User's guide to the

© 2005 by Ramon Egli

magnetic unmixing software packet

Release 1 - April 2005

Includes the programs CODICA and GECA for the analysis of magnetization curves and coercivity distributions. Runs on MS Windows with *Mathematica 5.0* and later versions.

Preface

The software package MAG-MIX provides computer programs for the analysis of magnetization curves and coercivity distributions. This first release includes the programs CODICA and GECA. CODICA is a program that calculates the coercivity distribution of a magnetization curve and estimates the measurement errors. GECA is a program for modeling a coercivity distribution as a linear combination of special model functions that are supposed to represent the coercivity distribution of specific groups of magnetic particles, called *components*.

The analysis of magnetization curves is a relatively recent and fast-developing method used in environmental and rock magnetic studies to characterize materials that are a mixture of different magnetic minerals. The success of "unmixing" such materials depends strongly on our knowledge about the fundamentals of magnetization processes and coercivity distributions. This knowledge is evolving rapidly, and might provide more efficient and easy-to-use unmixing methods in the future.

The MAG-MIX manuals contain detailed instructions for using the programs and provide basic knowledge about the theory of magnetization curves, coercivity distributions and component analysis.

MAG-MIX will be updated periodically. A profound revision of GECA that takes into account recent progess in understanding coercivity distributions is already planned. Other programs for the automated analysis of a large number of similar samples – such as those collected from a sediment profile – will be added in the future.

Your feedback is important to improve the MAG-MIX programs. If you encounter problems in using the programs or if you have suggestions, please write me (eglix007@umn.edu). Many suggestions of "early" users have been already taken into consideration in the present version of CODICA.

Ramon Egli
Minneapolis, april 4, 2005

CODICA

COercivity **DI**stribution **CA**lculator

Version 5.0

User's manual

Preface to CODICA 5.0

The previous versions of CODICA have been used to analyze magnetization curves of the most different materials measured with various instruments. This program was first conceived in 2000 as a particular "filter" for the analysis of natural sediment samples. By that time, the knowledge related to the measurement of magnetization curves for the purpose of component analysis was based on the seminal papers of *Robertson and France* [1994] and *Kruiver et al.* [2001]. Since then, a few hundred sediment samples have been measured in detail and analyzed with CODICA. The experience collected led to interesting conclusions about the occurrence of magnetic components in sediments [*Egli*, 2004a,b,c] and gave raise to the need of a user-friendlier and stable version of the program which is accessible also to non-specialized users. Several bugs have been already corrected in version 4. The use of CODICA for the analysis of different types of magnetic materials generated occasionally critical errors. The main problems reported by users were:

- The incapability of reading a data file stored directly in a disk partition and not in a folder (e.g. `C:/myfile.dat` instead of `C:/myfolder/myfile.dat`).
- The instability of the fitting routines included in CODICA, which led to fatal error in some cases, for example with particular materials such as magnetic tapes.
- The need to enter manually special fitting parameters whose meaning is not intuitive, and the need of special characters such as "{", which are not available on all keyboards.

A profound revision of CODICA has been undertaken to make the program as most user-friendly, automated and stable as possible. The result is a fully new version where the user is asked only to control the degree of "smoothing" of the final result. Experienced users will appreciate the fully automated optimization of all the mathematical steps required to fit a magnetization curve and calculate the corresponding coercivity distribution. New users will not need to train intensively with CODICA before getting useful results. CODICA 5.0 can be installed exactly like the previous version. Owners of a previous version may just replace the old source code file `Codica.m` with the new one.

In the following, a detailed list of the improvements and changes made in CODICA 5.0 is given.

- Acquisition curves are analyzed as such and not transformed into demagnetization curves.
- The error estimation does no longer require the evaluation of all measurement errors sources by the user. A long experience on measuring magnetization curves demonstrated that the sources of measurement errors are extremely complex and difficult to predict. The errors are now estimated empirically during the analysis of the magnetization curve.
- Various errors sources, such as the lack of measuring points at critical regions of the magnetization curve and digital truncation effects, are recognized by CODICA. A warning message is produced and the negative effects of the identified error sources are minimized as far as possible.
- All rescaling procedures applied by CODICA are now fully automated. The user has the possibility of modifying the magnetic field scaling; however, a manual correction is almost never

required. CODICA also analyzes the distribution of the measurement points and suggests improvements for better results.

- The Butterworth low-pass filter has been replaced with a least-squares collocation method based on *Moritz* [1978]. Least-squares collocation is a very general and efficient method used for the analysis of signals and potentials in geodesy.
- The user can choose the range of fields in which the coercivity distribution is calculated. This is a useful option if the results are used for component analysis.

References:

Egli, R. (2004a). Characterization of individual rock magnetic components by analysis of remanence curves, 1. Unmixing natural sediments, *Studia Geophysica et Geodaetica*, **48**, 391-446. (back to text)

Egli, R. (2004b). Characterization of individual rock magnetic components by analysis of remanence curves, 2. fundamental properties of coercivity distributions, *Physics and Chemistry of the Earth*, **29**, 851-867. (back to text)

Egli, R. (2004c). Characterization of individual rock magnetic components by analysis of remanence curves, 3. bacterial magnetite and natural processes in lakes, *Physics and Chemistry of the Earth*, **29**, 869-884. (back to text)

Kruiver P. P., M. J. Dekkers and D. Heslop (2001). Quantification of magnetic coercivity components by the analysis of acquisition curves of isothermal remanent magnetization, *Earth and Planetary Science Letters*, **189**, 269-276. (back to text)

Moritz, H., Least-Squares Collocation (1978). *Reviews of Geophysics and Space Physics*, **16**, 421-430. (back to text)

Robertson D. J. and D. E. France (1994). Discrimination of remanence-carrying minerals in mixtures, using isothermal remanent magnetisation acquisition curves, *Physic of the Earth and Planetary Interiors*, **82**, 223-234, 1994. (back to text)

Index

1. Introduction 5
2. Before using CODICA: some suggestions to optimize your measurements 11
3. Basic theory of coercivity distributions: all what you should know 17
4. Install and run CODICA 5.0 25
5. CODICA 5.0 tutorial 29
6. Technical reference 51
6.1. Weighted fit 51
6.2. Scaling the magnetic field 51
6.3. Model function for the magnetization curve 51
6.4. Search the best field scale for symmetry 53
6.5. Model the magnetization curve 53
6.6. Calculate the residuals 54
6.7. A stochastic model for the residuals 54
6.8. Subtract a trend with polynomials 55
6.9. Rescale the residuals 56
6.10. A least-squares collocation model for the residuals 57
6.11. Filtering the residuals 58
6.12. Filtered magnetization curve 59
6.13. Coercivity distribtuions 59
Appendix A 61
Appendix B 62

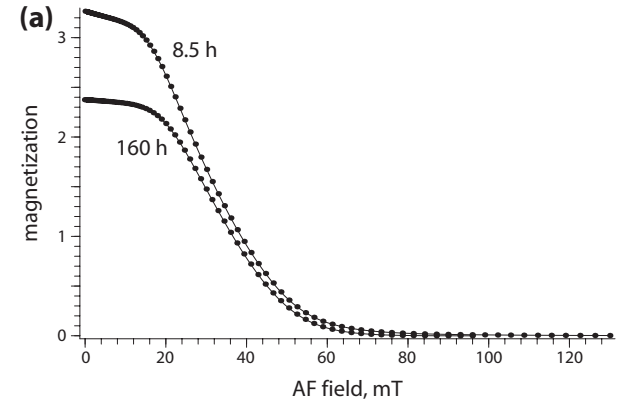
1. Introduction

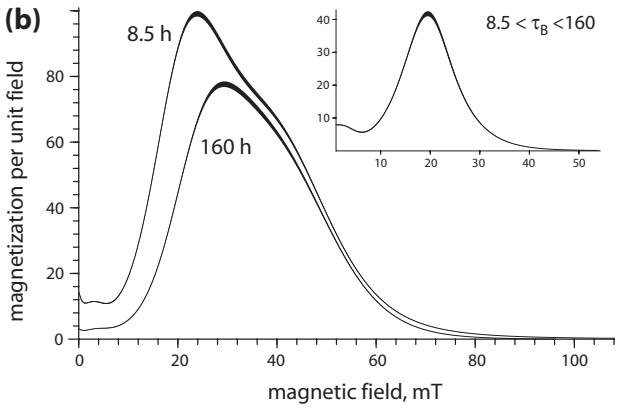
Why magnetization curves?

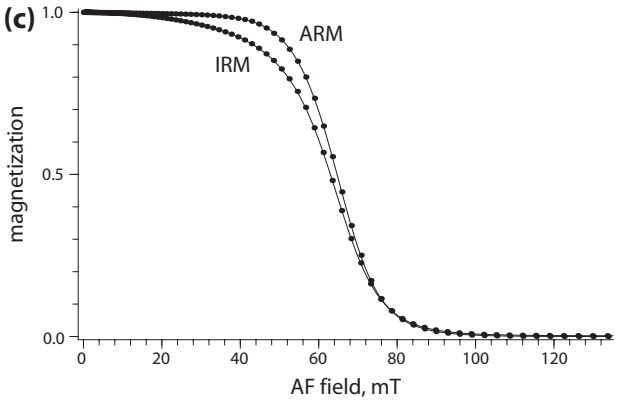
Rocks and sediments inevitably contain mixtures of magnetic minerals, grain sizes, and weathering states. Most rock magnetic interpretation techniques rely on a set of value parameters, such as susceptibility and isothermal/anhysteretic remanent magnetization (ARM or IRM). These parameters are usually interpreted in terms of mineralogy and domain state of the magnetic particles. In some cases, such interpretation of natural samples can be misleading or inconclusive. A less constrained approach to magnetic mineralogy models is based on the analysis of magnetization curves, which are decomposed into a set of elementary contributions. Each contribution is called a magnetic component, and characterizes a specific set of magnetic grains with a unimodal distribution of physical and chemical properties. Magnetic components are related to specific biogeochemical signatures rather than representing traditional categories, such as SD magnetite. This unconventional approach can be regarded as a kind of principal component analysis (PCA) that gives a direct link to the interpretation of natural processes on a multidisciplinary level. Since magnetic components rarely occur alone in natural samples, unmixing techniques and rock magnetic models are interdependent.

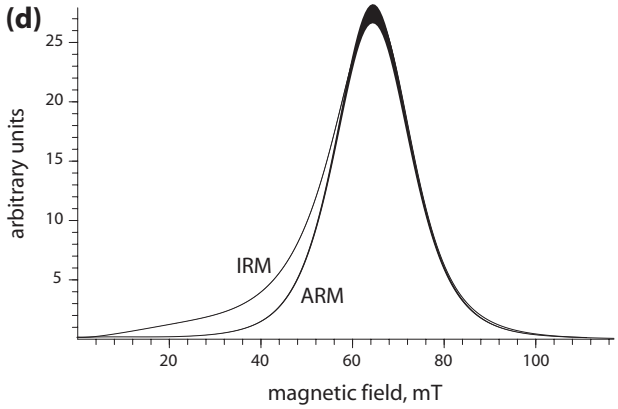
Unmixing problems dealing with unknown components are strongly nonlinear and have usually multiple solutions. Therefore an initial guess of the model parameters is required. This guess relies on additional information about the geological and geochemical history of the sample. Valuable information for rock magnetic and environmental studies can be obtained directly from the coercivity distribution of the sample, which provides a richness of details hidden in the measurement curve (Fig. 1).

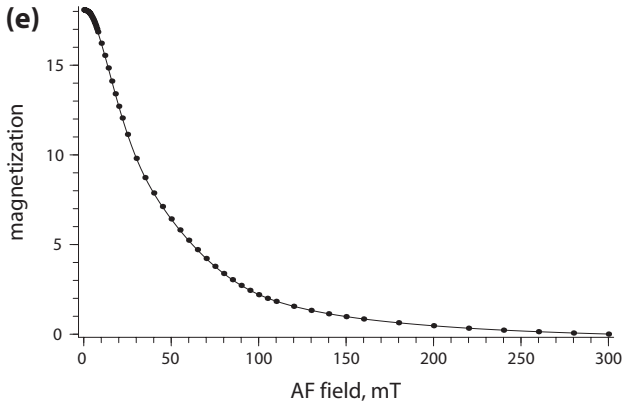
Fig. 1 (next page). Some applications of CODICA to rock magnetic and environmental studies. The left plots show the original measurements, the right plots are coercivity distributions calculated with CODICA. The thickness of the curves corresponds to the error estimate of CODICA. **(a)** AF demagnetization curves of a Tiva Canyon Tuff that contains acicular magnetite in the SP/SSD grain size range. Measurements have been started 8.5 and 160 hours after an ARM was imparted (see *the IRM Quaterly*, 14(3), 2004, for more details about the Tiva Canyon Tuff). **(b)** Coercivity distributions calculated from (a). Notice the bimodal character of the 8.5 h curve, showing the magnetization of viscous and more stable particles. The difference of the to curves (inset) shows the coercivity distribution of the viscous particles with relaxation times between 8.5 and 160 h. **(c)** Modified Lowrie-Fuller test performed on a sample of intact MV1 magnetotactic bacteria, and **(d)** the corresponding coercivity distributions of ARM and IRM. Notice that the two magnetizations are identical except for a low-coercivity IRM contribution which may be related to collapsed or not well formed chains of magnetosomes (sample kindly provided by D. Bazylinski). The properties of this low-coercivity contribution are very similar to those of clustered SD particles. **(e)** AF demagnetization of an ARM imparted to an anoxic sediment from lake Baldeggersee, Switzerland (see R. Egli, *Physics and Chemistry of the Earth*, 29, 869-884, 2004, for a detailed discussion about magentic measurements on this lake). **(f)** Coecivity distribution calculated from (e), and results of a component analysis (colored areas). Three components can be clearly distinguished: the lowest coercivity component has been attributed to detrital magnetite, the middle coercivity component to magnetosomes that survived reductive dissolution, and the high coercivity component to a high coercivity mineral such as hematite. **(g)** AF demagnetization curves of ARM from samples of particulate matter collected from the atmosphere at three places in the city of Zürich, Switzerland (see S. Spassov et al., *Geophysical Journal International*, 159, 555-564, 2004). **(h)** The coercivity distributions show the increasing contribution of a high coercivity component in more polluted areas (GMA: forest near Zürich, WDK: center of Zürich, GUB: highway tunnel).

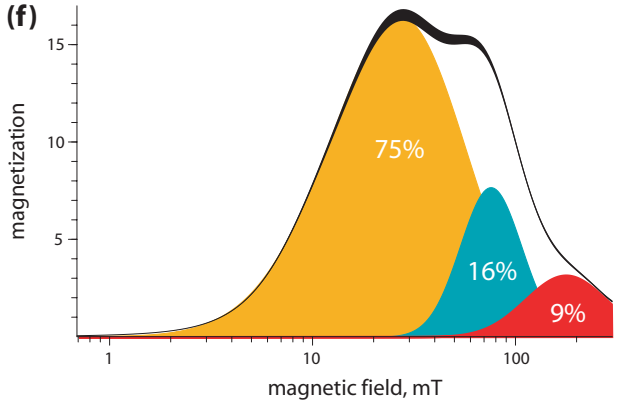

(a)
8.5 h
160 h
magnetization
AF field, mT

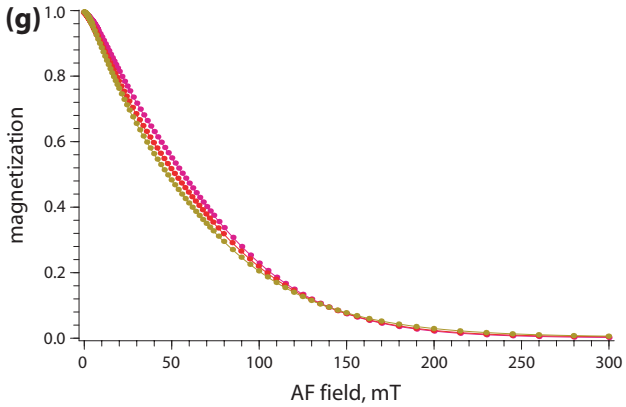

(b)
8.5 h
160 h
8.5 < τB <160
magnetization per unit field
magnetic field, mT

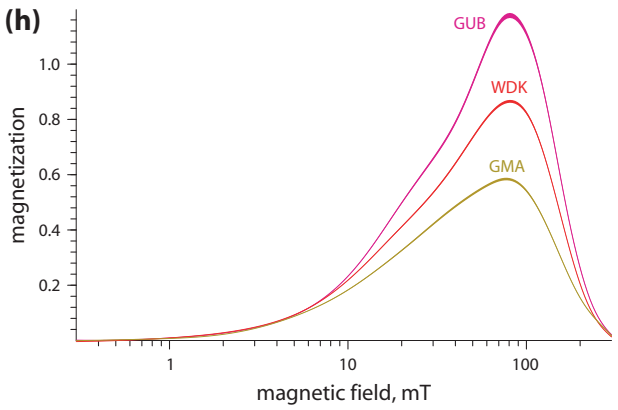

(c)
ARM
IRM
magnetization
AF field, mT


(d)
IRM
ARM
arbitrary units
magnetic field, mT


(e)
magnetization
AF field, mT


(f)
75%
16%
9%
magnetization
magnetic field, mT


(g)
magnetization
AF field, mT


(h)
GUB
WDK
GMA
magnetization
magnetic field, mT

What is CODICA?

CODICA (COercivity DIstribution CAlculator) is a program for the detailed analysis of magnetization curves and the calculation of coercivity distributions. CODICA takes advantage of the universal properties of magnetization curves and uses advanced mathematical tools to model magnetization curves and measurements errors without using restrictive assumptions. The calculated coercivity distributions are free of distortions and typical artifacts of common filtering methods. These are *intrinsically* inadequate for the analysis of asymptotic functions such as magnetization curves. The confidence limits provided with the results are particularly useful for evaluating the significance of multicomponent mixing models based on coercivity distributions. Fig. 2 shows a comparison between the performance of CODICA and that of a commercial software built-in filtering method.

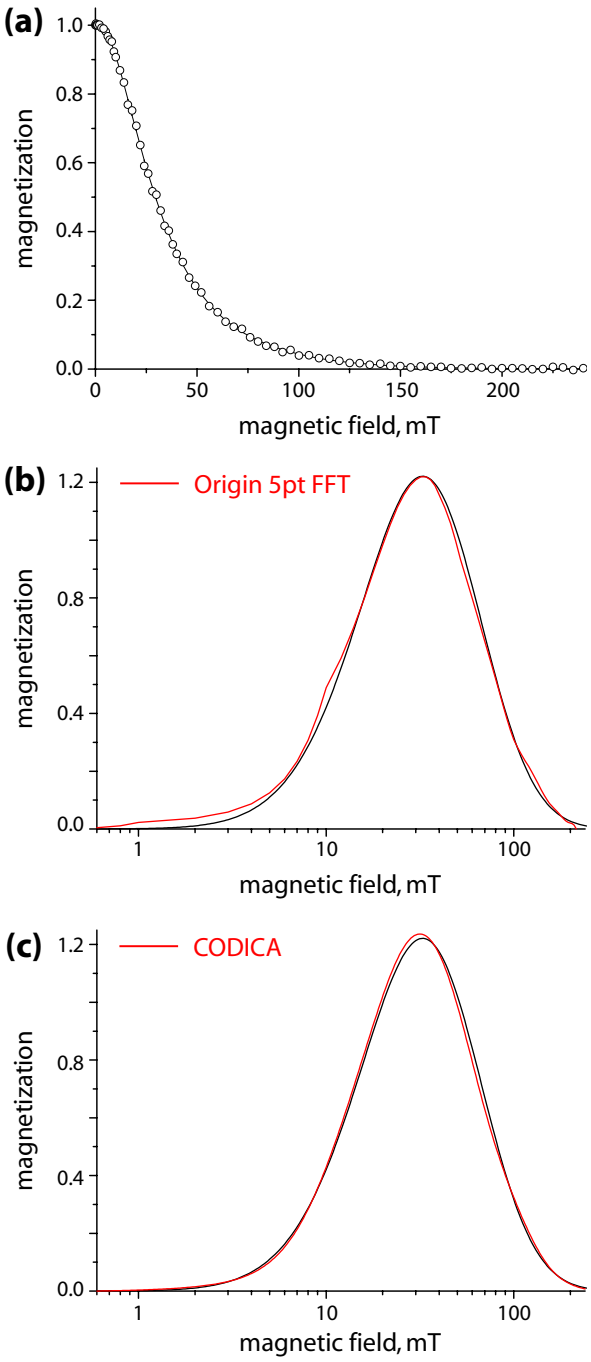


Fig. 2: Performance comparison between CODICA and a commercial software built-in filtering method. **(a)** An artificial magnetization curve was created by simulating the magnetization curve (solid line) of a logarithmic Gaussian coercivity distribution. Measurements (dots) has been calculated using a random number generator, and assuming Gaussian errors with a standard deviation of 0.002 for the magnetization measurements, and a standard deviation of 2% for the magnetic field. The red curves in **(b)** and **(c)** show the coercivity distribution calculated from the simulated measurements in (a), using the software *Origin* and CODICA, respectively. The black curve in both plot is the error-free logarithmic Gaussian distribution used to simulate the measurements. The coercivity distribution calculated using *Origin* was obtained by first-order numerical differentiation and subsequent 5 points FFT low-pass filtering. FFT low-pass filters based on a different number of points gave worse results. Notice the distortions at low fields in (b).

How does CODICA work?

The method used by CODICA is based on three fundamental properties that characterize *any* magnetization curve:

1) The curve is *monotonic* (it always either increases or decreases over the entire range of fields),
2) The curve has a horizontal asymptote that is approached – but not reached – at high fields (i.e. the sample *saturates* at high fields),
3) The slope of the curve at zero field is always *finite* (i.e. the sample has a stable magnetization in a zero field)

CODICA proceeds essentially on three steps that are shown in Fig. 3 with a simple example. First, a set of scale transformations is applied to the field and the magnetization (Fig. 3b). After field and magnetization scales have been changed, the magnetization curve becomes close to *straight line*, and is said to be *linearized* (Fig. 3c). A so-called *residual curve* is obtained after removing the linear trend of the scaled curve by subtraction of a polynomial (Fig. 3d). The residual curve has the characteristics of a stochastic signal, because it oscillates more or less randomly around a mean value of zero. The wiggles arise from small asymmetries of the original magnetization curve, as well as from the measurement errors. Measurement errors are easy to recognize in the residual curve, since they are highly amplified. This gives you the possibility to optimize and correct your experiments for optimal results. The residual curve is then fitted with a method called *least-squares collocation*, which is a particularly effective model for stochastic (non-periodic) signals (Fig. 3d). The interpolated residual curve – supposed to be free of measurement errors – is transformed back into a magnetization curve (Fig. 3e) and its first derivative, called *coercivity distribution* (Fig. 3f). The least-squares collocation method provides also a way to estimate the error associated to the operations described above and thus provides confidence limits for the results it produces.

