String-averaging expectation-maximization for maximum likelihood estimation in emission tomography

2014 Inverse Problems 30 055003

(http://iopscience.iop.org/0266-5611/30/5/055003)

Inverse Problems **30** (2014) 055003 (20pp) doi:10.1088/0266-5611/30/5/055003

String-averaging expectation-maximization for maximum likelihood estimation in emission tomography

Elias Salomão Helou[1], Yair Censor[2], Tai-Been Chen[3], I-Liang Chern[4,5], Álvaro Rodolfo De Pierro[1], Ming Jiang[6] and Henry Horng-Shing Lu[7]

[1] Department of Applied Mathematics and Statistics, State University of São Paulo, Postal Box 668, São Carlos, SP, Brazil
[2] Department of Mathematics, University of Haifa, Mt. Carmel, Haifa 3190501, Israel
[3] Department of Medical Imaging and Radiological Sciences, I-Shou University, Kaohsiung City, Taiwan 82445, ROC
[4] Department of Applied Mathematics, Center of Mathematical Modeling and Scientific Computing, National Chiao Tung University, Hsin Chu, Taiwan 30010, ROC
[5] Department of Mathematics, National Taiwan University, Taipei, Taiwan 10617, ROC
[6] LMAM, School of Mathematical Sciences, Beijing International Center for Mathematical Research, Peking University, Beijing 100871, People's Republic of China
[7] Institute of Statistics, National Chiao Tung University, 1001 University Road, Hsinchu, Taiwan 30010, ROC

E-mail: elias@icmc.usp.br, yair@math.haifa.ac.il, ctb@isu.edu.tw, chern@math.ntu.edu.tw, alvaro@ime.unicamp.br, ming-jiang@ieee.org and hslu@stat.nctu.edu.tw

Received 8 May 2013, revised 31 January 2014
Accepted for publication 5 February 2014
Published 17 April 2014

Abstract

We study the maximum likelihood model in emission tomography and propose a new family of algorithms for its solution, called string-averaging expectation-maximization (SAEM). In the string-averaging algorithmic regime, the index set of all underlying equations is split into subsets, called 'strings', and the algorithm separately proceeds along each string, possibly in parallel. Then, the end-points of all strings are averaged to form the next iterate. SAEM algorithms with several strings present better practical merits than the classical row-action maximum-likelihood algorithm. We present numerical experiments showing the effectiveness of the algorithmic scheme, using data of image reconstruction problems. Performance is evaluated from the computational cost and reconstruction quality viewpoints. A complete convergence theory is also provided.

Keywords: positron emission tomography (PET), string-averaging, block-iterative, expectation-maximization (EM) algorithm, ordered subsets expectation maximization (OSEM) algorithm, relaxed EM, string-averaging EM algorithm

(Some figures may appear in colour only in the online journal)

1. Introduction

The expectation-maximization (EM) algorithm (see, e.g., [1, chapter 7] or [2]) has become a household tool for maximum likelihood estimation in emission tomography (ET). The original EM algorithm is *simultaneous* since when passing from a current iterate x^k to the next one x^{k+1}, *all* equations $\langle a^i, x\rangle = b_i$, $i = 1, 2, \ldots, m$, in the linear system $Ax = b$ of the underlying problem are used. In contrast to this, there is the row-action maximum likelihood algorithm (RAMLA) (of Browne and De Pierro [3]) which is structurally *sequential* since each iteration k involves *only one* equation $\langle a^{i(k)}, x\rangle = b_{i(k)}$, where $i(k)$ is one of the indices $\{1, 2, \ldots, m\}$ chosen at the kth iteration, from the linear system of the underlying problem. In-between these two structural extremes we have also a block-RAMLA structure [3, page 689] and the ordered subsets EM (OSEM) algorithm (of Hudson and Larkin [4]), both allowing to process in each iteration a 'block' (i.e., a subset) of the m underlying equations, by applying to the equations of the block the simultaneous original EM iterative step. Section 3 brings a deeper discussion on alternative methods and related works.

In this article we propose and experiment with a new variant of the EM algorithm which uses *string-averaging* (SA), thus we call the resulting algorithm the *SAEM algorithm*. In the SA algorithmic regime the index set of the m underlying equations is again split into subsets, now called 'strings', and from a current iterate x^k the algorithm first proceeds sequentially along the indices of each string separately (which can be done in parallel) and then the endpoints of all strings are averaged to form the next iterate x^{k+1}. This is in stark contrast with how the above mentioned OSEM and block-RAMLA treat the equations of a block. Full details regarding the algorithm will be given in section 4.

We advocate SAEM as a theoretically sound way to provide better algorithmic behaviour in ET image reconstruction. In order to support our claims, simulation studies performed with the intention to evaluate image quality and computational effort are reported in section 6 and a theoretical study of the convergence properties of the method can be found in section 5. Next section introduces the fundamentals of the problem and section 7 brings our concluding remarks.

2. ET image reconstruction: the problem, approaches and related works

Positron Emission Tomography (PET) and *Single Photon Emission Tomography* (SPECT) are ideal modalities for in vivo functional imaging. Thus, ET can be used to monitor the effects of therapy inside the living body or in oncological studies. The image reconstruction problem in ET can be fully-discretized and modeled by a system of linear equations [10]:

$$Ax = b \tag{1}$$

where the measured data is $b = (b_i)_{i=1}^m \in \boldsymbol{R}^m$, the image vector is $x = (x_j)_{j=1}^n \in \boldsymbol{R}^n$, and A is an $m \times n$ real matrix, and the problem is to estimate the image vector x from the measured data b. Solution of this system by classical direct elimination methods will likely be impractical because of the huge data and image dimensions, ill-posedness of A coupled with noisy and/or incomplete data b, unstructured imaging matrix A, among several other well-known reasons [10, 11].

Instead, iterative methods, especially row-action methods, such as the algebraic reconstruction technique (ART) [17, 18], are usually applied for computationally efficient high-quality reconstruction [16]. Iterative algorithms have been important in this field because of their superior reconstructions over analytical methods in many instances, and their ability to handle the very large and sparse data associated with fully-discretized image reconstruction, see, e.g., [5, 11–13]. For reviews see, e.g., [14] and [15].

Another key advantage of iterative methods is their flexibility. For example, least-squares and I-divergence optimization approaches are commonly used for image reconstruction. In these approaches, the true image is estimated by the minimizer of an objective function, i.e., as a solution of a convex optimization problem. The least-squares functional is given by

$$L_{\mathrm{LS}}(x) := \tfrac{1}{2}\|b - Ax\|^2, \quad \forall x \in \boldsymbol{R}^n. \tag{2}$$

ART is a prominent algorithm in this approach, which can be turned to be efficient for ET [16] even though it was originally proposed for x-ray tomography [17, 18]. The I-divergence (also called the Kullback–Leibler distance) functional is given by

$$L_{\mathrm{KL}}(x) := I(b, Ax) = \sum_{i=1}^{m} b_i \log \frac{b_i}{\langle a^i, x\rangle} + \langle a^i, x\rangle - b_i, \quad \forall x \in \boldsymbol{R}^n_+, \tag{3}$$

where the vector $a^i = \left(a^i_j\right)_{j=1}^n$ is the transpose of the ith row of A. From a statistical perspective, the least-squares approach is equivalent to finding an image as the maximum likelihood estimate from data with Gaussian noise, while the I-divergence approach does the same with Poissonian noise.

For the image reconstruction problem in ET, the I-divergence approach is more appropriate because Poissonian noise is dominant. The following iterative maximum likelihood expectation maximization recursion, termed the MLEM (or the classical EM) algorithm, is popular since the 1980s [9] for the minimization of (3):

$$x_j^{k+1} := x_j^k \frac{1}{\sum_{i=1}^m a^i_j} \sum_{i=1}^{m} a^i_j \frac{b_i}{\langle a^i, x^k\rangle}, \quad \text{for} \quad j = 1, 2, \ldots, n. \tag{4}$$

It was first proposed independently by Richardson [19] and Lucy [20] and for that reason is still known as the Richardson–Lucy (RL) algorithm in applications to astronomy and microscopy. It was later rediscovered in [7] and in [8] by applying the general EM framework from statistics [21, 22] to the ET reconstruction problem.

In addition to the above two approaches, there are those which utilize other Bregman divergences (distances) and the Bayesian framework for image reconstruction [10, 11, 31], of which the above least-squares and I-divergence approaches are special cases. In fact, SAEM can be readily generalized to cover these cases, because our analysis of convergence applies to the minimization of a general convex function over the nonnegative orthant.

