



Contents lists available at ScienceDirect

Computers & Geosciences

journal homepage: www.elsevier.com/locate/cageo

Analysis of numerical differentiation methods applied to time domain electromagnetic (TDEM) geophysical data in the S-layer differential transform

M Combrinck*

University of Pretoria, Pretoria 0002, South Africa

ARTICLE INFO

Article history:

Received 31 July 2006

Received in revised form

21 August 2008

Accepted 28 August 2008

Keywords:

TEM

Numerical differentiation

S-layer

TDEM

ABSTRACT

The S-layer differential transform produces subsurface conductivity-depth images from Time Domain Electromagnetic (TDEM) data. It is a very fast method, but suffers from high noise levels due to the implementation of two consecutive numerical differentiations that are performed in the algorithm. In this paper, twelve numerical differentiation strategies are compared in order to find the most efficient differentiation scheme, specifically for TDEM data and the S-layer differential transform. The twelve strategies are made up through combinations of three differentiation methods, optional smoothing of data and optional resampling of data to equally spaced intervals. Comparisons are made on analytical, synthetic and field data.

© 2009 Elsevier Ltd. All rights reserved.

1. Introduction

1.1. Behaviour of time domain electromagnetic (TDEM) fields

The TDEM method used in geophysical exploration utilises a step function current with a sharp turn-off in a transmitter loop. The sharp turn-off generates a pulse-shaped electromotive force (EMF) in the subsurface. This EMF induces currents to flow in the subsurface. These currents are not connected to a source and will therefore decay (decrease in amplitude) as a function of time. The time rate of decay of the associated secondary magnetic field is measured at a number of time channels. The number of these logarithmically spaced channels varies between 20 and 100, depending on the TDEM system being used. The time-dependent behaviour of this field is well documented in EM literature (Nabighian, 1979; Nabighian and Macnae, 1988), and is divided in an early time, middle time and late time behaviour. In the “late time” the $\frac{dB}{dt}$ field will exhibit either an exponential decay (two- or three-dimensional subsurface features) or power-law decay (one-dimensional or layered earth). The “late-time” is a term dependent not only on the actual time, but also the conductivity of the subsurface. In some surveys, the acquisition parameters are chosen carefully to try and ensure that measurements are taken at late time. Another property of the diffusing currents is that the diffusion velocity is inversely proportional to time.

These very specific properties of being logarithmically sampled and exhibiting power-law or exponential decays make TDEM distinctly unsuitable for the most common numerical differentiation schemes, which are based on functions exhibiting polynomial behaviour, for example the Lagrange formulas (Burden and Faires, 1993). A further very important feature of TDEM data is that the values of interest to the TDEM explorationist are the relatively small-scale perturbations superimposed on the general decay curve. Any process that eliminates these perturbations, or worse still, add perturbations, will lead to erroneous interpretations of the TDEM data.

1.2. Interpretation of TDEM data

One of the applications of the TDEM technique in exploration geophysics is to generate conductivity-depth sections, ultimately leading to an approximate image of the subsurface. The technique is applied to mineral exploration and groundwater exploration.

There are three different ways to derive a conductivity-depth section from TDEM data, namely:

1. Forward modelling.
2. Inversion (or optimization).
3. Mathematical transforms.

Methods 1 and 2 require a priori knowledge of the subsurface to ensure convergence to a plausible solution. Though these methods are being applied in practice, the two- and three-dimensional

* Tel.: +27 82 787 9158; fax: +27 12 362 5219.

E-mail address: magdel.combrinck@up.ac.za

solutions can require several hours of computer time. In lieu of the fact that TDEM systems have been operating from airborne platforms, generating several thousand observations per survey-day, these methods are of limited value in mineral exploration programs where turn around time for the interpretation of geophysical data is critical.

The mathematical transforms as an interpretation technique are by far the most time efficient. The transforms require no knowledge of the subsurface and can therefore be fully automated, making it possible to obtain real-time conductivity-depth sections from the TDEM data. A transform that seems to hold great potential for accurate and real-time interpretation of TDEM data is the S-layer transform, whereby a conductance and depth is calculated for every time channel that is used by the TDEM system. This transform is discussed by Tartaras et al. (2000) and although very elegant in its simplicity, its effectiveness is limited by the fact that a first- and second-order numerical differentiation are required in the algorithm, as discussed in Section 1.3. This is but one of many different schemes for obtaining conductivity-depth sections. Some algorithms, e.g., which are not dependent on the late time approximations are given by Macnae et al. (1991), Wolfgram and Karlik (1995) and Nekut (1987).

1.3. The S-layer differential transform

The S-layer (thin, conductive layer) differential transformation results in two equations that have to be solved for every time channel value obtained at every station. These are summarised by Tartaras et al. (2000) as:

$$S = \frac{16\pi^{1/3}}{(3MA n)^{1/3} \mu_0^{4/3}} \frac{(|V|^{5/3})}{(|V|^{4/3})} \quad (1)$$

$$d = \frac{1}{\mu_0 S} \left(-\frac{4|V|}{|V|} - t \right) \quad (2)$$

where S is the cumulative conductance [S]; d is the depth [m]; M is the magnetic dipole moment of transmitter [Am^2]; A is the receiver area [m^2]; n is the number of turns of receiver coil; t is the time [s]; $|V|$ is the electromotive force of ideal receiver loop [$V = -An(\partial B_z/\partial t)$]; $|V|$ is the time derivative of measured voltage [V] [V/s].

The time derivative of the measured voltage ($|V|$) is the parameter that has to be determined through numerical differentiation. This value is used to calculate both cumulative conductance (S) and depth (d) of an equivalent S-layer, for each time channel and noise generated by differentiation will be consequently introduced into both these numbers. However, in order to find the earth parameter of interest, conductivity (σ), another numerical differentiation has to be performed as

$$\sigma(d) = \frac{\partial S}{\partial d} [\text{Siemens m}^{-1}] \quad (3)$$

which follows from the definition of cumulative conductance. Thus, not only is a new set of errors generated in this step, but all errors from the first differentiation are carried over and enlarged by this second differentiation.

The following features are investigated:

- three numerical differentiation techniques are analysed in terms of the specific nature of TDEM data, as described above,
- the most appropriate use of smoothing filters, and
- the effects of the resampling of data points to equal time intervals.

2. Derivative transforms in logarithmic and semi-logarithmic domains

2.1. Introduction

In the late time, TDEM responses measured over a one-dimensional subsurface can be described as a power-law function (Eq. (4)), with $k = 2.5$ for a half space and $k = 4$ for a thin conductive layer (S-layer). This creates the possibilities of using standard polynomial approximations, but in the logarithmic and semi-logarithmic domains. Consider the following power-law decay:

$$V(t) = At^{-k}, \quad (4)$$

where A is the constant; and k is the decay constant.

If the natural logarithm of this function is taken on both sides, it becomes

$$\ln(V(t)) = \ln A - k \ln(t) \quad (5)$$

Substituting $\ln(V(t))$ with $y(x)$, $\ln(t)$ with x and $-k$ with m , we end up with the equation of a straight line (polynomial function) in the logarithmic domain.

