Randomize-Then-Optimize: A Method for Sampling from Posterior Distributions in Nonlinear Inverse Problems

SIAM J. SCI. COMPUT.
Vol. 36, No. 4, pp. A1895–A1910

RANDOMIZE-THEN-OPTIMIZE: A METHOD FOR SAMPLING FROM POSTERIOR DISTRIBUTIONS IN NONLINEAR INVERSE PROBLEMS*

JOHNATHAN M. BARDSLEY†, ANTTI SOLONEN‡, HEIKKI HAARIO§, AND MARKO LAINE¶

Abstract. High-dimensional inverse problems present a challenge for Markov chain Monte Carlo (MCMC)-type sampling schemes. Typically, they rely on finding an efficient proposal distribution, which can be difficult for large-scale problems, even with adaptive approaches. Moreover, the autocorrelations of the samples typically increase with dimension, which leads to the need for long sample chains. We present an alternative method for sampling from posterior distributions in nonlinear inverse problems, when the measurement error and prior are both Gaussian. The approach computes a candidate sample by solving a stochastic optimization problem. In the linear case, these samples are directly from the posterior density, but this is not so in the nonlinear case. We derive the form of the sample density in the nonlinear case, and then show how to use it within both a Metropolis–Hastings and importance sampling framework to obtain samples from the posterior distribution of the parameters. We demonstrate, with various small- and medium-scale problems, that randomize-then-optimize can be efficient compared to standard adaptive MCMC algorithms.

Key words. nonlinear inverse problems, Bayesian methods, uncertainty quantification, computational statistics, sampling methods

AMS subject classifications. 15A29, 65C05, 65C60

DOI. 10.1137/140964023

1. Introduction.

We consider Gaussian statistical models of the form

$$\mathbf{y} = \mathbf{f}(\boldsymbol{\theta}) + \boldsymbol{\epsilon}, \tag{1.1}$$

where $\mathbf{y} \in \mathbb{R}^m$ is the response vector, $\boldsymbol{\theta} \in \mathbb{R}^n$ is the vector of unknown parameters, $\mathbf{f} : \mathbb{R}^n \to \mathbb{R}^m$, with $m \geq n$, is the nonlinear parameter-to-response vector map, and $\boldsymbol{\epsilon}$ is an m-dimensional Gaussian random vector with mean $\mathbf{0}$ and $m \times m$ covariance matrix $\boldsymbol{\Sigma}$, i.e., $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma})$.

Assuming $\boldsymbol{\Sigma}$ is known, which we do in this paper, we can left multiply both sides of (1.1) by $\boldsymbol{\Sigma}^{-1/2}$. The noise vector of the resulting transformed model is Gaussian with mean $\mathbf{0}$ and covariance $\mathbf{I}$. This allows us to assume, without loss of generality, that $\boldsymbol{\Sigma} = \mathbf{I}$, leading to the likelihood function

$$p(\mathbf{y}|\boldsymbol{\theta}) \propto \exp\left(-\frac{1}{2}\|\mathbf{f}(\boldsymbol{\theta}) - \mathbf{y}\|^2\right).$$

*Submitted to the journal's Methods and Algorithms for Scientific Computing section April 8, 2014; accepted for publication (in revised form) June 9, 2014; published electronically August 14, 2014.

http://www.siam.org/journals/sisc/36-4/96402.html

†Department of Mathematical Sciences, University of Montana, Missoula, MT 59812-0864 (bardsleyj@mso.umt.edu).

‡Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, MA 02139, Department of Mathematics and Physics, Lappeenranta University of Technology, Lappeenranta, Finland, and Finnish Meteorological Institute, Helsinki, Finland (antti.solonen@gmail.com).

§Department of Mathematics and Physics, Lappeenranta University of Technology, Lappeenranta, Finland (heikki.haario@lut.fi).

¶Finnish Meteorological Institute, Helsinki, Finland (marko.laine@fmi.fi).

It is common in practice for the noise covariance $\mathbf{\Sigma}$ to be known. For example, the variance is often estimated from repeated measurements or residual analysis [5], or it can be estimated using a hierarchical model [1, 6].

In a number of cases, the Bayesian posterior density function is of least squares form as well. This is true, for example, when the prior $p(\boldsymbol{\theta})$ is constant, since then, by Bayes' Law,

$$p(\boldsymbol{\theta}|\mathbf{y}) \propto p(\mathbf{y}|\boldsymbol{\theta})p(\boldsymbol{\theta}) \propto p(\mathbf{y}|\boldsymbol{\theta}).$$

Such posteriors appear in a variety of small-scale parameter estimation problems, a few of which we will consider in our numerical experiments.

Alternatively, in classical inverse problems (see, e.g., [11, 16]), it is common to assume a Gaussian noise model of the form (1.1) and a Gaussian prior of the form $p(\boldsymbol{\theta}) \propto \exp(-\frac{1}{2}(\boldsymbol{\theta}-\boldsymbol{\theta}_0)^T\mathbf{L}(\boldsymbol{\theta}-\boldsymbol{\theta}_0))$. Then the posterior will have the form

$$\begin{aligned} p(\boldsymbol{\theta}|\mathbf{y}) &\propto p(\mathbf{y}|\boldsymbol{\theta})p(\boldsymbol{\theta}) \\ &\propto \exp\left(-\frac{1}{2}\left(\|\mathbf{f}(\boldsymbol{\theta})-\mathbf{y}\|^2 + \|\mathbf{L}^{1/2}(\boldsymbol{\theta}-\boldsymbol{\theta}_0)\|^2\right)\right) \\ &= \exp\left(-\frac{1}{2}\|\tilde{\mathbf{f}}(\boldsymbol{\theta})-\tilde{\mathbf{y}}\|^2\right), \end{aligned} \tag{1.2}$$

where

$$\tilde{\mathbf{f}}(\boldsymbol{\theta}) = \begin{bmatrix} \mathbf{f}(\boldsymbol{\theta}) \\ \mathbf{L}^{1/2}\boldsymbol{\theta} \end{bmatrix} \quad \text{and} \quad \tilde{\mathbf{y}} = \begin{bmatrix} \mathbf{y} \\ \mathbf{L}^{1/2}\boldsymbol{\theta}_0 \end{bmatrix}. \tag{1.3}$$

Thus we see that in this case, the posterior has least squares form as well. To simplify notation later, we drop the tilde notation in (1.2), so that in all of the cases of interest to us, the posterior can be expressed

$$p(\boldsymbol{\theta}|\mathbf{y}) \propto \exp\left(-\frac{1}{2}\|\mathbf{f}(\boldsymbol{\theta})-\mathbf{y}\|^2\right). \tag{1.4}$$

Our focus in this paper is on the problem of sampling from posterior density functions of the form (1.4). Because $\mathbf{f}(\boldsymbol{\theta})$ is nonlinear, we cannot, in general, sample directly from $p(\boldsymbol{\theta}|\mathbf{y})$ in (1.4). A commonly used class of algorithms for doing this are the Markov chain Monte Carlo (MCMC) methods; see, e.g., [4, 6] for a general introduction. For nonlinear inverse problems, a variety of MCMC methods have been developed. For example, in [7, 8], adaptive MCMC algorithms are introduced that work well on small-to-medium scale parameter estimation problems, while in [10, 11, 13, 17], the Gibbs sampler is implemented for use on large-scale, nonlinear inverse problems. Another class of MCMC methods used for large-scale inverse problems—which construct Gaussian proposals using the Hessian of the likelihood function—are presented in [12, 14].

The approach we take in this paper, which we call randomize-then-optimize (RTO), is based on obtaining candidate samples by repeatedly optimizing a randomly perturbed cost function. Our approach is somewhat similar to the randomized maximum likelihood method presented in [2, section 2.1], where a randomized likelihood function is repeatedly maximized to obtain approximative samples from the posterior. However, as we will see in the following sections, we modify the optimization problem used within RTO, so that we can derive the probability density function for the RTO

samples. This allows us to use RTO samples within either a Metropolis–Hastings or importance sampling framework to obtain theoretically correct posterior sampling methods.

