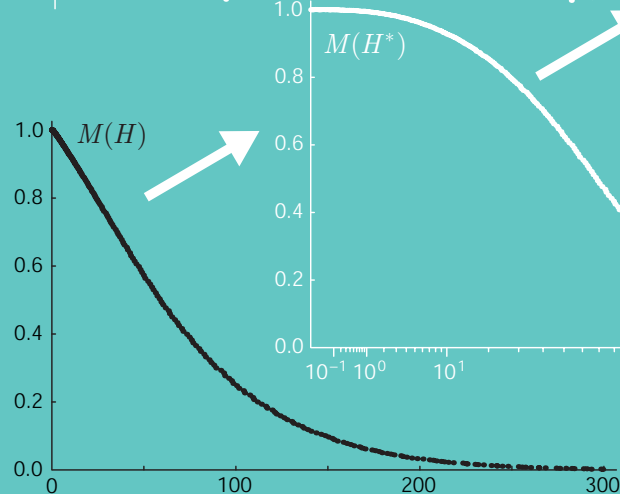
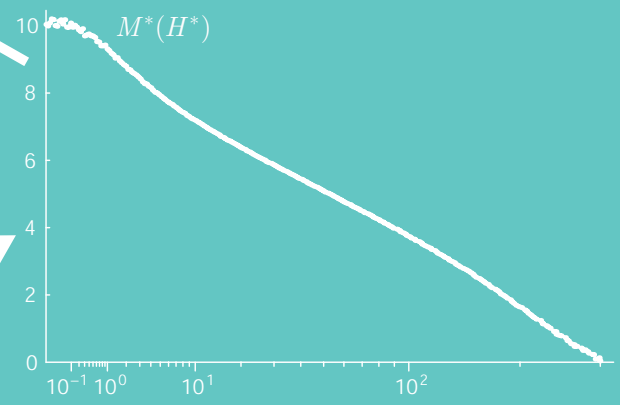
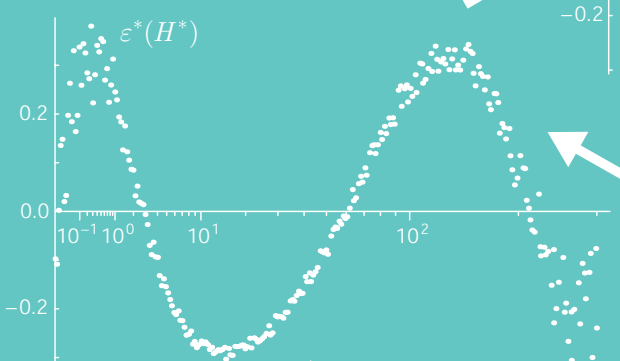
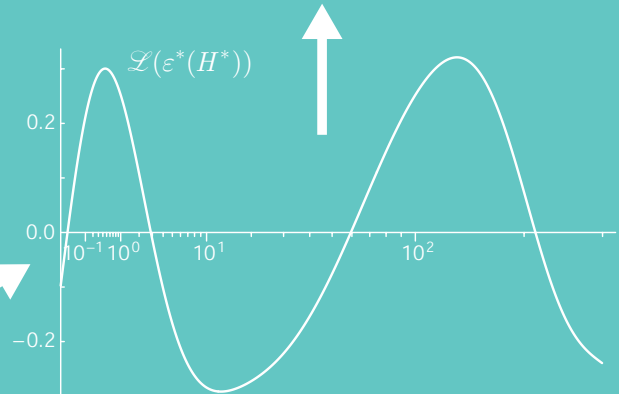
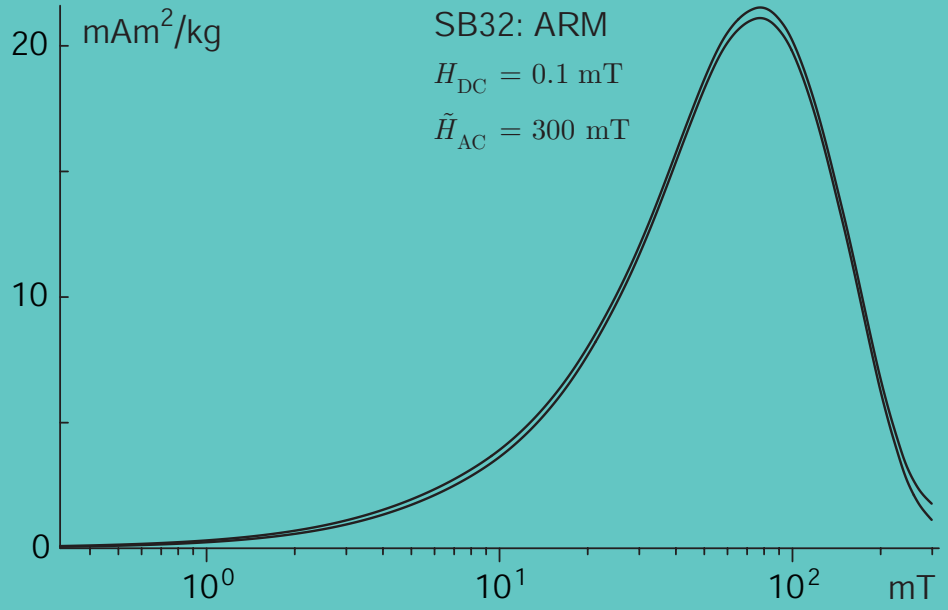


# CODICA 3.0

© Ramon Egli 2004



**CODICA**  
**CO**ercivity  
**DI**stribution  
**CA**lculator

## Introduction

CODICA (COercivity DIstribution CAculator) is a program which efficiently removes the measurement errors from acquisition/demagnetization curves and calculates the corresponding coercivity distribution on different field scales. The efficiency of the measurement errors removal is based on several rescaling procedures which converts the original acquisition/demagnetization curve to a residuals curve. The residuals curve is low-pass filtered and transformed back into an acquisition/demagnetization curve where the noise has been reduced more efficiently than by direct filtering. An error estimation is performed as well, and the coercivity distribution is plotted together with confidence limits.

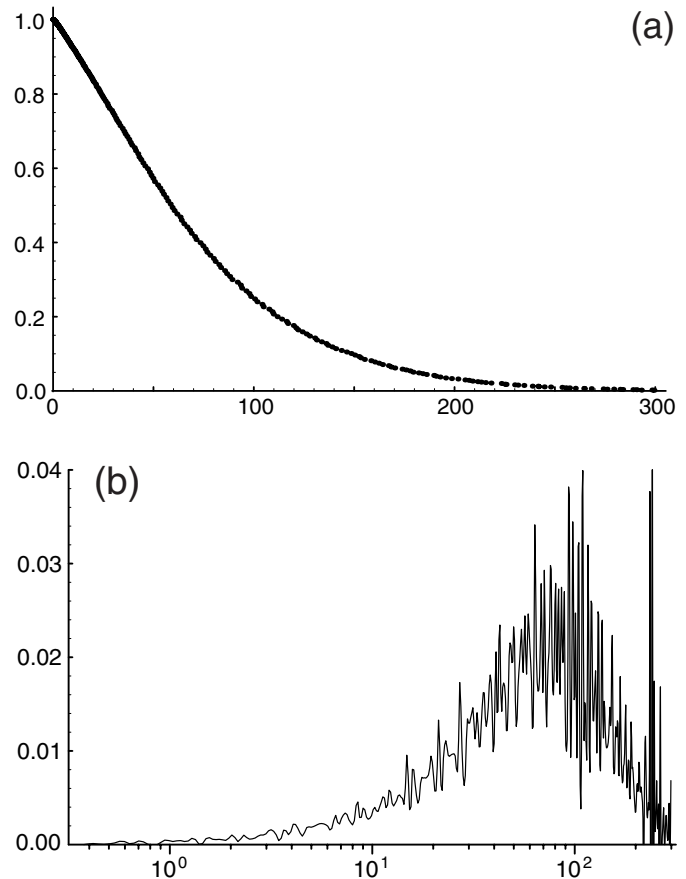
Read carefully this manual to learn about CODICA. Many steps of CODICA seems complicate, but in reality they are quite intuitive once you are familiar with the program. This manual contains a theoretical part, which gives you the background to understand the basic ideas of CODICA, and a practical part, which guides you through each step of the program. You can practice with the examples delivered together with this program. CODICA is designed to perform an optimized data treatment of acquisition/demagnetization curves. It allows you to easily recognize and efficiently remove the measurement errors. The parameters you are asked to enter do not directly affect the final result since they only control the efficiency of the measurement errors removal procedure. Therefore, the choice of these parameters is not critical and the results are not “user dependent”, as long as the instructions given in this manual are followed. You should always consider that CODICA does not add any information to the original measurements, as any other program for data analysis: bad measurements will give poorly defined coercivity distributions affected by large errors. Nevertheless, CODICA helps you to identify the measurement errors and better redesign the experiment, by optimizing the magnetization/demagnetization steps and stacking more measurements, if necessary. Click on the following topics to see the contents of this manual:

<b><u>Theoretical background</u></b>	<b>2</b>
• <a href="#">Coercivity distributions</a>	2
• <a href="#">Rescaling the field axis</a>	3
• <a href="#">Linearization procedure</a>	3
• <a href="#">Residuals curve</a>	4
<b><u>CODICA step by step</u></b>	<b>6</b>
• <a href="#">Data checking</a>	6
• <a href="#">Scaling the magnetic field</a>	6
• <a href="#">Scaling the magnetization</a>	6
• <a href="#">Plotting the residuals</a>	8
• <a href="#">Scaling the residuals</a>	8
• <a href="#">Filtering the residuals</a>	9
• <a href="#">Calculating the filtered demagnetization curve</a>	10
• <a href="#">Calculating and plotting the coercivity distribution</a>	10
<b><u>A program example</u></b>	<b>12</b>
<b><u>Install CODICA</u></b>	<b>22</b>
<b><u>Cautionary note</u></b>	<b>33</b>

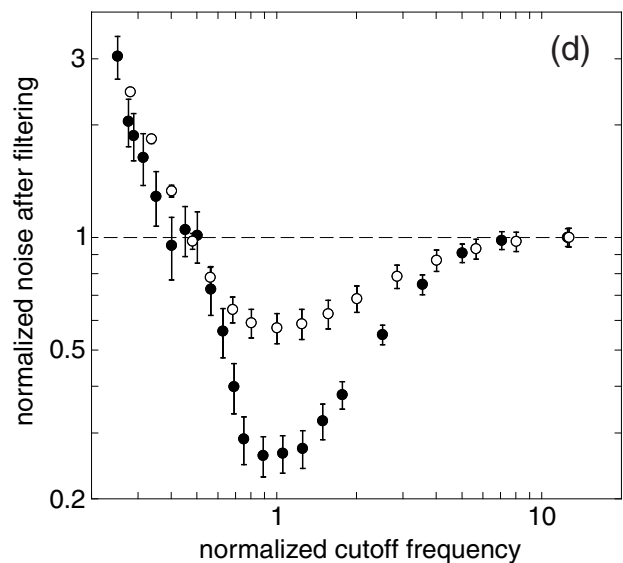
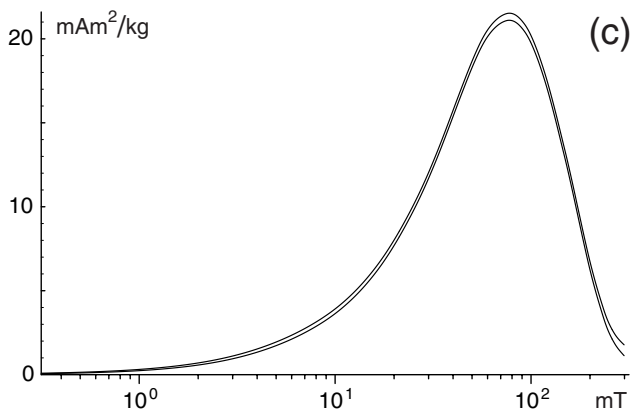
## Theoretical background

Coercivity distributions are defined as the absolute value of the first derivative of acquisition/demagnetization curves. In terms of Fourier analysis, the first derivative is equivalent to a high-pass filter, whose effect is to enhance small details of the original curve (see Fig. 1). For this reason, any information contained in the original curve will be more evident in the resulting coercivity distribution. This applies also to the measurement errors, which are generally small in the measured curve, but are enhanced in the resulting derivative. The enhancement of small measurement errors is the main reason why coercivity distributions have not been used very often in the interpretation of magnetic measurements, despite the potential advantages.

Measurement errors may be removed with an appropriate filter. The measured curves are often asymmetric and they require a different degree of filtering at different fields. CODICA (COercivity DIstribution CALculator) is a program which removes the measurement errors by appropriately rescaling and filtering the acquisition/demagnetization curve. It calculates the corresponding coercivity distribution on different field scales and estimates the confidence limits. The latter is particularly useful when the significance of a component analysis performed on the coercivity distribution has to be evaluated.



**Fig 1:** An AF demagnetization curve (a), and the corresponding coercivity distribution (b), calculated by numerical differentiation. In (c), the coercivity distribution calculated with CODICA is plotted on a logarithmic field scale. The curve pair indicates the confidence limits. In (d), the efficiency of CODICA in removing the measurement noise (dots) is compared with a conventional low-pass filter (open circles).



Coercivity distributions are mathematically described by probability distribution functions (PDF) and can be calculated on different field scales for better isolation of different components. The shape of a coercivity distribution changes according to the field scale adopted, because an integration over all coercivities corresponds to the total magnetization of the sample (normalization property). The scale change of a distribution  $f$  generates a new distribution  $f^*$  defined by the following transformation rule:

$$H^* = g(H) ; \quad f^*(H^*) = f(g^{-1}(H^*)) \frac{d}{dH^*} g^{-1}(H^*) \quad (1)$$

where  $H^*$  is the new field scale,  $g$  the transformation rule between the old and the new scale, expressed by an injective function with inverse  $g^{-1}$ , and  $f^*$  the coercivity distribution with respect to the new scale. For example, the transformation rule from a linear to a logarithmic scale is expressed as follows:

$$H^* = \log H ; \quad f^*(H^*) = \ln 10 \cdot 10^{H^*} f(10^{H^*}) \quad (2)$$

Another useful transformation is the following:

$$H^* = H^p ; \quad f^*(H^*) = \frac{(H^*)^{1/p-1}}{p} f((H^*)^{1/p}) \quad (3)$$

where  $p$  is a positive exponent. We will refer to this transformation as the power transformation. The power transformation converges to the logarithmic transformation for  $p \rightarrow 0$ . If  $0 < p < 1$ , high coercivities are quenched on the field scale, and distributions with large coercivities are enhanced. The same effect is obtained with a logarithmic scale, and the opposite effect with  $p > 1$ . [Click here](#) to see the effect of the field scale transformation on the shape of a coercivity distribution.

