Stochastic electrical resistivity tomography with ensemble smoother and deep convolutional autoencoders

Mattia Aleardi°, Alessandro Vinciguerra*,°, Eusebio Stucchi°, Azadeh Hojat+,§

°University of Pisa, Earth Sciences Department, via S. Maria 53, 56126, Pisa, Italy

*University of Florence, Earth Sciences Department, via. G. La Pira 4, 50121, Florence, Italy

+Shahid Bahonar University of Kerman, Department of Mining Engineering, Jomhouri Blvd., Kerman, 76188, Iran

§Politecnico di Milano, Department of Civil and Environmental Engineering, Piazza Leonardo da Vinci 32, 20133, Milano Italy

Corresponding author: Mattia Aleardi, mattia.aleardi@unipi.it

ABSTRACT

To reduce both the computational cost of probabilistic inversions and the ill-posedness of geophysical problems, model and data spaces can be re-parameterized into low-dimensional domains where the inverse solution can be computed more efficiently. Among the many compression methods, deep learning algorithms based on deep generative models provide an efficient approach for model and data space reduction. We present a probabilistic electrical resistivity tomography inversion in which the data and model spaces are compressed through deep convolutional variational autoencoders, while the optimization procedure is driven by the ensemble smoother with multiple data assimilation, an iterative ensemble-based algorithm. This method iteratively updates an initial ensemble of models that are generated according to a previously defined prior model. The inversion outcome consists of the most likely solution and a set of realizations of the variables of interest from which the posterior uncertainties can be numerically evaluated. We test the method on synthetic data computed over a schematic subsurface model, and then we apply the inversion to field measurements. The model predictions and the uncertainty assessments provided by the presented approach are also compared with the results of an MCMC sampling working in the compressed domains, a gradient-based algorithm, and with the outcomes of an ensemble-based inversion running in the uncompressed spaces. A finite-element code constitutes the forward operator. Our experiments show that the implemented inversion provides most likely solutions and uncertainty quantifications comparable to those yielded by the ensemble-based inversion running in the full model and data spaces, and the MCMC sampling, but with a significant reduction of the computational cost.

Keywords: Electrical Resistivity Tomography; Inversion;

INTRODUCTION

Electrical resistivity tomography (ERT) is widely used to image the resistivity distribution of the subsurface in a variety of engineering, hydrogeological and environmental problems (e.g., Rucker et al. 2011; Moradipour et al. 2016; Whiteley et al. 2017; Arosio et al. 2017; Bièvre et al. 2018; Hojat et al. 2019a; Dahlin 2020; Hermans and Paepen 2020; Aleardi et al. 2020; Loke et al. 2020; Aleardi et al. 2021a; Norooz et al. 2021). Due to incomplete data coverage and noise contamination, the ERT is an ill-posed problem characterized by a non-unique and unstable solution (i.e., small variations of the data produce large perturbations in the predictions; Tarantola, 2005; Aster et al. 2018; Binley and Slater, 2020), and hence, an accurate estimation of the model uncertainty is of primary importance. However, the most common approach to ERT solves the inversion through deterministic, gradient-based algorithms. These methods employ optimization algorithms to minimize a predefined objective function that measures the difference between the predicted and the observed data. Usually, model constraints are also infused in the objective function to reduce the ill-conditioning of the problem. Such methods are generally computationally efficient but provide an estimation of the model (i.e., the most likely solution) without accurately quantifying the associated uncertainty. On the contrary, a probabilistic (Bayesian) inversion framework considers the model parameters as random variables and formulate the inversion as a probability density function that is proportional to the product of the prior and the data likelihood. The prior term corresponds to the regularization term in deterministic methods, whereas the likelihood incorporates information about the observed data. For linear forward operators and Gaussian model and data assumptions, the posterior can be analytically computed from which model realizations can be efficiently simulated. Otherwise, Markov Chain Monte Carlo (MCMC; Sambridge and Mosegaard, 2002; Sen and Stoffa, 2013) algorithms can be employed for accurate posterior probability density (PPD) estimations in non-linear problems. However, the considerable numbers of samples needed for accurate uncertainty appraisals often discouraged their applications in large dimensional parameter spaces and for expensive forward model evaluations (Sajeva et al. 2014; Aleardi and Salusti, 2020; Pradhan and Mukerji, 2020). To mitigate this problem, model and data compression strategies can be employed such as singular-value decomposition, wavelet transform, discrete cosine transform (Grana et al. 2019; Aleardi, 2020) and in this context, the inversion is run in the reduced model and data spaces. Another promising approach is based on the dimension reduction of model and data spaces via deep neural networks (Goodfellow et al., 2014; Laloy et al., 2018) that presents several advantages over linear compression strategies. Ensemble-based data assimilation methods such as ensemble smoother with multiple data assimilation (ES-MDA) (Emerick and Reynolds, 2013) can constitute an efficient alternative to MCMC algorithms because they are computationally faster but might underestimate the model uncertainty in high-dimensional parameter and data spaces. This undesirable phenomenon is usually called ensemble collapse (Sætrom and Omre, 2013). To mitigate this issue a local analysis can be employed to eliminate spurious correlations between data and model parameters (Chen and Oliver, 2017; Luo et al., 2019). Otherwise, reduction methods can be employed to eliminate the redundant information (Luo et al., 2018). Therefore, compression strategies have also been extensively implemented in ensemble-based methods (Bao et al. 2020). In this context, the compression of model and data space allows developing a fast and efficient probabilistic inversion. However, the unavoidable information loss due to reduction might lead to underestimation or overestimation of the model uncertainty (Grana et al. 2019). For this reason, the trade-off between model resolution and model uncertainty must be always considered when reparameterization techniques are applied (Aleardi, 2015). Recently ensemble-based methods and convolutional autoencoders have extensively been used to solve geophysical problems and some applications can be found in Liu and Grana (2018), Mandelli et al. (2018), Kang et al. (2019), Tso et al. (2020), Saad and Chen (2020), Gao et al. (2020), Kang et al. (2021), to name just a few.

In this work, we present a probabilistic ERT inversion in which deep convolutional variational autoencoders (DCVAEs; Kingma and Welling, 2013) are used to compress data and model spaces, while the ES-MDA provides multiple posterior realizations from which the uncertainty can be numerically assessed. DCVAEs are a variant of variational autoencoders (VAEs) in which convolutional filters are used to extract latent features from the network input. We first discuss a synthetic example over a schematic subsurface model before applying the method to field data. The outcomes of the proposed approach are also benchmarked against those yielded by a gradient-based algorithm, an ES-MDA inversion running in the full data and model spaces, and an MCMC sampling working in the compressed domains. The employed MCMC recipe is described in Vinciguerra et al. (2021) with the only difference that the probabilistic sampling is here performed in DCVAE compressed data and model spaces. The MCMC method employed is the differential evolution Markov chain, a popular algorithm that employs interactive chains to improve the efficiency of probabilistic sampling (Vrugt, 2016). In all cases, a 2.5D finite-elements (FE) Matlab modeling routine constitutes the forward operator (Karaoulis et al., 2013). All the codes have been written in Matlab, and all the tests have been run on a notebook equipped with Intel i7-10750H CPU@2.60GHz, 16Gb of RAM, and an NVIDIA GeForce RTX 2060.

This work aims to assess the applicability of DCVAEs to increase the computational efficiency of a probabilistic ERT inversion solved via the ES-MDA algorithm. As far as the authors are aware, this is the first paper in which these two approaches are combined to solve this geophysical problem.

METHODS

The Bayesian framework and the ensemble-based inversion

In a Bayesian context the solution of an inverse problem is fully expressed by the PPD in the model space, which is expressed as:

$$p(\mathbf{m}|\mathbf{d}) = \frac{p(\mathbf{d}|\mathbf{m})p(\mathbf{m})}{p(\mathbf{d})}, \quad (1)$$

where $p(\mathbf{m})$ and $p(\mathbf{d})$ denote the a-priori distributions of model parameters and data, respectively; $p(\mathbf{m}|\mathbf{d})$ is the target PPD, whereas $p(\mathbf{d}|\mathbf{m})$ is the data likelihood. For nonlinear inverse problems, the posterior distribution can not be analytically computed because the forward operator can not be expressed in a matrix form. Therefore, a numerical evaluation of the posterior must be derived using, for example, MCMC sampling algorithms or ensemble-based methods.

The ES-MDA is an iterative procedure in which the updated models are used as the prior in the next iteration. The method starts with an ensemble of models generated according to the prior assumptions. Then, these models are updated by applying a Bayesian updating step to a stochastic observation of the data $\tilde{\mathbf{d}}_k$ under model and data Gaussian assumptions with empirical parameters estimated from the ensemble members. A single ES-MDA iteration can be written as:

$$\mathbf{m}_k^u = \mathbf{m}_k^p + \tilde{\mathbf{C}}_{\mathbf{md}}^p(\tilde{\mathbf{C}}_{\mathbf{dd}}^p + \mathbf{C}_{\mathbf{d}})^{-1}(\tilde{\mathbf{d}}_k - \mathbf{d}_k^p), \quad (2)$$

where:

$$\tilde{\mathbf{C}}_{\mathbf{md}}^p = \frac{1}{N-1}\sum_{j=1}^{N}(\mathbf{m}_k^p - \bar{\mathbf{m}}^p)(\mathbf{d}_k^p - \bar{\mathbf{d}}^p)^T, \quad (3)$$

$$\tilde{\mathbf{C}}_{\mathbf{dd}}^p = \frac{1}{N-1}\sum_{j=1}^{N}(\mathbf{d}_k^p - \bar{\mathbf{d}}^p)(\mathbf{d}_k^p - \bar{\mathbf{d}}^p)^T, \quad (4)$$

with k=1,…,N, where N represents the number of models in the ensemble and $\tilde{\mathbf{d}}_k$ is a random perturbation of the observed data according to the Gaussian distribution $\mathcal{N}(\mathbf{d}, \mathbf{C}_{\mathbf{d}})$, in which $\mathbf{C}_{\mathbf{d}}$ is the data covariance. The subscripts u and p denote the updated (current iteration) and prior (previous iteration) variables, respectively; $\tilde{\mathbf{C}}_{\mathbf{md}}^{p}$ and $\tilde{\mathbf{C}}_{\mathbf{dd}}^{p}$ represent the empirical covariance matrices estimated from the ensemble members, whereas $\bar{\mathbf{m}}^{p}$ and $\bar{\mathbf{d}}^{p}$ are the empirical ensemble mean of the model parameters and predicted data, respectively.