The paper is organized as follows. First, in section 2, we present the basic idea and motivation behind RTO. In section 3, we derive the form of the RTO probability density function. Then in section 4, we present the RTO Metropolis–Hastings (RTO-MH) method for sampling from $p(\boldsymbol{\theta}|\mathbf{y})$, and also show how RTO can be used within an importance sampling framework to sample from $p(\boldsymbol{\theta}|\mathbf{y})$. We then use RTO-MH for sampling from the posterior in several examples in section 5, comparing the results with those obtained using the state of the art adaptive MCMC method of [7]. In section 6, we collect some specific remarks about the RTO method and discuss some experiences gained by the numerical examples. We end with conclusions in section 7.

2. The RTO method. RTO can be motivated from the linear case, in which $\mathbf{f}(\boldsymbol{\theta}) = \mathbf{J}\boldsymbol{\theta}$, where $\mathbf{J} \in \mathbb{R}^{m\times n}$ is assumed to have full column rank. Then

$$p(\boldsymbol{\theta}|\mathbf{y}) = (2\pi)^{-m/2}|\mathbf{J}^T\mathbf{J}|^{1/2}\exp\left(-\frac{1}{2}\|\mathbf{J}\boldsymbol{\theta}-\mathbf{y}\|^2\right),$$

where $|\cdot|$ denotes determinant, is a Gaussian probability density with mean $(\mathbf{J}^T\mathbf{J})^{-1}\mathbf{J}^T\mathbf{y}$ and covariance matrix $(\mathbf{J}^T\mathbf{J})^{-1}$. Samples from $p(\boldsymbol{\theta}|\mathbf{y})$ can be computed by solving the stochastic optimization problem

$$\boldsymbol{\theta} = \arg\min_{\boldsymbol{\psi}} \|\mathbf{J}\boldsymbol{\psi} - (\mathbf{y}+\boldsymbol{\epsilon})\|^2, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0},\mathbf{I}). \tag{2.1}$$

This is the approach taken in [1], where it is shown to be an efficient way to sample from high-dimensional Gaussian posteriors arising in linear inverse problems.

To motivate our later discussion, we compute the thin $\mathbf{QR}$-factorization of $\mathbf{J}$ in (2.1), which we denote by $\mathbf{J} = \bar{\mathbf{Q}}\bar{\mathbf{R}}$ with $\bar{\mathbf{Q}} \in \mathbb{R}^{m\times n}$ and $\bar{\mathbf{R}} \in \mathbb{R}^{n\times n}$. Then the solution of (2.1) can be equivalently expressed as the solution of the stochastic linear system

$$\bar{\mathbf{Q}}^T\mathbf{J}\boldsymbol{\theta} = \bar{\mathbf{Q}}^T(\mathbf{y}+\boldsymbol{\epsilon}), \quad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0},\mathbf{I}). \tag{2.2}$$

Note that $\bar{\mathbf{Q}}^T\mathbf{J} = \bar{\mathbf{R}}$ is upper triangular, and hence (2.2) can be solved using backward substitution.

Our goal in this work is to extend the idea of (2.1)–(2.2) to the nonlinear case. If we simply replace $\mathbf{J}\boldsymbol{\psi}$ by $\mathbf{f}(\boldsymbol{\psi})$ in (2.1), we obtain the *randomized maximum likelihood* method [2, section 2.1]

$$\boldsymbol{\theta} = \arg\min_{\boldsymbol{\psi}} \|\mathbf{f}(\boldsymbol{\psi}) - (\mathbf{y}+\boldsymbol{\epsilon})\|^2, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0},\mathbf{I}). \tag{2.3}$$

Samples from (2.3) are not samples from $p(\boldsymbol{\theta}|\mathbf{y})$, though we have found them to be quite accurate approximations in many cases. This motivates the desire to use (2.3) as a proposal mechanism within a Metropolis–Hastings framework. However, the form of the probability density for the corresponding random vector $\boldsymbol{\theta}$ is unknown.

In this paper, we present an alternative, but closely related, approach, for which we can derive the probability density. First, let $\bar{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} p(\boldsymbol{\theta}|\mathbf{y})$ be the MAP estimator, and define $\mathbf{J}(\bar{\boldsymbol{\theta}}) = \bar{\mathbf{Q}}\bar{\mathbf{R}}$ to be the thin $\mathbf{QR}$-factorization of $\mathbf{J}(\bar{\boldsymbol{\theta}})$. Then in (2.2) we use this $\bar{\mathbf{Q}}$ and replace $\mathbf{J}\boldsymbol{\theta}$ by $\mathbf{f}(\boldsymbol{\theta})$, yielding the nonlinear stochastic equation

$$\bar{\mathbf{Q}}^T\mathbf{f}(\boldsymbol{\theta}) = \bar{\mathbf{Q}}^T(\mathbf{y}+\boldsymbol{\epsilon}), \quad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0},\mathbf{I}). \tag{2.4}$$

In the next section, we derive the probability density for the random variable $\boldsymbol{\theta}$ defined by (2.4). In practice, to compute samples using (2.4), we solve the stochastic optimization problem

$$\boldsymbol{\theta} = \arg\min_{\boldsymbol{\psi}} \|\bar{\mathbf{Q}}^T(\mathbf{f}(\boldsymbol{\psi}) - (\mathbf{y} + \boldsymbol{\epsilon}))\|^2, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}). \tag{2.5}$$

Hence, we call the approach RTO. Note that while (2.1)–(2.2) defines $\boldsymbol{\theta}$ as a linear transformation of the Gaussian random variable $\boldsymbol{\epsilon}$, (2.4)–(2.5) defines $\boldsymbol{\theta}$ as a nonlinear transformation of $\boldsymbol{\epsilon}$. We will show that under reasonable assumptions the probability density function for the random vector $\boldsymbol{\theta}$ defined by (2.4)–(2.5) has the form

$$p_{\bar{\boldsymbol{\theta}}}(\boldsymbol{\theta}) \propto c(\boldsymbol{\theta}) p(\boldsymbol{\theta}|\mathbf{y}), \tag{2.6}$$

where $c(\boldsymbol{\theta})$ is a scaling term derived below, and $p(\boldsymbol{\theta}|\mathbf{y})$ is the posterior density.

We make use of (2.6) as a proposal within a Metropolis–Hastings framework, and show that the results agree with posterior samples obtained via the adaptive MCMC method of [7]. We also show how (2.6) can be used within an importance sampling framework. The high-dimensional posterior densities arising in nonlinear statistical inverse problems, e.g., in geophysical and medical imaging, are difficult to sample from using a Metropolis–Hastings MCMC. In such cases, our approach, which requires repeated solution of an optimization problem, may be a viable alternative. To implement the method, an efficient optimization algorithm for solving (2.5) is required, and each RTO sample is obtained from one application of the optimizer.

3. The RTO probability density function. First, we assume that the MAP estimation problem,

$$\bar{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\psi}} \frac{1}{2}\|\mathbf{f}(\boldsymbol{\psi}) - \mathbf{y}\|^2,$$

has a unique solution $\bar{\boldsymbol{\theta}}$, and that $\mathbf{f}$ is continuously differentiable with Jacobian $\mathbf{J}(\boldsymbol{\theta})$ that is rank n for all $\boldsymbol{\theta}$ in the domain of $\mathbf{f}$. Then the first order necessary conditions for optimality are given by

$$\mathbf{J}(\bar{\boldsymbol{\theta}})^T(\mathbf{y} - \mathbf{f}(\bar{\boldsymbol{\theta}})) = \mathbf{0}. \tag{3.1}$$

Next, we compute the $\mathbf{QR}$-factorization of $\mathbf{J}(\bar{\boldsymbol{\theta}})$: $\mathbf{J}(\bar{\boldsymbol{\theta}}) = [\bar{\mathbf{Q}}, \tilde{\mathbf{Q}}] \left[\begin{smallmatrix} \bar{\mathbf{R}} \\ \mathbf{0} \end{smallmatrix}\right]$, where the columns of $\bar{\mathbf{Q}} \in \mathbb{R}^{m\times n}$ and $\tilde{\mathbf{Q}} \in \mathbb{R}^{m\times(m-n)}$ are orthonormal bases for the column space of $\mathbf{J}(\bar{\boldsymbol{\theta}})$ and its orthogonal complement, respectively; $\bar{\mathbf{R}} \in \mathbb{R}^{n\times n}$ is upper triangular and is invertible since $\mathbf{J}(\bar{\boldsymbol{\theta}})$ has rank n; and $\mathbf{0} \in \mathbb{R}^{(m-n)\times n}$ is the zero matrix. Note that the thin $\mathbf{QR}$-factorization of $\mathbf{J}(\bar{\boldsymbol{\theta}})$ is then given by $\mathbf{J}(\bar{\boldsymbol{\theta}}) = \bar{\mathbf{Q}}\bar{\mathbf{R}}$, and that (3.1) implies $\bar{\mathbf{Q}}^T(\mathbf{y} - \mathbf{f}(\bar{\boldsymbol{\theta}})) = \mathbf{0}$, which can be equivalently written

$$\mathbf{F}_{\bar{\boldsymbol{\theta}}}(\bar{\boldsymbol{\theta}}) = \bar{\mathbf{Q}}^T\mathbf{y}, \quad \text{where} \quad \mathbf{F}_{\bar{\boldsymbol{\theta}}}(\boldsymbol{\theta}) \stackrel{def}{=} \bar{\mathbf{Q}}^T\mathbf{f}(\boldsymbol{\theta}).$$