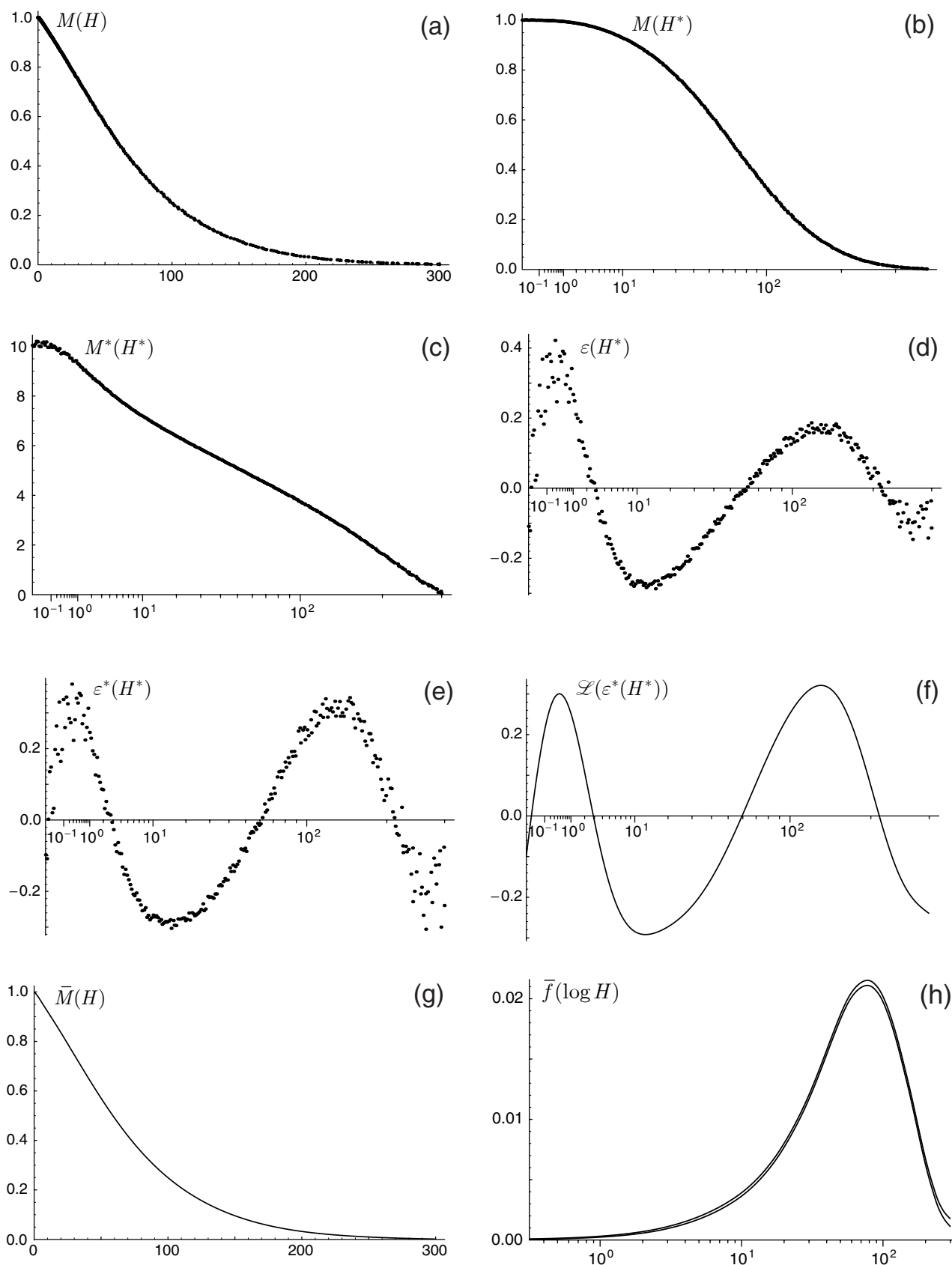
In the following, the technique used by CODICA to remove the measurement noise homogeneously along the entire curve is presented. It is based only on the following assumption: all acquisition/demagnetization curves have two regions where the corresponding coercivity distribution is zero on a logarithmic field scale, one at  $H \rightarrow 0$  and the other at  $H \rightarrow \infty$ . In other words, any acquisition/demagnetization curve has two horizontal asymptotes on a logarithmic field scale. The physical meaning of this assumption for  $H \rightarrow \infty$  is obvious: all magnetic minerals have a maximum finite coercivity. For  $H \rightarrow 0$  the physical explanation is related to thermal activation processes in SD and PSD particles, and with domain wall motions in MD particles. Measurements with a sufficient number of points near  $H = 0$  are necessary in order to obtain a correct coercivity distribution for small fields. Appropriate scaling of field and magnetization allows the linearization of acquisition/demagnetization curve. On the linearized curve, each measurement point and the rela-

ted error have the same relative importance, so that a simple low-pass filter can be applied to remove the measurement noise with the same effectiveness for all coercivities. An acquisition/demagnetization curve  $M = M(H)$  can be linearized in a simple way by rescaling the field according to the transformation rule  $H^* = M(H)$ . However, because of the measurement errors, the function  $M(H)$  is unknown. A good degree of linearization is reached when a model function  $\tilde{M}(H)$  expressed by analytical functions is taken as transformation rule instead of  $M(H)$ . The choice of the appropriate model function  $\tilde{M}(H)$  becomes simpler if field and magnetization are both rescaled. If  $M^* = \mu(M)$  and  $H^* = g(H)$  are the rescaling functions for the field and the magnetization, respectively, then the model function is given by  $\tilde{M}(H) = \mu^{-1}(g(H))$ . The relation  $M^*(H^*)$  between scaled field and scaled magnetization approaches a straight line when the model function  $\tilde{M}(H)$  approaches the (unknown) noise-free magnetization curve. The curve defined by  $\varepsilon(H^*) = M^*(H^*) - H^*$  represents the deviations of  $M^*(H^*)$  from a perfect linear relation. We call  $\varepsilon(H^*)$  the *residuals curve*. If the model function  $\tilde{M}(H)$  used for the scaling procedure is identical with the noise-free magnetization curve, then the residuals curve contains only the measurement errors. In reality, since it is impossible to guess the noise-free magnetization curve, the residual curve is a superposition of the nonlinear component of  $M^*(H^*)$  and the measurement errors. The fundamental advantage of considering the residuals curve instead of the original curve is that the measurement errors are highly enhanced in the residuals curve and can be homogeneously removed with a low-pass filter. The choice of the filter parameters is not critical, and has little effect on the shape of the resulting coercivity distribution. Under ideal conditions,  $\varepsilon(H^*)$  represents only the measurement errors, which can be simply removed by fixing  $\varepsilon(H^*) = 0$ .

The filtered residual curve can be transformed back into a magnetization curve as follows:

$$\bar{M}(H) = \mu^{-1}[\mathcal{L}[\varepsilon(g^{-1}(H))] + g^{-1}(H)] \quad (4)$$

where  $\mathcal{L}(\cdot)$  is the low-pass filter operator.  $\bar{M}(H)$  is now supposedly free of measurement errors.



**Fig. 2:** Main steps of data processing by CODICA. Original demagnetization curve (a). Demagnetization curve after rescaling of the field axis (b), and the magnetization axis (c). Residuals curve before (d) and after rescaling (e). Filtered residuals (f), filtered demagnetization curve (g) and coercivity distribution (h). See the text for more details.

## CODICA step by step

The processing of an acquisition/demagnetization curve with CODICA consists in the following steps.

### 1. *Data checking* ([figure 2a](#))

The measurement curve is always displayed as a [demagnetization curve](#): this does not affect the calculation of the coercivity distribution. The user is asked to enter an estimation of the measurement error (if known). This estimate may come from inspection of the measurement curve and/or experimental experience with the instruments used. A combination of a relative and an absolute error is assumed to affect the measurements and the AF field. Systematic errors, like magnetization offsets and temperature effects on the sample and on the AF coil do not affect significantly the shape of the measured curve and may not be included. The calculation of a coercivity distribution and the related measurement error is independent of the error estimate entered by the user. This first estimate is needed by the program in order to display a rough estimation of the confidence limits of the measured curve, which should help the user through the following steps of the program.

### 2. *Scaling the magnetic field* ([figure 2b](#))

Acquisition/demagnetization curves are supposed to have a sigmoidal shape on a logarithmic field scale. However, the measured curves are not symmetrical. In case of lognormal coercivity distributions, the measured curve represented on a logarithmic field scale becomes symmetric. In all other cases the curve is asymmetric on both a linear and a logarithmic field scale. An appropriate scale change which offers a set of intermediate scales between linear and logarithmic is defined by the power function  $H^* = H^p$ ,  $p$  being a positive exponent. An appropriate value of  $p$  is chosen, so that the scaled curve reaches maximum symmetry. The symmetry of the curve is compared with a reference sigmoidal curve, expressed by a tanh function. For this purpose, the scaled curve is therefore represented together with the best-fitting tanh function. An automatic routine optimizes the scaling exponent  $p$  so that the difference between the original curve and the model curve is minimal. In most cases,  $p \approx 0.3$ .

### 3. *Scaling the magnetization* ([figure 2c](#))

A tanh function was chosen as a reference function for scaling the field, because of its mathematical simplicity. It does not have a particular meaning and every other similar function could be used instead. If the measured curve  $M(H)$  coincides with a tanh function, the application of the inverse function  $\operatorname{arctanh}$  to the magnetization values generates a linear relation between scaled field and scaled magnetization. The scale transformation applied to the magnetization values is based on the following model for the relationship between the scaled field  $H^*$  and the measured magnetization  $M(H^*)$  in a demagnetization curve:

$$M(H^*) = M_{rs} \left[ 1 - \tanh \left( a(H^* - H_{1/2}^*) \right) \right] + M_0 \quad (5)$$

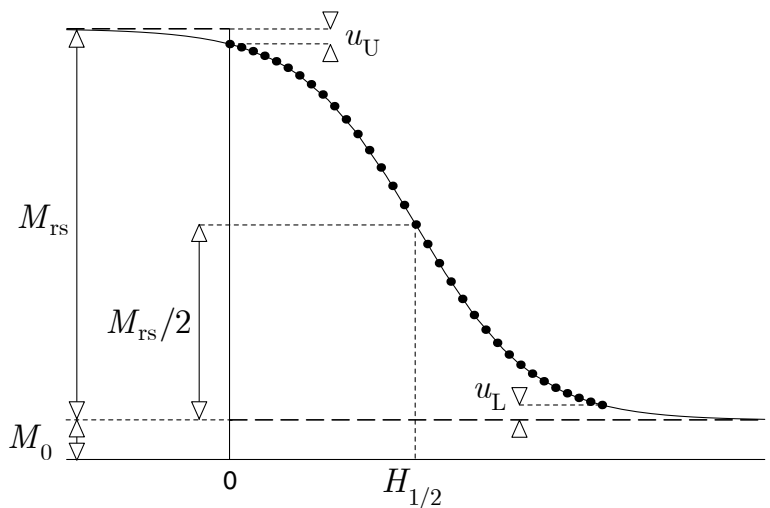
where  $M_{rs}$  has the physical meaning of a saturation remanence (if the measured curve is saturated at the highest field value),  $M_0$  has the physical meaning of an unremoveable magnetization ( $M_0 = 0$  if saturation of all minerals can be reached),  $H_{1/2}^*$  is the scaled median destructive field and  $a$  is a parameter that controls the steepness of the curve. The following scale transformation

$$M^* = \operatorname{artanh} \left( 1 - \frac{M - M_0}{M_{rs}} \right) \quad (6)$$

generates the linear relation  $M^* = a(H^* - H_{1/2}^*)$  between scaled field and scaled magnetization. The four parameters  $M_{rs}$ ,  $H_{1/2}^*$ ,  $M_0$  and  $a$  have to be chosen in a way that the scaled magnetization curve becomes as linear as possible. The program optimizes the parameters  $H_{1/2}^*$  and  $a$  automatically using a Levenberg-Marquard algorithm for non-linear fitting. The parameters  $M_0$  and  $M_0 + M_{rs}$  represent the asymptotic values of the magnetization curve. Their optimization is controlled by the user, since it was found that the optimization procedure can be very unstable with respect to these parameters. The scaled curve is represented together with a least-squares linear fitting. Deviation from the least-squares line can be minimized with an appropriate choice of  $M_0$  and  $M_0 + M_{rs}$ . Instead of  $M_0$  and  $M_0 + M_{rs}$  CODICA uses equivalent parameters, called *relative unsaturation degrees*, defined as:

$$\begin{aligned} u_U &= (M_0 + M_{rs} - M(0)) / M_{rs} \\ u_L &= (M(H_{\max}) - M_0) / M_{rs} \end{aligned} \quad (7)$$

where  $u_U$ ,  $u_L$  are the unsaturation degrees of the upper and lower part of  $M(H^*)$ , respectively. Furthermore,  $M(0)$  and  $M(H_{\max})$  are the initial and the final magnetization, respectively. In general, too large values of  $u_U$ ,  $u_L$  produce a flattening at the left and the right end of  $M^*(H^*)$ , respectively. On the other hand, too small values of  $u_U$ ,  $u_L$  produce a steepening of  $M^*(H^*)$  at the left and right end of the scaled curve, respectively. Random deviations from the least-squares line indicate the



**Fig. 3:** Definition of the parameters which characterize  $M(H^*)$ . Measured points are indicated by dots.

presence of measurement noise, systematic smooth deviations indicate a divergence of the measured curve from equation (5). Best results are achieved by analyzing samples where one magnetic com-