The following steps are implemented for the ES-MDA:

1. Define the number of models in the ensemble *N*, the maximum number of iterations *Q*, and the inflation coefficient α for each iteration with $\sum_{i=1}^{Q} \frac{1}{\alpha_i} = 1$;
2. Generate realizations according to the prior $p(\mathbf{m})$;
3. For each iteration:
    a. Apply the forward operator and compute the observation for each ensemble member $\{\mathbf{d}^{p}\}_{1,\dots,N}$;
    b. Perturb the observations according to: $\tilde{\mathbf{d}}_k = \mathbf{d} + \sqrt{\alpha_i}\mathbf{C}_{\mathbf{d}}^{-1/2}\mathbf{n}$, with $\mathbf{n} = \mathcal{N}(0, \mathbf{I})$, where $\mathbf{I}$ is the identity matrix;
    c. Update the ensemble using equations 2-4 with $\mathbf{C}_{\mathbf{d}}$ replaced by $\alpha_i \mathbf{C}_{\mathbf{d}}$.

All the ensemble members at the last iteration represent possible subsurface scenarios in agreement with the acquired geophysical data and with the prior assumptions. From this ensemble of models, the PPD can be numerically evaluated. Theoretically, the method converges when the ensemble size *N* tends to infinity. In practical applications, a sensitivity analysis is generally required to determine the optimal number of ensemble members that guarantees accurate posterior uncertainty assessments. In particular, the number of ensemble members should be large enough to get an accurate estimate of the $\mathbf{C}_{\mathbf{dd}}^{p}$ and $\mathbf{C}_{\mathbf{md}}^{p}$ matrices but small enough not to make the forward evaluations computationally impractical. Usually, the number of ensemble members needed to get accurate uncertainty assessments increases with the dimension of the model space.

Variational autoencoders

Autoencoders are a class of unsupervised neural networks that are widely employed for representation learning (Goodfellow et al., 2016). Autoencoders are more powerful than linear dimensionality reduction methods (e.g., principal component analysis) because deep neural networks can learn nonlinear features underlying the uncompressed, input space. An autoencoder consists of two components: an encoder and a decoder. The encoder extracts latent features $\mathbf{z}$ from the high dimensional input data $\mathbf{x}$; the decoder recovers the predicted input data $\mathbf{x}$ from the latent features minimizing the reconstruction error. Autoencoders force a sparse representation of the input by imposing a bottleneck in the network such that the dimension of the latent features is much lower than that of the original input. Mathematically, the autoencoder is described as a set of two functions:

$$\mathbf{z} = \mathbf{h}(\mathbf{x};\, \Omega_{\mathbf{enc}}), \quad (5)$$

$$\hat{\mathbf{x}} = \mathbf{g}(\mathbf{z};\, \Omega_{\mathbf{dec}}), \quad (6)$$

where $\mathbf{h}$ represents the encoder that projects the input $\mathbf{x}$ to the sparse latent features $\mathbf{z}$, whereas $\mathbf{g}$ denotes the decoder that recovers the estimated input $\hat{\mathbf{x}}$ from $\mathbf{z}$; $\Omega_{\mathbf{enc}}$ and $\Omega_{\mathbf{dec}}$ are the network internal parameters (i.e., learnable weight matrices and biases) in the encoder and decoder. The internal network parameters are randomly initialized and then updated during the learning phase that involves the generation of appropriate training and validation sets, and minimization of a loss function. VAEs are a generalization of the standard approach to learning the probability distribution of the latent space. The encoder in VAEs learns two vectors: a vector of mean $\boldsymbol{\mu}$ and a vector of standard deviations $\boldsymbol{\sigma}$. In our case, the inputs to the encoder are models and data generated according to prior assumptions. The inputs to the decoders in the variational approach are random vectors drawn from the Gaussian distribution $\mathbf{z} \sim \mathcal{N}\ (\boldsymbol{\mu},\boldsymbol{\sigma}^2)$, which allows the decoder to sample in the latent space. Figure 1 schematically represents a generic VAE architecture.

As previously mentioned, the learning process minimizes a loss function that is here defined as a linear combination of L2 norm difference between target and reconstructed network outputs (E_x), and the Kullback–Leibler divergence (E_{KL}) that quantifies the similarity of two probability distributions.

Introducing the KL divergence allows making the variational distribution as close as possible to the prior distribution. Therefore, the loss function can be written as:

$$E = E_x + \varepsilon E_{KL}, \qquad (7)$$

with:

$$E_x = \frac{1}{L}||\mathbf{x} - \hat{\mathbf{x}}||^2, \qquad (8)$$

$$E_{KL} = -\frac{1}{2L}\sum_{i=1}^{L} 1 + \log(\sigma_i^2) - \mu_i^2 - \sigma_i^2, \qquad (9)$$

where L denotes the dimensionality of the input data $\mathbf{x}$, and μ_i and σ_i are the ith components of the output vectors $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$ of the encoder, respectively. The term ε in equation 7 represents the trade-off parameter that must be optimally tuned to ensure that the reconstructed output can reproduce the original input and that the learned distribution is similar to the target distribution (see Lopez Alvis et al., 2021 for a detailed discussion). In this way, the autoencoders can successfully learn the compact latent features that represent the original data $\mathbf{x}$.

In this work, we use deep convolutional VAEs to compress model and data spaces in a ES-MDA inversion framework. In other terms, the model unknowns and the data points in our approach are defined in latent spaces whose geometrical properties are defined by properly trained VAE networks. When the compression is applied to the full model space $\mathbf{m}$, we get:

$$\hat{\mathbf{m}} = \mathbf{h_m}(\mathbf{m};\, \Omega_{\mathbf{enc}}), \qquad (10)$$

where $\hat{\mathbf{m}}$ represents the reduced model vector through the $\mathbf{h_m}$ encoder. Otherwise, when the compression is applied to the data we obtain the reduced data vector:

$$\hat{\mathbf{d}} = \mathbf{h_d}(\mathbf{d};\, \Omega_{\mathbf{enc}}), \qquad (11)$$

with $\mathbf{h_d}$ representing the trained encoder for data compression. Therefore, the data likelihood in the reduced model and data space becomes:

$$p(\hat{\mathbf{d}}|\hat{\mathbf{m}}) = \mathcal{N}\left(\hat{\mathbf{d}}; \mathbf{h_d}\left(G\left(\mathbf{g_m}(\hat{\mathbf{m}}; \Omega_{\mathbf{dec}})\right)\right), \mathbf{C}_{\hat{\mathbf{d}}}\right), \qquad (12)$$

where G is the nonlinear forward operator, while the data covariance matrix in the compressed space $\mathbf{C}_{\hat{\mathbf{d}}}$ is learned by the VAE.

The inversion is performed in the compressed model space, then the samples are projected in the full space before the data computation through the FE code. The computed data are then compressed before the evaluation of the data matching. Note that the encoding and decoding operations can be accomplished almost in real-time with a negligible computational cost. Also, note that the encoding and the decoding applied to the data and model space are different and learned during separate training phases. The samples forming the ensemble of the ES-MDA inversion at the last iterations can be finally projected onto the full model space (using the trained decoder) to numerically compute the most likely solution and the associated uncertainties (i.e., model standard deviation) in the original, uncompressed parameter space.

RESULTS

Synthetic inversions

We consider a schematic subsurface resistivity model represented by a rectangular block with a resistivity of 50 Ωm hosted in a homogeneous half-space with resistivity equal to 150 Ωm (Figure 2). The study area is discretized with $11 \times 35 = 385$ rectangular cells with vertical and lateral dimensions of 0.5 m and 1 m, respectively. The resistivity values within the cells correspond to the model parameters to be estimated. We simulate a Wenner acquisition layout with 36 electrodes with a=1 m. The maximum *a* value is 11. This configuration results in 198 data points. In this example, we employ the Wenner layout because it has been also used for the field data acquisition, but the presented inversion framework can be applied to other electrode configurations as well. The FE code was used to compute the noise-free observed dataset that was contaminated with uncorrelated Gaussian noise with a standard deviation equal to the 10% of the total standard deviation of the noise-free apparent resistivity data (i.e., a noise standard deviation equal to 2.06). Figure 3 represents the prior model assumptions used to generate the training, validation, and tests sets. We employ a stationary log-Gaussian prior, while a Gaussian variogram is used as the spatial continuity pattern with horizontal and vertical variogram ranges equal to 4 and 1.5 m, respectively.

For both model and data compression, we use DCVAEs. To simplify the network configuration for the model compression we add a column and a row to the dimension of the study area (11 x 35) so to obtain a grid of 12 rows and 36 columns with dimensions that can be repeatedly and conveniently divided by integer numbers. This additional row and column are removed in the inversion phase before the forward modeling computation and are not considered in the visualization of the final results. For model compression, we first generate 5000 realizations from the prior; 4000 are used for training, whereas 500 form the validation and test sets. The time needed to generate the prior models is negligible while the training runs in less than five minutes (20 epochs) on the GPU previously mentioned. The characteristics of the implemented DCVAE for model compression are shown in Table 1. Note that in this case the full 385D model space is reduced to a 40D domain. The Adam optimizer (Balles and Hennig, 2018) is used to minimize the loss function. We employ a batch size of 24, whereas a dropout of 10% is used before the fully connected layer to prevent overfitting (Wu and Gu, 2015). We set the trade-off parameter ε in the loss function to 0.1. In all layers, we adopt the LeakyRelu activation function with a leakage value of 0.1 (Dubey and Jain, 2019). Batch normalization is used as a regularization operator (Santurkar et al., 2018), while the initial learning rate is set to 0.001 and this value is multiplied by 0.95 every epoch.

