Chapter 1

Fundamentals: The single site method

The term "single site" refers to the fact that only data of one magnetotelluric station is required for this simplest of all processing techniques. I will use it in the following to demonstrate in detail how the solution of a transfer function equation is obtained (section 1.1). Furthermore, the way from measured time series to spectra providing the input for the obtained solution will be documented for the code mainly applied in this work (section 1.2). Third, there will be discussed how different kinds of noise can affect transfer functions in section 1.3.

1.1 The impedance equation and its solution

In general, transfer functions are operators that, when applied to a measured quantity, yield another measured quantity. In magnetotellurics they are complex-valued functions of the frequency ω depending also on the station's position.
In case of a single site, two transfer functions can be determined: the impedance $\mathbf{Z}$, a tensor of the shape

$$\mathbf{Z} = \begin{pmatrix} Z_{xx} & Z_{xy} \\ Z_{yx} & Z_{yy} \end{pmatrix} \tag{1.1}$$

and the tipper with the elements (T_x, T_y).

The impedance transfers the horizontal magnetic field $\vec{B_h} = \begin{pmatrix} B_x & B_y \end{pmatrix}^T$ (^{T} denotes the transpose) to the horizontal electric field $\vec{E} = \begin{pmatrix} E_x & E_y \end{pmatrix}^T$:

$$\vec{E} = \mathbf{Z}\vec{B_h}, \tag{1.2}$$

or

$$E_x = Z_{xx}B_x + Z_{xy}B_y, \tag{1.3}$$

$$E_y = Z_{yx}B_x + Z_{yy}B_y \tag{1.4}$$

respectively, when written in detail. Impedances are usually presented as curves of apparent resistivity

$$\rho_a = \frac{\mu_0}{2\pi} |Z_{xx}|^2 T \tag{1.5}$$

and phase

$$\phi = arctan \frac{\mathfrak{Im} Z_{xx}}{\mathfrak{Re} Z_{xx}} \tag{1.6}$$

over the period $T = 2\pi/\omega$. This holds for the other components of $\mathbf{Z}$ as well. Since in magnetotellurics, $\vec{E}$ is measured in mV/km and $\vec{B}$ in nT, the unit of our "impedance"is km/s. In order to get Ω, the proper impedance unit, one would have to construct the impedance via $\vec{H} = \mu_0\mu\vec{B}$ instead of just via $\vec{B}$. μ is thereby a material constant and $\mu_0 = 4\pi * 10^{-7} Vs/Am$. The units of ω, T, and ρ_a are as expected Hz, s, and Ωm.

The tipper is the transfer function from horizontal to vertical magnetic field

$$B_z = T_x B_x + T_y B_y, \tag{1.7}$$

and the induction arrow constructed from its elements to display it

$$arrow = \begin{pmatrix} T_x \\ T_y \end{pmatrix} \tag{1.8}$$

shows the horizontal direction connected to the maximal vertical magnetic field.

The equations (1.3, 1.4, 1.7) of these transfer functions have a very similar shape and are solved in an analogous way. This will be shown for the example of equation 1.3. Along general lines, I am following the argumentation of Buttkus [1998] here.

$$E_x = Z_{xx}B_x + Z_{xy}B_y \tag{1.9}$$

is an equation with two unknown variables[1] (Z_{xx} and Z_{xy}), thus impossible to solve just in an algebraic way by rearranging its terms. Instead of this, one can take advantage of the means of linear regression since there is the possibility to get not only one, but many data examples holding this equation. For this purpose, one divides the measured time series into N segments and calculates the Fourier coefficients $E_{xi}, B_{xi}, B_{yi}, i = 1..N$ at the given frequency for each of them (see section 1.2 for details about this steps). Permitting errors δE_{xi} which reflect the parts of E_{xi} that are not includable in the transfer functions, one obtains the equation

$$\begin{pmatrix} E_{x1} \\ E_{x2} \\ \vdots \\ E_{xN} \end{pmatrix} = Z_{xx} \begin{pmatrix} B_{x1} \\ B_{x2} \\ \vdots \\ B_{xN} \end{pmatrix} + Z_{xy} \begin{pmatrix} B_{y1} \\ B_{y2} \\ \vdots \\ B_{yN} \end{pmatrix} + \begin{pmatrix} \delta E_{x1} \\ \delta E_{x2} \\ \vdots \\ \delta E_{xN} \end{pmatrix}. \tag{1.10}$$

Demanding now that Z_{xx} and Z_{xy} have to assume a shape that prompts the square of all errors together to become minimum[2],

$$\sum_{i=1}^{N} \delta E_{xi} \delta E_{xi}^{*} = min \tag{1.11}$$

(a superscript * denotes the conjugate complex value), one has already set up the conditions that have to be fulfilled to solve the equation for the transfer functions:

$$\frac{\partial \left(\sum_{i=1}^{N} \delta E_{xi} \delta E_{xi}^{*} \right)}{\partial Z_{xx}} = 0 \tag{1.12}$$

and

$$\frac{\partial \left(\sum_{i=1}^{N} \delta E_{xi} \delta E_{xi}^{*} \right)}{\partial Z_{xy}} = 0. \tag{1.13}$$

Note that from equation 1.10 follows

$$\delta E_{xi} \delta E_{xi}^{*} = (E_{xi} - Z_{xx} B_{xi} - Z_{xy} B_{yi})(E_{xi}^{*} - Z_{xx}^{*} B_{xi}^{*} - Z_{xy}^{*} B_{yi}^{*}) \tag{1.14}$$

and that the derivative with respect to a complex function means, e.g.

$$\frac{\partial \left(\delta E_{xi} \delta E_{xi}^{*} \right)}{\partial Z_{xx}} = \frac{\partial \left(\delta E_{xi} \delta E_{xi}^{*} \right)}{\partial \mathfrak{Re} Z_{xx}} + i \frac{\partial \left(\delta E_{xi} \delta E_{xi}^{*} \right)}{\partial \mathfrak{Im} Z_{xx}}. \tag{1.15}$$

With 1.14 and 1.15 applied to equations 1.12 and 1.13 one obtains

$$\sum_{i=1}^{N} E_x B_x^{*} = Z_{xx} \sum_{i=1}^{N} B_x B_x^{*} + Z_{xy} \sum_{i=1}^{N} B_y B_x^{*} \tag{1.16}$$

[1] We refer to this as a "bivariate problem" in this context.

[2] For this condition the method is called "Least Square".

and

$$\sum_{i=1}^{N} E_x B_y^* = Z_{xx} \sum_{i=1}^{N} B_x B_y^* + Z_{xy} \sum_{i=1}^{N} B_y B_y^*. \tag{1.17}$$

From this follows

$$Z_{xx} = \frac{\sum_{i=1}^{N} E_{xi} B_{xi}^* \sum_{i=1}^{N} B_{yi} B_{yi}^* - \sum_{i=1}^{N} E_{xi} B_{yi}^* \sum_{i=1}^{N} B_{yi} B_{xi}^*}{\sum_{i=1}^{N} B_{xi} B_{xi}^* \sum_{i=1}^{N} B_{yi} B_{yi}^* - \sum_{i=1}^{N} B_{xi} B_{yi}^* \sum_{i=1}^{N} B_{yi} B_{xi}^*} \tag{1.18}$$

and

$$Z_{xy} = \frac{\sum_{i=1}^{N} E_{xi} B_{yi}^* \sum_{i=1}^{N} B_{xi} B_{xi}^* - \sum_{i=1}^{N} E_{xi} B_{xi}^* \sum_{i=1}^{N} B_{xi} B_{yi}^*}{\sum_{i=1}^{N} B_{xi} B_{xi}^* \sum_{i=1}^{N} B_{yi} B_{yi}^* - \sum_{i=1}^{N} B_{xi} B_{yi}^* \sum_{i=1}^{N} B_{yi} B_{xi}^*} \tag{1.19}$$

So it is known how to estimate Z_{xx} and Z_{xy} now.

