Stochastic electrical resistivity tomography with ensemble smoother and deep convolutional autoencoders

Mattia Aleardi\textsuperscript{1,}\textsuperscript{*}, Alessandro Vinciguerra\textsuperscript{2,1}, Eusebio Stucchi\textsuperscript{1} and Azadeh Hojat\textsuperscript{3,4}

\textsuperscript{1}University of Pisa, Earth Sciences Department, via S. Maria 53, Pisa, 56126, Italy, \textsuperscript{2}University of Florence, Earth Sciences Department, via. G. La Pira 4, Florence, 50121, Italy, \textsuperscript{3}Shahid Babonar University of Kerman, Department of Mining Engineering, Jomhouri Blvd., Kerman, 76188, Iran, and \textsuperscript{4}Politecnico di Milano, Department of Civil and Environmental Engineering, Piazza Leonardo da Vinci 32, Milano, 20133, Italy

Received September 2021, revision accepted December 2021

\textbf{ABSTRACT}

To reduce both the computational cost of probabilistic inversions and the ill-posedness of geophysical problems, model and data spaces can be reparameterized 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 a Markov Chain Monte Carlo 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 Markov Chain Monte Carlo sampling, but with a significant reduction of the computational cost.

\textbf{Key words:} Electrical resistivity tomography, Inversion.

\textsuperscript{*}Email: mattia.aleardi@unipi.it

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

\textbf{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 \textit{et al.}, 2011; Moradipour \textit{et al.}, 2016; Whiteley \textit{et al.}, 2017; Arosio \textit{et al.}, 2017; Bièvre \textit{et al.}, 2018; Hojat \textit{et al.}, 2020).
2019a; Dahlin, 2020; Hermans and Paepen, 2020; Aleardi et al., 2020; Loke et al., 2020; Norooz et al., 2021; Aleardi et al., 2021a). Due to incomplete data coverage and noise contamination, 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 Mos engage, 2002; Sen and Stoffa, 2013) algorithms can be employed for accurate posterior probability density (PPD) estimations in nonlinear problems. However, the considerable number 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 and 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., 2016; 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 an underestimation or overestimation of the model uncertainty (Grana et al., 2019). For this reason, the trade-off between model resolution and model uncertainty must always be 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 (2018a), 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 (DC-VAEs; 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. DC-VAEs 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 DC-VAE-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 modelling 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.

© 2021 European Association of Geoscientists & Engineers, Near Surface Geophysics, 20, 160–177
For each iteration:

1. Apply the forward operator and compute the observables.
2. Generate realizations according to the prior.

$\tilde{C}_{\text{dd}}$ and $\tilde{C}_{\text{md}}$ represent the empirical covariance matrices estimated from the ensemble members, whereas $\mathbf{m}^p$ and $\mathbf{d}^p$ 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 $\alpha$ 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 $(\mathbf{d}^p)_1,...,N$;
   b. Perturb the observations according to: $\tilde{\mathbf{d}}_k = \mathbf{d} + \sqrt{\alpha} \mathbf{C}^{-1/2}_d \mathbf{n}$, with $\mathbf{n} \sim \mathcal{N}(0,1)$, where $\mathbf{I}$ is the identity matrix;
   c. Update the ensemble using equations (2)–(4) with $\mathbf{C}_d$ replaced by $\alpha/\mathbf{C}_d$.

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 $\mathbf{C}_{\text{dd}}$ and $\mathbf{C}_{\text{md}}$ 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 observed data according to the Gaussian distribution $\mathcal{N}(\mathbf{d}; \mathbf{C}_d)$, in which $\mathbf{C}_d$ is the data covariance. The subscripts $m$ and $p$ denote the updated (current iteration) and prior (previous iteration) variables, respectively; $\tilde{\mathbf{C}}_{\text{md}}^p$ and $\tilde{\mathbf{C}}_{\text{dd}}^p$ represent the empirical covariance matrices estimated from the ensemble members, whereas $\mathbf{m}^p$ and $\mathbf{d}^p$ are the empirical ensemble mean of the model parameters and predicted data, respectively.
Figure 1 A schematic representation of VAEs.

predicted input data \( x \) 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:

\[
z = h(x; \Omega_{\text{enc}}),
\]

\[
\hat{x} = g(z; \Omega_{\text{dec}}),
\]

where \( h \) represents the encoder that projects the input \( x \) to the sparse latent features \( z \), whereas \( g \) denotes the decoder that recovers the estimated input \( \hat{x} \) from \( z \); \( \Omega_{\text{enc}} \) and \( \Omega_{\text{dec}} \) 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. Variant or variational autoencoders (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 \( \mu \) and a vector of standard deviations \( \sigma \). 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 \( z \sim \mathcal{N}(\mu, \sigma^2) \), 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:

\[
E = E_x + \varepsilon E_{KL},
\]

with:

\[
E_x = \frac{1}{L} \| x - \hat{x} \|^2,
\]

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

where \( L \) denotes the dimensionality of the input data \( x \), and \( \mu_i \) and \( \sigma_i \) are the \( i \)th components of the output vectors \( \mu \) and \( \sigma \) of the encoder, respectively.