From yet another perspective, the image reconstruction task can also be viewed as a *convex feasibility problem* (CFP) see [14]. Consequently the method based on projection onto convex sets (POCS) provides significant inspiration when developing iterative reconstruction algorithms, see, e.g., [11, chapter 5] or [23]. There have been various iterative projection algorithms based on different kinds of projections such as the Euclidean, oblique and Bregman projections. Among them, the string-averaging projections (SAP) algorithmic scheme has attracted attention recently since its presentation in [24]. Therefore, the present paper brings an algorithmic bridge for this SA scheme, going from the set theoretic to the maximum likelihood framework.

3. EM-type algorithms

3.1. The classical EM algorithm

For A, a^i, b, and b_i, as defined above for $i \in I := \{1, 2, \ldots, m\}$, we define $R^i_{\mathrm{EM}} : \boldsymbol{R}^n \to \boldsymbol{R}^n$ as the ith row operator for the EM algorithm,

$$R^i_{\mathrm{EM}}(x) := \frac{b_i}{\langle a^i, x\rangle} x. \tag{5}$$

For any index subset $B \subseteq I$, let $|B|$ denote the number of elements in B and let R_B be the averaging operator $R_B : \boldsymbol{R}^{|B|\times n} \to \boldsymbol{R}^n$ defined componentwise as

$$(R_B(\{y^i\}_{i\in B}))_j := \frac{1}{\sum_{i=1}^m a^i_j} \sum_{i\in B} a^i_j y^i_j, \quad \text{for} \quad j = 1, 2, \ldots, n. \tag{6}$$

The classical EM algorithm and many of its variants for image reconstruction are simultaneous algorithms, see, e.g., [11, section 1.3] on the classification of iterative algorithms from the perspective of the algorithmic parallelization strategy. In pseudo-code, the classical EM algorithm, that uses (4), is as follows:

Algorithm 3.1. *The classical EM algorithm*

```
for  $k = 0, 1, \ldots$
    for  $i \in I$
        $x^{k+1,i} := R^i_{EM}(x^k)$,                (7)
    end
    $x^{k+1} := R_I(\{x^{k+1,i}\}_{i\in I})$.        (8)
end
```

It first executes simultaneously (in parallel) row-action operations on all rows from the same current iterate x^k. Then the intermediate iteration vectors $\{x^{k+1,i}\}_{i=1}^m$ are combined by the averaging operator R_I to obtain the next iterate.

3.2. The block iterative EM algorithm

The classical EM algorithm can be accelerated by its block-iterative version [4, 25, 26]. A block-iterative version of a simultaneous algorithm, see [11, section 1.3], employs the algorithmic operators R^i_{EM} and R_B as the simultaneous algorithm does, but it first breaks up the index set $I = \{1, 2, \ldots, m\}$ into 'blocks' of indices so that for $t = 1, 2, \ldots, T$, the *block* B_t is a subset of I of the form

$$B_t := \left\{i^t_1, i^t_2, \ldots, i^t_{m(t)}\right\}. \tag{9}$$

with $m(t)$ denoting the number of elements in B_t. The block-iterative version of the classical EM algorithm works as follows.

Algorithm 3.2. The block-iterative EM algorithm

```
for  $k = 0, 1, \dots$
     $t = t(k) \in \{1, 2, \dots, T\}$                         (10)
     for  $i \in B_t$
          $x^{k+1,i} := R^i_{EM}(x^k)$,                         (11)
     end
     $x^{k+1} := R_{B_t}(\{x^{k+1,i}\}_{i \in B_t})$.             (12)
end
```

At the kth iteration, the active block B_t is determined by the *control sequence* $t = t(k)$. Then, for each block the algorithm performs a simultaneous step as if the classical EM method was applied to this block alone and the iteration is updated. In the literature of maximum likelihood reconstruction for ET, block-iterative EM algorithms are called OSEM algorithms as they were named in the work of Hudson and Larkin [4] see also [30, 33]. For block-iterative projection methods see, e.g., [27–29].

3.3. EM-type algorithms with relaxation

The OSEM algorithm has an improved experimental convergence rate, but may not converge when the system (1) is inconsistent [3]. This can be resolved by introducing relaxation as in the least-squares approach [35]. The first block-iterative EM algorithm with relaxation is the 'row-action maximum likelihood algorithm' (RAMLA) in [3]. See [34] for row-action methods in general. After statistical study of the noise propagation from the projection data to the reconstructed image, a modified version of the RAMLA, called 'dynamic RAMLA' (DRAMA), using variable relaxation parameters with blocks, was proposed in [36]. A formula extending RAMLA and DRAMA was proposed in [37] as follows,

$$x^{k+1} = x^k - \lambda_{k,t} D(x^k) \nabla L^{B_t}_{\mathrm{KL}}(x^k), \tag{13}$$

where $\lambda_{k,t}$ are stepsize parameters, $\nabla L^{B_t}_{\mathrm{KL}}$ is the gradient of the partial negative log-likelihood

$$L^{B_t}_{\mathrm{KL}}(x) := \sum_{i \in B_t} b_i \log \frac{b_i}{\langle a^i, x \rangle} + \langle a^i, x \rangle - b_i, \tag{14}$$

and the pre-conditioner D is given by

$$D(x) := \mathrm{diag}\left\{ \left(\frac{x_j}{p_j} \right) \middle| 1 \leqslant j \leqslant n \right\}, \tag{15}$$

with positive constants $p_j > 0$. Possible choices of p_j are

$$p_j = \max_{1 \leqslant t \leqslant T} \left\{ \sum_{i \in B_t} a^i_j \right\}, \tag{16}$$

or

$$p_j = \frac{\sum_{i=1}^m a^i_j}{T}. \tag{17}$$

Convergence results established in [37] require that the following conditions are met.

Condition A: $x^0 \in \boldsymbol{R}^n_+$, $b \in \boldsymbol{R}^m_+$, $A \in \boldsymbol{R}^{m\times n}_+$, $\sum_{t=1}^T L^{B_t}_{\mathrm{KL}}(x) = L_{\mathrm{KL}}(x)$.

Condition B: rank $(W(x)^{\frac{1}{2}}A) = n$, where $W(x)^{\frac{1}{2}}$ is the component-wise square root of the diagonal matrix

$$W(x) = \mathrm{diag}\left\{\left(\frac{b_i}{\langle a^i, x\rangle^2}\right)\middle| 1 \leqslant j \leqslant n\right\}. \tag{18}$$

This is equivalent to the strict convexity of the function $L_{\mathrm{KL}}(x)$.

Condition C: $0 < \lambda_{k,t} \leqslant \lambda$, where λ is chosen such that

$$\lambda < \min\left\{\frac{p_j}{\sum_{i\in B_t} a^i_j}\middle| 1 \leqslant j \leqslant n, 1 \leqslant t \leqslant T\right\}, \tag{19}$$

and, denoting $\lambda_k = \lambda_{k,1}$, such that

$$\sum_{k=0}^{\infty}\lambda_k = \infty; \quad \sum_{k=0}^{\infty}\lambda_k^2 < \infty; \quad \sum_{k=0}^{\infty}\left|\lambda_k - \lambda_{k,t}\right| < \infty; \tag{20}$$

$$\frac{\lambda_{k,t}}{\lambda_k} \to 1; \text{ and } \frac{\lambda_k}{\lambda_{k+1}} < \text{constant}. \tag{21}$$

The above assumptions on the relaxation parameters are very general and cover several situations of interest [32, 36, 40, 41]. If each block size is one, i.e., each block corresponds to one equation, then by using (13) and (15), we get the relaxed row-action iteration of RAMLA:

$$x_j^{k+1} = x_j^k + \lambda_{k,i}\frac{a^i_j}{p_j}\left(\frac{b_i}{\langle a^i, x^k\rangle} - 1\right)x_j^k, \text{ for } \quad j = 1, 2, \ldots, n. \tag{22}$$

4. The string-averaging scheme

4.1. The string-averaging prototypical scheme

The SA algorithmic regime was originally formulated in [24] in general terms and applied there for solving the CFP with iterative projection algorithms, see, e.g., [24, 42, 43]. In the SA paradigm, the index set $I = \{1, 2, \ldots, m\}$ is split into 'strings' of indices. From the current iterate x^k, certain algorithmic operators (we shall call them *step operators* in the following) are applied sequentially along the indices of each string and the end-points of all strings are then combined by an additional algorithmic operator (which we name the *combination operator* from now on) to yield the next iteration vector.

To define SA algorithms precisely, the same decomposition as in (9) for the index set I is utilized. The *index set*

$$S_t := \left\{i^t_1, i^t_2, \ldots, i^t_{m(t)}\right\} = \left\{i^t_s \middle| 1 \leqslant s \leqslant m(t)\right\} \tag{23}$$

is now serving as the *string* of indices in the current context, for $t = 1, 2, \ldots, T$. Viewed like that, strings and blocks are just names for index subsets $B_t \subseteq I$ or $S_t \subseteq I$ for $t = 1, 2, \ldots, T$. Interleaving of strings and blocks is possible, leading to algorithms with a tree-like parallelism structure, whose convergence properties can be almost directly devised from our analysis below.