For data compression, we first compute the data associated with all the 5000 models previously generated. Again this ensemble is divided into training, validation, and test with a split of 80/10/10. The network configuration used for data compression is represented in Table 2. Note that because of its trapezoidal shape, the apparent resistivity section is first flattened to a 1D vector before feeding into the DCVAE. In this case, the 198D data space has been sparsely re-parameterized by mean and variance vectors of dimensions 50. Again, the Adam optimizer is used to minimizes the loss functions while the trade-off parameter in the loss function is set to 0.05. The batch size and the learning rate are the same used for the model compression. The training phase takes three minutes on the same hardware resources previously mentioned.

As an example, Figure 4 represents some prior realizations extracted from the test set and the associated DCVAE approximations. The satisfactory agreement between target and approximated models proves that the network has been properly trained and hence it can capture most details of the original models. Note that once the network is fully trained it can also be used to generate models (e.g., the models forming the initial ensemble for the ES-MDA inversion) according to the prior without employing any geostatistical generation tool.

We run the ES-MDA inversion in the compressed domain using an ensemble of 250 resistivity models. We run inversions also with smaller and larger ensembles but this number revealed to be the optimal compromise between the computational costs related to the forward evaluations and the stability of the estimated uncertainties (see discussion below). With stability, we mean that the estimated uncertainty does not sensibly change for an increased ensemble size. Indeed, smaller ensembles resulted in underestimated posterior uncertainties, while larger ensembles (e.g., 500, 1000 models) provided uncertainty similar to the one obtained with 250 models. See Aleardi et al. (2021b) for a more detailed discussion on how the ensemble size affects the uncertainty estimation in ERT inversion solved via ensemble-based algorithms. For comparison, the MCMC employs 30 chains and runs in the compressed model space for 3000 iterations, with a burn-in period of 500. The potential scale reduction factor (PSFR; Brooks and Gelman, 1998) is used to monitor the convergence of the MCMC sampling towards a stable PPD. For computationally feasibility reasons the MCMC sampling has not been run in the full data and model spaces.

As a comparison Figure 5 illustrates, the most likely ES-MDA solution obtained with the implemented approach, the solution provided by the MCMC inversion, the one obtained by the ES-MDA inversion running in the full data and model space, and the predictions of a gradient-based inversion performed with the IP4DI software (Karaoulis et al., 2013). Both the ensemble-based inversions have been run for four iterations. The rectangular resistivity anomaly is well recovered and properly located by all methods, although the gradient-based inversion yields a final result that slightly underestimates the resistivity values in the deeper part of the model, while the probabilistic approaches slightly overestimate the maximum depth reached by the low resistivity body. From the many inversion tests carried out with the ES-MDA running in the full space we noted that stable uncertainties quantifications can be achieved with an ensemble of 1000 models (see again the discussion below), thus meaning that the compression of the model and data spaces provides similar model predictions but with a total number of forward evaluations (and computing time) four times smaller.

Figure 6 compares the posterior standard deviations estimated by the ES-MDAs running in the compressed and full spaces, and the MCMC sampling. The two ensemble-based inversions provide congruent uncertainty quantifications, although we observe that with DCVAE we get a slight underestimation of the posterior uncertainties due to the reduced model space dimension, especially in the least illuminated part of the subsurface. Some differences are also observed with respect to the MCMC results particularly for the cells poorly informed by the data, for which the two ES-MDA inversions tend to underestimate the posterior uncertainties. However, in all cases, we observe that the lower uncertainties are located in correspondence with the low resistivity anomaly while the precision of the results decreases moving at the lateral edge and the bottom of the study area.

To better investigate how the ensemble size affects the estimated uncertainty, Figure 7 compares the standard deviation sections computed for the ES-MDA inversion with and without model compression and running with different ensemble sizes. It emerges that with DCVAE the inversion yields stable posterior quantifications with smaller ensembles; In particular stable uncertainties can be achieved with 250 and 1000 models, respectively, for the inversion running in the reduced and full model and data spaces. Differently, the most likely models are very similar for all the tests illustrated previously, and hence they are not shown here.

Figure 8 shows for the MCMC inversion the evolution of the negative log-likelihood for the 30 chains and the PSRF for some model parameters. We observe that the steady-state of the Markov chain is attained in 500 iterations (i.e., corresponding to the selected burn-in period), while 1500 iterations are needed to reach stable PPD estimations (a PSRF lower than 1.1). This means that the MCMC inversion needs 45000 forward evaluations to converge (1500 iterations × 30 chains). This value is 45 and 11.25 times larger than the number of forward runs needed by the ensemble-based inversions running in the compressed and full spaces, respectively.

Figures 9a and 9b show some resistivity models from the initial ensemble generated with the trained network and the corresponding models at the last ES-MDA iteration, respectively. We observe that all the final models successfully predict the low resistivity anomaly located in the central part of the investigated profile.

Figure 10 shows a comparison between the observed apparent resistivity values and the data generated on the most likely solutions predicted by the two ES-MDA inversions and the MCMC sampling, along with the prediction of the gradient-based algorithm. All the methods achieve satisfactory data matching. Figure 11 illustrates for the ES-MDA inversion running in the compressed domains, a comparison between the observed data and the data computed on the initial and final ensemble of models. This comparison demonstrates that the inversion eventually converges toward an ensemble of resistivity profiles that satisfactorily reproduce the observed apparent resistivity values.

As a final and more quantitative assessment of the results, we list in Table 3 the 90% coverage ratio, and the root-mean-square errors (RMSE) between true and predicted models and observed and predicted data. We remind that the 90% coverage ratio quantifies the percentage of resistivity values in the true model that fall within the 90% confidence interval as estimated by the probabilistic inversion. Since the gradient-based inversion does not provide uncertainty quantifications the coverage ratios are only computed for the MCMC sampling and the two ES-MDA inversions. The four inversions give very similar data predictions while the model predictions are slightly more accurate for the three probabilistic inversions due to the underestimation of the background resistivity of the gradient-based approach. A better data matching can be achieved by the gradient-based inversion just by lowering the trade-off regularization parameter but at the expense of an increased scattering in the recovered solution. As expected, the two ES-MDA algorithms provide lower coverage ratios than the MCMC sampling, thereby demonstrating that this last method gives slightly more accurate uncertainty estimations, but this happens at the expense of a dramatic increase of the computational workload due to the higher number of forward evaluations needed to converge. For the two ES-MDA inversions, we also note that the coverage ratio slightly increases if we move from the compressed to the full model and data spaces, but this again happens at the expense of an increased computational cost. For example, if we consider parallel codes and the hardware resources previously mentioned, the ES-MDA with DCVAE runs in less than 10 minutes. The same inversion approach without compression takes more than 40 minutes, while the MCMC sampling running in the compressed domains takes 900 minutes to converge. Finally, from a practical point of view, we deem that the model uncertainties provided by the presented approach are reasonable and comparable to those yielded by the other probabilistic inversion methods.

Field data application

We now apply the presented approach to invert a field dataset acquired for levee monitoring along the Parma river (Italy). We refer the interested reader to Hojat et al. (2019b) for more information about the study area. We invert a single dataset acquired with electrodes buried in a 0.5 m-deep trench and employing the Wenner acquisition layout with 48 electrodes for a unit spacing of a = 2 m. The investigated site covers an area that is 94 m wide and 14 m deep and it is discretized with rectangular cells with vertical and lateral dimensions of 1 m and 2 m, respectively. This configuration results in $15\times47= 705$ resistivity values to be estimated from 360 data points. Similar to the synthetic example, we have conveniently added a column and a row to the dimension of the inversion grid to simplify the network configuration.

We exploit all the available information about the investigated site to define the prior distribution of model parameters. In particular, we still employ a log-Gaussian prior and a spatial variability pattern described by a Gaussian variogram with lateral and vertical ranges equal to 6 m and 2 m, respectively. In this area, we mainly expect a low-resistivity clay body that around 2-3 m depth hosts a more permeable layer with higher resistivity values associated with the presence of sand and gravel. The a-priori simplifies the actual distribution of the resistivity values in the synthetic model. Therefore, to validate this prior we compare summary statistics of observed and simulated data generated from prior realizations to determine if the observed data samples are outliers. Figure 12 demonstrates that the observations always lie within the 95 % confidence interval derived from apparent resistivity sections generated by prior realizations. In mathematical terms, this means that the observed data and the data derived from the prior can be considered as realizations of the same random variable (Pradhan and Mukerji, 2020).

To train the networks we again generate 5000 prior realizations and we define the training, validation, and test sets using the same split previously employed in the synthetic experiment. The main characteristics of the network used for model compression (see Table 4) are similar to those employed in the synthetic case and listed in Table 1, but in this application, the full model domain is compressed to a 150D space. We also use the same batch size, optimization algorithm, initial learning rate, and the maximum number of epochs. The fact that almost the same network configuration properly works in both the synthetic and field example illustrates the flexibility of the approach, which means that a successful application does not depend on the selected network configuration (see the discussion section for additional considerations). However, some care must be devoted to tuning the trade-off parameter that here is set to 0.2. The comparison between models extracted from the test set and the corresponding DCVAE approximations demonstrate that the network has been properly trained (Figure 13). Table 5 depicts the network hyperparameters used for data compressions. Again we employ an architecture similar to the one used in the synthetic example, but with a trade-off parameter of 0.08. In this example, the full data space is sparsely compressed into an 80D domain.