If there is a two- or three-dimensional conductor present, it will contribute to the late-time response in the form of an exponential function of the form shown in Eq. (6).

$$V(t) = Ae^{-kt}, \quad (6)$$

where A is the constant; and k is the decay constant.

If the natural logarithm of this function is taken on both sides, it becomes

$$\ln(V(t)) = \ln A - kt \quad (7)$$

Substituting y for $\ln(V(t))$ and x for t , Eq. (7) again becomes a linear function in the semi-log domain. Eq. (7) is again a linear function. This behaviour only becomes dominant on TDEM sounding data if the conductor is much more conductive than the surrounding host rock or when the host rock response has been removed from the data through subtraction or deconvolution. Raw field data (as on which the S-layer differential transform is normally applied) would mostly exhibit predominantly power-law decay, although it will be distorted to some extent in the presence of conductive layers or finite conductors. In fact, these distortions of the TDEM sounding curves are direct manifestations of the conductivity contrasts which the TDEM practitioners map. In the logarithmic domain TDEM, data appear as functions deviating from a general linear trend and can be locally approximated by second- or third-order polynomials (Fig. 1). Finding a relationship between the derivative of a function and the derivative of the same function in the logarithmic or semi-logarithmic domains would enable us to apply polynomial-based differentiation techniques to data in the domain, where they can be approximated best by polynomial functions.

2.2. Analytical relationship between derivatives in the various domains

The objective is to find an analytical relationship between the derivative of a function in the linear domain ($V(t)$) and the derivatives calculated in the semi-logarithmic ($g'(t)$) and logarithmic domains ($h'(\ln(t))$), respectively. The strategy is simply to apply the chain rule of differentiation to functions presenting the data in different domains.

2.2.1. Semi-logarithmic domain

In the semi-logarithmic domain, the natural logarithms of the V values are taken and treated as a function of (linear) time. It is

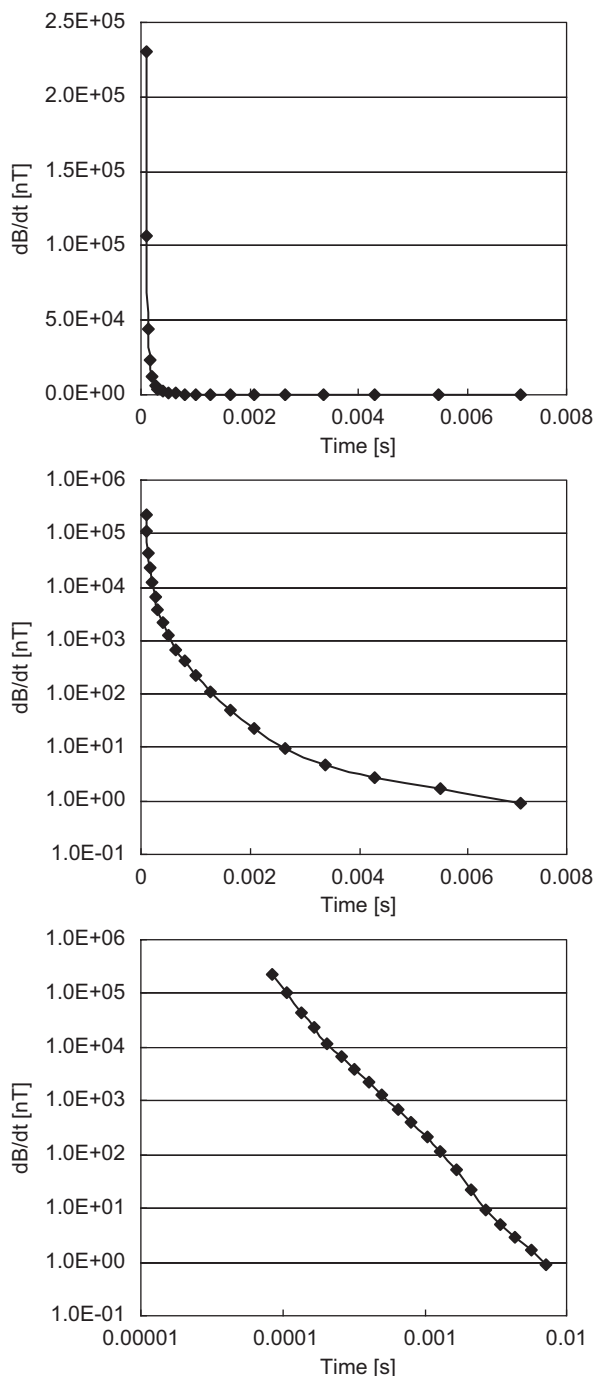


Fig. 1. TDEM ($\partial B_z/\partial t$) data for a 10 Ohm m half space in linear (top), semi-logarithmic (middle) and logarithmic domains (bottom).

presented by the function $g(t) = \ln(V(t))$.

$$g(t) = \ln V(t)$$

therefore,

$$V'(t) = V(t) \cdot g'(t) \quad (8)$$

2.2.2. Logarithmic domain

In the logarithmic domain, the natural logarithms of both the V values and times are taken and the function $h(\ln(t)) = \ln(V(t))$ is plotted against $\ln(t)$. A substitution of variables is made to

simplify the derivation.

$$h[\ln(t)] = \ln[V(t)].$$

$$h[\ln(t)] = \ln[V(e^{\ln(t)})].$$

Now, let $x = \ln(t)$ and $t = e^x$

Then,

$$h(x) = \ln[V(e^x)]$$

$$h'(x) \cdot V(e^x) \cdot \frac{1}{e^x} = V'(e^x)$$

therefore,

$$V'(t) = h'(\ln(t)) \cdot V(t) \cdot \frac{1}{t}. \quad (9)$$

The formulas for transforming the derivatives are straightforward, trivial to apply, and analytically correct (i.e. no truncation error is introduced in the transformation process). This enables us to choose the best domain for performing numerical differentiation based on the nature of data and available differentiation schemes. Synthetic TDEM data for a 10 Ohm m half space are used to illustrate this point. The data were calculated using MARCO software (version 4.0.19, AMIRA Project P223D) with central loop system geometry and the following specifications:

Transmitter loop: $50 \times 50 \text{ m}^2$
 Transmitter current: 5 A
 Receiver loop area: 1 m^2
 Elevation above ground: 0 m

The Lagrange three-point formula was applied to the example data set in the three different domains and transformed back to the linear domain using the appropriate formulae. The results were compared to the analytical derivative of the data and the percentage errors are shown in Fig. 2. The advantage of differentiating in the logarithmic domain is very clear. Also indicated on this graph are derivative values obtained from a two-point power-law formula in the linear domain. These values are exactly equal to the Lagrange 3-point formula applied in the logarithmic domain except at the end points. This formula was derived by fitting a power-law function to a pair of consecutive points and taking the derivative of this interpolating function in exactly the same way as the Lagrange formulae are derived from polynomials.