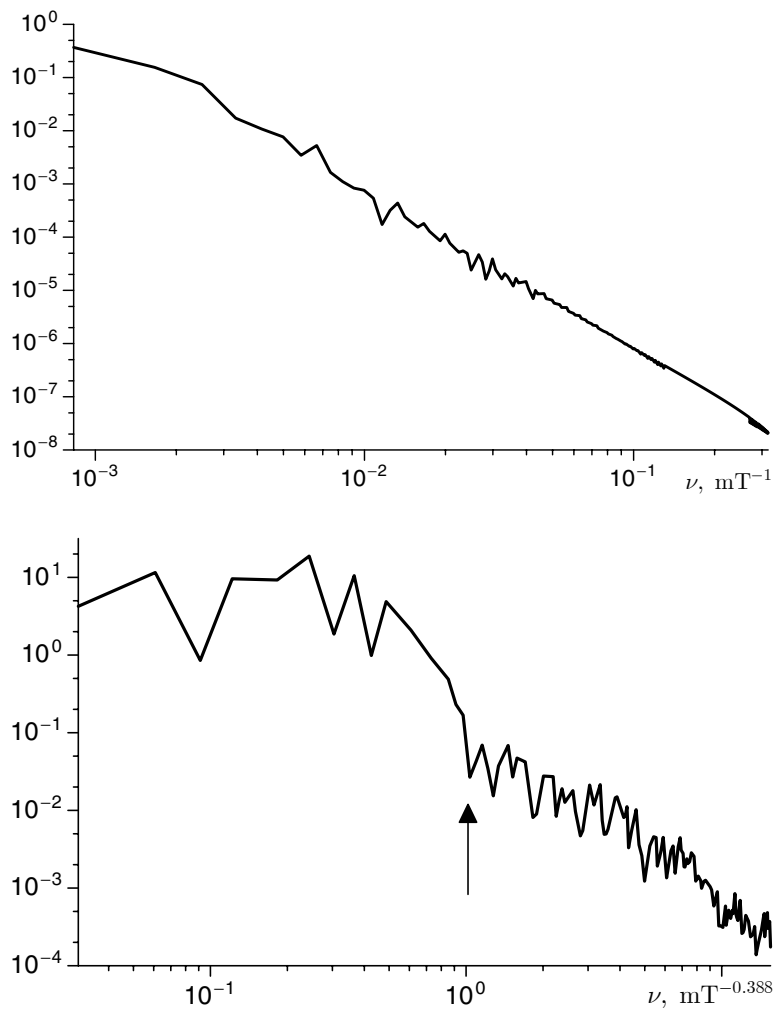
ponent is dominant or samples where different components have a wide range of overlapping coercivities. In both cases the choice of 5 independent parameters for the two scaling operations ( $p$ ,  $M_{rs}$ ,  $H_{1/2}$ ,  $M_0$  and  $a$ ) is sufficient to achieve an excellent linear relationship between scaled field and scaled magnetization. In case of components with drastically different coercivity ranges (i. e. magnetite and hematite) the scaling method is less effective, but in this case the separation of the different components is less critical as well, and can be performed even directly on the measured curve.

#### 4. Plotting the residuals (figure 2d)

Once the measured curve is scaled with respect to field and magnetization, the deviation of the scaled curve from the least-squares line is plotted. We will call this deviation the residuals curve. At this step, measurement errors are enormously enhanced, as it can be seen by comparing the residuals with the original measured curve.

#### 5. Scaling the residuals (figure 2e)

Generally, the residuals generate a sinusoidal curve, which is more or less „quenched“ at one end. As in step 2, the field axis can be rescaled with a power transformation in order to approach a quite regular sinusoidal curve. Variations of the amplitude are removed by normalizing the residuals with a mean value calculated from the first three Fourier coefficient of the residuals curve. After these steps, the rescaled residuals curve  $\varepsilon^*(H^*)$  is almost sinusoidal. Its Fourier spectrum is concentrated around a dominant wavelength (Fig. 4) so that a simple low-pass filter would easily remove the high-frequency measurement noise.



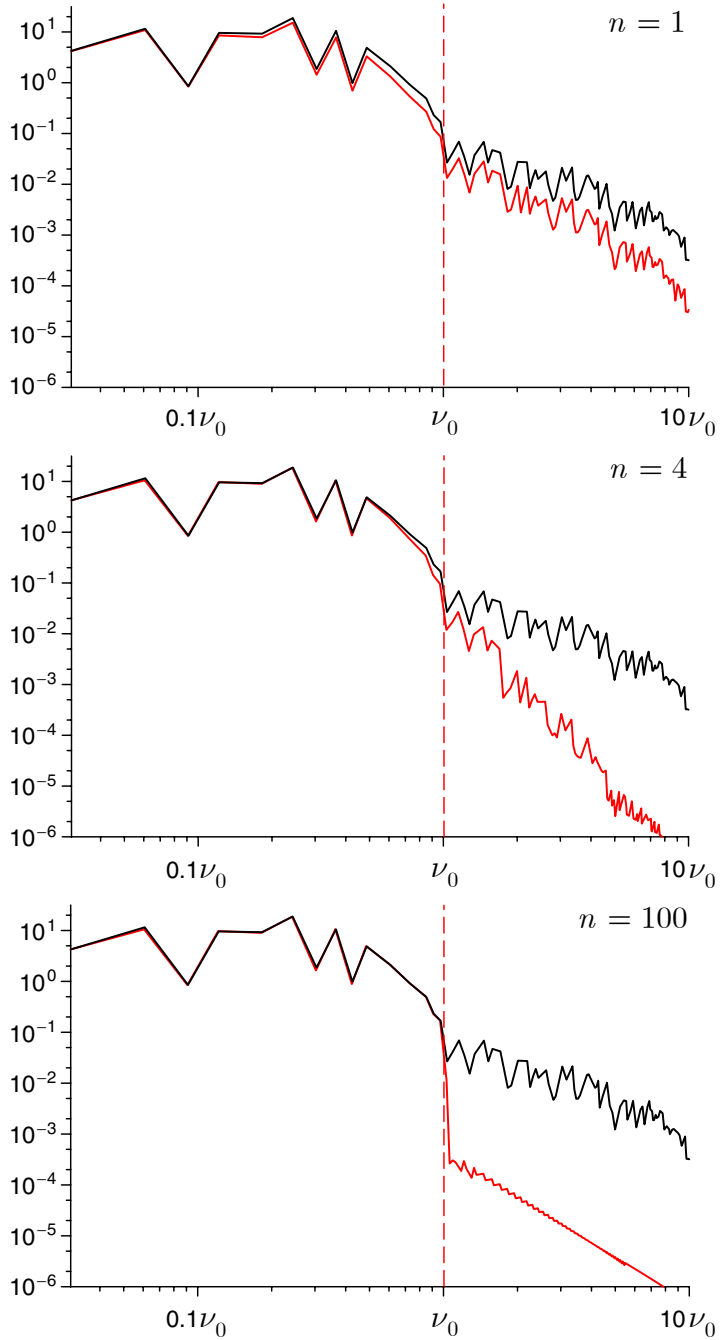
**Fig. 4:** *Fourier transform of the original measured curve  $M(H)$  (top) and of the rescaled residuals curve  $\varepsilon^*(H^*)$  (bottom). Few Fourier coefficients dominate the spectrum of  $\varepsilon^*(H^*)$  and the contribution of measurement noise is evident above the frequency indicated by the arrow. Contributions of those frequencies can be removed without affecting the shape of  $\varepsilon^*(H^*)$ . On the other hand, high frequencies are contributing significantly to the shape of  $M(H)$  and cannot be removed with a filter.*

6. Filtering the residuals (figure 2f)

The residuals curve is now ready to be filtered in order to remove the measurement noise. The filter applied by CODICA is a modified Butterworth low-pass filter, defined by:

$$\mathcal{L}(\nu, \nu_0, n) = \frac{1}{(1 + \nu_0^n / \nu^n)^{1/2n}} \tag{8}$$

where  $\nu$  is the frequency of the spectrum,  $\nu_0$  the so-called cutoff frequency, and  $n$  the order of the filter. The filter parameters  $\nu_0$  and  $n$  are chosen by the user. Details of the residuals curve with an extension smaller than  $1/\nu_0$  on the rescaled field axis are filtered out. The sharpness of the filter is controlled by its order  $n$ :  $n \rightarrow \infty$  gives a cutoff filter. Filters with a low order are inefficient, filters with a too high order produce undesired wiggling in the filtered curve. The filter order should be chosen so that the Fourier spectrum of the filtered curve is not sharply cutted around the cutoff frequency. The effect of the filter order on the Fourier transform is shown in Fig. 5.



**Fig. 5:** Effect of a Butterworth low-pass filter of order  $n$  on the Fourier transform of  $\varepsilon^*(H^*)$  with  $n = 1, 4, 100$  from top to bottom, and the same cutoff frequency  $\nu_0$  (indicated by a dashed line). The Fourier transform of  $\varepsilon^*(H^*)$  is plotted for the unfiltered curve (black), and for the filtered curve (red). A filter of order  $n = 1$  is inefficient and a filter of order  $n = 100$  produces a discontinuity in the Fourier transform.

The filter parameters should be chosen so that the measurement errors are suppressed without changing the global shape of the curve. This condition is met by choosing the smallest value of  $\nu_0$ , by which the difference between the filtered and the unfiltered curve attains the same maximal am-

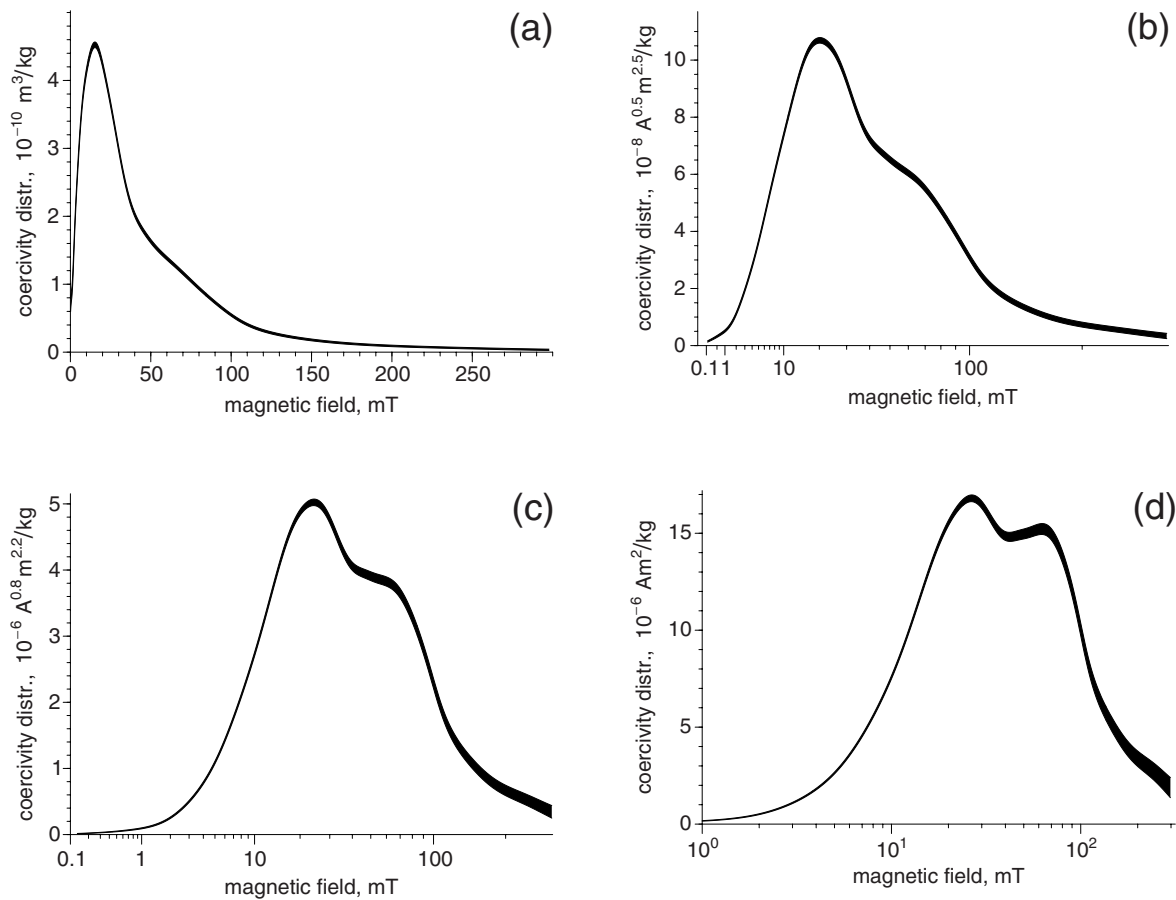
plitude as the estimated measurement errors. The choice of larger values of  $\nu_0$  leads to a coercivity spectrum that fits the measured curve better but still contains an unreduced noise component. The choice of smaller values of  $\nu_0$  may produce an alteration of the shape of the curve and suppress significant details.

### 7. Calculating the filtered demagnetization curve (figure 2g)

Now, the filtered residuals are converted back to the original curve by inverting steps 2 to 5 in reverse order. The result is a demagnetization curve, which is supposed to be free of measurement errors.

### 8. Calculating and plotting the coercivity distribution (figure 2h)

The coercivity distribution is now calculated as the absolute value of the filtered demagnetization curve obtained at point 7. The user can choose between a linear, a [logarithmic](#) and a [power](#) field scale to represent the results. The effect of the different field scales on the shape of a coercivity distribution is illustrated in Fig. 6.



**Fig. 6:** Effect of field rescaling on the shape of a coercivity distribution (sediment sample from Baldeggersee, Switzerland). Four different field scales have been chosen: (a) linear field scale, (b) power field scale according to equation (3) with exponent  $p = 0.5$ , (c) power field scale according to equation (3) with exponent  $p = 0.2$ , (d) logarithmic field scale.

---

The maximum error amplitude of the coercivity distribution is estimated by CODICA by comparing the measured curve with the filtered curve. The error estimation is displayed as an error band around the plotted coercivity distribution, and can be considered as a confidence interval of the plotted data.

## A program example

`In[1]:= <<Utilities`Codica``

[Load the program](#)

Program package Codica v.2.3 for Mathematica 3.0 and later versions.  
Copyright 2000-2003 by Ramon Egli. All rights reserved.

`In[2]:= Codica`

[Start the program](#)

Data from file  
C:/users/ramon/papers/fitting/SB32-arm.dat

[Enter file name](#)

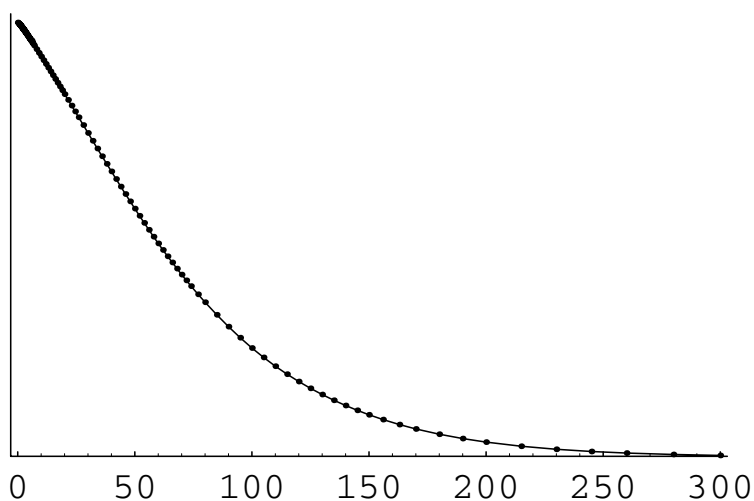
Checking double points...

