MCMC using an Approximation

J. Andrés Christen *jac@cimat.mx*
CIMAT, Guanajuato, MEXICO.

Colin Fox *fox@math.auckland.ac.nz*
Department of Mathematics
The University of Auckland
New Zealand

Abstract

We present a method for generating samples from an unnormalized posterior distribution $f(\cdot)$ using MCMC in which the evaluation of $f(\cdot)$ is very difficult or computationally demanding. Commonly a less computationally demanding, perhaps local, approximation to $f(\cdot)$ is available, say $f^*_x(\cdot)$. An algorithm is proposed to generate an MCMC that uses such an approximation to calculate acceptance probabilities at each step, of a modified Metropolis–Hastings algorithm. Once a proposal is accepted using the approximation, $f(\cdot)$ is calculated with full precision ensuring convergence to the desired distribution. We give sufficient conditions for the algorithm to converge to $f(\cdot)$ and give both theoretical and practical justifications for its usage. Typical applications are in inverse problems using physical data models where computing time is dominated by complex model simulation. We outline Bayesian inference and computing for inverse problems. A stylized example is given of recovering resistor values in a network from electrical measurements made at the boundary. While this inverse problem has appeared in studies of underground reservoirs, it has primarily been chosen for pedagogical value since model simulation has precisely the same computational structure as a finite element method solution of the complete electrode model used in conductivity imaging, or 'electrical impedance tomography'. This example shows a dramatic decrease in CPU time, compared to a standard Metropolis–Hastings algorithm.

Keywords: MCMC, inverse problems, conductivity imaging, impedance tomography.

1 Introduction

We present a method for generating samples from an objective function $f(\cdot)$ (e.g. an unnormalized posterior distribution) using MCMC sampling, when evaluation of $f(\cdot)$ is highly computationally demanding. Examples of these occur in Bayesian image reconstruction or inverse problems where simulation of the measurements, and hence calculation of the likelihood, requires numerical implementation of a complex data model. Such inverse problems occur in imaging from strong wave scattering where simulation of the wave field requires solving a system of partial differential equations, and includes electrical impedance tomography (Fox and Nicholls 1997, Vauhkonen, et al. 1999, Kaipio, et al. 2000, Andersen, Brooks and Hansen 2003) ultrasound imaging (Greenleaf 1983, Fox and Nicholls 1998, Huttunen et al. 2004), inverse obstacle scattering (Kress and Rundell, 1998), and optical diffusion tomography (Arridge, 1999), amongst many other imaging modalities. In all these examples a far simpler (less computationally demanding), perhaps local, approximation to $f(\cdot)$ is available, that we denote $f^*_x(\cdot)$. In the imaging problems mentioned, such approximations are typically based on a local or global linearization of the forward map, or coarsening of the representation of unknowns. In this paper we are particularly interested in the local linearizations typically first derived for use in gradient-based optimization algorithms implementing regularized inversion (see references above), though our algorithm could work with any approximation. A local linear approximation to the forward map typically corresponds to a local Gaussian approximation to the likelihood and posterior distribution.

Fox and Nicholls (1997) proposed an algorithm to generate an MCMC using an approximate forward map, and hence likelihood, to calculate the acceptance probabilities at each step of a Metropolis–Hastings algorithm. Once a proposal is accepted, the likelihood is calculated with full precision, and cost. Examples were shown in the field of conductivity imaging for which calculation of the likelihood involves numerically solving a PDE with boundary conditions. The algorithm was tested with some examples exhibiting correct behavior in reasonable settings. In particular, for one example, they reported that 99.4% of the moves were rejected, implying that at only 0.6% of the proposed moves was the “exact” likelihood actually calculated. Calculating the approximate likelihood represented about 1% of the cost of calculating the likelihood with full precision. This means that Fox and Nicholls ’ (1997) algorithm represents approximately 1.6% of the cost of the corresponding standard Metropolis–Hastings MCMC. However, Fox and Nicholls (1997) did not prove the ergodic properties of their MCMC, and justified its adequacy in terms of particular examples. Indeed, because an approximation to the target distribution was used to calculate acceptance probabilities, it is not clear that the resulting chain necessarily has a stationary distribution.

In this paper we formalize Fox and Nicholls ’ (1997) algorithm, in a general setting and, with a modification, obtain the correct ergodic characteristics for the resulting MCMC. In the following Section we present our algorithm and in Section 3 we explain its limiting and performance characteristics. In Section 5 we outline the statistical and computational issues encountered in inverse problems, and in Section 5 an example is presented where we compare the standard Metropolis–Hastings algorithm with ours, where a local linear approximation to the forward map is used. A discussion of the paper is given in Section 6.

2 The algorithm

We construct a Metropolis–Hastings MCMC using the approximation $f^*_x(\cdot)$. The idea of the algorithm is the following. Consider a proposal distribution $q(y \mid x)$. To avoid calculating $f(y)$ for proposals that are rejected, we first “correct” the proposal with the approximation $f^*_x(y)$ to create a second proposal distribution $q^*(y \mid x)$, to be used in a standard Metropolis–Hastings algorithm. This second proposal has, in general, a high acceptance probability. In other words, the original proposal is tested using the cheap approximation to find moves that are likely to be accepted. We thereby sample from $f(\cdot)$, but avoid calculating $f(y)$ when proposals are rejected by $f^*_x(y)$, and hence gain in computational efficiency. See Section 3 and Theorem 1 for regularity conditions on $f_x^*(\cdot)$ to assure that Algorithm 1 creates a Markov chain with limiting distribution $f(\cdot)$.

Algorithm 1

1. *At $x^{(t)}$ generate a proposal y from $q(\cdot \mid x^{(t)})$.*
2. *Let*
$$g(x,y) = \min\left\{1, \frac{q(x \mid y)}{q(y \mid x)}\frac{f_x^*(y)}{f_x^*(x)}\right\}.$$
With probability $g(x^{(t)}, y)$, "promote" y to be used as a proposal for the standard Metropolis–Hastings algorithm. Otherwise use $y = x^{(t)}$ as a proposal. The actual proposal distribution used is
$$q^*(y \mid x) = g(x,y)q(y \mid x) + (1 - r(x))\delta_x(y)$$
where $r(x) = \int g(x,y)q(y \mid x)dy$ and $\delta_x(\cdot)$ denotes the Dirac mass at x (e.g. see Robert and Casella, 1999, p.235).
3. *Let*
$$\rho(x,y) = \min\left\{1, \frac{q^*(x \mid y)}{q^*(y \mid x)}\frac{f(y)}{f(x)}\right\}.$$
With probability $\rho(x^{(t)}, y)$ accept y setting $x^{(t+1)} = y$. Otherwise reject y setting $x^{(t+1)} = x^{(t)}$. This defines a transition kernel $K(\cdot,\cdot)$ from $x^{(t)}$ to $x^{(t+1)}$.