Figure 14 compares the most likely models estimated by the two ES-MDA inversions and by the gradient-based approach. The three methods again provide similar and comparable estimates and the slight low resolution of the two ES-MDA approaches with respect to the gradient-based outcomes is related to the different regularizations strategies applied. Figure 15 compares the posterior standard deviations numerically estimated from the final ensembles associated with the ES-MDA inversion running in the compressed and full model and data space, respectively. Again, we note that the uncertainty estimated when the data and model spaces are reduced is slightly lower than that estimated without compression. This is particularly evident for the model parameters less informed by the data for example for the cells located at the bottom and the lateral edges of the study site. Similar to the synthetic example, we note that the two ES-MDA inversions achieve stable posterior assessments with very different ensemble sizes: When the DCVAE are employed only 500 models are needed while 2000 models are requested by the inversion without compression. Again both these inversions have been run for four iterations. Therefore, the use of the DCVAE still guarantees a significant decrease in the number of forward evaluations, and thus a decrease in the computing time of the probabilistic inversion. For example, the ES-MDA with DCVAE runs in 15 minutes while about an hour is needed without model and data compression. These computing times are still referred to parallel codes running on the same hardware resources previously described.

Figures 16 shows some models forming the final ensemble for the ES-MDA inversion with DCVAE. Again, we observe that the inversion satisfactorily converges toward congruent results. Indeed, all the models at the very last iteration show similar characteristics such as the low resistivity anomaly in the shallowest and central part of the study area, and the high resistivity body buried around 3 m depth.

DISCUSSION

We applied a probabilistic approach to solve the ERT problem in which DCVAEs have been used to increase the computational efficiency of the inversion procedure and to avoid the so-called ensemble collapsing issue. On the one hand, the computational burden of the ES-MDA inversion largely depends on the number of ensemble members and the cost of running the forward computations. On the other hand, the ensemble size should be large enough to get accurate uncertainty evaluations, and its dimension should increase with the dimension of the parameter space. Therefore, running the inversion in compressed spaces significantly reduces the number of ensemble members and the computational cost needed for reliable uncertainty quantifications.

In our application, we employ log-Gaussian prior but deep generative models are helpful for data assimilation and inverse problems with non-Gaussian models as well (Canchumuni et al., 2019; Bao et al., 2020). In our implementation, the use of nonparametric priors is theoretically possible, but it requires the application of a normal score transformation. In this context, the sampling would be performed in the original domain, whereas the inversion would run in the normal score transformed space. We expect this approach to be quite accurate for unimodal distributions, but further investigations are needed in the case of multimodal priors.

The reason for uncertainty underestimation or overestimation in the case of model and data space compressions is that data reduction makes the inverse problem underdetermined while model compression makes the inversion overdetermined. Ideally, the compression of model space should be as small as possible to sparsely represent the original domain and to effectively mitigate the ill-conditioning of the problem. For this reason, the reduction of the parameter space should be a compromise between the expected model resolutions, and the accuracy of the uncertainty assessments. Also, note that the posterior uncertainty is underestimated in the ES-MDA if the number of ensemble members is not sufficient to statistically represent the model space (Aleardi et al. 2021b). Reducing the data space partially mitigates the underestimation because it makes the problem more underdetermined, thus increasing its condition number and consequently the posterior uncertainties

(Grana et al. 2019). However, in practice, it is often difficult (especially for nonlinear problems) to determine the optimal dimensions of the reduced model and data to get uncertainty quantification equal to the one obtained in the full spaces.

In the proposed approach a sufficient number of prior realizations and associated data are needed to train the networks. From our experience, 4000 examples are enough for successful training. In our many experiments (not shown here for brevity) we found that many different DCVAE architectures (with a different number of layers, filter dimensions) work similarly. The final one has been selected as a reasonable compromise between the computational cost of the training phase and the accuracy of the predictions. However, special care must be devoted to properly tune the trade-off parameter of the loss function, thus ensuring that the reconstructed output can reproduce the original input and that the learned distribution approximates the target distribution. In this work, we selected this parameter using a trial-and-error procedure that is facilitated by the limited computational cost of the training phase (very few minutes on the employed hardware resources). We also found that the optimal range for this parameter is not that narrow: for example, in the synthetic application, all the values between 0.05-0.15 provide very similar model approximations. If needed the modeling error related to the uncertainty in the network reconstruction can additionally be propagated into the final PPDs. This is an interesting point that is worthy of a deeper investigation in further studies. Here, we limit the comparison of ES-MDA and MCMC only to the synthetic example because running an MCMC sampling to solve the field inversion is computationally impractical on the limited hardware resources employed in this study (it would probably require a couple of weeks to converge).

Reducing the computational cost of a probabilistic ERT inversion is needed to make this approach more appealing than popular local inversion algorithms. The Bayesian framework provides crucial information regarding the uncertainties affecting the recovered solution. Such estimated model uncertainties can be used to generate different subsurface scenarios in agreement with the prior assumptions and the acquired data. We deem that the outcome of such a probabilistic approach adds an extra layer of information over gradient-based solutions that could contribute to a more informed decision-making process in many ERT applications (e.g., monitoring applications). For this reason, we are also working to extend the presented approach to time-lapse ERT inversion.

As demonstrated in Aleardi et al. (2021) also linear compression methods are very effective to reduce model and data spaces in 2D ERT inversion. However, the popularity that machine learning compression methods have recently gained in the geophysical community, motivated us to contemporarily assess the applicability of DCVAEs to solve the same problem. The limited computational effort needed for network training, along with the limited human effort needed to set up an appropriate DCVAE architecture, make the total computational cost of ESMDA inversions with linear and non-linear compressions very similar (i.e., note that also for linear compressions, an accurate analysis conducted on prior realizations must be done to select the optimal number of basis functions to retain). However, deep neural networks exploit the nonlinear and spatial patterns in the input, and thus they generally outperform linear dimension reduction methods for more complex (e.g., three-dimensional) models/data. Indeed, our preliminary attempts on 3D and time-lapse ERT inversion indicate superior performances of DCVAE over linear strategies. We are still investigating these challenging topics, but some preliminary results can be found in Vinciguerra and Aleardi (2021).

CONCLUSIONS

This work was aimed at decreasing the computational cost of a probabilistic ERT inversion by exploiting the sampling ability of ES-MDA and the compression ability of DCVAEs. The DCVAEs were used as a dimensionality reduction strategy to avoid spurious correlation and ensemble collapse and to decrease the dimensionality of the problem, hence reducing the computational cost of the inversion. Indeed, our tests illustrated that the ensemble size needed for stable uncertainty quantifications significantly decreases for an ES-MDA inversion running in the compressed space with respect to the same inversion approach working in the full model and data domains. More in detail, the use of DCVAE reduced the total number of forward evaluations of the stochastic inversion by four times in both the synthetic and field data experiments. Our tests also demonstrated that the implemented inversion can provide most likely models and uncertainty quantifications comparable to those yielded by an ES-MDA algorithm running in the full model and data space, and an MCMC sampling working in the compressed domains. All these probabilistic approaches estimated most likely solutions very similar to the results of a gradient-based inversion. We also observed that due to the dimensionality reduction, the proposed ES-MDA inversion is prone to slightly underpredict the uncertainties for the parameters poorly informed by the data. However, from a practical point of view, the estimated uncertainties remain extremely valuable since they offer insights into the accuracy of the recovered model features and allow assessing the precision of the results. The presented method can be easily adapted to solve other geophysical inverse problems.

Conflict of interest

The authors declare no conflict of interest

Data and code availability

The data that support the findings of this study are available from the corresponding author upon reasonable request

REFERENCES

Aleardi, M., Vinciguerra, A., and Hojat, A. (2020). A geostatistical Markov Chain Monte Carlo inversion algorithm for electrical resistivity tomography. Near Surface Geophysics, 19(1), 7-26.

Aleardi, M., Vinciguerra, A., and Hojat, A. (2021a). A convolutional neural network approach to electrical resistivity tomography. Journal of Applied Geophysics, 193, 104434.

Aleardi, M. (2015). The importance of the Vp/Vs ratio in determining the error propagation, the stability and the resolution of linear AVA inversion: a theoretical demonstration. Bollettino di Geofisica Teorica ed Applicata, 56(3), 357-366.

Aleardi, M., (2020). Discrete cosine transform for parameter space reduction in linear and non-linear AVA inversions. Journal of Applied Geophysics, 179, 104106.

Aleardi, M., Vinciguerra, A., and Hojat, A. (2021b). Ensemble-based electrical resistivity tomography with data and model space compression. Pure and Applied Geophysics, 178, 1781-1803.

Aleardi, M., and Salusti, A. (2020). Markov chain Monte Carlo algorithms for target-oriented and interval-oriented amplitude versus angle inversions with non-parametric priors and non-linear forward modellings. Geophysical Prospecting, 68(3), 735-760.

Arosio, D., Munda, S., Tresoldi, G., Papini, M., Longoni, L., and Zanzi, L. (2017). A customized resistivity system for monitoring saturation and seepage in earthen levees: installation and validation. Open Geosciences, 9(1), 457-467. doi: 10.1515/geo-2017-0035.

Aster, R. C., Borchers, B., and Thurber, C. H. (2018). Parameter estimation and inverse problems. Elsevier.

Balles, L., and Hennig, P. (2018). Dissecting adam: The sign, magnitude and variance of stochastic gradients. International Conference on Machine Learning, 404-413.

Bao, J., Li, L., and Redoloza, F. (2020). Coupling ensemble smoother and deep learning with generative adversarial networks to deal with non-Gaussianity in flow and transport data assimilation. Journal of Hydrology, 590, 125443.

Bièvre, G., Oxarango, L., Günther, T., Goutaland, D. and Massardi M. (2018). Improvement of 2D ERT measurements conducted along a small earth-filled dyke using 3D topographic data and 3D computation of geometric factors. Journal of Applied Geophysics 153, 100-112.