[Enter the type of measurement](#)

Calculating data parameters...

[Enter sample weight](#)

Measurement curve:



Measurement type: cryo6

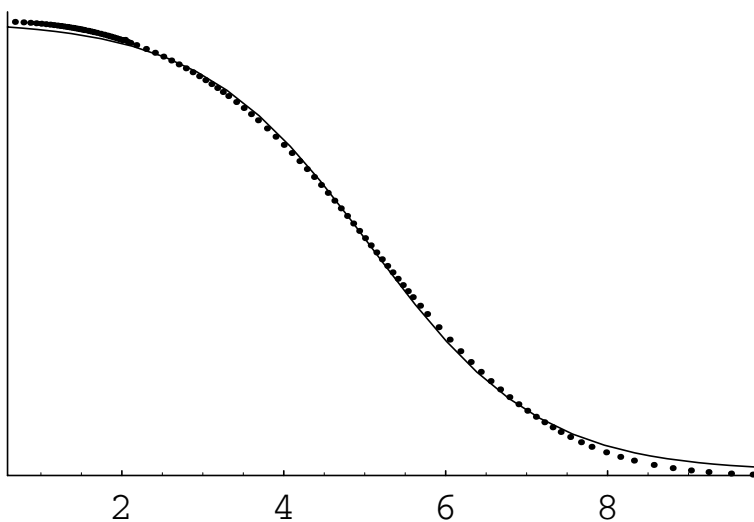
Assumed measurement errors,

field: 0.1 mT + 0.2%

magnetization: 2.3e-8 emu + 0%

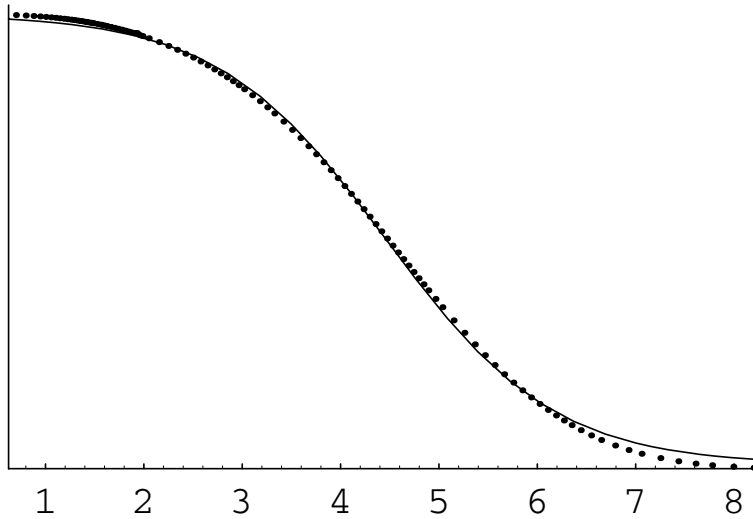
Measured curve with scaled field. Scaling exponent: 0.4

[Rescale the field axis](#)



```
Iteration #1: exponent 0.4, residual 0.0001990844346492842
Iteration #2: exponent 0.373225, residual 0.00019330045337513505
Iteration #3: exponent 0.368493, residual 0.0001932552874762185
Optimized exponent: 0.368889
Measured curve with scaled field. Scaling exponent: 0.368889
```

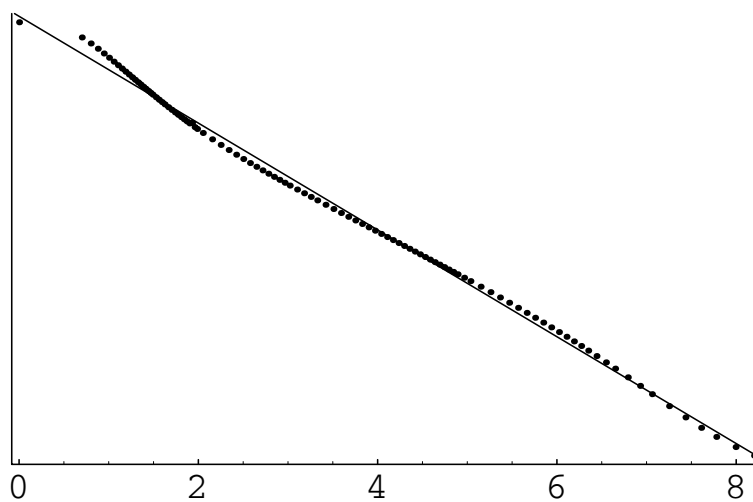
[Optimize the field axis rescaling](#)



```
Estimating the measurement error...
upper unsaturation degree: 0.00007028366343771037
lower unsaturation degree: 0.006469552438032343
```

```
Linearized measurement curve with:
upper unsaturation degree: 0.005
lower unsaturation degree: 0.017
```

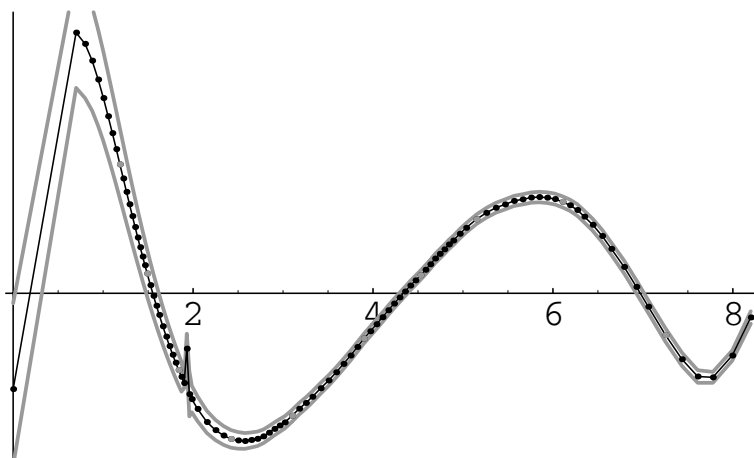
[Linearize the measurement curve](#)



[Optimize the linearization](#)

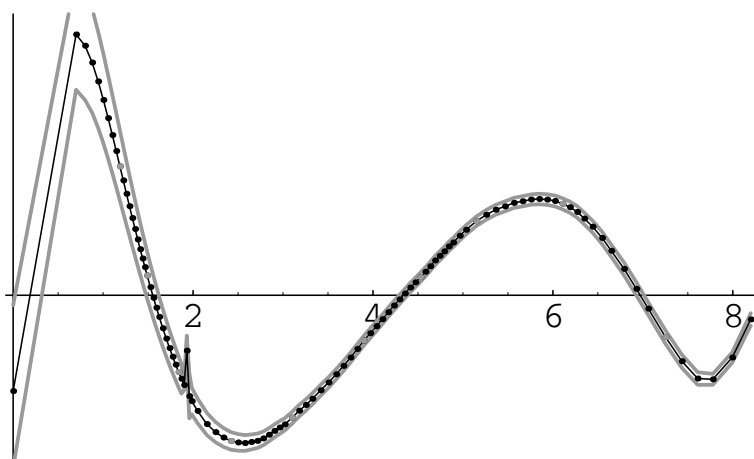
Residuals curve (every 10th point in grey).  
Measurement error margins in gray:

[Calculate the residuals](#)



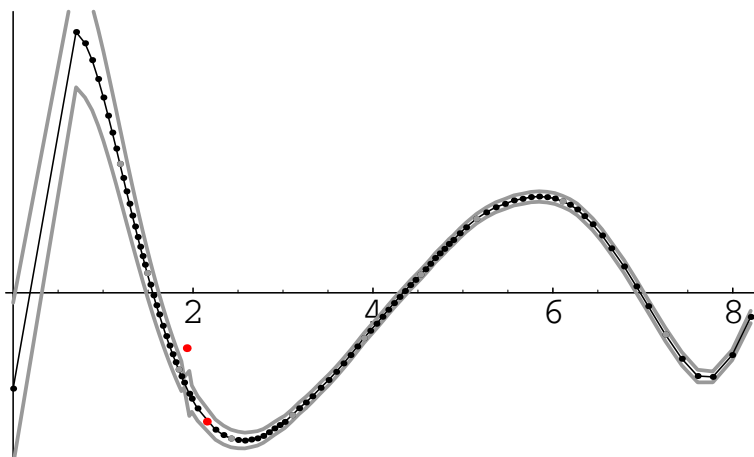
Residuals curve with rescaled field.  
Scaling exponent:  $0.368889 \times 1. = 0.368889$

[Rescale the residuals field axis](#)



Residuals curve with removed outlying points (in red):

[Remove outlying points](#)

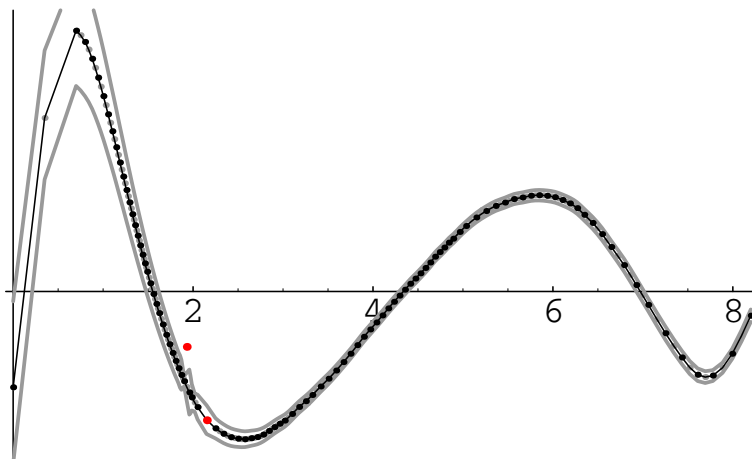


List of cancelled points (red):

Point #	field
33	5.9439
37	8.

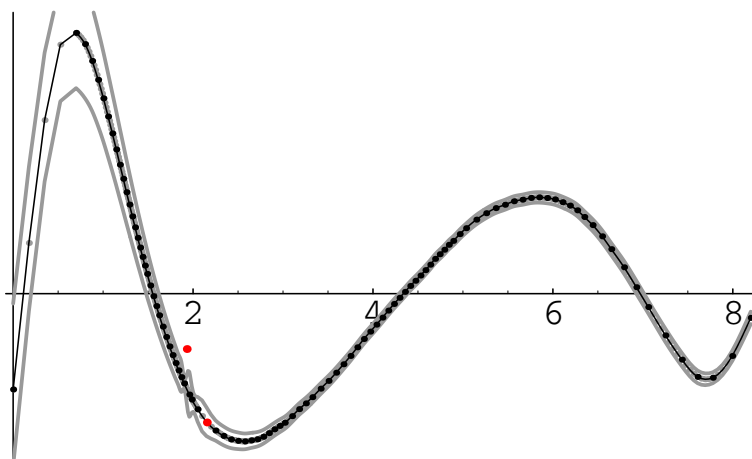
Interpolated residuals with minimum curvature,  
1 interpolated point (in grey) every measured point:

[Minimum curvature interpolation](#)



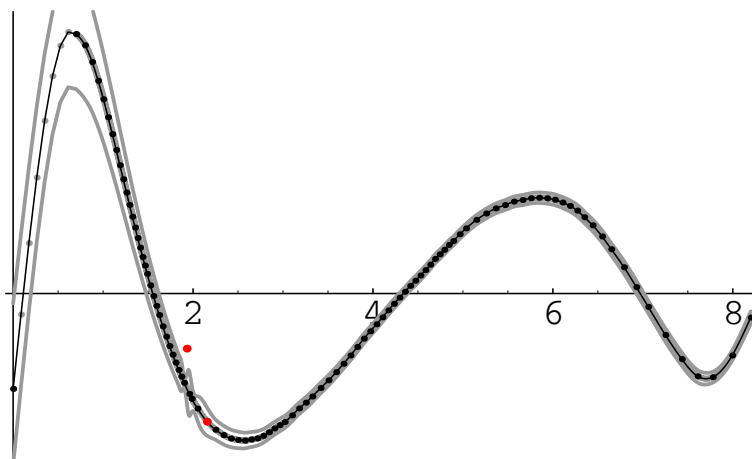
Interpolated residuals with minimum curvature,  
3 interpolated point (in grey) every measured point:

[Minimum curvature interpolation](#)



Interpolated residuals with minimum curvature,  
7 interpolated point (in grey) every measured point:

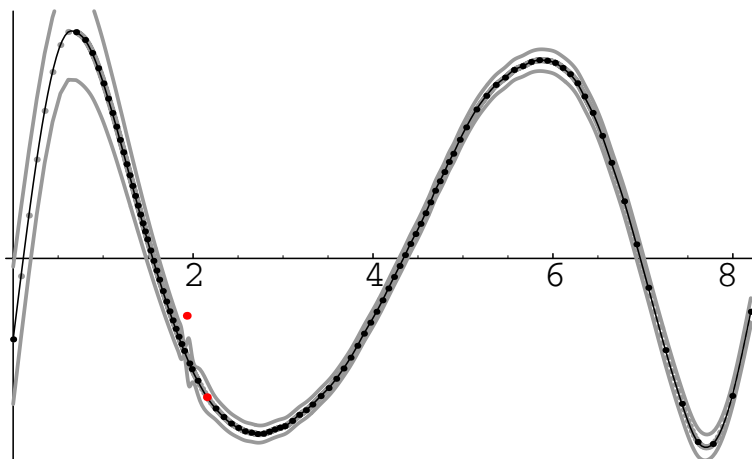
[Minimum curvature interpolation](#)





Normalize the residuals to approach a sinusoidal curve...