Note (as is also the case in the standard Metropolis–Hastings algorithm) that there is never a need to calculate $r(x)$ in Algorithm 1. When $y = x^{(t)}$ (i.e. the proposal was not promoted) $\rho(x,y) = 1$ and the chain remains at the same point. Conversely when $x \neq y$, $q^*(y \mid x) = g(x,y)q(y \mid x)$.

The reduction in computational work occurs because only when y is promoted (step 2) is $f(y)$ evaluated to calculate $\rho(x,y)$. For a special type of objective and approximation functions that typically occur in inverse problems, we prove in Section 3 that $\rho(x,y)$ is close to 1, depending on the quality of the approximation, and therefore we only update our approximation for proposals with a high probability of being accepted, thus avoiding unnecessary calculations of $f(\cdot)$. In Section 3 we investigate regularity conditions on $q(y \mid x)$ and $f_x^*(\cdot)$ to achieve convergence of the MCMC to $f(\cdot)$.

As explained above, Fox and Nicholls (1997) used a similar algorithm in which they simply accepted all promoted y's (in our terms, they had $\rho(x,y) \equiv 1$). Their algorithm showed good convergence properties for particular examples and was used without a guarantee of convergence. However, provided the approximation used is good so that $\rho(x,y) \approx 1$, Fox and Nicholls 's algorithm may be regarded as an approximation to ours. Algorithm 1 improves on Fox and Nicholls 's by having the correct limiting distribution with virtually identical computational cost.

Elements of Algorithm 1 may also be found in Liu's (2001) 'surrogate transition method' which also highlights the utility of approximations to $\log f(\cdot)$ when simulating a complex physical problem. However, it does not include a state-dependent approximation, which is an important aspect of Algorithm 1 since that case commonly occurs in efficient computational approximations to nonlinear problems. Interestingly, Liu suggests taking multiple steps with the approximate distribution before correcting with the exact distribution. It is not clear how such a procedure would perform in the presence of a state-dependent approximation. However forming a hierarchy of 'promoted' proposals using a sequence of increasingly better approximations may be a valuable generalization.

There are also interesting parallels between Algorithm 1 and the 'delayed rejection' algorithms of Tierney and Mira (1999) and Green and Mira (2001). In delayed rejection a second, perhaps modified, proposal is attempted following a rejection, with a modified acceptance probability that ensures detailed balance for the composite step. The primary aim of delayed rejection is to increase statistical efficiency, though some examples showed a rather small improvement in computational efficiency (Green and Mira 2001). Algorithm 1 similarly uses a composite step with a modified acceptance probability ensuring detailed balance, though it focusses on reducing computation per acceptance and requires two accept/reject steps to achieve an acceptance. In this spirit we may call Algorithm 1 'delayed acceptance'. It seems likely that Algorithm 1 and delayed rejection could be combined for particular examples to significantly increase both statistical and computational efficiency, over standard Metropolis–Hastings dynamics.

3 Limiting and performance characteristics

We prove the following theorem that, given a f-irreducible Metropolis–Hastings algorithm with proposal distribution $q(x \mid y)$, establishes regularity conditions on $f^*_x(y)$ in Algorithm 1 to achieve convergence properties in the resulting MCMC.

Theorem 1 *If the Metropolis–Hastings algorithm with q as a proposal (kernel $K_q(\cdot,\cdot)$) is f-irreducible, q is reversible and $q(y \mid x) > 0$ implies $f^*_x(y) > 0$, then f is an invariant distribution for K and K is f-irreducible. Moreover, if for any x, $K_q(x,x) > 0$ then $K(x,x) > 0$, and the resulting chain is strongly aperiodic.*

Proof: Let x with $f(x) > 0$, and A with $\int_A f(x)dx > 0$. We have that $K^n_q(x,A) > 0$, for some integer n. This implies that there exist $x^{(1)} = x, x^{(2)}, \ldots, x^{(n)} \in A$ with $K_q(x^{(t)}, x^{(t+1)}) > 0$. Without loss of generality we may assume that $x^{(t)} \neq x^{(t+1)}$, and thus we have $\rho_q(x^{(t)}, x^{(t+1)})q(x^{(t+1)} \mid x^{(t)}) > 0$ and $\rho_q(x^{(t+1)}, x^{(t)})q(x^{(t)} \mid x^{(t+1)}) > 0$, where $\rho_q(x,y) = \min\left\{1, \frac{f(y)}{f(x)}\frac{q(x|y)}{q(y|x)}\right\}$. From the conditions on $f^*_z(\cdot)$ we see that $\rho(x^{(t)}, x^{(t+1)})q^*(x^{(t+1)} \mid x^{(t)}) > 0$ for $t = 1, 2, \ldots, n-1$, which implies $K^n(x,A) > 0$. For the second part of the proof, note that $\rho_q(x,y) > 0$ implies $\rho(x,y) > 0$. Therefore if $\int \rho_q(x,y)q(y \mid x)dy < 1$, $r(x) < 1$ and $K(x,x) > 0$. Otherwise, if $\int \rho_q(x,y)q(y \mid x)dy = 1$, $q(x \mid x) > 0$ and $K(x,x) > q^*(x \mid x) = q(x \mid x) > 0$. It follows from reversibility that f is an invariant distribution for K. •

With the f-irreducibility and aperiodicity of the chain we may use standard ergodic results (see, for example, Robert and Casella 1999, p. 237) to prove that Algorithm 1 produces simulations from f.

For a particular case of objective functions typically encountered in inverse problems (see section 4), the following theorem states bounds for the acceptance probability $\rho(x,y)$.