Binley, A., and Slater, L. (2020). Resistivity and induced polarization: Theory and applications to the near-surface earth. Cambridge University Press.

Brooks, S. P., and Gelman, A. (1998). General methods for monitoring convergence of iterative simulations. Journal of computational and graphical statistics, 7(4), 434-455.

Chen, Y., and Oliver, D. S. (2017). Localization and regularization for iterative ensemble smoothers. Computational Geosciences, 21(1), 13-30.

Dahlin, T. (2020). Geoelectrical monitoring of embankment dams for detection of anomalous seepage and internal erosion - experiences and work in progress in Sweden. Fifth International Conference on Engineering Geophysics (ICEG), Al Ain, UAE, DOI: 10.1190/iceg2019-053.1.

Dubey, A. K., and Jain, V. (2019). Comparative study of convolution neural network's relu and leaky-relu activation functions. In Applications of Computing, Automation and Wireless Systems in Electrical Engineering, 873-880. Springer, Singapore.

Emerick, A. A., and Reynolds, A. C. (2013). Ensemble smoother with multiple data assimilation. Computers & Geosciences, 55, 3-15.

Gao, Z., Li, C., Liu, N., Pan, Z., Gao, J., and Xu, Z. (2020). Large-Dimensional Seismic Inversion Using Global Optimization With Autoencoder-Based Model Dimensionality Reduction. IEEE Transactions on Geoscience and Remote Sensing, 59(2), 1718-1732.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). Deep learning. MIT press.

Grana, D., Passos de Figueiredo, L., and Azevedo, L. (2019). Uncertainty quantification in Bayesian inverse problems with model and data dimension reduction. Geophysics, 84(6), M15-M24.

Hermans, T., and Paepen, M. (2020). Combined inversion of land and marine Electrical resistivity tomography for submarine groundwater discharge and saltwater intrusion characterization. Geophysical Research Letters, 47(3), e2019GL085877.

Hojat, A., Arosio, D., Di Luch, I., Ferrario, M., Ivanov, V.I., Longoni, L., Madaschi, A., Papini, M., Tresoldi, G. and Zanzi, L. (2019a). Testing ERT and fiber optic techniques at the laboratory scale to monitor river levees. 25th European Meeting of Environmental and Engineering Geophysics, The Hague, Netherlands, DOI: 10.3997/2214-4609.201902440.

Hojat, A., Arosio, D., Longoni, L., Papini, M., Tresoldi, G., and Zanzi, L. (2019b). Installation and validation of a customized resistivity system for permanent monitoring of a river embankment. EAGE-GSM 2nd Asia Pacific Meeting on Near Surface Geoscience and Engineering, Kuala Lumpur, Malaysia. Doi: 10.3997/2214-4609.201900421.

Kang, X., Shi, X., Revil, A., Cao, Z., Li, L., Lan, T., and Wu, J. (2019). Coupled hydrogeophysical inversion to identify non-Gaussian hydraulic conductivity field by jointly assimilating geochemical and time-lapse geophysical data. Journal of Hydrology, 578, 124092.

Kang, X., Kokkinaki, A., Kitanidis, P. K., Shi, X., Lee, J., Mo, S., and Wu, J. (2021). Hydrogeophysical Characterization of Nonstationary DNAPL Source Zones by Integrating a Convolutional Variational Autoencoder and Ensemble Smoother. Water Resources Research, 57(2), e2020WR028538.

Karaoulis, M., Revil, A., Tsourlos, P., Werkema, D. D., and Minsley, B. J. (2013). IP4DI: A software for time-lapse 2D/3D DC-resistivity and induced polarization tomography. Computers & Geosciences, 54, 164-170.

Kingma, D. P., and Welling, M. (2013). Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114.

Laloy, E., Hérault, R., Jacques, D., and Linde, N. (2018). Training-image based geostatistical inversion using a spatial generative adversarial neural network. Water Resources Research, 54(1), 381-406.

Liu, M., and Grana, D. (2018b). Stochastic nonlinear inversion of seismic data for the estimation of petroelastic properties using the ensemble smoother and data reparameterization. Geophysics, 83(3), M25-M39.

Loke, M. H., Papadopoulos, N., Wilkinson, P. B., Oikonomou, D., Simyrdanis, K., and Rucker, D. F. (2020). The inversion of data from very large three-dimensional electrical resistivity tomography mobile surveys. Geophysical Prospecting, 68(8), 2579-2597.

Lopez-Alvis, J., Laloy, E., Nguyen, F., and Hermans, T. (2021). Deep generative models in inversion: The impact of the generator's nonlinearity and development of a new approach based on a variational autoencoder. Computers & Geosciences, 104762.

Luo, X., Bhakta, T., Jakobsen, M., and Nævdal, G. (2018). Efficient big data assimilation through sparse representation: A 3D benchmark case study in petroleum engineering. PloS one, 13(7), e0198586.

Luo, X., Lorentzen, R. J., Valestrand, R., and Evensen, G. (2019). Correlation-based adaptive localization for ensemble-based history matching: applied to the Norne field case study. SPE Reservoir Evaluation & Engineering, 22(03), 1084-1109.

Mandelli, S., Borra, F., Lipari, V., Bestagini, P., Sarti, A., and Tubaro, S. (2018). Seismic data interpolation through convolutional autoencoder. In SEG Technical Program Expanded Abstracts 2018 (pp. 4101-4105). Society of Exploration Geophysicists.

Moradipour, M., Ranjbar, H., Hojat, A., Karimi-Nasab, S. and Daneshpajouh, S. (2016). Laboratory and field measurements of electrical resistivity to study heap leaching pad no. 3 at Sarcheshmeh copper mine. 22nd European Meeting of Environmental and Engineering Geophysics, DOI: 10.3997/2214-4609.201602140.

Norooz, R., Olsson, P.I., Dahlin, T., Günther, T., Bernston, C. (2021). A geoelectrical pre-study of Älvkarleby test embankment dam: 3D forward modelling and effects of structural constraints on the 3D inversion model of zoned embankment dams. Journal of Applied Geophysics, 191, 104355.

Pradhan, A., and Mukerji, T. (2020). Seismic Bayesian evidential learning: Estimation and uncertainty quantification of sub-resolution reservoir properties. Computational Geosciences, 24, 1121–1140.

Rucker, D.F., Fink, J.B., Loke, M.H. (2011) Environmental monitoring of leaks using time-lapsed long electrode electrical resistivity. Journal of Applied Geophysics, 74(4), 242-254.

Saad, O. M., and Chen, Y. (2020). Deep denoising autoencoder for seismic random noise attenuation. Geophysics, 85(4), V367-V376.

Sambridge, M., and Mosegaard, K. (2002). Monte Carlo methods in geophysical inverse problems. Reviews of Geophysics, 40(3), 3-1.

Santurkar, S., Tsipras, D., Ilyas, A., and Madry, A. (2018). How does batch normalization help optimization?. In Advances in neural information processing systems, 2483-2493.

Sajeva, A., Aleardi, M., Mazzotti, A., Bienati, N., and Stucchi, E. (2014). Estimation of velocity macro-models using stochastic full-waveform inversion. In SEG Technical Program Expanded Abstracts 2014. Society of Exploration Geophysicists, 1227-1231.

Sætrom, J., and Omre, H. (2013). Uncertainty quantification in the ensemble Kalman filter. Scandinavian Journal of Statistics, 40(4), 868-885.

Sen, M. K., and Stoffa, P. L. (2013). Global optimization methods in geophysical inversion. Cambridge University Press.

Tarantola, A. (2005). Inverse problem theory, siam.

Tso, C. H. M., Johnson, T. C., Song, X., Chen, X., Kuras, O., Wilkinson, P., Uhlemann, S., Chambers J., and Binley, A. (2020). Integrated hydrogeophysical modelling and data assimilation for geoelectrical leak detection. Journal of Contaminant Hydrology, 234, 103679.

Vinciguerra, A., Aleardi, M., Hojat, A., and Stucchi, E. (2021a). Discrete Cosine Transform for Parameter Space Reduction in Bayesian Electrical Resistivity Tomography. Geophysical Prospecting. Doi: 10.1111/1365-2478.13148.

Vinciguerra, A., and Aleardi, M. (2021). Ensemble-Based Time-Lapse ERT Inversion with Model and Data Space Compression Through Deep Variational Autoencoders. In NSG2021 2nd Conference on Geophysics for Infrastructure Planning, Monitoring and BIM. European Association of Geoscientists & Engineers, 2021(1), 1-5.

Vrugt, J. A. (2016). Markov chain Monte Carlo simulation using the DREAM software package: Theory, concepts, and MATLAB implementation. Environmental Modelling & Software, 75, 273-316.

Whiteley, J., Chambers, J.E., and Uhlemann, S., (2017). Integrated monitoring of an active landslide in lias group mudrocks, north yorkshire, UK. In: Hoyer, S. (Ed.), GELMON 2017: 4th International Workshop on GeoElectrical Monitoring, 27.

Wu, H., and Gu, X. (2015). Towards dropout training for convolutional neural networks. Neural Networks, 71, 1-10.

FIGURES

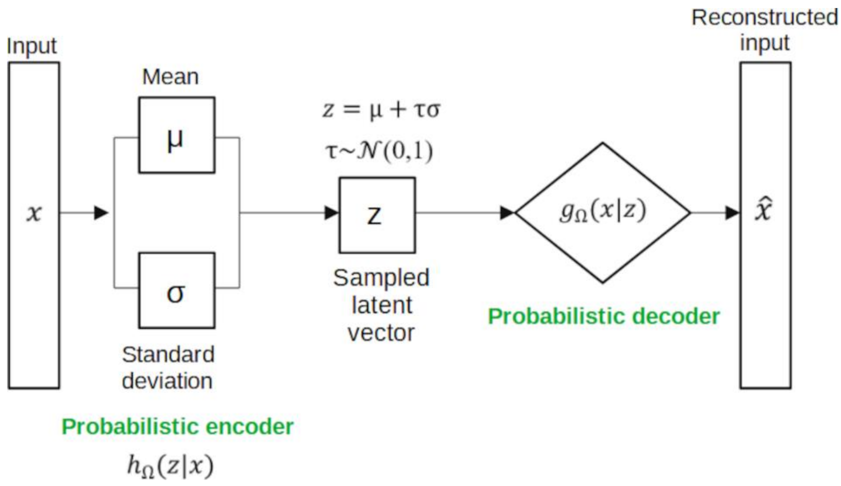

Figure 1: A schematic representation of VAEs.

