Fast probabilistic inversion of airborne and ground-based EM data using GPU-accelerated rejection sampling

Introduction

Airborne and ground-based electromagnetic (EM) surveys are routinely used to characterize the shallow subsurface for groundwater, raw-material, and environmental applications. End users are, however, rarely interested in resistivity per se, but in answers to decision-relevant questions such as *"what is the probability that a continuous layer of at least 20 m of producible high-quality raw material exists at this location, with no more than 5 m of overburden?"* Answering such questions requires integrating diverse geo-information (EM measurements, borehole logs, geological maps, expert knowledge) and propagating all associated uncertainties into a quantitative risk estimate.

A fully probabilistic treatment of the inverse problem provides exactly this capability. The posterior distribution of resistivity and lithology contains all the information needed to compute the probability of any user-defined criterion related to the model parameters. Despite its theoretical appeal, probabilistic inversion has seen limited adoption in EM practice, because (a) classical Markov chain Monte Carlo methods are difficult to combine with prior information from diverse sources, and (b) they are widely perceived as computationally expensive. We demonstrate that this perception is no longer justified. For the EM data considered here, a full non-linear probabilistic inversion is in fact faster than a classical linearized deterministic inversion.

We present an open-source Python module called `integrate_module` (MIT license) developed within the INTEGRATE[1] project (Hansen and Gulbrandsen, 2025), that addresses both limitations. `integrate_module` implements the extended rejection sampler with lookup tables, allowing integration of arbitrarily complex prior information (discrete lithology classes, non-Gaussian resistivity distributions, groundwater table, …), and produces independent posterior realisations efficiently enough that survey-scale probabilistic inversion becomes practical on a single laptop, and much faster on a GPU. Code, documentation, and reproducible examples, using both ground and airborne data, are available at GitHub[2].

Method and Theory

Following Tarantola and Valette (1982), the inverse problem is formulated as the combination of prior information $\rho(\mathbf{m},\mathbf{n})$ and likelihood $L(\mathbf{m},\mathbf{n})$ into a posterior $\sigma(\mathbf{m},\mathbf{n}) \propto \rho(\mathbf{m},\mathbf{n})\cdot L(\mathbf{m},\mathbf{n})$, where **m** denotes (as an example) resistivity, **n** denotes discrete lithology. Two characteristics make this formulation attractive. First, $\rho(\mathbf{m},\mathbf{n})$ jointly describes continuous resistivity and discrete lithology, so geological knowledge that can be sampled can be represented as prior information. The prior can be represented by any algorithm that produces realisations; no analytical form is needed. To construct and sample from geologically realistic priors we rely on Madsen et al. (2023). Second, $L(\mathbf{m},\mathbf{n})$ can represent heterogeneous data types (EM measurements with multivariate Gaussian noise, categorical borehole observations with multinomial noise, or any combination).

To sample $\sigma(\mathbf{m},\mathbf{n})$, `integrate_module` uses the extended rejection sampler with lookup tables (Hansen et al. 2016; Hansen 2021). A lookup table of N independent prior realizations and their forward responses is generated once using GA-AEM (Brodie, 2015). Then, for each observed dataset, the likelihood $L(\mathbf{m},\mathbf{n})$ is evaluated for every entry of the table, and a model is accepted as a posterior sample with probability $P_{acc} = (L / c)^{1/T}$, where $c \geq \max_i L_i$. For Gaussian noise, the annealing temperature T is equivalent to inflating the data covariance from C_d to $T\cdot C_d$, so the accepted models are exact realizations of the posterior under an inflated-noise model. T is selected automatically to ensure a minimum number of accepted models per sounding, guaranteeing robust posterior statistics and acting as a diagnostic for

1 http://integrate.nu/
2 https://github.com/AUProbGeo/integrate_module/

prior-forward-data consistency. Approximate inversion results can thus be obtained in minutes, for surveys with 10000+ data points, simply by adjusting the size N of the lookup table. The Bayesian evidence is obtained as a by-product, enabling formal hypothesis testing and model comparison.