[Normalize the residuals](#)

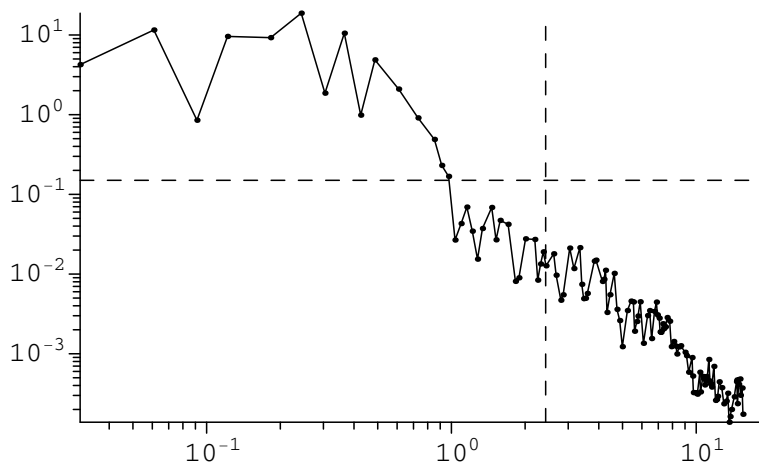


Fourier spectrum of the residuals:

[Fourier spectrum of the residuals](#)

Dashed vertical lines are the lower/upper limit of the aliasing frequency.

Dashed horizontal line is the measurement noise level of the residuals.

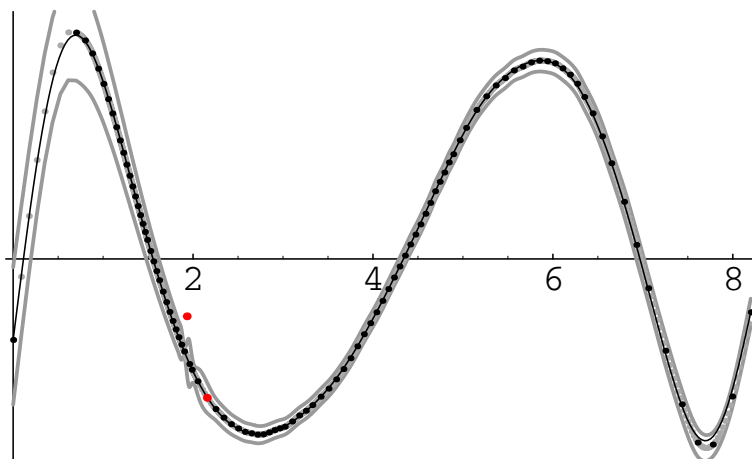


Parameters of the Butterworth low pass filter:

[Apply a low-pass filter to the residuals](#)

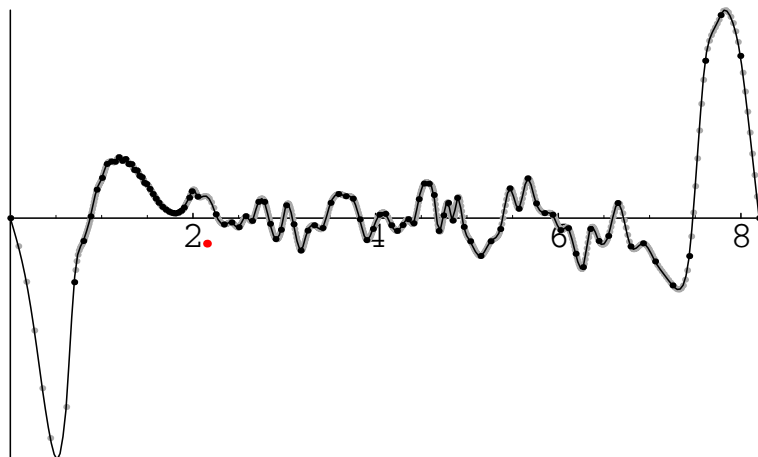
cutoff frequency: 1  
order : 4

Filtered residuals (solid line):



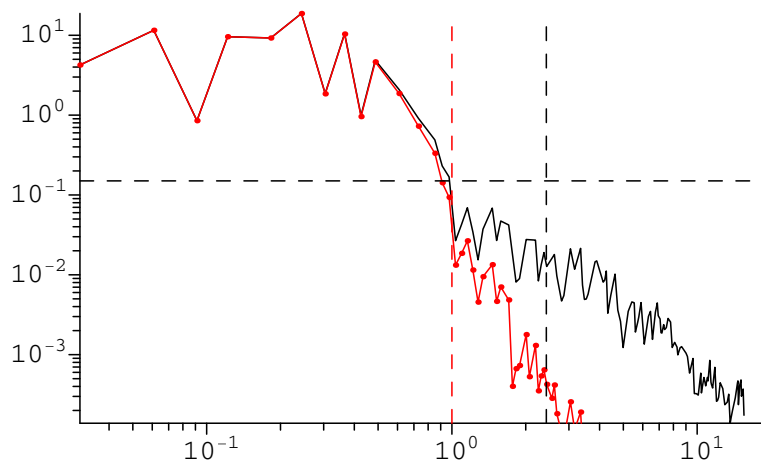
Difference between filtered and unfiltered residuals;  
this curve should represent the measurement errors.

[Measurement errors](#)



Fourier spectrum of the filtered residuals.  
Dashed red line is the corner frequency of the filter:

[Fourier spectrum of the filtered residuals](#)



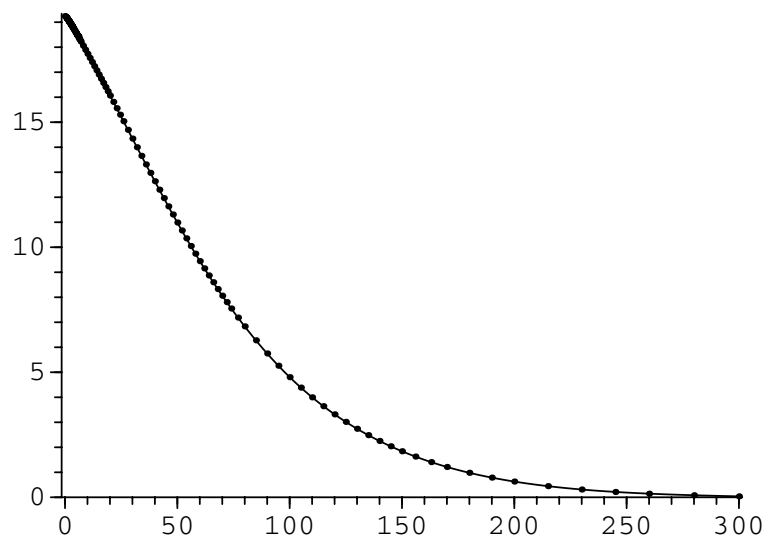
Number of independent parameters for a component analysis: 16

Number of filtering iterations: 1

Measured points and fitted curve.

Units:  $10^0$  mT  $10^{-3}$  emu

[Plot the filtered measurements](#)

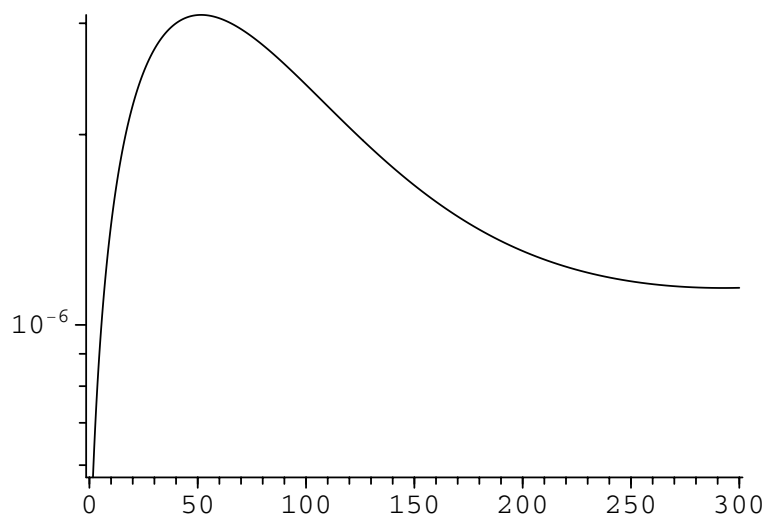


Estimating the fitting error. Please wait...

Estimated maximal absolute error of the fitted curve:

Units:  $10^0$  mT emu

[Plot the measurement errors](#)



Saving the fitted measurements in file: SB32-arm.cum

[Saving the fitted measurements](#)

1. column: field in mT
2. column: magnetization in emu
3. column: absolute fitting error in emu

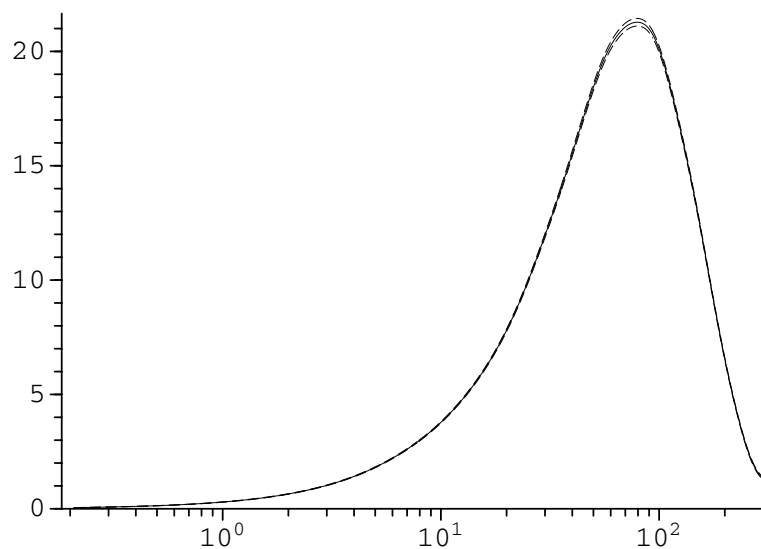
Estimating the spectral error. Please wait...

Spectrum with logarithmic field scale.

[Plotting a spectrum on a log-scale](#)

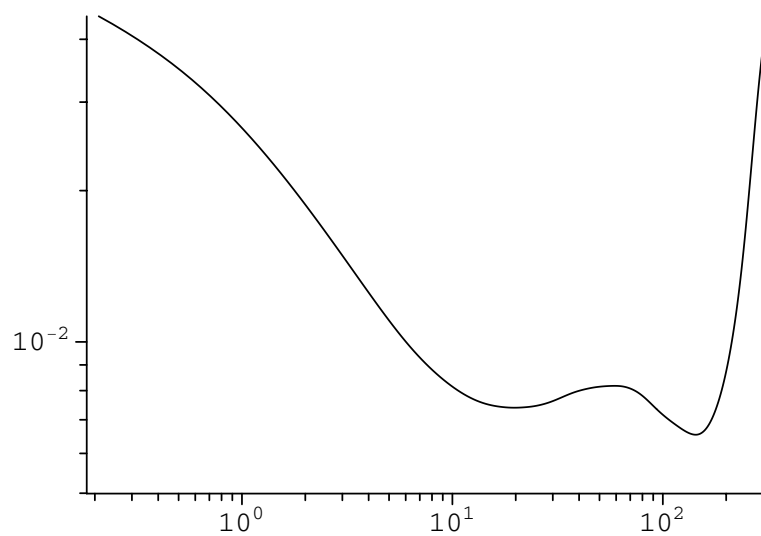
Dashed lines represent the lower and upper confidence limit.

Units: mT  $10^{-3}$  emu



Estimated maximal relative error of the spectrum:

[Relative error of a coercivity distribution](#)



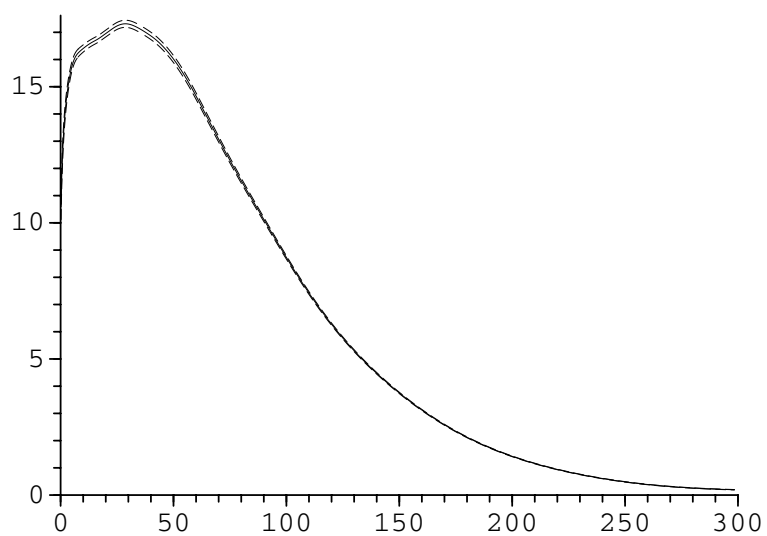
Saving logarithmic spectrum in file: SB32-arm.slog

[Saving the coercivity distribution](#)

1. column: field in mT
2. column: magnetization per field unit, in emu
3. column: relative fitting error

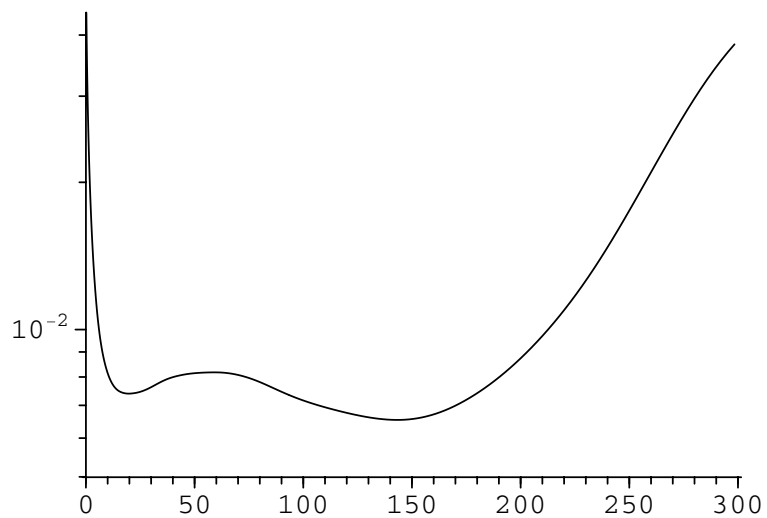
Spectrum with linear field scale (solid line).  
Dashed lines represent the lower and upper confidence limit.  
Units:  $10^0$  mT  $10^{-5}$  emu/mT

[Plotting a spectrum on a linear scale](#)



Estimated maximal relative error of the spectrum:

[Relative error of a coercivity distribution](#)