Let us consider a set $Q \subset \boldsymbol{R}^n$ and family of operators $\{R^i\}_{i=1}^m$ mapping Q into itself, and an additional operator R which maps Q^T (i.e., the product of T copies of Q) into Q. In the SA paradigm these operators $\{R^i\}_{i=1}^m$ are the step operators and the operator R serves as the combination operator. The SA prototypical scheme is as follows.

Algorithm 4.1. The string-averaging prototypical scheme [24]

Initialization $x^0 \in Q$ *is an arbitrary starting point.*

Iterative Step *Given the current iterate* x^k,

(i) *for all* $t = 1, 2, \dots, T$, *compute in parallel as follows: apply successively the step operator along the string* S_t,

$$x^{k+1,t} := R^{i^t_{m(t)}} \circ \dots \circ R^{i^t_2} \circ R^{i^t_1}(x^k), \tag{24}$$

(ii) *apply the combination operator*

$$x^{k+1} := \mathcal{R}\big(\{x^{k+1,i}\}_{t=1}^T\big). \tag{25}$$

For every $t = 1, 2, \dots, T$, this algorithmic scheme first applies to x^k successively the step operators R^i whose indices i belong to the tth string S_t. This can be done in parallel for all strings because the jobs of going along each string from (one and the same current iterate) x^k to the end-point $x^{k+1,t}$ are independent. Then the combination operator $\mathcal{R}$ maps all end-points onto the next iterate x^{k+1}. The iteration from x^k to x^{k+1} is called one cycle of iteration, whereas the iteration from x^k to $x^{k+1,t}$ is called the tth sub-iteration in the kth cycle. Notice that we can always obtain from this framework a *fully-sequential* algorithm by the choice $T = 1$ and $S_1 = I$ or a *fully-simultaneous* algorithm by the choice $T = m$ and $S_t = \{t\}$, $t = 1, 2, \dots, T$.

4.2. The string-averaging EM algorithm

Here we merge the SA algorithmic structure described above with the maximum likelihood estimator in order to create the new string-averaging EM (SAEM) algorithm. To this end we adopt the row-action operation of RAMLA as step operators (22), namely:

$$\left(R^i_{\lambda_{k,i}}(x^k)\right)_j := x^k_j + \lambda_{k,i}\frac{a^i_j}{p_j}\left(\frac{b_i}{\langle a^i, x^k\rangle} - 1\right)x^k_j, \text{ for } \quad j = 1, 2, \dots, n. \tag{26}$$

To combine end-points of strings we use convex combinations, thus, our algorithm takes the following form.

Algorithm 4.2. The *string-averaging-EM* (SAEM)

Initialization *Choose parameters* p_j, *for* $j = 1, 2, \dots, n$, *and parameters* $\lambda_{k,i}$. *Choose* $x^0 > 0$ *as an arbitrary initial vector, construct a family of strings* $\{S_t\}_{t=1}^T$, *choose a weight system* $\{w_t\}_{t=1}^T$ *such that* $w_t > 0$ *for all* $t = 1, 2, \dots, T$, *and* $\sum_{t=1}^T w_t = 1$.

Iterative Step *Given the current iterate* x^k,

(i) *for all* $t = 1, 2, \dots, T$, *compute (possibly in parallel) as follows: apply successively the RAMLA row-action iterative step given by (26) along the string* S_t,

$$x^{k+1,t} := R^{S_t}(x) := R^{i^t_{m(t)}} \circ \dots \circ R^{i^t_2} \circ R^{i^t_1}(x^k), \tag{27}$$

where the dependence of each step operator in (27) on the relaxation parameter λ_{k,i^t_s}, $s = 1, 2, \dots, m(t)$ *has been left out for clarity.*

(ii) *then combine the end-points by*

$$x^{k+1} = \sum_{t=1}^T w_t x^{k+1,t} \tag{28}$$

using the weights system $\{w_t\}_{t=1}^T$.

5. Theoretical justification

In this section we provide a general justification for why SAEM algorithms present a convergent behavior. We base our proofs on the fact that the convex combination operator, used in the averaging step, preserves certain asymptotic characteristics of the step operator and, therefore, we can expect convergence whenever the step operator comes from a convergent algorithm as is the case here.

Incremental algorithms [38], such as those used for the stringing by the step operator of the SAEM algorithm, satisfy an approximation given by

$$R^{S_t}(x, \lambda) = x - \lambda D(x)\nabla L^{S_t}(x) + O(\lambda^2), \tag{29}$$

according to proposition 5.3 below. In the current theory, the general form (29) allows us to prove convergence under suitable hypotheses for non-averaged algorithms, mainly because $L^{S_t} = L$ and, therefore, for such iterations we have:

$$x^{k+1} = x^k - \lambda_k D(x^k)\nabla L(x^k) + O\big(\lambda_k^2\big). \tag{30}$$

But any conclusion drawn from equation (30) should hold as well for a properly averaged algorithm, by replacing L by the following $\tilde{L}$:

$$\tilde{L}(x) := \sum_{t=1}^{T} w_t L^{S_t}(x), \tag{31}$$

where $w_t > 0$ are the weights. For the SAEM algorithm 4.2 with $\sum w_i = 1$, we have

$$x^{k+1} = x^k - \lambda_k D(x^k)\nabla \tilde{L}(x^k) + O\big(\lambda_k^2\big). \tag{32}$$

If the strings S_t are pairwise disjoint and such that $\bigcup_{t=1}^{T} S_t = \{1, 2, \ldots, m\}$ and $w_t = 1/T$, we have

$$x^{k+1} = x^k - \frac{\lambda_k}{T} D(x^k)\nabla L(x^k) + O\big(\lambda_k^2\big) \tag{33}$$

for the averaged iteration, and convergence will be toward the optimizer of $\tilde{L} = L$. In order to make the discussion more precise, from now on we consider the following general form of a string-averaging algorithm:

Algorithm 5.1. *General string-averaging algorithm*

```
for  $k = 0, 1, \ldots$
    for  $t = 1, 2, \ldots, T$
        $x^{k+1,t} := R^{S_t}(x^k, \lambda_k),$                (34)
    end
    $x^{k+1} := \sum_{t=1}^{T} \omega_t x^{k+1,t}$               (35)
end
```

By considering this algorithm we handle step operators R^{S_t} of a rather general form, but stick to the concrete realization of the combination operator. For this kind of iteration we have the following result, wherein Γ is a real nonnegative function of λ.

Proposition 5.2. *Suppose that $\omega_t > 0$ and $\sum_{t=1}^{T} \omega_t = 1$ in algorithm 5.1, and that for every $t \in \{1, 2, \ldots, T\}$ the following equality holds for all $k \geqslant 0$,*

$$R^{S_t}(x^k, \lambda_k) = x^k - \lambda_k D(x^k)\nabla L^{S_t}(x^k) + O(\Gamma(\lambda_k)). \tag{36}$$

Then we have

$$x^{k+1} = x^k - \lambda_k D(x^k)\nabla \tilde{L}(x^k) + O(\Gamma(\lambda_k)), \tag{37}$$

where $\tilde{L}$ is defined in (31).

Proof. The result is verified by observing that

$$\begin{aligned} x^{k+1} &= \sum_{t=1}^{T} \omega_t R^{S_t}(x^k, \lambda_k) \\ &= \sum_{t=1}^{T} \omega_t x^k - \lambda_k D(x^k)\nabla \sum_{t=1}^{T} \omega_t L^{S_t}(x^k) + \sum_{t=1}^{T} \omega_t O(\Gamma(\lambda_k)) \\ &= x^k - \lambda_k D(x^k)\nabla \tilde{L}(x^k) + O(\Gamma(\lambda_k)), \end{aligned} \tag{38}$$

where we have used the definition of the algorithm for the first equality, hypothesis (36) for the second, and then applied $\sum_{t=1}^{T} \omega_t = 1$, the definition of $\tilde{L}$, and a trivial property of the O notation to obtain the third equation. □

We now prove the claim that our stringing operators R^{S_t} do satisfy an equation such as (36) with $\Gamma(\lambda_k) = \lambda_k^2$, as long as Lipschitz continuity holds for each parcel of $D(x^k)L^{S_t}(x^k)$ used during each stringing operation, according to the next statement.

