submitted to *Geophys. J. Int.*

RamBO: Randomized blocky Occam, a practical algorithm for generating blocky models and associated uncertainties.

Eliana Vargas Huitzil, Matthias Morzfeld, Steven Constable

evargashuitzil@ucsd.edu, mmorzfeld@ucsd.edu, sconstable@ucsd.edu

Institute of Geophysics & Planetary Physics, Scripps Institution of Oceanography,

University of California, San Diego, La Jolla, CA 92093-0225

3 December 2024

SUMMARY

Key words: Regularized inversion; blocky models; total-variation regularization; electromagnetic inversion

1 INTRODUCTION

Some geophysicists are lucky, and maps or images of their data carry meaningful information that is directly interpretable in terms of geological structure. Examples include maps of the gravity or magnetic field and seismic or radar reflection profiles. Those of us who work with electromagnetic methods are not so lucky, and from the beginning have had to use some sort of inverse method to extract models of electrical resistivity from otherwise obscure data (e.g. Parker (1970); Inman et al. (1973)). Of course, other geophysicists use inverse methods also, particularly those who seek the seismic velocity structure of the mantle, but as Sven Treitel (personal communication) pointed out, the electromagnetic community has made significant contributions to inverse methods because it needs them more than most.

Model space is infinite – even for a one dimensional resistivity function of depth – yet data are both finite and noisy. This means that the problem is under-determined and ill-posed, and also non-unique; if one solution fits the data then an infinite number will. Early approaches to tackling these problems were to reduce the size of model space by inverting for the resistivities and thicknesses of a small number of layers (Inman et al. 1973) or by solving for averages over some kind of resolving kernel (Parker 1970). Layered inversions have the problem that the solution depends on the number of layers chosen *a priori*, and including too many layers made the inverse problem unstable. Resolving kernels also had to be chosen *a priori*. For nonlinear problems the linearized iterative inversion scheme had to be started fairly close to a solution in both cases.

The introduction of a smoothing regularization algorithm called Occam's inversion (Constable et al. 1987) solved all of these problems. An Occam model can be made from any number of layers and the smoothing regularization keeps the inversion stable and independent of the layer number. A problem with an infinite number of solutions was collapsed to a single unique solution – the smoothest model (as defined by the particular regularization chosen) that fits the data adequately. The inversion is stable enough that starting from a featureless half-space is possible and indeed desirable. Although introduced for one dimensional (1D) problems it was readily scaled up to 2D (DeGroot-Hedlin & Constable 1990) and 3D (Siripunvaraporn & Sarakorn 2011) geometries. The Occam approach has become ubiquitous in geophysical inversion, but it has its problems.

The first problem is that if Earth resistivity structure is not smooth, then Occam's inversion can produce artifacts in the model and a bias in estimated depth of structure. This is not an "academic" problem – sharp resistivity contrasts can occur in the real world, such as edges of sedimentary basins, faults, and many other geological structures. If smooth inversions are carried out for models that have sharp changes in resistivity one observes a Gibbs type phenomenon (Gibbs 1899), in which the regularized inversion overshoots the resistivity jump[⋆]. We illustrate this Gibbs phenomenon with a simple synthetic model study in which MT data with various error levels are inverted for a jump in resistivity (see Appendix A for details). The resulting models are displayed in Figure 1(a), showing that once the error is below 10% an overshoot

[⋆] The Gibbs phenomenon in Occam inversions has been known since the introduction of the algorithm, but to the best of our knowledge never documented in print.

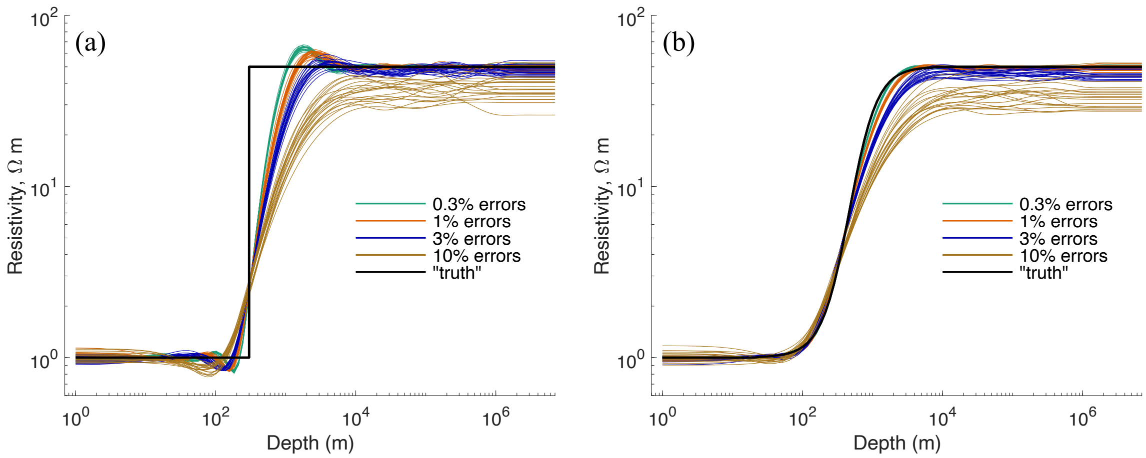


Figure 1. Inversion of synthetically generated MT data with various levels of noise added. (a): Starting model ("truth", black) is a step increase in resistivity. (b): Starting model is a smooth (sigmoid, black) increase in resistivity. In both panels, green lines correspond an error level of 0.3%, orange lines correspond to an error level of 1%, blue lines correspond to an error level of 3%, golden lines correspond to an error level of 10%. The true resistivity of this synthetic numerical experiment is shown in black in both panels.

develops on both sides of the resistivity jump, but more so on the resistive side (something that persists if the layers are swapped to make the top layer resistive). There is the danger that for more complicated models the spurious peaks in resistivity could be interpreted as real structure. Taking the midpoint of the resistivity change in the regularized models over-estimates the depth of the resistivity jump by about a factor of 2. We can verify that smooth inversions recover smooth models without such artifacts. In Figure 1(b) the step function is replaced with a sigmoid function. No overshoot is observed as the error level is reduced, and all except inversions of the most noisy data recover the model faithfully.

A second problem is that creating a uniquely smoothest model makes it extremal. The resistivity contrasts are thus the minimum required to fit the data, not the most likely. A bounded model can be useful in many circumstances, but sometimes the best estimate of the actual rock resistivity is what is wanted, say for a porosity estimate. In Figure 1, it can be seen that in both cases the models generated from data with 10% noise underestimate the half-space resistivity by up to 40%.

The third problem is that it is difficult to assign any sort of uncertainty measure to a regularized model. Even for sparsely parameterized layered models, projecting the data errors back into the model parameters though the inversion matrix is only valid if the model parameters are fully independent. Otherwise a singular value decomposition is used to identify independent eigen-parameters (Inman et al. 1973), but even these are only based on linearizations around the final solution. For a regularized model with many parameters the smoothing function creates covariance between all parameters, and additionally the number of parameters can be increased without changing the solution, so the uncertainty in any single parameter is a meaningless concept. The method currently in vogue for uncertainty quantification (UQ) in inverse problems, quasi-random Markov chain Monte Carlo (MCMC) searches of model space, must resort to using sparsely parameterized models in order to force stability and limit computational cost (see, e.g., Malinverno (2002); Blatter et al. (2021) for applications of MCMC in EM geophysics).

In this paper we present algorithms that provide all the benefits of Occam's inversion but that can (i) recover sharp resistivity contrasts; (ii) generate a UQ; and (iii) give an estimate of the most probable models. We first consider a single inversion (no UQ) and enforce a blocky model by swapping the smoothing regularization for a Total Variation (TV) regularization (Rudin et al. 1992). TV regularization has had a great successes in image deblurring and compressed sensing, and we incorporate it into a nonlinear Occam-style inversion which we call "*blocky Occam*" (see Section 3). Blocky Occam follows the tried and true recipe of an Occam's inversion. We linearize around the current model and obtain a *linear* TV-regularized problem. We then adjust the regularization strength to minimize misfit of the *nonlinear* model. These steps are iterated until convergence. Key to success here is our use of the split Bregman method (Goldstein & Osher 2009) to solve the linearized TV-regularized problem at each iteration (see Section 2.4). Split Bregman is one of the fastest methods to solve linear TV-regularized inverse problems, but it has not been used within an iterative, nonlinear inversion.