Is CODICA difficult to use?

In the previous versions of CODICA, the steps described above had to be controlled by the user, which was asked to enter various rather cryptic parameters. As a result, the user was required to have some mathematical skills and a lot of endurance. This new version of CODICA is fully automated. You have to decide only the degree of smoothing used to interpolate the measurements, and this operation is rather intuitive. The complete analysis of a magnetization curve takes no more than a couple of minutes – in addition to the computation time in case of curves with a large number of measurements. To take full advantage of CODICA, read carefully this manual, which contains a practical guide through each step of the program with a real example. The most interested users can read the technical reference, which contains detailed information about the operations performed by CODICA and their theoretical background. However, this information is not necessary for a standard use of CODICA.

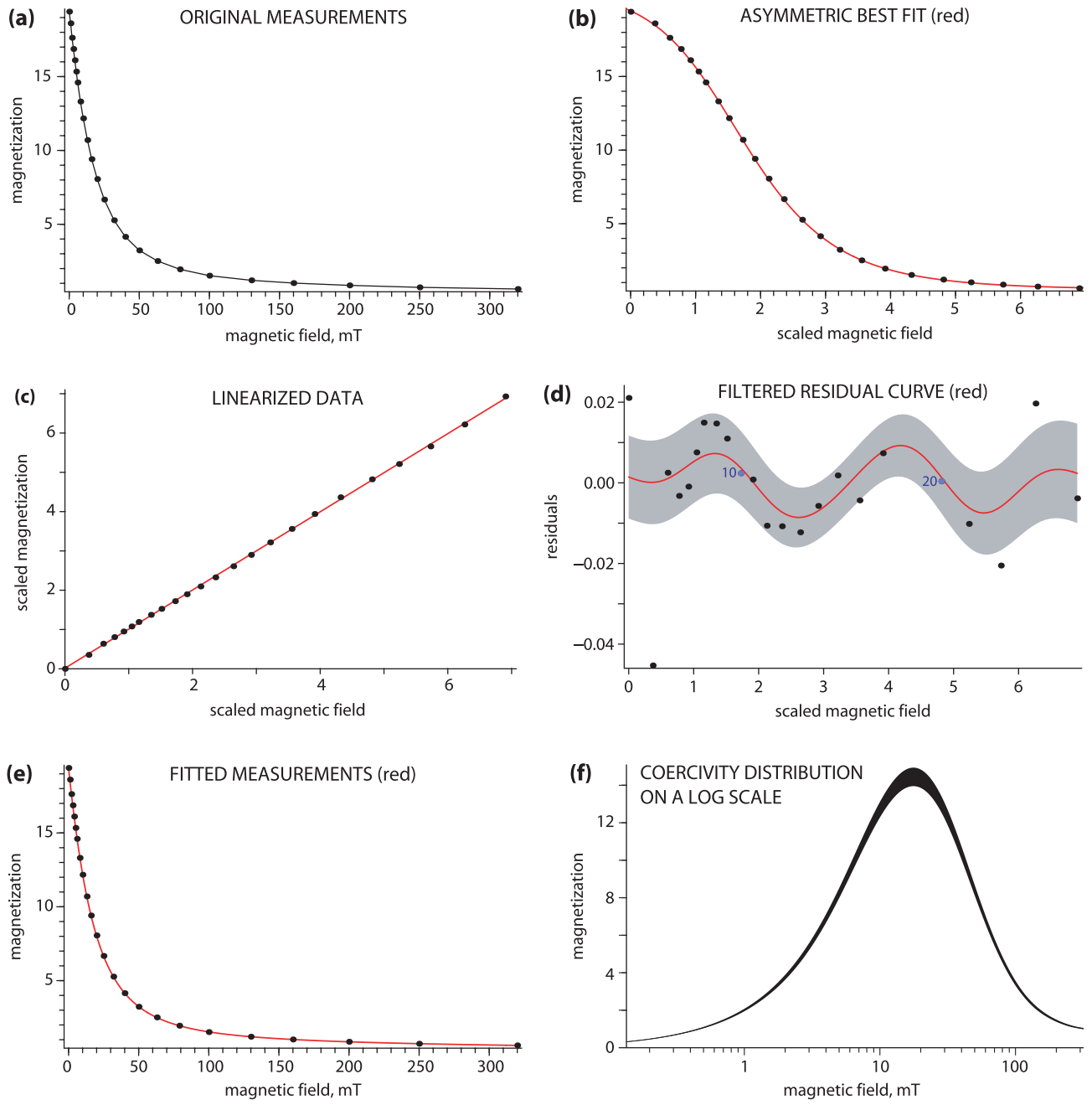


Fig. 3. The working principle of CODICA on a simple example (sample kindly provided by Christoph Geiss). **(a)** Original AF demagnetization curve showing a characteristic MD shape. **(b)** The field axis is rescaled in order to get a sigmoidal-shaped curve. To do so, the scale is expanded at small fields. **(c)** The magnetization is now rescaled in order to linearize the magnetization curve. To obtain this result, CODICA expands the magnetization scale near the beginning and the end of the magnetization curve. The red line is the linear best-fit to the data. **(d)** The linear trend of the rescaled curve (red line in c) is subtracted to obtain the so-called residuals. As it can be clearly seen, the measurement errors are quite evident in this plot. The red curve is a best-fit of the residuals that CODICA obtains from an autocorrelation model. **(e)** A model for the "error-free" magnetization curve (red) is obtained from the fitted residuals (red curve in d), by inverting the mathematical functions used to transform the original measurement (a) into the residuals (d). **(f)** A coercivity distribution is calculated as the analytical derivative of (e). The thickness of the line corresponds to the estimated confidence limits of the coercivity distribution. CODICA calculates the coercivity distribution on a logarithmic field scale, as in (f), as well as on a linear scale.

Why is a compiled version of CODICA not available?

To run CODICA you need *Mathematica 5.0* or a later version to be installed on a Windows system. Of course, a compiled version of CODICA would be more attractive. However, CODICA uses highly sophisticated mathematical routines that are embedded in *Mathematica* and are not easy to include in a compiled program.

Does CODICA make miracles?

Some users may be surprised by the poor performance of CODICA on some typical "paleomagnetic quality" magnetization curves, which can be obtained with standard experiments. However, you should always consider that CODICA – as any other software – does not *add* a bit of information to your original measurements: bad measurements will give poorly defined coercivity distributions, affected by large errors. However, CODICA finds the best fit to your data and helps to identify the reason of poor results. You can repeat the measurements with a better sample and with optimized magnetization/demagnetization steps. This manual contains a section that helps you in optimizing your measurements.

Does CODICA contain bugs?

Maybe! The program code is almost 4000 lines long and its performance is not guaranteed under all circumstances. However, all possible user-controlled options of CODICA have been tested on many measurements. If a result is obtained, it is guaranteed to be correct (i.e. free of coarse errors such as units or scale errors).

2. Before using CODICA: some suggestions to optimize your measurements

Why remanence measurements?

The measurement of remanence curves is intrinsically more difficult than in-field measurements, such as hysteresis loops. There are essentially two reasons that explain this fact. The first reason resides essentially in the technical difficulty of switching on and off a magnetic field produced by an electromagnet in an exactly reproducible manner. In fact, many electromagnet control systems produce a slightly positive overshoot before reaching the programmed field intensity or a slightly negative overshoot when the field is turned off. On the other hand, pulse magnetizer become slightly resonant at high fields, so that small negative fields may be produced after the main discharge peak. The second reason is that some samples are themselves a source of error, especially when the remanent magnetization to be measured is much smaller than the saturation magnetization of the sample. In such cases, the smallest amount of magnetization induced by random variations of the ambient field is sufficient to produce a noticeable effect in the measurements. So, why use remanent magnetization to characterize a sample or to perform a component analysis? Component analysis can be performed on any kind of magnetization curves, including hysteresis loops [*von Dobeneck*, 1996; *Carter-Stiglitz et al.*, 2001]. The problem with hysteresis loops is that they cannot be easily unmixed, unless the properties of the individual components are known *a-priori*. Paradoxically, the most suitable curves for component analysis are those obtained from weak-field magnetizations, such as ARM and TRM, even if these magnetizations are among the most difficult to measure. The advantage of using weak-field magnetizations relies on their sensitivity to parameters such as grain size and oxidation state [*Egli*, 2004].

Measuring remanent magnetization curves

Errors in remanent magnetization measurements are mostly of technical nature and can be reduced by improving the measuring technique. The standard way of measuring remanent magnetization curves has been established in the early story of paleo- and rock magnetism, and underwent little changes despite the evolution of experimental equipments. A classic (de)magnetization curve consists of 10-15 measurement points more or less equally distributed over the coercivity range of the sample. Imagine to perform a component analysis with three components on such measurements. Each component is described by a 5 parameter model function, and 15 parameters are needed for the three components. The number of measurements corresponds barely to the number of parameters, and the model is exactly determined. Small measurement errors are sufficient to drive such a model into a completely wrong solution! In fact, a model with three components and a redundance factor of 5 (five measurements for each model parameter) requires 75 measurement points. If you find this number excessive, think about a Day plot: the five parameters required (two for the magnetization ratio, two for the coercivity ratio and one for the paramagnetic correction) are obtained from hysteresis loops and backfield curves consisting of a *large* number of points. The reason for the difference between the 'common sense' accuracy of a hysteresis loop and that of a remanent magnetization curve resides in the extra time required in the latter case to switch on and off the (de)magnetizing field. Magnetization curves consisting of 80 measurements require few minutes to 5 hours running time, depending on the (de)magnetization device and the degree of automation. An interesting and extremely fast system for the acquisition of remanent magnetization curves has been developed by the paleomagnetic labora-

tory of the Kazan State University, using a self-made equipment informally named 'Pashameter' after its constructor [*Jasonov et al., 1998*]. With this equipment, complete acquisition curves with thousands of measurements up to 500 mT can be obtained in a few minutes (Fig. 4).

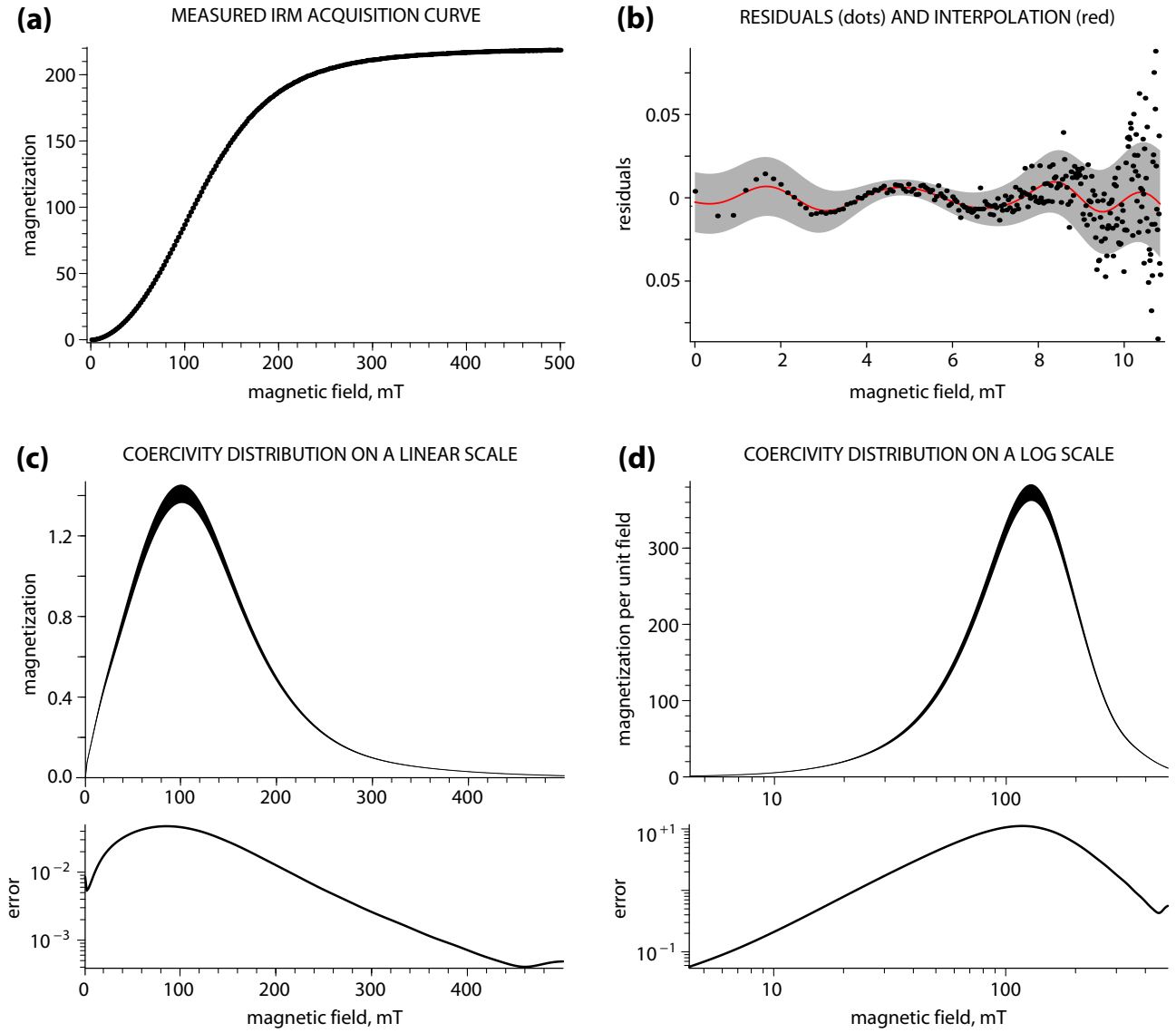


Fig. 4. Example of a IRM acquisition curve measured with the coercivity spectrometer described in *Jasonov et al.*, 1998 (courtesy of R. Enkin), and subsequent coercivity analysis with CODICA. **(a)** Raw measurements. A typical measurement up to 500 mT consists of 1500 steps acquired in few minutes. **(b)** The residual curve calculated by CODICA shows clearly the measurement errors. The increasing amplitude of these errors at high fields indicates that the applied magnetic field account for a large portion of the errors. The red line is the fit calculated by CODICA. **(c)** and **(d)** are the corresponding coercivity distributions calculated on a linear, respectively logarithmic scale. The thickness of the curve is proportional to the estimated error, which is also plotted below. Notice that the maximum error of the coercivity distribution is less than 4%. The smoothness of the coercivity distribution suggests a single magnetic component for this sample. On the logarithmic scale, however, a small contribution from a high-coercivity component is suggested by the inflection of the curve above 300 mT.

More effective than increasing the number of measurement points is the minimization of the experimental errors. These errors arise from (1) magnetometer errors, (2) rounding to an insufficient number of digits by the acquisition software, (3) errors in the application of the (de)magnetizing field, (4) imprecise sample positioning both in the field and in the magnetometer, (5) time effects on viscous samples, (6) insufficient shielding of external magnetic perturbations, and (7) mechanical unblocking of magnetic grains in strong magnetic fields. In the following, we shall briefly discuss some of these experimental errors and how to reduce or avoid them.

1. Magnetometer errors. They are intrinsic to the magnetometer used.

2. Digital rounding. Most acquisition softwares give three-digits results which are sufficient for all traditional purposes but are likely to produce nasty rounding effects in the saturation region of detailed magnetization curves (Fig. 5). SQUID magnetometers have a much higher intrinsic accuracy, which can be exploited by using a 5 digits reading. Some acquisition softwares for the 2G cryogenic magnetometer may combine erroneously the flux count and the analog signal. This error does not produce noticeable effects on standard measurements, unless the magnetization of the sample corresponds to about one flux jump of the SQUID sensor. Such software errors have to be removed if a component analysis is performed, regardless of the sample's intensity.

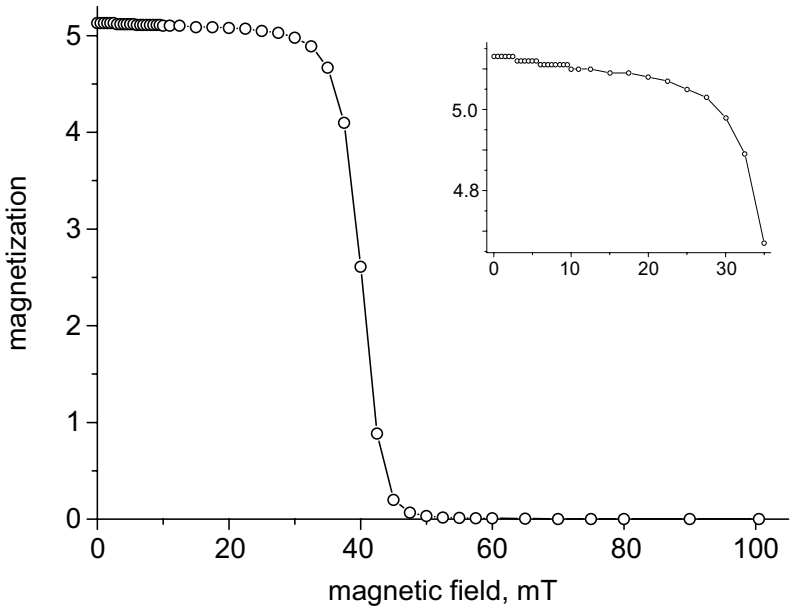


Fig. 5. Example of digital rounding effects on a magnetization curves (measurement kindly provided by C. Geiss). Digital rounding is particularly evident in the flat region of the curve (small inset). Rounding to 3 digits can introduce noticeable errors in the calculation of coercivity distributions. CODICA can handle digital truncation to minimize errors, however, obtaining full-digits data is a preferable option.

3. Errors in the applied field. The precision of the field applied on the sample depends on the control unit of the magnet. The field control of large magnets is critical. Pulse magnetizers may become slightly underdamped at high fields and produce a small negative field after main discharge. This results sometimes in IRM acquisition curves that *decrease* ant high fields. The generation of alternating fields for AF demagnetization is very sensitive to electromagnetic interferences. Short-pulsed interferences are cancelled through thermal activations if the decay rate of the alternating field is small enough. As a role-of-thumb, avoid decay rates higher than 4 mT/s. To calculate the decay rate of your equipment, take the peak AF field and divide it by the time required by the AF field to decay from its maximum value to zero (do *not* include the ramping-up time!). Some AF demagnetization devices are more precise than other. We noticed some systematic small errors produced by the AF demagnetization system of 2G (Fig. 6). These errors are not noticeable in standard applications, but are significant when detailed demagneti-

zation curves must be obtained. We do not know the origin of these errors, but since they are systematic, a correction formula is provided (see Appendix B). Pass-trough devices are even more critical, since they are faster and sensitive to the speed of the sample through the magnet.

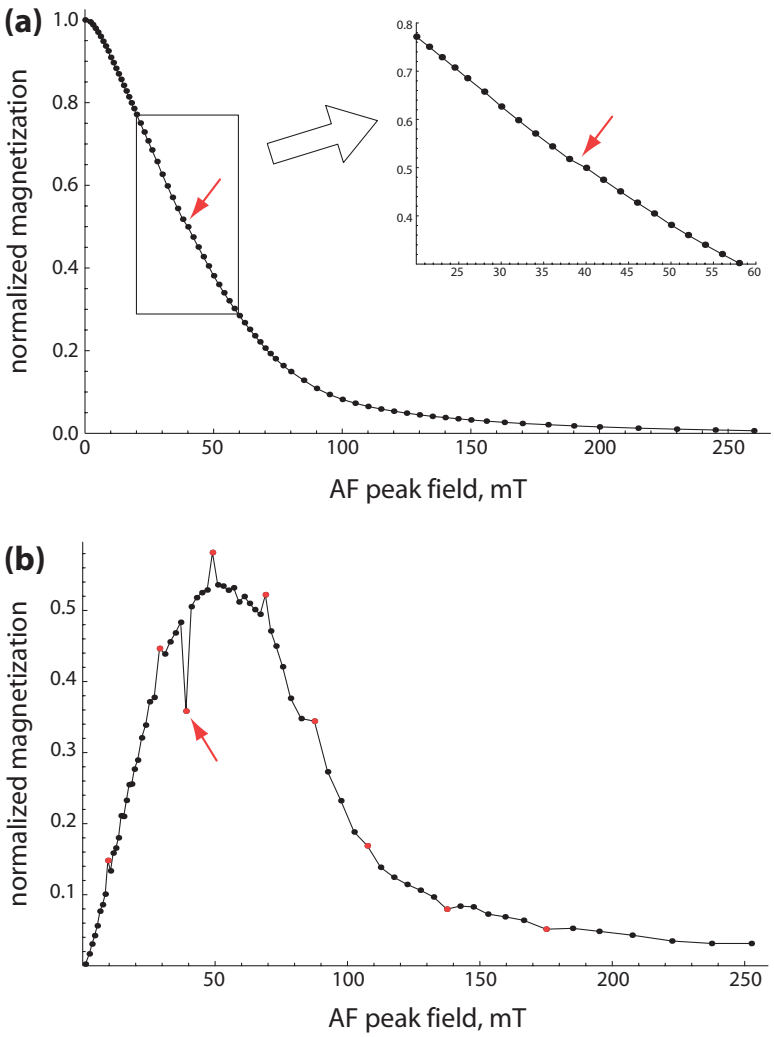


Fig. 6. (a) Average of 606 AF demagnetization curves that have been normalized by their initial magnetization. Both AF demagnetization and measurements have been performed using a 2G cryogenic magnetometer with a 300 mT AF demagnetizing system (AF settings: ramp rate =5, dwell time = 3). These curves include ARMs and IRMs of different samples with magnetic moments spanning over more than 6 orders of magnitude. In the averaged curve, random errors are reduced by a factor 25 with respect to a single measurement, while systematic errors are unaffacted. The averaged curve shows discontinuities occurring at specific AF peak fields (small inset shows the most evident discontinuity at 40 mT). This problem indicates some systematic errors in the applied field. **(b)** Coercivity distribution calculated by finite differences of the curve in (a). Discontinuities in (a) shows up as distinct peaks in the coercivity distribution. See Appendix B for an empirical correction of these errors.