Theorem 2 *Assume that $f(x) \propto e^{-h(x)}$ and that $f^*_x(y) \propto e^{-h^*_x(y)}$, $|h^*_x(y) - h(y)| < C|x-y|^P$, for $C > 0$ and $P \geq 0$, where $h(x) = h^*_x(x)$ is uniformly continuous. If $g(x,y) \leq e^{-2C|x-y|^P}$ or $g(y,x) \leq e^{-2C|x-y|^P}$ then $\rho(x,y) \geq e^{-C|x-y|^P}$.*

Proof: If $g(x,y) = 1$ and $g(y,x) < 1$, or $g(x,y) < 1$ and $g(y,x) = 1$, it is easy to see that $\rho(x,y) = \min\{1, r\}$ where $r = \frac{f(y)}{f^*_x(y)}$ or $\frac{f^*_y(x)}{f(x)}$, respectively. Given the Lipschitz condition on $h^*_x(y)$ we have

$$e^{-C|x-y|^P} \leq r \leq e^{C|x-y|^P}, \tag{1}$$

and therefore $\rho(x,y) \geq e^{-C|x-y|^P}$. Now, $g(x,y) < 1$ implies $g(x,y) = \frac{q(x|y)}{q(y|x)}\frac{f^*_x(y)}{f(x)}$. In order to have $g(y,x) = 1$ we need $1 < \frac{q(y|x)}{q(x|y)}\frac{f^*_y(x)}{f^*(y)} = g(x,y)^{-1}\frac{f^*_x(y)}{f^*(x)}\frac{f^*_y(x)}{f^*(y)}$ or $g(x,y) < \frac{f^*_x(y)}{f^*(x)}\frac{f^*_y(x)}{f^*(y)}$. To achieve this we only need $g(x,y) \leq e^{-2C|x-y|^P}$, and equivalently for the case when $g(x,y) = 1$ and $g(y,x) < 1$. •

We see that $\rho(x,y)$ is close to 1 when the approximation used is good. If both $g(x,y)$ and $g(y,x)$ are close (or equal) to 1, it may also be proved, depending on the smoothness of the objective function, that $\rho(x, y)$ is similar to the acceptance probability using the original proposal q. However, there will always be pathological cases using bad approximations for which Algorithm 1 will not be of any benefit.

Since both K and K_q have the same invariant distribution and are derived from the same proposal distribution, K is dominated by K_q off the diagonal since, as Peskun (1973) established, K_q is maximal amongst such kernels. (This also follows directly from $g(x, y) \leq 1$.) As these kernels are also reversible it follows (Peskun, 1973, Tierney, 1998) that the asymptotic variance of sample averages calculated using Algorithm 1 are greater than or equal to those calculated using the standard Metropolis–Hastings algorithm. Therefore, in general, Algorithm 1 will be less statistically efficient than the standard Metropolis–Hastings algorithm.

However, a good approximation will give an algorithm that is more computationally efficient, i.e., will achieve a smaller sample variance for given CPU time. Consider the ideal case where the computational cost of the approximation is negligible compared to the cost of the exact calculation occurring in step 3 of Algorithm 1, or in the standard Metropolis–Hastings algorithm. When the approximation is good, hence $\rho \approx 1$, the speedup of Algorithm 1 over the standard Metropolis–Hastings algorithm is the inverse of the acceptance rate. Thus Algorithm 1 is most useful when the rejection ratio is high. However, incorrect classification of proposals by the approximation leads to lower statistical efficiency, thereby reducing the speedup of variance reduction per CPU time. These simple considerations are sufficient to explain the speedup achieved in the computational example in Section 5.

4 Inverse Problems, Bayesian Inference, and Computation

Inverse problems occur when observed data d depend on unknowns x via a measurement process, and we want to recover x from d. In a mathematical setting, we represent the measurement process by a family of models parameterized by x, where all necessary parameters are contained in x, including 'nuisance parameters'. In the language of inverse problems, simulation of the model for given x defines the *forward map* $A : x \mapsto d$ giving data in the absence of errors.

The term 'inverse problem' is usually reserved for cases where the mapping from x to d is a complex physical relationship and where inversion of the forward map presents special difficulties. As mentioned above, examples of inverse problems include the various modalities of imaging from wave scattering used in non-invasive medical diagnostics, geophysical prospecting, and industrial process monitoring. The stylized example presented later, of imaging resistors in a network, comes from this class. Inverse problems also occur in a myriad of other settings such as inverse spectral problems (determining internal structure or shape from resonance frequencies), interferometric imaging, and mapping of flows subject to physical laws, to name just a few.

The classical, or deterministic, inverse problem is to invert the the function A to obtain unknowns x in terms of data d. Studies in classical inverse problems typically center on determining whether or not the the inverse problem is *well-posed* in the sense (of Hadamard) that: a solution x exists for any d, that solution is unique, and the inverse map $d \mapsto x$ is continuous. Practical inverse problems are usually *ill-posed* by failing the first or second requirement, while idealized inverse problems in which all possible measurements are made usually fail the last and hence are unstable, i.e., small changes in data d cause large or unbounded changes in recovered value(s) x. This latter property is routinely displayed by least-squares or maximum likelihood solutions to inverse problems, including in the limit of infinite number of data. For many inverse problems this behavior can be understood mathematically when the forward map is compact, implying that the inverse is discontinuous. Classical inversion consists of applying a regular approximation to the inverse.

The ubiquitous presence of measurement errors, or noise, means that a practical measurement process is probabilistic, and the inverse problem is naturally a problem in statistical inference. To fix ideas, consider additive noise n with density function $f_N(n)$. Then the likelihood for data d given

x is

$$l(d|x) = f_N\left(d - A(x)\right),$$

since the Jacobian determinant for the change of variables from n to d is 1. Most commonly the measurement error has an exponential family or Gibbs distribution (Kaipio and Somersalo, 2004).

In a Bayesian formulation, inference about x is based on the posterior density

$$f(x|d) \propto l(d|x)p(x)$$

where $p(x)$ denotes the prior density modelling beliefs about the unknown x independent of the data d. Exploratory analyses typically employ a low-level (e.g. pixel or voxel) representation with a Gibbs-MRF prior (Geman and Geman 1984). Classical inversion may be viewed as a special case since regularized inverses are the same as maximum a posteriori (MAP) estimates with regularization functionals that almost always correspond to a proper (or improper) Gibbs distribution prior written as the exponential of minus a norm (or semi-norm) of the unknown x. The most frequently used posterior density thus has the form

$$f(x|d) \propto \exp\left\{-\chi\left(d - A(x)\right) - \rho\left(x\right)\right\} \tag{2}$$

where χ and ρ are relatively simple functions. For example $\chi(y) = y^{\mathrm{T}}B^{-1}y/2$ when the noise comes from a Gaussian process with known covariance matrix B (and d is written as a vector), while ρ is an energy function computable as a sum of potentials defined over cliques of the pixel graph when x is modelled using a MRF. This is the form of objective functions for which we developed Theorem 2 with $h(x) = \chi\left(d - A(x)\right) + \rho\left(x\right)$ and the approximation to f resulting from an approximation to A. As in the field of image analysis, geometric information about the unknowns may be included using a prior based on an intermediate-level representation, such as Nicholls' (1998) continuum triangulation of the plane (see e.g. Anderson et al. 2003), or using a high-level representation of the type introduced by Grenander and Miller (1994). In all these analyses, the key computational features are that the state variable x comes from a very high dimensional space, and computing the posterior density is dominated by an expensive calculation of $A(x)$.