It is shorter, more elegant, and for programming applications more practical to write this formula completely with vectors and matrices following Oettinger et al. [2001]. First, there are defined

$$\vec{E_x} = \begin{pmatrix} E_{x1} \\ E_{x2} \\ \vdots \\ E_{xN} \end{pmatrix} \tag{1.20}$$

and

$$\mathbf{B} = \begin{pmatrix} B_{x1} & B_{y1} \\ B_{x2} & B_{y2} \\ \vdots & \vdots \\ B_{xN} & B_{yN} \end{pmatrix}. \tag{1.21}$$

Then there can be written simply

$$\left(\vec{Z_x}\right)^T = \left(\mathbf{B}^\dagger \mathbf{B}\right)^{-1} \left(\vec{E_x}^\dagger \mathbf{B}\right) \tag{1.22}$$

where a † indicates the Hermitean transpose. The detailed notation (where $\sum_{i=1}^{N}$ is abbreviated with $\sum$ for reasons of space)

$$\begin{pmatrix} Z_{xx} \\ Z_{xy} \end{pmatrix} = \begin{pmatrix} \sum B_{xi} B_{xi}^* & \sum B_{yi} B_{xi}^* \\ \sum B_{xi} B_{yi}^* & \sum B_{yi} B_{yi}^* \end{pmatrix}^{-1} \begin{pmatrix} \sum E_{xi}^* B_{xi} \\ \sum E_{xi}^* B_{yi} \end{pmatrix} =$$

$$\frac{1}{\sum B_{xi}B_{xi}^{*}\sum B_{yi}B_{yi}^{*}-\sum B_{xi}B_{yi}^{*}\sum B_{yi}B_{xi}^{*}}\begin{pmatrix}\sum E_{xi}^{*}B_{xi}\sum B_{yi}B_{yi}^{*}-\sum E_{xi}^{*}B_{yi}\sum B_{yi}B_{xi}^{*}\\ \sum E_{xi}^{*}B_{yi}\sum B_{xi}B_{xi}^{*}-\sum E_{xi}^{*}B_{xi}\sum B_{xi}B_{yi}^{*}\end{pmatrix}$$

shows that 1.22 is equivalent to equations 1.18 and 1.19.

Analogically hold

$$\left(\vec{Z_y}\right)^T=\begin{pmatrix} Z_{yx} & Z_{yy}\end{pmatrix}=\left(\mathbf{B}^{\dagger}\mathbf{B}\right)^{-1}\left(\vec{E_y}^{\dagger}\mathbf{B}\right) \tag{1.23}$$

with

$$\vec{E_y}=(E_{y1},E_{y2}\ldots E_{yN})^T \tag{1.24}$$

for the rest of the elements of the impedance tensor and

$$\begin{pmatrix} T_x & T_y\end{pmatrix}=\left(\mathbf{B}^{\dagger}\mathbf{B}\right)^{-1}\left(\vec{B_z}^{\dagger}\mathbf{B}\right) \tag{1.25}$$

with

$$\vec{B_z}=(B_{z1},B_{z2}\ldots B_{zN})^T \tag{1.26}$$

for the tipper.

It is important to note that there are some assumptions included in this approach that can cause problems when they are not fulfilled. So equation 1.10 implies that errors occur only in E_{xi}, but neither in B_{xi} nor in B_{yi}[3]. In this context the error-free B_{xi} and B_{yi} are called independent variables or input channels, whereas the E_{xi}, that are allowed to be afflicted with errors, are referred to as a dependent variable or output channel. So it becomes plausible that this method tolerates a certain amount of noise in the output channels quite well, whereas it reacts to noise in the input channels rather sensitively with distorted transfer function results. This will be shown in detail in sections 1.3.1 and 1.3.2. Another restriction is that the transfer functions that would be obtained from single data segments (or, since we face a bivariate problem, rather from two neighboring ones) have to be Gaussian-distributed. This expires, for instance, if a near-by artificial source of an intermittent electromagnetic signal temporarily suggests a different relationship between the measured quantities (cf. section 1.3.3).

[3]In fact, it holds for LMT (*long-period magnetotellurics*) that noise in the electric channels is larger than in the magnetic ones because of instabilities of the electrodes in this period range. However, in AMT (*audiomagnetotellurics*) circumstances are rather contrary, and so there occurs the inverse approach, too, e.g. in Brasse [1993].

1.2 On cascade decimation, Fourier coefficients, spectra, and instruments' responses

In this section there will be shown how the quantities (i.e. spectra, see text below) used in the formulas 1.22, 1.23, and 1.25 are obtained. Being aware of the fact that there is more than one way to do it[4], I use this opportunity to introduce the solution implemented in my own code. This section is rather technical and contributes very little to an understanding of the effects of different equations used in processing approaches. Thus it is recommended that readers mainly interested in that effects neglect this part.

The program has been written in the script language Perl. Ill-willed people refer to it as a "write-only" language, because the syntax admitting many approaches to solve a problem is, in fact, easier to compose than to understand. A certain disadvantage is the relative slowness compared to executable programs of compiled languages: processing the entire data set of a station having regard to two reference sites can take a dozen minutes. On the other hand, the process of developing even complex code is much easier and the result more stringent than with the mentioned alternatives: The single site version of my code contains only 850 lines, although it includes 18 subroutines.

The program reads input data of the binary time series format TRD, which is used by default at the Institute of Geophysics of the Polish Academy of Sciences, since there is excellent software[5] available for contemplating, appraising, and editing time series in this format (see figs. 1.4, 1.5, and many similar ones in this work). The reading takes place in a way that only the maximum 2^n (where $n \epsilon \mathbb{N}$) samples of the given record and – if remote sites are used – only synchronous parts of data are included. The data length of 2^n samples is required for the *Fast Fourier Transform* (FFT, see later on in this section), which is carried out by means of the Perl module (i. e. program library) `Math-FFT`. The second module used is `Time-Local` providing conversions between time strings and epoch seconds (cf. Christiansen and Torkington [2003]). It is very helpful for finding overlapping records of different stations.

The Fourier transform is so important, because it transfers the data from time domain, where they have been measured, to frequency domain, where the desired relationship

[4]The sentence serves a bit as a transition, since it's the slogan of Perl (cf. Christiansen and Torkington [2003]).

[5]VTRD developed by K. Nowożyński

between e. g. electric and magnetic field assumes a simple linear shape.[6] For certain advantages concerning calculating time and memory requirements, in connexion with computer issues usually the FFT is applied. It is given for a real discrete function $y(i)$, $i = 0, \ldots, N-1$ where $N = 2^n$ and $i, n \,\epsilon\, \mathbb{N}$, by the formulas

$$A(f) = \frac{1}{N} \sum_{i=0}^{N-1} y(i) cos\left(\frac{2\pi f i}{N}\right) \tag{1.27}$$

and

$$B(f) = \frac{1}{N} \sum_{i=0}^{N-1} y(i) sin\left(\frac{2\pi f i}{N}\right), \tag{1.28}$$

where $f = 1/T$ is the frequency for which the complex coefficient $FC = A + iB$ is taken. The FFT program of the Perl library returns an array of 2^n values, the Fourier coefficients (real and imaginary part written in tandem) for 2^{n-1} periods. The first period (or rather its coefficient) corresponds to a constant value, the second one to the entire length of the input time series, the third one to a half of it, the fourth one to a third and so on. The shortest period provided with a Fourier coefficient has the length of two sampling intervals. Its reciprocal is called Nyquist frequency.