Now, because $\bar{\boldsymbol{\theta}}$ is the unique solution of (3.1), $\mathbf{F}_{\bar{\boldsymbol{\theta}}}$ is an invertible function at $\bar{\mathbf{Q}}^T\mathbf{y}$, and $\bar{\boldsymbol{\theta}} = \mathbf{F}_{\bar{\boldsymbol{\theta}}}^{-1}(\bar{\mathbf{Q}}^T\mathbf{y})$. Moreover, since $\mathbf{f}(\boldsymbol{\theta})$ is continuously differentiable with respect to $\boldsymbol{\theta}$, by the inverse function theorem, $\mathbf{F}_{\boldsymbol{\theta}}$ is invertible in a neighborhood of $\bar{\mathbf{Q}}^T\mathbf{y}$. This motivates using the inverse mapping $\mathbf{F}_{\bar{\boldsymbol{\theta}}}^{-1}$ to define a random vector $\boldsymbol{\theta}$ via

$$\boldsymbol{\theta} = \mathbf{F}_{\bar{\boldsymbol{\theta}}}^{-1}(\mathbf{v}), \quad \text{where} \quad \mathbf{v} = \bar{\mathbf{Q}}^T(\mathbf{y} + \boldsymbol{\epsilon}), \quad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}). \tag{3.2}$$

Note that because we fixed $\mathbf{J}(\bar{\boldsymbol{\theta}})$ in (3.1), $\bar{\mathbf{Q}}$ is fixed in (3.2), and hence, the random vector $\mathbf{v}$ is Gaussian with probability density function

$$p_{\mathbf{v}}(\mathbf{v}) \propto \exp\left(-\frac{1}{2}\|\mathbf{v} - \bar{\mathbf{Q}}^T\mathbf{y}\|^2\right). \tag{3.3}$$

This will be important in a moment.

In order to make the mapping (3.2) well-defined, we have to resolve a couple of issues. First, $p_{\mathbf{v}}(\mathbf{v})$ may have support outside of the range of $\mathbf{F}_{\bar{\boldsymbol{\theta}}}$. This can be overcome by defining a new random vector $\mathbf{w}$ with probability density function

$$p_{\mathbf{w}}(\mathbf{w}) \propto \chi_{\mathcal{R}}(\mathbf{w}) p_{\mathbf{v}}(\mathbf{w}), \tag{3.4}$$

where $\chi_{\mathcal{R}}$ is the indicator function on the range of $\mathbf{F}_{\bar{\boldsymbol{\theta}}}$. We then replace (3.2) by

$$\boldsymbol{\theta} = \mathbf{F}_{\bar{\boldsymbol{\theta}}}^{-1}(\mathbf{w}), \quad \text{where} \quad \mathbf{w} \sim p_{\mathbf{w}}(\mathbf{w}). \tag{3.5}$$

Second, $\mathbf{F}_{\bar{\boldsymbol{\theta}}}$ needs to be a one-to-one function. This is guaranteed, by the inverse function theorem, if the Jacobian of $\mathbf{F}_{\bar{\boldsymbol{\theta}}}$, given by $\bar{\mathbf{Q}}^T\mathbf{J}(\boldsymbol{\theta})$, is invertible for all $\boldsymbol{\theta}$ in the domain of $\mathbf{F}_{\bar{\boldsymbol{\theta}}}$. We make this assumption.

Finally, we derive the probability density function $p_{\bar{\boldsymbol{\theta}}}(\boldsymbol{\theta})$ for $\boldsymbol{\theta}$ defined by (3.5). By the theory of transformations of a random vector, if $\mathbf{J}_{\bar{\boldsymbol{\theta}}}$ denotes the Jacobian of $\mathbf{F}_{\bar{\boldsymbol{\theta}}}$, we have

$$\begin{aligned}
p_{\bar{\boldsymbol{\theta}}}(\boldsymbol{\theta}) &= |\mathbf{J}_{\bar{\boldsymbol{\theta}}}(\boldsymbol{\theta})|\ p_{\mathbf{w}}(\mathbf{F}_{\bar{\boldsymbol{\theta}}}(\boldsymbol{\theta})) \\
&\propto \left|\bar{\mathbf{Q}}^T\mathbf{J}(\boldsymbol{\theta})\right| \chi_{\mathcal{R}}(\mathbf{F}_{\bar{\boldsymbol{\theta}}}(\boldsymbol{\theta})) p_{\mathbf{v}}(\mathbf{F}_{\bar{\boldsymbol{\theta}}}(\boldsymbol{\theta})) \\
&\propto \left|\bar{\mathbf{Q}}^T\mathbf{J}(\boldsymbol{\theta})\right| \exp\left(-\frac{1}{2}\|\bar{\mathbf{Q}}^T(\mathbf{f}(\boldsymbol{\theta}) - \mathbf{y})\|^2\right) \\
&= \left|\bar{\mathbf{Q}}^T\mathbf{J}(\boldsymbol{\theta})\right| \exp\left(\frac{1}{2}\|\tilde{\mathbf{Q}}^T(\mathbf{f}(\boldsymbol{\theta}) - \mathbf{y})\|^2 - \frac{1}{2}\|\mathbf{f}(\boldsymbol{\theta}) - \mathbf{y}\|^2\right) \\
&= c(\boldsymbol{\theta}) p(\boldsymbol{\theta}|\mathbf{y}),
\end{aligned} \tag{3.6}$$

where $p(\boldsymbol{\theta}|\mathbf{y}) \propto \exp(-\frac{1}{2}\|\mathbf{f}(\boldsymbol{\theta}) - \mathbf{y}\|^2)$ is the posterior density function, and

$$c(\boldsymbol{\theta}) = \left|\bar{\mathbf{Q}}^T\mathbf{J}(\boldsymbol{\theta})\right| \exp\left(\frac{1}{2}\|\tilde{\mathbf{Q}}^T(\mathbf{f}(\boldsymbol{\theta}) - \mathbf{y})\|^2\right). \tag{3.7}$$

We summarize our assumptions and results in the following theorem.

THEOREM 3.1. *Assume that* $p(\boldsymbol{\theta}|\mathbf{y}) \propto \exp(-\frac{1}{2}\|\mathbf{f}(\boldsymbol{\theta}) - \mathbf{y}\|^2)$, *where* $\mathbf{y} \in \mathbb{R}^m$, *and that* $\mathbf{f} : \mathbb{R}^n \to \mathbb{R}^m$ *is a continuously differentiable function with Jacobian* $\mathbf{J}(\boldsymbol{\theta}) \in \mathbb{R}^{m\times n}$, *with* $m \geq n$, *that is rank* n *for every* $\boldsymbol{\theta}$ *in the domain of* $\mathbf{f}$. *Then if the MAP estimator* $\bar{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} p(\boldsymbol{\theta}|\mathbf{y})$ *is unique,* $\mathbf{J}(\bar{\boldsymbol{\theta}}) = [\bar{\mathbf{Q}}, \tilde{\mathbf{Q}}] \left[\begin{smallmatrix} \bar{\mathbf{R}} \\ \mathbf{0} \end{smallmatrix}\right]$ *is the* $\mathbf{QR}$*-factorization of* $\mathbf{J}(\bar{\boldsymbol{\theta}})$, *and* $\bar{\mathbf{Q}}^T\mathbf{J}(\boldsymbol{\theta})$ *is invertible for all* $\boldsymbol{\theta}$ *in the domain of* $\mathbf{f}$, *the inverse mapping defined by* (3.5), (3.4), (3.3) *yields a random vector* $\boldsymbol{\theta}$ *with a probability density function of the form* $p_{\bar{\boldsymbol{\theta}}}(\boldsymbol{\theta}) \propto c(\boldsymbol{\theta})p(\boldsymbol{\theta}|\mathbf{y})$, *where* $c(\boldsymbol{\theta})$ *is given by* (3.7).