We equip blocky Occam with a UQ via a modified "*randomize-then-optimize*" (RTO) approach (see Section 2.2). RTO generates a UQ by repeatedly solving perturbed inverse problems and RTO has been used for years under various names in various fields. In short, we can re-purpose blocky-Occam for UQ, by essentially running blocky-Occam inversions in a parallel for-loop on perturbed inverse problems. We call the resulting algorithm RamBO (randomized blocky-Occam).

The use of blocky-Occam and RamBO is illustrated on two marine EM data sets (Constable et al. 1984; Gustafson et al. 2019) and we compare the new inversions and UQ to Occam's inversions and UQs obtained via trans-dimensional MCMC (Malinverno 2002; Blatter et al. 2019). Code for blocky-Occam and RamBO is available on github and Zenodo (links to the code will be included at a later stage).

2 BACKGROUND

We provide some background materials to set up the notation, to review Occam's inversion and uncertainty quantification (UQ) via randomization of a cost function. We also briefly review the split Bregman method for solving linear inverse problems with total variation (TV) regularization.

2.1 Occam's inversion: finding the smoothest model

Regularized inversion remains the standard method for solving geophysical inverse problems. The basic idea is to define and subsequently optimize a cost function that combines data misfit and model regularization (see, e.g., Parker 1994). To set up the notation, we denote the data by the n_d-dimensional vector d, the unknown model parameters (e.g. resistivities) of a discretized model are stored in the n_m-dimensional vector m and the forward model that predicts the data (usually a sophisticated computer code) is denoted by $\mathcal{F}(m)$. Errors associated with the data are stored in a $n_d \times n_d$ (diagonal) matrix W (reciprocal error weights). A typical cost function can now be written as

$$\mathcal{C}(m) = \|W(\mathcal{F}(m) - d)\|^2 + \mu \|Dm\|^2 , \tag{1}$$

where D is a finite differencing matrix and where two vertical bars denote the ℓ_2-norm of a vector, i.e., $\|x\|_2 = \sqrt{\sum_i x_i^2}$. Throughout, we will refer to the first term of the cost function as the "data-misfit" and the second term as the "regularization." The "strength" of the regularization is controlled by the scalar $\mu > 0$.

Occam's inversion (Constable et al. 1987) is an iterative algorithm that has been used for decades for regularized inversion. During the iteration, Occam's inversion adjusts the regularization strength μ and finds the smoothest model that fits the data – the quadratic regularization term favors smooth models. The iteration of Occam's inversion is as follows. At step k, the model is m_k and we approximate the forward model via Taylor expansion:

$$\mathcal{F}(m_{k+1}) \approx \mathcal{F}(m_k) + J_k(m_{k+1} - m_k), \tag{2}$$

where $J_k = \partial \mathcal{F}/\partial m$ is the Jacobian matrix, evaluated at m_k. Using the linearization in (1), yields a quadratic cost function for m_{k+1}

$$\mathcal{C}(m_{k+1}) = \left\| W(J_k m_{k+1} - \hat{d}) \right\|^2 + \mu \left\| D m_{k+1} \right\|^2, \tag{3}$$

where

$$\hat{d} = d - \mathcal{F}(m_k) + J_k m_k, \tag{4}$$

is "a kind of data vector" that accounts for errors due to linearization. We can now easily optimize the quadratic function (least squares) to find m_{k+1} and we do so for various regularization strengths μ. Once a regularization μ is selected, the process repeats until the iteration converged or reached a desired root mean squared error (RMS)

$$\text{RMS} = \frac{1}{\sqrt{n_d}} \left\| W(d - \mathcal{F}(m)) \right\|. \tag{5}$$

A good choice for a target RMS is one or slightly larger. During the iterations, we either chose μ to minimize RMS (of the *nonlinear* model) or, if RMS is below the target RMS, we use the *largest* μ that results in the target RMS. Some implementations of Occam's inversion, e.g., MARE2DEM (Key 2016), include a "fast Occam" option which dispenses with the line search minimization and accepts any μ that decreases misfit at a given iteration.

2.2 Uncertainty quantification for Occam's inversion

The popular approach to uncertainty quantification (UQ) is via Bayes' theorem, which states that

$$p(m|d) \propto p(d|m)p(m), \tag{6}$$

where $p(m|d)$ is the probability of the model given the data (the posterior probability), $p(m)$ is a prior probability of the model (often taken to be Gaussian), and where $p(d|m)$ is the likelihood, connecting the model m to the data d via the forward model $\mathcal{F}$. The symbol $\propto$ denotes proportionality, i.e., the quantity to the left differs from the quantity to the right by a multiplicative constant. In Bayes' theorem, the missing constant is the probability of the data, $p(d)$, which is called the "evidence." The evidence is not so relevant for UQ, but it can be useful for model selection (Sambridge et al. 2006).

There are many connections between regularized inversion and Bayesian UQ (see, e.g., Blatter et al. 2022a). For example, we can interpret a classical Occam-style optimization (with a cost function as in equation (1)) as the search for the model that maximizes the posterior probability

$$p(m|y) \propto \exp\left(-\frac{1}{2}\left(\|W(\mathcal{F}(m)-d)\|^2+\mu\,\|Dm\|^2\right)\right). \tag{7}$$

These connections between a Bayesian posterior distribution and optimization can be exploited to yield efficient and scalable, but approximate sampling methods for UQ. Specifically, one can sample the posterior distribution by solving perturbed optimization problems

$$\arg\min_m \left(\|W\left(\mathcal{F}(m)-(d+\eta)\right)\|^2+\mu\,\|Dm+\xi\|^2\right), \tag{8}$$

where η and ξ are Gaussian random variables that represent perturbations to the data (η) and to the regularization (ξ). More specifically, the data perturbations η are mean zero Gaussians and their covariance is matrix is $(W^TW)^{-1}$, which is representative of the assumed errors in the data. The perturbations ξ are mean zero Gaussian with covariance matrix $(1/\mu)I$, where I is the $n_m \times n_m$ identity matrix. Both perturbations (data and regularization) are needed or else variances may be underestimated (see Blatter et al. 2022a).

The above optimization-based sampling process has been invented and re-invented in many fields. It is called RTO (randomize-then-optimize, Bardsley et al. (2014); Blatter et al. (2022a)) in the mathematical community, "ensemble of data assimilation" in numerical weather prediction (Isaksen et al. 2010), it goes by the name of "randomized maximum likelihood" in the oil and gas industry (Oliver et al. 1996; Chen & Oliver 2012), and is referred to as "parametric bootstrapping sampling" in hydrology (Kitanidis 1995; Lee & Kitanidis 2013). The process is thus well-understood and known to scale to large models and large data sets. We note, however, that RTO is exact only if the forward model is linear, but RTO has proven to be very useful for solving nonlinear problems in a large number of very different applications.

2.3 Blocky models

The philosophy behind Occam's inversion is to construct models devoid of features not required by the data, achieved by finding the smoothest model (in some sense). However, many, perhaps even most, geological features of interest are associated with rapid, not smooth, changes in physical properties. Examples include the interface between sedimentary and igneous or volcanic rocks, groundwater tables, edges of magmatic reservoirs, fault structures, and many others. Occam models are useful in such circumstances because the interpreter understands that sharp boundaries will be smoothed by the inversion algorithm, but the actual boundary in question is not localized in space, and the physical property contrast (e.g. electrical resistivity) is smaller than it is in the true Earth.

One way forward is to move from quadratic (Thikonov) regularization to ℓ_1-norm regularization, which produces "blocky" (piecewise constant) models. Indeed, smooth and blocky inversions have competed with each other for decades (see, e.g., Portniaguine & Zhdanov 1999; Farquharson & Oldenburg 1998), and variations of the idea have been pondered over for many years, (see, e.g., Farquharson & Oldenburg 1998; Portniaguine & Zhdanov 1999; Guitton & Symes 2003; Theune et al. 2010; Lee & Kitanidis 2013; Sun & Li 2014; Wang et al. 2017; Fournier & Oldenburg 2019; Tang et al. 2021; Wei & Sun 2021). But the methods have never really found their way to mainstream applications. We suspect that the reasons include that some methods are computationally expensive, while others are awkwardly described or unnecessarily complicated. Moreover, some methods do not address the required search over the "nuisance" parameter μ and a UQ has rarely (if ever) been attempted. We address these issues and port ℓ_1 regularization ideas to the well-known, robust and efficient framework of Occam's inversion. We then further equip our inversions with an efficient UQ, implemented via a modified RTO approach.