Proposition 5.3. *Let* $L^{S_t} = \sum_{i \in S_t} L^i$ *and* $R^{S_t}(x, \lambda) := R_\lambda^{i^t_{m(t)}} \circ \ldots \circ R_\lambda^{i^t_2} \circ R_\lambda^{i^t_1}(x)$, *where each* R_λ^i *for* $i \in \{1, 2, \ldots, m\}$ *is given by*

$$R_\lambda^i(x) := x - \lambda D(x)\nabla L^i(x) \tag{39}$$

For $k = 1, 2, \ldots$, *denote*

$$y^{k,0} := x^k, \tag{40}$$

$$y^{k,s} := R_{\lambda_k}^{i^t_s}(y^{k,s-1}), \quad s = 1, 2, \ldots, m(t), \tag{41}$$

$$x^{k+1,t} := y^{k,m(t)} \tag{42}$$

$$x^{k+1} := \sum_{t=1}^{T} \omega_t x^{k+1,t}. \tag{43}$$

Assume that each $D(\cdot)\nabla L^{i^t_j}(\cdot)$ *is Lipschitz continuous with a Lipschitz constant* M *and bounded by an upper bound* N *on the set* $\{x^k, y^{k,s}\}_{k \in \mathbb{N}, s \in I}$, *then the operator* R^{S_t} *satisfies*

$$R^{S_t}(x^k, \lambda_k) = x^k - \lambda_k D(x^k)\nabla L^{S_t}(x^k) + O\big(\lambda_k^2\big). \tag{44}$$

Proof. Unfolding of the definition of the operator leads to

$$R^{S_t}(x^k, \lambda_k) = x^k - \lambda_k \sum_{s=1}^{m(t)} D(y^{k,s-1})\nabla L^{i^t_s}(y^{k,s-1}). \tag{45}$$

Consider the magnitude of the following difference,

$$\epsilon^k := D(x^k)\nabla L^{S_t}(x^k) - \lambda_k \sum_{s=1}^{m(t)} D(y^{k,s-1})\nabla L^{i^t_s}(y^{k,s-1}), \tag{46}$$

which we can estimate as follows,

$$\|\epsilon^k\| = \left\| \sum_{s=1}^{m(t)} (D(x^k)\nabla L^{i^t_s}(x^k) - D(y^{k,j-1})\nabla L^{i^t_s}(y^{k,j-1})) \right\| \tag{47}$$

$$\leqslant \sum_{s=1}^{m(t)} \|D(x^k)\nabla L^{i^t_s}(x^k) - D(y^{k,j-1})\nabla L^{i^t_s}(y^{k,j-1})\| \tag{48}$$

$$\leqslant \sum_{s=1}^{m(t)} M\|x^k - y^{k,s-1}\|, \tag{49}$$

where the first inequality follows from the triangular inequality, and the latter from Lipschitz continuity. Now we notice that if $s > 1$, then

$$\|x^k - y^{k,s}\| = \left\| \lambda_k \sum_{l=1}^{s} D(y^{k,l-1}) \nabla L^{i_t^j}(y^{k,l-1}) \right\| \leqslant \lambda_k mN \tag{50}$$

because of the boundedness hypothesis. Putting (49) and (50) together leads to $\|\epsilon^k\| \leqslant \lambda_k K$ for some large enough constant K, i.e., $\epsilon^k = O(\lambda_k)$. On the other hand, from (45) and (46)

$$R^{S_t}(x^k, \lambda_k) = x^k - \lambda_k D(x^k) \nabla L^{S_t}(x^k) + \lambda_k \epsilon^k, \tag{51}$$

which implies the assertion, since $\lambda O(\lambda) = O(\lambda^2)$. □

We conclude by making the following connection.

Corollary 5.4. *Under the assumptions of propositions 5.3 and 5.2, algorithm 5.1 satisfies, for all* $k \geqslant 0$,

$$x^{k+1} = x^k - \lambda_k D(x^k) \nabla \tilde{L}(x^k) + O\big(\lambda_k^2\big). \tag{52}$$

Proof. Use proposition 5.3 followed by proposition 5.2. □

So far we have shown how the string-averaging iteration approximates a scaled gradient iteration. From now on we study how to obtain convergence from these approximate steps, which requires conditions on the sequence of parameters $\{\lambda_k\} \subset \boldsymbol{R}_+$. We will need $\sum_{k=1}^{\infty} \lambda_k = \infty$ because we will usually assume some sort of boundedness on $D(x^k)\nabla L(x^k)$ and, consequently, we would never reach optimal points if they happen to be too far away from the initial guess. On the other hand, if we do not impose $\lambda_k \to 0$, the error $O(\lambda_k^2)$ in the approximation does not necessarily vanish and will eventually dominate the computations when $\|D(x^k)\nabla L(x^k)\|$ becomes small. This must happen if we wish to converge to an optimal point since a necessary condition on a solution x^* for the non-negatively constrained problem is $D(x^*)\nabla L(x^*) = 0$. We now show that, with no further hypotheses, at least one subsequence generated by the algorithm is of interest.

We will need the following lemma to prove the next proposition.

Lemma 5.5 (lemma 2, [44])**.** *Assume that* $\{e_k\}$, $\{\nu_k\}$ *and* $\{d_k\}$ *are sequences of nonnegative numbers satisfying for all* $k \geqslant 0$,

$$e_{k+1} \leqslant e_k - \nu_k d_k \tag{53}$$

and that $\sum_{k=0}^{\infty} \nu_k = +\infty$. *Then* 0 *is a cluster point of* $\{d_k\}$.

Proposition 5.6. *Suppose that the algorithm generating* $\{x^k\}_{k\in\mathbb{N}}$ *satisfies (52),* $\sum_{k\in\mathbb{N}} \lambda_k = \infty$, $\lambda_k \to 0^+$, $\tilde{L}$ *is smoothly differentiable and* $\tilde{L}(x^k)$ *is well-defined. Assume also that* $\{x^k\} \subset \boldsymbol{R}_+^n$ *is bounded. Then either* $\lim \tilde{L}(x^k) = -\infty$ *(i.e., the sequence of objective values is unbounded from below) or there is a subsequence* l_k *such that*

$$\lim_{k\to\infty} D(x^{l_k}) \nabla \tilde{L}(x^{l_k}) = 0. \tag{54}$$

Proof. We first estimate how much does the objective function improve at each iteration. By using (52) followed by a first-order Taylor expansion we get:

$$\begin{aligned} \tilde{L}(x^{k+1}) &= \tilde{L}\big(x^k - \lambda_k D(x^k) \nabla \tilde{L}(x^k) + O\big(\lambda_k^2\big)\big) \\ &= \tilde{L}(x^k) - \lambda_k \nabla \tilde{L}(x^k)^T D(x^k) \nabla \tilde{L}(x^k) + O\big(\lambda_k^2\big). \end{aligned} \tag{55}$$

Then we rewrite the above equality as the following inequality, holding for some $M > 0$, by definition of $O(\lambda^2)$:

$$\tilde{L}(x^{k+1}) \leqslant \tilde{L}(x^k) - \lambda_k(\nabla\tilde{L}(x^k)^T D(x^k)\nabla\tilde{L}(x^k) - M\lambda_k). \tag{56}$$

Now suppose that $\nabla\tilde{L}(x^k)^T D(x^k)\nabla\tilde{L}(x^k)$ is bounded from below by a positive β. Then, for k large enough, the factor between parentheses in the second term on the right-hand side of (56) will be nonnegative, satisfying the nonnegativity hypotheses of lemma 5.5. Therefore, if the sequence of objective values is bounded from below, it follows from lemma 5.5 that there is a subsequence l_k such that

$$\lim_{k\to\infty} \{\nabla\tilde{L}(x^{l_k})^T D(x^{l_k})\nabla\tilde{L}(x^{l_k}) + O(\lambda_{l_k})\} = 0. \tag{57}$$

Because $\lambda_k \to 0^+$, we have

$$\lim_{k\to\infty} \nabla\tilde{L}(x^{l_k})^T D(x^{l_k})\nabla\tilde{L}(x^{l_k}) = 0. \tag{58}$$

Consequently, for each $j = 1, 2, \ldots, n$,

$$\lim_{k\to\infty} D_j(x^{l_k})(\partial_j\tilde{L}(x^{l_k}))^2 = 0. \tag{59}$$

By the boundeness assumption, we can assume that, with the same subsequence $\{l_k\}$,

$$\lim_{k\to\infty} D(x^{l_k}) = d, \tag{60}$$

for some $d \in \boldsymbol{R}^{n\times n}$. Hence, we obtain, for each $j = 1, 2, \ldots, n$,

$$\lim_{k\to\infty} (D_j(x^{l_k}))^2(\partial_j\tilde{L}(x^{l_k}))^2 = d_{jj}\cdot 0 = 0. \tag{61}$$

Therefore, for each $j = 1, 2, \ldots, n$,

$$\lim_{k\to\infty} D_j(x^{l_k})\partial_j\tilde{L}(x^{l_k}) = 0, \tag{62}$$

from which the conclusion follows. □

In particular, if it is known beforehand that $\tilde{L}$ is bounded from below, then we have shown that the algorithm does provide, at least, a useful subsequence. This is likely to be the case, since in real optimization problems (like in tomographic reconstruction), one will not be able to obtain an unboundedly good solution. Using the above result with some further assumptions (which include strict convexity) we obtain the following global convergence result.