2.3. Description of three polynomial-based numerical differentiation techniques

The specific numerical differentiation implemented by Tartaras et al. (2000) is described only as "...a numerical differentiation scheme that computes a first-order derivative for the provided TDEM data using polynomial interpolation." This is unfortunately not clear enough to reconstruct with confidence and an exact comparison of this method with the other methods is not included in this paper. Tartaras et al. (2000) also applied smoothing of data "...prior to and following differentiation". This is a commonly used technique applied to reduce noise, but ultimately alters data.

Numerical differentiation is an unstable method, because although the accuracy of the formulas increases with smaller values of "h" (distance between successive nodes), this also causes the round-off error to increase. Divisions by small numbers tend to exaggerate round-off errors and should be avoided if possible. A better way of increasing the accuracy is to use formulas derived from higher order polynomials with the same inter-node distance "h", e.g., the three-point and five-point Lagrange polynomial-based formulas (Burden and Faires, 1993). Three differentiation strategies are compared in this study.

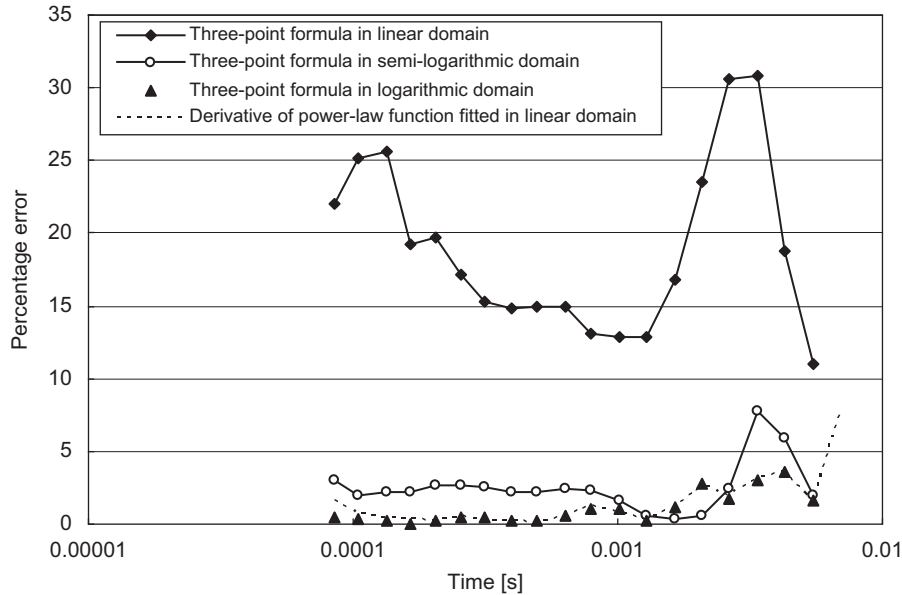


Fig. 2. Percentage errors for differentiation in different domains of TDEM data (10 Ohm m half space).

2.3.1. Lagrange three-point formula

The Lagrange three-point numerical differentiation formula is described in most text books on numerical analysis and a detailed derivation can be found in Burden and Faires (1993). In short, derivative values of discretely sampled data points are approximated by the analytical derivative of a second-order Lagrange polynomial fitted through three consecutive data points, i.e.

$$f'(x) \approx D_x \left[\frac{(x-x_i)(x-x_{i+1})}{(x_{i-1}-x_i)(x_{i-1}-x_{i+1})} f(x_{i-1}) + \frac{(x-x_{i-1})(x-x_{i+1})}{(x_i-x_{i-1})(x_i-x_{i+1})} \times f(x_i) + \frac{(x-x_{i-1})(x-x_i)}{(x_{i+1}-x_{i-1})(x_{i+1}-x_i)} f(x_{i+1}) \right] = D_x [L_{i-1}(x)f(x_{i-1}) + L_i(x)f(x_i) + L_{i+1}(x)f(x_{i+1})] \text{ for } x \in \{x_{i-1}; x_i; x_{i+1}\}. \quad (10)$$

Effectively the method reduces to a sum of weighted function values and for the special case of calculating $f'(x_i)$ (the derivative at the centre point) for equally spaced data points it reduces to

$$f'(x_i) \approx \frac{1}{h} \left(-\frac{1}{2} \cdot f(x_{i-1}) + 0 \cdot f(x_i) + \frac{1}{2} \cdot f(x_{i+1}) \right) \quad (11)$$

where $h = (x_i - x_{i-1})$.

End points of equally spaced values have weights of $(-1.5; 2; -0.5)$ and $(0.5; -2; 1.5)$ for first and last points, respectively; while the weights for unequally spaced data points have to be calculated for every point using Eq. (10). Errors (or noise) in the calculated derivative are dependent on sampling interval, errors in data values (small errors in data are enhanced in derivative calculations), computer round-off effects and how accurately the causative function can be approximated locally by a second-order polynomial.

2.3.2. Derivative of cubic spline interpolated function (referred to as “cubic spline derivative method”)

The Lagrange three-point method will give analytically correct results for functions of order one or two. This approximation can be extended to a wider range of functions by increasing the degree of the Lagrange approximating polynomial. However, this does not always improve the final result. Lagrange interpolation requires that the approximated function values on the measured positions be very similar to the discrete data points and this introduces

unwanted oscillations which are especially detrimental to any consecutive derivative calculations. An alternative is to fit higher order functions using the least squares errors approach. However, low-order polynomials will result in loss of high-frequency information and high-degree polynomials still suffer from oscillations. TDEM is based on the diffusion of electrical currents into the earth and this is a smooth process, never oscillating in nature, and therefore not suited to this type of interpolation. Furthermore, although TDEM data locally approximate polynomial functions in the logarithmic domain, they cannot in general be represented by the same order polynomial on all time channels. A better strategy to follow in this case is piecewise polynomial approximation. One method to interpolate data under these conditions is known as the cubic spline method that fits third-order polynomials to each consecutive pair of data points requiring the function values to be equal to the measured data values as well as continuity of the first- and second-order derivatives ensuring smoothness of the function. As with the Lagrange three-point formula, an approximation to the derivative of the sampled function is obtained with the analytical derivative of the third-order cubic spline polynomials at every point. The cubic spline method is computationally more intensive than the Lagrange three-point formula, but it can also be used to resample data to equally spaced intervals (which simplifies smoothing of data), if required. The free or natural boundary conditions were invoked in this study (assume that the second derivatives of the end points are zero), as the information required for clamped boundary points are not available.

2.3.3. Differentiation as the inverse of integration

In contrast to numerical differentiation, numerical integration (the inverse of differentiation) is a very stable method and not particularly sensitive to noise. The same relationship holds for the downward (unstable) and upward (stable) continuation filters used in the processing of potential field data. Cooper (2004) introduced the “inverse of upward continuation” as a more stable downward continuation filter and also discussed how differentiation can be implemented as an “inverse of integration” filter. This can be formulated as follows. Define $f(x_i)$ as the data points to be differentiated, $f'(x_i)$ as the required derivative values and \mathbf{A} an

operator of integration. Then,

$$\mathbf{A}f'(x_i) = f(x_i), \quad i = 0, \dots, n. \quad (12)$$

For the case where \mathbf{A} is singular, not square or a non-linear operator, Eq. (12) must be solved with a method such as generalised least squares inversion or other equivalent techniques (Cooper, 2004). Disadvantages of solving systems of equations in this way are:

- need for a good initial model,
- no guarantee of convergence,
- no guarantee of uniqueness of the solution, and
- mathematical intensiveness.