2.4 Split Bregman

Before describing our nonlinear inversion algorithms, we take a short detour and discuss the solution of *linear* inverse problems with total variation (TV) regularization via split Bregman (Goldstein & Osher 2009). Specifically, we wish to minimize

$$\mathcal{C}(x) = \|Jm - d\|^2 + \mu\,|Dm|\,, \tag{9}$$

where m and d are vectors of size m_n and m_d, J is a $n_d \times n_m$ matrix, D is a finite difference matrix and $\mu > 0$ is a (given) scalar; here $|\cdot|$ denotes the ℓ_1-norm, i.e., for a n_x-dimensional vector

$$|x| = \sum_{i=1}^{n_x} |x_i|. \tag{10}$$

The regularization $|Dm|$, i.e., the ℓ_1 norm applied to the derivative of the unknown m is often called total variation (TV) regularization (Rudin et al. 1992).

The split Bregman method, applied to this problem, introduces the auxiliary variable $u = Dm$ and the Bregman variable b to reformulate the cost function as

$$\mathcal{C}_{\text{Breg}}(m,u) = \|Jm - d\|^2 + \mu|u| + \gamma\|u - Dm - b\|^2 \tag{11}$$

where γ is a second Lagrange multiplier. The above cost function is optimized by iterating the following three steps.

(i) For a given u_k and b_k, minimize $\mathcal{C}_{\text{Breg}}$ over m by solving the least squares problem

$$m_{k+1} = \arg\min_m \left(\|Jm - d\|^2 + \gamma\|u_k - Dm - b_k\|^2\right) \tag{12}$$

(ii) Given b_k and m_{k+1}, minimize $\mathcal{C}_{\text{Breg}}$ over u by solving the optimization problem

$$u_{k+1} = \arg\min_u \left(\mu|u| + \gamma\|u - Dm_{k+1} - b_k\|^2\right). \tag{13}$$

The solution is a soft-thresholding so that

$$u_{k+1} = \text{ST}(Dm_{k+1} + b_k; 2\mu/\gamma), \tag{14}$$

where

$$\text{ST}(x;\alpha) = \text{sign}(x)\max\left(|x| - \alpha, 0\right) \tag{15}$$

is the soft-thresholding function (applied element-wise to the vector in (14)).

(iii) The third step updates the Bregman variable

$$b_{k+1} = b_k + (Dm_{k+1} - u_{k+1}). \tag{16}$$

The above three steps are iterated until we reach convergence (which can sometimes be guaranteed). Note that all three steps are easy to implement and scalable: step (i) is a least squares solve; step (ii) is a simple soft-thresholding; and step (iii) is a simple updating (vector addition and matrix-vector multiplication). Indeed, split Bregman is arguably the fastest and most robust (Goldstein & Osher 2009) method for minimizing the TV regularized cost function (9) and, has been very successfully applied to various large scale linear inverse problems.

In the numerical illustrations in Section 5 we set $\gamma = 2\mu$ (as recommended) and use a simple convergence criteria and stop the iteration if

$$\frac{\|m_{k+1} - m_k\|}{\|m_k\|} \leq \text{tol}_{\text{SB}}, \tag{17}$$

where we set the tolerance $\text{tol}_{\text{SB}} = 10^{-4}$, or if a maximum number of iterations is reached. We set the maximum number of iterations to 300.

3 BLOCKY OCCAM

We now describe "a kind of" Occam's inversion which we call *blocky Occam*. Blocky Occam discovers the blockiest model that fits the data with the fewest changes in resistivity. To find blocky models, we swap the quadratic regularization in (1) with a total-variation (TV) regularization (Rudin et al. 1992)

$$\mathcal{C}(m) = \|W(\mathcal{F}(m) - d)\|^2 + \mu\,|Dm|\,, \tag{18}$$

where $|\cdot|$ denotes the ℓ_1-norm. The TV regularization ($\mu\,|Dm|$) enforces sparsity of the *derivative* of the model, because we apply the sparsity promoting ℓ_1-norm to the model's derivative. For these reasons, TV regularization promotes piece-wise constant, blocky models as desired.

We mimic the classical Occam's inversion and set up an iteration. Linearizing (see equation (2)) around the current iterate m_k gives

$$\mathcal{C}(m_{k+1}) = \left\|W(J_k m_{k+1} - \hat{d})\right\|^2 + \mu\,|Dm_{k+1}|\,, \tag{19}$$

Algorithm 1 Blocky Occam

while $k \leq k_{\text{max}}$ **do**

Compute the Jacobian J_k and the modified data vector $\hat{d} = d - \mathcal{F}(m_k) + J_k m_k$

for $\mu \in [\mu_{\text{min}}, \mu_{\text{max}}]$ **do**

Apply split Bregman to solve the optimization problem

$$\arg\min_{m_{k+1}} \left\| W(J_k m_{k+1} - \hat{d}) \right\|^2 + \mu \left| D m_{k+1} \right|,$$

Compute RMS of the optimizer using the nonlinear model $\mathcal{F}(\cdot)$

end for

if RMS $\leq$ RMS$_{\text{target}}$ **then**

Pick largest μ that leads to RMS below target

else

Pick μ to minimize RMS

end if

$m_k \leftarrow m_{k+1}$

end while

where, as in Occam's inversion, J_k is the Jacobian of the forward model and $\hat{d} = d - \mathcal{F}(m_k) + J_k m_k$ (compare the above equation with (3)). In Occam's inversion, one obtains a least squares problem after linearization (which is easy to solve). Linearization in blocky Occam leads to a linear TV-regularized inverse problem. This problem can be solved efficiently with split Bregman (see Section 2.4) for a range of regularization parameters μ. Once we chose a μ, we can proceed with the iteration. During the iterations, we either chose μ to minimize RMS or, if RMS is below the target RMS, we use the *largest* μ that results in the target RMS (generating the blockiest model). One may consider adapting ideas of fast Occam (Key 2016) to the TV-regularized problem. We summarize blocky Occam in Algorithm 1.

Blocky Occam inherits the robustness and numerical efficiency from Occam's inversion:

(i) The regularization strength is adjusted automatically during the iteration, which enhances robustness of the iteration and almost always results in quick convergence (rarely divergence). The only tunable parameter in blocky Occam is the desired target RMS and the initial model, which is usually a half space (constant resistivity). The same is true for classical Occam's inversion.

(ii) Just as in Occam's inversion, one does not need to worry about the layer thickness or, more generally the grid of the forward model. The TV regularization enforces blocky models with few resistivity changes independently of the underlying grid.

(iii) One can create a blocky Occam code with only minor modifications to a classical Occam code. The only difference is that we swap the least squares solves after linearization with a split Bregman method, which is also easy to implement and scalable (almost like least squares). The additional Lagrange multiplier that occurs during split Bregman is adjusted automatically and in accordance with the regularization strength μ.

In Section 5 we demonstrate how to use blocky Occam on the Schlumberger data set (Constable et al. 1984) classical data set (also used in Constable et al. (1987), Malinverno (2002), and Blatter et al. (2022a)) and a more recent magnetotelluric (MT) data set collected offshore New Jersey (Gustafson et al. 2019) (also used in Blatter et al. (2022b)). Implementation and testing in 2D models will be done in future work in the context of specific electromagnetic data sets.

4 RANDOMIZED BLOCKY OCCAM

It is desirable and increasingly important to not only invert for one model, but to equip the inversion with an estimate of associated uncertainties in the model. We use a randomize-then-optimize (RTO) approach (Bardsley et al. 2014), originally proposed by Kitanidis (1995) and extended to TV regularized problems by Lee & Kitanidis (2013). The RTO approach entails solving perturbed optimization problems with perturbed cost functions

$$\mathcal{C}(m) = \|W(\mathcal{F}(m) - (d + \eta))\|^2 + \mu\,|Dm + \nu|\,, \tag{20}$$

where, as before, η is Gaussian with mean zero and covariance matrix $(W^TW)^{-1}$ and where $\nu \sim \mathcal{L}(0, 1/\mu)$ has a Laplace distribution with scale parameter $1/\mu$. We can optimize the perturbed cost functions using the blocky Occam described above, but with *fixed* regularization strength μ. The implementation is easy and only requires that we replace the data d in the cost function (18) by the perturbed data $(d+\eta)$ and that we account for the perturbation ν in split Bregman (which

Algorithm 2 Randomized blocky Occam (RamBO)

for $k \leq k_{\max}$ **do**

Draw a sample η_k from $\eta \sim \mathcal{N}(0, (W^T W)^{-1})$ and a sample ν_k from $\nu \sim \mathcal{L}(0, 1/\mu)$.