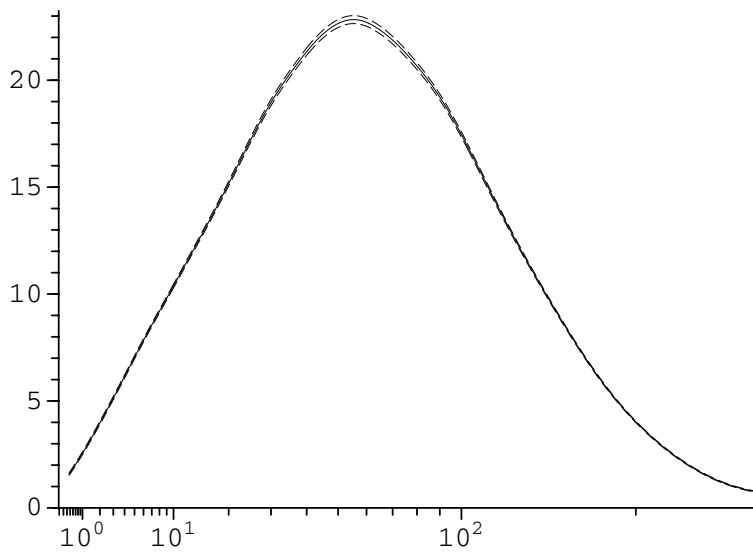
Saving a linear spectrum in file: SB32-arm.slin

[Saving the coercivity distribution](#)

1. column: field in mT
2. column: magnetization per field unit, in emu/mT
3. column: relative fitting error

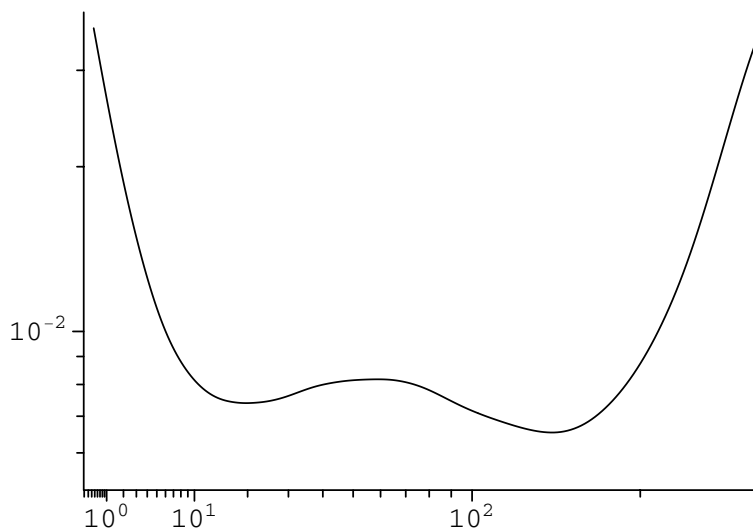
Spectrum with power field scale (solid line).  
 Dashed lines represent the lower and upper confidence limit.  
 Units:        mT         $10^{-4}$  emu/mT<sup>0.5</sup>

[Plotting a spectrum on a power-scale](#)



Estimated maximal relative error of the spectrum:

[Relative error of a coercivity distribution](#)



Saving a power spectrum in file: SB32-arm.spow

[Saving the coercivity distribution](#)

1. column: field in mT
2. column: magnetization per field unit, in emu/mT<sup>0.5</sup>
3. column: relative fitting error

## Installing CODICA

To install CODICA, you need to add the files `codica.m`, `codicaset`s and `components.txt` to *Mathematica* v.3.0 or later running on a PC. The file `codica.m` contains the source code of CODICA and GECA, and is a so-called *Mathematica package*. The file `codicaset`s contains informations about the [measurement parameters](#) (units, errors). The file `components.txt` contains informations about the component analysis results of GECA.

Copy `codica.m`, `codicaset`s and `components.txt` to:

```
C:/.../Programs/Wolfram Research/Mathematica/4.1/AddOns/StandardPackages/Utilities
```

Cursive directories depend upon the operating system and the installed version of *Mathematica*. If you copy the files `codicaset`s and `components.txt` from a CD, you have then to change the file properties by removing the read-only attributes, otherwise CODICA will not have access to these files.

## Loading and running CODICA

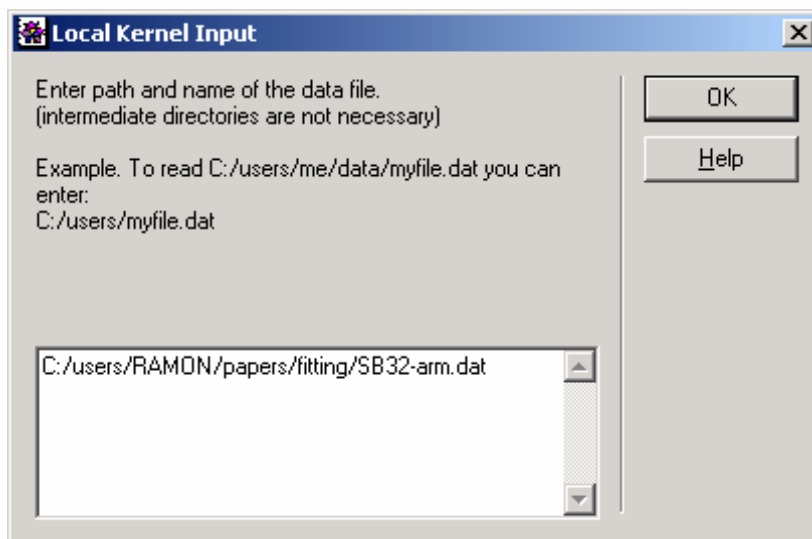
To run CODICA, open a new *Mathematica* notebook by clicking on the *Mathematica* program icon. Type `<<Utilities`Codica`` on the input prompt `In[]` and press the keys *Shift + Enter* to load CODICA. On the next input prompt, type `Codica` and press the keys *Shift + Enter* to start CODICA. From now on, the program asks you to enter specific commands step by step. In the following, all CODICA commands are explained in order of appearance.

[Back to the program](#)

### Enter the name of the data file

The prompt window on the right asks you to enter the name on the file which contains the measurement data. Type the path of the data file. You can skip intermediate directories if other files with the same name are not stored. The data file should be an ASCII file with two columns of numbers separated by spaces or tabulators. The file should not contain comment lines or text in general. The first column is the applied field, the second column is the magnetic moment or the magnetization.

[Back to the program](#)



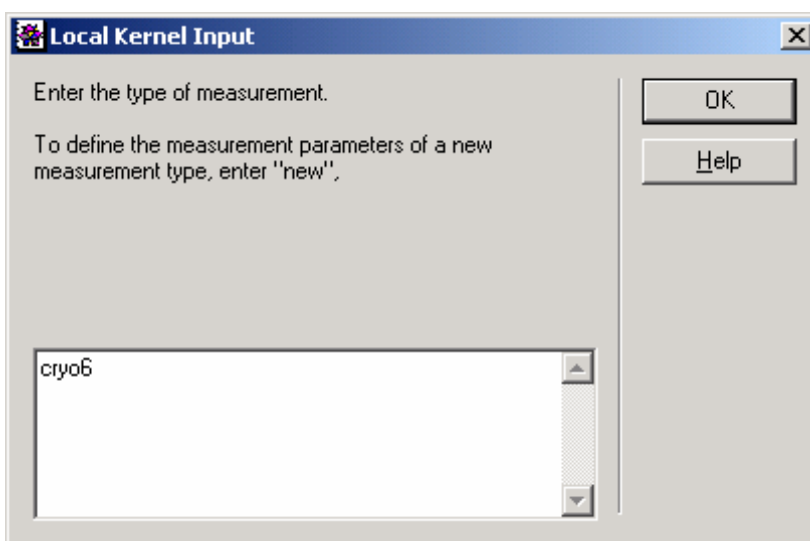
### Enter the type of measurement

CODICA needs some informations about [measurement parameters](#) like the field and the magnetization units and the measurement errors. To avoid you entering every time these parameters, CODICA stores them in a file. The first time you analyze a given type of measurement with CODICA, you can [store the measurement parameters](#) and enter a name to recall them whenever you want. Every time you run CODICA you will be asked for the measurement parameters with the prompt window on the right. You can enter the name by which you stored them, or you can type “new” to define the parameters of a new type of measurement. Two measurement types (cryo1 and cryo6) are already stored as a default example. They refer to an automatic AF demagnetization system of 2G enterprises. Type “cryo1” denotes a single measurement, type “cryo6” denotes the mean of six identical AF demagnetization curves. You can view these parameters by opening the program file [codicaset](#)s with a text editor. Each measurement type is a list with following parameters:

```

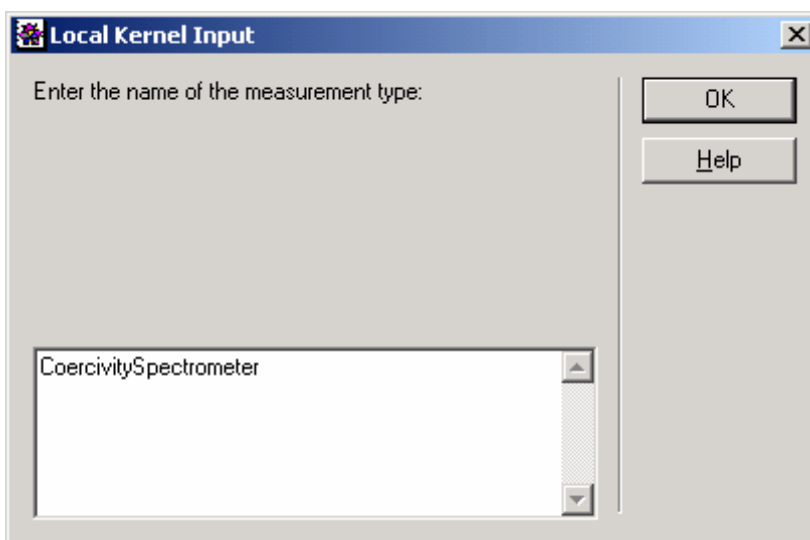
{"type", {"filed unit", "magnetic moment unit"},
  {absolute error of the field, relative error of the field},
  {absolute error of the measurement, relative error of the measurement}}

```



### Store the measurement parameters

To store new measurement parameters type “new” in the prompt window for entering the [type of measurement](#). You will be asked to enter a new name with which you can recall these measurement parameters next time. You can also [change the parameters](#) of an already defined measurement type.





Enter the appropriate units for the field and the magnetization values to store for the measurement type “CoercivitySpectrometer”.

The dialog box is titled "Local Kernel Input" and contains the following text: "Enter the units of field and magnetization: {field,magnetization}". Below this is a text input field containing the value "{mT,emu}". On the right side, there are two buttons: "OK" and "Help".

Enter the absolute and relative error of the applied field. If you do not know these parameter, enter indicative values, e.g. 1 mT for the absolute error and 1% for the relative error. The program will use these parameters to help you during the data elaboration, but *not* for [estimating the error](#) which affects the final result.

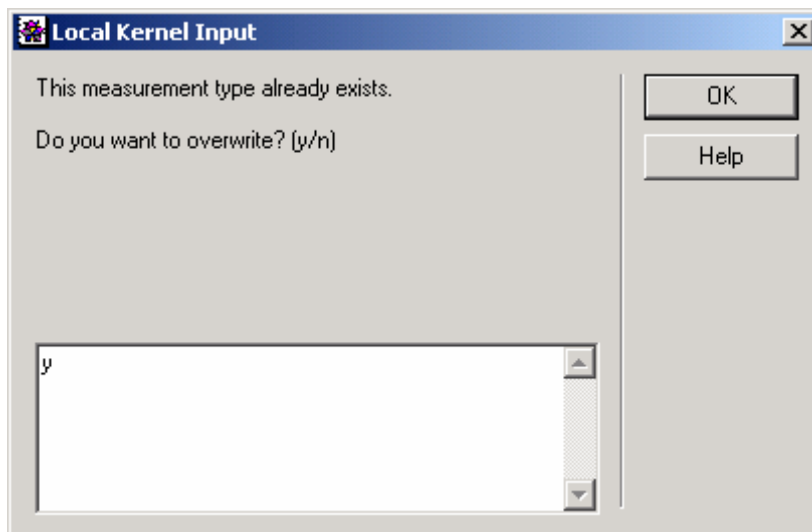
The dialog box is titled "Local Kernel Input" and contains the following text: "Enter the absolute and relative error {absolute,relative} of the applied field." Below this is a text input field containing the value "{0.5,1}". On the right side, there are two buttons: "OK" and "Help".

Enter the absolute and relative measurement error. If you do not know these parameter, enter indicative values, e.g. the lower detection limit of the magnetometer for the absolute error and 1% for the relative error. The program will use these parameters to help you during the data elaboration, but *not* for [estimating the error](#) which affects the final result. The measurement error refers to the measurement of a *magnetic moment*, regardless of how the measurements are normalized in the original data file.

The dialog box is titled "Local Kernel Input" and contains the following text: "Enter the absolute and relative measurement error {absolute,relative}". Below this is a text input field containing the value "{0.00001,1}". On the right side, there are two buttons: "OK" and "Help".

If you want to change the parameters of an existing measurement type, [enter its name](#) when you are asked. The prompt window on the right will appear. If you really want to overwrite the parameters of this measurement type, enter “y”, and you will be asked for [new parameters](#).

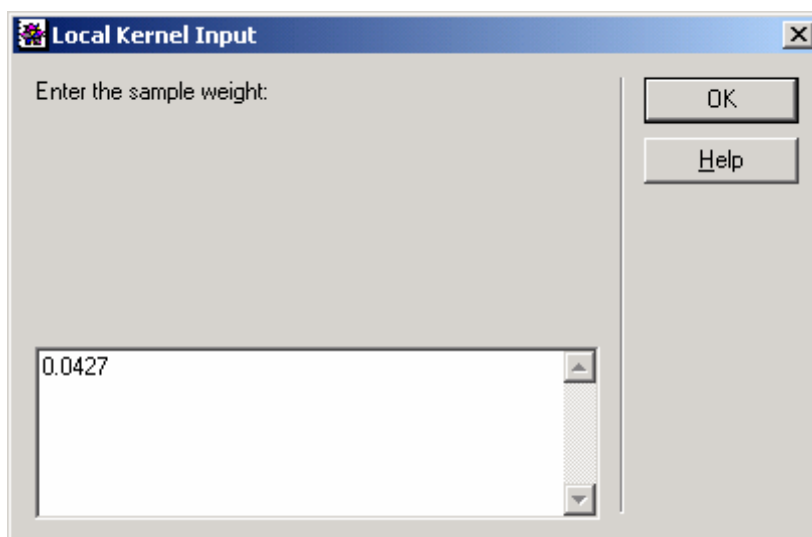
[Back to the program](#)



### Enter the sample weight

The [measurement error](#) is related to the magnetic moment measured by the magnetometer. In general it does not have the same unit as the magnetization values of your data file. To calculate a magnetization, the measured magnetic moment is divided by a normalizing factor, generally the weight or the volume of the sample. Enter this factor in the prompt window on the right to normalize the measurement error as well. If the measurement data are not normalized, type “1” in the prompt window.