4. Positioning errors. The magnetic field produced by electromagnets is homogeneous only in a small region. If the dimensions of that region are comparable to those of the sample, a correct positioning becomes critical. The same apply to the response function of the magnetometer. Keep in mind that an error of 0.1% in both the applied field and the measurement is an upper limit for component analysis applications. If the sample is placed by hand at each (de)magnetization step, errors arising from its misorientation are generally not negligible, unless the sample fits firmly in a fixed holder, both during field application *and* measurement.

5. Time effects. Time effects are a *very* important and often disregarded error source during the measurement of magnetization curves. All samples –regardless of their composition– have a time-dependent magnetization (known as *magnetic viscosity*) and a time-dependent coercivity (which is related to Néel's concept of *fluctuation field*). Furthermore, the effectiveness of the AF demagnetization depends on the decay rate of the alternating field. Thus, time effects influence the shape of a coercivity distribution (Fig. 7). Hence, a precise and constant timing of both the applied field and the interval between field application and measurement is necessary to obtain

good magnetization curves. Precise timing is generally fulfilled by automated measurements. If you perform manual measurements, you should always apply the field for, let say, 2 seconds and wait a precise amount of time before measuring. Demagnetization curves need an additional consideration. Immediately after the magnetization is acquired, there is often a significant viscous decay. We recommend waiting long enough before starting with demagnetization measurements, since the viscous decay of the sample increases the initial gradient of the demagnetization curve, which is erroneously interpreted as a low-coercivity component.

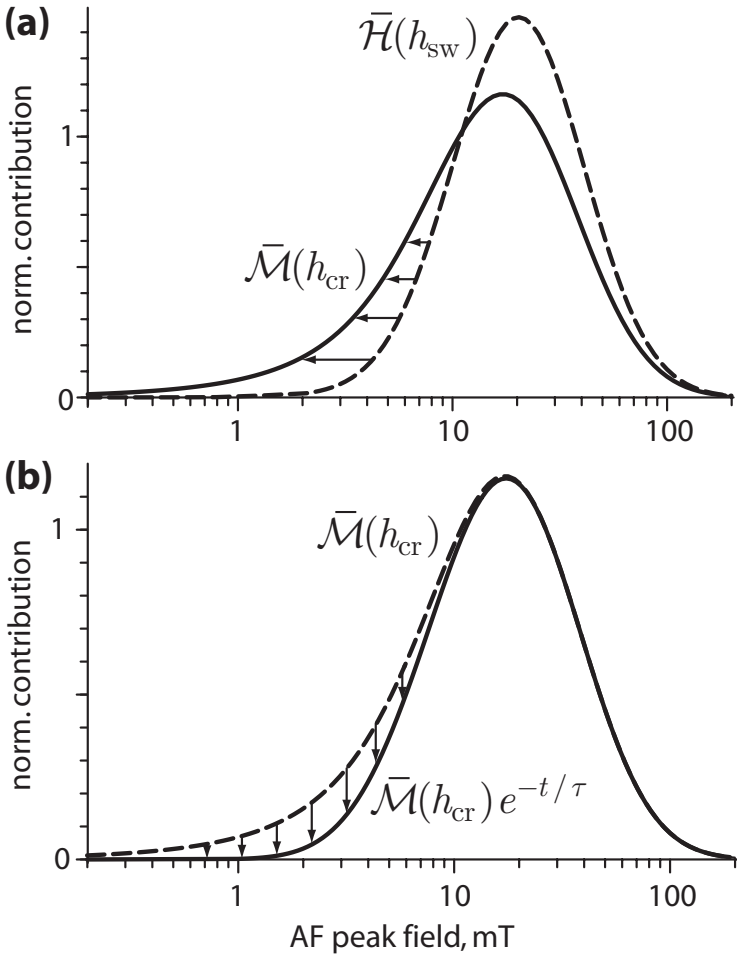


Fig. 7. Calculated effects of thermal activation effects on the coercivity distribution of a set of non-interacting SD particles with a logarithmic Gaussian distribution of microcoercivities and grain sizes (redrawn from *R. Egli, Physics and Chemistry of the Earth, 29, 851-867, 2004*). The dashed (solid) curves indicate the calculated coercivity distribution without (with) thermal activations. **(a)** Effect of thermal activations on the switching field of the particles, which is reduced by an amount called the fluctuation field (arrows). This effect also depends on how long the (de)magnetizing field is applied. **(b)** Effect of thermal activations on the remanent magnetization of the grains. The magnetization of grains with a smaller microcoercivity decay faster with time (arrows). This effect depends also on the time lag between the acquisition of a magnetization and its measurement.

6. Shielding. Ideally, the sample *and* the magnetometer should be shielded against external variations of the magnetic field and other electromagnetic interferences.

7. Mechanical unlocking. Unconsolidated powder samples may contain magnetic grains that can rotate under the influence of a strong magnetic field. Alternating fields are especially effective in 'shaking' lose magnetic grains. Powder samples should be firmly pressed in sample boxes. Empty space in the box should be filled with a nonmagnetic material, such as folded, thin plastic foil or calcium fluoride. Badly sorted, clay-poor sediments, such as loesses and glacial deposits, are good candidates for mechanical unblocking effects. In this case, mix them with at least 40% of a nonmagnetic 'binding' material, such as calcium fluoride or wax.

References:

Carter-Stiglitz, B., B. Moskowitz, and M. Jackson (2001). Unmixing magnetic assemblages and the magnetic behavior of bimodal mixtures, *Journal of Geophysical Research*, **106**, 26397-26411. (back to text)

Egli, R. (2004). Characterization of individual rock magnetic components by analysis of remanence curves, 1. Unmixing natural sediments, *Studia Geophysica et Geodaetica* **48**, 391-446. (back to text)

Jasonov, P. G., D. K. Nurgaliev, et al. (1998). A Modernized Coercivity Spectrometer, *Geologica Carpathica* **49**, 224-225. (back to text)

von Dobeneck, T. (1996). A systematic analysis of natural magnetic mineral assemblages based on modeling hysteresis loops with coercivity-related hyperbolic basis functions, *Geophysical Journal International* **124**, 675-694. (back to text)

3. Basic theory of coercivity distributions: all what you should know

Coercivity distributions still represent an unusual way to analyze a magnetization curve. Two major sources of confusion about coercivity distributions arise from the use of different field scales (e.g. linear and logarithmic) and from the related units. The shape of a coercivity distribution as well as its unit depend on the scale used for the magnetic field. Appendix A reports tables for the appropriate units to use for coercivity distributions. In the following, the mathematical and physical meaning of a coercivity distribution is discussed. Understanding this meaning is very important for a correct interpretation of the results obtained with CODICA.

Let *define* the coercivity distribution $\mathscr{M}(H)$ of a magnetization curve $M(H)$ as the absolute value of the first derivative of $M(H)$:

$$\mathscr{M}(H) = \gamma \left| \frac{\partial M(H)}{\partial H} \right| \tag{1}$$

whereby the factor γ depends on the type of magnetization curve and accounts for the Stoner-Wohlfarth relationships in non-interacting SD particles [*Stoner and Wohlfarth, 1948*]. Thus, $\gamma = 1$ for all acquisition and AF demagnetization curves, and $\gamma = 1/2$ for DC demagnetization curves (also called backfield curves). Coercivity distributions can be regarded as the statistical distribution of so-called *switching fields*. In some cases, the switching field can be identified with the field required to switch the magnetization of a single magnetic particle. This is for example true for an assemblage of non-interacting SD grains. However, this simplification does *not* apply to MD particles or assemblages of strongly interacting grains. The statistical character of a coercivity distribution is formalized by assuming $\mathscr{M}(H)$ to be proportional to a probability density function (PDF):

$$m(H) = M_0 f(H) \tag{2}$$

where the proportionality constant M_0 is simply the *total magnetization* of the sample. Conversely, $f(H)$ can be regarded as the coercivity distribution of a sample with unit magnetization.

Any PDF $f(x)$ can be represented on different scales of x through a variate transformation $y = g(x)$, where y is the variate on the new scale. In the case of coercivity distributions, let $H^* = g(H)$ be a scaled magnetic field. The coercivity distribution $\mathscr{M}^*$ on the new scale is given by:

$$\mathscr{M}^*(H^*) = \mathscr{M}(g^{-1}(H^*)) \frac{\partial g^{-1}(H^*)}{\partial H^*} \tag{3}$$

Notice that $\mathscr{M}^*$ cannot be plotted just by changing the scale of the horizontal axis because of the normalization property (2). Of special interest for the following discussion is the particular case given by $h = \log H$, whereby (3) becomes:

$$\mathscr{M}^*(h) = \ln 10 \; 10^h \mathscr{M}(10^h) \tag{4}$$

In the following, we designate with $\overline{f}(h)$ a PDF obtained from $f(H)$ with $h = \log H$. One could ask why a coercivity distribution should be calculated on different field scales. The reason for a scale change has profound roots in the mechanism that generates a PDF in nature. This mecha-

nism is so important for understanding coercivity distributions, that the subject is discussed in the following.

The coercivity distribution of an assemblage of magnetic grains depends on: (1) the statistical distribution $\mathscr{H}(H_{\mathrm{K}})$ of the microcoercivities H_{K} of the grains, (2) the coercivity distribution $\mu(H_{\mathrm{K}}, H)$ of all grains with the same microcoercivity H_{K}, and (3) the magnetostatic interactions between the grains. If magnetostatic interactions are negligible, the coercivity distribution of the grain assemblage is given by:

$$\mathscr{M}(H) = \int_0^\infty \mathscr{H}(H_{\mathrm{K}})\, \mu(H_{\mathrm{K}}, H)\, \mathrm{d}H_{\mathrm{K}} \tag{4}$$

For uniaxial SD grains as well as for MD grains, $\mu(H_{\mathrm{K}}, H) = \mu(H/H_{\mathrm{K}})$ [*Stoner and Wohlfarth, 1948*; *Xu and Dunlop, 1995*]. It is reasonable to assume that this relationship has a general validity. In that case, (4) becomes a simple convolution between $\mathscr{H}(H_{\mathrm{K}})$ and $\mu(H_{\mathrm{K}}, H)$ on a logarithmic scale:

$$\overline{\mathscr{M}}(h) = \int_{-\infty}^\infty \overline{\mathscr{H}}(h_{\mathrm{K}})\, \overline{\mu}(h - h_{\mathrm{K}})\, \mathrm{d}h_{\mathrm{K}} \tag{5}$$

$\overline{\mathscr{H}}$ is a PDF function that depends on the distribution of all physical (e.g. volume, shape, defects) and chemical (e.g. oxidation state) parameters that control H_{K}. It is typically a broad distribution. The only relevant exception in nature is represented by magnetosomes, whose size, shape and composition is strictly controlled by bacteria. $\overline{\mu}$ depends strongly on the domain state. In SD particles, $\overline{\mu}$ is a very narrow function with a peak at $h = \log(0.5 H_{\mathrm{K}})$. MD particles are characterized by a much broader μ with a typical exponential shape that depends on the grain size and dislocation density [*Xu and Dunlop, 1995*].

Both $\overline{\mathscr{H}}$ and $\overline{\mu}_{\mathrm{MD}}$ (where the index MD indicates MD particles) depend on the statistical distribution of physical and chemical processes that result form stochastic processes. For seek of simplicity, we discuss the case of $\overline{\mathscr{H}}$. Consider an initial collection of *identical* grains, all characterized by the same H_{K}. The magnetization curve of such grains is a step function, and the corresponding PDF is given by $\delta(H - H_{\mathrm{K}})$, where δ is the so-called Dircac δ-function (Fig. 8a). Imagine now that the magnetic grains are subjected to an alteration process – such as oxidation or corrosion – that changes H_{K} over time. This change might consist of a systematic term that reflects the *mean* effect of the alteration process to H_{K}, and a *random* or *stochastic* term. The latter depends on the heterogeneity of the alteration process, which might be more effective at some portions of space that are – for example – more exposed to a certain substance. The mean effect of the alteration process will be ignored in the following without loss of generality. A short time unit $\mathrm{d}t$ after alteration began, the microcoercivity of the particles is given by:

$$H_{\mathrm{K}}(t + \mathrm{d}t) = H_{\mathrm{K}}(t) \pm s(H_{\mathrm{K}})\mathrm{d}t \tag{6}$$

where s is a function that describes how the stochastic alteration process depends on H_{K}. For example, we can assume that the *relative* change of H_{K} after a unit time is, let say, 10%. In that case, $s = 0.1 H_{\mathrm{K}}$. Another choice could be that of a constant s, which is unrealistic because it produces negative values of $H_{\mathrm{K}}(t + \mathrm{d}t)$. Since (6) describes a random process, we expect half of

the grains to *increase* their microcoercivity, while the microcoercivity of the other half decreases. The coercivity distribution is now given by two peaks (Fig. 8b). The same alteration process is allowed to go on for another time unit $\mathrm{d}t$, whereby equation (6) is now applied to the *two* microcoercivites that resulted from the previous step. A coercivity distribution with four peaks is obtained in this way (Fig. 8c). If the simulation of this stochastic process is allowed to proceed for a long time, a clear trend characterizes the resulting coercivity distribution (Fig. 8d).

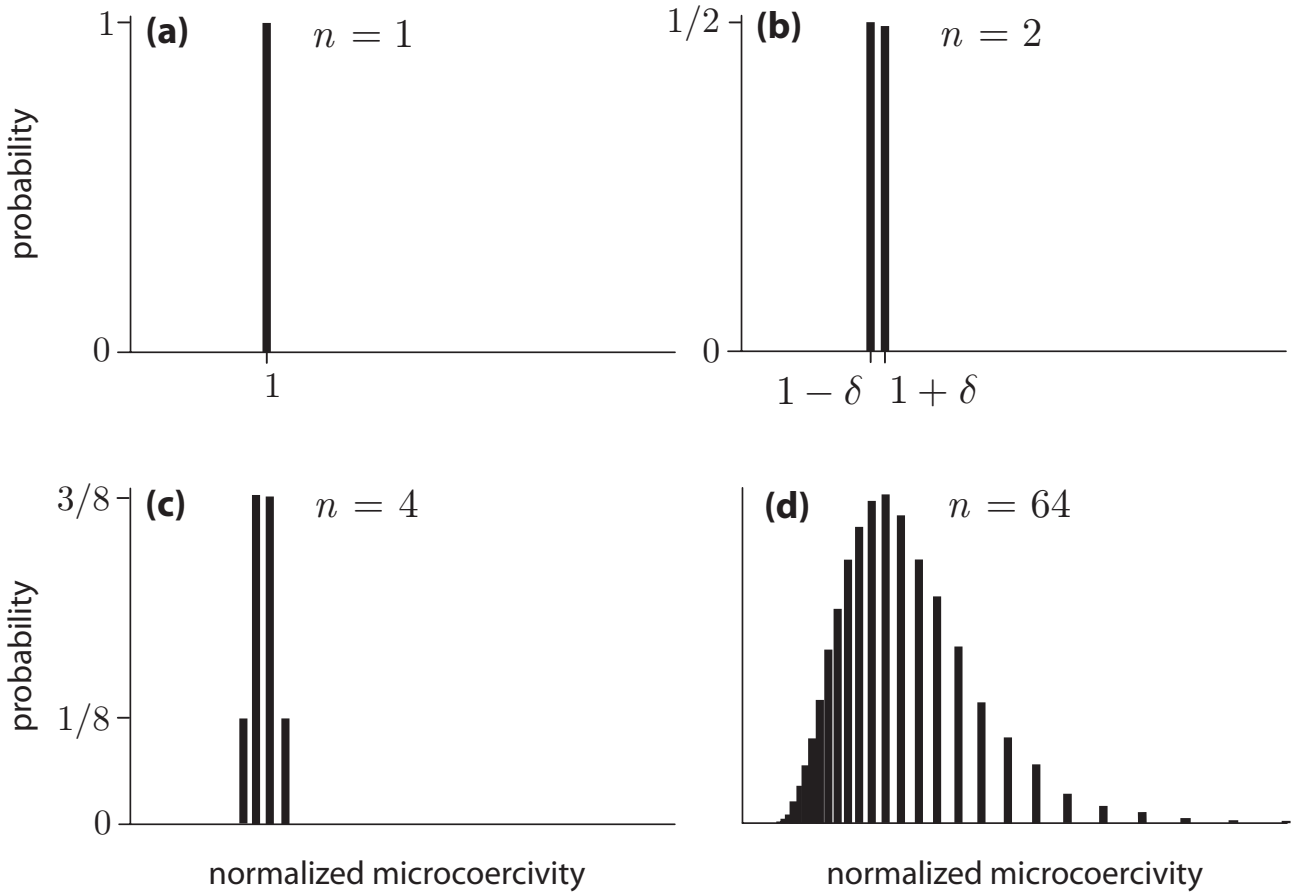


Fig. 8. Stochastic model of a coercivity distribution (redrawn from *R. Egli, Physics and Chemistry of the Earth, 29, 851-867, 2004*). **(a)** An initial set of identical particles has the same microcoercivity, $H_{\mathrm{K}} = 1$. The probability of having $H_{\mathrm{K}} = 1$ is obviously 1. **(b)** A random process changes the microcoercivity of the particles by a small amount $\delta(H_{\mathrm{K}})$ over a given time. Now, $H_{\mathrm{K}} = 1 - \delta$ or $H_{\mathrm{K}} = 1 + \delta$ with equal probability. **(c)** The same random process affects the microcoercivity over the next time interval, whereby each microcoercivity value changes again by δ. Four microcoercivities values are obtained in this way. **(d)** The probability distribution of the microcoercivities after six time intervals, with $\delta(H_{\mathrm{K}}) = 0.1H_{\mathrm{K}}$. This distribution approaches fairly well the limit case of an infinite number of time intervals, which is given by a logarithmic Gaussian PDF.

As the alteration process is going on, the coercivity distribution converges to a characteristic shape that depends only on s. PDFs obtained from a stochastic process as described above have an interesting property: when represented on an appropriate scale, they are *self-similar*. This means that if $f(x, \sigma)$ is a self-similar PDF with width parameter σ, the convolution of two such PDFs gives again the same self-similar PDF: $f(x, \sigma_1) * f(x, \sigma_2) = f(x, \sigma_1 + \sigma_2)$. In other words, the sum of random variates described by a self-similar PDF is a variate with the *same* self-similar PDF. Such PDFs have very peculiar mathematical properties that make them so important in

statistics and in the description of stochastic processes. The most famous self-similar PDF is the Gaussian (or Normal) function:

$$\mathcal{N}(x,\mu,\sigma) = \frac{1}{\sqrt{2\pi\sigma}} \exp\left[-\frac{(x-\mu)^2}{2\sigma^2}\right] \tag{7}$$

with mean μ and variance σ^2. This PDF can be generated by using (6) with a constant s to describe a so-called *additive* stochastic process. The process defined by $s \propto H_{\mathrm{K}}$ (Fig. 8) is a *multiplicative* stochastic process that generates the logarithmic Gaussian (or Normal) PDF:

$$\bar{\mathcal{N}}(x,\mu,\sigma) = \frac{1}{\sqrt{2\pi\sigma}\, x} \exp\left[-\frac{\ln^2(x\,/\,\mu)}{2\sigma^2}\right] \tag{8}$$

with median μ and dispersion parameter σ. The logarithmic Gaussian PDF itself is not self-similar, but it is transformed into the self-similar Gaussian PDF through the scale transformation $x^* = \ln x$. Thus, $x^* = \ln x$ can be regarded as the *natural scale* of the logarithmic Gaussian distribution. This example can be generalized to the following statement: *a stochastic process produces a PDF that is self-similar on an appropriate scale, hereon called the natural scale of that process*. A wide and very general class of self-similar PDF is represented by the so-called Lévy stable PDFs [*Sato et al., 1999*]. These functions are symmetric about their median μ. If we limit our considerations to this class of self-similar PDF we obtain a more useful statement: *a stochastic process produces a PDF that is symmetric if represented on its natural scale*. The representation of a coercivity distribution on its natural scale offers the advantage of a great simplification, since additional parameters for the asymmetry of the PDF are not necessary.

The stochastic nature of coercivity distributions brings us to the concept of *magnetic component*. If we define a magnetic component as *an assemblage of magnetic grains with a common origin and a common biogeochemical history*, the coercivity distribution of such a component is the result of the stochastic nature of all processes that led to the formation of the magnetic grains (e.g. processes of nucleation and growth), and their subsequent chemical/physical alteration and selection. According to this definition of a magnetic component and the above discussion about stochastic processes, *the coercivity distribution of a magnetic component is represented by a symmetric, self-similar PDF on an appropriate field scale*. We shall consider this statement as the fundamental hypotheses of component analysis, since it justifies the use of model functions for the coercivity distributions of each magnetic component.

The above discussion shows that there is a well-defined relationship between the natural scale of a PDF and the stochastic process that generated it. We recall the description of a stochastic process given by (6). The general form of (6) deals with a variate X with PDF f, whose change with time is given by:

$$X(t+\mathrm{d}t) = X(t) + s(X)\,Z\,\mathrm{d}t \tag{9}$$

where Z is a variate described by a PDF φ. Equation (9) is called a stochastic process. If s is *independent* of X, the PDF of $X(t+\mathrm{d}t)$ is given by the convolution of f and φ. The reiteration

of (9) generates an infinitely divisible PDF as $t \to \infty$. In the general case of s depending on X we shall consider the variate transformation $Y = g(Z)$, whereby

$$Y(t + \mathrm{d}t) = g[X(t) + s(X)\, Z\, \mathrm{d}t] \approx Y(t) + g'(X) s(X)\, Z\, \mathrm{d}t \tag{10}$$

A self-similar PDF is obtained from (10) if $g'(X)s(X)$ is independent of X, whence:

$$g(x) = \int \mathrm{d}x / s(x) \tag{11}$$

defines the natural scale of the stochastic process (9). Conversely, a PDF that is self-similar on a scale defined by $g(x)$ might result from a stochastic process with $s(x) = 1/g'(x)$.

The hypothesis that coercivity distributions can be modeled with self-similar PDF on an appropriate field scale is difficult to confirm for natural magnetic components, since rocks and sediments are almost always mixtures of at least two components. *Robertson and France* [1984] first realized that the coercivity distribution of synthetic assemblages of magnetite and hematite particles can be modeled with logarithmic Gaussian functions. They implicitly introduced the logarithmic scale as a *natural* scale for coercivity distributions, and the Gaussian function as the self-similar PDF. Detailed measurements of sediments and rocks demonstrated that the coercivity distributions of natural magnetic components are slightly skewed on a logarithmic scale [*Egli*, 2004]. Their natural scale has been tentatively approximated by $H^* = H^p$, with $p > 0.5$ for SD magnetite and $p \approx 0.2$ for other natural magnetites. Since the observed natural scales are peculiar to each component and are probably not universal, the logarithmic scale can be still considered as a fairly good scale for plotting coercivity distributions. Some cases that confirm the fundamental hypotheses of component analysis are shown in Fig. 9.