This consideration shows two problems that have to be solved when adapting a FFT to processing purposes. First, the time series has to be divided into many segments (and the FFT applied to all of them) to get a big number of coefficients for each period as signified in equation 1.10. Second, if coefficients for long periods are desired, the time series segments must be as long as possible. These requirements are somewhat contradicting since, to avoid redundant information, the segments should not overlap. An obvious solution is to execute the segmentation via a cascade decimation, i. e. in my case that the time series is successively cut into halves. These subsequent steps are called decimation levels. Before the FFT is applied to the actual segment, the latter is subjected to some procedures reducing unwanted effects in the FFT due to its fragmentation, i. e. constant values and linear trends are removed and the series is weighted with a Hanning window H according to

$$H = 0.54 - 0.46\ cos\left(\frac{2\pi i}{N-1}\right) \tag{1.29}$$

where N is the length of the actual segment and i the index of the sample.

[6] However, there are processing approaches carrying out the most essential steps in time domain, for example Nowożyński [2004].

Length of entire time series: 2^N samples with $N = 18$		i = decimation level L = length [samples] of time series segment = $2^{(N-i+1)}$						
		$i = 1$ $L = 2^{18}$	$i = 2$ $L = 2^{17}$	$i = 3$ $L = 2^{16}$	...	$i = 10$ $L = 2^9$	$i = 11$ $L = 2^8$	$i = 12$ $L = 2^7$
Period [samples] of coefficients	L/4	65 536	32 768	16 384	...	128	64	32
	L/5	52 429	26 214	13 107	...	102	51	26
	L/6	43 691	21 845	10 923	...	85	43	21
	L/7	37 449	18 725	9 362	...	73	37	18
	L/8	32 768	16 384	8 192	...	64	32	16
	L/9	29 127	14 564	7 282	...	57	28	14
	L/10	26 214	13 107	6 554	...	51	26	13
	L/11	23 831	11 916	5 958	...	47	23	12
	L/12	21 845	10 923	5 461	...	43	21	11
	L/13	20 165	10 082	5 041	...	39	20	10
	L/14	18 725	9 362	4 681	...	37	18	9
	L/15	17 476	8 738	4 369	...	34	17	8.5

Figure 1.1: The table demonstrates for which period Fourier coefficients are taken depending on the actual decimation level during the cascade decimation for a 6 days long time series sampled with a 2 s interval.

Now, the FFT yields coefficients for a different set of periods shortening with each decimation level. Only a part of them is included into further usage. Concretely, it is the assemblage referring to the periods

$$\frac{N}{i}, \quad i = 4, 5, \ldots, 15 \tag{1.30}$$

where N is, again, the length of the actual segment. Fig. 1.1 demonstrates that pattern for a case study of a 2^{18} samples long time series. This kind of selecting coefficients has a somewhat funny consequence: Certain periods occur on two decimation levels and are thus more frequently represented in the resulting collection than neighboring ones. Their number of coefficients aggregates to the threefold compared to single-level periods. Fig. 1.2 illustrates that distribution of coefficients over the period.
It may be added that the lowest period for which coefficients are obtained is slightly above the octuple sampling interval and the longest one further used is one sixteenth of the maximum record length.

The obtained Fourier coefficients have to be corrected for the response function of the instrument (e.g. caused by filters) that has measured the given channel. This would not be necessary if all devices involved in the data acquisition for a transfer function had the same one. Fig. 1.3 shows that this is not the case. The difference between the responses of telluric and magnetic recording systems would distort impedances,

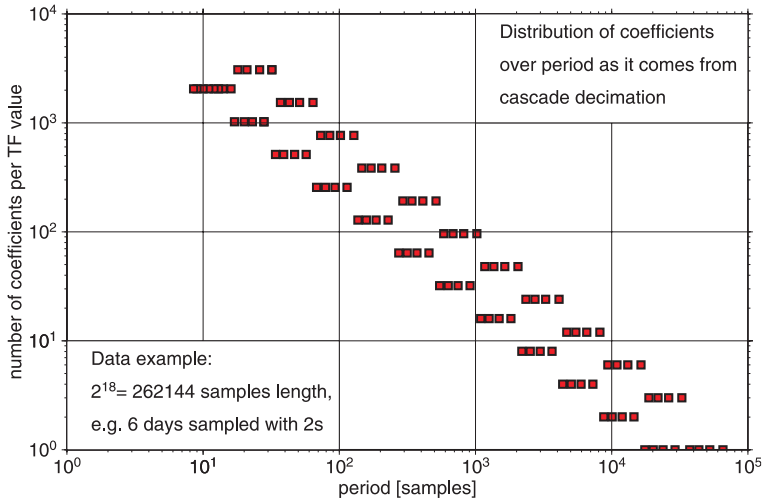

Figure 1.2: Histogram of coefficients obtained by the applied cascade decimation for the example given in fig. 1.1.

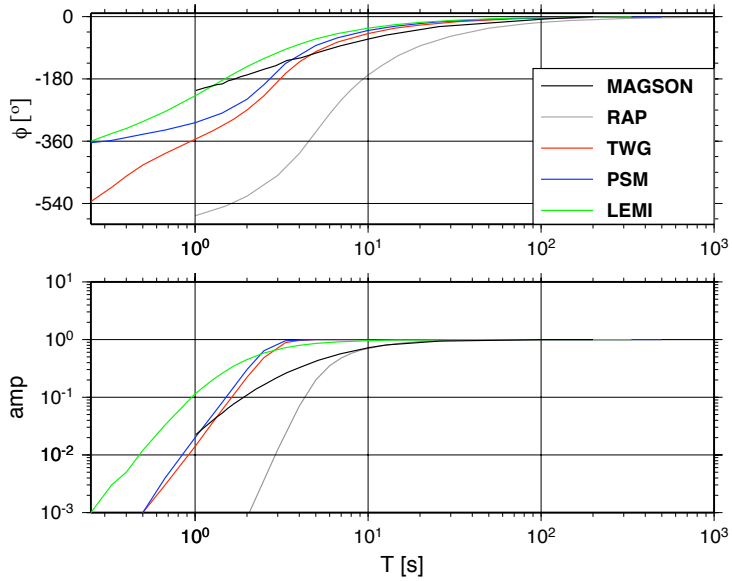

Figure 1.3: Typical instruments' responses of the LMT devices used for the profile on fig. 1. MAGSON marks German fluxgate magnetometer, RAP – German telluric amplifier, TWG – Polish telluric amplifier, PSM – Polish quartz torsion magnetometer, and LEMI – a Polish fluxgate magnetometer of Ukrainian origin.

and inter-station magnetic transfer functions measured between different magnetometer types would suffer, as well.

Often the result of a Fourier transform is displayed not in form of coefficients, but as spectra. These values are obtained by multiplying a coefficient with its complex conjugate. This operation yields a zero imaginary part and the square of the amplitude at the

given period in the real part. Figs. 1.4 and 1.5 show time series and the corresponding spectra, respectively. More correctly, the spectra are called auto(power)spectra in this context. In contrast, the product of a Fourier coefficient with the complex conjugate of the coefficient of another measured channel[7] is called cross spectra.
Since all expressions used in the equations for the transfer functions (1.22, 1.23, 1.25) are acquired now, the latter can be evaluated. The results are stored in edi[8] files. Fig. 1.6 shows processing results obtained with the described method (left hand side) and, for comparison, with the well-established algorithm after Egbert and Booker [1986] (right hand side). In this example, the data are from site DRE (cf. fig 1), which has time series of high quality. Obviously, the developed code serves its purpose, at least under convenient circumstances.