Remarks. In practice, we compute random draws from (3.5) by solving the stochastic optimization problem

$$\boldsymbol{\theta} = \arg\min_{\boldsymbol{\psi}} \left\{\ell(\boldsymbol{\psi}, \boldsymbol{\epsilon}) \stackrel{\text{def}}{=} \|\bar{\mathbf{Q}}^T(\mathbf{f}(\boldsymbol{\psi}) - (\mathbf{y} + \boldsymbol{\epsilon}))\|^2\right\}, \quad \text{where} \quad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}). \tag{3.8}$$

In theory, we reject those $(\boldsymbol{\theta}, \boldsymbol{\epsilon})$ for which $\ell(\boldsymbol{\theta}, \boldsymbol{\epsilon}) > 0$, since this implies that $\mathbf{v} = \bar{\mathbf{Q}}^T(\mathbf{y} + \boldsymbol{\epsilon})$ is not in the range of $\mathbf{F}_{\bar{\boldsymbol{\theta}}}$, and hence that $p_{\bar{\boldsymbol{\theta}}}(\boldsymbol{\theta}) = 0$ since $\chi_{\mathcal{R}}(\mathbf{F}_{\bar{\boldsymbol{\theta}}}(\boldsymbol{\theta})) = 0$.

In practice, we reject those $(\boldsymbol{\theta}, \boldsymbol{\epsilon})$ for which $\ell(\boldsymbol{\theta}, \boldsymbol{\epsilon}) > \eta$, where η is some small positive number; in all of the examples below, we used $\eta = 10^{-8}$. This procedure for computing samples from $p_{\bar{\boldsymbol{\theta}}}(\boldsymbol{\theta})$ motivates our name for the method: randomize-then-optimize. It requires only straightforward modifications to the method used to compute the MAP estimator $\bar{\boldsymbol{\theta}}$. For solving (3.8), we use the MATLAB routine `lsqnonlin` with the built-in stopping tolerances and $\bar{\boldsymbol{\theta}}$ as the initial guess, in all cases. Hence no tuning of the algorithm was needed, which we believe is due in part to the stabilizing presence of $\mathbf{Q}^T$ in (3.8).

Finally, as was stated above, it is not necessary in practice to compute the full $\mathbf{QR}$ factorization of $\mathbf{J}(\bar{\boldsymbol{\theta}})$, since (3.7) can be equivalently expressed as

$$c(\boldsymbol{\theta}) = \left|\bar{\mathbf{Q}}^T \mathbf{J}(\boldsymbol{\theta})\right| \exp\left(\frac{1}{2}\|\mathbf{f}(\boldsymbol{\theta}) - \mathbf{y}\|^2 - \frac{1}{2}\|\bar{\mathbf{Q}}^T(\mathbf{f}(\boldsymbol{\theta}) - \mathbf{y})\|^2\right).$$

4. Using RTO to sample from the posterior density function. In this section, we present a Metropolis–Hastings method that uses the RTO probability density as an independence proposal. We also show how RTO can be used within an importance sampling framework.

4.1. A Metropolis–Hastings algorithm using RTO. The samples obtained using RTO can be used within the framework of the Metropolis–Hastings (MH) algorithm [6]. In this case, the RTO density, $p_{\bar{\boldsymbol{\theta}}}(\boldsymbol{\theta})$, is used as the proposal density, and since each sample from $p_{\bar{\boldsymbol{\theta}}}(\boldsymbol{\theta})$ is independent, the result is an independence MH method. At step k, a candidate sample $\boldsymbol{\theta}^*$ is drawn from the RTO density $p_{\bar{\boldsymbol{\theta}}}(\boldsymbol{\theta})$, and is then accepted with probability r defined by the ratio (see [6])

$$\begin{aligned} r &= \min\left(1, \frac{p(\boldsymbol{\theta}^*|\mathbf{y}) p_{\bar{\boldsymbol{\theta}}}(\boldsymbol{\theta}^{k-1})}{p(\boldsymbol{\theta}^{k-1}|\mathbf{y}) p_{\bar{\boldsymbol{\theta}}}(\boldsymbol{\theta}^*)}\right) \\ &= \min\left(1, \frac{p(\boldsymbol{\theta}^*|\mathbf{y}) c(\boldsymbol{\theta}^{k-1}) p(\boldsymbol{\theta}^{k-1}|\mathbf{y})}{p(\boldsymbol{\theta}^{k-1}|\mathbf{y}) c(\boldsymbol{\theta}^*) p(\boldsymbol{\theta}^*|\mathbf{y})}\right) \\ &= \min\left(1, \frac{c(\boldsymbol{\theta}^{k-1})}{c(\boldsymbol{\theta}^*)}\right). \end{aligned} \tag{4.1}$$

We note that (4.1) can be numerically unstable, especially in large-scale cases, in which case, one can use instead $c(\boldsymbol{\theta}^{k-1})/c(\boldsymbol{\theta}^*) = \exp(\ln c(\boldsymbol{\theta}^{k-1}) - \ln c(\boldsymbol{\theta}^*))$.

Finally, the RTO-MH algorithm is given by the following.

The RTO Metropolis–Hastings algorithm.

1. Choose initial vector $\boldsymbol{\theta}^0$, parameter $0 < \eta \ll 1$, and samples N. Set $k = 1$.
2. Compute an RTO sample $\boldsymbol{\theta}^*$ using (3.8) with corresponding $\boldsymbol{\epsilon}^*$.
3. If $\ell(\boldsymbol{\theta}^*, \boldsymbol{\epsilon}^*) > \eta$, return to step 2, else go to step 4.
4. Define the acceptance probability r by (4.1).
5. Simulate $u \sim U(0,1)$. If $u < r$, set $\boldsymbol{\theta}^k = \boldsymbol{\theta}^*$, else set $\boldsymbol{\theta}^k = \boldsymbol{\theta}^{k-1}$.
6. If $k < N$, set $k = k + 1$ and return to step 2.

Remark. First, given our assumption that $\bar{\mathbf{Q}}^T \mathbf{J}(\boldsymbol{\theta})$ is invertible, $c(\boldsymbol{\theta}) \neq 0$, so that (4.1) is well-defined. Moreover, we note that the RTO-MH method can be embedded within a hierarchical sampling scheme, such as what would arise if we were also sampling the noise variance via a conjugate hyperprior, which for the variance of a Gaussian is the inverse-gamma density [6].

A different, but related, class of MCMC methods for large-scale inverse problems constructs a Gaussian proposal density at each step using the Hessian (or Gauss–

Newton approximation $\mathbf{J}(\boldsymbol{\theta})^T\mathbf{J}(\boldsymbol{\theta})$) of the posterior density function evaluated at the most recent sample; see, e.g., [12, 14]. We note that since these methods rely on Gaussian proposals, in large-scale cases, the proposed samples will be rejected a high percentage of the time if the posterior is truly non-Gaussian, which leads to convergence issues in the associated MCMC method. RTO, on the other hand, yields a non-Gaussian proposal that matches well with the support of the posterior.

4.2. Importance sampling using RTO. RTO also lends itself well to use within an importance sampling framework [15]. The idea in importance sampling is to compute an approximation of the integral

$$\int_{\boldsymbol{\theta}} g(\boldsymbol{\theta})p(\boldsymbol{\theta}|\mathbf{y})\,d\boldsymbol{\theta} \tag{4.2}$$

using a Monte Carlo method. Note that if $g(\boldsymbol{\theta}) = \boldsymbol{\theta}$, then (4.2) yields the posterior mean, $\boldsymbol{\mu}$, while if $g(\boldsymbol{\theta}) = (\boldsymbol{\theta} - \boldsymbol{\mu})^2$, (4.2) yields the posterior variance.