However, if \mathbf{A} can be formulated to be a non-singular, square and linear matrix, Eq. (12) becomes a system of linear equations with the same number of unknowns as equations (implying the existence of an inverse operator and a unique solution) which can be solved by multiplying both sides with \mathbf{A}^{-1} .

$$f'(x_i) = \mathbf{A}^{-1} \cdot f(x_i), \quad i = 0, \dots, n. \quad (13)$$

Alternatively, Eq. (12) can be solved for the $f(x_i)$ by applying Gaussian elimination with backward substitution. Depending on whether \mathbf{A} and \mathbf{A}^{-1} will be calculated only once (for a fixed number of equally spaced data points) or whether it will have to be recalculated (for unequally spaced data points) for every sounding data set, the most efficient of the two methods are chosen. For 20 data points, it will require 3060 multiplications/divisions and 2850 additions/subtractions to solve equation 4.8, using Gauss elimination with backward substitution. The equivalent numbers for calculating \mathbf{A}^{-1} are 10,660 and 10,070. Thereafter, 400 multiplications/divisions and 380 additions/subtractions are required to solve for the $f(x_i)$.

2.3.4. Construction of the integration matrix

Integration and differentiation can be done very simply in the frequency domain by respective division or multiplication with the factor ($i\omega$), where ω is the angular frequency and $i = \sqrt{-1}$. However, the process of Fourier transformation on discrete data points adds noise to data and time domain calculations, although more complex and time consuming, are preferred (especially in small data sets) to obtain cleaner data. The first point that has to be addressed in any integration procedure is that of the integration constant, formulated as

$$\int f'(x) dx = f(x) + c, \quad (14)$$

where c can take on any value (implying an infinite number of solutions). An easy solution is to use definite integrals instead, so that

$$\int_a^b f'(x) dx = f(b) - f(a). \quad (15)$$

From Eq. (12), the system of equations that has to be solved now reduces to a general form as shown in Eq. (16).

$$\begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \\ a_{n1} & \dots & \dots & a_{nn} \end{bmatrix} \begin{bmatrix} f'(x_1) \\ \dots \\ \dots \\ f'(x_n) \end{bmatrix} = \begin{bmatrix} f(x_b) - f(x_a) \\ \dots \\ \dots \\ f(x_n) - f(x_{n-1}) \end{bmatrix}. \quad (16)$$

The elements of \mathbf{A} (a_{ii}) must now be defined so as to produce numerical integration of the $f'(x_i)$. Simpson's rule (Burden and Faires, 1993) is a standard and very effective method suitable for polynomial

functions. For equally spaced points, it is given by

$$\int_{x_{i-1}}^{x_{i+1}} f'(x) dx = f(x_{i+1}) - f(x_{i-1}) \approx \frac{h}{3} [f'(x_{i-1}) + 4 \cdot f'(x_i) + f'(x_{i+1})], \quad (17)$$

where $h = (x_{i+1} - x_i) = (x_i - x_{i-1})$.

There is one drawback; this formula can only be used to generate $n-2$ equations. In order to have a square matrix and unique solutions, we need two more independent equations. This is done by deriving two end-point Simpson's rule formulas. Normal Simpson's rule can be derived by fitting a second-order polynomial through three consecutive points and integrating this polynomial from the first to the third point. The same strategy is followed for the end points, except that we integrate from the first to the second point only at the start of the sequence and from the second to the third point only at the end. This results in the following two equations:

$$\int_{x_1}^{x_2} f'(x) dx = f(x_2) - f(x_1) \approx \frac{h}{3} \left[\frac{5}{4} f'(x_1) + 2 \cdot f'(x_2) - \frac{1}{4} f'(x_2) \right], \quad (18)$$

and

$$\int_{x_{n-1}}^{x_n} f'(x) dx = f(x_n) - f(x_{n-1}) \approx \frac{h}{3} \left[-\frac{1}{4} f'(x_{n-2}) + 2 \cdot f'(x_{n-1}) + \frac{5}{4} f'(x_n) \right], \quad (19)$$

where $h = (x_{i+1} - x_i)$.

For equally spaced points, Eq. (16) can thus be rewritten as

$$\frac{h}{3} \begin{bmatrix} \frac{5}{4} & 2 & -\frac{1}{4} & 0 & \dots & 0 \\ 1 & 4 & 1 & 0 & \dots & 0 \\ 0 & 1 & 4 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & 4 & 1 & \dots & 0 \\ 0 & \dots & \dots & \dots & \dots & \dots & 0 \\ 0 & \dots & \dots & 0 & 1 & 4 & 1 \\ 0 & \dots & \dots & 0 & -\frac{1}{4} & 2 & \frac{5}{4} \end{bmatrix} \begin{bmatrix} f'(x_1) \\ f'(x_2) \\ \dots \\ f'(x_{n-1}) \\ f'(x_n) \end{bmatrix} = \begin{bmatrix} f(x_2) - f(x_1) \\ f(x_3) - f(x_1) \\ f(x_4) - f(x_2) \\ \dots \\ f(x_{n-1}) - f(x_{n-3}) \\ f(x_n) - f(x_{n-2}) \\ f(x_n) - f(x_{n-1}) \end{bmatrix} \quad (20)$$

or

$$\begin{bmatrix} 2\left(\frac{1}{2}\right) & 4 & -\frac{1}{2} & 0 & \dots & 0 \\ 1 & 4 & 1 & 0 & \dots & 0 \\ 0 & 1 & 4 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & 4 & 1 & \dots & 0 \\ 0 & \dots & \dots & \dots & \dots & \dots & 0 \\ 0 & \dots & \dots & 0 & 1 & 4 & 1 \\ 0 & \dots & \dots & 0 & -\frac{1}{2} & 4 & 2\left(\frac{1}{2}\right) \end{bmatrix} \begin{bmatrix} f'(x_1) \\ f'(x_2) \\ \dots \\ f'(x_{n-1}) \\ f'(x_n) \end{bmatrix} = \frac{3}{h} \begin{bmatrix} 2[f(x_2) - f(x_1)] \\ f(x_3) - f(x_1) \\ f(x_4) - f(x_2) \\ \dots \\ f(x_{n-1}) - f(x_{n-3}) \\ f(x_n) - f(x_{n-2}) \\ 2[f(x_n) - f(x_{n-1})] \end{bmatrix}. \quad (21)$$

The matrix \mathbf{A} is square and symmetrical with the highest values on the diagonal as pivot elements, except for the first and last rows. The inverse of this matrix, \mathbf{A}^{-1} , depends on the number of data points, n , and can be calculated using any standard algorithm. In the general case where data points are unequally spaced, the weights (a_{ii}) are different for every data set and both the matrix and its inverse have to be recalculated for every sounding.