Use blocky Occam with fixed μ to solve the perturbed optimization problem

$$\arg\min_m \|W(\mathcal{F}(m) - (d + \eta_k))\|^2 + \mu\, |Dm + \nu_k| \,,$$

end for

we describe in the Appendix B). The resulting procedure, which we call "randomized blocky Occam" (RamBO), is summarized in Algorithm 2 and essentially amounts to running blocky Occam within a (parallel) for-loop. For numerical efficiency, we initialize all optimizations during RamBO with the result of a blocky Occam (with adjustable regularization strength μ as described in Section 3).

Note the blocky Occam within RamBO does not automatically adjust the regularization strength μ. For that reason, the iteration can be slightly less stable and we introduce a stepsize $\alpha \in (0, 1]$ so that the model in the next iteration is a linear combination of the model we found via split Bregman and the current model, i.e., the "replace" step in Algorithm 1 becomes

$$m_k \leftarrow \alpha m_{k+1} + (1 - \alpha) m_k, \tag{21}$$

where m_{k+1} is chose along with a regularization strength μ to either minimize RMS, or, if the target RMS is reached, along with the largest μ that yields the target RMS (blockiest model).

The remaining question is: *If RamBO does not automatically adjust the regularization strength μ, how should μ be determined*? One way forward is to adopt a hierarchical approach and sample models m and regularization strengths μ jointly from the posterior distribution $p(m, \mu | d)$. This strategy is, for example, used in the RTO-TKO (Blatter et al. 2022a,b), and this technology could be adapted to TV regularized problems. An easier and more efficient way forward is to pick a relatively small value for μ, e.g., we pick $\mu = 0.1$ in the numerical illustrations in Section 5. The reason is that by choosing a small μ, we compute the most uncertain blocky models (another use of Occam's principle). The value $\mu = 0.1$ may not be universal and we recommend first run a blocky Occam (which one may be tempted to do anyways) and monitor the range of regularization strength encountered during blocky Occam.

Finally, we note that Wang et al. (2017) explored ℓ_1 regularization in the context of RTO via a clever *invertible* change of variables. The TV regularization we need here for blocky models, however, make the change of variables not invertible and, hence, not applicable (see also (Lee 2021)).

4.1 RamBO and trans-dimensional MCMC

A common approach to UQ in geosciences is trans-dimensional Markov chain Monte Carlo (trans-D MCMC) (see, e.g., Sambridge et al. 2013, 2006; Malinverno 2002). For layered models as we discuss them here, we parameterize the subsurface by the *number* of layers and their thicknesses and then use the trans-D MCMC to determine the resistivity in each layer. Critically, the number of layers is an unknown, but the trans-dimensional approach induces a natural parsimony and favors models with a small number of layers over models with a large number of layers. RamBO as presented here follows very similar principles, because the TV regularization will enforce that the number of "blocks" is as small as possible. Remember, however, that the underlying model parameterization in RamBO has a large number of layers, and if smoothing is required by the data the inversion will allow that.

We expect that RamBO and trans-D MCMC give somewhat similar results when applied to invert the same data set. RamBO, however, has a large computational advantage: We only need relatively few samples and the samples are easy to compute (see examples below). Perhaps more importantly, every inverse problem requires crafting a new and tailored trans-D MCMC code, but RamBO is easy to apply, especially if an Occam-style code is already available. On the downside, RamBO assumes access to the Jacobian, which is not required by trans-D MCMC. The use of the Jacobian, however, implies faster convergence of RamBO and trans-D MCMC codes are notoriously slow to converge.

5 NUMERICAL ILLUSTRATIONS

We illustrate the use of blocky Occam and RamBO on two data sets. The Schlumberger data set, collected at the Wauchope station in central Australia (Constable et al. 1984), was also inverted in the Occam's inversion paper (Constable et al. 1987). A marine magnetotelluric (MT) data set was recently collected off-shore New Jersey (Gustafson et al. 2019) to understand low salinities observed on wells in nearby areas. For the marine MT data set we note that the relatively shallow waters in the region (20m-100m) were insufficient to attenuate high frequencies ($1 - 100$ Hz), and this allowed to to resolve upper subsurface structures. Following Blatter et al. (2021), we consider station N05 and invert for 1D resistivity models.

We first perform blocky Occam inversions in Section 5.1 and compare the blocky models to smooth models obtained via classical Occam's inversion. In Section 5.2, we compute uncertainties using RamBO, and we compare our results to the trans-D MCMC inversions of Malinverno (2002) and Blatter et al. (2019). In our inversions and UQ, we use the standard deviations reported as part of the Schlumberger and marine MT data sets to construct the weighting matrix W in the cost functions (1) and (18). The model Jacobians are computed via finite-differences, but a more careful implementation should use adjoints or automatic differentiation to reduce the number of required forward model evaluations – we use finite differences here to keep the code clean and because the 1D forward models are computationally inexpensive.

5.1 Blocky Occam inversions

We apply blocky Occam to invert the Schlumberger and marine MT data sets and compare the results to Occam's inversions that generate smooth models. All inversions start with a half-space model and both inversion algorithms are given a range of regularization strengths and a target RMS (which we set to one). The inversion algorithms stop the iteration if the RMS is below the target RMS *or* if RMS does not change much from iteration to the next, i.e., if

$$|\text{RMS}_{j+1} - \text{RMS}_j| \leq \text{tol}_{\text{RMS}}. \tag{22}$$

If (22) is satisfied, we say the the iteration "converged." For the results shown below, we chose the tolerance $\text{tol}_{\text{RMS}} = 10^{-4}$.

For the Schlumberger data set, Occam's inversion and blocky Occam converge in 3-4 iterations and lead to a nearly identical RMS of 1.077 (blocky Occam) and 1.080 (Occam's inversion) (** in the figure it is 0.99 and 0.98 **). The resistivity models obtained by blocky Occam and Occam's inversion are shown in Figure 2(a) and the fits to the data are shown in the supple-

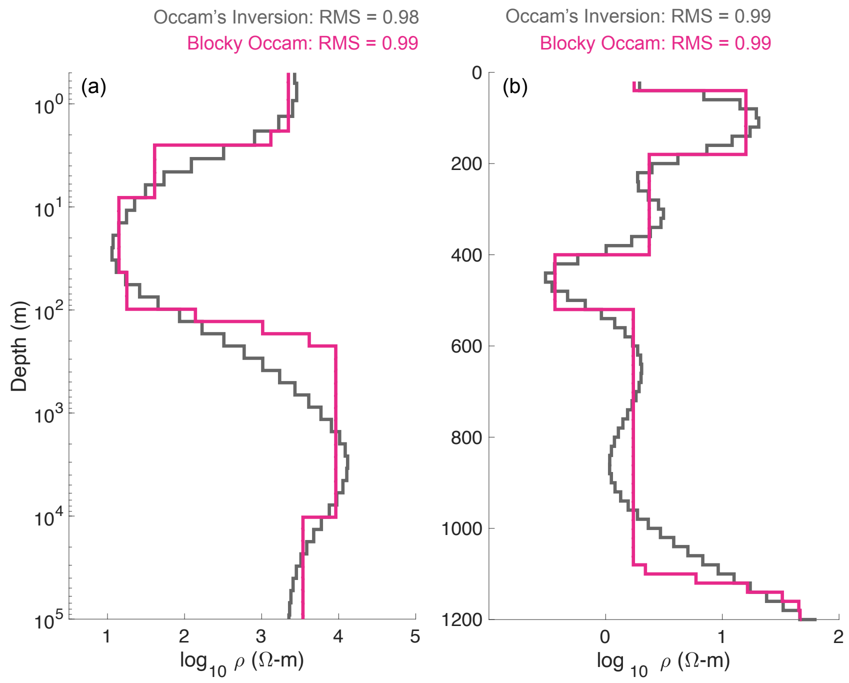


Figure 2. Blocky Occam compared to Occam's inversion. Shown are the resistivities as a function of depth for (a) the Schlumberger data set and (b) the marine MT data set. Blocky Occam (pink) and Occam's inversion (grey) lead to nearly identical RMS and the blocky Occam solution looks like a blocky version of the Occam's inversion as desired and as expected.