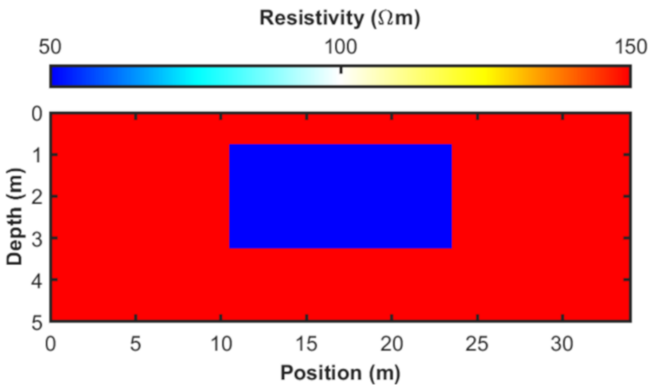

Figure 2: The true model for the synthetic inversion.

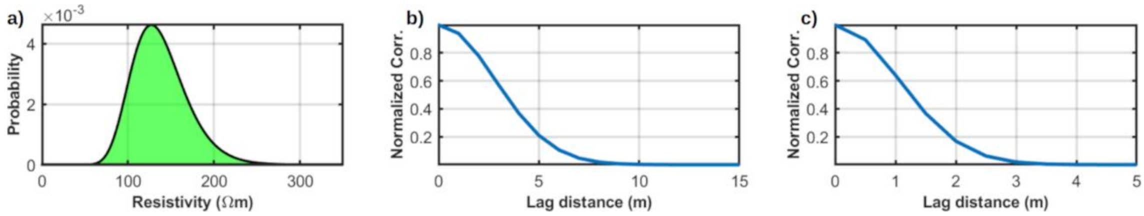

Figure 3: a) Log-Gaussian prior distribution for the synthetic example. b), and c) spatial correlation functions associated with the assumed 2-D variogram model for the horizontal and vertical directions, respectively.

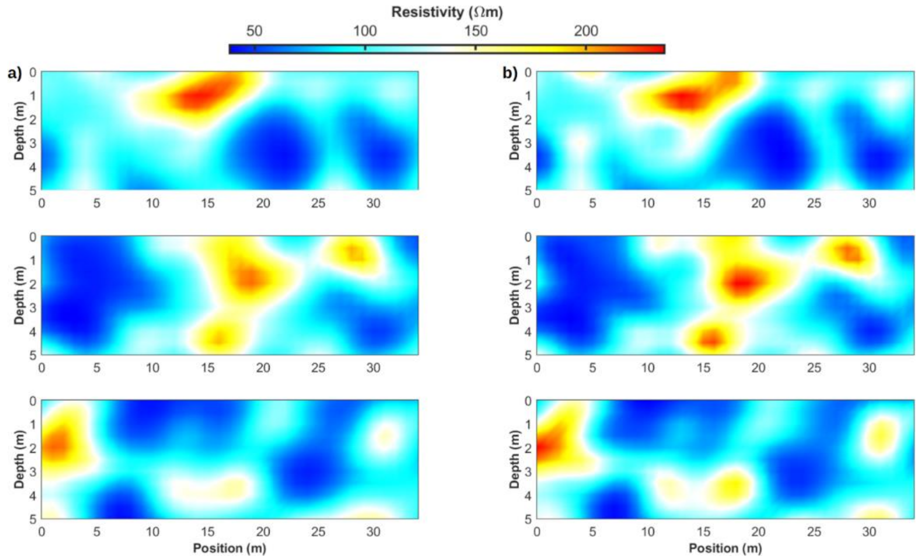

Figure 4: a) Example of DCVAE approximations of resistivity models extracted from the test set. b) The corresponding target, uncompressed models.

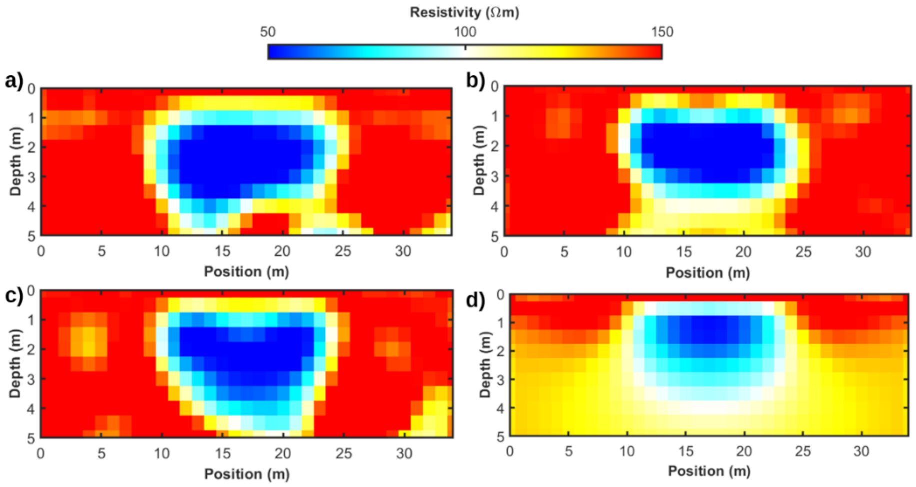

Figure 5: a) The most likely model predicted by the ES-MDA with DCVAE. b) The most likely solution predicted by the ES-MDA without DCVAE. c) The most likely model provided by the MCMC inversion with DCVAE. d) Gradient-based solution.

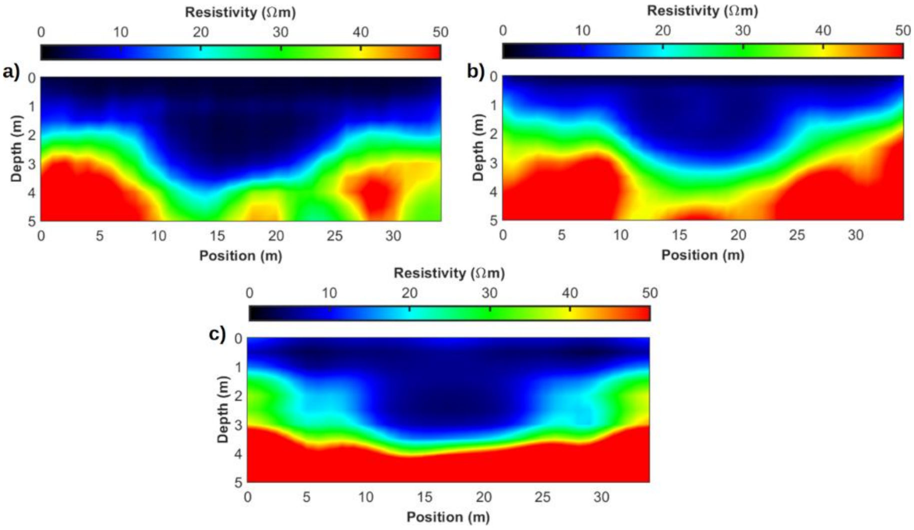

Figure 6: Posterior standard deviation estimated with the ES-MDA running in the compressed and full model and data spaces (a), and b), respectively). c) Posterior standard deviation estimated by the MCMC inversion working in the compressed domains.

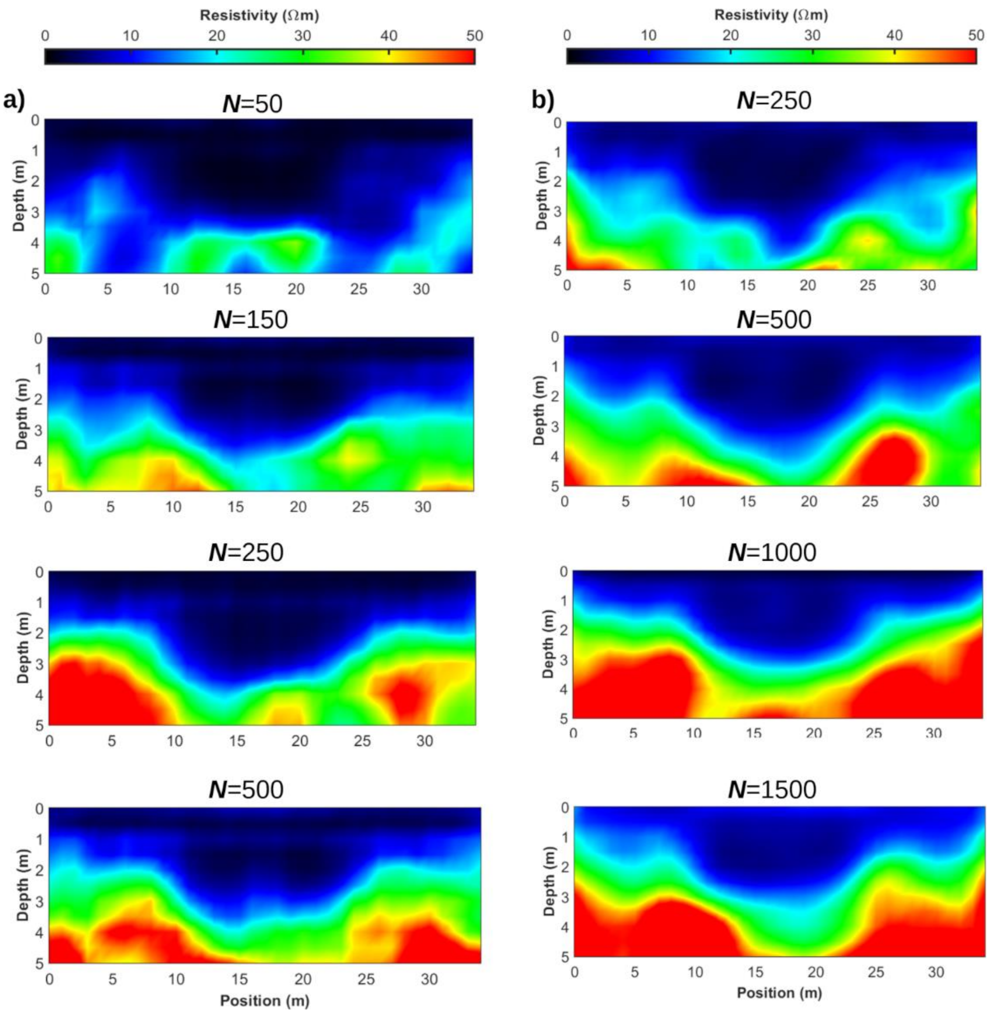

Figure 7: Standard deviation sections derived from inversion tests that employ different numbers of models in the ensemble (*N*). a) ES-MDA inversion with DCVAE. b) ES-MDA inversion without model and data compression.