The representation of a coercivity distribution on different field scale is a very effective tool for checking whether the distribution is strictly unimodal or it is the result of the overlapping of two or more magnetic components. Subtle details that are completely hidden in one field scale become evident if a different scale is chosen. *Egli* [2003] discussed extensively the use of the power scale transformation defined by:

$$H^* = H^p \tag{12}$$

with the scaling exponent $p > 0$. For $p \to 0$ the power scale transformation converges to the logarithmic transformation $H^* = \ln H$. Thus, (12) provides a set of scales that contain the logarithmic and the linear scales as a special case. Fig. 10 shows an example where the the power scale transformation has been used to check the unimodality of a coercivity distribution.

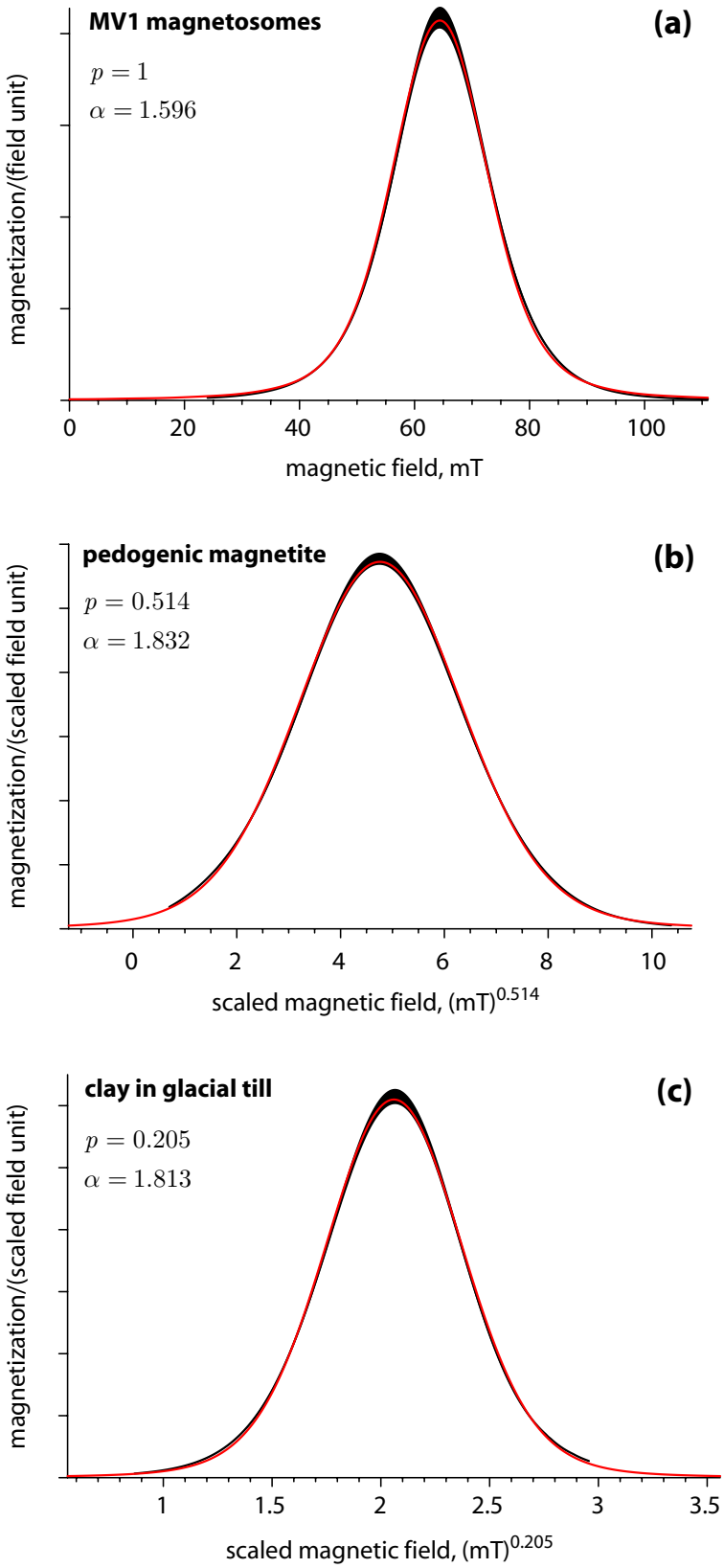


Fig. 9. Validation of the stochastic model for the coercivity distribution of some natural magnetic components. First, a field scale transformation $H^* = H^p$ has been applied to the measured coercivity distributions (thick black lines) . The scaling exponent p has been chosen so that the transformed coercivity distribution is symmetric (e.g. its skewness is zero). The estimated measurements errors are given by the thickness of the black line. The scaled coercivity distribution has been fitted with a Lévy self-similar PDF $\mathscr{L}(x,\sigma,\alpha)$ with width parameter σ and shape parameter α (red line). Special cases of $\mathscr{L}(x,\alpha)$ are the Gaussian PDF ($\alpha = 2$), and the Cauchy PDF ($\alpha = 1$). The parameter α controls the squareness of $\mathscr{L}(x,\alpha)$: tip-shaped PDFs with heavy tails are obtained with $\alpha < 2$. **(a)** Coercivity distribution obtained from the AF demagnetization of ARM for cells of the magnetotactic bacterium MV1 mixed with pure kaolin. Notice that a scale transformation was not required. The coercivity distribution is heavily tailed, as indicated by the low α. **(b)** Coercivity distribution obtained from the AF demagnetization of ARM for pedogenic magnetite in a soil developed from glacial till in Minnesota. The coercivity distribution of the pedogenic component has been obtained by subtracting the contribution of the glacial till from the demagnetization curve. **(c)** Coercivity distribution obtained from the AF demagnetization of ARM for the clay fraction of a glacial till in Minnesota. Notice that the coercivity distributions in (b) and (c) are plotted on different field scales, but are characterized by almost the same Lévy self-similar PDF with $\alpha \approx 1.8$.

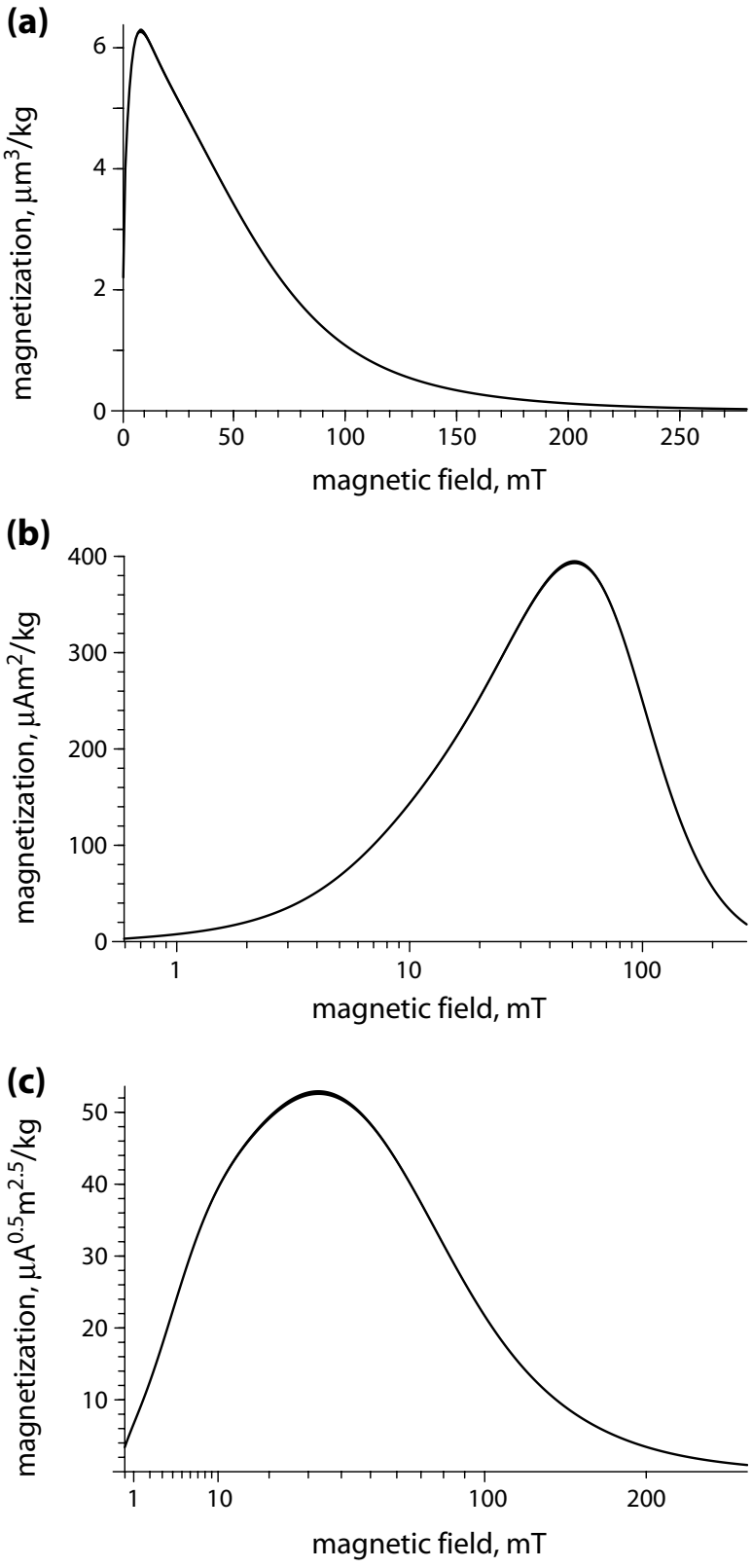


Fig. 10. Example of the use of different field scales to check whether a coercivity distribution is unimodal or is the result of two overlapped magnetic components. The sample was a filter used to collect particulate matter from the atmosphere of a rural area in Switzerland [*Spassov et al.*, 2004]. Unlike other samples collected in polluted areas, this sample was expected to contain only one magnetic component carried by the natural dust. The coercivity distribution of this sample has been calculated with CODICA on **(a)** a linear scale, **(b)** a logarithmic scale, and **(c)** a field scale defined by $H^* = H^{1/2}$. Notice that on this last scale the coercivity distribution is not perfectly unimodal, suggesting that the sample might contain a small contribution from far located anthropogenic pollution sources.

References:

Egli, R. (2003). Analysis of the field dependence of remanent magnetization curves, *Journal of Geophysical Research-Solid Earth* **108**(B2), doi: 2081. (back to text)

Egli, R. (2004). Characterization of individual rock magnetic components by analysis of remanence curves, 2. fundamental properties of coercivity distributions, *Physics and Chemistry of the Earth*, **29**, 851-867. (back to text)

Robertson, D. J. and D. E. France (1994). Discrimination of remanence-carrying minerals in mixtures, using isothermal remanent magnetization acquisition curves, *Physics of the Earth and Planetary Interiors*, **82**, 223-234. (back to text)

Sato, K., B. Bollobas, W. Fulton, A. Katok, F. Kirvan, and P. Sarnak (eds) (1999). *Lévy Processes and Infinitely Divisible Distributions*, Cambridge studies in advanced Mathematics, **68**, Cambridge University Press, 479 pp. (back to text)

Stoner, E. C. and E. P. Wohlfarth (1948). A mechanism of magnetic hysteresis in heterogeneous alloys, *Philosophical Transactions of the Royal Society of London, Series A*, **240**, 599–602. (back to text)

Xu, S. and D. Dunlop (1995). Toward a better understanding of the Lowrie-Fuller test, *Journal of Geophysical Research B: Solid Earth*, **100**: 22533-22542. (back to text)

4. Install and run CODICA 5.0

Requirements

To run CODICA 5.0 you need *Mathematica 5.0* and later versions installed on a Windows OS. At least 128 MB RAM and a 1 GHz CPU are recommanded.

Install CODICA 5.0

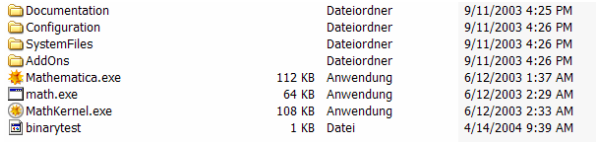
To install CODICA 5.0 copy the source code file `MAG_MIX_1/CODICA/Install/Codica.m` into the following directory:
`C:/.../Wolfram Research/Mathematica/5.0/AddOns/StandardPackages/Utilities`
whereby `C:/.../` depends on the installation of *Mathematica* on your computer.

A short introduction to Mathematica (all what you need to know about)

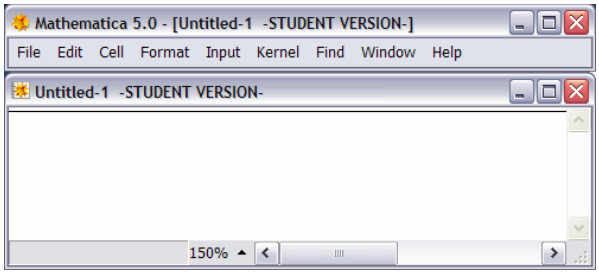
Mathematica is a software conceived to perform high-demanding logic, symbolic and numeric mathematic operations. It integrates a numeric and symbolic computational engine, graphics system, programming language, documentation system, and advanced connectivity to other applications.

When you launch *Mathematica*, a so-called notebook is opened. This notebook is initially empty. A *Mathematica* notebook is simply a command shell to enter commands and see results. You can save and edit the notebook, and export graphics and other output results. Every *Mathematica* notebook contains input lines, numbered by `In[#]`, where `#` is the input number, and output lines, numbered by `Out[#]`, where `#` is the number of the corresponding input. Everything you type in a notebook is interpreted as a command line, and an input number is automatically assigned when you ask to run the command line. You can do so by pressing the keys *Shift* + *Enter* at the same time (Example 1). *Enter* alone is used to enter a new line. When you evaluate the first command of a notebook, the so-called *Mathematica kernel* is launched. The kernel is the core of *Mathematica* that performs all the calculations requested through the input lines.

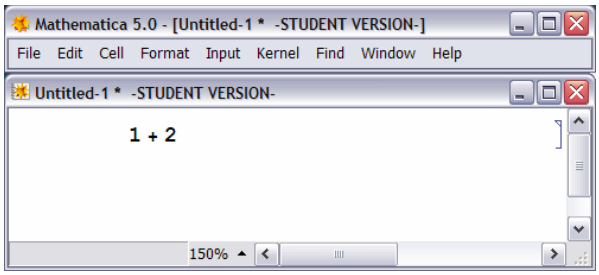
Example 1: A simple run with Mathematica.



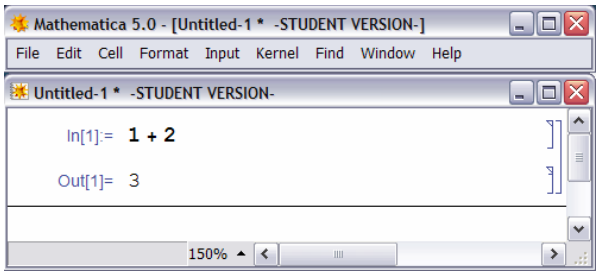
1. Launch *Mathematica.exe*



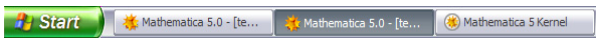
2. An empty notebook appears



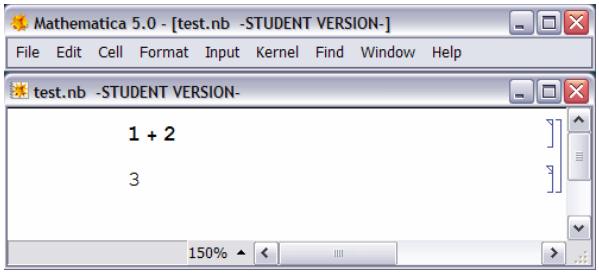
3. Type a command, for example "1+2"



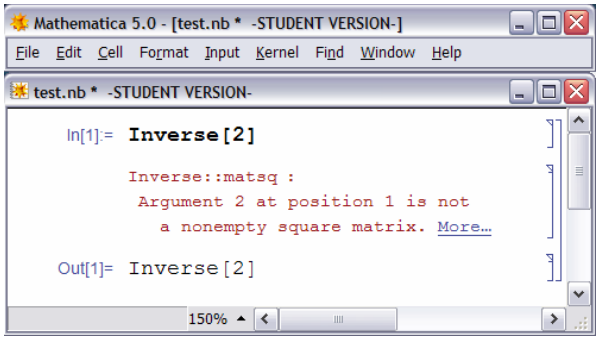
4. Press *Shift* + *Enter* to evaluate the command. Now, "1+2" is the first input, In[1], and the result "3" the first output Out[1].



5. After evaluating the first input, the *Mathematica Kernel* is started automatically.



6. You can save the notebook by clicking on the Menu "File" and then "Save". By double-clicking on the saved notebook you can launch *Mathematica* and load the notebook with all stores inputs and outputs automatically.



7. If you try to evaluate a wrong expression, an error message appears.

When you try to evaluate a wrong expression, an error message appears. Usually, the kernel is still running properly after encountering an error, however, in case of very complicated calculations, the kernel may stuck. In such cases, quit the kernel from the task list or by invoking the task manager.

Load CODICA in a Mathematica session

To work with CODICA you need to load the program from a *Mathematica* session. Open a new Mathematica notebook by clicking on the *Mathematica* icon. An empty window will appear. Type *exactly* the following string: `<<Utilities`Codica`` , whereby the symbol " ` " is the grave accent. If your keyboard does not provide this accent, open an example notebook provided with this manual. You find this string at the beginning of the notebook. Press the keys *Shift* and *Enter*

at the same time to upload CODICA. After CODICA is uploaded, a welcome message appears on the screen. To run CODICA, type `Codica`, followed by *Shift* + *Enter*. After this step, CODICA interacts with the user via promt windows.

5. CODICA 5.0 tutorial

Introduction

This tutorial is intended to provide you with a basic knowledge of CODICA. The functions of the program are illustrated step by step with an example based on real data. Each step is marked by a book symbol followed by a number: (e.g. 🕮 1). Click on the book symbol or other interactive topics to jump to the example at the end of this section. You also find this example in the file `MAG-MIX_1/Codica/Examples/Example.nb` provided with this manual. You can open this file and run the example by yourself to familiarize with CODICA. Change the input parameters to see how the results are affected. To run the example you need to copy the folder `MAG-MIX_1/Codica/Examples/data` provided with this manual onto your computer. This folder contains the data used for the example. The leading example of this tutorial was chosen to activate all the options of CODICA. In a typical run with your data, you will encounter only some of these options.

This tutorial provides you with all the important informations you need to run CODICA and understand the basic principles of the data processing. Details about how CODICA performs the data processing are provided in section 6. You do not need to read that section unless you are interested in the software development or in the mathematics behind the calculation of coercivity distributions.

Structure of CODICA

CODICA interacts with you through prompt windows. When a prompt window appears, you are asked to enter your answer in the prompt window. If a prompt window is open and you click on any part of the *Mathematica* notebook, the notebook may hide the prompt window. If you need to scroll through the notebook while the prompt window is open, shift the latter to one side, so that you can click on it when you are ready to enter your answer.

While running, CODICA generates different types of outputs in the *Mathematica* notebook: (1) progress messages (black), (2) information messages (blue), (3) warning messages (red), (4) critical warning messages (bold, red), and (5) graphics. CODICA will stop after critical warning messages, because any further data process is impossible. Critical warning messages indicate that your data cannot be analyzed, probably because they do not represent a correctly measured magnetization curve.

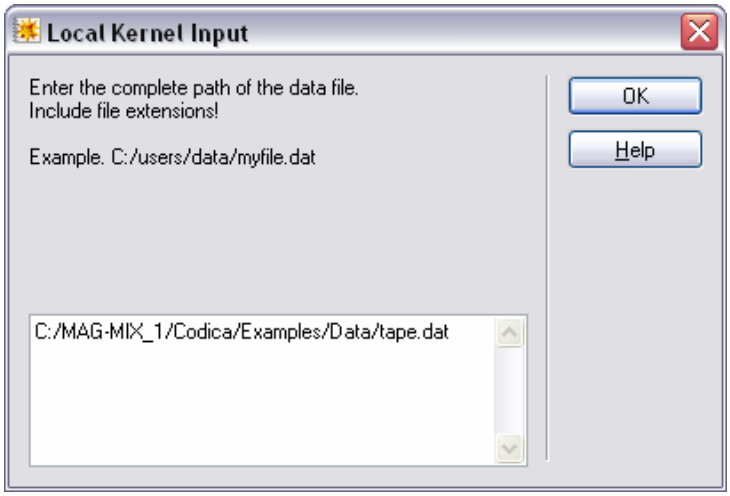
At the end of a CODICA session, useful results are saved into different files that can serve as input files for other programs of the MAG-MIX package. You can open these files with any other program capable of reading numerical data, such as *Excel*, *Gnuplot, Kaleidagraph* or *Origin*.

Exit CODICA

You can exit CODICA before the program is finished by (1) typing `abort` in any prompt window produced by CODICA, or (2) quitting the *Mathematica* Kernel. These options are useful if CODICA is producing unexpected error messages or if you want to check the data file and restart again.

📖 1. Enter the path of the data file

When you start CODICA, a prompt window asks you to enter the full path of the file that contains your data. Type *exactly* the complete path, with correct upper- and lower-cases and *including* the extension of the file (e.g. `.txt` or `.dat`). To avoid typing long paths it is recommended to store your data in a easily accessible folder (e.g `C:/users/myself/datafile.dat`).



CODICA can read any file that contains at least two columns of numbers. Header and footer lines will be ignored. The first column is supposed to contain the field values, the second column the corresponding measurements. Eventual further columns to the right will be ignored. The colums must be separated by at least one of the following characters: *space*, *tabulator*, or the punctuation signs " , " , " : " , " ; ". CODICA starts to read the file at the first row that has the appropriate format and stops when it encounters a row that has not the appropriate format (e.g. a footer line) or at the end of the file. Therefore, header lines must *not* begin with numbers in a column format.