3. Smoothing and resampling of data points

Before testing of the three methods discussed above, one needs to consider the process and effect of smoothing of the data at

various stages of the differentiation process. Smoothing of data before and after the numerical differentiation is an accepted method of reducing the noise in data (Tartaras et al., 2000). However, any smoothing applied to the data will change the data and influence the final interpretation. In the S-layer differential transform algorithm, there are four possible opportunities to smooth data and the question arises which of these will optimise the algorithm with minimum alteration of the data.

The last factor taken into consideration is the spacing of data points. The time channels at which data are measured are almost equally spaced in the logarithmic domain. The calculated depths (to which the second differentiation has to be performed) will have intervals dependent on the conductivity of the subsurface. The advantage of resampling data to equal intervals is that the weights (and inverse matrices) used for differentiation and smoothing only have to be calculated once. Applying the transform to unequally spaced data points would have a minimum impact on the measured data but is much more time intensive.

The method used for smoothing is the three-point moving average filter for equally spaced points and a weighted extension for unequally spaced points (Eq. (22)).

$$f(x_2)_{\text{Weighted Moving Average}} = \left[\frac{2 \cdot (x_3 - x_2)}{(x_3 - x_1)} \cdot f(x_1) + f(x_2) + \frac{2 \cdot (x_2 - x_1)}{(x_3 - x_1)} \cdot f(x_3) \right] / 3 \quad (22)$$

4. Comparison of differentiation methods applied to synthetic data

4.1. First-order numerical differentiations compared to the analytical derivative of a function

A TDEM decay curve is simulated by an analytical function (Eq. (23)), defined as the sum of exponential and power-law terms. (There are no analytical formulations of complete layered earth TDEM responses in closed form; therefore an “equivalent” analytical function is used.)

$$f(x) = 0.000001 \cdot t^{-5/2} + 0.0000000001 \cdot t^{-4} + 0.00000001 \cdot t^{-3} + 2000 \cdot e^{-2500t} + 12000 \cdot e^{-8000t} + 10 \cdot e^{-400t} \quad (23)$$

The first-order analytical derivative of this function is compared to the following numerical scenarios:

1. All three numerical differentiation algorithms are applied to unequally spaced data points and without smoothing of the data. The results are shown in Fig. 3(a).
2. Same as Eq. (1), but with smoothing of the function before the derivatives were calculated. The results are shown in Fig. 3(b).
3. The same as Eq. (1), but with the unequally spaced data points resampled to equally spaced data points using the cubic spline interpolation method (Fig. 3(c)).
4. The results in Fig. 3(d) are the equally spaced data points' equivalent of 3(b) (Fig. 4).

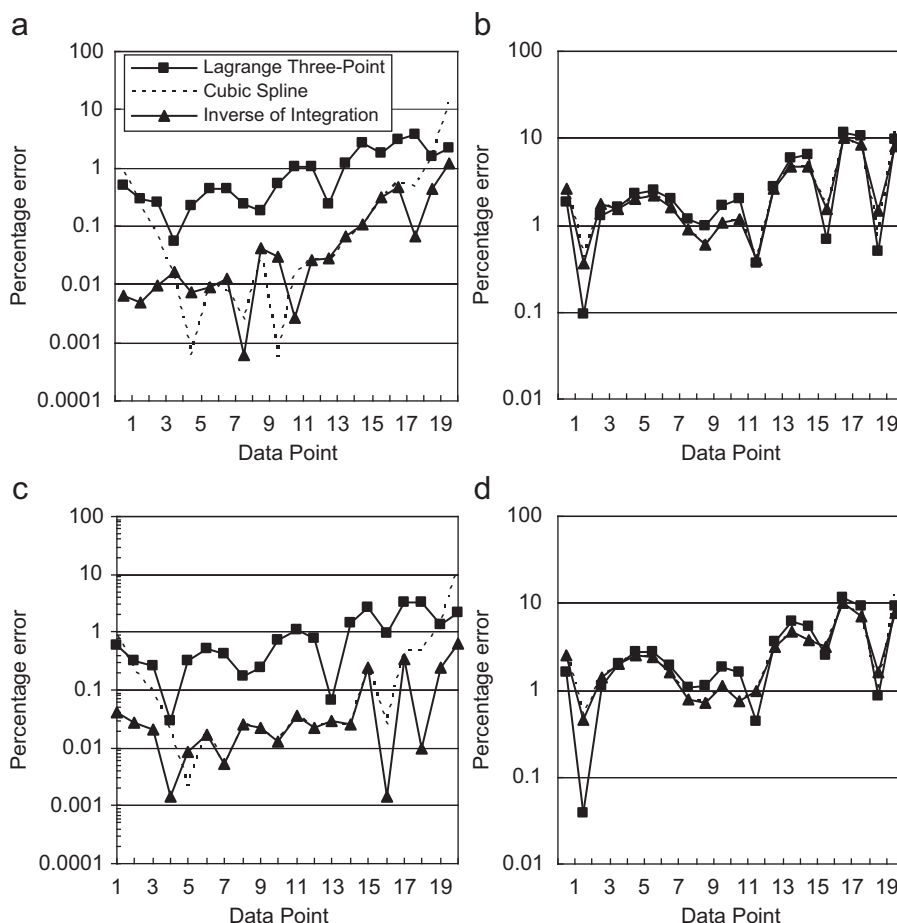


Fig. 3. Comparison of three numerical differentiation methods applied on (a) unequally spaced data points, without smoothing of data, (b) unequally spaced data points, with smoothing of data, (c) equally spaced data points, without smoothing of data and (d) equally spaced data points with smoothing of data.

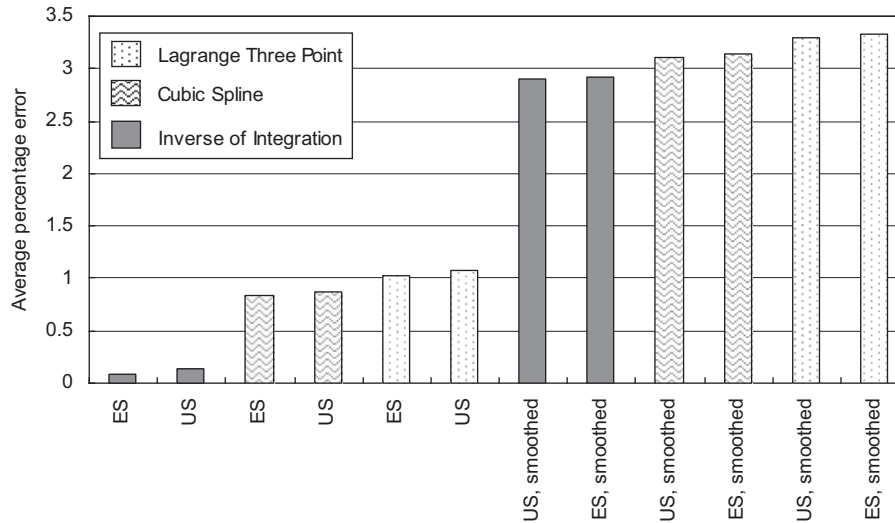


Fig. 4. A summary (in ascending order) of average error over twenty data points for each alternative in Fig. 3 (ES: equal spacing, US: unequal spacing).