In principle, the posterior density in equation 2 can be evaluated and hence sampled via MCMC allowing summary statistics to be evaluated, effectively solving the inverse problem. Indeed a basic advantage of statistical (or optimization-based) solutions to inverse problems is that A^{-1} is not required, while the ability to apply MCMC is a major advantage of the statistical approach because of the range of image representations and prior distributions that may be used. However, the need to calculate the posterior density at each step in a standard Metropolis–Hastings algorithm, with typically many thousands or millions of steps required to give sufficiently small variance in estimates, appears to be computationally prohibitive for realistic inverse problems. However, there are a few demonstrations of comprehensive posterior sampling, conditioned on measured data, for inverse problems implementing a physically-based forward simulator. Recent examples include the work by McKeague, Nicholls and Speer (2004) mapping ocean circulation, Haario et al. (2004) recovering atmospheric gas density, and Cornford et al. (2004) who retrieve fields of wind vectors.

The massive scale of computation in each of these examples indicates that considerable improvement in efficiency of MCMC algorithms for inverse problems is required if the method is to be widely applied. Indeed, each of the works cited employs an enhanced MCMC to improve computational efficiency. For example Haario *et al.* (2004) used a novel adaptive Metropolis algorithm in which the covariance matrix in a d-dimensional Gaussian proposal distribution is calculated from the history of the output chain. The resulting chain is not Markov, but is provably ergodic with the desired equilibrium distribution. Another interesting development is the Metropolis coupled MCMC of Higdon, Lee and Holloman (2003) that simultaneously runs chains with the spatial parameters coarsened to various degrees. Information from the faster running, though approximate, coarse formulations speeds up mixing in the finest scale chain, from which samples are taken. These enhancements largely focus on improving proposal distributions and hence mixing.

We think of Algorithm 1 as reducing the computational cost per step by drawing on computational efficiencies developed for gradient ascent algorithms. In that field many sophisticated ideas have been developed, such as local linear or quadratic approximations, trust regions, search directions or subspaces (see e.g. Nocedal and Wright 1999), all in an attempt to reduce the computational burden of having to calculate a complex function at each of many iterations. In this aspect, numerical optimization shares many of the goals and problems of computational MCMC for inverse problems. We expect the use of local linear, or higher order, approximations to be most useful in Algorithm 1. We find appealing the feature that computer implementation of Algorithm 1 requires the same problem-specific functions required for a gradient-based optimization, so the effort in making the optimization efficient may be used to also increase efficiency of the MCMC. We expect that other techniques developed in computational optimization can be adapted to speed up sample-based inferential solutions to inverse problems.

5 An Electrical Network Test Problem

We consider recovering the positive resistance values in a square network of resistors, from noisy measurements made at the boundary of the network. We choose this example because the resulting computational problem allows us to concisely demonstrate Algorithm 1, while having the same structure as the discreticized equations for electrical impedance tomography (EIT). The correspondence is exact in the limit of fine discretization or large network. The 'complete electrode' forward model for EIT and its discretization, calibrated against data, is given in Kaipio *et al.* (2000), while references contained therein provide examples of attempted inversion from measured data.

5.1 Nodal Equations

Let $Z_N = \{(i,j) : 1 \leq i,j \leq N+1\}$ denote the $(N+1)\times(N+1)$ integer lattice. We refer to the elements of Z_N as the *nodes* of the network. Nodes (i,j) and (k,l) are adjacent when $(i-k)^2+(j-l)^2 = 1$, i.e., the usual first-order adjacency. Nodes (i,j) with $i,j = 1, N+1$ are on the boundary, while all others are in the interior. Let $n = (N+1)^2$ and $g : \{1,2,\ldots,n\} \to Z_N$ be any one-to-one function, i.e., an ordering of the nodes. We say that node $g(i)$ is indexed by i, and will use the index to refer to the node. The dual lattice to the lattice of nodes is $D_N = \{(l,m) : 1 \leq l,m \leq N+1 \text{ with } g(l) \text{ and } g(m) \text{ adjacent}\}$.

In a network of resistors, it is usual to think of resistors occupying the edges of the undirected graph $\{\{1,2,\ldots,n\}, D_N\}$ with the nodes (vertices) representing the electrical connection between resistors. A given set of resistors is denoted $\mathbf{r} = \left\{r_{(l,m)} : (l,m) \in D_N\right\}$ with the property $r_{(l,m)} = r_{(m,l)}$. Figure 1 shows the resistor network topology when there are $N = 4$ resistors per side.

We take the node with index n to be the electrical reference node, which in our examples we take (w.l.o.g.) to be the bottom right-most node, as shown in Figure 1. The vector of voltages $\mathbf{v} = (v_1, v_2, \ldots, v_{n-1})^{\mathrm{T}}$ at nodes (with respect to the reference node) is related to the vector of currents injected into nodes $\mathbf{i} = (i_1, i_2, \ldots, i_{n-1})^{\mathrm{T}}$ (and removed from the reference node) by a combination of Ohm's and Kirchoff's laws summarized by the nodal equations (e.g. Kuo 1962)

$$Y\mathbf{v} = \mathbf{i}. \tag{3}$$

Here Y is the $(n-1)\times(n-1)$ reduced admittance matrix

$$Y_{lm} = \begin{cases} -\sigma_{(l,m)} & l \neq m \\ \displaystyle\sum_{k=1}^{n} \sigma_{(l,k)} & l = m \end{cases} \tag{4}$$

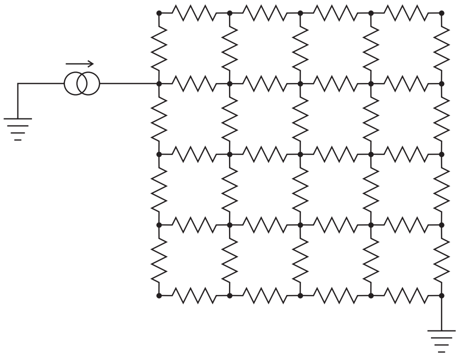

Figure 1: The resistor network for the case $N = 4$. Also shown are the reference node and current being injected at one node on the boundary.

for $l, m = 1, 2, \ldots, n-1$, in which

$$\sigma_{(l,m)} = \begin{cases} 1/r_{(l,m)} & (l,m) \in D_N \\ 0 & \text{otherwise} \end{cases}$$

is the conductance between nodes indexed by l and m.

