Banach spaces*, Academic, New York, 3 edition, 1973.

[60] J. M. Ortega and W. C. Rheinboldt, *Iterative solution of nonlinear equations in several variables*, Academic, New York, 1970.

TABLE I

OUTLINE FOR INCREMENTAL OPTIMIZATION TRANSFER ALGORITHMS. THE RIGHT SIDE OF (T-2) IS DUE TO (4) AND (5).

Initialize $\boldsymbol{x}^0, \bar{\boldsymbol{x}}_1^0, \ldots, \bar{\boldsymbol{x}}_M^0 \in \mathcal{X}$

for $n = 0, \ldots, n_{\text{iter}} - 1$

for $m = 1, \ldots, M$

$$\boldsymbol{x}^{\text{new}} = \arg\max_{\boldsymbol{x} \in \mathcal{X}} F\left(\boldsymbol{x}; \bar{\boldsymbol{x}}_1^{n+1}, \ldots, \bar{\boldsymbol{x}}_{m-1}^{n+1}, \bar{\boldsymbol{x}}_m^n, \bar{\boldsymbol{x}}_{m+1}^n, \ldots \bar{\boldsymbol{x}}_M^n\right) \quad \text{(T-1)}$$

$$\bar{\boldsymbol{x}}_m^{n+1} = \boldsymbol{x}^{\text{new}} = \arg\max_{\bar{\boldsymbol{x}}_m \in \mathcal{X}} F\left(\boldsymbol{x}^{\text{new}}; \bar{\boldsymbol{x}}_1^{n+1}, \ldots, \bar{\boldsymbol{x}}_{m-1}^{n+1}, \bar{\boldsymbol{x}}_m, \bar{\boldsymbol{x}}_{m+1}^n, \ldots \bar{\boldsymbol{x}}_M^n\right) \quad \text{(T-2)}$$

end

$$\boldsymbol{x}^{n+1} = \bar{\boldsymbol{x}}_M^{n+1} \quad \text{(T-3)}$$

end

TABLE II

OUTLINE FOR TRANSMISSION INCREMENTAL OPTIMIZATION TRANSFER (TRIOT) ALGORITHM USING MAXIMUM CURVATURE (MC).

Initialize: $\hat{\boldsymbol{x}} = \hat{\boldsymbol{x}}^0 = \left[\mathrm{FBP} \left\{ \log\left(\frac{b_i}{y_i - r_i} \right) \right\}_{i=1}^{N} \right]^+$

Precompute: $d_{mj}^{\mathrm{MC}} = \sum_{i \in S_m} a_{ij} a_i \left[\left(1 - \frac{y_i r_i}{(b_i + r_i)^2} \right) b_i \right]_+$ and $d_j^{\mathrm{PC}} = \frac{1}{M} \sum_{i=1}^{N} a_{ij} a_i c_i^{\mathrm{PC}}, \quad \forall m,\ j$

for each iteration $n = 1, \ldots, n_{\mathrm{iter}}$

for each subset (subiteration) $m = 1, \ldots, M$

$$\hat{l}_i = \sum_{j=1}^{p} a_{ij} \hat{x}_j, \quad \dot{h}_i = \left(1 - \frac{y_i}{b_i e^{-\hat{l}_i} + r_i} \right) b_i e^{-\hat{l}_i}, \quad \forall i \in S_m$$

$$\dot{L}_{mj} = \sum_{i \in S_m} a_{ij} \dot{h}_i, \quad r_{mj} = \frac{2\beta}{M} \sum_{k \in \mathcal{N}_j} w_{jk} \omega_\psi (\hat{x}_j - \hat{x}_k), \quad \forall j$$

$$\bar{x}_{mj} = \hat{x}_j, \quad \forall j$$

if $n \leq n_{\mathrm{iter}}^{\mathrm{OS}}$, perform the following OS-SPS update:

$$\hat{x}_j = \left[\bar{x}_{mj} + \frac{\dot{L}_{mj} - \frac{\beta}{M} \sum_{k \in \mathcal{N}_j} w_{jk} \dot{\psi}(\bar{x}_{mj} - \bar{x}_{mk})}{\max\left\{ d_j^{\mathrm{PC}} + r_{mj},\, \epsilon \right\}} \right]^+, \quad \forall j \tag{T-4}$$

else, perform the following TRIOT-MC update:

$$\hat{x}_j = \left[\frac{\sum_{l=1}^{M} \left[\bar{x}_{lj} \max\left\{ d_{lj}^{MC} + r_{lj},\, \epsilon \right\} + \left(\dot{L}_{lj} - \frac{\beta}{M} \sum_{k \in \mathcal{N}_j} w_{jk} \dot{\psi}(\bar{x}_{lj} - \bar{x}_{lk}) \right) \right]}{\sum_{l=1}^{M} \max\left\{ d_{lj}^{\mathrm{MC}} + r_{lj},\, \epsilon \right\}} \right]^+, \quad \forall j \tag{T-5}$$

end

end

if $n = n_{\mathrm{iter}}^{\mathrm{OS}}$ (the last iteration of OS-SPS), then perform (T-5), **end**

$\hat{\boldsymbol{x}}^n = \hat{\boldsymbol{x}}$

end

Here ϵ is some small positive value; c_i^{PC} is defined in (23); and $[x]^+ \triangleq \mathrm{median}\{0, x, U\}$, which should not be confused with $[x]_+ \triangleq \max\{x, 0\}$.