Corollary 5.7. *Suppose that the assumptions of propositions 5.3 and 5.2 hold, assume also that $\tilde{L}$ is smoothly differentiable and strictly convex and that $\tilde{L}(x^k)$ is well-defined and that $\{x^k\} \subset \boldsymbol{R}^n_+$ is bounded. If $\tilde{L}(x^k)$ converges, then any sequence generated by algorithm 5.1 converges to the solution of*

$$\min_{x\in\boldsymbol{R}^n_+} \quad \tilde{L}(x). \tag{63}$$

Proof. If $\{x^k\}$ is bounded then we may assume, without loss of generality, that the subsequence which we have proven to exist in proposition 5.6 is convergent, say $x^{l_k} \to x^*$. If $x^* \in \boldsymbol{R}^n_+$ and $\tilde{L}$ is convex, then $D(x^*)\tilde{L}(x^*) = 0$ is a necessary and sufficient condition for x^* to be an optimal solution of problem (63).

By continuity, because $\tilde{L}(x^{l_k})$ converges, it must converge to $\tilde{L}(x^*)$. Therefore, the objective value at every accumulation point of $\{x^k\}$ equals $\tilde{L}(x^*)$. However, since we assumed that $\tilde{L}$ is strictly convex, this reasoning implies that all accumulation points of the bounded sequence $\{x^k\}$ are the same, namely x^*. Thus, we have proven that $x^k \to x^*$. □

We now remove some of the hypotheses used in the last corollary by referring to works available in the literature. This leads to our main result, stated below.

Theorem 5.8. *Assume that all components of $x^0 \in \boldsymbol{R}^n_+$ are positive, $0 < \lambda_k \leqslant \lambda$ for some suitably small $\lambda > 0$, $\sum_{k=0}^{\infty} \lambda_k = \infty$ and $\sum_{k=0}^{\infty} \lambda_k^2 < \infty$. Then, if L is the negative Poisson log-likelihood and if $\tilde{L}$ is strictly convex, then any sequence generated by algorithm 5.1 converges to the solution of (63).*

Proof. Boundedness of $\{x^k\}$, for small enough λ_k, is a consequence of [3, proposition 1]. Convergence of $\tilde{L}(x^k)$ may then be obtained by assuming that $D(x)\nabla\tilde{L}(x)$ is Lipschitz continuous on the closure of the convex hull of the iterates and that $\sum_{k=0}^{\infty} \lambda_k^2 < \infty$, as in [32, lemma 3]. Furthermore, we see that, if $\lambda_k \leqslant \lambda$ for some suitably small $\lambda > 0$, it is possible to adapt the ideas leading to [37, corollary 1] to our case, yielding the conclusion that $D(x)\tilde{L}(x)$ is Lipschitz continuous within the closure of the convex hull of the iterates (and subiterates) of algorithm 5.1. Therefore, since $\tilde{L}$ is naturally bounded from below, we can apply corollary 5.7. □

6. Experimental work

This section is devoted to the experimental setup we have used in order to test the practical performance of SAEM algorithms and the results obtained. We will base our conclusions on two different figures of merit, the first is the squared error from the ground truth image, and the second being a total-variation based analysis, which puts on firmer grounds our claims about image quality. In what follows we provide a detailed description of the experimental setup. Subsection 6.1 will report the experimental results.

There are certain sources of pseudo-randomness in the numerical experiments we have performed, such as the choice of strings and the noise simulations, but the presented results are representative of the typical situation, as we have noticed little deviation from this behavior from one run to another of the experiments. Moreover, to enable any interested reader to reproduce our research, the full source code of the numerical algorithms used in this paper is available upon request.

(a) Data generation

In our simulated studies we used the modified Shepp–Logan head phantom [7] in order to compare the quality of images reconstructed by RAMLA and SAEM. Let us define the so-called Radon transform $\mathcal{R}[f]$ of a function $f : \mathbf{R}^2 \mapsto \mathbf{R}$:

$$\mathcal{R}[f](\theta, t) := \int_{\mathbf{R}} f(t(\cos\theta, \sin\theta)^T + s(-\sin\theta, \cos\theta)^T)\, ds. \tag{64}$$

Figure 1 shows a schematic representation of the basic geometry of the Radon transform. The Shepp–Logan phantom is shown, together with its *sinogram*, i.e., its Radon transform presented as an image in the $\theta \times t$ coordinate system.

We have used a parallel beam geometry where the samples were measured at the pairs (θ, t) in the set $\{\theta_1, \theta_2, \dots \theta_v\} \times \{t_1, t_2, \dots, t_r\}$ where $\theta_i = \pi(i-1)/v$ and $t_j = -1+2(j-1)/(r-1)$ for $1 \leqslant i \leqslant v$ and $1 \leqslant j \leqslant r$. That is, we uniformly sample the Radon transform on the box $[0, \pi) \times [-1, 1]$. Data was generated as samples of a Poisson random variable having as parameter the exact Radon transform of the scaled phantom:

$$b_i \sim \text{Poisson}(\kappa\mathcal{R}[f](\theta_{v_i}, t_{r_i})), \quad 1 \leqslant i \leqslant vr, \tag{65}$$

where v_i is one plus the largest integer smaller than $(i-1)/r$, $r_i = i - r(v_i - 1)$, and where the scale factor $\kappa > 0$ is used to control the simulated photon count, i.e., the noise level. We

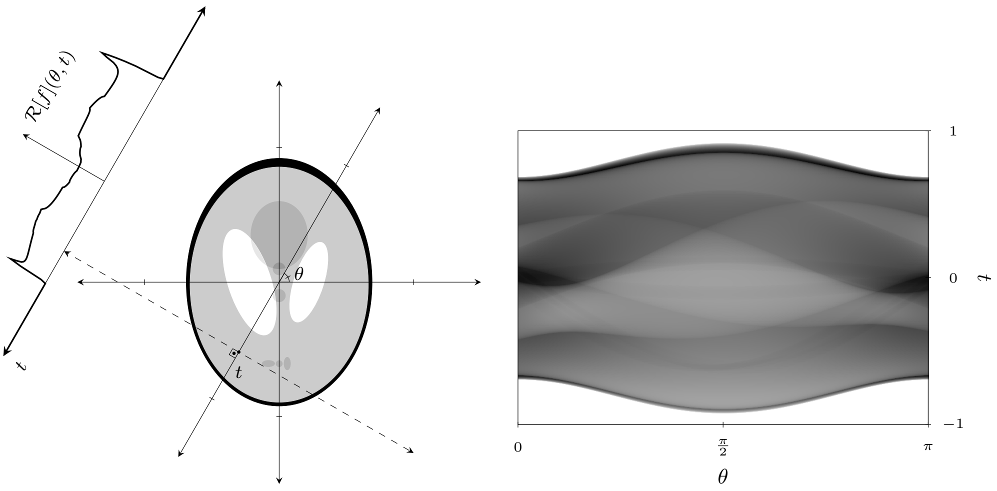

Figure 1. Left: schematic representation of the Radon transform. In the definition, θ is the angle between the normal to the integration path and the vertical axis, while t is the displacement from origin of the line of integration. Right: image of the Radon transform of the image shown on the left in the $\theta \times t$ coordinate system.

will also denote the vector $b^\dagger$ as the one with ideal data: $b_i^\dagger = \kappa\mathcal{R}[f](\theta_{v_i}, t_{r_i})$. In these studies, we have used $v = 288$ and $r = 256$ in order to reconstruct images consisting of 256×256 pixels.

We also experiment using data from a human cardiac SPECT study. In this set of experiments, the data set had $v = 60$ and $r = 64$ and we reconstructed images with 64×64 pixels. No attenuation correction has been applied to the data.

(b) String selection and weights

We denote by SAEM-T an SAEM algorithm with T strings, and the strings were built by randomly shuffling the data and then partitioning it in T contiguous chunks of uniform size, each of which would become a string. It was done like that for two reasons. First, it is often recognized that a random order performs as well, if not better, than most others of practical viability in sequential algorithms. Second, this avoids unfair comparisons that could arise from specific choices of subsets, which could possibly be more suitable to one EM variant than to others.