➔ If CODICA experiences problems in reading the file, remove the header. If CODICA does not find the file, or if the file format is wrong, a critical error message is displayed and the program is stopped.

📖 2. Check the measurements

CODICA checks the validity of the measurements. Valid measurements represent any kind of magnetization or demagnetization curves but *not* hysteresis loops. Ideally, the magnetization curve starts at a zero field, and the field increases (decreases) monotonically to a maximum (minimum) value. Some automated measurement systems merge different sets of data (e.g. an IRM acquisition curve, followed by a DC demagnetization). CODICA selects all data that define the *first* magnetization curve encountered in the data file. If you want to analyze the successive magnetization curves, you have to save them in a separated file. The field increment steps of automated systems may be smaller than the error of the applied field: in such cases the fields are not strictly monotonic, but have nevertheless a monotonic trend. CODICA stops to read the data as soon as the increasing/decreasing *trend* of the applied field is inverted. Because of measurement errors and other instrumental effects, initial values of the field may have the wrong sign (e.g. small negative fields instead of zero, if the nominal applied field is positive). CODICA skip these fields automatically. A magnetization curve must contain *at least* 9 measurements in order to be analyzed by CODICA, and at least 20 measurements are recommended. If the checking procedure fails, a critical error message is displayed and the program is stopped. Otherwise, the measured magnetization curve is plotted.

➔ *Multiple measurements.* Multiple measurements are sequences of measurements characterized by the same field. CODICA detects multiple measurements and calculates an average value. A warning message is displayed. You should ignore this message if the data file has been produced automatically by a measurement system. However, if the measurement steps were performed by hand, you should consider the possibility that some steps were remeasured and only the last value of a multiple measurement might by correct. In this case, you should exit CODICA and check the data file manually.

➔ *More than 300 measurements.* Error-free magnetization curves are smooth. The amount of data produced by some automatic system does *not* increase the resolution of a magnetization curve: the highly redundant data are helpful only in reducing the effects of measurement errors. The least-squares collocation method used by CODICA to filter the data is very efficient in removing measurement errors; however, the amount of mathematical operations required is proportional to the square of the number of measurements. Accordingly, the computation time increases enormously for large dataset. If the dataset contains more than 300 measurements, CODICA uses a moving average filter to reduce the number of data. This procedure does *not* discard information from the original file, and you should not be worried about the quality of the results. A warning message remembers you that CODICA is dealing with a reduced amount of points.

🕮 3. Evaluate the properties of the magnetization curve

Before analyzing the magnetization curve, CODICA evaluates its general properties, such as the initial and the final magnetization, the median field and the 75% quantiles. These parameters are necessary for further processing of the data. CODICA prints a summary of these properties. The information reported in this summary does not affect the final result of the program.

➔ *Digital rounding.* Digital rounding effects occur when the digitalization of measurements by the acquisition software produces errors that are *larger* than the measurement error of the magnetometer. The resulting magnetization curve contains characteristic steps that are recognized by CODICA (Fig. 5). A warning message is displayed, together with a list of points where digital rounding has been detected. At places where steps have been identified, CODICA reorganizes the measurements in order to minimize the negative consequences of digital rounding. This is done by taking the arithmetic mean of the two measurements across a step produced by digital rounding, and by discarding all the other measurements between two or more such steps. By doing so, the maximum error introduced by rounding to the n-th digit is reduced from 10^{-n} to the half. In case of large amounts of data produced by an automatic measurement system, some small jumps in the magnetization curve can occur by chance. CODICA might confuse them with digital rounding effects, without negative consequences on the data processing.

🕮 4. Search an optimal field scaling exponent

As discussed in section 3, the magnetic field can be rescaled to obtain a symmetric coercivity distribution. This applies to a magnetization curve as well. As a first step in the data processing, CODICA looks for a scale transformation that gives the most symmetric magnetization curve. The scale transformation used by CODICA is a power transformation $H^* = (H + q)^p$, where q is a damping term and p is the scaling exponent (see section 6.2 for more details). Usually, sym-

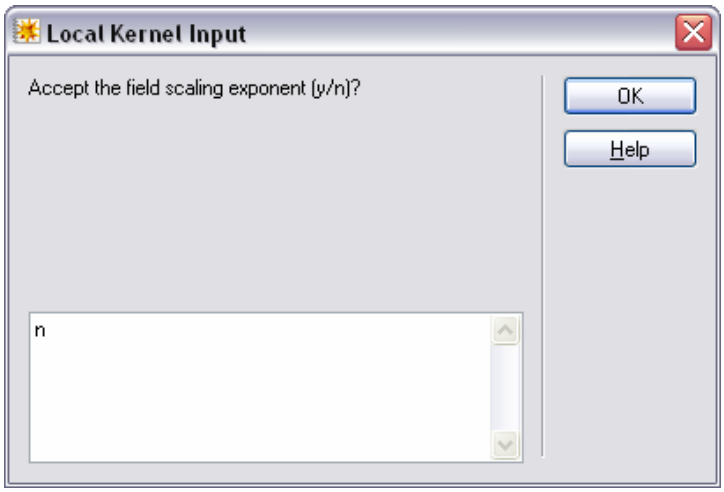
metric magnetization curves are obtained with $0 < p < 0.5$. Accordingly, CODICA explores systematically exponents $p < 1$. This operation can take some time, depending on the number of data. If an optimal value is not obtained with $p < 1$, CODICA will try with $p > 1$. Since large values of p can produce numerical problems, the search is limited to $0.01 \leq p \leq 2.7$. This range is adequate for all types of magnetization curves. However, it can seldomly occur that the search for an optimal value of p fails. In this case, CODICA will assume $p = 1$. The result of this optimization is an exponent p_{sym} for best symmetry of the magnetization curve (see section 6.4).

The performance of CODICA is influenced by the distribution of the field values. The ideal distribution of field values produces equally spaced points when the magnetization curve is scaled for best symmetry. Typically, small steps are required at small fields, and large steps at large fields. However, the ideal choice of field values depend also on the error affecting the applied field. Generally, a precise control of the field coil become increasingly difficult at large fields. Accordingly, more measurement are required at high fields to compensate for this effect. Many automated measurement system use a constant field increment which compensates for the larger error at high fields. If the distribution of field values becomes too irregular when plotted after the scale transformation defined by p_{sym}, large "holes" may result in the scaled magnetization curve. To avoid this situation, CODICA compares the scale transformation with the distribution of field values and finds a compromise between both.

The results of the operations described above is summarized in a table that reports the mean misfit between the scaled magnetization curve and a reference hyperbolic function used to evaluate the symmetry. The misfit is expressed as % of the amplitude of the magnetization curve for (1) the original field scale, (2) the scale for the most equally spaced fields, (3) the scale for the most symmetric curve, (4) the scale suggested by CODICA, which is a compromise between (2) and (3).

CODICA uses the suggested field scale to fit the magnetization curve with an asymmetric hyperbolic function (see section 6.3 for more details). The shape of this function is controlled by 8 parameters and is flexible enough to account for a variety of magnetization curves, including difficult cases given by mixtures of low- and high-coercivity minerals, such as magnetite and hematite. The best-fit curve is plotted together with the measurements. This fit is *not* the final model of the magnetization curve and you should not worry too much about discrepancies with the measured data.

A prompt window asks you whehter the scale used is acceptable or not. The scale suggested by CODICA is *almost always* acceptable. If you are not an experienced user, you should accept this scale by typing the letter "y" for yes. More experienced users can choose a different scale by typing "n".



➔ The choice of a field scale is not critical for the quality final results, so you should not worry too much about it. CODICA automatically avoids critical scale transformations, and accepting the suggested scale is seldomly a bad choice.

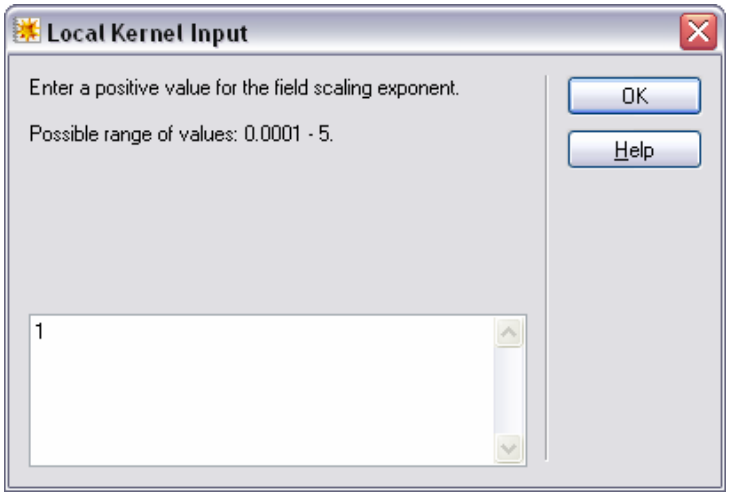
If you decide to enter the scaling exponent manually, you should read section 6.3 and feel confident with the mathematics behind. Avoid exponents >1 if possible. A manual choice of the scaling exponent can be taken into consideration in the following cases: (1) the scaling exponent suggested by CODICA is >1, (2) the plotted best-fit curve does not fit the measurements equally well at small and at large fields, (3) you want to see how the magnetization curve is affected by different values of the scaling exponent.

➔ For illustrative purposes only, the scale suggested by CODICA in this example was rejected.

5. Entering a scaling exponent manually

This step is necessary only if you wand to discard the field scale suggested by CODICA.

A prompt window asks you to enter a positive scaling exponent in the range between 0.0001 and 5. You should consider *very carefully* the reasons for entering exponents >1.5. The exponent you enter in this window is not definitive: you will have the possibility to reject it and make a different choice.



The performance of CODICA is not guaranteed for scaling exponents >2.7.

➔ A criterion for choosing the scaling exponent is given by the difference between the measurements and the best-fit curve. If this difference is highest at small fields, you should choose a *smaller* scaling exponent. On the other hand, if the difference is highest at large fields, you should choose a *larger* scaling exponent.

CODICA will rescale the measurements according to the exponent you entered, and recalculate a best-fit with an asymmetric hyperbolic function. After the results are plotted, a prompt window will ask you to accept or reject the field scale. If you do not accept the field scale, a prompt window asks you for a new scaling exponent, until you accept a solution.

➔ It is strongly recommended to either accept the scaling suggested by CODICA, or try several different scaling exponents. You can compare the results by looking the mean standard deviation of the fitted curve reported in the title of the plot.

📖 6. Calculate the residuals

The functions used to fit the magnetization curve serves as a model to rescale the measurements in such a way, that the magnetization curves is transformed into a nearly straight line. This operation is called *linearization* of the magnetization curve. The linear trend is then subtracted from the linearized magnetization curve to obtain so-called residuals (see section 6.6 for the mathematical details of this operation). The residuals are just a kind of difference between the measurements and the model used by CODICA to fit them, plotted on a suitable scale. A typical residual curve contains several oscillations around a mean value of zero. These oscillations are the sum of (1) very detailed features of the magnetization curve – such as those arising from the overlapping of different magnetic components –, and (2) measurement errors. Measurement errors are highly amplified in the residual curve and can be recognized clearly. The scope of transforming the magnetization curve into the residual curve is that data processing on the latter is much more efficient. The residual curve is handled as a stochastic signal, which is characterized by a so-called autocorrelation distance. The autocorrelation distance is the typical range of fields over which the error-free residuals are oscillating around a mean value of zero. CODICA uses a least-squares collocation method to model the error-free residual curve and make a first estimate of the measurement errors (see sections 6.7 to 6.10 for the mathematical details). The error-free residual curve is characterized by the typical range of fields spanned by individual "wiggles" of the curve.

The results obtained as described above are plotted in a graphics with (1) the residuals (dots), (2) the least-squares collocation model (black line), and (3) a first estimate of the measurement errors (gray area around the points).

➔ The appearance of the estimated measurement errors in the plot depends strongly on your data. If you measured a sample whose magnetic properties are dominated by a single magnetic component, the magnetization curve fits well with the model function used by CODICA. In such case, a large contribution to the residuals comes from the measurement errors. Accordingly, the estimated measurement errors are larger than the amplitude of the residual curve. You should not be worried about this result: it does *not* mean that the quality of the measurements was low. On the other hand, some mixtures of different magnetic components fit less well with the model function used by CODICA. The contribution of the measurement errors to the residual curve is small, and the data are apparently "cleaner".

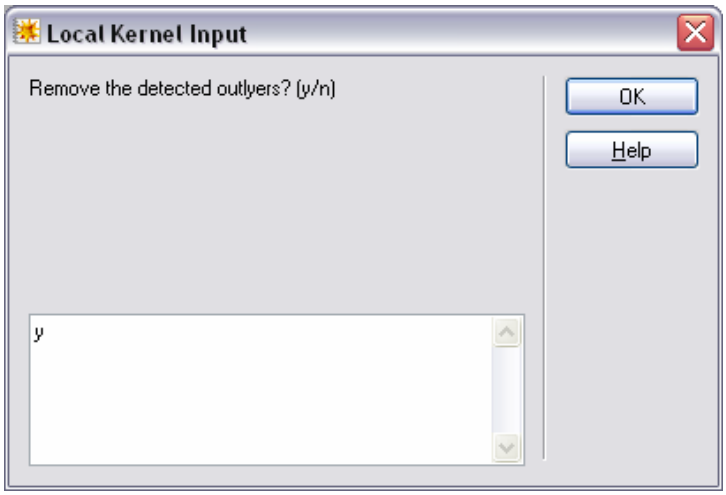
➔ The error estimation produced by CODICA at this point is not definitive, and you might find it incorrect. You should not worry about, since the definitive error estimation is performed in a later stage of the data processing. This error estimation is intended to give you an idea about the significance of the residuals. As a general role-of-thumb, the reliability of the error estimation is proportional to the number of measurements. Since the modeled residual curve depends on the error estimate, it might also be incorrect at this stage of the data processing.

📖 7. Remove outliers

The magnetization curve might contain some odd measurements produced for example by interferences with other devices in the laboratory, or by spikes in the electrical power supply. Often, these odd measurements are barely visible in a standard plot of the magnetization curve, but become very evident when the residuals are plotted. These odd measurements are recognized by

CODICA as *outliers*, when they fall apart the overall trend of the residuals by more than three times the estimated measurement error.

Outliers are marked as red points in the plot of the residuals. If some outliers have been detected, a prompt window will ask you if you want to remove those points from further data processing.



➔ The human brain has a unmatched visualizing and pattern recognition ability. Therefore, you will recognize outliers much easier and reliably than CODICA can. If you think that the points marked by CODICA as outliers are normal measurements, you can choose to keep these points for further data processing.

➔ If you accept to remove the outliers, CODICA will repeat all the previous data processing steps *without* these points. It is strongly recommended to remove outliers for better final results. Outlyers may bias many estimated parameters used by CODICA during the data processing.

➔ After removing outliers, it is possible that a second run through the data reveals other "minor" outliers. You can repeat the process of removing outliers until you are satisfied with the result.

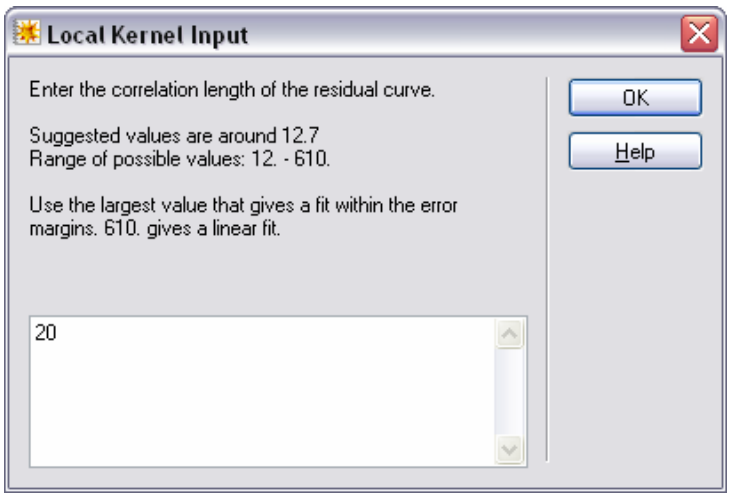
The case where the first and/or the last point are considered as outliers needs special attention. Often, these points are not really outliers: they rather reflect some particular processes occurring at the beginning or at the end of the magnetization curve. The initial part of a demagnetization curve is possibly affected by viscosity effects. On the other hand, the last step of a demagnetization curve is probably performed after treating the sample with the same magnetic field as the field used to impart the magnetization. Thermal activation effects might be responsible for a mismatch between the "effectiveness" of the magnetic field during acquisition and later demagnetization. For example, if a 200 mT field was used to impart an IRM, your sample might be completely AF demagnetized already at 180 mT. In this case, the magnetization curve becomes suddenly "flat" near the last measurement point. The first and the last point of a magnetization curve can therefore represent an "anomalous" trend or be real outliers.

8. Filter the residuals

At this point, CODICA has calculated a model for the residuals and the measurement errors. As already mentioned, the human brain has a unmatched ability in visualizing and recognize patterns. The measurement errors are highly enhanced in the residual curve, and your eyes are instinctively able to "see" them. By doing so, you have the ability to "tune" the CODICA model of the residuals. If CODICA underestimated the measurement errors, you will find that the modeled residual curve follows too closely the measurement points and has an excessively irregular ap-

pearance. On the other hand, if CODICA overestimated the measurement errors, the modeled residual curve is too smooth and important details of the residual curve are lost.

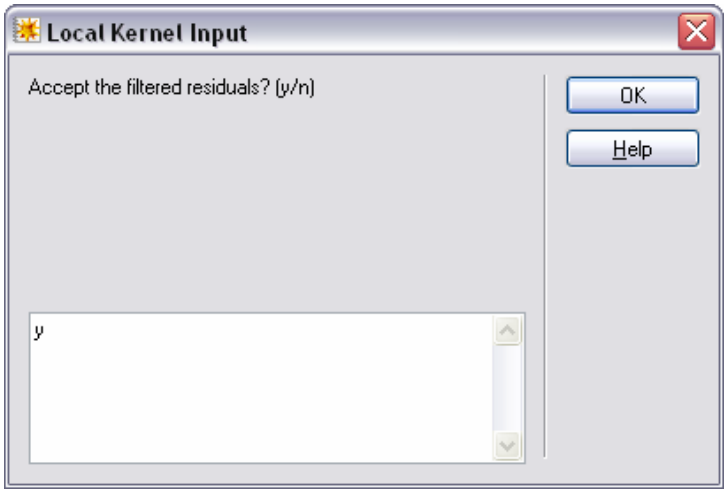
You can choose the degree of "smoothness" of the residuals model curve with a parameter called *correlation length*, that you are asked to enter by a prompt window.



The correlation length is used by the least-squares collocation model as a parameter that indicates the scale of the smallest details in the residual curve. Filtering the residuals with a correlation length r_0 means that features with an extension $< r_0$ are filtered out. The larger is r_0, the smoother will appear the filtered residual curve (see section 6.11 for mathematical details).

After entering the correlation length, CODICA will calculate the corresponding least-squares collocation model, and a graphics with the new estimate of the residual curve and the measurement errors will be displayed.

A prompt window will ask you if you want to accept the displayed model. Type "y" if you are satisfied with it, or "n" if you want to enter a different value for the correlation length.



➔ You can try as many models with different correlation lengths as you want, until you are satisfied with the result.

➔ If you should choose r_0 so, that the modeled residual curve follows the general trend shown by the residuals without being affected by the errors. To help you in your choice, CODICA proposes a starting value from its own estimate of the residual curve. This is generally a good starting point.

➔ There is a range for the possible values of r_0 you can choose. The lower limit is given by the maximum distance between two consecutive residuals, the higher limit by the range of fields spanned by the measurements. If you choose values close to the lower limit, the filtered residuals

will be very close to the measurements and will probably be affected by "wiggles" produced by the measurement errors. On the other hand, choosing values close to the upper limit gives you a straight line close to zero. You might choose this latter option if you have the impression that the residuals are completely "noisy".

➔ It is recommended for new users to play with r_L by choosing different values in the entire permitted range. You will soon get a "feeling" of how least-squares collocation filtering works. It is also recommended to run CODICA to the end with different values of r_0 to see the difference between the results obtained.

A dangerous temptation for many users is to choose too small values of r_0 to keep all "information" contained in the data. You should remember that (1) magnetization curves are naturally smooth and do not change much over a few mT range (except if you are measuring special materials such as thin films), and (2) short-range oscillations, such as "wiggles", are produced by measurement errors that do not have a completely random appearance (so-called "pink" noise).

➔ Ideal, "white" noise is completely uncorrelated and is easily recognized and removed. However, measurement errors can be correlated, for example if they arise from fluctuations of the ambient field during measurements. In this case the measurement errors have their own correlation length. Least squares collocation can remove measurement errors efficiently if the correlation length of the noise is much smaller than the correlation length of the residuals.

📖 9. Calculate the filtered measurement curve

Once you accepted a model for the residuals, CODICA is ready to calculate the filtered, supposedly error-free magnetization curve. To do so, all previous steps are inverted, starting from the filtered residuals, in order to obtain again a magnetization curve (see section 6.12 for mathematical details). The magnetization curve is plotted together with the original measurements. The estimated error of the modeled magnetization curve is plotted as well.

➔ If the quality of the measurement was good, you probably will not notice any difference between the raw measurements and the magnetization curve calculated by CODICA. However, the differences become evident if you compare a numerical derivative of the magnetization curve obtained using the raw data on one hand, and the magnetization curve calculated by CODICA on the other.

➔ The estimated error of the modeled magnetization curve is typically small and peaks at the median field, where the slope of the magnetization curve is highest. To understand this result, keep in mind that measurement errors do not arise only from the magnetometer, but also from applied field. Errors in the applied field are more visible at places where the magnetization curve has a large slope.

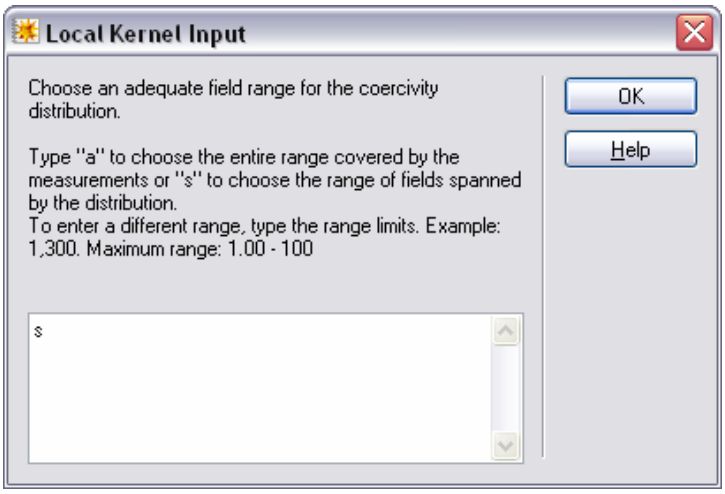
📖 10. Calculate the coercivity distribution

The coercivity distribution is calculated in a similar way as the magnetization curve, starting from the first derivative of the modeled residual curve and inverting all th steps used to transform the magnetization curve into the residuals (see section 6.13 for mathematical details). You have several options for calculating the corcivity distribution on different scales and for choosing the appropriate range of fields covered by the coercivity distribution.