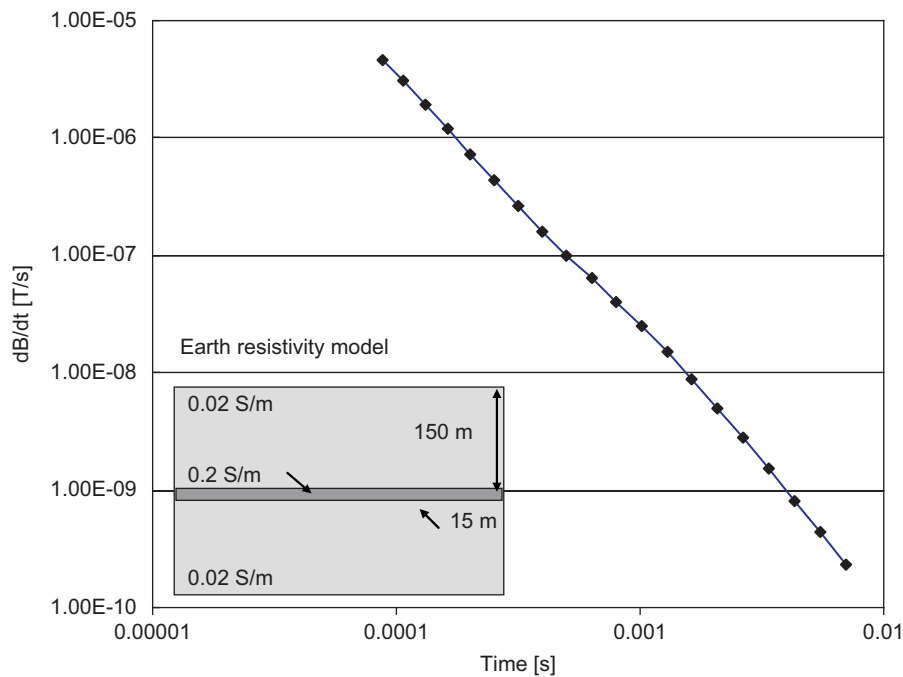


Fig. 5. Synthetic data and model for three-layer earth.

A summary of the average errors (absolute values) for every alternative is given in Fig. 5.

From these comparisons, the following are concluded:

- The inversion method gives the smallest average error, followed by the cubic spline and three-point methods.
- For unsmoothed data, the equally spaced points slightly outperform the unequally spaced points, but the reverse is true for smoothed data.
- Smoothing of data seems to be the largest contributing factor to errors in noise-free data.

4.2. S-layer differential transform applied to synthetic data

A set of data was calculated for a three-layered earth using MARCO software developed by AMIRA. The model and data are shown in Fig. 5. A flow diagram for the numerical calculation of the S-layer transform is the following:

1. Input $|V|$ values ($= \partial B_z / \partial t$ if normalised to receiver area of 1 m^2).
2. Calculate $|V|'$, i.e. $\partial^2 B_z / \partial t^2$.
3. Calculate S from Eq. (1).
4. Calculate d from Eq. (2).
5. Calculate $\partial S / \partial d = \sigma$.

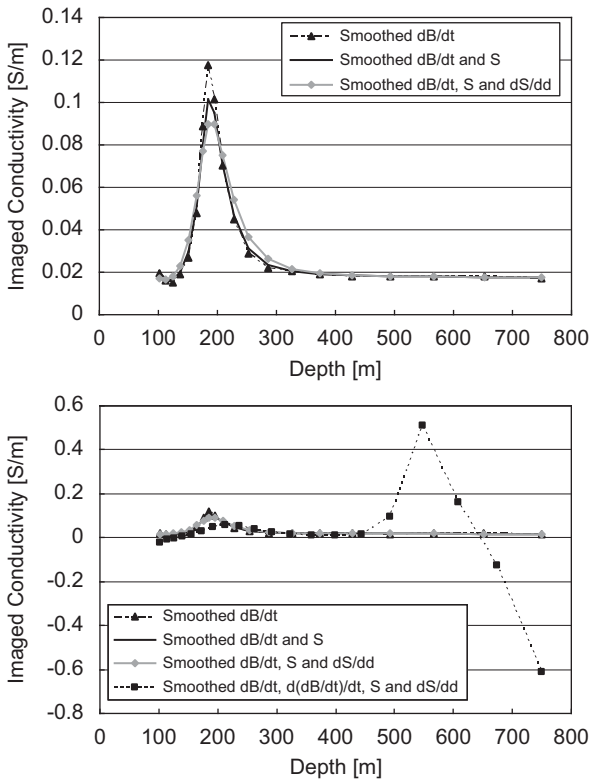


Fig. 6. Effects of smoothing at different points in S-layer transform algorithm. Bottom panel shows same three profiles as in top panel with an additional profile, where $d(dB/dt)/dt$ data were also smoothed. (Two panels are used to accommodate difference in scale.)

Numerical differentiations are performed in steps 2 and 5. Smoothing filters are applied at various stages in the algorithm culminating up to the strategy proposed by Tartaras et al. which is smoothing before and after each differentiation. Fig. 6 (top) shows the results of three variations:

- smoothing of $|V|$ before step 1,
- smoothing of $|V|$ before step 1 and S after step 3, and
- smoothing of $|V|$ before step 1, S after step 3 and σ after step 5.

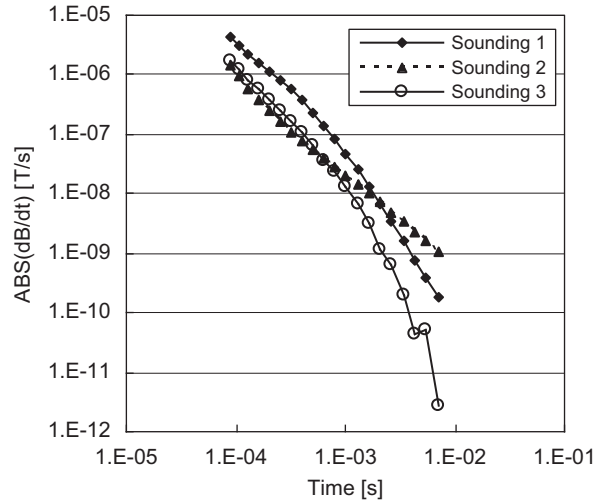


Fig. 8. Three field data soundings.