For computational purposes it is noteworthy that Y is sparse, symmetric, and positive definite when all $r_{(l,m)} \in (0, \infty)$. We will exploit the feature that Y is a linear function of the conductances when making a cheap approximation to the forward map.

5.2 Forward Map

Electrical measurements are made on the network by injecting currents into nodes and measuring the resulting nodal voltages. To maintain the parallel with EIT as a non-invasive imaging technique, we restrict the nodes used for current injection or voltage measurement to a subset of nodes on the boundary – though in reservoir modelling applications measurements are often made at a set of internal nodes. Let $E \subset \{1, 2, \ldots, n-1\}$ be the set of boundary nodes used for injecting current or measuring voltages, excluding the reference node. We refer to these nodes as electrodes. At all other nodes $m \in \{1, 2, \ldots, n-1\} \setminus E$ current is conserved, i.e. $i_m = 0$, and the voltage v_m is unknown.

Since nodal voltages $\mathbf{v}$ are a linear function of applied currents $\mathbf{i}$, all possible measurements are made by applying the (usual) basis of current vectors at electrodes $\mathbf{i}^k = e_k$ for $k \in E$, and measuring the resulting voltage at all electrodes, v_l for $l \in E$. Hence noise-free measurements consist of the block of the inverse of the admittance matrix $\left\{ Y^{-1}_{l,k} : l, k \in E \right\}$, which we denote using the (Matlab-like) notation Y^{-1}_{EE}. The inverse problem may be stated as: given a block of the inverse of the reduced admittance matrix, is it possible to recover the full admittance matrix of the form in equation 4 and hence the resistor values?

Calculating the forward map $A : \mathbf{r} \mapsto Y^{-1}_{EE}$ requires solving the matrix equation in (3) for each current vector $\mathbf{i^k}$, a total of $|E|$ solutions, where $|E|$ denotes the number of electrodes. For typical networks containing thousands of resistors, and tens of electrodes, this step dominates computational cost of evaluating the posterior density, even using the fastest solving algorithms.

5.3 Computing the Forward Map Exactly and Approximately

An efficient exact forward map is computed by forming the reduced admittance matrix Y, computing the Cholesky factorization of Y, then using that factorization to solve the $|E|$ instances of matrix equation (3), one for each $\mathbf{i}^k = e_k$, to produce the submatrix Y_{EE}^{-1}. Computational cost is dominated by the work required to perform the factorization, which scales as $O(n^3)$ with respect to the number of nodes n (or resistors).

A cheap approximate calculation of the forward map is based on the first-order Taylor approximation to $A(\mathbf{r})$ with respect to conductivity. If $\mathbf{r}$ is the current state, consider a proposed state $\mathbf{r}' = \mathbf{r}$ except with the single resistor difference $r'_{(l,m)} = r_{(l,m)} + \Delta r_{(l,m)}$, which is the conductivity difference $\Delta\sigma_{(l,m)} = 1/r'_{(l,m)} - 1/r_{(l,m)}$. The first-order approximation centered on the current state $\mathbf{r}$ is

$$A^*_{\mathbf{r}}(\mathbf{r}') = A(\mathbf{r}) + \frac{\partial A}{\partial \sigma_{(l,m)}}(\mathbf{r})\Delta\sigma_{lm}.$$

For two-resistor changes, the rightmost term appears twice. Because of the linear dependence of Y on $\sigma_{(l,m)}$, the Jacobian term has the simple form $\frac{\partial A}{\partial \sigma_{lm}}(\mathbf{r}) = -U^T U$ where $U = Y_{El}^{-1} - Y_{Em}^{-1}$ with the associated term absent if $l = n$ or $m = n$. Hence all components of the Jacobian are available from the previous, exact, evaluation of $A(\mathbf{r})$ made within the most recent acceptance step. Calculation of the approximate forward map and likelihood requires $O(|E|^2)$ operations and does not depend on the number of resistors, so is $O(1)$ with respect to image size. This is a significant saving over the exact solution.

5.4 Likelihood

We consider the current vector is exactly known but that the voltage measurements are subject to additive errors. Noisy measurements d were simulated by adding independent noise $\sim N(0, s^2)$ to each component of Y_{EE}^{-1}. Hence the pdf for measuring voltages d given resistances $\mathbf{r}$, i.e. the likelihood, is

$$l(d \mid \mathbf{r}) \propto \exp\{-\chi(\mathbf{r})\} \qquad \text{where} \qquad \chi(\mathbf{r}) = \frac{\|d - A(\mathbf{r})\|_{\mathrm{F}}^2}{2s^2} \tag{5}$$

in which $\|\cdot\|_{\mathrm{F}}$ is the Frobenius norm that is simply the square root of the sum of squares of each element in the matrix, and d is written as a $|E| \times |E|$ matrix.

In our examples we take $N = 24$ resistors per side, hence 1200 resistors in all and $n = 625$ nodes. Resistances are either 2Ω or 3Ω and measurements are made on $|E| = 24$ electrodes with 6 electrodes evenly spaced on each side of the square network. The phantom 'true' image used is shown in figure 2 in which 2Ω resistors are shown as a black line while 3Ω are shown as grey, along with electrode positions.

With 24 electrodes we make $24 \times 24 = 576$ measurements, of which 300 are independent because of symmetry of Y, and hence Y^{-1}. Noise standard deviation was $s = 0.005$ giving a signal to noise ratio of about 1000.

Since evaluation of the likelihood in equation 5 requires computing the forward map, this is the expensive step that we will approximate in implementing Algorithm 1. An approximate likelihood is calculated by using $A^*_{\mathbf{r}}(\cdot)$ in place of $A(\cdot)$ when at state $\mathbf{r}$.