The term \( \varepsilon \) 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 \( x \).

In this work, we use deep convolutional VAEs to compress model and data spaces in an 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 \( m \), we get:

\[
\hat{m} = h_m(m; \Omega_{\text{enc}}),
\]

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

\[
\hat{d} = h_d(d; \Omega_{\text{enc}}),
\]

with \( h_d \) representing the trained encoder for data compression. Therefore, the data likelihood in the reduced model and data space becomes:

\[
p(\hat{d} | \hat{m}) = \mathcal{N}(\hat{d}; h_d(G(\hat{m}; \Omega_{\text{dec}})) \cdot C_d),
\]

where \( G \) is the nonlinear forward operator, while the data covariance matrix in the compressed space \( C_d \) 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 finite-elements 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 negli-
gible 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 iter-
ations 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 repre-
sented by a rectangular block with a resistivity of 50 $\Omega$ m
hosted in a homogeneous half-space with resistivity equal to
150 $\Omega$ m (Fig. 2). The study area is discretized with $11 \times 35 =
385$ rectangular cells with vertical and lateral dimensions of
0.5 m and 1.0 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 $a = 1.0$ m. The maximum $a$ value is 11. This config-
uration results in 198 data points. In this example, we em-
ploy 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
finite-elements code was used to compute the noise-free ob-
served dataset that was contaminated with uncorrelated Gauss-
ian 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 test sets. We employ a stationary log-
Gaussian prior, while a Gaussian variogram is used as the spa-
tial continuity pattern with horizontal and vertical variogram
ranges equal to 4.0 and 1.5 m, respectively.

For both model and data compression, we use deep con-
volutional variational autoencoders (DCVAEs). To simplify
the network configuration for the model compression, we
add a column and a row to the dimension of the study area
($11 \times 35$) to obtain a grid of 12 rows and 36 columns with di-
mensions that can be repeatedly and conveniently divided by
integer numbers. The additional row and column are removed
in the inversion phase before the forward modelling computa-
tion and are not considered in the visualization of the final
results. For model compression, we first generate 5000 realiza-
tions 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 men-
tioned. 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 min-
imize 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 $\varepsilon$ 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 reg-
ularization 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 reparameterized by mean and variance vectors of dimensions 50. Again, the Adam optimizer is used to minimize 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 ensemble smoother with multiple data assimilation (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 also run inversions 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 and 1000 models) provided uncertainty similar to the one obtained with 250 models. See Aleardi et al. (2021b) for a more detailed
Figure 4 (a) Example of DCVAE approximations of resistivity models extracted from the test set. (b) The corresponding target, uncompressed models.

discussion on how the ensemble size affects the uncertainty estimation in electrical resistivity tomography inversion solved via ensemble-based algorithms. For comparison, the Markov chain Monte Carlo (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 (PSRF; Brooks and Gelman, 1998) is used to monitor the convergence of the MCMC sampling towards a stable posterior probability density (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 uncertainty 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.
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.

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

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 45,000 forward evaluations to converge (1500 iterations \( \times \) 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.

Figure 9(a) and (b) 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...
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.

The final ensemble of models. This comparison demonstrates that the inversion eventually converges towards 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 computed only 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
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 pseudosections 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.

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 $a = 2.0$ 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 1.0 m and 2.0 m, respectively. This configuration results in $15 \times 47 = 705$ 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 (Fig. 13). Table 5 depicts the network hyperparameters used for data compression. 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.