11. Coercivity distribution on a log scale

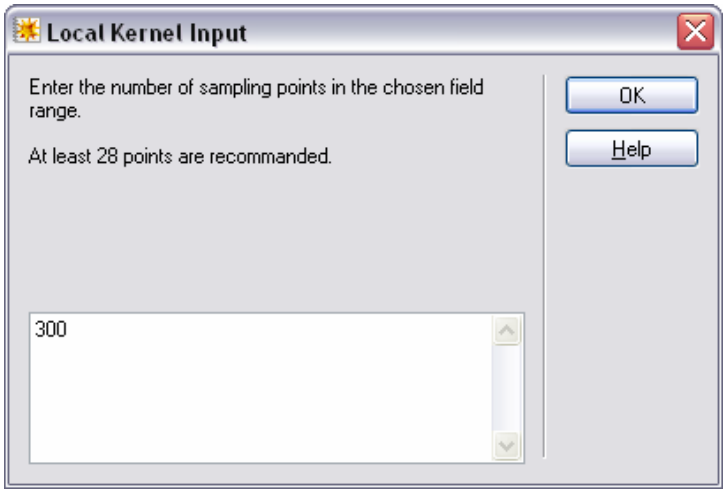
Because of the particular importance of the logarithmic field scale, CODICA will first display the coercivity distribution on this scale over the entire range of fields spanned by the measurements.

A prompt window will ask you to choose a range of fields over which CODICA should recalculate the coercivity distribution on a large number of points. You can choose the entire range of fields spanned by the measurements (type “a”). In some cases the coercivity distribution is significantly different from zero on a smaller range. You can choose this range by typing “s”. You can also specify a different range by typing its limits.



If you enter the field range of the coercivity distribution manually, keep in mind that you can only chose a range that is covered by measurements. CODICA does *not* allow extrapolations outside this range.

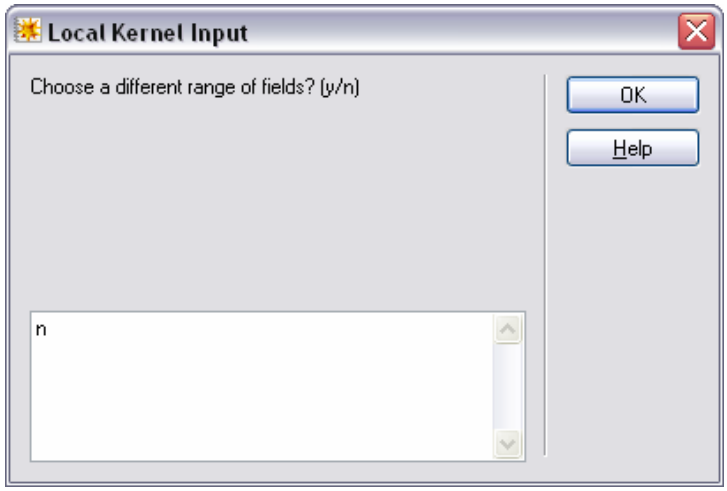
A second prompt window will ask you to enter the number of points where CODICA should calculate the coercivity distribution. These points will be equally spaced within the range you chose previously. CODICA suggests you to use a number of points that is at least as large as the number of measurement. However, you are free to enter any number of points you desire.



After entering the field range and the number of points, CODICA will recalculate and plot the coercivity distribution on the field range you chose. The thickness of the curve (small errors), or the grey region around it (large errors), indicates the estimated 95% confidence interval of the coercivity distribution. The error estimate is also plotted below the coercivity distribution.

➔ You can choose to recalculate the coercivity distribution on a different range of fields.

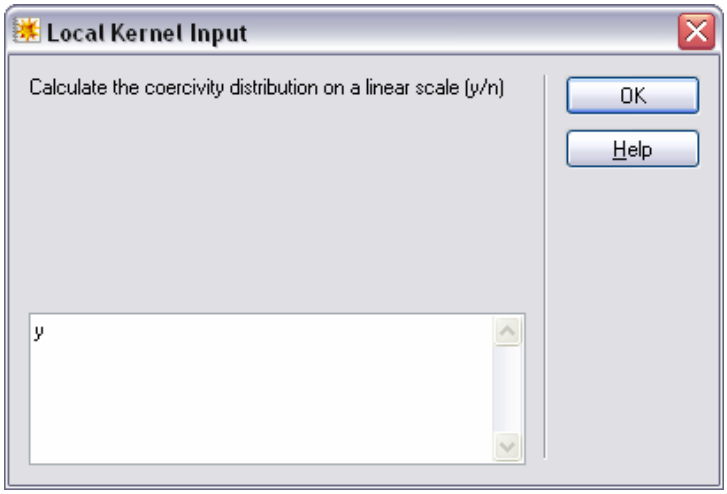
A prompt window will ask you if you want to choose a different range of fields. Type “n” if you are satisfieed with the result. If you type “y”, CODICA will ask you to enter a new range of fields. Then, it will recalculate and the coercivity distribution on the new range. This process is repeated until you type “n” in the prompt window to the right.



12. Coercivity distribution on a linear scale

It is sometimes useful to plot the coercivity distribution on a linear scale. A linear scale will show different features of the coercivity distribution that may be hidden in the logarithmic scale representation (see Fig. 10).

A prompt window will ask you if you want to calculate the coercivity distribution on a linear scale. Type “y” if you are interested in this option.



If you accepted to calculate the coercivity distribution on a linear scale, CODICA will first display the coercivity distribution on this scale over the entire range of fields spanned by the measurements.

The behavior of some coercivity distributions at $H \to 0$ might be discontinuous. This is for example the case of an exponential distribution. In such cases, the coercivity distribution calculated by CODICA might show a large peak at $H = 0$. This peak is an artifact produced by a field scale change. You can remove this peak by choosing to calculate the coercivity distribution on a range of fields that does not include $H = 0$. This problem is avoided if the magnetization curve is defined by enough points at $H \to 0$.

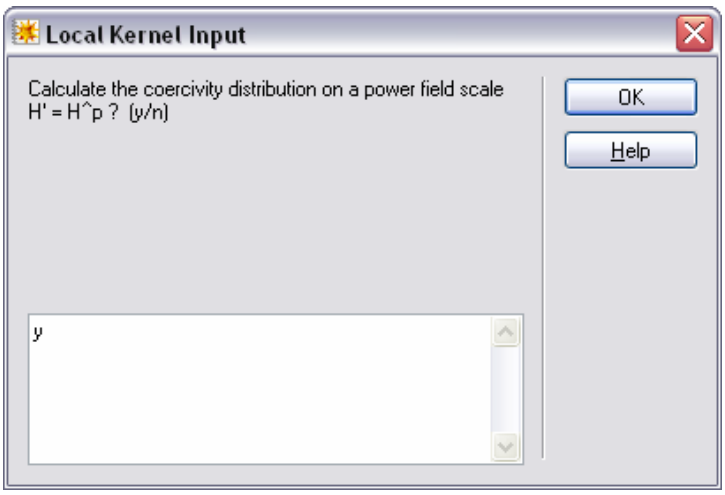
At this points, you have the same options as in 11 for choosing the range of fields and the number of points of the coercivity distribution, and the coercivity distribution is plotted on the desired range.

13. Coercivity distribution on a power scale

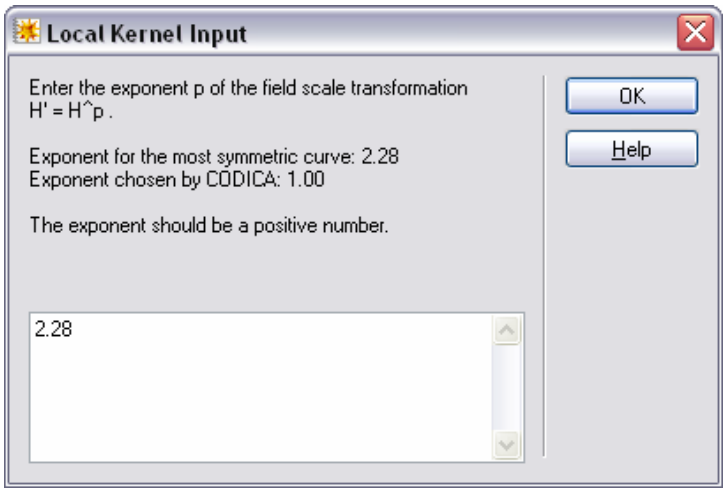
It is sometimes useful to plot the coercivity distribution on a field scale that is intermediate between a logarithmic and a linear scale. The scale transformation defined by $H^* = H^p$, hereon called the power scale transformation, is a suitable scale for this purpose. The exponent p is a positive number. Special cases are $p \to 0$, whose limit is the logarithmic scale, and $p = 1$, which is the linear scale.

A power scale will show different features of the coercivity distribution that may be hidden in the logarithmic and the linear scale representation (see Fig. 10). It is recommended to use a power scale to check if a coercivity distribution is unimodal.

A prompt window will ask you if you want to calculate the coercivity distribution on a power scale. Type "y" if you are interested in this option.



Another prompt window will ask you to enter the exponent p of the power scale transformation. You can enter any positive number. Numbers close to zero will give results that are similar to those on a logarithmic scale.



The behavior of some coercivity distributions at $H \to 0$ might be discontinuous for power scales with $p \geq 1$. This is for example the case of an exponential distribution. In such cases, the coercivity distribution calculated by CODICA might show a large peak at $H = 0$, especially if $p > 1$. This peak is an artifact of changing the field scale used by CODICA into a power scale. You can remove this peak by choosing to calculate the coercivity distribution on a range of fields that does not include $H = 0$. The use of $p > 1.5$ is not recommended.

After you entered a value for p, CODICA will first display the coercivity distribution on this scale over the entire range of fields spanned by the measurements.

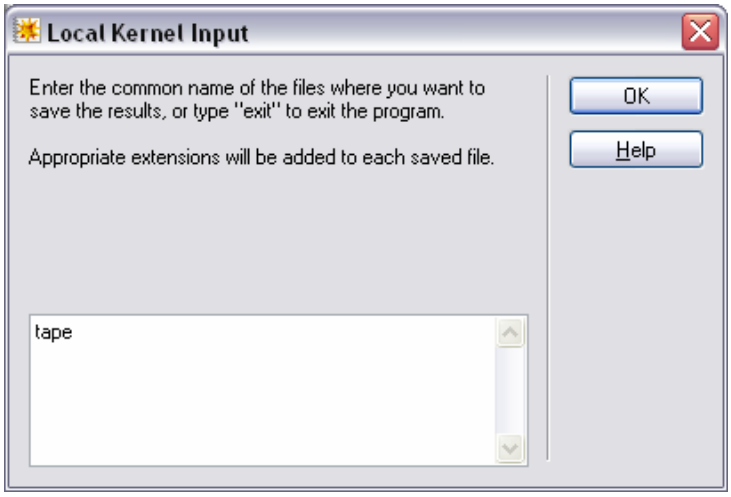
At this points, you have the same options as in 📖 11 for choosing the range of fields and the number of points of the coercivity distribution to be plotted on a power scale.

➔ You have the possibility to try different values of p. Answer "n" to the prompt window asking you to choose a different range of fields. Thereafter, CODICA re-asks you if you want to plot the coercivity distribution on a power scale. Anwer "y", and enter a new power exponent in the next prompt window.

📖 14. Saving the results

After plotting the coercivity distribution on al field scales and ranges you wanted, CODICA is ready to save the results.

A prompt window will ask you to enter a name for the files where you want to save the data. This name will be used to write different files which are distinguished by their extension. If you do not want to save the data, type "exit" to exit CODICA. All files produced by CODICA are saved in the same directory where the original data file is located.



CODICA will save the filtered magnetization curve in a file with extension "`.cum`" (for cumulative distribution, which is the integral of the coercivity distribution). The coercivity distribution on a logarithmic scale is saved in a file with extension "`.slog`" ("s" for spectra, "log" for logarithm).

If you chose to calculate the coercivity distribution on a linear scale or a power scale, files with extensions "`.slin`" ("s" for spectra, "lin" for linear), and/or "`.spow`" ("s" for spectra, "pow" for power) will be created as well.

➔ CODICA saves always the *last* coercivity distribution calculated on a particular scale. If you want to save the results of a CODICA session, be sure that the *last* coercivity distribution you asked CODICA to calculate on a given scale is the distribution you want to save.

➔ You can save the coercivity distributions plots produced by CODICA by clicking on the corresponding graphics in the *Mathematica* notebook. The selected plot will be surrounded by a selection rectangle. On the top menu bar of *Mathematica*, select: `Edit → Save Selection As → Format`, whereby `Format` is a graphics format. Full-quality images are obtained using the EPS format.

A program example

Notice: Numbers to the left refer to sections in the text with the corresponding explanation. Click on the link text on the left to return to the corresponding section of the manual.

text
```
<<Utilities`Codica`
```

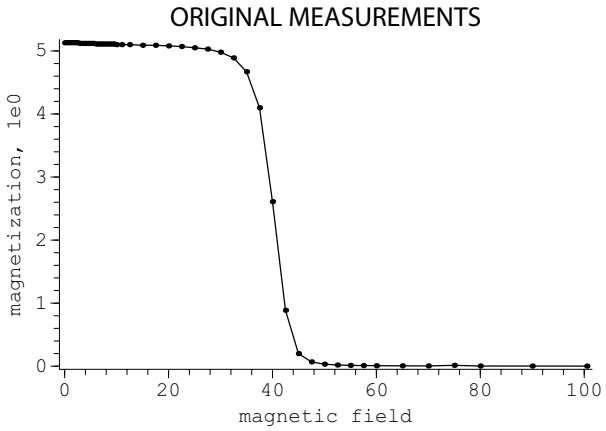
```
CODICA v.5.0 for Mathematica 5.0 and later versions.
Distributed with the package MAG-MIX release 1, 04/04/2005.
Copyright 2005 by Ramon Egli. All rights reserved.
```

text
```
Codica
```

1
```
C:/MAG-MIX/Codica/Examples/tape.dat
```

2
```
Check the measurements...
```

2



3
```
Evaluate the properties of the measured curve...
```

3
```
WARNING! CODICA detected possible digital truncation effects.
List of points where jumps arising from digital truncation have been identified:

point #     field     detected jump
6           2.5       0.01
12          5.5       0.01
20          9.5       0.01
23          12.5      0.01

Some points have been discarded/resampled in order to minimize digital
truncation effects.
```

3
```
Estimated initial magnetization: 5.126 ± 0.010
Estimated final magnetization:   0.004773 ± 0.0033 (extrapolated)
Estimated median field:          40.03
```

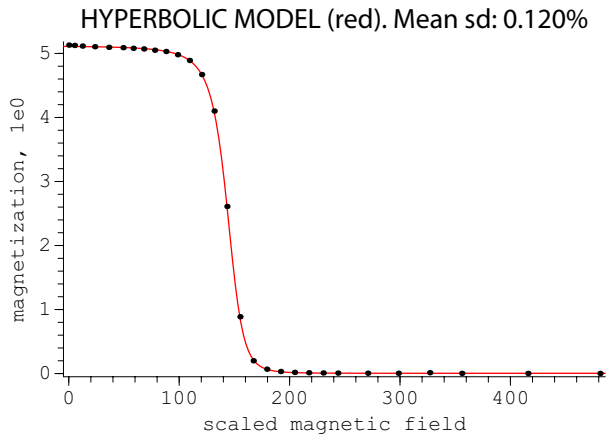
4
```
Searching for an optimized field scale.

Scan scaling exponents < 1. Please wait...

Scan scaling exponents > 1. Please wait...
```

4
```
Field scale                    Transformation       Mean sd of fit
original:                      H' = H               1.01%
for equally spaced points:     H' = H^0.849         1.04%
for the most symmetric curve:  H' = (H+1.37)^2.28   0.547%
suggested scale:               H' = (H+1.37)^1.34   0.899%
```

📖 4 Searching for an asymmetric best-fit hyperbolic model. Please wait...

Estimated contribution of a high-coercivity component: 2.58e-13 %

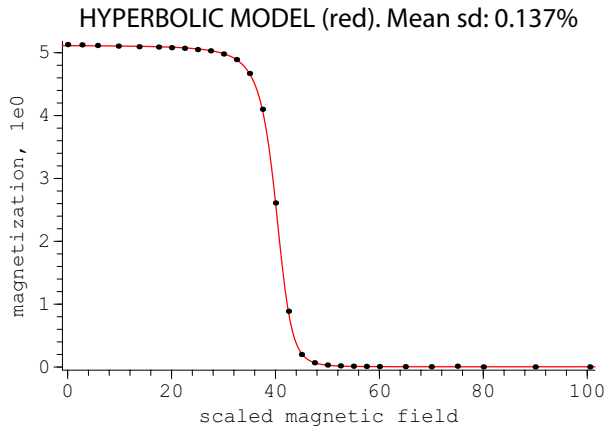
📖 4



📖 5 Manually chosen field scaling exponent: 1.00

Searching for an asymmetric best-fit hyperbolic model. Please wait...

Estimated contribution of a high-coercivity component: 2.58e-13 %



📖 6 Search a scale for the magnetization. Please wait...

Iteration #1...

Iteration #2...

Iteration #3...

Estimate the measurement errors. Please wait...

Calculate the residuals.

Subtract the residuals trend with a 2nd order polynomial.

Estimate the autocorrelation function of the residuals. Please wait...

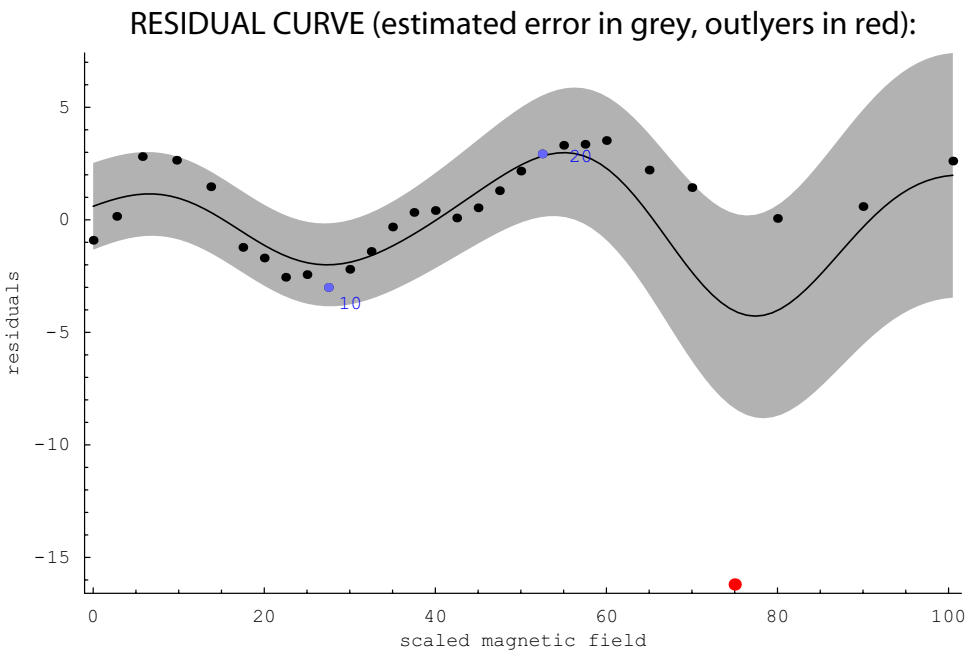
WARNING! The residuals are almost random.

📖 6 Estimated lower limit of the autocorrelation distance: 17.6

📖 6 Fit the residuals with an autocorrelation model.

WARNING! CODICA detected some outlyers.