We wanted to keep the classical maximum likelihood model, so we used non-weighted averaging, that is $\omega_i = 1/T$. More sophisticated weight selection schemes are conceivable, but this subject is beyond the scope of the present paper.

(c) Scaling matrix

In the algorithms that we use, the scaling matrix $D(x)$ is a diagonal matrix with nonzero elements given by x_j/p_j for a fixed, but unspecified, set of weights $p_j > 0$. We have chosen $p_j = \sum_{i=1}^n a_{ij}$. Several other forms where possible, including some dependent on the particular method being used, but we chose to keep the experiments understandable and to have a uniform treatment among the algorithms, rather than to complicate the results introducing another varying parameter.

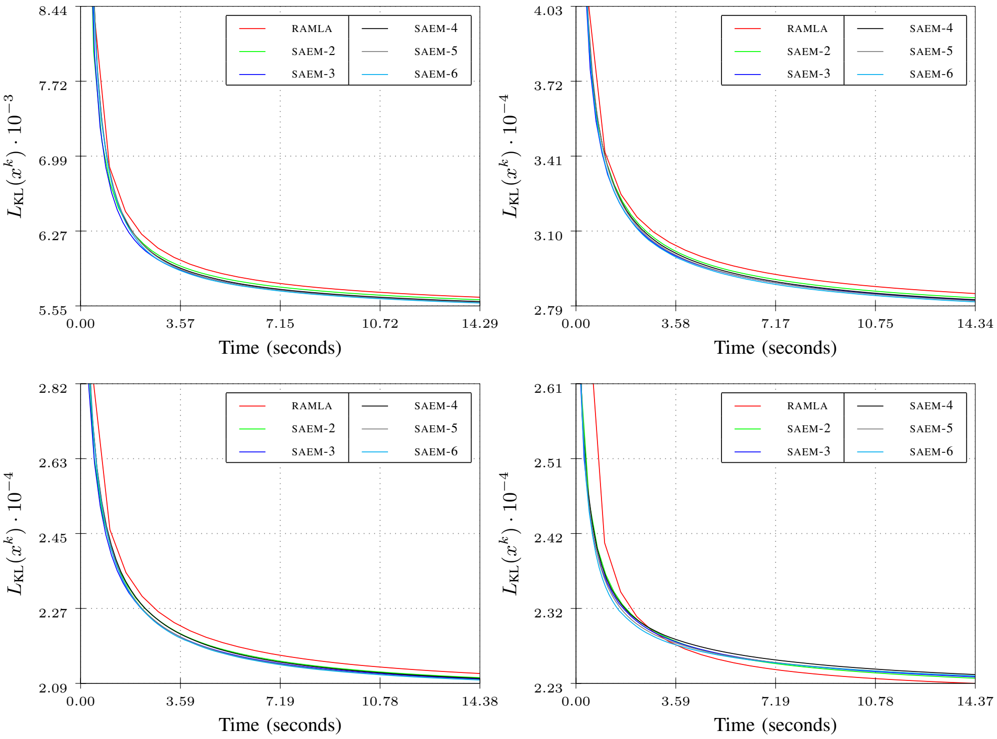

Figure 2. Convergence speed of several SAEM variations under different noise condition. From top left in clockwise direction: 0.00%, 3.96%, 7.94% and 25.03% of relative noise $\|b - b^\dagger\|/\|b^\dagger\|$. Notice that if up to $T_{\max}$ strings can be run in parallel, then at any given time point, SAEM-T has similar log-likelihood values for $T = 1, \ldots, T_{\max}$. Running time was estimated by accumulating high-definition clock information of iteration computation, ignoring input/output time.

(d) Stepsize

We are required to determine the nonsummable stepsize sequence $\{\lambda_k\} \subset \mathbf{R}_+$. We use the rule, for SAEM-T:

$$\lambda_k = \frac{\lambda_0^T}{k^{0.51}/T + 1}, \tag{66}$$

where λ_0^T was selected as the largest value for which SAEM-T would not lead to negative values in the first iteration. This technique for the starting parameter leads to the fastest possible convergence in terms of objective function decrease, and the scaling by the inverse of the number of strings accounts for the $1/T$ factor, caused by the averaging, in (33). Figure 2 shows that this is indeed a well balanced parameter rule, because it leads all SAEM variations to have similar log-likelihood values at any time point, as long as all the strings can be run simultaneously in hardware. It can also be seen that this behavior remains consistent under several noise regimes. This same kind of plot is shown in the left-hand side of figure 6 for the, smaller, real data example, where the comparison is no longer as balanced, because parallelism overhead does not pay off with small images, but in this case computation time is negligible anyway.

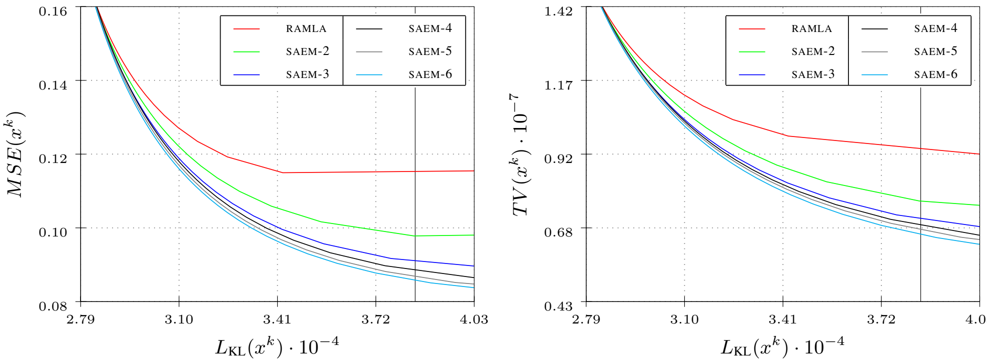

Figure 3. Relative squared error and total variation as functions of the log-likelihood for iterations of SAEM variants. Notice that both the relative error and the total variation values, for a fixed likelihood level, are decreasing functions of the number of strings. The solid vertical lines show the log-likelihood level of the images of figure 4.

(e) Starting image

Initial guess for the algorithms was a uniform image with an expected photon count equal to the data. That is, $x_j^0 = \alpha$ where α is such that

$$\sum_{i=0}^{m} (Ax^0)_i = \sum_{i=0}^{m} b_i. \tag{67}$$

The value of α can be easily obtained as $\alpha = \sum_{i=0}^{m} b_i / \sum_{i=0}^{m} (A\mathbf{1})_i$, where $\mathbf{1}$ is the vector whose components are all equal to 1.

6.1. Image reconstruction analysis

Figure 2 shows that all SAEM-T algorithms present similar convergence speeds when measured as the ratio of objective function decrease and computation time. Notice that at iteration k, SAEM-T has a smaller log-likelihood value than SAEM-$(T+1)$. Hence, more strings means slower algorithms iteration-wise because there is less incrementalism present. It is widely recognized, however, that log-likelihood cannot be used as an image quality measure because over-fitting the image to the data will amplify high frequency components of the noise. This leads to the need for other ways of assessing reconstruction quality.

In our simulated studies, we make use of the mean squared error, which requires full knowledge about the ideal sought-after image:

$$\mathrm{MSE}(x) := \frac{\|x - x^\dagger\|_2^2}{\|x^\dagger\|_2^2}, \tag{68}$$

where $x^\dagger$ is the noise-free discretization of the Shepp–Logan head phantom, properly scaled by κ. Another functional we have used to measure image quality was the total variation [39]:

$$\mathrm{TV}(x) := \sum_{i=1}^{r_1} \sum_{j=1}^{r_2} \sqrt{(x_{i,j} - x_{i,j-1})^2 + (x_{i,j} - x_{i-1,j})^2}, \tag{69}$$

where $r_1 r_2 = n$ and we use the convention $x_{i,j} = x_{i(r_2-1)+j}$ for $i \in \{1, 2, \ldots, r_1\}$ and $j \in \{1, 2, \ldots, r_2\}$ with the boundary condition $x_{0,k} = x_{k,0} = 0$. Unlike the MSE, the total