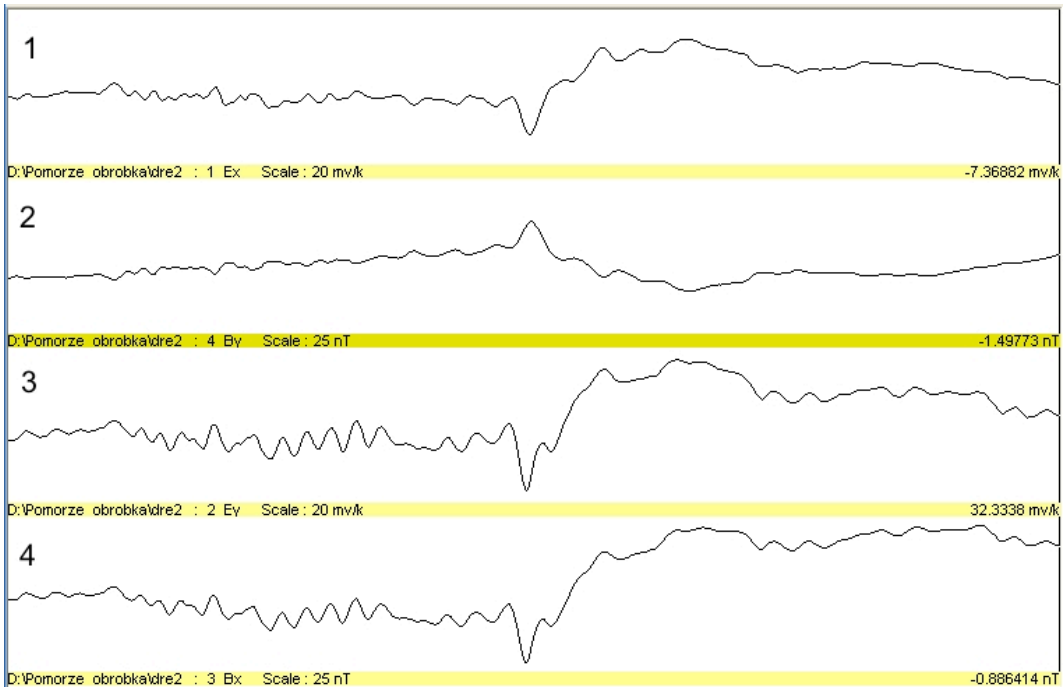

Figure 1.4: Time series of site DRE (cf. fig. 1) for channels E_x (1), B_y (2), E_y (3), and B_x (4). The length of the window is 25 minutes. The scale is 20 mV/km for telluric channels and 25 nT for magnetic ones.

[7]Of course, it has to be taken for the same period and must stem from the same time segment.

[8]The edi format is an international standard for *e*lectronic *d*ata *i*nterchange purposes, see *http://www.geophysics.dias.ie/mtnet/docs/ediformat.txt*

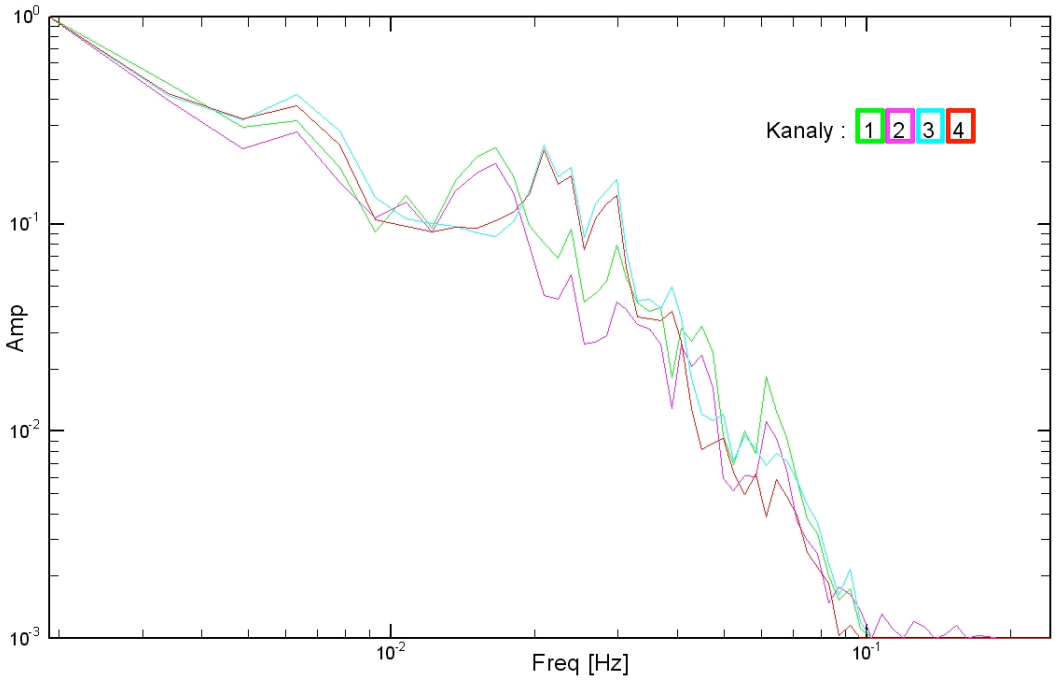

Figure 1.5: Spectra corresponding to time series in fig. 1.4. Marks refer to the same channels. The amplitude is normalized to 1. Note the correlation between E_x and B_y and between E_y and B_x.