Unlike Metropolis-based methods, the rejection sampler requires neither random-walk proposals nor a differentiable prior, and accepted samples are independent by construction (no burn-in, no autocorrelation analysis). The lookup table is built once per shared prior and reused across all data locations, which is the source of its efficiency for localized 1D problems (Hansen 2021). `integrate_module` is implemented in Python with a modular HDF5 interface. Any external forward modeler or prior generator can used. One only must be able to store the output can be if properly formatted HDF5 files.

The methodology applies to both ground-based and airborne EM data, with one practical difference. For ground-based systems such as tTEM, the source-receiver geometry is fixed and known. For airborne systems, the flight height varies and is measured with uncertainty. It is therefore treated both as an additional model parameter in the prior and as observed data with associated uncertainty. The lookup table must span this extra dimension, requiring a substantially larger number of prior realizations N. The sampling algorithm itself is unchanged.

Implementation and computational performance using NumPy and JAX

The rejection sampler reduces to a sequence of vectorizable array operations, by evaluating the likelihood of every lookup-table entry for every sounding, followed by probabilistic acceptance. We provide two pure-Python implementations: NumPy, exploiting fast parallel computation on the CPU, and JAX (Bradbury et al., 2018), which just-in-time (JIT) compiles the same computation for the GPU. For the examples shown below, the JAX/GPU implementation accelerates the rejection sampler (EM data only) by a factor of about 90. All 11693 soundings are inverted in less than 15 seconds on the GPU versus around 23 minutes on the CPU, while the one-off forward computation of the lookup table takes about 25 minutes. JAX/CPU provides a speedup of about 4, compared to NumPy for EM data inversion. The bottleneck thus shifts almost entirely to the forward computation, which is embarrassingly parallel and reusable across surveys sharing the same prior.

The examples shown below use data from the Daugaard, Denmark, tTEM survey (11,693 dual-moment soundings; Hansen et al., 2026). Figure 1 illustrates the output of the inversion: posterior statistics of resistivity and lithology along a profile. For these data, a classical linearized spatially constrained inversion (SCI; Viezzoli et al., 2008; Vignoli et al., 2015) requires about 3.5 hours of computation, while the complete probabilistic, non-linear workflow, including building the lookup table, completes in less than 1 hour using NumPy alone, and far less using the GPU. The long-standing argument that probabilistic methods are too slow therefore no longer holds. The non-linear probabilistic inversion is faster than the linearized deterministic alternative, while delivering full uncertainty quantification. We therefore advocate probabilistic inversion not only for final, geologically informed data integration, but also for the initial resistivity-only inversion of a survey, where one would traditionally reach for a deterministic method.

Interrogating the posterior using a JSON query tool with a natural-language interface

Given joint posterior samples of $\sigma(\mathbf{m}, \mathbf{n})$ at every sounding location, any user-defined function of the model parameters (typically describing the subsurface) can be evaluated as a probability by simple Monte Carlo: $P(C \mid d_{obs}) \approx (1/N_{acc}) \, \Sigma \, f_C(m_i, n_i)$, where f_C is the indicator function of criterion C and the sum is over accepted posterior realizations. Such statistics are a key advantage of the probabilistic approach, but require the user to handle posterior realisations rather than a single model. To remove this barrier between the posterior and the end user, `integrate_module` implements a query tool in which any criterion is expressed as a structured set of rules through a JSON interface, specifying, e.g., lithology classes, cumulative or first-occurrence thickness, depth intervals, and resistivity thresholds. Queries can be written manually and are evaluated exactly, by Monte Carlo, on the stored posterior realizations, without re-running the inversion.