5.5 Representation and Prior

For the sake of simplicity, we condition inversion on knowledge of the network topology and that resistors take the value 2Ω or 3Ω. A representation in which resistors take one of a few values is appropriate in high contrast EIT or where a type field is being reconstructed, though in both these cases the representation may also allow for the resistor values to have some variability.

We specify a prior distribution for the probability that some trial set of resistances $\mathbf{r}$ coincides with the unknown true resistances as follows. A cell is a region within the network bounded by four resistors. We say that $(l,m) \in D_N$ and $(p,q) \in D_N$ are *neighbors* and write $(l,m) \sim (p,q)$ if resistor sites (l,m) and (p,q) are on the boundary of a common cell. Thus the four resistor sites bounding a

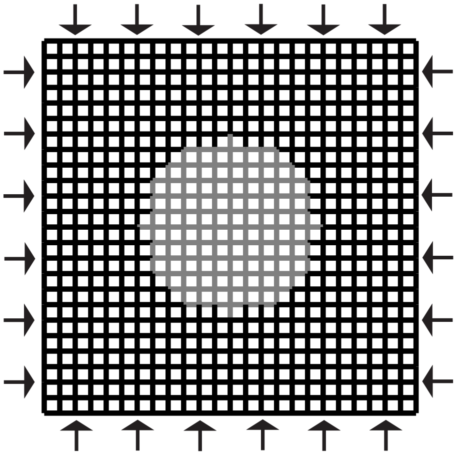

Figure 2: The true resistor network from which data was simulated. Black lines indicate a resistance of 2 Ohm while the grey lines denote 3 Ohm. Arrows indicate nodes used as electrodes.

cell form a clique. We model the spatial dependence of resistors in the grid with the Markov random field

$$p\left(\mathbf{r}\right) \propto \exp\left\{-\rho\left(\mathbf{r}\right)\right\} \qquad \text{where} \qquad \rho\left(\mathbf{r}\right) = -\theta \sum_{(l,m)\in D_N} \sum_{(p,q)\sim(l,m)} \delta_{r_{(l,m)},r_{(p,q)}} \tag{6}$$

in which $\delta_{a,b} = 1$ if $a = b$ and is otherwise zero, and θ is a lumping constant. In our simulations we took $\theta = 0.5$.

Note that each resistor site in the interior has six neighbors and therefore this MRF is different to the familiar Ising model. A depiction of this neighborhood system is given by Geman and Geman (1984), Fig. 1(f).

5.6 MCMC

We solve the inverse problem (of finding $\mathbf{r}$ given d) using Bayesian inference based on MCMC sampling of the posterior distribution. As an example of image recovery we report the marginal posterior mode (MPM). For a two-level image the MPM corresponds to taking the mean image and then setting each resistor to the closest allowable value.

Proposal of a candidate state is achieved using several ‘moves’, chosen to give ergodic behavior over useful time scales. At each step a move from the following list is picked, with relative probability 1:2:4, respectively:

1. Pick a resistor at random and set it to a possible value at random
2. Pick two resistors at random and swap them
3. Pick a resistor at random, then a resistor at each end, and swap the end resistors

Move 1 is sufficient to give irreducibility while moves 2 and 3 are designed to give improved mixing. These moves are naive and in the computed example 88% of proposals correspond to no change. However those proposals take negligible CPU time to identify and reject, and do not affect our efficiency results.

We implemented the standard Metropolis–Hastings algorithm and Algorithm 1 in MatLab, each using the proposal distribution defined above. The standard algorithm uses a standard Metropolis–Hastings acceptance step with the objective function calculated exactly. Algorithm 1 uses the approximate calculation given above, in step 2, and the exact calculation of the forward map, and objective function, in step 3. The exact prior was used for both implementations, as computing the prior incurs negligible cost. Since the approximate likelihood contains the approximate forward map in the exponential, and the exact prior is used, the approximate objective function is always positive and the conditions of Theorem 1 are satisfied. Theorem 1 guarantees that the chain produced by Algorithm 1, like that from the standard algorithm, converges to the desired posterior distribution.

5.7 Computational results

Each chain was initialized from a random pattern of allowable resistor values, and an estimate of the MPM at each resistor was computed from the resulting output. The resulting images are shown in figures 3 (Algorithm 1) and 4 (standard Metropolis–Hastings). We recorded a number of statistics along each run, including the log-likelihood which in this case is also the sum-of-squares of the residuals. Convergence is reliable in both cases, as can be judged from the output statistics plotted in figures 3 and 4. The integrated autocorrelation time of the log-likelihood was computed. This quantity is small in efficient MCMC since it is, roughly speaking, the number of correlated MCMC samples from the posterior distribution with the variance-reducing power of one independent sample. In these figures one MCMC update equals 2000 steps of the chain, i.e. 2000 proposals.

From the plots of autocorrelation it can be seen that the standard Metropolis–Hastings algorithm generates an independent sample each 32 MCMC updates while Algorithm 1 is slightly less efficient with an independent sample per 42 MCMC updates, i.e., the integrated autocorrelation times where 32 and 42, respectively. However Algorithm 1 was significantly faster and took 15.5 seconds of CPU time to produce an independent sample whereas the standard algorithm took 379 seconds per independent sample. Hence the use of the approximation reduces CPU time by close to a factor of 25 for this problem.

In the example, 2.9% of proposals (with resistance changes) are accepted in the standard Metropolis–Hastings algorithm so we would expect a speedup by a factor of 35 when the approximation has negligible cost and is accurate. The primary misclassification introduced by the approximation was that 30% of proposals were falsely rejected, i.e. were rejected at step 2 but would have been accepted by the exact calculation. Hence each of efficiency and speedup is reduced by that factor.

6 Discussion

We developed a simple algorithm to improve a Metropolis–Hastings MCMC when an approximation to the objective function is available. As mentioned earlier, the resulting chain is less efficient than the standard Metropolis–Hastings, however a great gain in CPU time may be obtained when the approximation uses negligible CPU time in comparison to the exact calculation of the objective function. The example provided, in the field of conductivity imaging, showed a decrease in CPU time by a factor of 25.