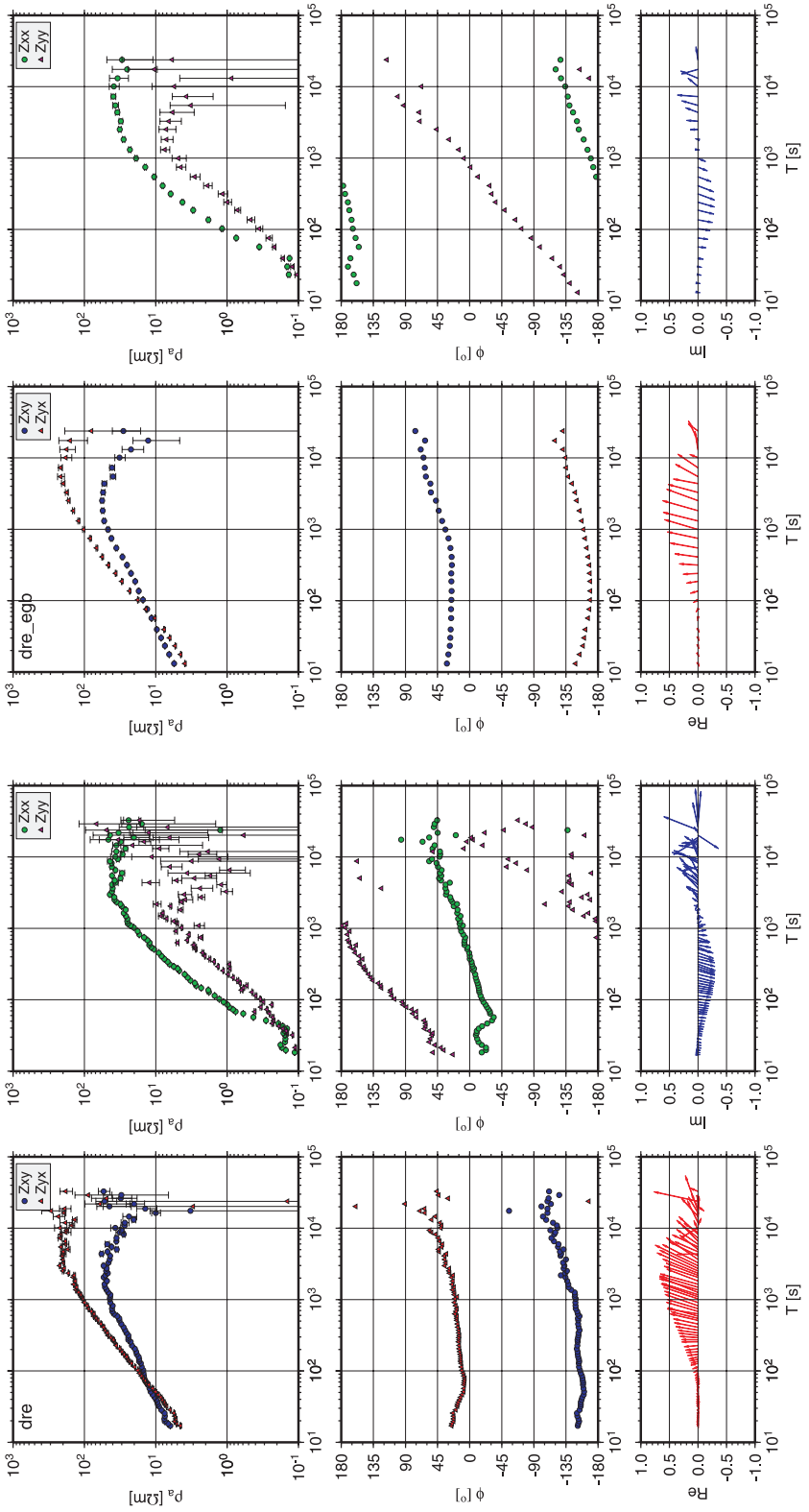

Figure 1.6: Processing results for station DRE, left hand site obtained with the described code, right after Egbert and Booker [1986]. It strikes that the phases are shifted by $\pm 180^o$, so they appear interchanged. That's a matter of convention and not tangent to their meaning. The results look alike.

1.3 Noise

Initially, noise is everything, that is present in the measured data, but has nothing to do with the relationship between different field components that is used in magnetotellurics. However, since the least-square estimation produces a consistent relationship between data measured for those components, the transfer functions resulting from that estimation are not necessarily appropriate for the further magnetotelluric evaluation. In this section there will be explained how disturbed transfer functions can be recognized and which kind of noise causes such disturbances.
Noise can have miscellaneous reasons. Concerning the electric channels, instabilities of the electrodes play a role as well as emissions from grounded power cables, if the station is installed close to a consumer load (e.g. a household). Magnetic channels also suffer from a certain instruments' noise, which is especially visible in a period range $< 20s$ at the MAGSON fluxgates. Furthermore, there are disturbances in the magnetic channels caused by close-by moving magnetic objects. Nearby artificial sources of electromagnetic signals like DC railways, corrosion protected gas pipelines, and pasture fences, produce correlated noise in electric and magnetic channels (cf. Szarka [1988], Junge [1996]). In many cases, noise can easily be recognized in the time series by its shape. Magnetotelluric signals are always sinusoidal, smooth, and correlated between certain channels. Everything else, like especially peaks, spikes, jumps, transient-like forms, and rectangular signals (even if somehow softened), is noise.
A non-robust single-site processing is the simplest, least sophisticated, and hence least stable method in the matter of noise. Therefore the negative consequences of different kinds of noise can be demonstrated very well considering it as example. This will be done in the following. The degree to which more elaborated methods are able to deal with that negative consequences of noise will be essential to evaluate their capability in subsequent chapters of this work.

1.3.1 Statistic noise in output channels

As already mentioned in section 1.1, electric channels and B_z are the output channels in a single-site context, and the least square approach copes best with noise in just these channels. A not too high degree of output noise leaves the estimated transfer functions nearly untouched and increases only their error bars, the latter being thereby a practical measure for such noise. However, if that degree is exceeded, the obtained transfer function will scatter and become unusable. But at least it is immediately

clear that the processing has failed in such a case[9] and that the results are not to be included in further work. In this respect, noise on output channels is relatively harmless.
In the following I will explain origin and significance of error bars and show two case studies, where the mentioned noise is still compensated or leads the processing into failure, respectively. The common instrument against such noise is robust procedures.

My error estimation follows Meju [1994] along general lines. It contains two important steps. First, there is determined the sum of square errors, referring to "error" as the difference between measured output data and those reconstructed with the estimated transfer functions, devided by the difference of the number of data and the number of independent variables. If we stay with the example given by equation 1.10, we get

$$\chi^2 = \frac{\sum_{i=1}^{N} \left(E_{xi} - Z_{xx}B_{xi} - Z_{xy}B_{yi}\right)\left(E^*_{xi} - Z^*_{xx}B^*_{xi} - Z^*_{xy}B^*_{yi}\right)}{N-2} \tag{1.31}$$

Therewith, a measure for the success of the minimization according to equation 1.11 is already given. However, χ^2 carries the (squared) unit of the output channel and does not yield any information about how to distribute the error among the both transfer functions contained in equation 1.10. Thus, a second step is required. It bases on the term of the covariance matrix. If one has an arbitrary linear inverse problem

$$\vec{d} = \mathbf{G}\vec{m} + \delta\vec{d} \tag{1.32}$$

with the data $\vec{d}$ (here: E_{xi}), the data kernel matrix $\mathbf{G}$, the error $\delta\vec{d}$, and the desired model parameters $\vec{m}$ (here: Z_{xx} and Z_{xy}), then the covariance matrix $\mathbf{COV}$ is (Meju [1994])

$$\mathbf{COV} \propto \left(\mathbf{G}^T\mathbf{G}\right)^{-1}, \tag{1.33}$$

where the residual square sum from eq. 1.31 can be taken as constant of proportionality after Eadie et al. [1971], p. 164/165. So the covariance matrix for our problem is

$$\mathbf{COV} = \chi^2 \left(\mathbf{B}^\dagger\mathbf{B}\right)^{-1}. \tag{1.34}$$

The covariance matrix is real, square, and of the dimension of the number of model parameters. Its diagonal elements are the variances σ^2 of the model parameters, which in this form are written to the edi file. The off-diagonal elements are in the ideal case

[9] It's a matter of common knowledge that "correct" magnetotelluric transfer functions are continous and, by nature, smooth.