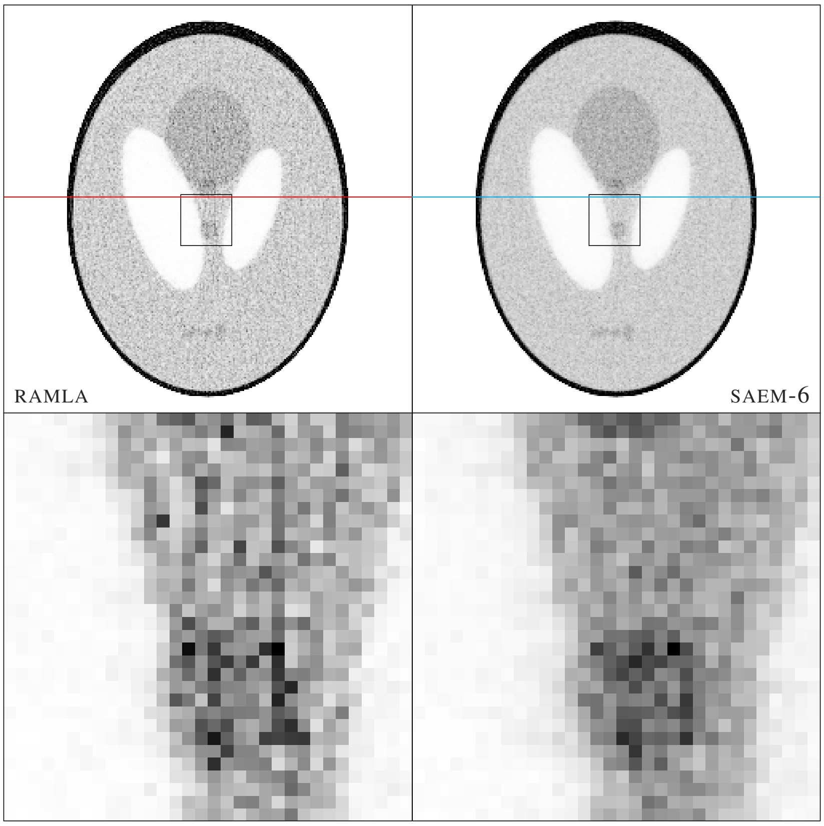

Figure 4. Details of reconstruction obtained by RAMLA (left) and SAEM-6 (right). Both images where these details taken from have, up to linear interpolation errors, the log-likelihood level indicated by the solid vertical lines in figure 3. It is noticeable that the smaller total variance is reflected in both smoother large areas and better resolved sharp transitions. The horizontal solid colored lines show where the profiles of figure 5 where taken from.

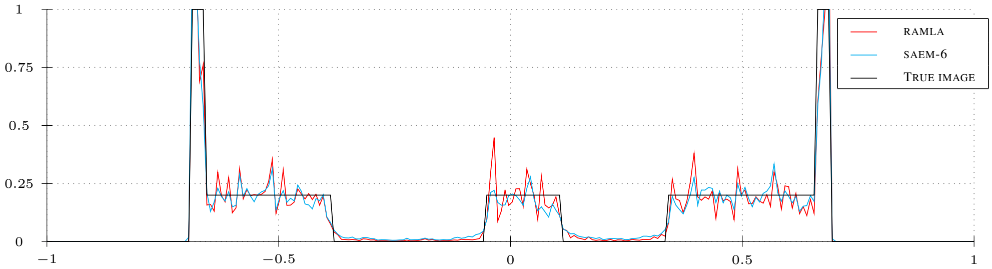

Figure 5. Profile lines from images in figure 4. Notice how SAEM-6 reconstruction presents less overshoot and more smoothness than RAMLA reconstruction.

variation (TV) alone cannot be used as a measure of image quality. However, given two reconstructed images of piecewise constant emission rates with the same Poisson likelihood, the one with the smaller total variation is more likely to have better image quality. Since

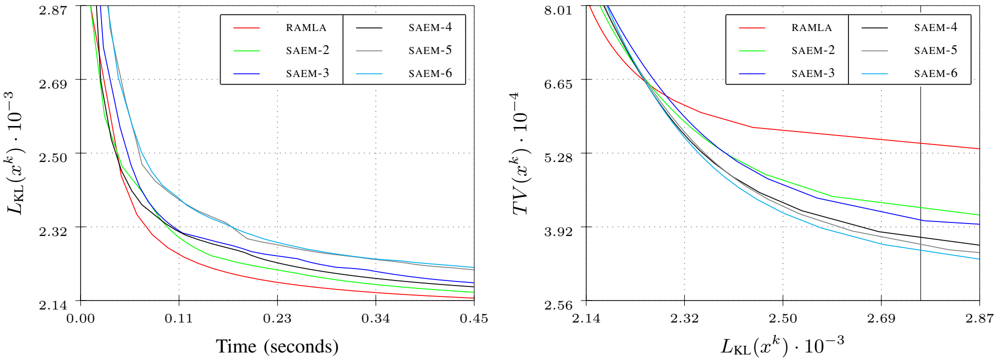

Figure 6. Speed convergence and total variation results for the human SPECT study. The solid line in the graphic at the right indicates the log-likelihood level of the images in figure 7.

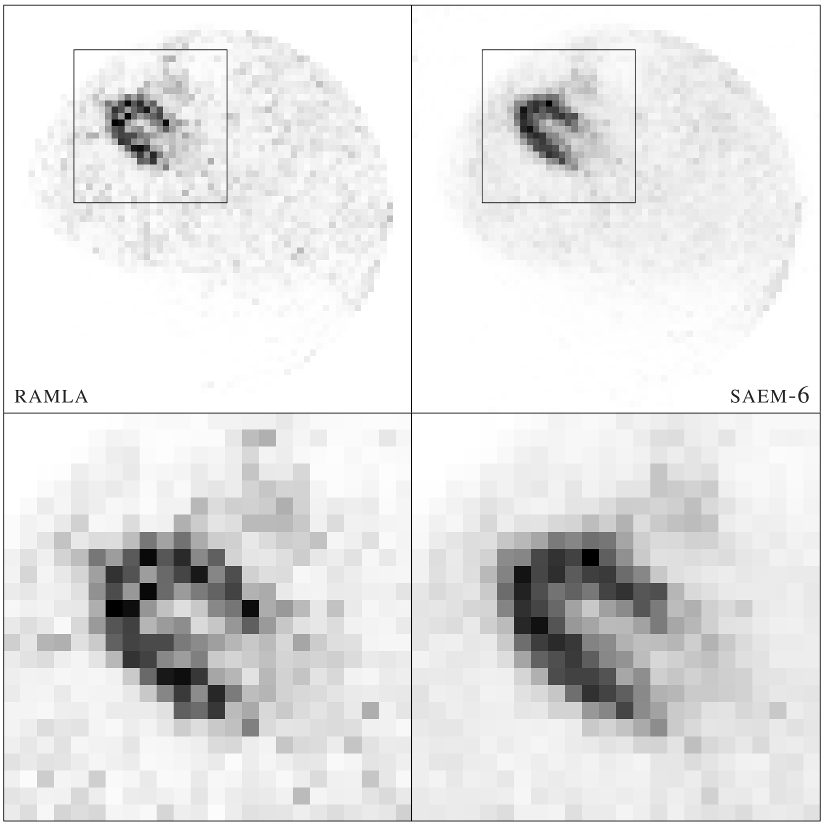

Figure 7. Male human cardiac SPECT study. The log-likelihood level of the images is indicated by the solid lines in the graphic at the right in figure 6. Data kindly provided by Roberto Isoardi—Fundación Escuela de Medicina Nuclear (FUESMEN)—Mendoza, Argentina.

neither the log-likelihood nor the total variation require knowledge of the ideal image, the combination of both can be used also in the evaluation of the SPECT reconstruction.

Figure 3 shows plots of both figures of merit as functions of the log-likelihood for one of the simulated experiments. On the left we have MSE and on the right TV. The remarkable feature of these plots is the fact that the quality of the image obtained by SAEM-T, at any given fixed likelihood level is an increasing function of T, the number of strings. This behavior is unaltered as we vary the noise level. Notice the same effect in the TV versus L_{KL} curve shown in the graphic at the right of figure 6, and that the difference in the total variation translates into significant visual improvement in the real data study (figure 7), even more than it does in the simulated case (figures 4 and 5).

7. Conclusions

We have presented the new string-averaging expectation-maximization family of algorithms. The theoretical convergence properties of the method were studied and preliminary experimental evidence was provided of the technique's suitability for good quality reconstruction.

The theory we have developed here can handle more general regularized objective functions, as they can achieve a better balance between smoothness and adherence to the data than traditional techniques currently used in tomographic scanners.

Future work could focus on comparative evaluation of SAEM against other paradigms, such as OS-EM or block-RAMLA, in light of the TV versus L_{KL} plots, which we have shown to be useful to assess image quality. It is potentially worth investigating *a posteriori* stopping criteria based on such plots, in the spirit of the L-curve [6].

Additional medical task oriented, and statistically based, experimental work is called for to asses the SAEM method's potential in realistic situations.