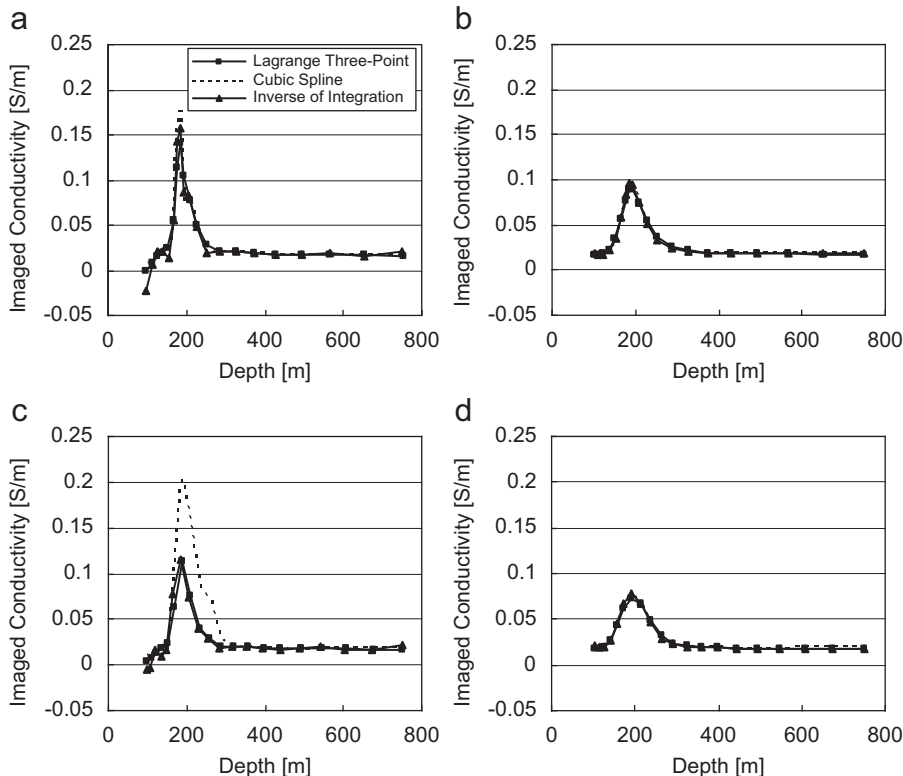


Fig. 7. Comparison of S-layer differential transform results for three numerical differentiation methods applied on (a) unequally spaced data points, without smoothing of data, (b) unequally spaced data points, with smoothing of data, (c) equally spaced data points, without smoothing of data and (d) equally spaced data points with smoothing of data.

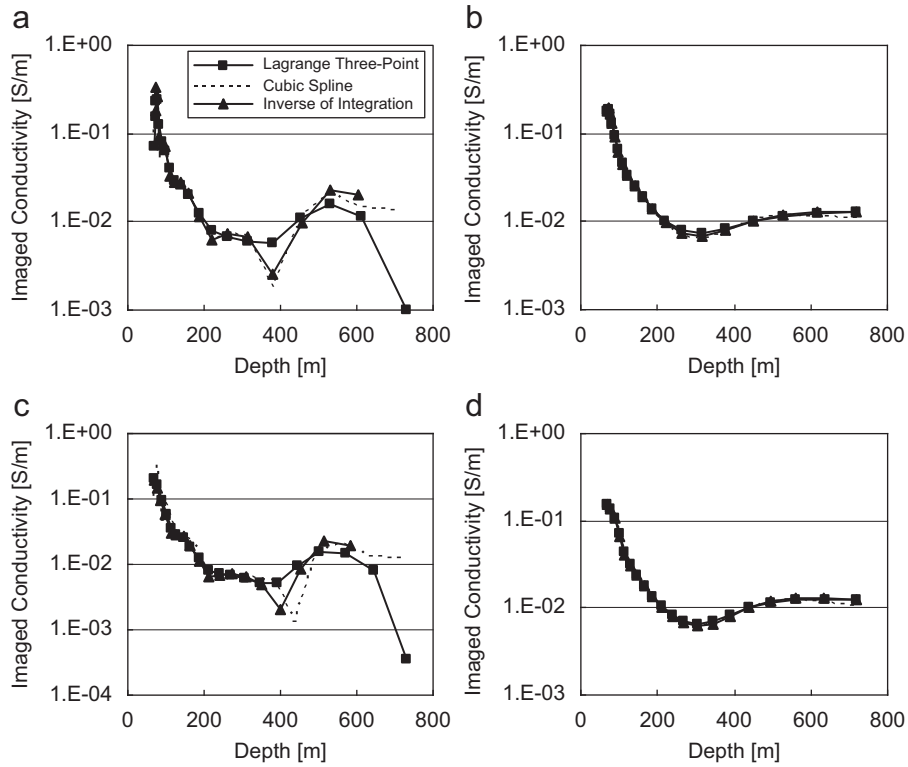


Fig. 9. S-layer differential transform results for Sounding 1. Panels are the same as in Fig. 7.

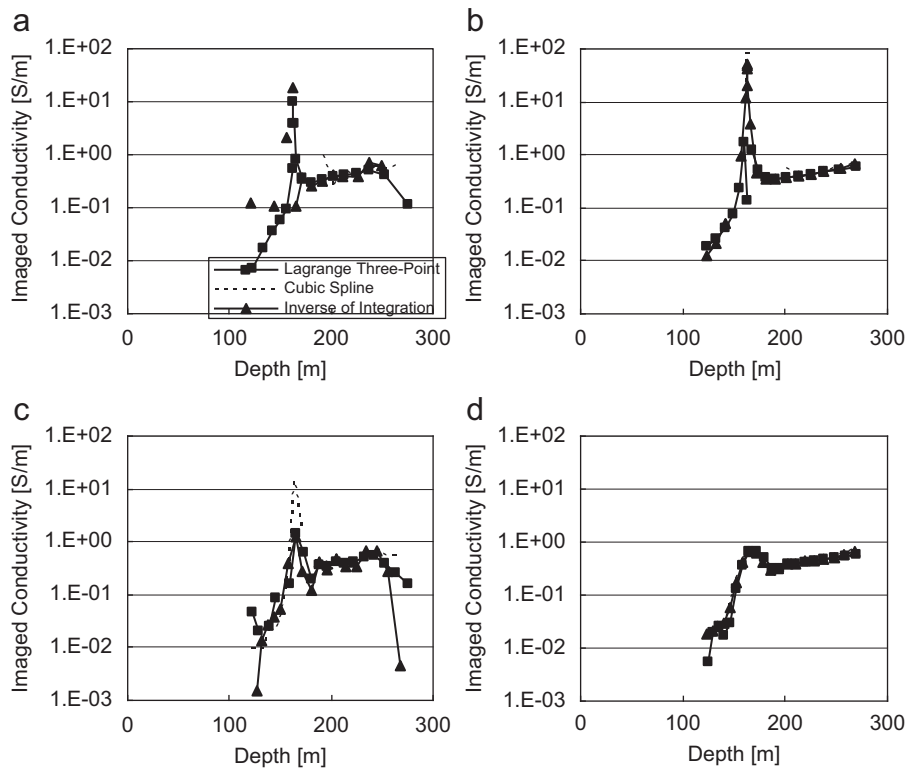


Fig. 10. S-layer differential transform results for Sounding 2. Panels are the same as in Fig. 7.