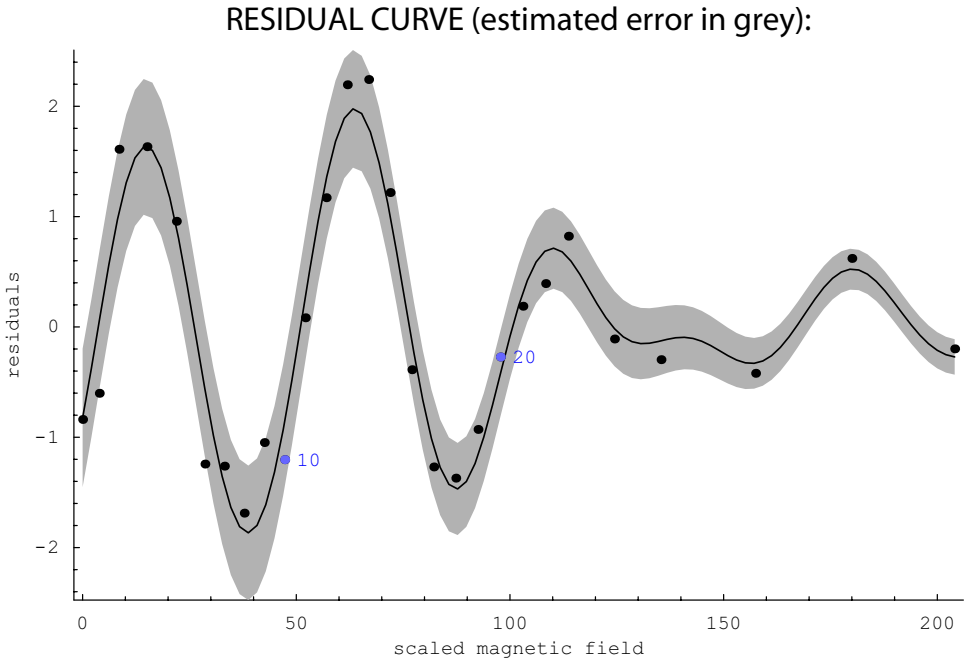
📖 6



📖 7 `Outlyers will be removed and the cleaned measurements will be analyzed again.`

📖 7 *[The repeated steps are not shown here…]*

📖 8

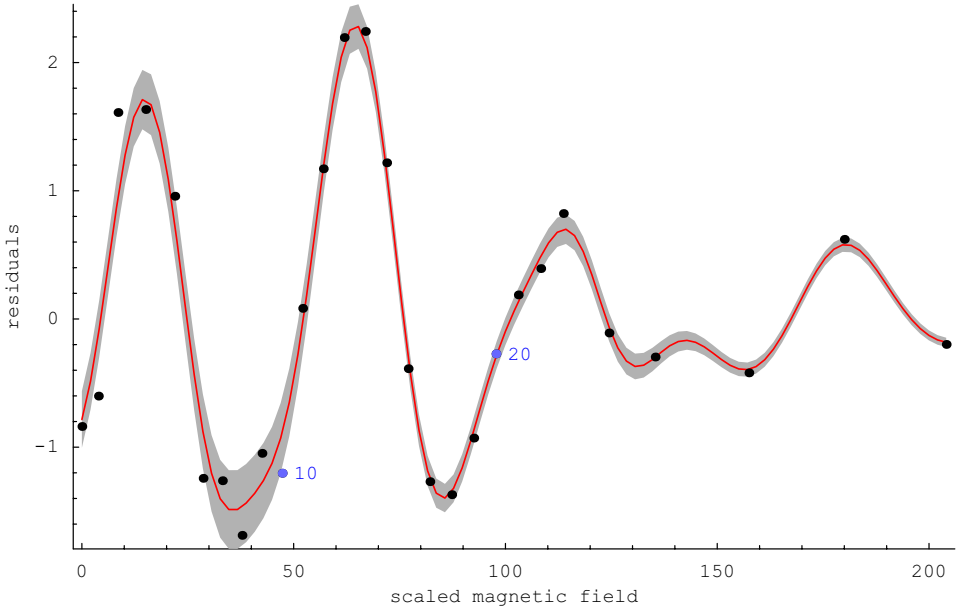


📖 8 `Filter the residual curve using least-squares collocation with correlation length: 12.7`

`Estimate the measurement errors...`

📖 8

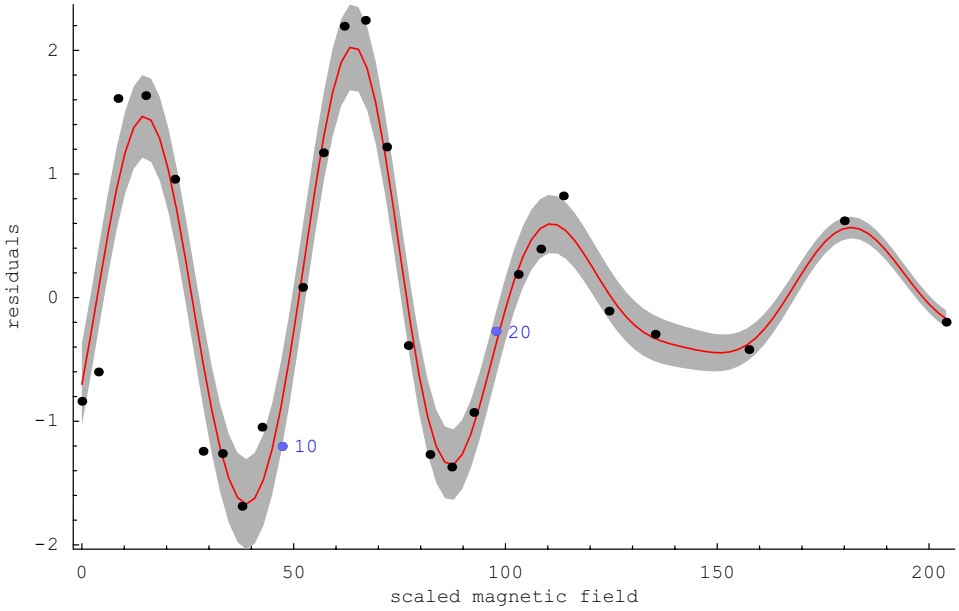
FILTERED RESIDUAL CURVE with cl = 12.7 (estimated error in grey):



📖 8

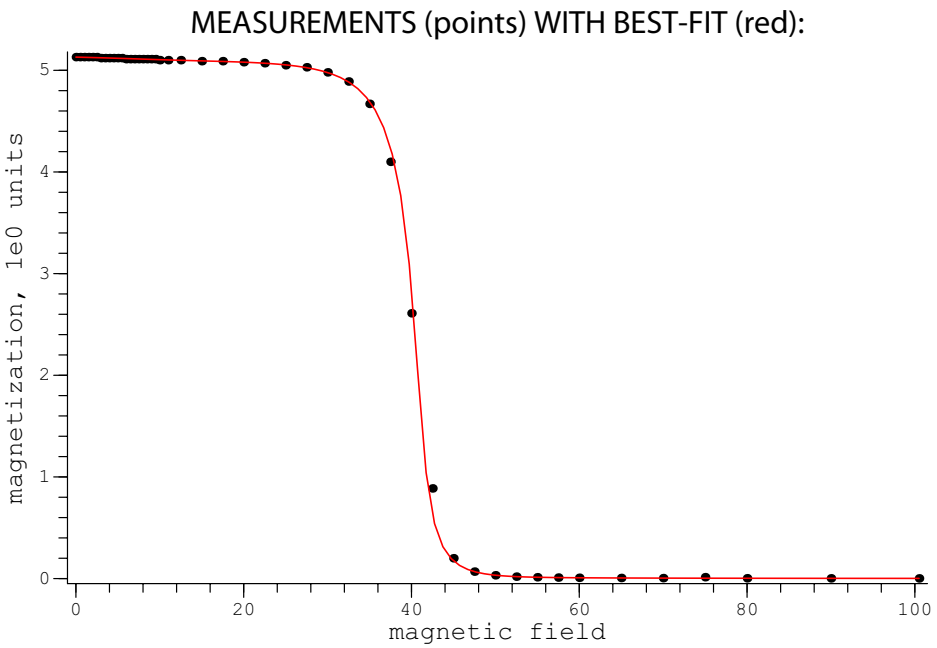
```
Filter the residual curve using least-squares collocation with
correlation length: 20.0

Estimate the measurement errors...
```

FILTERED RESIDUAL CURVE with cl = 20.0 (estimated error in grey):

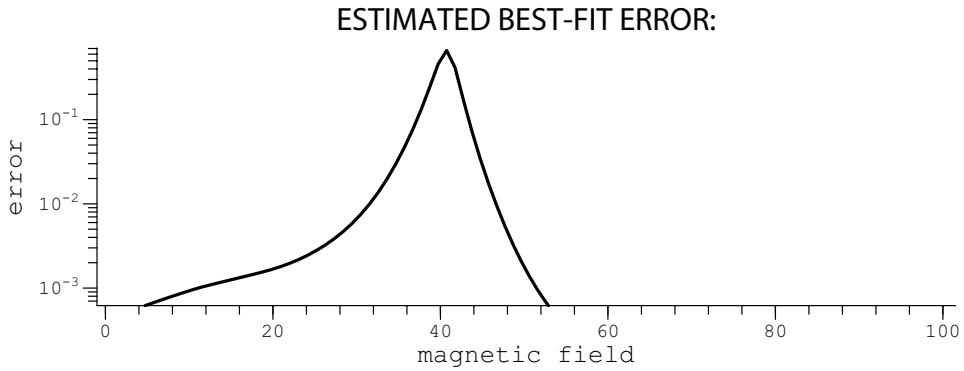


📖 9 Calculate the fitted demagnetization curve...

📖 9

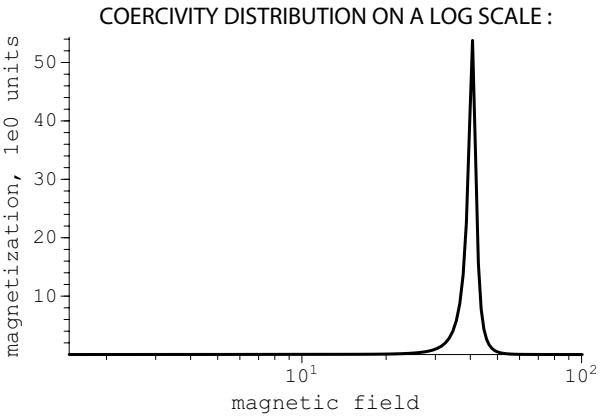


📖 9



📖 10 Calculate the coercivity distribution. Please wait...

📖 11



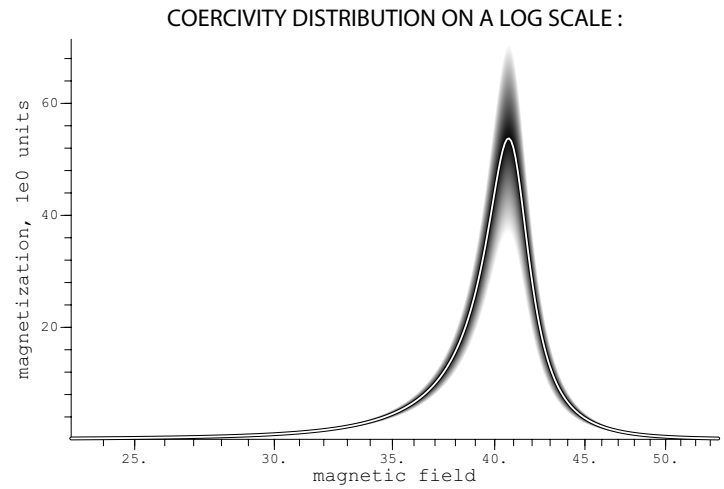
Range of fields spanned by the coercivity distribution: 23.0 - 53.6

The coercivity distribution is significant over the entire range of fields.

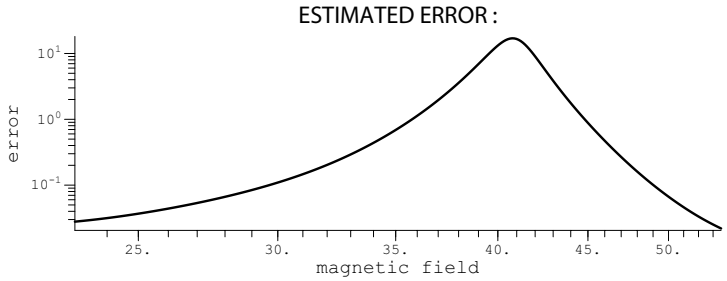
📖 11 Resample the coercivity distribution in the field range 23.0 -53.6 ...

The grey band around the curve in the following plot corresponds to the estimated error (2 sd)

📖 11

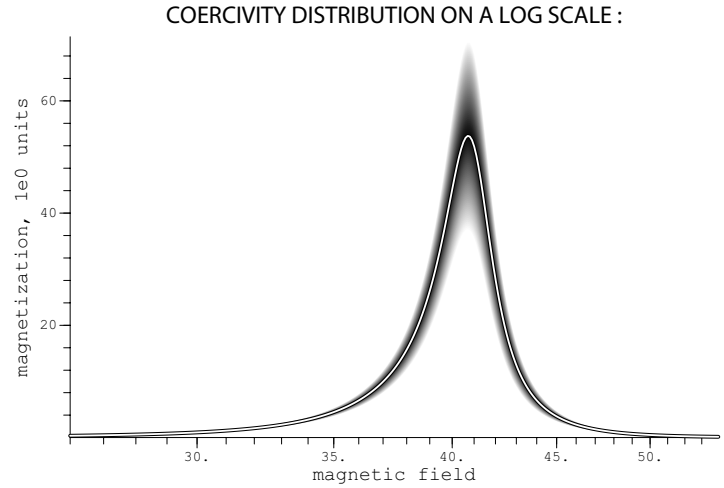


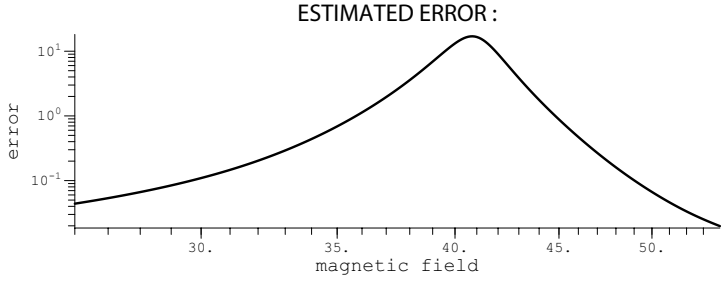
📖 11

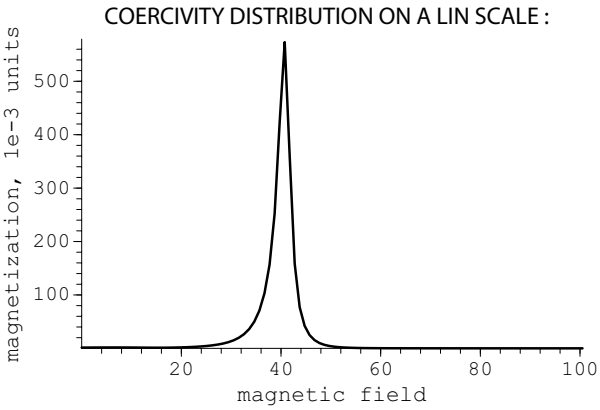


📖 11 Resample the coercivity distribution in the field range 26.0 - 54.0 ...

The grey band around the curve in the following plot corresponds to the estimated error (2 sd)



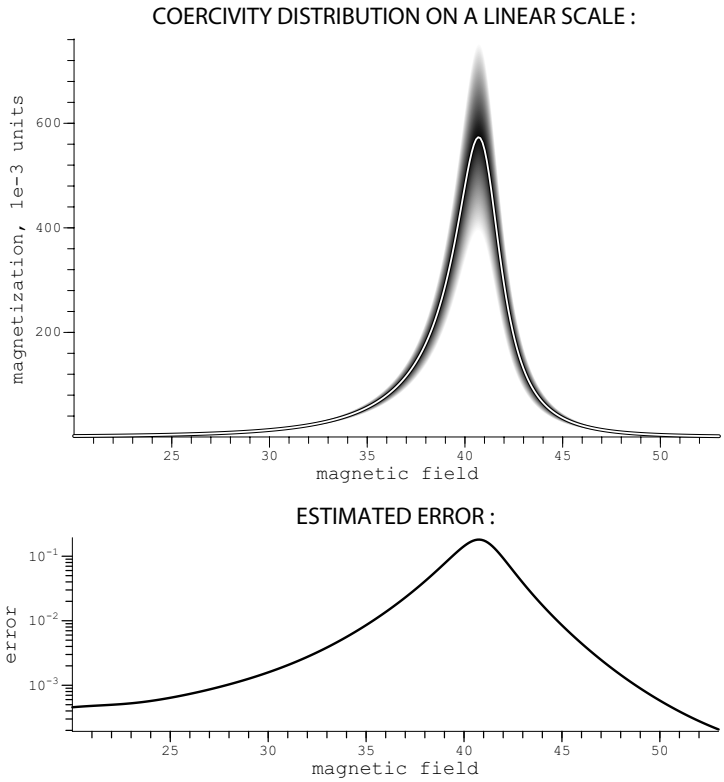


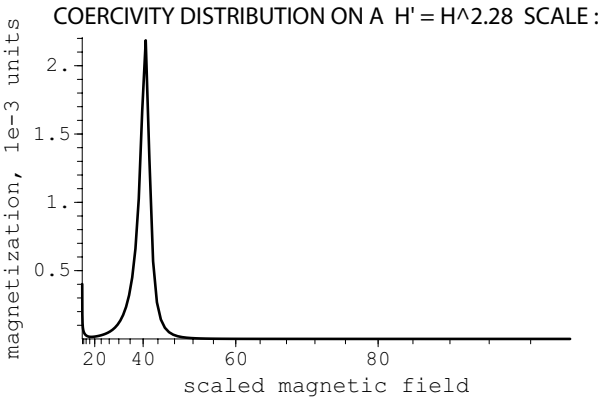


Range of fields spanned by the coercivity distribution: 0 - 52.6

Resample the coercivity distribution in the field range 20.0 -53.0 ...

The grey band around the curve in the following plot corresponds to the estimated error (2 sd)

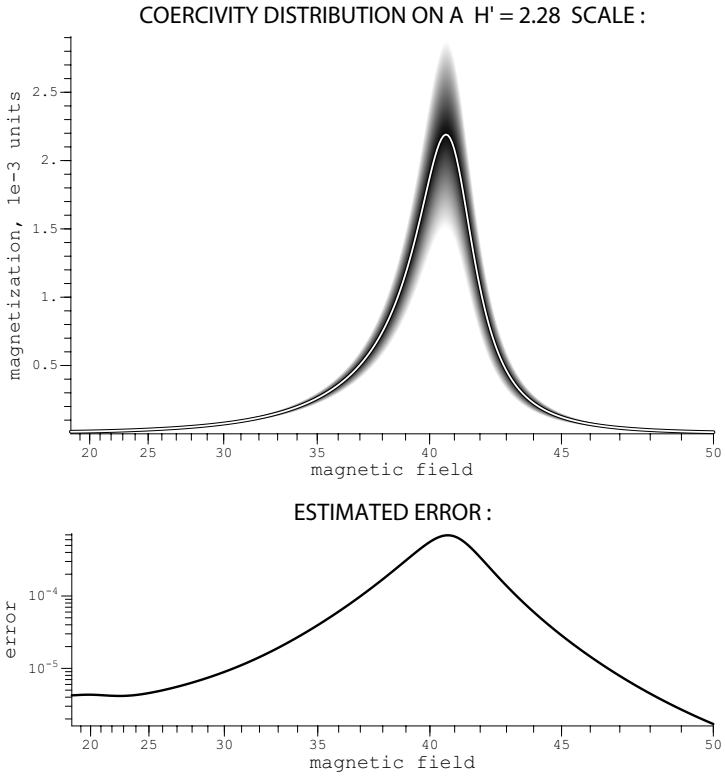




Range of fields spanned by the the coercivity distribution: 1.47 - 51.6

Resample the coercivity distribution in the field range 18.0 -50.0 ...

The grey band around the curve in the following plot corresponds to the estimated error (2 sd)



📖 14

```
Save the fitted measurements in file: tape.cum
 1. column: magnetic field,
 2. column: magnetization,
 3. column: absolute fit error.

Save the log-scaled coercivity distribution in file: tape.slog
 1. column: log(magnetic field),
 2. column: coercivity distribution (unit: magnetization),
 3. column: relative error.

Save the linear scaled coercivity distribution in file: tape.slin
 1. column: magnetic field,
 2. column: coercivity distribution (unit: magnetization/field),
 3. column: relative error.

Save the linear scaled coercivity distribution in file: tape.spow
 1. column: (magnetic field)^2.28,
 2. column: coercivity distribution (unit: magnetization/field^2.28),
 3. column: relative error.

Thank you for using CODICA. Are you satisfied?
Please report eventual problems or suggestions to the author!
```

6. Technical reference

6.1. Weighted fit

Since the measurement points are not necessarily equally spaced, an unweighted fit would be biased by the parts of the magnetization curve where the points are closely spaced. To avoid this bias, all data fits performed by CODICA are weighted to avoid this effect. In a least-squares fit, the function:

$$\varepsilon^2 = \sum_k w_k \left[y_k - f(x_k, a_j) \right]^2 \tag{1}$$

is minimized with respect to the parameters a_j, whereby (x_k, y_k) are the points to be fitted, $k = 1 \ldots N$, $f(x, a_j)$ is the parameterized fitting function and:

$$\begin{aligned} w_k &= \frac{x_{k+1} - x_{k-1}}{2(x_N - x_1)}, \quad 2 \le k \le N-1 \\ w_1 &= \frac{x_2 - x_1}{x_N - x_1}, \quad w_N = \frac{x_N - x_{N-1}}{x_N - x_1} \end{aligned} \tag{2}$$

are the fitting weights.

6.2. Scaling the magnetic field

The magnetic field is scaled according to the transformation:

$$H_k^* = \mathscr{P}(H_k) = (H_k + q)^p - (H_1 + q)^p \tag{3}$$

where $p > 0$ is the scaling exponent, and

$$q = \begin{cases} H_2 / 5, & \text{if } H_1 = 0 \\ \min[H_1, H_2 - H_1] / 5, & \text{if } H_1 > 0 \end{cases} \tag{4}$$

is a damping coefficient that avoids divergence at $x = 0$. The inverse transformation is given by:

$$H_k = \mathscr{P}^{-1}(H_k^*) = [H_k^* + (H_1 + q)^p]^{1/p} - q \tag{5}$$

6.3. Model function for the magnetization curve

The magnetization curve $M(H)$ is modeled with the sum of two empirical functions $S(H)$ and $U(H)$. The function

$$S(H) = a_3 \left\{ \frac{1^+}{2} - \frac{1}{2} \tanh \left[a_5 \operatorname{arsinh} \left(\frac{a_1}{a_5} (H - a_2) \right) \right] \right\}^{a_6} + a_4 \tag{6}$$

accounts for the sigmoidal shape of $M(H)$. The symbol 1^+ means that CODICA uses the machine number > 1 that is closest to 1, to avoid numerical instability problems. The parameters a_3 and a_4 control the two horizontal asymptotes:

$$\lim_{H\to\pm\infty} S(H) = a_3(1 \mp \operatorname{sgn} a_1)/2 + a_4 \tag{7}$$

The position of the median field is controlled by the parameter a_2, and the "width" of $S(H)$ by a_1. The sign of a_1 decides whether $S(H)$ is a monotonically increasing or decreasing curve. The parameter $a_5 > 0$ influences the "tails" of $S(H)$: if $a_5 \to 0$ the horizontal asymptotes are approached very slowly, on the other hand, if $a_5 \to \infty$ the asymptotes are approached like a $\tanh$ function. The parameter $a_6 > 0$ controls the asymmetry of $S(H)$: $S(H) = S_0(H)$ is symmetric about $H = a_2$ if $a_6 = 1$. A left-skewed $S(H)$ is obtained with $0 < a_6 < 1$, and a right-skewed $S(H)$ is characterized by $a_6 > 1$. The effect of the different parameters on $S(H)$ is illustrated in Fig. 11.