mentary Figure A1(a) in Appendix C (because the data fits are obviously very good given the small RMS). As expected and as desired, the blocky Occam models looks like blocky versions of the smooth models obtained via Occam's inversion. Occam and blocky Occan models can be obtained at roughly the same computational cost and exhibit a very similar fit to the data (see Figure A1(a)). More specifically, we find that Occam's inversion reveals two main features: a conductive zone beneath a 2 m dry surface layer and a deeper resistive zone. Because Occam's inversion generates the smoothest model that fits the data, the transition between the conductive and resistive zones is blurry and not well-defined. In comparison, blocky Occam provides a more distinct separation between the conductive and resistive layers, particularly the base of the conductive layer at approximately 200 m depth.

For the marine MT data set, both inversion algorithms require more iterations: blocky Occam requires 13 iterations and Occam's inversion requires 19 iterations to reach the target RMS. Both inversions lead to a nearly identical RMS (0.986 for blocky Occam and 0.987 for Occam's inversion). The resistivity models are illustrated in Figure 2(b) and the data fits can be found in supplementary Figures A1(b)-(c). The smooth model obtained via Occam's inversion shows two distinct peaks that correspond to resistive and conductive features. The resistive zone between 40 m to 160 m is associated with sediments hosting low salinity water. The conductive feature at about 400 m suggests sediments hosting seawater. The smooth model shows oscillations around 300 m, where the transition between low and high resistivity zones occurs. These oscillations appear because the smooth inversion is really blocky or in other words, we have sharp changes in resistivity, and for that reason, we observe Gibbs-type phenomenon in the transitions in smooth models. In contrast, the blocky Occam model defines a simpler boundary between the high and low resistivity zones.

Finally, we note that blocky Occam applies the split Bregman iteration within the linearizing "outer loop" of an Occam's inversion. The overall computational cost of blocky Occam thus depends on how fast split Bregman converges. Here, convergence of split Bregman is assessed by equation (17) and we chose a small tolerance to obtain very blocky models. The split Bregman iteration converges faster if we use a larger tolerance, but then the resulting models are not really blocky. With our choices, split Bregman converges on average within 181 iterations for the Schlumberger data set and within 154 iterations for the marine MT data set. We acknowledge that the number of iterations is quite large, which may result in high computational costs in 2D or 3D problems for which the linear algebra of solving least squares problems is more involved than in our 1D test cases (step (i) of split Bregman, see Section 2.4). Our experiments with 1D electromagnetic data thus suggest that the large number of iterations in split Bregman generates a computational overhead compared to Occam's inversion, but this overhead is needed to obtain truly blocky models. We are unaware of numerical techniques that are more efficient than split Bregman. All other ideas we tried, including approximating ℓ_1 norms via Eckblom norms or Huber losses, interior point methods for ℓ_1 convex optimization (see, e.g., Nocedal & Wright 2006), or trans-dimensional MCMC, were computationally more expensive, led to smoother models, or both. The search for blocky models may always be computationally more expensive than searching for smooth models: the TV-regularized inverse problem (18) is inherently more difficult to solve than the nonlinear least squares problem in (1).

5.2 UQ with RamBO

We now apply RamBO to the Schlumberger and marine MT data sets to compute an uncertainty quantification. RamBO amounts to running blocky Occam, with a fixed regularization strength $\mu = 0.1$ in a parallel for-loop. We note that we obtain very similar results with similar μ, but if we choose μ to large (e.g., $\mu = 2$), then the uncertainty bounds are very narrow due to the large influence of the Laplacian prior. If μ is too small (e.g, $\mu = 0.01$), the optimization is unstable. In general, one should adjust μ for blocky Occam to be as *small* as possible to compute the largest possible uncertainty. A range of possible regularization strength values is often apparent after inspecting the results of a blocky Occam or Occam's inversion.

Since the 1D inversions are inexpensive, and since competing trans-D MCMC codes usually require a very large number of forward model evaluations, we draw a large number of samples (10^4). For both data-sets, the optimization of RamBO occasionally "crashes," and leads to a large RMS > 3 or NaNs. We filter out these failed attempts and are then left with 9202 samples for the Schlumberger data set and 9639 samples for the marine MT data set. We use these samples in Figure 3 to create histograms of resistivity (log-scale) as a function of depth, similar to Figure 12 in Malinverno (2002) and Figure 10(b) in Blatter et al. (2019).

For the Schlumberger data set (Figure 3(a)), we find an uncertain but resistive surface layer to a depth of 2 m, followed by three similarly conductive layers (3.5 m–10 m, 10 m–30 m and 30 m–100 m). Between 170 m and 4500 m, we detect a resistive layer and beneath this the uncertainty becomes large. These results are in good agreement with the trans-D MCMC results reported by Malinverno (2002) and, to a lesser extent, also with the results of Blatter et al. (2022a), which uses a quadratic regularization. In addition, we note that uncertainty is *not* symmetric about the blocky Occam model (pink line in Figure 3(a)). This is to be expected because the blocky Occam model is an extreme model – the blockiest model that fits the data.

For the marine MT data set, RamBO defines a resistive layer (40 m–200 m) and a conductive layer (400 m–500 m). Below 500 m, the uncertainty is rather large, which is in good agreement with the trans-D results reported by Blatter et al. (2019). Again, the uncertainty is not symmetric around the blocky Occam solution (as expected). The blocky Occam solution rather picks out the least resistive model that is rendered likely by RamBO when the data are very informative

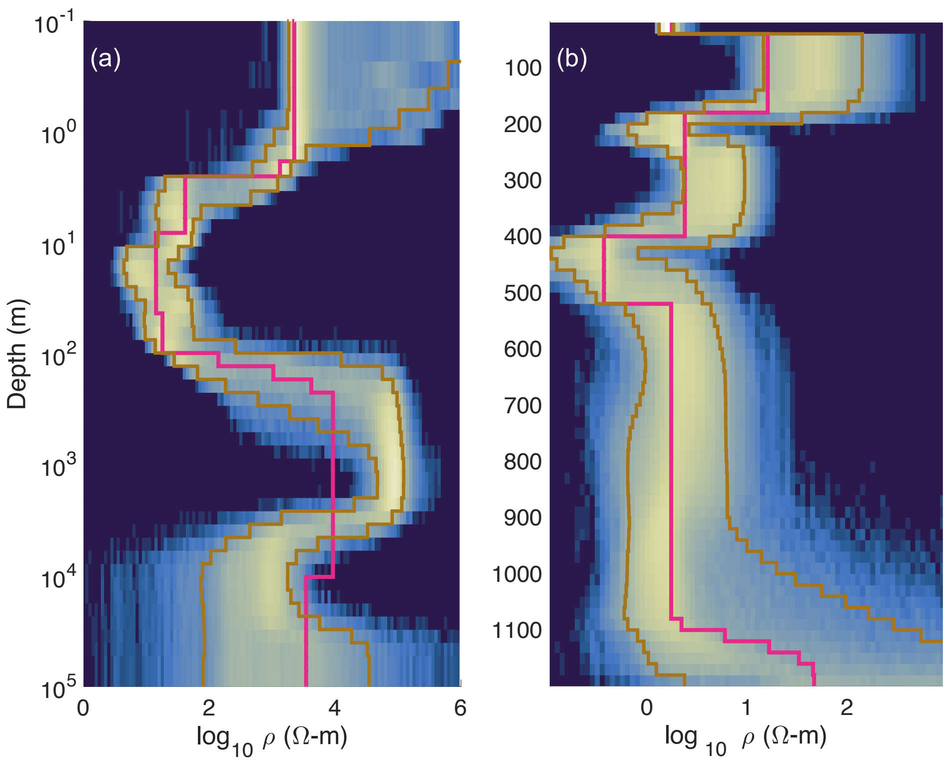


Figure 3. Uncertainty quantification for the Schlumberger data set (a) and marine MT data set (b). Shown are histograms of resistivity (log-scale) as a function of depth. Warmer colors (green and yellow) indicate higher probability and cool colors (blues) indicate low or no probability (dark blue). The brown lines indicate 5% and 95% quartiles and the pink lines correspond to the blocky Occam results described above.