On top of the JSON interface, an optional large language model (LLM) layer is also implemented, translating a question posed in plain language into the corresponding JSON query, together with a plain-English interpretation that the user can verify before evaluation. The LLM, locally hosted or external, never computes the probability itself but only provides the query representing the criteria *C* in JSON format. This allows geologists, water managers, and decision-makers with no programming or geophysical background to interrogate the full probabilistic inversion result directly, using natural language. As a concrete, operationally meaningful example, consider the plain-language question: *"What is the probability that there exists a continuous layer of at least 20 m of producible high-quality raw material (sand/gravel-class lithology) starting at a depth not exceeding 5 m below ground?"* The LLM layer translates this into a JSON query, evaluated on each accepted posterior realization at each sounding location. The fraction of realizations satisfying the criterion is a direct, uncertainty-aware estimate of $P(C \mid d_{obs})$. Figure 2 shows the resulting probability map. Areas of high probability are spatially coherent and correspond to known buried-valley features. The query also handles questions like "What is the 10th, 50th and 90th percentile of the cumulative sand/gravel thickness in the upper 50 m?". Any other criteria, e.g. minimum aquifer thickness or presence of a confining layer, require only a new query, not a new inversion.

Conclusions

`integrate_module` provides an efficient, open-source framework for probabilistic inversion of airborne and ground-based EM data that (i) integrates arbitrarily structured prior information with EM data, borehole observations, and other information sources; (ii) returns joint posterior samples of resistivity and lithology from which any decision-relevant probability can be computed; and (iii) is faster than classical linearized deterministic inversion for the data shown here. Combined with a JSON query tool and an LLM-based natural-language interface, the framework removes both the computational and the accessibility arguments against probabilistic methods, which we argue should replace deterministic inversion as the default choice for these types of EM data.

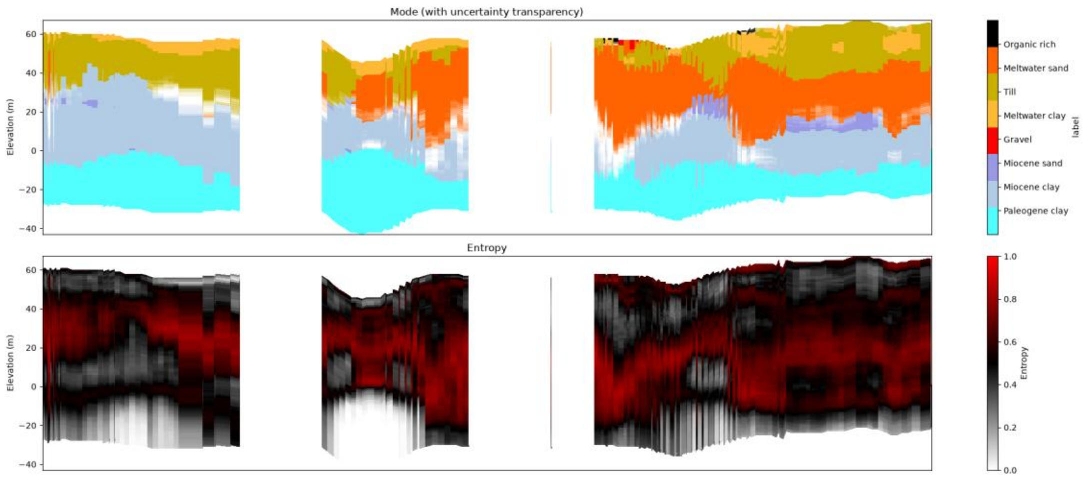

Figure 1 *Example of a North-South profile at UTMX=543500, showing the marginal posterior mode (most probable) (top) and entropy (bottom) of lithology. Transparency is added to the mode class with highest entropy.*