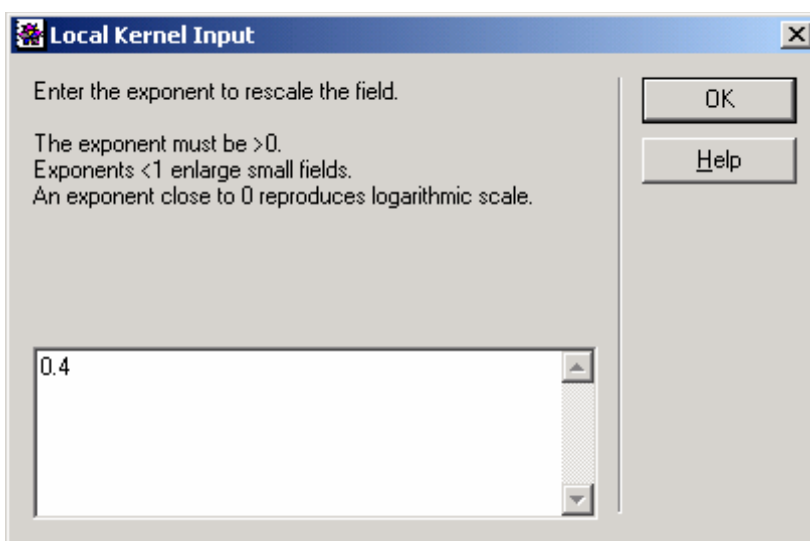
[Back to the program](#)



### Rescale the field axis

Generally, acquisition/demagnetization curves have a sigmoidal shape. However, they are not symmetrical, and are generally heavy-tailed at high fields. As a first step, CODICA makes the curve symmetrical by applying a rescaling function to the field axis. In case of lognormal coercivity distributions, the measured curve becomes symmetrical on a logarithmic field scale. CODICA uses a [power transformation](#) to rescale the field axis. The exponent  $p$  of the power transformation offers a set of intermediate field scales between the original linear scale ( $p = 1$ ) and a logarithmic scale ( $p \rightarrow 0$ ). An appropriate value of  $p$  can be chosen, so that the scaled curve reaches maximum symmetry. The symmetry of the curve is compared with a [reference sigmoidal curve](#) (plotted as a solid line). You can enter different values of  $p$  until you reach a satisfying result.

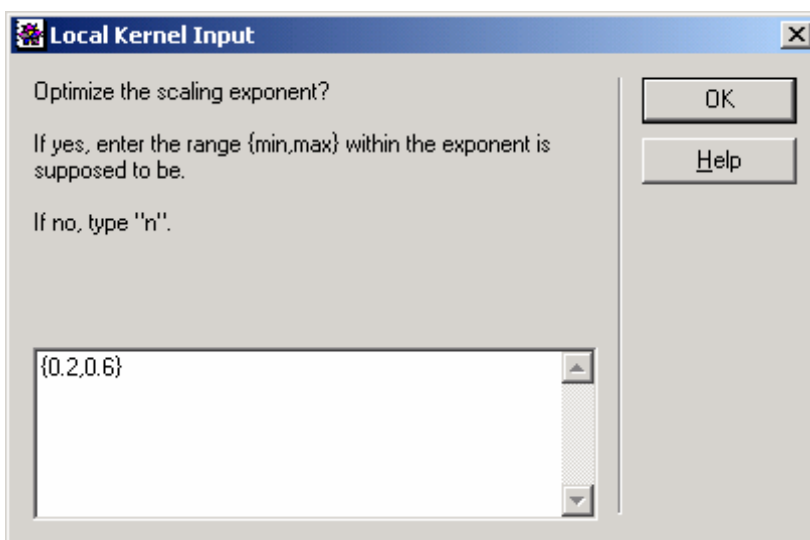
[Back to the program](#)



### Optimize the field axis rescaling

CODICA can automatically optimize the field scaling exponent  $p$  so that the scaled curve reaches a maximum symmetry. Enter the range within CODICA should perform the optimization. If the data is very noisy, the optimization may not be possible. In this case a warning message appears and you are asked to [enter a scaling exponent](#).

[Back to the program](#)

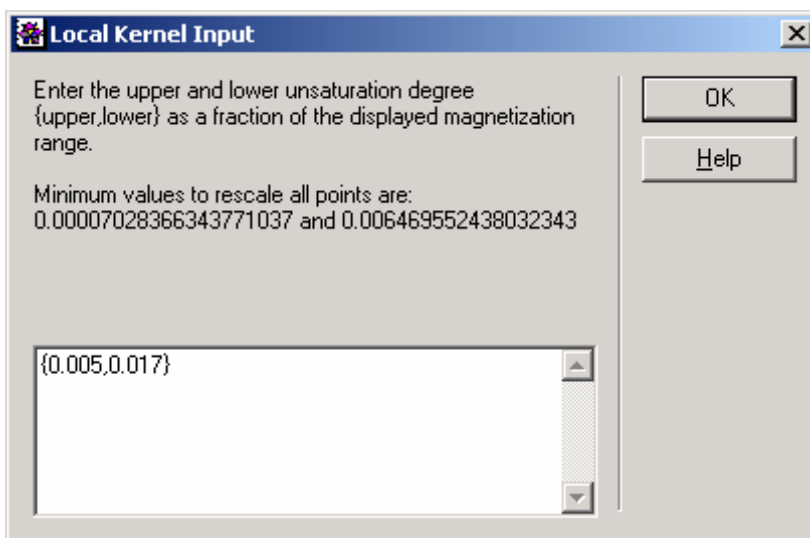


### Linearize the measurement curve

As a second step, CODICA linearizes the scaled measurements. [Linearization](#) is performed by modeling the scaled measurements with a symmetric [sigmoidal curve](#). The shape of this curve is controlled by following [parameters](#):

- upper asymptotic magnetization
- lower asymptotic magnetization
- median destructive field
- dispersion parameter
- vertical offset

Median destructive field, dispersion parameter and vertical offset can be easily calculated by the program.



An estimation of the upper/lower asymptotic magnetization is more critical, since these values are never reached with real measurements. A good approximation is obtained by measuring a sample in the most wide field range made possible by the instrument. If the field range is not wide enough, the rescaled curve does not approach the asymptotic magnetization values. The gap between the measurements and the asymptotic magnetization values is called [upper/lower unsaturation degree](#). The upper/lower unsaturation degree is expressed as the difference between the absolute maximum/minimum values of the measured curve and the (unknown) upper/lower asymptotic magnetization, normalized by the total magnetization of the sample. You are asked to enter these parameters in the prompt window shown above, where minimum values are suggested. You should not enter smaller values than the suggested limits, otherwise the linearization cannot be performed with all points and a warning message appears. You may enter smaller values if the measurements are very noisy.

To guess suitable values of the unsaturation degrees, start by doubling the minimum values given in the prompt window (0.0002 and 0.01 in the example shown above). A linearized curve calculated with these values will be shown, together with a best-fit line. If the upper/lower unsaturation degree is too small, a steepening of the linearized curve results at the left/right end. Vice-versa, a flattening of the linearized curve results from a large unsaturation degree. You are asked to reenter the upper/lower unsaturation degree until you decide to accept the resulting linearization. This operation needs typically 4-10 trials to an experienced user.

[Back to the program](#)

### Optimize the linearization (not shown)

You can let CODICA optimizing the upper/lower unsaturation degree to minimize the difference between the linearized curve and a best-fit line. In this case you will be asked to enter a range for the upper and the lower unsaturation degree: CODICA performs the optimization within these ranges. An automatic optimization is usually not needed, since the manual linearization performed by an experienced user is already very close to the optimum.

[Back to the program](#)

### Calculate the residuals

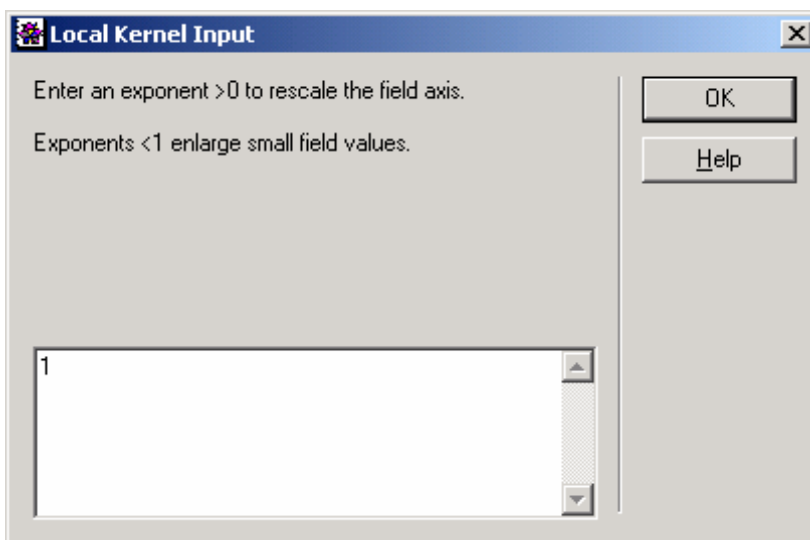
The best-fit line is now subtracted from the linearized curve. The resulting curve will be called [residuals curve](#) in the following. The residuals contain both detailed informations about the shape of the acquisition/demagnetization curve, and a noise signal produced by the measurement errors. The latter is highly enhanced by the rescaling procedure, and outliers can be easily identified. The noise-free residuals curve has generally a sinusoidal shape. The estimated maximum measurement error is plotted in the form of a band around the residuals curve. It should help the user with the further elaboration steps. The error estimation is based on the [measurement parameters](#) entered at the beginning. These parameters are not necessarily correct; nevertheless, they should provide a rough estimation of the measurement errors.

[Back to the program](#)

### Rescale the field axis of the residuals

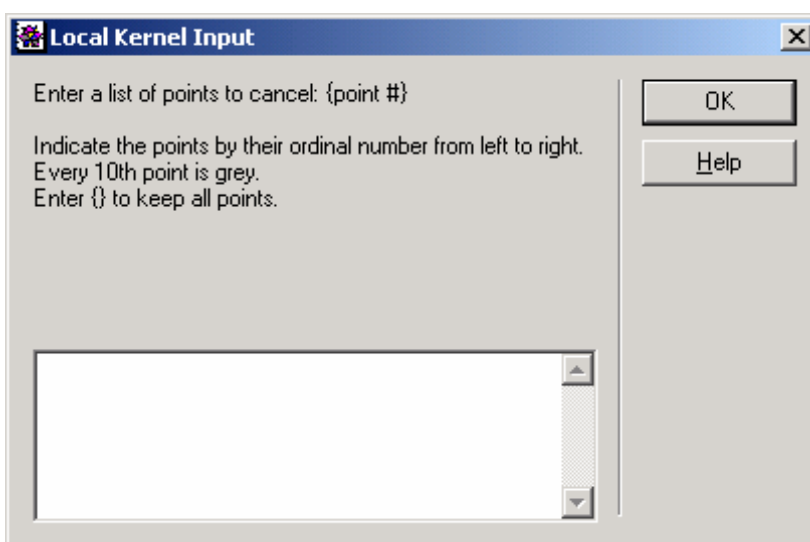
Often, the residuals curve is similar to a sinusoidal curve “stretched” on one side. An appropriate rescaling of the field axis can bring the residuals closer to a sinusoidal curve. The rescaling is based on a [power transformation](#), exactly as the [field axis rescaling](#) of the measurement curve. You can enter a value for the rescaling exponent  $p$  until a satisfying result is attempted. It is important that the rescaled residuals approach a sinusoidal curve. In this way, measurement errors can be filtered more efficiently.

[Back to the program](#)



### Remove outlying points

Often, the measurements contain some points affected by a particularly large error. The most evident among them can already be identified on the acquisition/demagnetization curve. Other outliers can be recognized only in the residuals curve. Notice that the outliers of the program example cannot be recognized on the [measurement curve](#). In the prompt window on the right you can enter a list of outlying points to be removed. You indicate each point with a number, 1 being the first point on the left. To facili-

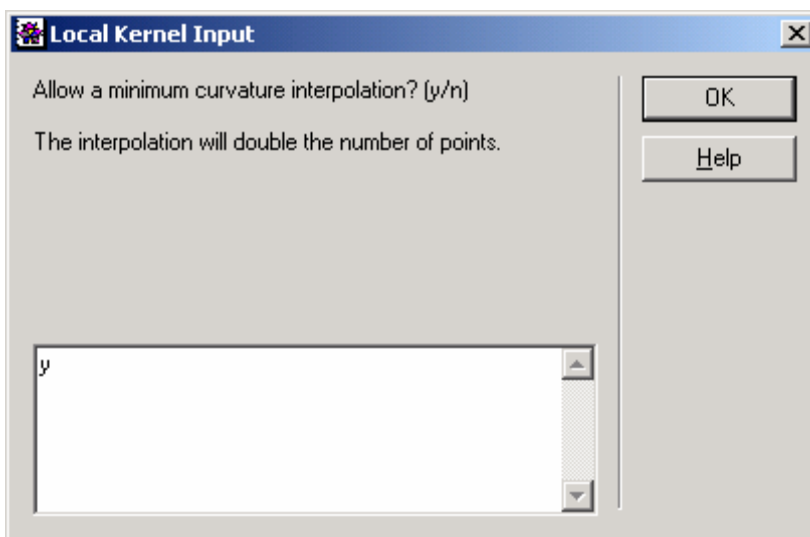


tate the identification of an outlier, every 10<sup>th</sup> point is drawn in grey. In the program example, points 33 and 37 are removed.