We can approximate (4.2) using RTO as follows: compute samples $\boldsymbol{\theta}^i \sim p_{\bar{\boldsymbol{\theta}}}(\boldsymbol{\theta})$, for $i = 1, \ldots, N$, then

$$\begin{aligned}\int_{\boldsymbol{\theta}} g(\boldsymbol{\theta})p(\boldsymbol{\theta}|\mathbf{y})\,d\boldsymbol{\theta} &= \int_{\boldsymbol{\theta}} g(\boldsymbol{\theta})\frac{p(\boldsymbol{\theta}|\mathbf{y})}{p_{\bar{\boldsymbol{\theta}}}(\boldsymbol{\theta})}p_{\bar{\boldsymbol{\theta}}}(\boldsymbol{\theta})\,d\boldsymbol{\theta}\\ &\approx \frac{1}{N}\sum_{i=1}^{N} g(\boldsymbol{\theta}^i)\frac{p(\boldsymbol{\theta}^i|\mathbf{y})}{p_{\bar{\boldsymbol{\theta}}}(\boldsymbol{\theta}^i)}\\ &= \frac{1}{N}\sum_{i=1}^{N} w_i g(\boldsymbol{\theta}^i),\end{aligned} \tag{4.3}$$

where $w_i = 1/c(\boldsymbol{\theta}^i)$ is known as an importance weight.

We can also use the RTO samples $\{\boldsymbol{\theta}^i\}_{i=1}^N$ to construct approximate samples from the posterior $p(\boldsymbol{\theta}|\mathbf{y})$, using what is known as sample importance resampling.

The RTO sample importance resampling algorithm.

1. Compute N samples $\{\boldsymbol{\theta}^i\}_{i=1}^N$ using RTO.
2. Compute the standardized importance weights $p_i = w_i/(\sum_{i=1}^N w_i)$, where $w_i = 1/c(\boldsymbol{\theta}^i)$, with $c(\boldsymbol{\theta}^i)$ defined by (3.7).
3. Resample $\{\hat{\boldsymbol{\theta}}^j\}_{j=1}^M$ from $\{\boldsymbol{\theta}^i\}_{i=1}^N$ with replacement and with probability p_i for each $\boldsymbol{\theta}^i$.

In our experience, the RTO-MH method is more numerically stable than the importance sampling approach. This is due to the fact that the correction term $c(\boldsymbol{\theta})$ defined in (3.7) can be very large, making the computation of the standardized importance weights in step 2 more subject to numerical roundoff issue than the computation of the acceptance ratio (4.1).

5. Numerical experiments. In this section, we present three numerical examples. We start with two simple algebraic models and use them to verify that the theory works and to study various aspects of the RTO method. As a more computationally challenging test case, we study a higher-dimensional parameter estimation problem arising in atmospheric remote sensing.

5.1. MONOD and BOD models. Here, we demonstrate RTO with two simple algebraic models, the MONOD model and the BOD model. Both are often used in,

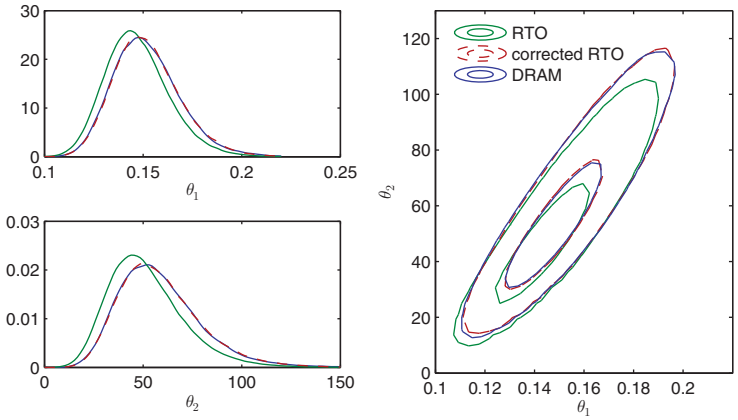

FIG. 1. *The marginal densities for the two parameters of the Monod model obtained with standard MCMC (blue) and RTO with (red) and without (green) importance weighting.*

e.g., biological modeling to describe the increase and saturation of growth of a response $\mathbf{f}(\boldsymbol{\theta})$ with respect to a given factor vector $\mathbf{x}$. The MONOD model is given by

$$\mathbf{f}(\boldsymbol{\theta}) = \frac{\theta_1 \mathbf{x}}{\theta_2 \mathbf{1} + \mathbf{x}}, \tag{5.1}$$

where $\boldsymbol{\theta} = (\theta_1, \theta_2)$; $\mathbf{1}$ is a vector of 1's of the same size as $\mathbf{x}$; and division is performed componentwise. The factor values and corresponding measurements used for the first experiment are given by

$$\mathbf{x} = [28, 55, 83, 110, 138, 225, 375]^T,$$
$$\mathbf{y} = [0.053, 0.060, 0.112, 0.105, 0.099, 0.122, 0.125]^T.$$

The data $\mathbf{y}$ were generated by adding independent and identically distributed Gaussian noise to the curve simulated using the "true" parameter values. The observation error standard deviation was approximated from the residuals of the least squares fit, which gave $\sigma = 0.012$.

We generate 300 000 samples with the delayed rejection adaptive Metropolis algorithm (DRAM, [7]) and RTO with the MH correction. For the optimization in RTO, we use the trust-region algorithm implemented in the MATLAB routine `lsqnonlin`, with an analytically computed Jacobian matrix. The results are compared in Figure 1. One can see that the plain RTO without the MH correction gives slightly different results than MCMC, but the correction makes the densities agree.

To compare the efficiency of RTO and DRAM, we plot a short part of the RTO and DRAM chains and the autocorrelation functions of the chains for θ_1 in Figure 2 (the results for θ_2 are very similar). We observe that the RTO proposal yields slightly better mixing than DRAM. On the other hand, RTO needs more work per sample. To compare the performance more precisely, we compute the integrated autocorrelation time (IACT) of the DRAM chain, which gives an estimate of how many MCMC steps are needed to obtain one independent sample. We compare this to the number of function evaluations needed in the optimization steps within RTO and the IACT of the RTO samples. In the MONOD model, the posterior is rather Gaussian, and MCMC works well, giving an IACT of around 8 for both parameters. The optimization steps

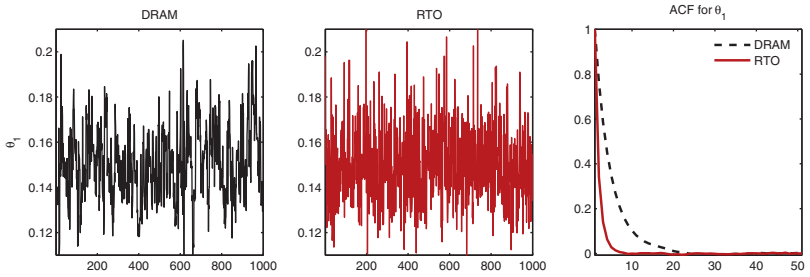

FIG. 2. 1000 *consecutive steps of DRAM (left) and RTO (middle) and the autocorrelation functions for both samplers (right).*

require, on average, 3.7 iterations per sample, and the IACT for the final RTO chain is around 2 for both parameters. The Jacobian here is simple to compute, and the computational cost of a single optimization iteration is roughly equal to one model evaluation. Thus, the computational cost of the methods is about the same.

Next, we study another simple algebraic model called the BOD model, given as

$$\mathbf{f}(\boldsymbol{\theta}) = \theta_1(1 - \exp(-\theta_2 \mathbf{x})). \tag{5.2}$$

The parameters $\boldsymbol{\theta} = (\theta_1, \theta_2)$ are estimated from data $\mathbf{x} = (1, 3, 5, 7, 9)$ and $\mathbf{y} = (0.076, 0.258, 0.369, 0.492, 0.559)$. The data $\mathbf{y}$ were generated by adding independent and identically distributed Gaussian noise to the curve simulated using the "true" parameter values. The observation error standard deviation was approximated from the residuals of the least squares fit, which gave $\sigma = 0.014$.

The problem yields a non-Gaussian, "banana-shaped" posterior distribution. The one-dimensional (1D) marginal densities obtained with standard MCMC and RTO are compared in Figure 3. Here, the proposed RTO samples without the MH correction are already rather close to the posterior obtained by MCMC, and the effect of the correction is less visible. The colors in the two-dimensional (2D) scatter plot show the logarithmic correction factors $\log(c(\boldsymbol{\theta}))$. We see that the range of the factors is relatively narrow compared to the previous MONOD case (where the range was roughly 3.3–7.6), which explains the small effect of the MH correction.