As can be expected, the successive smoothing slightly lowers the maximum amplitude and widens the response curve. However, it is critical *not* to smooth $|V|$ (Fig. 6, bottom), which introduces extensive noise. This is due to the ratios of $|V|$ and $|V|'$ used to calculate S and d .

Fig. 7 shows the output of the S-layer transform (imaged conductivity versus depth) for twelve different combinations of differentiation method, smoothing and spacing of data. (Smoothing of data here implies smoothing $|V|$, S and σ .) Both smoothing and resampling of data reduce the maxima of

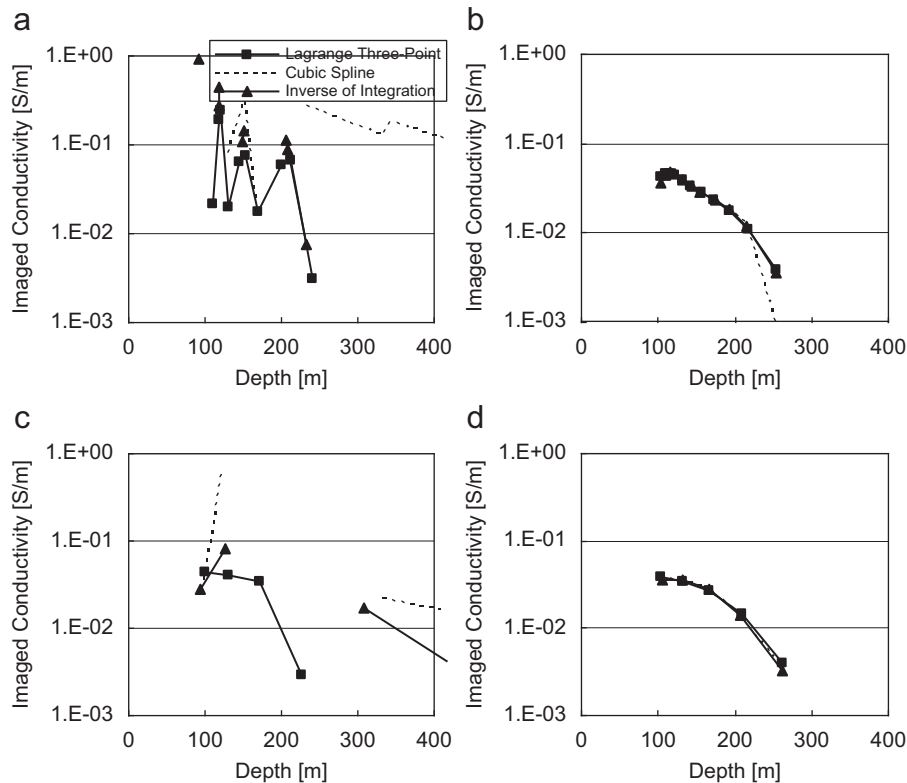


Fig. 11. S-layer differential transform results for Sounding 3. Panels are the same as in Fig. 7.

responses leading to an underestimation of the conductivity contrast.

5. Comparison of differentiation methods on field data

Three soundings are chosen from a field data set (courtesy of Kumba Resources) to represent both clean and noisy data (Fig. 8) and all twelve processing combinations are applied to these soundings. The data were acquired with a Geonics Protem system in the central loop sounding configuration utilising a $100 \times 100 \text{ m}^2$ transmitter loop, 15 A current, 25 Hz base frequency and effective receiver area of 100 m^2 . Figs. 9–11 illustrate the results with (a) unequally spaced data points without smoothing of data, (b) unequally spaced data points with smoothing of data, (c) equally spaced data points without smoothing of data and (d) equally spaced data points, with smoothing of data in each case. Sounding 1 (Fig. 9) would be considered “noise-free” in that none of the data points would be discarded in the manual interpretation of data. However, comparing the smoothed and unsmoothed results, it is clear that it does contain some noise which can be filtered out successfully with the weighted moving average filter. The unequally spaced points appear less sensitive to the noise and after smoothing there is no noticeable difference between the three different methods of differentiation. (With field data it is not possible to calculate percentage errors and smoothness of conductivity-depth curves are used as an indication of accuracy based on the smoothly dissipative nature of TDEM currents.) Sounding 2 contains slightly more noise. The most noticeable effect is how the resampling to equally spaced data points acts as an additional smoothing filter. For Sounding 3 (Fig. 11), a data set was chosen with noise already visible from channel 14 onwards. Here, the advantage of smoothing is very clear and the indication is that this method can be applied successfully to field data. Resampling data points to equal intervals in this case reduces the

number of significant (smooth) data points to five compared to the twelve of the unequally spaced data, although the effective depth coverage remains the same. The unequally spaced points thus, have better vertical resolution.

6. Concluding remarks

The S-layer differential transform (and other imaging techniques) are useful tools in automated interpretation of TDEM data, as it is fast and does not require a starting model. It is not as accurate as inversion methods, but remains useful to provide starting models for these more time-consuming procedures. As numerical differentiation is very unstable, it is important to take great care when applying this operator in deriving the S-layer transform. When working with field data, data have to be smoothed but with cognizance of the influence on the final result. Also, even though time-effectiveness can be increased in using algorithms for equally spaced data points, it produces an underestimation of conductivity values and loss of resolution when applied to the S-layer transform. As for the method of differentiation, the “inverse of integration” performed best on synthetic data, but it is arguable whether the 0.4% increase in accuracy is worth the additional time compared to the Lagrange three-point formula. The S-layer transform remains very fast (less than five seconds on a PC for 1500 soundings), no matter which differentiation method is chosen.

Acknowledgements

The author gratefully acknowledges the comments and suggestions of Prof. W.J. Botha as well as Kumba Resources for the use of MARCO software and field data sets.

References

- Burden, R.L., Faires, J.D., 1993. Numerical Analysis, fifth ed. Prindle, Weber & Schmidt, Publishing Company, Boston, p. 768.
- Cooper, G.R.J., 2004. The stable downward continuation of potential field data. *Exploration Geophysics* 35, 260–265.
- Macnae, J.C., Smith, R., Polzer, B.D., Lamontagne, Y., Klinkert, P.S., 1991. Conductivity-depth imaging of airborne electromagnetic step-response data. *Geophysics* 56, 102–114.
- Nabighian, M.N., 1979. Quasi-static response of a conducting half-space: an approximate representation. *Geophysics* 44, 1700–1705.
- Nabighian, M.N., Macnae, J.C., 1988. TDEM prospecting methods. In: Nabighian, M.N. (Ed.), *Electromagnetic Methods in Applied Geophysics 1, Theory*. Society of Exploration Geophysicists, Tulsa, pp. 427–520.
- Nekut, A.G., 1987. Direct inversion of time-domain electromagnetic data (short note). *Geophysics* 52, 1431–1435.
- Tartaras, E., Zhdanov, M.S., Wada, K., Saito, A., Hara, T., 2000. Fast imaging of TDEM data based on S-inversion. *Journal of Applied Geophysics* 43, 15–32.
- Wolfgram, P., Karlik, G., 1995. Conductivity-depth transform of GEOTEM data. *Exploration Geophysics* 26, 179–185.