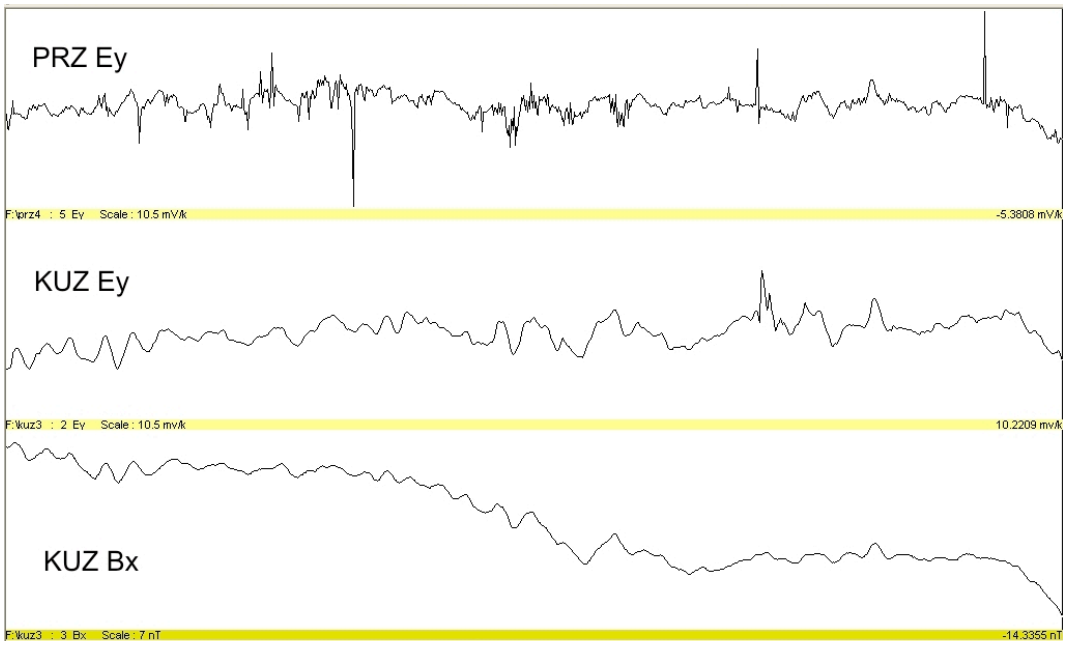

Figure 1.7: Typical time series of sites PRZ and KUZ. The scale for E_y is 10.5 mV/km, for B_x 7 nT. The window length is 34 min. See text for comments.

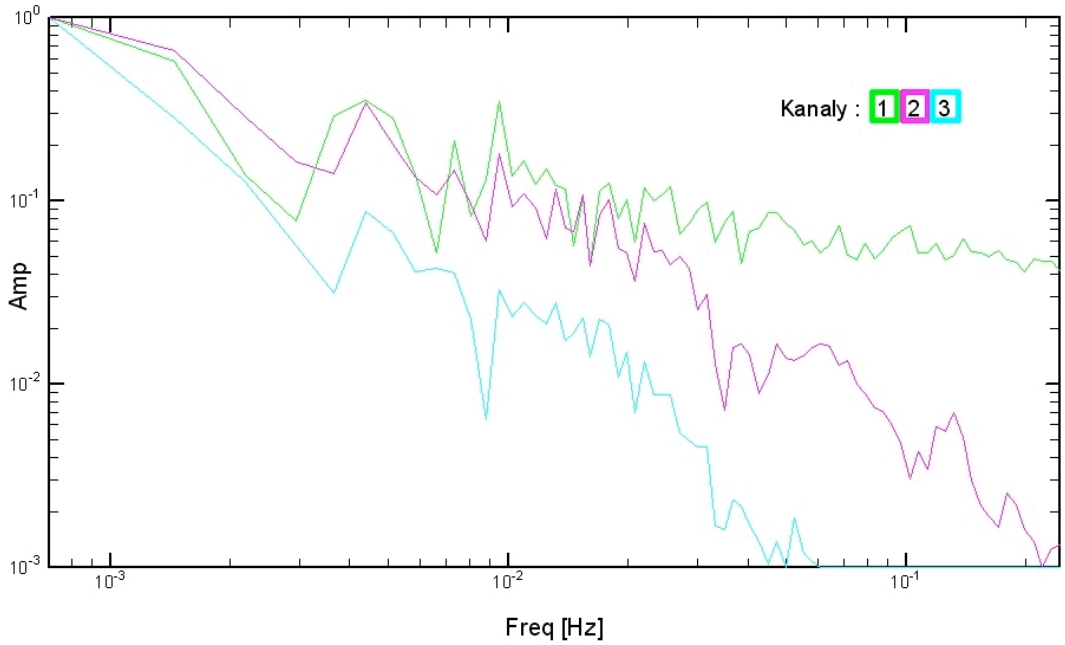

Figure 1.8: Normalized spectra of time series in fig. 1.7. 1 corresponds to E_y (PRZ), 2 to E_y (KUZ), and 3 to B_x (KUZ). See text for comments.

close to zero. They indicate to what extent the model parameters are correlated with each other. The value used for plotting is the standard deviation σ. Added positively and negatively to the given impedance value, it forms its error bar.

Fig. 1.7 shows time series of the stations KUZ and PRZ (cf. fig. 1). The first

one is occasionally disturbed on channel E_y, the latter almost permanently. Since the disturbances of both stations are, obviously, not correlated and the magnetic channel B_x of KUZ looks untouched, the stations are suited for the problem treated of here. Fig. 1.8 shows the spectra of these time series. Although B_x and E_y of site KUZ diverge strongly at high frequencies, some correlation between them is visible. The electric channel of PRZ lacks such correlation[10]. Already from this can be guessed that stable transfer functions will be found for KUZ but not for PRZ. For the following diagram a three days long[11] record of both stations has been processed. This led to 1024 Fourier coefficients at the period $32s$. After estimating the impedance from them, the square errors between measured and reconstructed E_y have been calculated for all 1024 cases. Their distribution is displayed in fig. 1.9. It becomes clear that almost all errors are minimum for KUZ. This holds for PRZ for a majority, too. However, more than one third of the PRZ errors is - partly significantly - larger. This is a further hint on convergence problems during the determination of transfer functions Z_{yx} and Z_{yy} for PRZ. Fig. 1.10 confirms the suspicion. The processing results for KUZ are exemplary[12]. In contrast, there couldn't be obtained curves for PRZ. The values are so scattered that the processing has to be regarded as failed. However, a robust single-site approach like Egbert and Booker [1986] does not encounter scattering difficulties even with such a poor data quality.

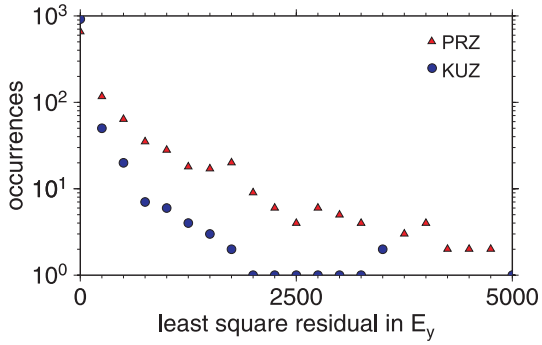

Figure 1.9: Distribution of errors in E_y among the 1024 data examples at $32\,s$ for the case study described in the text. KUZ appears to be well determined, but for PRZ more than 1/3 of the examples doesn't support the correct value where the error is minimum.

[10]Indeed, B_x of PRZ is not displayed, but it is not very different from that of KUZ, since the sites are close to each other, cf. fig. 1.

[11]This corresponds to 2^{17} samples with a $2s$ sampling interval.

[12]These curves do not differ from the results obtained with the robust code by Egbert and Booker [1986].