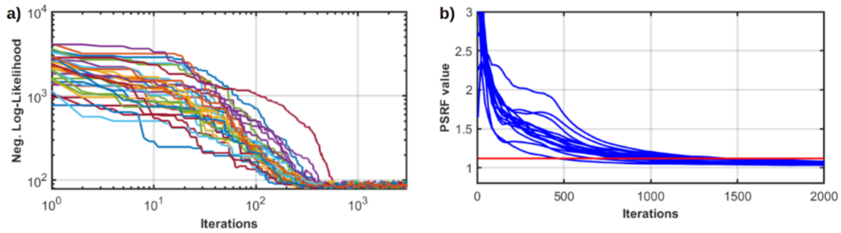

Figure 8: Evolution of the negative log-likelihood for the 30 chains during the MCMC sampling. b) For some model parameters we show the evolution of the PSRF. The red line represents the threshold of convergence fixed (as usual) at 1.1.

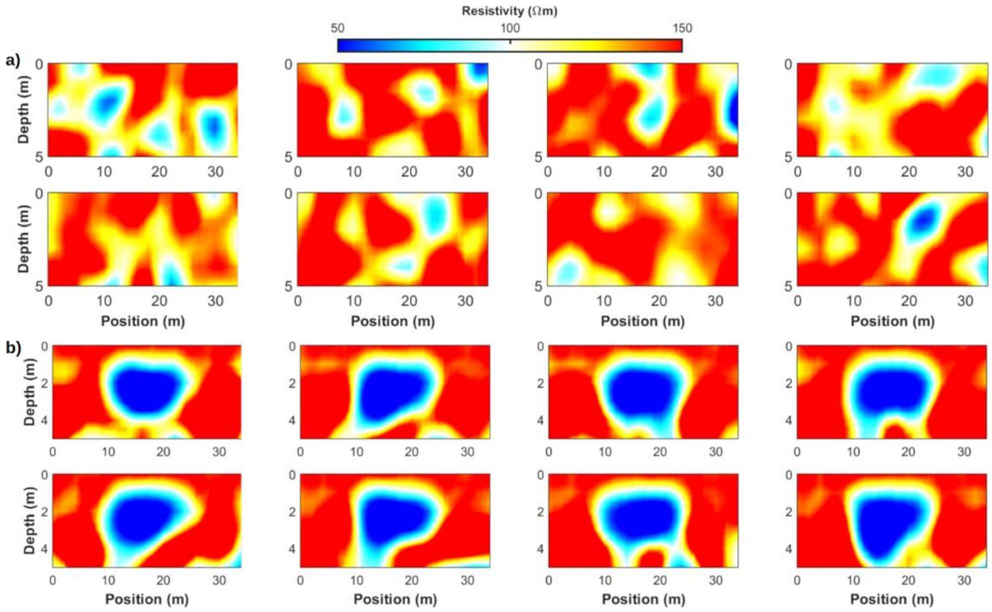

Figure 9: Some examples of prior (a) and posterior (b) resistivity models forming, respectively, the initial and final ensemble of the ES-MDA inversion running in the compressed spaces.

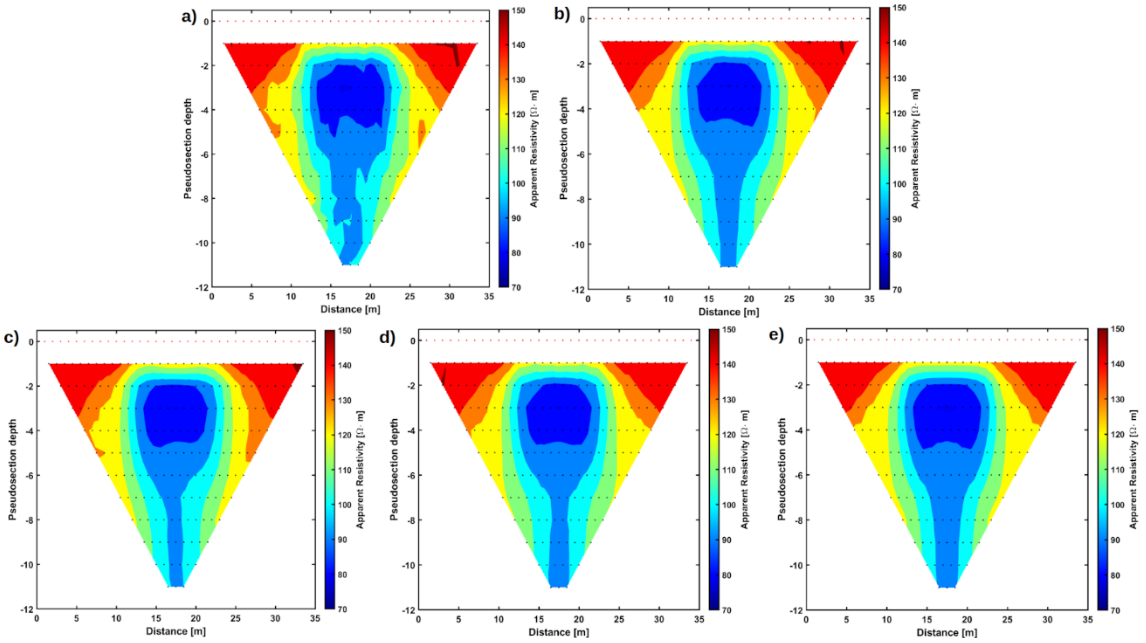

Figure 10: a) Observed pseudosection. b) Data predicted from the most likely solution of the ES-MDA inversion with DCVAE. c) Data predicted from the most likely solution of the ES-MDA inversion without DCVAE. d) Data predicted from the most likely solution of the MCMC inversion. e) Data computed from the gradient-based result.

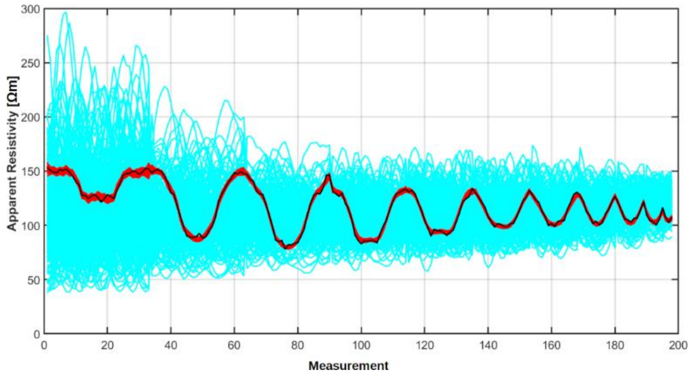

Figure 11: Comparison between the observed data (black line), the data computed on the initial ensemble of models (cyan lines), and the data associated with the models at the last ES-MDA iteration (red lines). For graphical convenience, all the computed pseudo sections have been flattened to 1D vectors. This figure refers to the inversion running in the compressed domains, but similar conclusions would have been drawn for the ES-MDA inversion running in the full model and data spaces.

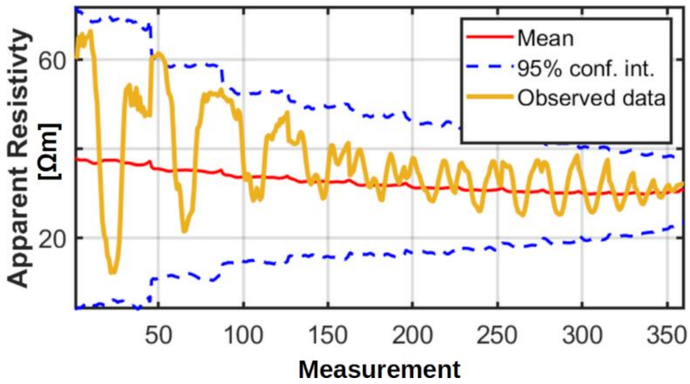

Figure 12: The observed data (yellow line) compared with the data computed on the mean prior model (red line) and the 95% confidence interval (blue dotted lines) derived from data generated on prior realizations.