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

<table>
<thead>
<tr>
<th>Method</th>
<th>RMSE Model</th>
<th>RMSE Data</th>
<th>90% coverage ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>ES-MDA with DCVAE</td>
<td>114.11</td>
<td>3.12</td>
<td>84.31</td>
</tr>
<tr>
<td>ES-MDA without DCVAE</td>
<td>113.26</td>
<td>3.02</td>
<td>86.64</td>
</tr>
<tr>
<td>MCMC with DCVAE</td>
<td>114.40</td>
<td>3.08</td>
<td>88.24</td>
</tr>
<tr>
<td>Gradient based</td>
<td>118.03</td>
<td>4.10</td>
<td>Not available</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Layer</th>
<th>Dimension</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input</td>
<td>16 × 48</td>
</tr>
<tr>
<td>ENCODER</td>
<td></td>
</tr>
<tr>
<td>Conv2D(FilterSize = 3 × 3, Layers = 16, Stride = 2) + LeakyRelu(0.1) + BatchNorm</td>
<td>8 × 24 × 16</td>
</tr>
<tr>
<td>Conv2D(FilterSize = 3 × 3, Layers = 32, Stride = 2) + LeakyRelu(0.1) + BatchNorm</td>
<td>4 × 12 × 32</td>
</tr>
<tr>
<td>Conv2D(FilterSize = 3 × 3, Layers = 64, Stride = 2) + LeakyRelu(0.1) + BatchNorm</td>
<td>2 × 6 × 64</td>
</tr>
<tr>
<td>Flatten</td>
<td>768</td>
</tr>
<tr>
<td>Fully connected layer (dropout 10%)</td>
<td>300</td>
</tr>
<tr>
<td>Latent Space</td>
<td></td>
</tr>
<tr>
<td>Mean</td>
<td>150</td>
</tr>
<tr>
<td>Variance</td>
<td>150</td>
</tr>
<tr>
<td>DECODER</td>
<td></td>
</tr>
<tr>
<td>Fully connected layer</td>
<td>48</td>
</tr>
<tr>
<td>Reshape</td>
<td>4 × 12</td>
</tr>
<tr>
<td>TransposeConv2D(FilterSize = 3 × 3, Layers = 32, Stride = 2) + LeakyRelu(0.1) + BatchNorm</td>
<td>8 × 24 × 32</td>
</tr>
<tr>
<td>TransposeConv2D(FilterSize = 3 × 3, Layers = 16, Stride = 2) + LeakyRelu(0.1) + BatchNorm</td>
<td>16 × 48 × 16</td>
</tr>
<tr>
<td>TransposeConv2D(FilterSize = 3 × 3, Layers = 1, Stride = 1) + LeakyRelu(0.1) + BatchNorm</td>
<td>16 × 48</td>
</tr>
</tbody>
</table>

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

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

<table>
<thead>
<tr>
<th>Layer</th>
<th>Dimension</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input</td>
<td>360 x 1</td>
</tr>
<tr>
<td>ENCODER</td>
<td>Conv1D(FilterSize = 3 x 1, Layers = 8, Stride = 2) + LeakyRelu(0.1) + BatchNorm</td>
</tr>
<tr>
<td></td>
<td>Conv1D(FilterSize = 3 x 1, Layers = 16, Stride = 2) + LeakyRelu(0.1) + BatchNorm</td>
</tr>
<tr>
<td></td>
<td>Flatten</td>
</tr>
<tr>
<td></td>
<td>Fully connected layer (dropout 10%)</td>
</tr>
<tr>
<td>Latent Space</td>
<td>Mean</td>
</tr>
<tr>
<td></td>
<td>Variance</td>
</tr>
<tr>
<td>DECODER</td>
<td>Fully connected layer</td>
</tr>
<tr>
<td></td>
<td>TransposeConv1D(FilterSize = 3 x 1, Layers = 16, Stride = 2) + LeakyRelu(0.1) + BatchNorm</td>
</tr>
<tr>
<td></td>
<td>TransposeConv1D(FilterSize = 3 x 1, Layers = 1, Stride = 2) + LeakyRelu(0.1) + BatchNorm</td>
</tr>
</tbody>
</table>

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.

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.

Figure 16 shows some models forming the final ensemble for the ES-MDA inversion with DCVAE. Again, we observe that the inversion satisfactorily converges towards 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 electrical resistivity tomography (ERT) problem in which deep convolutional variational autoencoders (DCVAEs) have been used to increase the computational efficiency of the inversion

© 2021 European Association of Geoscientists & Engineers, Near Surface Geophysics, 20, 160–177
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.

procedure and to avoid the so-called ensemble collapsing issue. On the one hand, the computational burden of the ensemble smoother with multiple data assimilation (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 (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 and 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 and 0.15 provide very similar model approximations. If needed the modelling error related to the uncertainty in the network reconstruction can additionally be propagated into the final posterior probability densities (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 Markov chain Monte Carlo (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. (2021b), 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 ES-MDA inversions with linear and nonlinear compressions very similar (note also that 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 electrical resistivity tomography inversion by exploiting the sampling ability of ensemble smoother with multiple data assimilation (ES-MDA) and the compression ability of deep convolutional variational autoencoders (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 a Markov chain Monte Carlo 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.

**DATA AND CODE AVAILABILITY**

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

**ORCID**

Mattia Aleardi [https://orcid.org/0000-0003-1433-0281](https://orcid.org/0000-0003-1433-0281)

**REFERENCES**


hydraulic conductivity field by jointly assimilating geochemical and time-lapse geophysical data. *Journal of Hydrology*, 578, 124092.