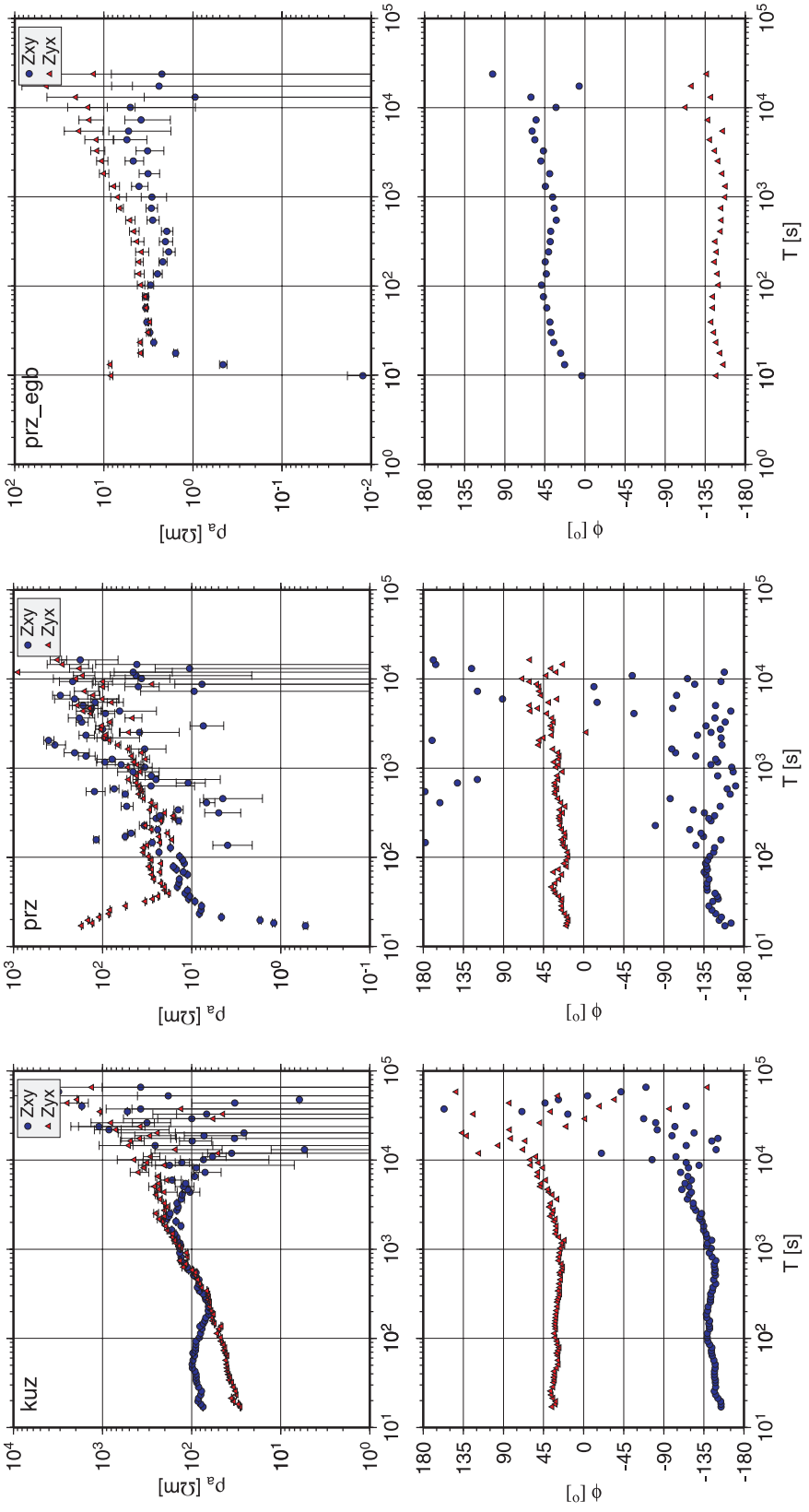

Figure 1.10: Processing results for off-diagonal impedances of sites KUZ (left) and PRZ (middle and right). For KUZ the non-robust single-site processing has succeeded. For PRZ (middle) it failed. In contrast, the robust version by Egbert and Booker [1986] was able to estimate stable transfer functions (right).

1.3.2 Statistic noise in input channels

In this single-site example, input channels are always B_x and B_y. Noise on them hits a sore spot of the described processing method. It is not compensated by statistics, and so a small number of affected time series segments is already able to cause distortions of the transfer functions. These distortions manifest themselves as a certain "dropping behavior" (i.e. a continuous down-weighting) of the apparent resistivity curves at short periods. In other words, the error does not consist in scattering, but in a systematic shift in one direction. Such errors are referred to as bias.

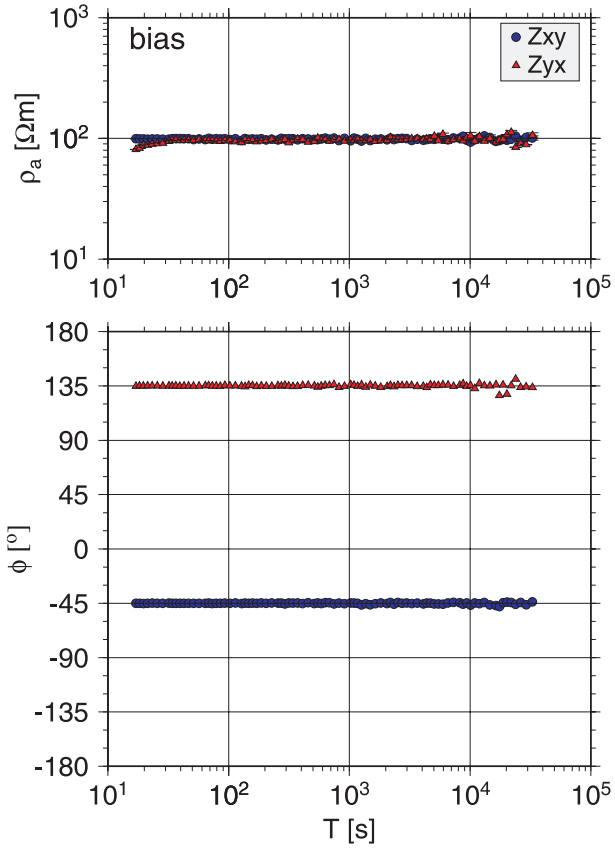

Figure 1.11: Processing results for synthetic data of a $100\,\Omega m$ homogeneous half-space with artificially disturbed channel B_x. The ρ_a curve of Z_{yx} "drops" slightly in a typical way at short periods.

Fig. 1.11 displays a slightly biased ρ_a-curve for the impedance element Z_{yx}. The case study bases on synthetic data of a $100\,\Omega m$ homogeneous half-space, where peaks of $16\,nT$ height have been added to the B_x channel in a way, that 6.25 percent of the time series segments used to obtain Fourier coefficients for the shortest periods are affected.

This small amount is enough to shift the first ρ_a value by $20\Omega m$. Furthermore, this happens without leading to warning error bars.
There is a definite reason for the specific form of the bias. The input channels go into the denominator of the formulas of the transfer functions as auto spectra (see e.g. equation 1.19). Uncorrelated noise increases the spectra by the auto spectra of the noise, so if some signal S is contaminated with some statistic noise N:

$$[(S+N)(S+N)^*] = [SS^* + SN^* + NS^* + NN^*] = [SS^*] + [NN^*]\,, \qquad (1.35)$$

where $[XX^*]$ indicates an averaging over a number of spectra of type given in between. Since a division by that too large value takes place then, the result is down-weighted. Only ρ_a is concerned by this effect, but not the phase, since auto spectra are real-valued and division by a real number does not change the ratio between imaginary and real part of the result, which determines the phase (equation 1.6).
It is possible to rescue the transfer functions from such a noise situation by means of the Remote Reference technique. The most important feature of it is the substitution of auto spectra by cross spectra with the horizontal magnetic channels of another site (see chapter 2). This works very well in practice. In methodic terms it is, maybe, not very satisfying. The problem is the assumption that some measured channels are noise-free, since measurements are always concerned with errors. The approach that instrumentally acquired data must be the determinative quantity in the problem is clearly transcended only by Egbert [1997].