(above 400 m), which makes sense since MT is more sensitive to thin conductors than thin resistors (Key et al. (2006)).

The data fits of models generated by RamBO for the Schlumberger and marine MT data sets are shown in Figures A2(a,c,d) in Appendix C. Histograms of RMS of models generated by RamBO are shown in Figures A2(b.e). RamBO explores many models that fit the data well and the distribution of RMS is near one for both data sets. We note "spikes" in the histograms near the target RMS, which are caused by the use of the target RMS as a stopping criteria for the iteration.

In summary, RamBO generates a UQ that is comparable to what other methods have produced. Compared to trans-D MCMC, however, RamBO has two advantages:

(i) The UQ can be computed at a reduced computational cost.

(ii) RamBO relies on optimization and can be implemented with only minor modifications

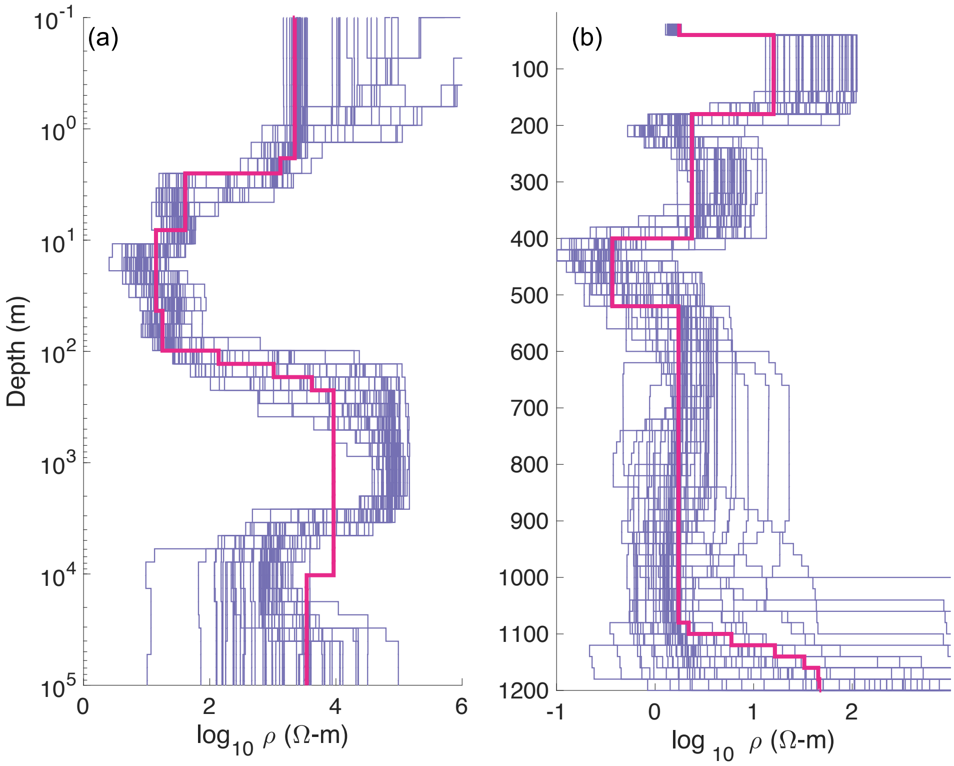


Figure 4. Spaghetti plots of 50 samples obtained by RamBO (purple) for the Schlumberger data set (a) and the marine MT data set (b). Shown in pink is the blocky Occam model

of an existing Occam's inversion code. Trans-D MCMC, on the other hand, is usually tailor-made for each problem and trans-D MCMC codes are not easily portable from one inversion to another.

The computational advantage of RamBO compared to trans-D MCMC is more apparent if we constrain the number of samples. With RamBO, about 50 models may be sufficient to get an idea of the uncertainty of the inversion. We illustrate this idea in Figure 4, where we show a "spaghetti plot" of 50 samples of RamBO. The 50 samples are sufficient to eyeball regions of large or small uncertainty and the 5% and 95% quartiles are already comparable to those obtained from $O(10^4)$ samples. RamBO inherits the computational efficiency for UQ from RTO, which was already reported and discussed at length in the context of inverting EM data by Blatter et al. (2022a,b). MCMC in general and trans-D MCMC in particular, routinely require thousands or millions of samples due to slow convergence (and the convergence becomes slower with dimension/the number of layers). RamBO may therefore be viewed as a computationally efficient and more robustly applicable alternative to trans-D MCMC.

6 SUMMARY AND CONCLUSIONS

Although blocky models are useful for representing abrupt changes in resistivity they have not found their way into the mainstream in EM applications. Methods like trans-D MCMC, Huber, and Ekblom norms have explored the blocky ideas over the years, but they are still not widely used because they are computationally expensive, unnecessarily complicated, or both.

In this work, we use TV regularization to force blocky models. We extend Occam's inversion to include TV regularization and use split Bregman for very efficient solutions. We call this blocky Occam. Then, we equip this algorithm with an efficient uncertainty quantification (UQ) method via a modified RTO approach which we call RamBO. The implementation of blocky Occam is remarkably simple once you have an Occam's code. With just one line change, you can incorporate Split Bregman, which is also easy to implement. RamBO is just as simple—once you have blocky Occam, you can run it with a parallel for loop.

Like the classical Occam code, blocky Occam and RamBO require minimum tuning. We illustrate the use of blocky Occam using 1D DC resistivity data and a marine MT data set. For both data sets, our blocky models display the same structures found using classical Occam's inversion but with sharper transitions and clearer distinctions between resistivity contrasts. A UQ generated by RamBO is comparable to one obtained by trans-dimensional methods, but RamBO is easier to implement and requires less computational cost.

As explained in the introduction, we are motivated by the desire to interpret electromagnetic data, but inversion algorithms know nothing of the physics in the forward problem, and our code has already been adopted by our seismic colleagues. As for the original Occam algorithm, we and others expect to apply it to 2D and perhaps 3D problems. However, 1D solutions are still useful in some aspects of geophysics. For example, the hugely popular SkyTEM system (Sorensen & Auken 2004) uses hundreds of stiched 1D inversions as an interpretation product, and might benefit from the combination of better depth resolution and minimal tuning of blocky Occam.

7 DATA AVAILABILITY STATEMENT

The data and code used in this paper are available on github and on Zenodo.

8 ACKNOWLEDGMENTS

EVH is supported by ONR grant N00014-21-1-2309 and ... MM is supported by ONR grant N00014-21-1-2309. SCC is supported by nobody at all.

8.1 Author Contribution

EVH and MM equally contributed to the project's conception and execution. SC took the lead on interpreting the results. EVH took the lead on writing the code. All three authors wrote the paper together.

APPENDIX A: PARAMETERS USED TO CREATE FIGURE 1

A total of 50 MT amplitudes and phases logarithmically spaced between 100 Hz and 100,000 s were computed for a simple one dimensional model of a 300 m thick 1 Ωm layer underlain by a 50 Ωm half-space and also a model replacing the step with a sigmoid function centered on 500 m depth. The data were perturbed with normally distributed noise and inverted using a standard Occam approach. The inverted model consisted of 100 layers increasing exponentially in thickness from 1 m to 1,000 km. Regularization was a first difference between each layer, unweighted by layer thickness or depth.

Noise was set to 0.3%, 1%, 3%, and 10% of linear apparent resistivity and propagated into $\log_{10}$(apparent resistivity) and linear phase, which were the inverted data. For each noise level 20 inversions were carried out to capture variations associated with the noise statistics, all converging to a root-mean-square misfit of 1.0.

APPENDIX B: SPLIT BREGMAN WITH OR WITHOUT PERTURBATIONS

We wish to minimize the cost function

$$\mathcal{C}(x) = \|Jm - d\|^2 + \mu \, |Dm + \nu| \, , \tag{B.1}$$

with split Bregman. The auxiliary and Bregman variables are as before and the optimization problem becomes:

$$\mathcal{C}_{\text{Breg}}(m, u) = \|Jm - d\|^2 + \mu|u + \nu| + \gamma\|u - Dm - b\|^2. \tag{B.2}$$

The reformulated optimization problem is solved by iterating the following three steps.