Our target application for Algorithm 1 is sample-based solutions to inverse problems where evaluation of the likelihood, requiring evaluation of the forward map, dominates computational cost. Here a local linear approximation to the forward map may be used to form a cheap approximation, as in the computed example. There may also be cases where it is advantageous to approximate the prior. In cases where the gradient is used within a proposal, such as in a Langevin proposal step, calculation of the *proposal* may also be computationally expensive. Then a straightforward generalization of Algorithm 1 can be applied in which an approximation to the reverse proposal $q(y|x)$ in step 2 is also used, perhaps based on a quadratic approximation to the forward map.

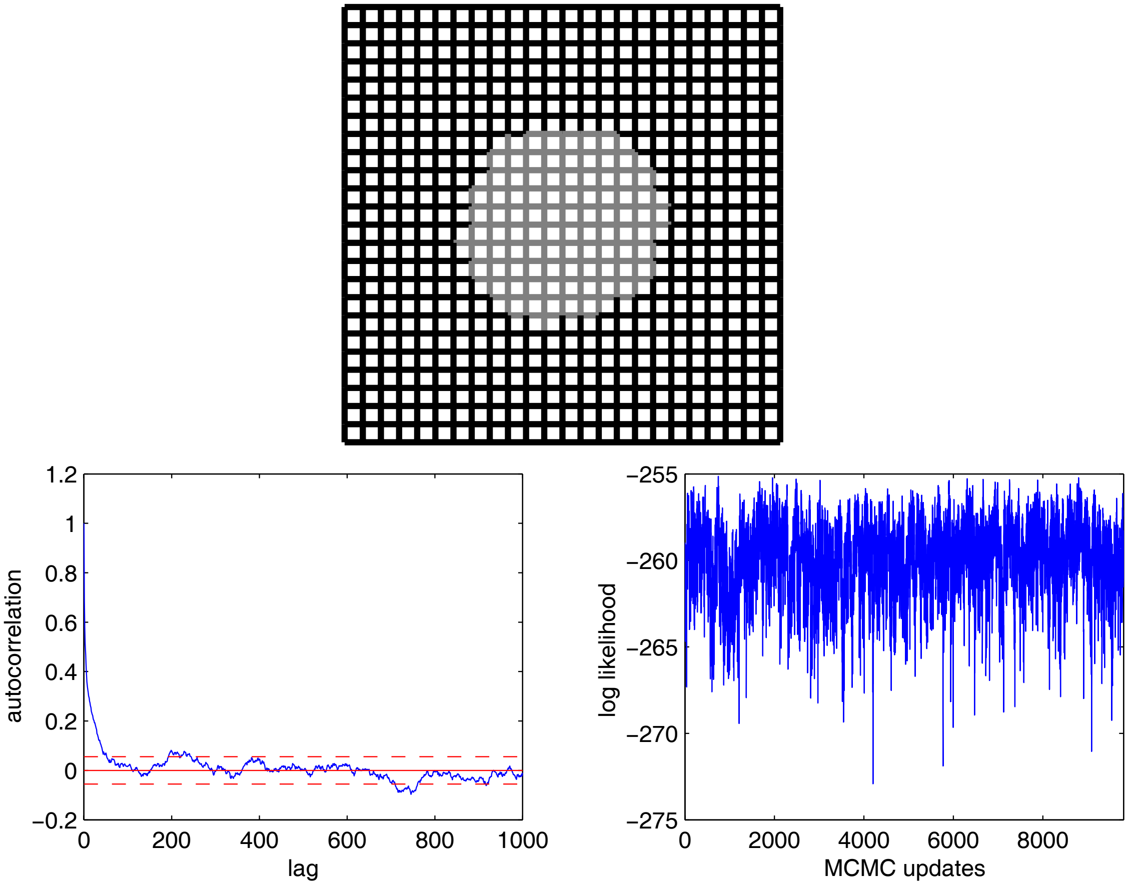

Figure 3: Marginal posterior mode (upper), the sample log-likelihood (lower right) up to a constant independent of r, plotted against the update number, along with the autocorrelation function (ACF), (lower left) of the MCMC output series, all for Algorithm 1. Dashed lines indicate variance of ACF asymptotic in the lag.

7 Acknowledgments

This work was undertaken while CF was a visitor at CIMAT. CF is grateful to CIMAT for providing financial support and a productive research environment.

References

[1] Andersen, K.E., Brooks, S.P. and Hansen, M.B. (2003), "Bayesian Inversion of Geoelectrical Resistivity Data", *Journal of the Royal Statistical Society, Series B*, **65**. 619–642.

[2] S. R. Arridge, (1999). Optical tomography in medical imaging, *Inverse Problems* **15**, R41-R93.

[3] Cornford, D., Csató, L., Evans, D. J. and Opper, M. (2004) "Bayesian analysis of the scatterometer wind retrieval inverse problems: some new approaches", *J. R. Statist. Soc. B* **66**(3), 609–626.

[4] Fox, C. and Nicholls, G. (1997). "Sampling Conductivity Images via MCMC", In: *The Art and Science of Bayesian Image Analysis*, K.V. Mardia, C.A. Gill, R.G. Aykroyd eds, Proceedings of