1.3.3 Correlated noise

Correlated noise (CN) is especially dangerous. As the term suggests, it occurs on input and output channels simultaneously, see fig. 1.12. Due to this it fulfills an important condition for a magnetotelluric signal, remains undetected even by methods capable of excluding statistic noise, e. g. via coherency criteria, and succeeds in going into the processing results. Transfer functions affected by CN are not usable for an induction-based interpretation. In serious cases, they carry only information about the source producing the CN, but none about the Earth's conductivity. Typical features of a correlated-noise-induced distortion are ρ_a curves rising in an angle of 45^o in an equidistant log-log plot, very low phases and large parallel induction arrows with real ones pointing towards the source. These features will be addressed in detail in section 4.3.
Fig. 1.13 shows such an example. Station SAR is situated 6.5 km from a DC railway

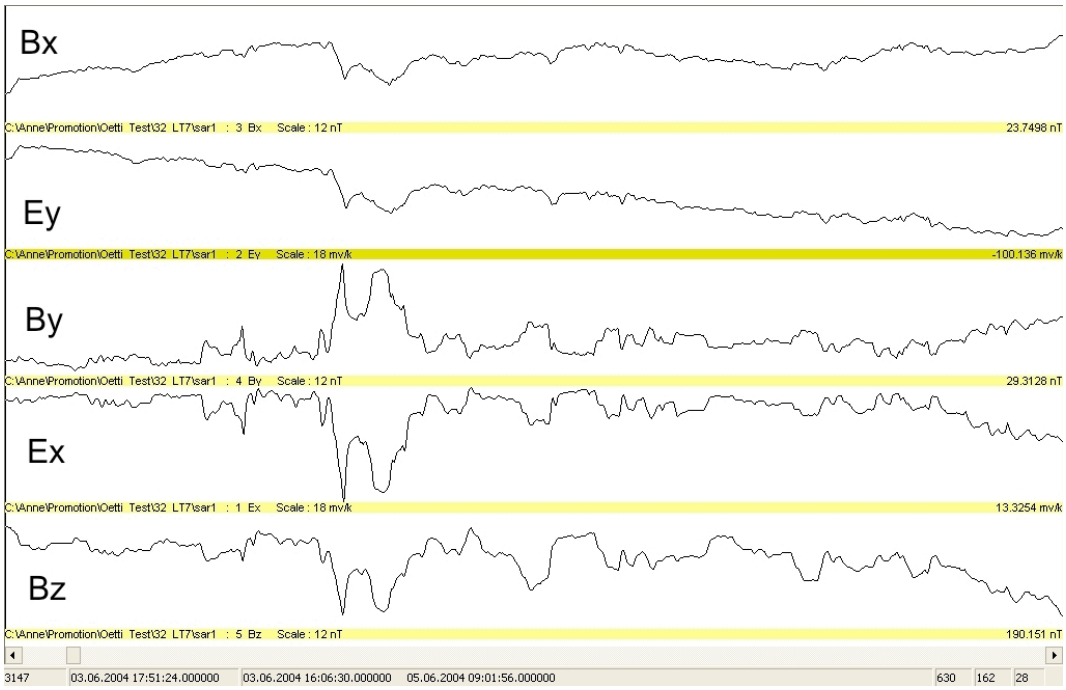

Figure 1.12: Time series of station SAR. The scales are $18\,mV/km$ and $12\,nT$, the window length is 34 min. The peaked features that are correlated between the channels are produced by a DC railway 7 km off the site.

in the West. It becomes also clear that a robust code is just as powerless.
The solution is to use a remote site that must be so far away from the source of the CN that it is beyond of its reach. Fortunately, such noise has the property to decrease relatively fast with distance to the source.
In all subsequent chapters of this work there will be described methods that meet the challenge of correlated noise.

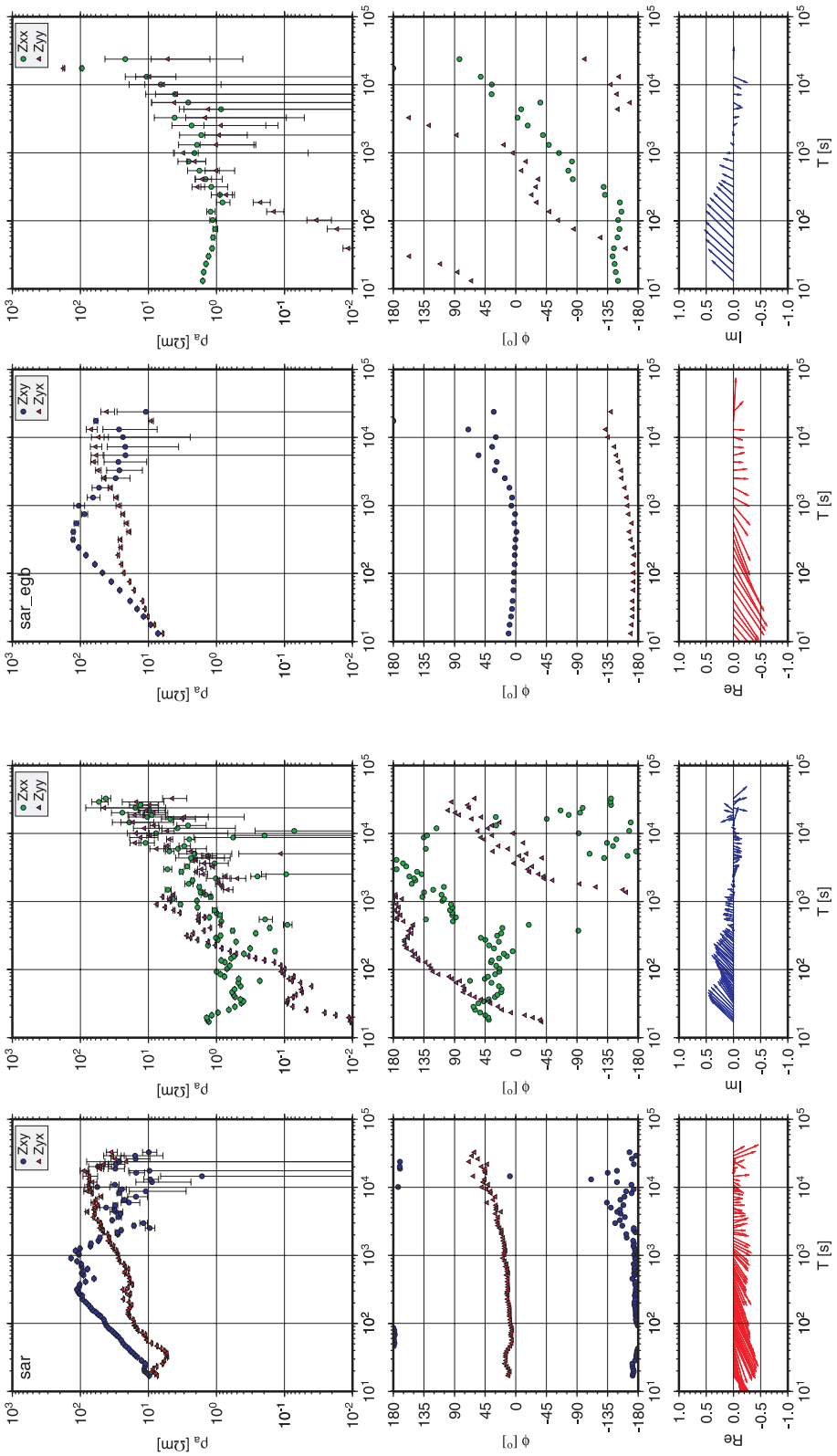

Figure 1.13: Processing results for station SAR. All transfer functions are heavily affected by correlated noise, both by the non-robust (left) and the robust (Egbert and Booker [1986], right) approach obtained. The differences below 100 s show the effect of smoothing.