(i) For a given u_k and b_k, minimize $\mathcal{C}_{\text{Breg}}$ over m by solving the least squares problem

$$m_{k+1} = \arg\min_m \|Jm - d\|^2 + \gamma\|u_k - Dm - b_k\|^2 \tag{B.3}$$

(ii) Given b_k and m_{k+1}, minimize $\mathcal{C}_{\text{Breg}}$ over u by solving the optimization problem

$$u_{k+1} = \arg\min_u \mu|u + \nu| + \gamma\|u - Dm_{k+1} - b_k\|^2. \tag{B.4}$$

via soft-thresholding:

$$u_{k+1} = \text{ST}(\nu + Dm_{k+1} + b_k; 2\mu/\gamma), \tag{B.5}$$

(iii) Update the Bregman variable

$$b_{k+1} = b_k + (Dm_{k+1} - u_{k+1}). \tag{B.6}$$

We summarize split Bregman with perturbations ν in Algorithm 3, where we set the Lagrange multiplyer $\gamma = 2\mu$, as recommended by Goldstein & Osher (2009). The algorithm for split Bregman *without* perturbations, as used in the blocky Occam of Section 3, can be obtained by setting $\nu = 0$.

APPENDIX C: ADDITIONAL FIGURES

Algorithm 3 Split Bregman

while $k \leq k_{\text{max}}$ **do**

Solve the least squares problem

$$m_{k+1} = \arg\min_{m} \|Jm_k - d\|^2 + \gamma \|u_k - Dm_k - b_k\|^2$$

Use soft-thresholding to find u_{k+1}

$$u_{k+1} = \text{ST}(\nu + Dm_{k+1} + b_k; 1),$$

Update the Bregman variable

$$b_{k+1} = b_k + (Dm_{k+1} - u_{k+1}).$$

if convergence **then**

Exit

else

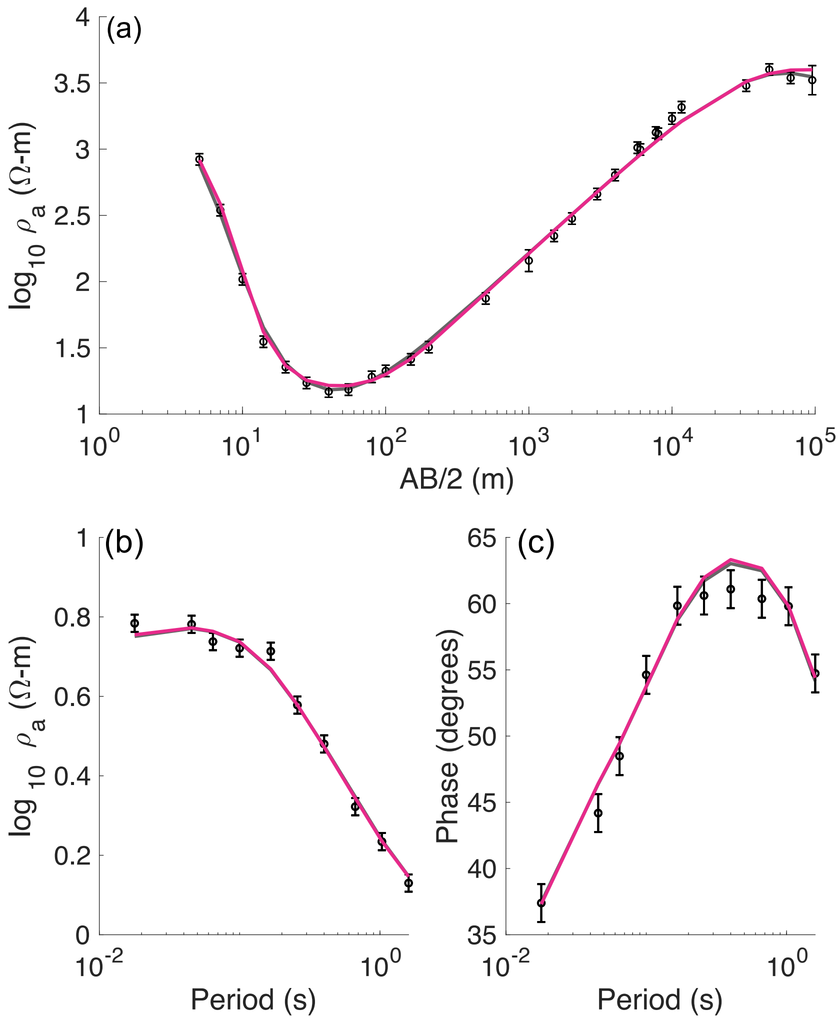
$m_k \leftarrow m_{k+1}$

$u_k \leftarrow u_{k+1}$

$b_k \leftarrow b_{k+1}$

end if

end while



[h!]

Figure A1. Blocky Occam compared to Occam's inversion. Panel (a) shows apparent resistivity (logspace) as a function of electrode spacing (AB/2) for the Schlumberger data set, along with error bars and the data fits of blocky Occam (pink) and Occam's inversion (grey, partially hidden). Panels (b) and (c) show apparent resistivity (logspace) and phase as a function of period, along with error bars. The data fits for blocky Occam and Occam's inversion are shown in pink and grey. The result of Occam's inversion is partially hidden by the result of blocky Occam.

[h!]

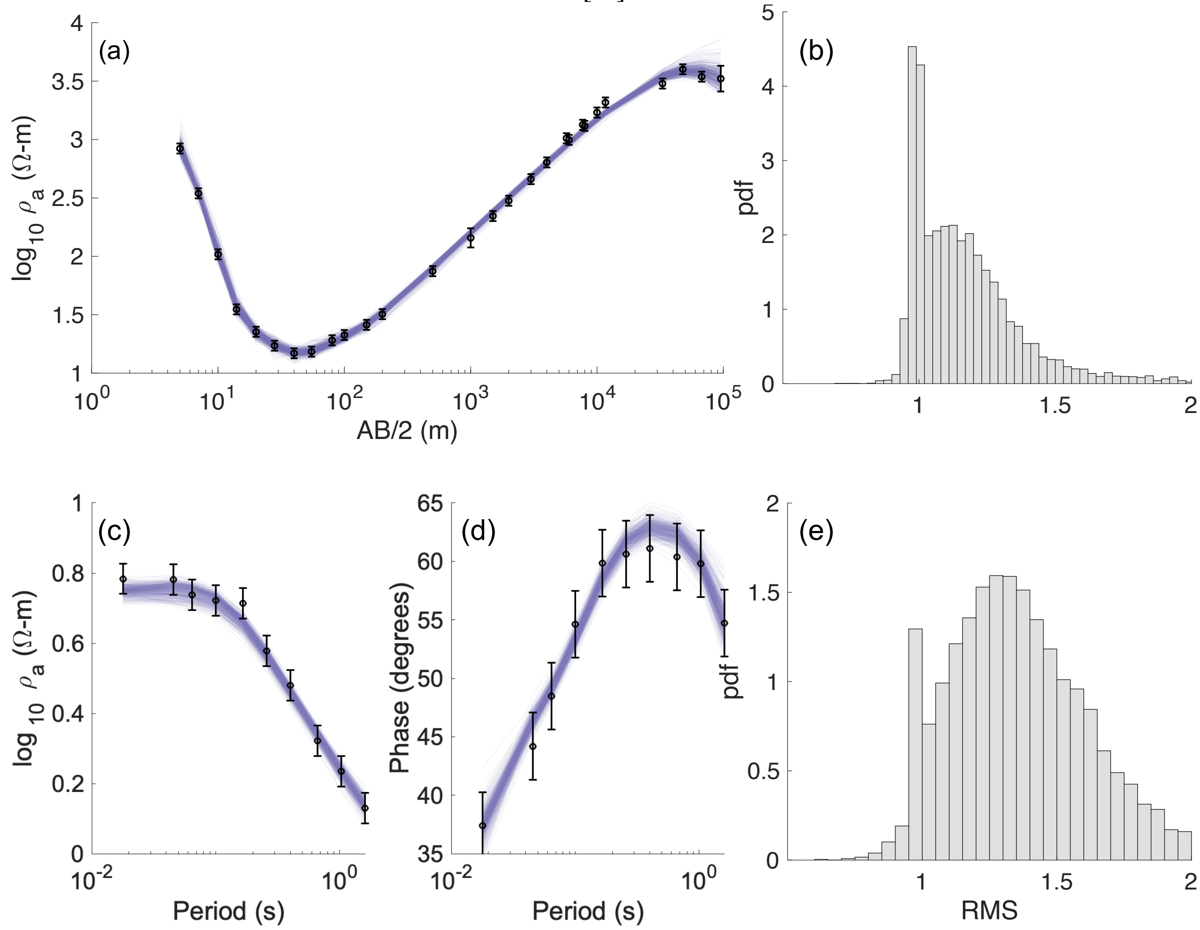


Figure A2. (a) Data fits of 500 models generated by RamBO for the Schlumberger data set. (b) Histogram of RMS corresponding to the models generated by RamBO (Schlumberger). (c,d) Data fits of 500 models generated by RamBO for the marine MT data set. (e) Histogram of RMS corresponding to the models generated by RamBO (marine MT).