The mixing of the RTO and DRAM chains is illustrated in Figure 4 for parameter θ_1. The difference is clearer here than in the previous MONOD case, as expected, since the posterior is rather non-Gaussian and thus challenging to capture with a Gaussian proposal. The computation of a single RTO sample requires, on average, 4.6 optimization iterations. The IACT of the RTO chain is around 1.4 for both parameters, whereas the IACT for the DRAM method is around 38 for θ_1 and 27 for θ_2. The Jacobian evaluation is about the cost of one model evaluation, so RTO is more computationally efficient than DRAM for this example.

To further illuminate the characteristics of the MCMC and RTO approaches, we run the BOD model again, but this time using data that are not able to identify the model parameters properly. We take 20 linearly spaced observations, but in the smaller interval $1 \leq x \leq 5$ to define $\mathbf{x}$, and create the observations by adding Gaussian noise from $N(0, \sigma^2)$, $\sigma = 0.01$, to the model values calculated by (5.2) with the "true" parameter values $\boldsymbol{\theta} = (1, 0.1)$. The parameter posterior is now an even thinner "banana" showing strong nonlinear correlations, causing difficulties for any sampling method. Figure 5 presents the results of example runs of the respective cases: 2D scatter plots of the sampled parameters and the 1D marginal densities.

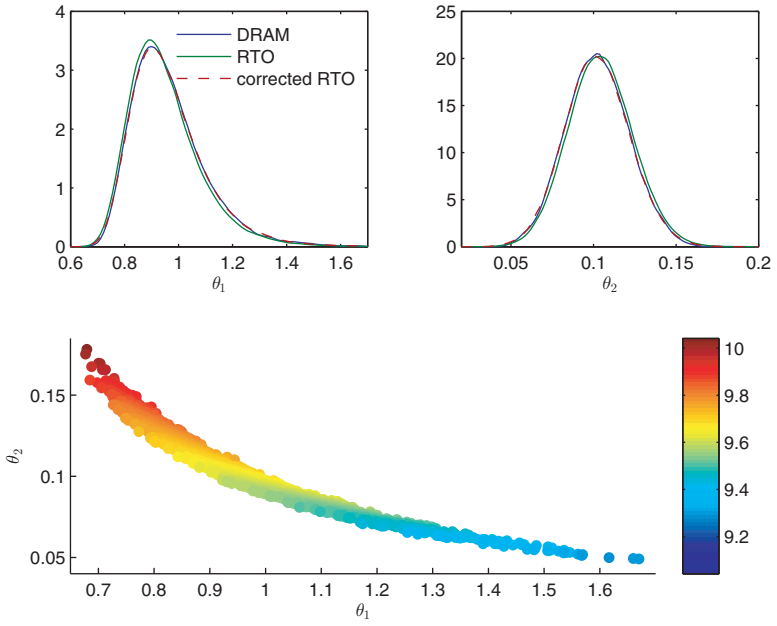

FIG. 3. *Left: the marginal* 1*D parameter densities of the BOD model obtained with standard MCMC (blue) and RTO with (red) and without (green) importance weighting. Right: the* 2*D scatter plot, colored with importance weights.*

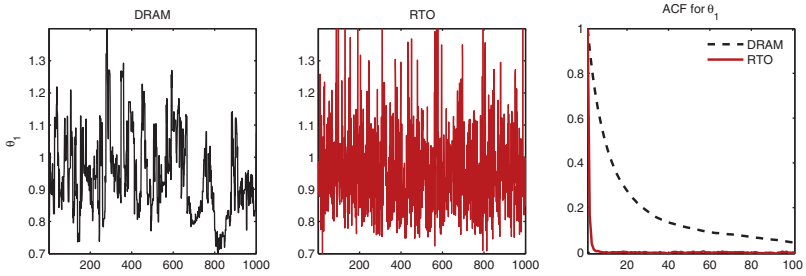

FIG. 4. 1000 *consecutive steps of DRAM (left) and RTO (middle) and the autocorrelation functions for both samplers (right).*

The 2D scatter plots show that the RTO approach is able to find more extreme values at the tails of the posteriors than MCMC. However, the 1D densities also show that on average too many RTO samples lie in the center of the posterior, close to the true parameter value. The correction factors $c(\boldsymbol{\theta})$ are lowest at the extreme tail, and the impact of the correction here is to shift the probability mass of RTO to the tail. The corrected RTO and MCMC results coincide again.

The thinner banana is even more difficult for DRAM, and the difference in mixing between RTO and DRAM is pronounced, as shown in Figure 6. However, the thinner banana target is also more challenging for the optimizer, and the optimizations in RTO require, on average, 12.1 iterations (compared to 4.6 iterations in the previous case). The IACT of the RTO chain was around 8 for θ_1 and 2 for θ_2, whereas the IACTs for DRAM were around 187 and 119. Thus, we can conclude that RTO was more efficient than DRAM in this case.

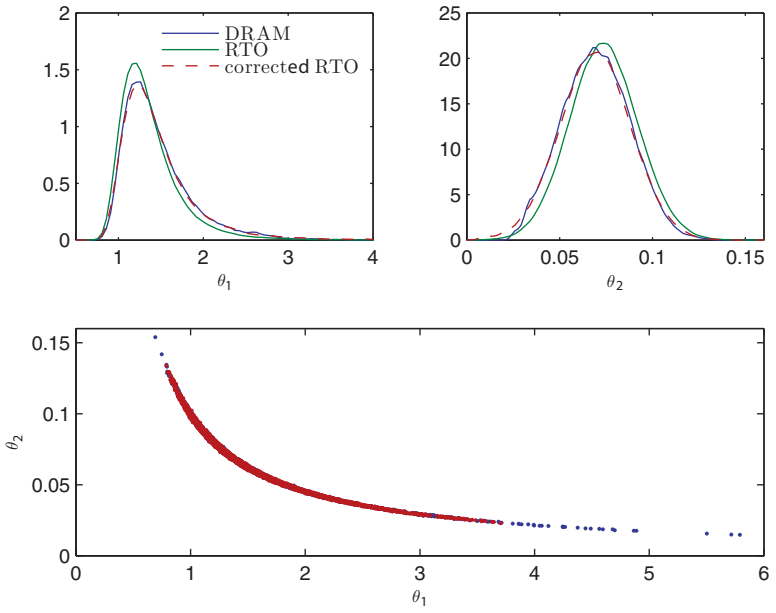

FIG. 5. *The BOD example with data that do not properly identify the parameters. Left: the marginal 1D parameter densities with standard MCMC (blue) and RTO with (red) and without (green) importance weighting. Right: the 2D scatter plots, with MCMC and RTO. The extreme points in the RTO tail with highest weights separately indicated.*

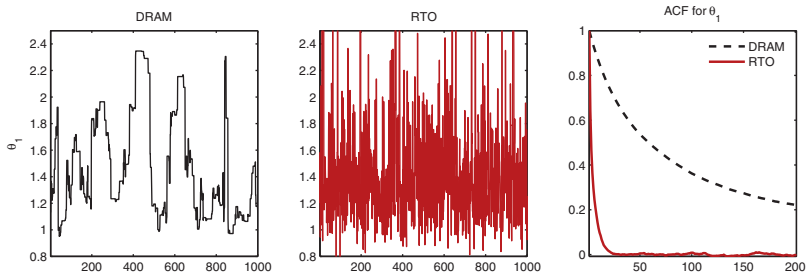

FIG. 6. 1000 *consecutive steps of DRAM (left) and RTO (middle) and the autocorrelation functions for both samplers (right).*

5.2. Atmospheric remote sensing. We consider a realistic inverse problem of the concentrations of various gases in the atmosphere using stellar occultation measurements. Such measurements were operationally performed by the Global Ozone Monitoring System (GOMOS) instrument on board ESA's Envisat satellite for 10 years, until the connection with the satellite was lost in May 2012.

In stellar occultation measurements, the absorption of starlight as it travels through the atmosphere is measured at different wavelengths. Different atmospheric constituents leave fingerprints in the measured intensity spectra. The task of the inversion algorithm is to infer the concentrations of these gases based on the measurements. The unknowns in the problem are local height profiles of concentrations of the different gases. Discretization of the profiles leads to high-dimensional inverse problems that are challenging for MCMC samplers. Here, we compare RTO to two existing MCMC algorithms (the MH sampler and the Metropolis adjusted Langevin

algorithm) in a synthetic GOMOS inverse problem. Before presenting the results, we briefly present the basics of GOMOS theory. For more details about the problem and its Bayesian treatment, see [9] and the references therein. The setting is similar to the one used to study dimension reduction techniques for MCMC in [3].