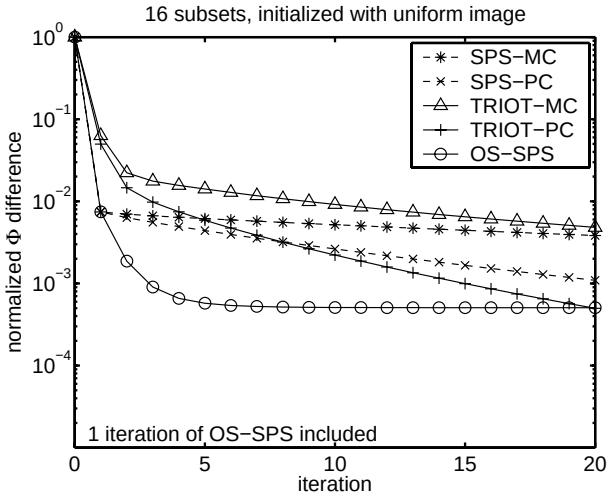


Fig. 1. Comparison of non-OS algorithms (SPS-MC/PC), an OS algorithm (OS-SPS), and incremental optimization transfer algorithms (TRIOT-MC/PC) for 2D attenuation map reconstruction using real PET data. This figure shows $(\Phi(\hat{\boldsymbol{x}}^{\mathrm{PL}}) - \Phi(\hat{\boldsymbol{x}}^n))/(\Phi(\hat{\boldsymbol{x}}^{\mathrm{PL}}) - \Phi(\hat{\boldsymbol{x}}^0))$ versus iteration number where $\hat{\boldsymbol{x}}^{\mathrm{PL}}$ is the PL optimal image. The OS-SPS and TRIOT algorithms used 16 subsets, and TRIOT and SPS algorithms included one initial iteration of OS-SPS. The starting image was a uniform image for all cases.

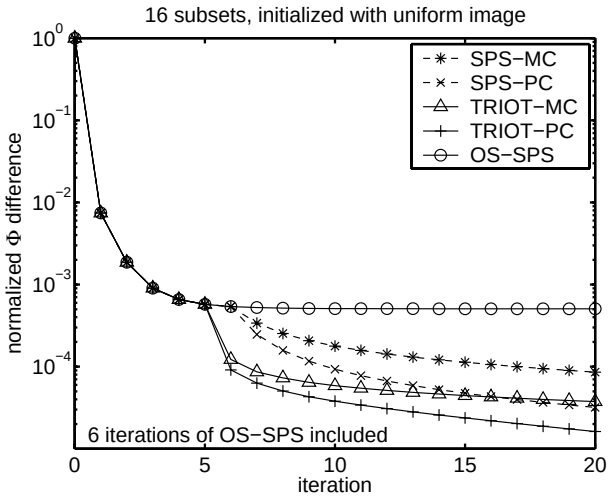


Fig. 2. Same as Fig. 1, but six initial iterations of OS-SPS were included for TRIOT and SPS algorithms.

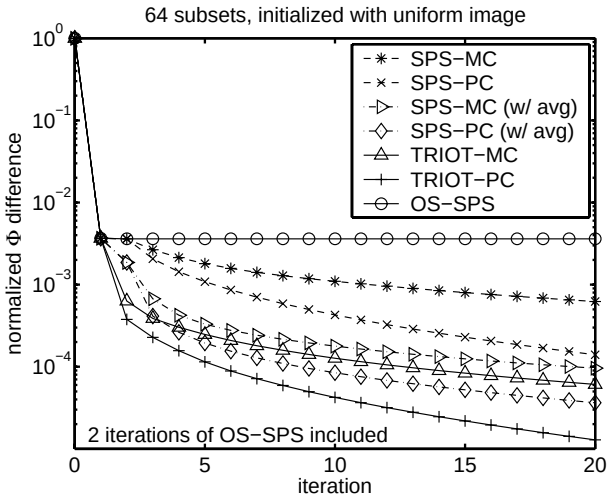


Fig. 3. Comparison of $(\Phi(\hat{\boldsymbol{x}}^{\mathrm{PL}}) - \Phi(\hat{\boldsymbol{x}}^{n}))/(\Phi(\hat{\boldsymbol{x}}^{\mathrm{PL}}) - \Phi(\hat{\boldsymbol{x}}^{0}))$ versus iteration number. For this figure, 64 subsets are used for OS-SPS and TRIOT algorithms, and two iterations of OS-SPS are included initially for TRIOT and SPS algorithms. This figure also shows the performance of SPS algorithms that include averaging 64 subiterates after 2 iterations of OS-SPS.