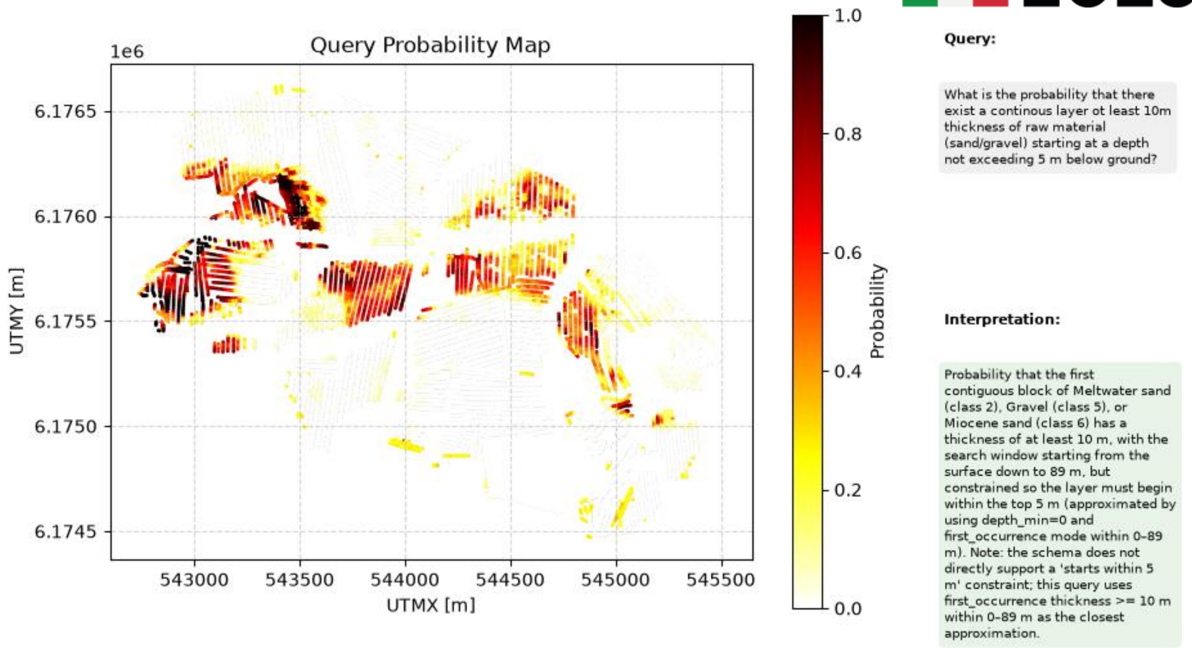

Figure 2 *Probabilistic outcome of a query on a sample of the posterior distribution, identifying the probability of locating producible raw materials (sand/gravel).*

Acknowledgements

This work is developed as part of the INTEGRATE project (https://integrate.nu/), funded by Innovation Fund Denmark, Grant #2081-00009B. We acknowledge that the input from all members of the INTEGRATE project, has been crucial for the development if the code we present here.

References

Bradbury et al. [2018]. JAX: Composable transformations of Python+NumPy programs. http://github.com/jax-ml/jax , v0.10.0.
Brodie, R.C. [2015]. GA-AEM open-source code for 1D forward modelling and inversion of AEM data. Geoscience Australia. https://github.com/GeoscienceAustralia/ga-aem.
Hansen, T.M., Cordua, K.S., Jacobsen, B.H. and Mosegaard, K. [2016]. Accounting for imperfect forward modelling in geophysical inverse problems—exemplified for crosshole tomography. Geophysics, 81(4), H1–H17.
Hansen, T.M. [2021]. Efficient probabilistic inversion using the rejection sampler—exemplified on airborne EM data. Geophysical Journal International, 224(1), 543–557.
Hansen, T.M. and Gulbrandsen, M.L. [2025]. Informative mapping of construction aggregate resources through statistical data analysis (INTEGRATE). In: International Geostatistics Congress 2024, Springer. doi:10.1007/978-3-031-92870-3_30.
Hansen, T.M., Falk, F.A., Jørgensen, F. and Nørgaard, J. [2026]. INTEGRATE – a Python package for fast localized probabilistic inversion using informed priors, applied to EM data. Submitted to Computers & Geosciences.
Madsen, R.B., Hansen, T.M. and Jørgensen, F. [2023]. A method for constructing joint resistivity-lithology priors for probabilistic geophysical inversion. Computers & Geosciences, 173, 105320.
Tarantola, A. and Valette, B. [1982]. Inverse problems = quest for information. Journal of Geophysics, 50(1), 159–170.
Viezzoli, A., Christiansen, A.V., Auken, E. and Sørensen, K. [2008]. Quasi-3D modeling of airborne TEM data by spatially constrained inversion. Geophysics, 73(3), F105–F113.
Vignoli, G., Fiandaca, G., Christiansen, A.V., Kirkegaard, C. and Auken, E. [2015]. Sharp spatially constrained inversion with applications to transient electromagnetic data. Geophysical Prospecting, 63(1), 243–255.