[Back to the program](#)

### Minimum curvature interpolation

The rescaling procedure described above generally produces a “stretching” of the residuals curve on the left end, where the curve is poorly defined, as it can be seen in the program example. For further elaborations, the residuals curve needs to be interpolated and resampled at regular intervals. Interpolation is performed by adding additional points to the residuals. Each point is added in the middle of two measurements in a way that minimizes the curvature of the residuals. You can repeat this operations until the



residuals curve looks smooth. Generally, minimum curvature interpolation is performed one to three times. It is important to know that minimum curvature interpolation, as any other interpolation method, does not add any information to the original curve, and is performed for numerical reasons only. You should avoid minimum curvature interpolation if the residual curve is dominated by measurement errors and has an irregular aspect.

[Back to the program](#)

### Normalize the residuals

Generally, at this stage the residuals curve is quite similar to a sinusoidal curve. However, the amplitude of the residual curve is not constant over the entire field range. Amplitude variations are removed by normalizing the residuals with a mean value calculated with the first three Fourier coefficients of the residuals curve. After this correction, the noise-free component of the residuals should be almost equivalent to a sinusoidal curve. In this way, measurement errors can be filtered very efficiently.

[Back to the program](#)

### Fourier spectrum of the residuals

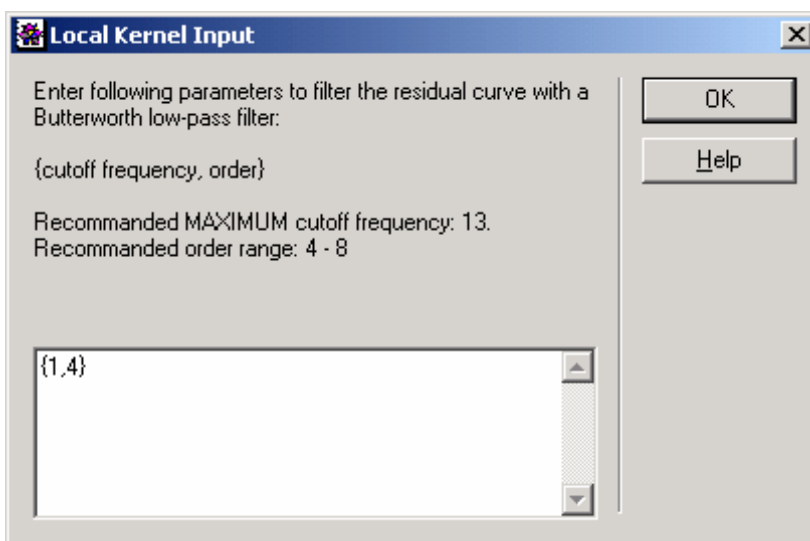
A Fourier spectrum of the residuals is calculated with FFT. To avoid windowing effects, the residuals curve is previously continued on the left and right side, whereby continuity is guaranteed up to the second derivative over the entire range. On the Fourier transform, the minimum and maximum aliasing frequencies are highlighted with dashed lines. The minimum/maximum aliasing frequencies are calculated as the reciprocal of the maximum/minimum separation between the measurement points of the residual curve. Additional points generated with a [minimum curvature interpolation](#) are not considered. Frequencies smaller than the minimum aliasing frequency carry informations about the entire range of the residuals curve. The frequencies between the minimum and the maximum aliasing frequency carry informations only about some parts of the residuals curve. Finally, frequen-

cies larger than the maximum aliasing frequency do not carry *any* real information about the residuals, and have to be considered as interpolation artefacts. The noise content of the residuals is indicated by a horizontal dashed line, whereby a white noise is assumed, whose amplitude is calculated from the [measurement parameters](#) entered at the beginning. Fourier coefficient below the noise limit are dominated by the measurement errors. Notice that the calculated noise content is not necessarily correct, nevertheless, it should provide a rough estimation of the measurement errors.

[Back to the program](#)

### Apply a low-pass filter to the residuals

Measurement errors that contribute to the residuals with a white noise signal can be removed with a low-pass filter. A [Butterworth low pass filter](#) is used for this purpose. You are asked to enter the parameters of this filter, e. g. the cutoff frequency and the order. The cutoff frequency should be chosen *below* the minimum aliasing frequency. If the estimate of the measurement errors given with the [measurement parameters](#) is correct, you should identify the cutoff frequency with the frequency at which the Fourier



spectrum of the residuals crosses the horizontal dashed line which indicates the amplitude of the measurement errors in the [last plot](#). The sharpness of the Butterworth filter is controlled by its order. Filters with a low order are inefficient, filters with a too high order produce undesired wiggling in the filtered curve. The filter order should be chosen so that the Fourier spectrum of the filtered curve is not sharply cutted around the cutoff frequency. [Click here](#) to see correct and incorrect filtering examples. The filtered residuals (solid line) are plotted together with the unfiltered residuals (dots). The filtered residuals should be compatible with the error margin defined by a gray band around the measured points.

[Back to the program](#)

### Measurement errors

Measurement errors are plotted as the difference between the low-pass filtered and the unfiltered residuals. If the parameters of the low-pass filter were correctly chosen, the measurement errors have a chaotic appearance along the entire range. These measurement errors are used for the final error estimation performed by CODICA, regardless of the values given as [measurement parameters](#). You can use the final error estimation performed by CODICA to reenter a better estimation of the measurement errors in the measurement parameters.

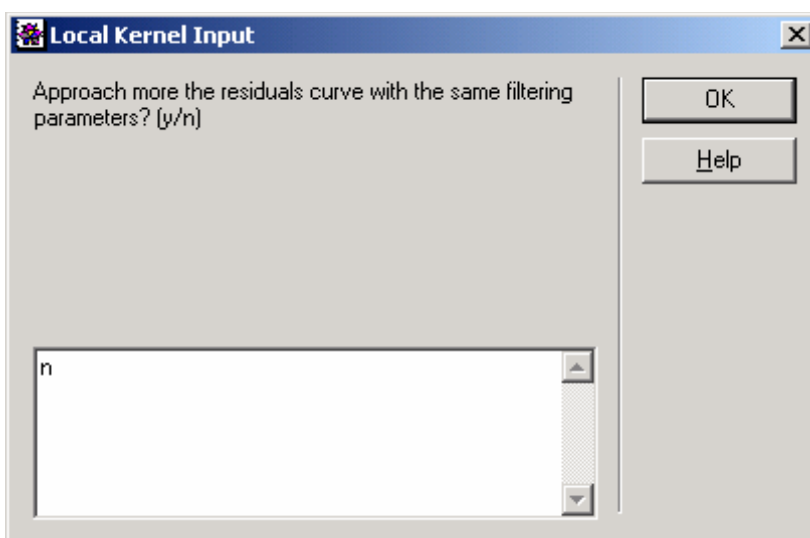
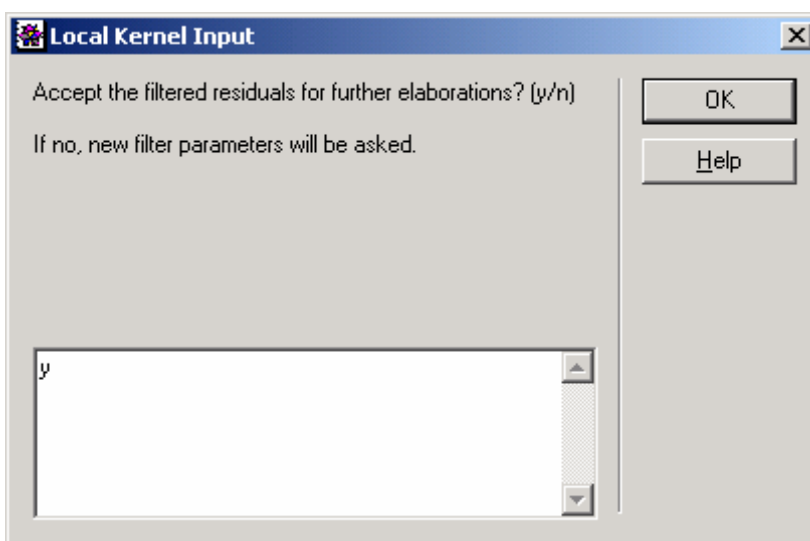
[Back to the program](#)

### Fourier spectrum of the filtered residuals

The Fourier transform of the filtered residuals (red) is plotted together with the Fourier spectrum of the unfiltered residuals (black). A dashed line indicates the cutoff frequency of the low-pass filter. The Fourier spectrum of the filtered residuals should not show a sharp drop around the cutoff frequency, otherwise the chosen low-pass filter order was too high. [Click here](#) to see Fourier transforms of correctly and incorrectly filtered residuals. The number of Fourier coefficients (red points) below the cutoff frequency, divided by five, is a good indicator for the maximum number of magnetic components which can be inferred in the resulting coercivity distribution. In the [program example](#), there are 16 Fourier coefficients below the cutoff frequency. If the coercivity distribution of each magnetic component is described by 5 parameters (amplitude, mean, standard deviation, skewness and kurtosis), the 16 coefficients are determined by a maximum of 3 magnetic components. The atmospheric particulate sample measured

for this example contains two magnetic components: natural dust and motor vehicles combustion products. If you are satisfied with the results of the filtered residual, you can accept them for further calculations. Otherwise you can reenter the [low-pass filter parameters](#) and repeat the filtering procedure. If you are not able to remove the measurement errors in a satisfactory way by choosing an appropriate cutoff frequency, you can tell CODICA to low-pass filter the resulting [measurement errors curve](#) and add it to the filtered residuals. In this way you can fit the measured residuals with a kind of fractal technique which may give better results for the particular case of a residuals curve which differs strongly from a sinusoidal. This should never occur for measurements of natural samples.

[Back to the program](#)

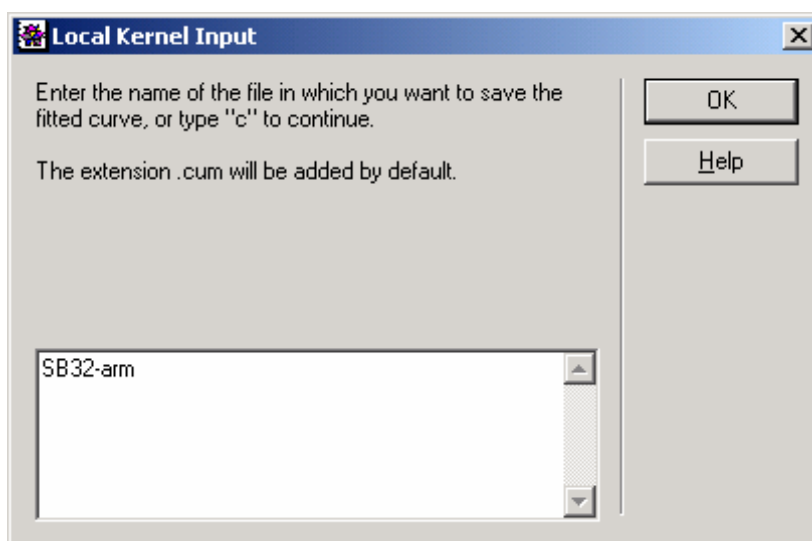




### Plot the filtered measurements and save them in a file

At this point, a filtered measurement curve is calculated from the [filtered residuals](#) by inverting all rescaling steps described above. The measurement errors are calculated from the difference between filtered and unfiltered residuals, and are displayed on a [separate plot](#). You can save the result in an ASCII file with three columns. The first column contains the applied field, the second column contains the magnetization and the third column contains the estimated absolute measurement error.

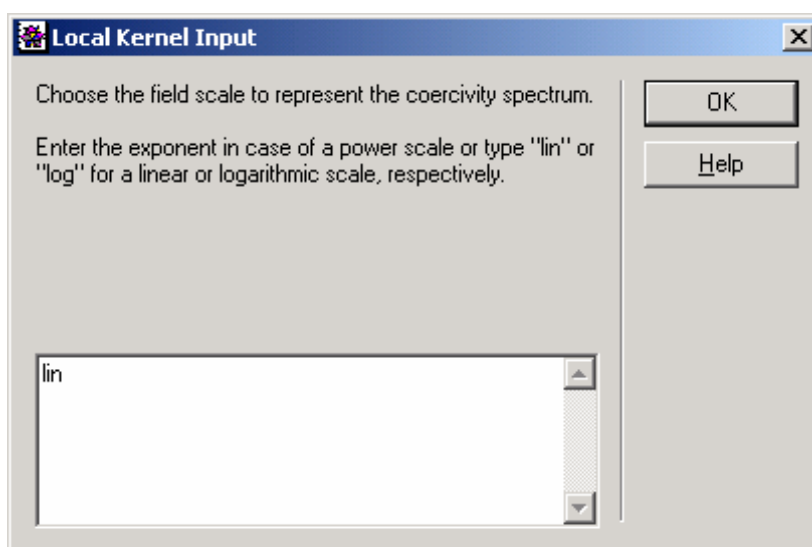
[Back to the program](#)



### Plot the coercivity distribution on different field scales

CODICA calculates the coercivity distribution by numerical differentiation of the [filtered measurements](#). You can choose between a [linear field scale](#), a [logarithmic field scale](#) and a [power field scale](#). The coercivity distribution is plotted on the requested scale (solid line), together with the confidence limits given by the estimated measurement errors (dashed lines). The maximal relative error affecting the calculated coercivity distribution is displayed on a [separated plot](#). You can save the result on a ASCII file with three columns. The first column contains the applied field, the second column contains the values of the coercivity distribution and the third column contains the estimated relative error of the coercivity distribution.

[Back to the program](#)



### Cautionary note

CODICA 3.0 has been tested more than 1000 times with measurements of various artificial and natural samples and the most different combinations of initial parameters. Nevertheless, there is a remote possibility that particular uncommon or noisy data or parameter sets will produce evaluation problems. In this case, blue-written warning messages appear on the *Mathematica* front-end. If more than one of these messages is displayed, you may force-quit the Kernel of *Mathematica* as follows: in the top menu bar choose Kernel → Quit Kernel → Local. You can also exit from CODICA at any time just by typing “abort” in any input prompt window.

CODICA 3.0 does not work on previous versions of *Mathematica 5.0* because of a substantial redefinition of a minimum-search routine embedded in *Mathematica 5.0*. Updates of *Mathematica* will generally not affect the functionality of packages such as CODICA, which is expected to run on later versions. Use CODICA 2.3 on previous versions of *Mathematica 5.0*.

In case of problems, write to the author (Ramon Egli) at the address given in the `install.txt` file of the installation packet or at the beginning of the source code file `Codica.m`. Please save and send a copy of the *Mathematica* session you were using when the problem arised, together with the data file you analyzed with CODICA.

### Reference

Egli, R., Analysis of the field dependence of remanent magnetization curves, *J. Geophys. Res.*, 102, doi 10.1029/2002JB002023, 2003.