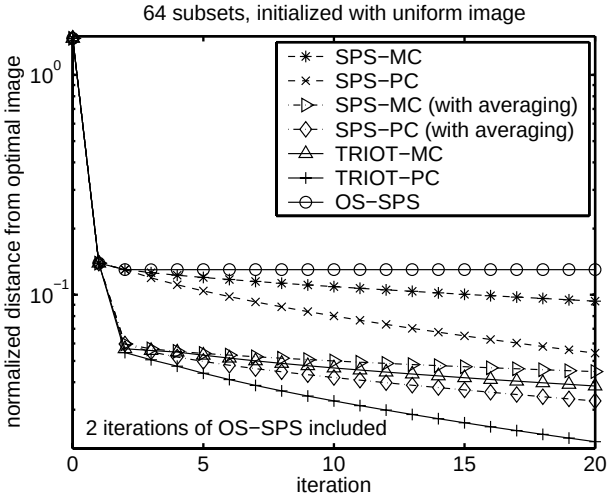
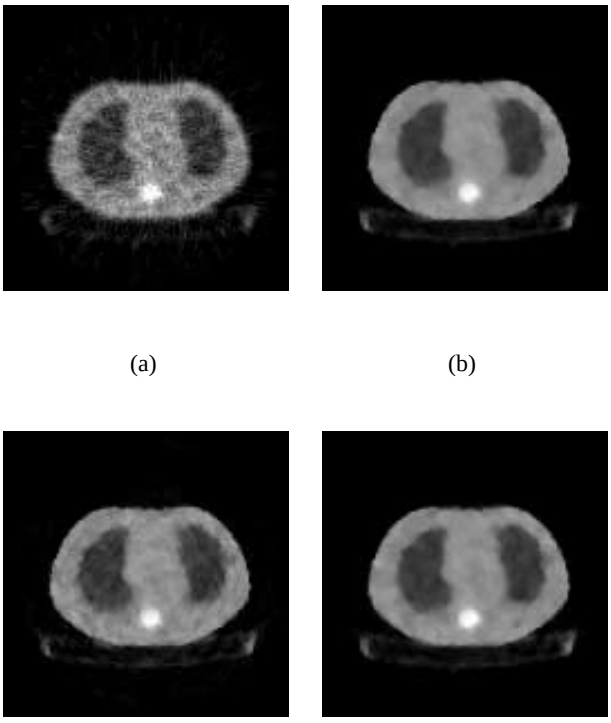


Fig. 4. Same as Fig. 3, but this figure shows a comparison of normalized distance from the optimal image, $\|\hat{\boldsymbol{x}}^{n} - \hat{\boldsymbol{x}}^{\mathrm{PL}}\|/\|\hat{\boldsymbol{x}}^{\mathrm{PL}}\|$, versus iteration number.



(a) (b)

(c) (d)

Fig. 5. Reconstructed attenuation maps. (a) FBP reconstruction. (b) PL estimate image $\hat{\boldsymbol{x}}^{\mathrm{PL}}$ obtained using 30 iterations of OS-SPS with 16 subsets followed by 800 iterations of SPS-OC. (c) PL reconstruction using 20 iterations of OS-SPS with 64 subsets (an image that is one point of a limit cycle). (d) PL reconstruction using 2 iterations of OS-SPS and 18 iterations of TRIOT-PC with 64 subsets.

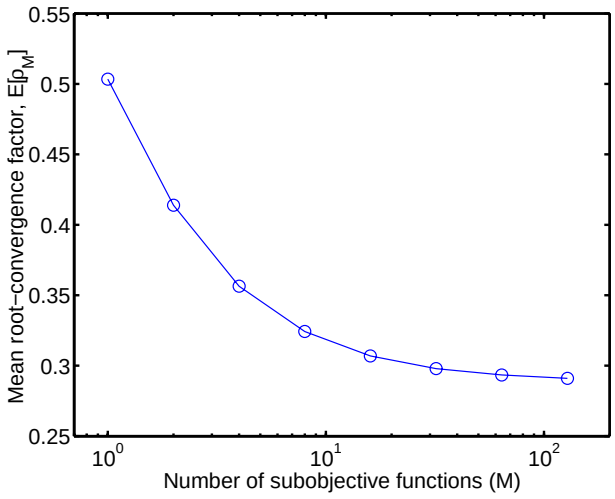


Fig. 6. Comparison of mean root-convergence factors of incremental optimization transfer algorithms (TRIOT-MC) with different numbers M of subobjective functions for a one-parameter transmission problem. Nonincremental ordinary SPS corresponds to the case $M = 1$.