5.2.1. GOMOS theory and problem setup. In the GOMOS instrument, light intensities I_λ are measured at different wavelengths λ. The intensity spectrum is compared to a reference intensity spectrum I_{ref} measured above the atmosphere, and the resulting *transmission spectrum* is defined as $T_\lambda = I_\lambda / I_{\text{ref}}$. The transmissions at wavelength λ along the ray path l are modeled using Beer's law:

$$T_{\lambda,l} = \exp\left(- \int_l \sum_{\text{gas}} \alpha_\lambda^{\text{gas}} \rho^{\text{gas}}(z(s)) ds \right), \tag{5.3}$$

where $\rho^{\text{gas}}(z(s))$ is the density of a gas at tangential height z. The so-called cross sections $\alpha_\lambda^{\text{gas}}$, known from laboratory measurements, define how much a gas absorbs light at a given wavelength.

To approximate the integrals in (5.3), the atmosphere is discretized. The geometry used for inversion resembles an onion: the gas densities are assumed to be constant within spherical layers around the Earth. The atmosphere is discretized into n_{alts} layers, and in the inverse problem we have n_{gas} gases and n_λ wavelengths. The discretization is fixed so that number of measurement lines is equal to the number of layers. Approximating the integrals by sums in the chosen grid, and combining information from all lines and all wavelengths, we can write the model in matrix form as follows:

$$\mathbf{T} = \exp(-\mathbf{C}\boldsymbol{\Theta}^\top \mathbf{A}^\top), \tag{5.4}$$

where $\mathbf{T} \in \mathbb{R}^{n_\lambda \times n_{\text{alts}}}$ are the modeled transmissions, $\mathbf{C} \in \mathbb{R}^{n_\lambda \times n_{\text{gas}}}$ contains the cross sections, $\boldsymbol{\Theta} \in \mathbb{R}^{n_{\text{alts}} \times n_{\text{gas}}}$ contains the unknown densities, and $\mathbf{A} \in \mathbb{R}^{n_{\text{alts}} \times n_{\text{alts}}}$ is the geometry matrix that contains the lengths of the lines of sight in each layer. For more details about the GOMOS imaging setting, see, for instance, [9].

To match our theoretical framework, we vectorize the above model using the identity $\text{vec}(\mathbf{C}\boldsymbol{\Theta}^\top \mathbf{A}^\top) = (\mathbf{A} \otimes \mathbf{C})\text{vec}(\boldsymbol{\Theta}^\top)$, where $\otimes$ denotes the Kronecker product and vec is the vectorization obtained by stacking the columns of the matrix argument on top of each other. Thus, the likelihood model is written in vector form as follows:

$$\mathbf{y} = \text{vec}(\mathbf{T}(\tilde{\boldsymbol{\theta}})) + \boldsymbol{\epsilon} = \exp\left(-(\mathbf{A} \otimes \mathbf{C})\tilde{\boldsymbol{\theta}} \right) + \boldsymbol{\epsilon}, \tag{5.5}$$

where $\tilde{\boldsymbol{\theta}} = \text{vec}(\boldsymbol{\Theta}^\top)$.

The data and prior setup are similar to the ones used in [3] to study dimension reduction techniques. We generate synthetic data by solving the forward model (5.5) with known gas densities $\boldsymbol{\theta}$. These densities are chosen to represent typical gas profiles in the atmosphere. In the example, we have 4 constituent profiles to be inverted: O_3, NO_2, NO_3, and aerosols. The atmosphere is discretized into 50 layers, and the total dimension of the problem is thus 200. We estimate the log-profiles $\boldsymbol{\theta} = \log(\tilde{\boldsymbol{\theta}})$ of the gases instead of the densities directly. We set a Gaussian process prior for the profiles, which yields $\boldsymbol{\theta}_i \sim N(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$, where $\boldsymbol{\theta}_i$ denotes the elements of vector $\boldsymbol{\theta}$ corresponding to gas i. The elements of the 50×50 covariance matrices are calculated based on the squared exponential covariance function

$$C_i(s, s') = \sigma_i \exp(-(s - s')^2 / 2L_i^2). \tag{5.6}$$

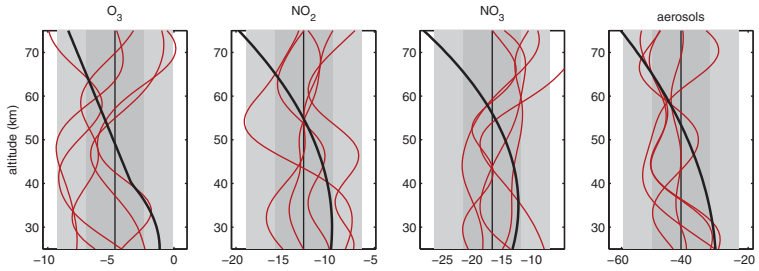

FIG. 7. *True log-profiles for the* 4 *gases (black solid lines),* 50% *and* 95% *confidence envelopes for the prior (grey areas), and* 5 *samples from the prior (red lines).*

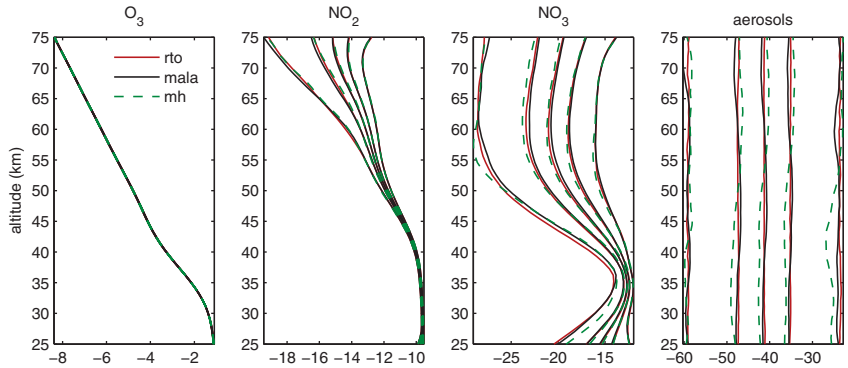

FIG. 8. *Median and* 50% *and* 95% *quantiles for the height profiles of* 4 *gases obtained by three different samplers: RTO (red), MALA (black), and MH (green).*

The prior parameters are chosen to promote smooth profiles and to give a rough idea of the magnitude of the density values. The parameter values are $\sigma_1 = 5.22$, $\sigma_2 = 9.79$, $\sigma_3 = 23.66$, $\sigma_4 = 83.18$, and $L_i = 10$ for all i. The final 200×200 prior covariance matrix is obtained by organizing the elements of the individual 50×50 matrices to match the ordering of the gases and altitudes in vector $\boldsymbol{\theta}$. The prior is illustrated by drawing samples from it in Figure 7.

5.2.2. Results. We generate 100 000 samples from the 200-dimensional posterior by three different methods: the proposed RTO method, the MALA algorithm, and the standard MH algorithm. In the MALA algorithm, we use the Hessian approximation $\mathbf{J}(\bar{\boldsymbol{\theta}})^T\mathbf{J}(\bar{\boldsymbol{\theta}})$ computed at the MAP estimate as the preconditioner, which leads to the stochastic Newton sampler [12]. For the MH sampler, we use the Gaussian approximation obtained by computing the inverse Hessian approximation $(\mathbf{J}(\bar{\boldsymbol{\theta}})^T\mathbf{J}(\bar{\boldsymbol{\theta}}))^{-1}$ as the proposed covariance matrix.

The posterior for the 3 methods is summarized in Figure 8. Ozone (O_3) is very accurately identified by the data, whereas the aerosols remain totally unidentified. NO_2 and NO_3 are well identified at some altitudes and poorly at others. We observe that all methods agree well; the RTO method is able to give the same results as two existing, provably convergent MCMC samplers.