Acknowledgments

We are sincerely grateful to the anonymous reviewers whose insightful comments encouraged us to extend some of the experimental work in the paper. This resulted in discovering further advantages of the proposed SAEM algorithm in the final version. Work by EH was partially supported by FAPESP grant 2013/16508-3, Brazil. YC was partially supported by the United States-Israel Binational Science Foundation (BSF) grant number 200912 and by US Department of Army award number W81XWH-10-1-0170. TB was partially supported by the National Science Council of the Republic of China, Taiwan (NSC 97-2118-M-214-001-MY2). IC was partially supported by the National Center for Theoretical Sciences (Taipei Office) and the National Science Council of the Republic of China (NSC 99-2115-M-002-003-MY3). AD was partially supported by CNPq grant no 301064/2009-1, Brazil. MJ was partially supported by the National Basic Research and Development Program of China (973 Program) (2011CB809105), National Science Foundation of China (61121002, 10990013, 60325101). HL was supported by grants from National Science Council, National Center for Theoretical Sciences, Center of Mathematical Modeling and Scientific Computing at National Chiao Tung University in Taiwan.

References

[1] Lange K 2004 *Optimization* (New York: Springer)
[2] McLachlan G J and Krishnan T 2008 *The EM Algorithm and Extensions* 2nd edn (Hoboken, NJ: Wiley-Interscience)

[3] Browne J and Pierro A R D 1996 A row-action alternative to the EM algorithm for maximizing likelihood in emission tomography *IEEE Trans. Med. Imaging* **15** 687–99

[4] Hudson H M and Larkin R S 1994 Accelerated image reconstruction using ordered subsets of projection data *IEEE Trans. Med. Imaging* **13** 601–9

[5] Leahy R and Byrne C 2000 Recent developments in iterative image reconstruction for PET and SPECT *IEEE Trans. Med. Imaging* **19** 257–60

[6] Hansen P C 1992 Analysis of discrete ill-posed problems by means of the L-curve *SIAM Rev.* **34** 561–80

[7] Shepp L A and Vardi Y 1982 Maximum likelihood restoration for emission tomography *IEEE Trans. Med. Imaging* **1** 113–22

[8] Lange K and Carson R 1984 EM reconstruction algorithms for emission and transmission tomography *J. Comput. Assist. Tomogr.* **8** 306–16

[9] Vardi Y, Shepp L A and Kaufman L 1985 A statistical model for positron emission tomography *J. Am. Stat. Assoc.* **80** 8–20 (with discussion)

[10] Herman G T 2009 *Fundamentals of Computerized Tomography: Image Reconstruction from Projections* 2nd edn (New York: Springer)

[11] Censor Y and Zenios S A 1997 *Parallel Optimization: Theory, Algorithms and Applications* (New York: Oxford University Press)

[12] Fessler J A 2000 Statistical image reconstruction methods for transmission tomography *Handbook of Medical Imaging* ed M Sonka and J M Fitzpatrick (Bellingham: SPIE) chapter 1 pp 1–70

[13] Jiang M and Wang G 2002 Development of iterative algorithms for image reconstruction *J. X-Ray Sci. Technol.* **10** 77–86

[14] Bauschke H H and Borwein J M 1996 On projection algorithms for solving convex feasibility problems *SIAM Rev.* **38** 367–426

[15] Censor Y, Chen W, Combettes P L, Davidi R and Herman G T 2012 On the effectiveness of projection methods for convex feasibility problems with linear inequality constraints *Comput. Optim. Appl.* **51** 1065–88

[16] Herman G T and Meyer L B 1993 Algebraic reconstruction techniques can be made computationally efficient *IEEE Trans. Med. Imaging* **12** 600–9

[17] Kaczmarz S 1937 Angenäherte Auflösung von Systemen linearer Gleichungen *Bull. Int. Acad. Pol. Sci. Lett.* **A35** 355–7

[18] Gordon R, Bender R and Herman G T 1970 Algebraic reconstruction techniques (ART) for three-dimensional electron microscopy and x-ray photography *J. Theor. Biol.* **29** 471–82

[19] Richardson W H 1972 Bayesin-based iterative method of image restoration *J. Opt. Soc. Am.* A **62** 55–59

[20] Lucy L B 1974 An iterative technique for the rectification of observed distribution *Astron. J.* **79** 745–54

[21] Dempster A P, Laird N M and Rubin D B 1977 Maximum likelihood from incomplete data via the EM algorithm *J. R. Stat. Soc. Ser.* B **39** 1–38

[22] Bertero M, Boccacci P, Desidera G and Vicidomini G 2009 Image deblurring with poisson data: from cells to galaxies *Inverse Problems* **25** 123006

[23] Stark H and Yang Y 1998 *Vector Space Projections: A Numerical Approach to Signal and Image Processing, Neural Nets, and Optics* (New York: Wiley)

[24] Censor Y, Elfving T and Herman G T 2001 Averaging strings of sequential iterations for convex feasibility problems *Inherently Parallel Algorithms in Feasibility and Optimization and their Applications* ed D Butnariu, Y Censor and S Reich (Amsterdam: North-Holland) pp 101–13

[25] Zhu H, Zhou J, Shu H, Li S and Luo L 2004 Improved SAGE algorithm for PET image reconstruction using rescaled block-iterative method *IEMBS'04: 26th Annu. Int. Conf. of the IEEE Engineering in Medicine and Biology Society* vol 1 pp 1353–6

[26] Byrne C 2005 Choosing parameters in block-iterative or ordered subset reconstruction algorithms *IEEE Trans. Image Process.* **14** 321–7

[27] Censor Y and Elfving T 2002 Block-iterative algorithms with diagonally scaled oblique projections for the linear feasibility problem *SIAM J. Matrix Anal. Appl.* **24** 40–58

[28] Censor Y and Herman G T 2002 Block-iterative algorithms with underrelaxed Bregman projections *SIAM J. Optim.* **13** 283–97

[29] Davidi R, Herman G T and Censor Y 2009 Perturbation-resilient block-iterative projection methods with application to image reconstruction from projections *Int. Trans. Oper. Res.* **16** 505–24

[30] Byrne C L 1998 Accelerating the EMML algorithm and related iterative algorithms by rescaled block-iterative methods *IEEE Trans. Image Process.* **7** 100–9

[31] Pierro A R De and Yamagishi M E B 2001 Fast EM-like methods for maximum 'a posteriori' estimates in emission tomography *IEEE Trans. Med. Imaging* **20** 280–8

[32] Ahn S and Fessler J A 2003 Globally convergent image reconstruction for emission tomography using relaxed ordered subset s algorithms *IEEE Trans. Med. Imaging* **22** 613–26

[33] Hsiao I T, Rangarajan A, Khurd P and Gindi G 2004 An accelerated convergent ordered subsets algorithm for emission tomography *Phys. Med. Biol.* **49** 2145–56

[34] Censor Y 1981 Row-action methods for huge and sparse systems and their applications *SIAM Rev.* **23** 444–66

[35] Jiang M and Wang G 2003 Convergence studies on iterative algorithms for image reconstruction *IEEE Trans. Med. Imaging* **22** 569–79

[36] Tanaka E and Kudo H 2003 Subset-dependent relaxation in block-iterative algorithms for image reconstruction in emission tomography *Phys. Med. Biol.* **48** 1405–22

[37] Helou E S and Pierro A R De 2005 Convergence results for scaled gradient algorithms in positron emission tomography *Inverse Problems* **21** 1905–14

[38] Helou E S and Pierro A R De 2010 Incremental subgradients for constrained convex optimization: a unified framework and new methods *SIAM J. Optim.* **20** 1547–72

[39] Rudin L I, Osher S and Fatemi E 1992 Nonlinear total variation based noise removal algorithms *Physica* D **60** 259–68

[40] Wang G and Jiang M 2004 Ordered-subset simultaneous algebraic reconstruction techniques (OS-SART) *J. X-Ray Sci. Technol.* **12** 169–77

[41] Tanaka E and Kudo H 2010 Optimal relaxation parameters of DRAMA (dynamic RAMLA) aiming at one-pass image reconstruction for 3D-PET *Phys. Med. Biol.* **55** 2917–39

[42] Censor Y and Tom E 2003 Convergence of string-averaging projection schemes for inconsistent convex feasibility problems *Optim. Methods Softw.* **18** 543–54

[43] Penfold S N, Schulte R W, Censor Y, Bashkirov V, McAllister S, Schubert K E and Rosenfeld A 2010 Block-iterative and string-averaging projection algorithms in proton computed tomography image reconstruction *Biomedical Mathematics: Promising Directions in Imaging, Therapy Planning and Inverse Problems* ed Y Censor, M Jiang and G Wang (Madison, WI: Medical Physics Publishing) pp 347–67

[44] Trummer M R 1981 Reconstructing pictures from projections: on the convergence of the ART algorithm with relaxation *Computing* **26** 189–95