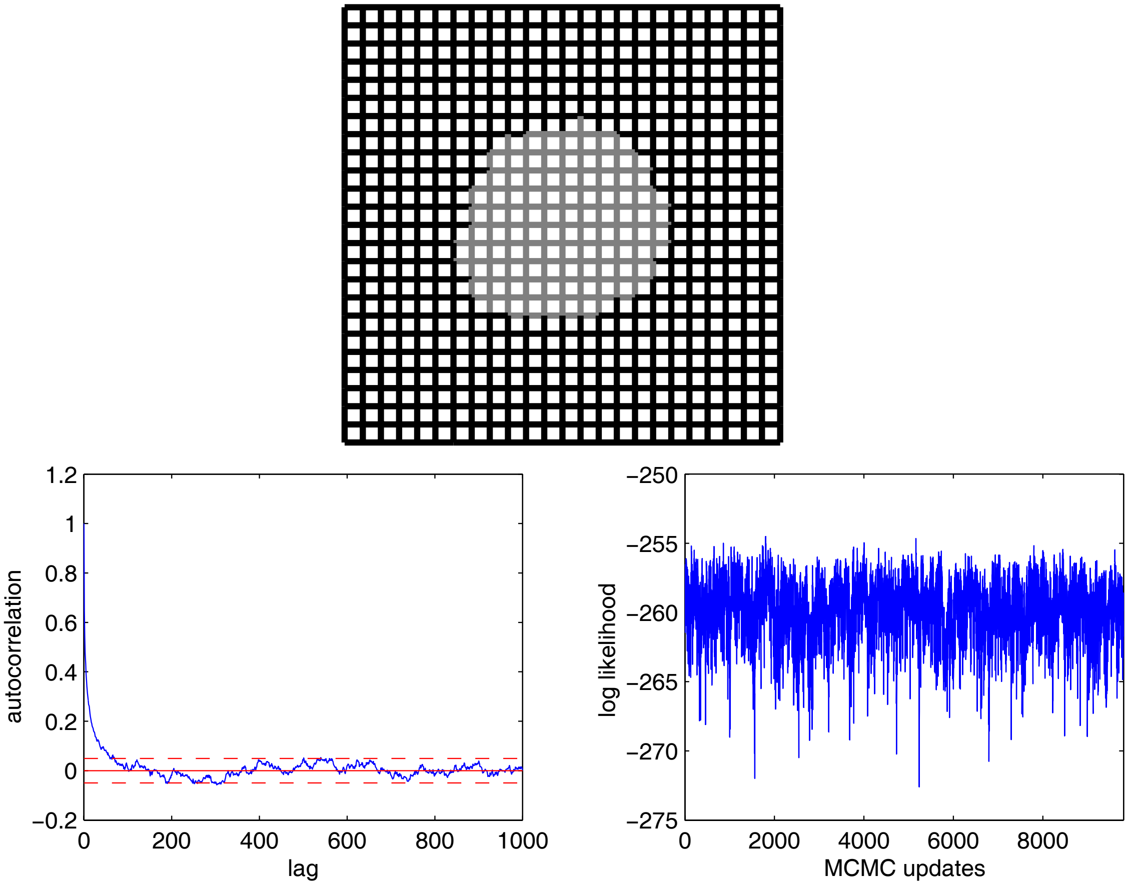

Figure 4: Equivalent output to figure 3 for the standard Metropolis–Hastings algorithm.

the Leeds Annual Statistical Research Workshop (LASR), 91-100, July 1997, Leeds University Press.

[5] Fox, C. and Nicholls, G. K. (1998) “Physically-based likelihood for ultrasound imaging”, In: *Bayesian Inference for Inverse Problems*, Proc. SPIE 3459, 92-99.

[6] Geman, S and Geman, D. (1984) “Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images”, *I.E.E.E. Trans. Pattern Anal. Machine Intell.*, **6**, 721–741.

[7] Green, P.J. and Mira, A. (2001). “Delayed Rejection in Reversible Jump Metropolis– Hastings”, *Biometrika*, **88**, 1035-1053.

[8] Greenleaf, J. F. (1983), “Computerized Tomography with Ultrasound”, *Proc. IEEE*, **71**(3), 330–337.

[9] Grenander, U. and Miller, M. (1994). “Representations of knowledge in complex systems”, *J. Roy. Statist. Soc. Ser. B* **56**(4) 549–603.

[10] Haario, H., Laine, M., Lehtinen, M., Saksman, E. and Tamminen, J.(2004) “Markov chain Monte Carlo methods for high dimensional inversion in remote sensing”, *J. R. Statist. Soc. B* **66**(3), 591-608.

[11] Higdon, D., Lee, H., and Holloman, C. (2003). “Markov chain Monte Carlo-based approaches for inference in computationally intensive inverse problems”, In: *Bayesian Statistics 7*, J. M. Bernardo, M. J. Bayarri, J. O. Berger, A. P. Dawid, D. Heckerman, A. F. M. Smith and M. West (Eds.), Oxford University Press.

[12] Huttunen, T., Malinen, M., Kaipio, J. P., White, P. J., and Hynynen, K. (2004), “A Full-Wave Helmholtz Model for Continuous-Wave Ultrasound Transmission”, *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, in press.

[13] Kaipio, J. P., Kolehmainen, V., Somersalo, E. and Vauhkonen, M. (2000), “Statistical inversion and Monte Carlo sampling methods in electrical impedance tomography”, *Inverse Problems*, **16**, 1487–1522.

[14] Kaipio, J. P., and Somersalo, E. (2004), *Statistical and Computational Inverse Problems*, Applied Mathematical Sciences 160, Springer-Verlag, ISBN: 0-387-22073-9, (in press).

[15] Kolehmainen, V., Vauhkonen, M., Kaipio, J. P. and Arridge, S. R. (2000), “Recovery of piecewise constant coefficients in optical diffusion tomography”, *Optics Express*, **7**, 468–480.

[16] Kolehmainen, V., Nicholls, G., and Fox, C. (2004), “MCMC inversion of Measured EIT Data”, *in preparation.*

[17] Kress, R., and Rundell, W. (1998) “Inverse Obstacle Scattering Using Reduced Data”, *SIAM Journal on Applied Mathematics*, **59**(2), 442–454.

[18] Kuo, F.F. (1962), *Network Analysis and Synthesis* John Wiley and Sons, Inc., New York.

[19] Liu, J.S. (2001). *Monte Carlo Strategies in Scientific Computing*, Springer-Verlag, New York.

[20] McKeague, I., Nicholls, G. K., and Speer, K. (2005). “Statistical Inversion of South Atlantic Circulation in an Abyssal Neutral Density Layer”, *Journal of Marine Research*, in press.

[21] Nicholls, G. K. (1998), “Bayesian image analysis with Markov chain Monte Carlo and coloured continuum triangulation mosaics”, *J. Roy. Statist. Soc. Ser. B* **60**, 643–659.

[22] Nocedal, J., and Wright, S. J. (1999). *Numerical Optimization.* Springer–Verlag, New York.

[23] Peskun, P.H. (1973). “Optimum Monte-Carlo sampling using Markov chains”, *Biometrika*, **60**(3), 607–612.

[24] Robert, C.P. and Casella, G. (1999). *Monte Carlo Statistical Methods*, Springer Texts in Statistics, Springer-Verlag, New York.

[25] Tierney, L. (1998). “A note on Metropolis–Hastings kernels for general state spaces”, *Annals of Applied Probability*, **8**(1), 1–9.

[26] Tierney, L. and Mira, A. (1999). “Some adaptive Monte Carlo methods for Bayesian inference”, *Statistics in Medicine*, **18**, 2507–2515.

[27] Vauhkonen, P., Vauhkonen, M., Savolainen, T. and Kaipio J. (1999). “Three-dimensional electrical impedance tomography based on the complete electrode model”, *IEEE Trans. Biomed. Eng.*, **46**, 1150-1160.