Next, we compare the performance of the three samplers in terms of the IACT. In Figure 9, we plot the IACT for each parameter. One can see that the standard MH sampler has the highest values and the gradient information used in the MALA

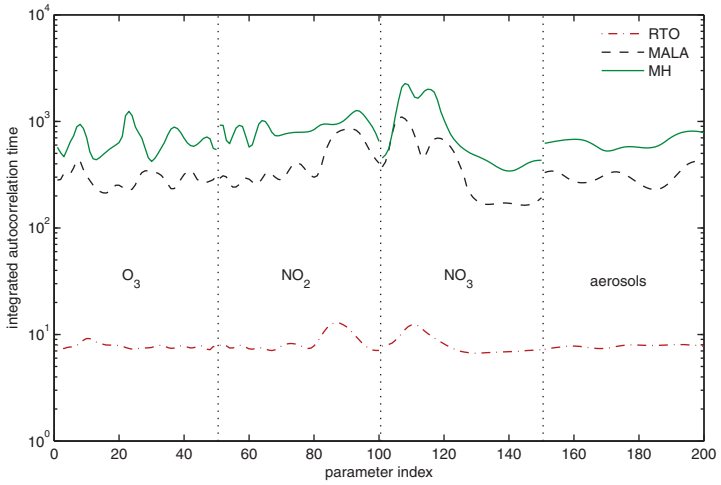

FIG. 9. *Integrated autocorrelation times for the parameters for RTO (red), MALA (black), and MH (green). First* 50 *parameters correspond to gas* 1*, second* 50 *to gas* 2*, and so on.*

algorithm reduces the IACT values compared to MH, as expected. With the RTO sampler, we obtain almost two orders of magnitude smaller autocorrelation times. This comes with a price though; computing each RTO sample is computationally more challenging than obtaining a single MH or MALA sample. However, the optimizations within RTO converged quite fast in this problem: RTO required, on average, 6.6 iterations per sample with the MATLAB routine `lsqnonlin`. Thus, in this case, RTO seems to be more efficient than the other two samplers, even when the added computational cost of RTO is taken into account.

6. Remarks. We collect here a few remarks about the RTO method and some experiences gained by the numerical examples.

Remark 1. Note that in the linear case, $c(\boldsymbol{\theta})$ defined by (3.7) is constant, and hence nonlinear RTO reduces to linear RTO, as defined in (2.1). Thus the effect of the nonlinearity in $\mathbf{f}$ on the RTO density $p_{\bar{\boldsymbol{\theta}}}(\boldsymbol{\theta})$ is entirely encoded in the expression (3.7) for $c(\boldsymbol{\theta})$. Also, care must be taken when computing $c(\boldsymbol{\theta})$, as it can be unstable due to the presence of the determinant. Moreover, in order to minimize computational cost, only the thin **QR**-factorization is needed when implementing RTO and computing $c(\boldsymbol{\theta})$.

Remark 2. For RTO implementations, it is necessary to compute the Jacobian matrix for each optimized parameter sample. While this could be done analytically in our small-dimensional toy cases, one often has to resort to numerical approximations. RTO might be sensitive with respect to the quality of the approximation. More complex dynamical models might require more elaborate calculations of the derivatives, such as integration of the additional sensitivity equations or the use of adjoint and tangent linear codes.

Remark 3. The success of RTO clearly depends on the efficiency of the optimizer and its tuning, e.g., the choice of stopping rules and initial parameter estimates. However, we note that in the examples we've considered, the presence of $\mathbf{Q}^T$ within the RTO least squares function (see (3.8)) stabilizes the optimization, so that the default tolerances with the MATLAB routine `lsqnonlin` are sufficient. Moreover, as the initial parameter estimate, we have exclusively used the MAP estimator $\bar{\boldsymbol{\theta}}$. In the

event that the optimization problem is well-posed but ill-conditioned, it may be that the set of parameter values satisfying the stopping criteria is relatively large, in which case such a deterministic selection of initial guess can lead to biased results. In this case, one could instead use a randomized initial guess drawn, for instance, from the prior.

Remark 4. The numerical efficiency of RTO against MCMC is rather case dependent. The benefit of basic MCMC is the ease of calculations: only cost function evaluations are needed. However, the autocorrelation times tend to increase with problem dimension, even with ideal proposal distributions. While RTO needs more CPU for each sample, the samples can have much lower autocorrelation times, as observed in the numerical examples of this paper. Moreover, no burn-in sampling is needed, and the optimizations can trivially be run in parallel.

Remark 5. A web site devoted to RTO together with the downloadable MATLAB codes for implementing the method, with examples, can be found at http://helios.fmi.fi/~lainema/rto/.

7. Conclusions. We have presented an alternative to traditional MCMC for sampling from posterior density functions that arise in nonlinear inverse problems when both the measurement error and prior are Gaussian. The method requires the repeated solution of a stochastic optimization problem, and we call it RTO. The randomization step is performed by adding noise from a known Gaussian distribution to the measured data and prior mean. With this perturbed "data" in hand, an application of the optimizer then yields an RTO sample.

We show that the RTO sample density has the form $p_{\bar{\boldsymbol{\theta}}}(\boldsymbol{\theta}) \propto c(\boldsymbol{\theta})p(\boldsymbol{\theta}|\mathbf{y})$, where $c(\boldsymbol{\theta})$ depends on the Jacobian matrix and residual evaluated at $\boldsymbol{\theta}$, and $p(\boldsymbol{\theta}|\mathbf{y})$ is the posterior density function. This form of the density allows us to embed RTO within MH (and also importance sampling) in a straightforward manner. While the RTO approach requires more work per sample than traditional MCMC samplers, the quality of the proposals can be much better, especially in non-Gaussian and high-dimensional problems, as we observe in our numerical examples.

REFERENCES

[1] J. M. Bardsley, *MCMC-based image reconstruction with uncertainty quantification*, SIAM J. Sci. Comput., 34 (2012), pp. A1316–A1332.

[2] Y. Chen and D. Oliver, *Ensemble randomized maximum likelihood method as an iterative ensemble smoother*, Math Geosci, 44 (2012), pp. 1–26.

[3] T. Cui, J. Martin, Y. Marzouk, A. Solonen, and A. Spantini, *Likelihood-Informed Dimension Reduction for Nonlinear Inverse Problems*, Inverse Problems, to appear.

[4] A. Doucet, N. D. Freitas, and N. Gordon, *Sequential Monte Carlo methods in practice*, Stat. Eng. Inf. Sci., Springer, New York, 2001.

[5] B. Efron and R. J. Tibshirani, *An Introduction to the Bootstrap*, Chapman and Hall/CRC, New York, 1993.

[6] D. Gamerman and H. F. Lopes, *Markov Chain Monte Carlo – Stochastic Simulation for Bayesian Inference*. 2nd ed., Chapman and Hall/CRC, Boca Raton, FL, 2006.

[7] H., Haario, M. Laine, A. Mira, and E. Saksman, *DRAM: Efficient adaptive MCMC*, Statist. Comput., 16 (2006), pp. 339–354.

[8] H. Haario, E. Saksman, and J. Tamminen, *An adaptive Metropolis algorithm*, Bernoulli, 7 (2001), pp. 223–242.

[9] H. Haario, M. Laine, M. Lehtinen, E. Saksman, and J. Tamminen, *Markov chain Monte Carlo methods for high dimensional inversion in remote sensing*, J. R. Stat. Soc. Ser. B Stat. Methodol., 66 (2004), pp. 591–607.

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

[11] J. P. Kaipio and E. Somersalo, *Statistical and Computational Inverse Problems*, Springer, New York, 2005.
[12] J. Martin, L. C. Wilcox, C. Burstedde, and O. Ghattas, *A stochastic Newton MCMC method for large-scale statistical inverse problems with application to seismic inversion*, SIAM J. Sci. Comput., 34 (2012), pp. A1460–A1487.
[13] G. Nicholls and C. Fox, *Prior modelling and posterior sampling in impedance imaging*, in Bayesian Inference for Inverse Problems, Proc. SPIE, 3459, SPIE, Bellingham, WA, 1998, pp. 116–127.
[14] Y. Qi and T. P. Minka, *Hessian-based Markov chain Monte-Carlo algorithms*, in First Cape Cod Workshop on Monte Carlo Methods, Cape Cod, MA, September, 2002.
[15] C. Robert and G. Casella, *Monte Carlo Statistical Methods*, 2nd ed., Springer, New York, 2004.
[16] C. R. Vogel, *Computational Methods for Inverse Problems*, Front. Appl. Math. 23, SIAM, Philadelphia, 2002.
[17] D. Watzenig and C. Fox, *A review of statistical modeling and inference for electrical capacitance tomography*, Meas. Sci. Tech., 20 (2009), 052002.