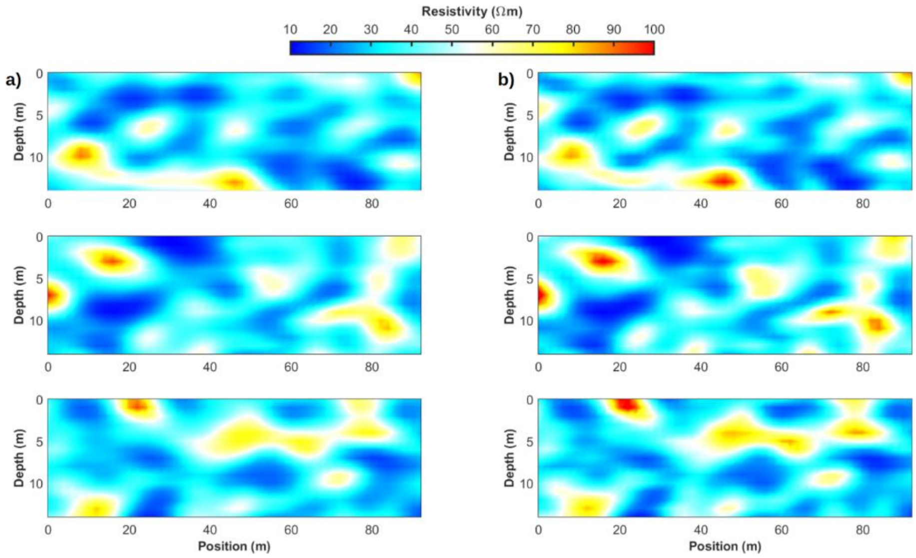

Figure 13: a) Example of DCVAE approximations of resistivity models extracted from the test set. b) The corresponding original, uncompressed models.

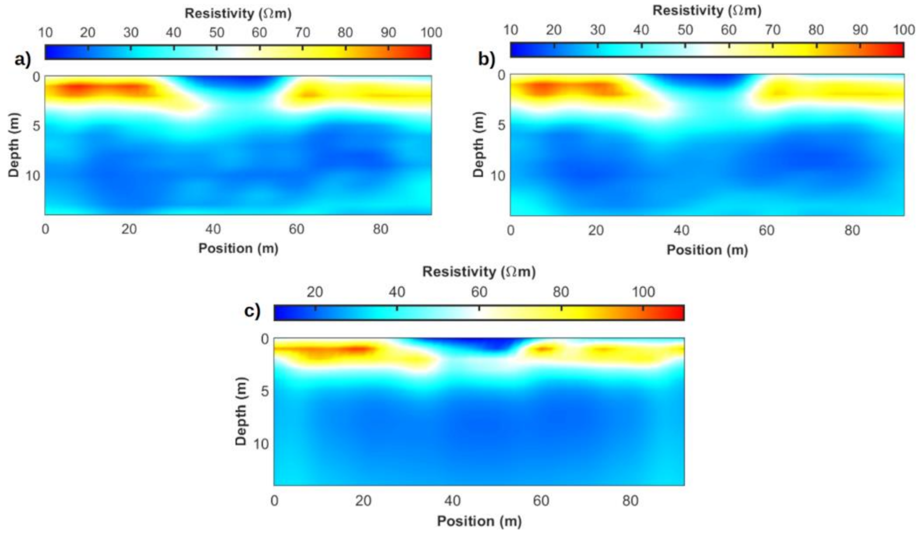

Figure 14: a) The most likely model estimated by the ES-MDA inversion with DCVAE. b) The most likely model provided by the ES-MDA without DCVAE. c) Model estimated by the gradient-based inversion.

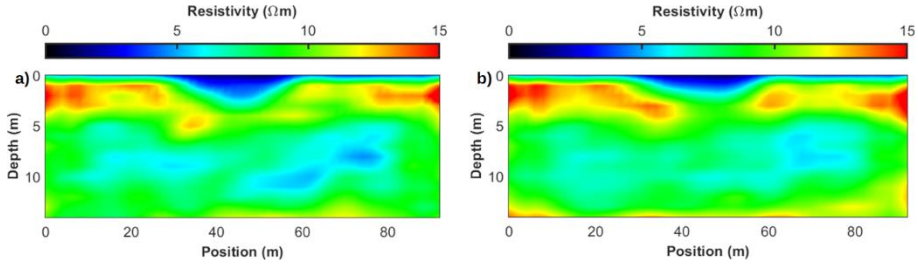

Figure 15: Posterior standard deviation estimated with the ES-MDA inversion running in the compressed spaces. b) Posterior uncertainty provided by the ES-MDA inversion without compression.

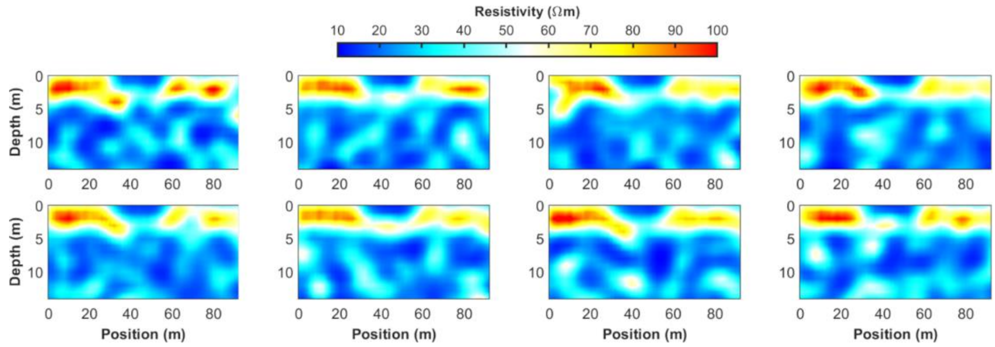

Figure 16: Models extracted from the ensemble at the last ES-MDA inversion working in the compressed model and data domains.

TABLES

	Layer	Dimension
	Input	12×36
ENCODER	Conv2D(FilterSize =3×3, Layers=8, Stride=2)+LeakyRelu(0.1)+BatchNorm	6×18×8
	Conv2D(FilterSize =3×3, Layers=16, Stride=2)+LeakyRelu(0.1)+BatchNorm	3×9×16
	Conv2D(FilterSize =3×3, Layers=32, Stride=2)+LeakyRelu(0.1)+BatchNorm	2×5×32
	Flatten	320
	Fully Connected Layer (Dropout 10%)	80
Latent Space	Mean Variance	40 40
DECODER	Fully Connected Layer	27
	Reshape	3×9
	TransposeConv2D(FilterSize=3×3,Layers=32, Stride=2)+LeakyRelu(0.1)+BatchNorm	6×18×32
	TransposeConv2D(FilterSize =3×3,Layers=16,Stride=2)+LeakyRelu(0.1)+BatchNorm	12×36×16
	TransposeConv2D(FilterSize =3×3,Layers=1,Stride=1)+LeakyRelu(0.1)+BatchNorm	12×36

Table 1: Network architecture of the DCVAE used for model compression in the synthetic example.

	Layer	Dimension
	Input	198
ENCODER	Conv1D(FilterSize =3×1, Layers=8, Stride=2)+LeakyRelu(0.1)+BatchNorm	99×8
	Conv1D(FilterSize =3×1, Layers=16, Stride=2)+LeakyRelu(0.1)+BatchNorm	50×16
	Flatten	800
	Fully Connected Layer (Dropout 10%)	100
Latent Space	Mean Variance	50 50
DECODER	Fully Connected Layer	99
	TransposeConv1D(FilterSize =3×1, Layers=16, Stride=2)+LeakyRelu(0.1)+BatchNorm	198×16
	TransposeConv1D(FilterSize =3×1,Layers=1,Stride=2)+LeakyRelu(0.1)+BatchNorm	198

Table 2: Network architecture of the deep convolutional VAE used to compress the data space in the synthetic example.

	RMSE Model	RMSE Data	90% coverage ratio
ES-MDA with DCVAE	114.11	3.12	84.31%
ES-MDA without DCVAE	113.26	3.02	86.64%
MCMC with DCVAE	114.40	3.08	88.24%
Gradient-Based	118.03	4.10	Not available

Table 3: Table listing for each considered inversion approach the RMSE between the true and the predicted models (shown in Figure 5), the RMSE between the observed and the data computed on the predicted models (see Figure 10), and for the three probabilistic inversions the 90% coverage ratios.

	Layer	**Dimension**
	Input	16×48
ENCODER	Conv2D(FilterSize =3×3, Layers=16, Stride=2)+LeakyRelu(0.1)+BatchNorm	8×24×16
	Conv2D(FilterSize =3×3, Layers=32, Stride=2)+LeakyRelu(0.1)+BatchNorm	4×12×32
	Conv2D(FilterSize =3×3, Layers=64, Stride=2)+LeakyRelu(0.1)+BatchNorm	2×6×64
	Flatten	768
	Fully Connected Layer (Dropout 10%)	300
Latent Space	Mean Variance	150 150
DECODER	Fully Connected Layer	48
	Reshape	4×12
	TransposeConv2D(FilterSize=3×3,Layers=32, Stride=2)+LeakyRelu(0.1)+BatchNorm	8×24×32
	TransposeConv2D(FilterSize =3×3,Layers=16,Stride=2)+LeakyRelu(0.1)+BatchNorm	16×48×16
	TransposeConv2D(FilterSize =3×3,Layers=1,Stride=1)+LeakyRelu(0.1)+BatchNorm	16×48

Table 4: Network architecture of the DCVAE used to compress the model space in the field data application.

	Layer	**Dimension**
	Input	360×1
ENCODER	Conv1D(FilterSize =3×1, Layers=8, Stride=2)+LeakyRelu(0.1)+BatchNorm	180×1×8
	Conv1D(FilterSize =3×1, Layers=16, Stride=2)+LeakyRelu(0.1)+BatchNorm	90×1×16
	Flatten	1440
	Fully Connected Layer (Dropout 10%)	160
Latent Space	Mean Variance	80 80
DECODER	Fully Connected Layer	180
	TransposeConv1D(FilterSize =3×1, Layers=16, Stride=2)+LeakyRelu(0.1)+BatchNorm	360×1×16
	TransposeConv1D(FilterSize =3×1,Layers=1,Stride=2)+LeakyRelu(0.1)+BatchNorm	360×1

Table 5: Network architecture of the DCVAE used to compress the data space in the field test.