REFERENCES

Bardsley, J. M., Solonen, A., Haario, H., & Laine, M., 2014. Randomize-then-optimize: A method for sampling from posterior distributions in nonlinear inverse problems, *SIAM Journal on Scientific Computing*, **36**(4), A1895–A1910.

Blatter, D., Key, K., Ray, A., Gustafson, C., & Evans, R., 2019. Bayesian joint inversion of controlled source electromagnetic and magnetotelluric data to image freshwater aquifer offshore New Jersey, *Geophys. J. Int.*, **218**(3), 1822–1837.

Blatter, D., Ray, A., & Key, K., 2021. Two-dimensional Bayesian inversion of magnetotelluric data using trans-dimensional Gaussian processes, *Geophys. J. Int.*, **226**(1), 548–563.

Blatter, D., Morzfeld, M., Key, K., & Constable, S., 2022a. Uncertainty quantification for regularized inversion of electromagnetic geophysical data – Part I: motivation and theory, *Geophys. J. Int.*, **231(2)**, 1057–1074.

Blatter, D., Morzfeld, M., Key, K., & Constable, S., 2022b. Uncertainty quantification for regularized inversion of electromagnetic geophysical data – Part II: application in 1-D and 2-D problems, *Geophys. J. Int.*, **231(2)**, 1075–1095.

Chen, Y. & Oliver, D., 2012. Ensemble randomized maximum likelihood method as an iterative ensemble smoother, *Math Geosci.*, **44**, 1–26.

Constable, S., Parker, R., & Constable, C., 1987. Occam's inversion: A practical algorithm for generating smooth models from electromagnetic sounding data, *Geophysics*, **52**(3), 289–300.

Constable, S. C., McElhinny, M. W., & McFadden, P. L., 1984. Deep Schlumberger sounding and the crustal resistivity structure of central Australia, *Geophys. J. Int.*, **79**(3), 893–910.

DeGroot-Hedlin, C. & Constable, S., 1990. Occam inversion to generate smooth, 2-dimensional models from magnetotelluric data, *Geophysics*, **55**(12), 1613–1624.

Farquharson, C. G. & Oldenburg, D. W., 1998. Non-linear inversion using general measures of data misfit and model structure, *Geophys. J. Int.*, **134**(1), 213–227.

Fournier, D. & Oldenburg, D. W., 2019. Inversion using spatially variable mixed lp norms, *Geophys. J. Int.*, **218**(1), 268–282.

Gibbs, J., 1899. Fourier's series, *Nature*, **9**, 606.

Goldstein, T. & Osher, S., 2009. The split Bregman method for L1-regularized problems, *SIAM journal on imaging sciences*, **2**(2), 323–343.

Guitton, A. & Symes, W. W., 2003. Robust inversion of seismic data using the Huber norm, *Geophysics*, **68**(4), 1310–1319.

Gustafson, C., Key, K., & Evans, R. L., 2019. Aquifer systems extending far offshore on the U.S. Atlantic margin, *Scientific Reports*, **9**(1), 1–10.

Inman, J., Ryu, J., & Ward, S., 1973. Resistivity inversion, *Geophysics*, **38**, 1088–1108.

Isaksen, L., Bonavita, M., Buizza, R., Fisher, M., Haseler, J., Leutbecher, M., & Raynaud, L., 2010.

Ensemble of data assimilations at ECMWF.

Key, K., 2016. MARE2DEM: a 2-D inversion code for controlled-source electromagnetic and magnetotelluric data, *Geophys. J. Int.*, **207**(1), 571 – 588.

Key, K. W., Constable, S. C., & Weiss, C. J., 2006. Mapping 3d salt using the 2d marine magnetotelluric method: Case study from gemini prospect, gulf of mexico, *Geophysics*, **71**(1), B17–B27.

Kitanidis, P. K., 1995. Quasi-linear geostatistical theory for inversing, *Water Resour. Res.*, **10**(31), 2411–2419.

Lee, J. & Kitanidis, P. K., 2013. Bayesian inversion with total variation prior for discrete geologic structure identification, *Water Resources Research*, **49**(11), 7658–7669.

Lee, Y., 2021. l_p regularization for ensemble kalman inversion, *SIAM Journal on Scientific Computing*, **43**(5), A3417–A3437.

Malinverno, A., 2002. Parsimonious Bayesian Markov chain Monte Carlo inversion in a nonlinear geophysical problem, *Geophysical Journal International*, **151**(3), 675–688.

Nocedal, J. & Wright, S. J., 2006. *Numerical Optimization*, Springer, New York, NY, USA, 2nd edn.

Oliver, D. S., He, N., & Reynolds, A. C., 1996. Conditioning permeability fields to pressure data.

Parker, R. L., 1970. The inverse problem of electrical conductivity in the mantle, *Geophysical Journal of the Royal Astronomcial Society*, **22**, 121–138.

Parker, R. L., 1994. *Geophysical Inverse Theory*, Princeton, New Jersey.

Portniaguine, O. & Zhdanov, M. S., 1999. Focusing geophysical inversion images, *GEOPHYSICS*, **64**(3), 874–887.

Rudin, L. I., Osher, S., & Fatemi, E., 1992. Nonlinear total variation based noise removal algorithms, *Physica D: Nonlinear Phenomena*, **60**(1), 259–268.

Sambridge, M., Gallagher, K., Jackson, A., & Rickwood, P., 2006. Trans-dimensional inverse problems, model comparison and the evidence, *Geophys. J. Int.*, **167**(2), 528–542.

Sambridge, M., Bodin, T., Gallagher, K., & Tkalčić, H., 2013. Transdimensional inference in the geosciences, *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, **371**(1984), 20110547.

Siripunvaraporn, W. & Sarakorn, W., 2011. An efficient data-space conjugate gradient Occam's method for three-dimensional magnetotelluric inversion, *Geophys. J. Int.*, **186**(2), 567–579.

Sorensen, K. I. & Auken, E., 2004. Skytem-a new high-resolution helicopter transient electromagnetic system, *Exploration Geophysics*, **35**(3), 194–202.

Sun, J. & Li, Y., 2014. Adaptive Lp inversion for simultaneous recovery of both blocky and smooth features in a geophysical model, *Geophys. J. Int.*, **197**(2), 882–899.

Tang, W., Li, J., Zhang, W., Zhang, J., Geng, W., & Li, Y., 2021. Time-lapse difference inversion based on the modified reflectivity method with differentiable hyper-Laplacian blocky constraint, *Geophysics*, **86**(6), R865–R878.

Theune, U., Jensas, I. O., & Eidsvik, J., 2010. Analysis of prior models for a blocky inversion of seismic

AVA data, *Geophysics*, **75**(3), C25–C35.

Wang, Z., Bardsley, J. M., Solonen, A., Cui, T., & Marzouk, Y. M., 2017. Bayesian inverse problems with l1 priors: a randomize-then-optimize approach, *SIAM Journal on Scientific Computing*, **39**(5), S140–S166.

Wei, X. & Sun, J., 2021. *3D probabilistic geology differentiation using mixed Lp norm joint inversion constrained by petrophysical information*, pp. 1231–1235.