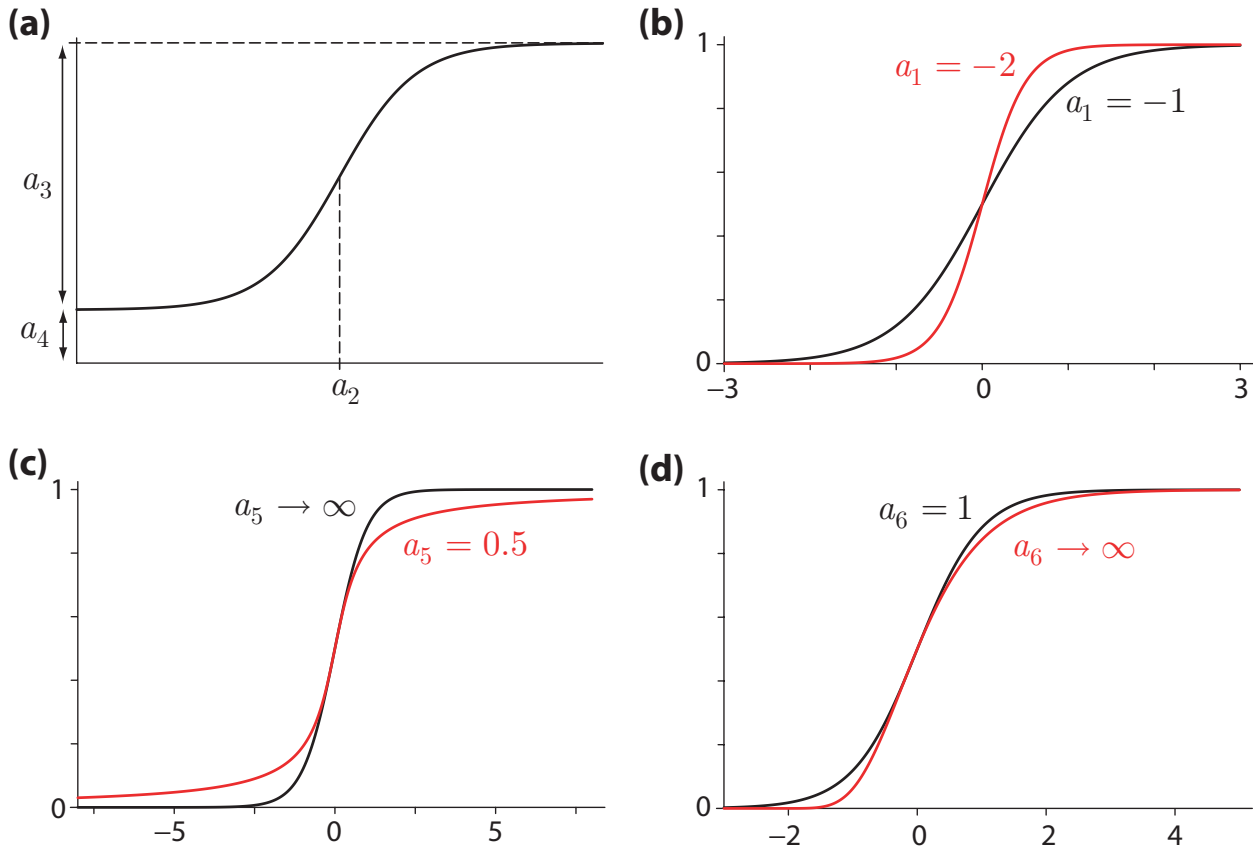


Fig. 11. Examples of the function $S(x, a_1, \ldots, a_6)$. **(a)** The parameters a_2, a_3 and a_4 control the center, the amplitude and the left asymptote of S, respectively. **(b)** The slope of the central part depends on a_1. **(c)** The way how the asymptotes are approached is controlled by a_5. **(d)** The parameter a_6 controls the symmetry of S. The symmetric case is given by $a_6 = 1$; $a_6 > 1$ gives a right-skewed S, while a left-skewed S is obtained with $0 < a_6 < 1$.

The function

$$U(H) = a_7 \operatorname{sgn} a_1 \left\{ (H - a_2) + \frac{1}{a_8} \ln\left[\cosh\left(a_8(H - a_2)\right)\right] + \ln 2 / a_8 \right\} \tag{8}$$

accounts for an eventual highly unsaturated high-coercivity component. $U(H)$ has the following two asymptotes:

$$\lim_{H\to-\infty} U(H) = 0$$
$$\lim_{H\to+\infty} U(H) = 2a_7 \operatorname{sgn} a_1 (H - a_2) \tag{9}$$

The parameter a_2 is the field at which the two asymptotes of $U(H)$ intersect, a_7 is proportional to the slope of $U(H \gg a_2)$, and a_8 controls the interval over which $U(H)$ changes from a constant value of 0 to a line with slope $2a_7 \operatorname{sgn} a_1$. The effect of the different parameters on $U(H)$ is illustrated in Fig. 12.

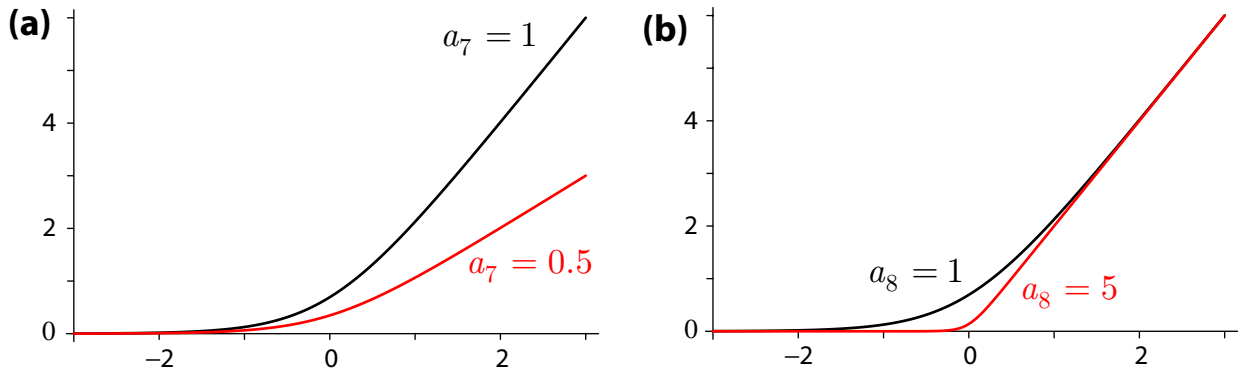


Fig. 12. Examples of the function $U(x, a_1, a_2, a_7, a_8)$. **(a)** The parameter a_7 controls the slope of the right asymptote. **(b)** The interval within which U changes its slope is controlled by a_8.

6.4 Searching the best scale for symmetry

The symmetry of a magnetization curve $M(H)$ is evaluated by fitting $M(H)$ with the sum of the symmetric sigmoid $S_0(H)$ and $U(H)$. The scale H^* for which:

$$\sum_k w_k^{1/2} \left[M(H_k^*) - S_0(H_k^*) - U(H_k^*) \right] = 0 \tag{10}$$

with the parameters $a_1 \ldots a_8$ chosen to obtain

$$\sum_k w_k \left[M(H_k^*) - S_0(H_k^*) - U(H_k^*) \right]^2 = \min! \tag{11}$$

is defined as the most symmetric scale of $M(H)$. CODICA evaluates (10) systematically for all power scales defined in 6.2 with $0.01 < p < 2.72$. If CODICA does not find a solution of (10) within this range of p, it just minimizes (11) with respect to $a_1 \ldots a_8$ and to H^*.

6.5. Model the magnetization curve

After an appropriate scale H^* has been chosen, CODICA models the magnetization curve with the two functions $S(H)$ and $U(H)$ by minimizing

$$\sum_k w_k \left[M(H_k^*) - S(H_k^*) - U(H_k^*) \right]^2 \tag{12}$$

with respect to the parameters $a_1 \dots a_8$. The eventual contribution of a strongly unsaturated high-coercivity component is subtracted from the magnetization curve to obtain the "nearly saturated" curve:

$$M^*(H) = M(H) - U(H) \tag{13}$$

6.6. Calculate the residuals

The residuals are obtained by rescaling the magnetization axis. In the ideal case, the magnetization curve $M^*(H^*)$ is identical to the model function $S(H^*)$. If the magnetization is rescaled using the inverse S^{-1} of S, the rescaled curve is given by $S^{-1}(M^*(H^*)) \approx S^{-1}(S(H^*)) = H^*$, which is the identity function. If the model function $S(H^*)$ is close enough to $M^*(H^*)$, the rescaled magnetization curve $S^{-1}(M^*(H^*))$ is close to a straight line. The residuals curve $R(H^*)$ is thus defined as:

$$R(H^*) = S^{-1}(M^*(H^*)) - H^* \tag{14}$$

with:

$$S^{-1}(M) = \frac{a_5}{a_1} \sinh\left[\frac{1}{a_5} \operatorname{artanh}\left(1^+ - 2\left(\frac{M - a_4}{a_3}\right)^{1/a_6}\right)\right] + a_2 \tag{15}$$

CODICA calculates the smallest possible residuals by minimizing

$$\sum_k w_k R^2(H_k^*) \tag{16}$$

with respect to the parameters $a_1 \dots a_6$.

6.7. A stochastic model of the residuals

The residuals are the superposition of (1) a curve that represents the deviation of the model function $S(H^*)$ from the measured curve, and (2) a random signal $e(H^*)$ that depends on the measurement errors. The properties of the residual curve are evaluated by calculating the autocovariance function:

$$C_R(h) = \int_{-\infty}^{\infty} R(H^*) R(H^* + h)\, \mathrm{d}H^* \tag{17}$$

Since $R(H^*)$ is a collection of discrete, unevenly spaced points, $C_R(h)$ is calculate indirectly through the Fourier transform R^{F} of R:

$$R^{\mathsf{F}}(\omega) = \int_{-\infty}^{\infty} R(H^*) e^{i\omega H^*}\, \mathrm{d}H^* \approx (H_N^* - H_1^*) \sum_k w_k R(H_k^*) e^{i\omega H_k^*} \tag{18}$$

whereby:

$$C_R(h) = \int_{-\infty}^{\infty} |R^{\mathsf{F}}(\omega)|^2 \cos(\omega h)\, \mathrm{d}\omega \tag{19}$$

A first, simple estimate of $e(H^*)$ is obtained by assuming that $e(H^*)$ is an uncorrelated random signal. In this case, $C_e(h) = \mathrm{var}[e]\delta(h)$, where $\mathrm{var}[e]$ is the variance of $e(H^*)$ and $\delta(h)$ is the Dirac δ-function. Under this assumption the variance of $e(H^*)$ is given by:

$$\mathrm{var}[e] = C_R(0) - C_R(\Delta h) \tag{20}$$

whereby Δh is chosen to be the mean distance between the points in $R(H^*)$. The first estimate of $e(H^*)$ is then $e^2(H^*) = \mathrm{var}[e]$.

6.8. Subtract a trend with polynomials

The residual curve contains generally a trend that can be removed by subtracting the $(n+1)$-th order polynomial $P_{n+1}(H^*)$ which minimizes:

$$\sum_k w_k \left[R(H_k^*) - P_{n+1}(H_k^*)\right]^2 \tag{21}$$

CODICA chooses n to be the number of significant local minima and maxima $\hat{H}_1^*, \ldots, \hat{H}_n^*$ of the residual curve. A local minimum or maximum $\hat{H}_k^*$ is considered significant if and only if:

$$\left| R(\hat{H}_k^*) - R(\hat{H}_{k\pm1}^*) \right| > \max_{k-1\le j\le k+1} \left[e(\hat{H}_j^*)\right] \tag{22}$$

This means that the difference between two successive significant local extrema must be larger than the estimated error (Fig. 13).

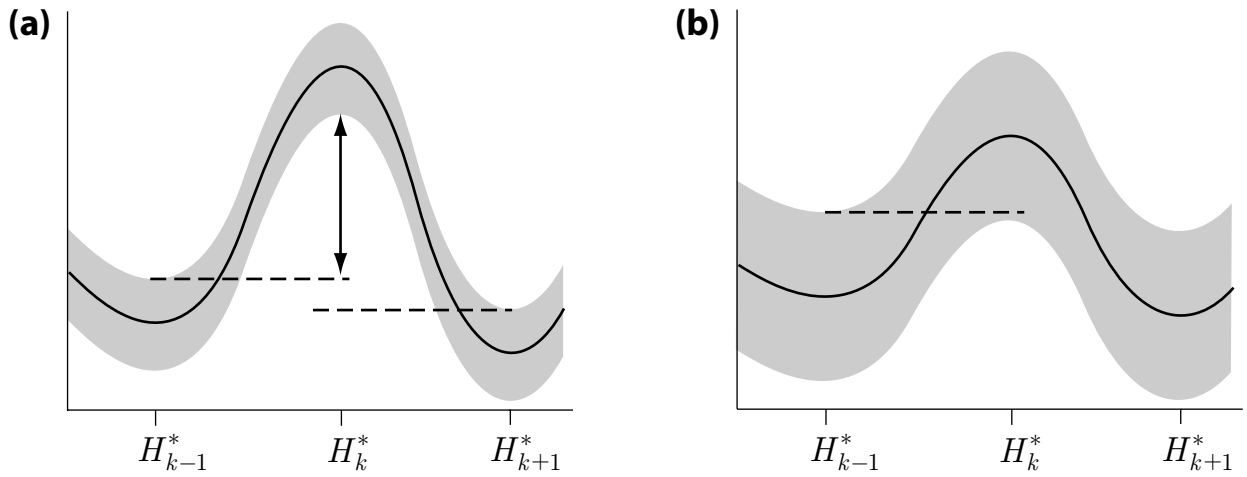


Fig. 13. Example of a residual curve (solid line) with three local maxima/minima. The error is represented by the grey band around the residual curve. In **(a)** the central local maximum is significant, because the difference with the neighbor minima is larger than the error. The opposite is true in **(b)**.

The "trend free" residuals:

$$r(H^*) = R(H^*) - P_{n+1}(H^*) \tag{23}$$

are the superposition of (1) a more or less sinusoidal curve $d(H^*)$ that represents the deviation of the model function $S(H)$ from the measured curve, and (2) a random signal $e(H^*)$ that depends on the measurement errors:

$$r(H^*) = d(H^*) + e(H^*) \tag{24}$$

6.9. Rescale the residuals

The "noise free" residual curve $d(H^*)$ is supposedly more or less sinusoidal. Accordingly, its Fourier spectrum $d^{\mathsf{F}}(\omega)$ peaks at a dominant frequency ω_0, whereby $2\pi / \omega_0$ is the typical "wavelength" of the wiggles of $d(H^*)$. CODICA applies an additional field scale transformation:

$$\begin{aligned} H_k^{**} &= \mathscr{P}^*(H_k^*) = (H_k^* + \overline{q})^{\overline{p}} - (H_1^* + \overline{q})^{\overline{p}} \\ \overline{q} &= H_2^* / 5 \end{aligned} \tag{25}$$

by choosing an exponent $\overline{p}$ that maximizes the peak of the Fourier transform $r^{\mathsf{F}}(\omega)$ of $r(H^{**})$. The new scale makes $d(H^{**})$ more similar to a sinusoidal function as possible (Fig. 14).

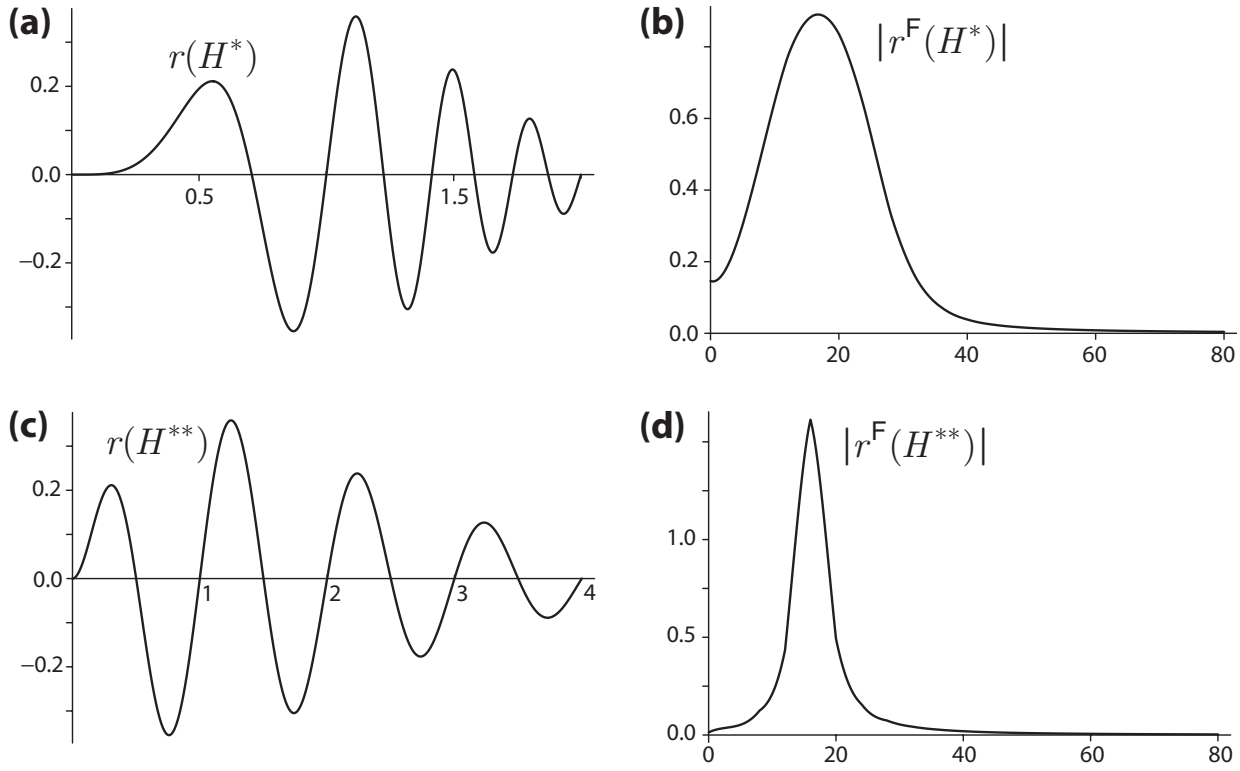


Fig. 14. Example of how CODICA rescales the residual curve to approach the sinusoidal curve. **(a)** Original residual curve $r(H^*)$ and **(b)** its power Fourier spectrum $|r^{\mathsf{F}}(H^*)|$. Notice how the "wavelength" of $r(H^*)$ changes with H^*. **(c)** A scale transformation $H^{**} = (H^*)^2$ makes the residual more similar to a sinusoidal curve. **(d)** The corresponding Fourier spectrum shows a higher and more localized peak at the dominant frequency.

6.10. A least-squares collocation model for the residuals

Least-squares collocation is used to interpolate a measured signal whose autocovariance function is known. Hence, the first step of least-squares collocation consists in modeling the autocovariance function of $r(H^{**}) = d(H^{**}) + e(H^{**})$. Since $d(H^{**})$ is almost sinusoidal, and $e(H^{**})$ is a stochastic signal, CODICA models the corresponding autocovariance functions with:

$$\begin{aligned} C_d(h) &= d_0^2 \exp[-(h/r_d)^2]\cos(\omega_d h) \\ C_e(h) &= e_0^2 \exp[-(h/r_e)^2] \end{aligned} \tag{26}$$

where $d_0^2 = \mathrm{var}[d]$ and $e_0^2 = \mathrm{var}[e]$ are the variances of d and e, respectively; r_d and r_e are the so-called correlation distances of d and e, respectively; and $2\pi/\omega_d$ is the dominant wavelength of d. Assuming that d and e are uncorrelated:

$$C_r(h) \approx C_d(h) + C_e(h) \tag{27}$$

On the other hand, according to (19):

$$C_r(h) = \int_{-\infty}^{\infty} |r^{\mathsf{F}}(\omega)|^2 \cos(\omega h)\,\mathrm{d}\omega \tag{28}$$

To find a best-fit correlation model of $C_r(h)$, CODICA minimizes the squared difference between (27) and (28):

$$\int_0^{H^{**}_{\max} - H^{**}_{\min}} [C_d(h) + C_e(h) - C_r(h)]^2\,\mathrm{d}h \tag{29}$$

with respect to the parameters d_0, e_0, r_d, r_e and ω_d. The *correlation distance* r_0 of the residuals is assumed to be the value of h for which $C_d(h)$ is reduced to the half: $C_d(r_0) = 0.5 C_d(0)$. If either $e_0 > 0.8 d_0$ or $r_d < r_e$, CODICA consider the residuals to be dominated by a noise signal produced by the measurement errors. In this case, the estimate of $C_d(h)$ using (26) is not reliable, and is replaced by:

$$C_d(h) = d_0^2 \exp[-(h/r_0)^2] \tag{29}$$

whereby $C_r(r_0) = 0.5 C_r(0)$.

If $r_0 \gg r_e$ is known, a more accurate estimate of $e(H^{**})$ by evaluating the variance of the difference δr between $r(H^{**})$ and the moving average

$$\overline{r}(H_k^{**}) = \frac{\sum_j r(H_j^{**}) \exp[-(H_j^{**} - H_k^{**})^2/r_0^2]}{\sum_j \exp[-(H_j^{**} - H_k^{**})^2/r_0^2]} \tag{30}$$

If the same moving average filter is applied to $\delta r^2 = (r - \overline{r})^2$ we obtain following estimate of $e(H^{**})$:

$$e_k^2 \approx \frac{\sum_j [r(H_j^{**}) - \overline{r}(H_j^{**})]^2 \exp[-(H_j^{**} - H_k^{**})^2/r_0^2]}{\sum_j \exp[-(H_j^{**} - H_k^{**})^2/r_0^2]} \tag{31}$$

An interpolation $\hat{d}(x_k)$ of the error-free residuals at the points $x_1,\ldots,x_N$ is performed using a least-squares collocation based on the covariance matrices $\mathbf{C}_{dd}$ of $d(H_k^{**})$, $\mathbf{C}_{\hat{d}d}$ of $\hat{d}(x_k)$ and $d(H_k^{**})$, and $\mathbf{C}_{ee}$ of $e(H_k^{**})$, defined as:

$$\begin{aligned}
[\mathbf{C}_{dd}]_{kj} &= C_d(H_k^{**} - H_j^{**}) \quad , k,j = 1\ldots n \\
[\mathbf{C}_{\hat{d}d}]_{kj} &= C_d(x_k - H_j^{**}) \quad , k = 1\ldots N \ , \ j = 1\ldots n \\
[\mathbf{C}_{ee}]_{kj} &= e_k^2 \delta_{kj} \quad , k,j = 1\ldots n
\end{aligned} \tag{32}$$

where $\delta_{kj} = 1$ if $j = k$ and $\delta_{kj} = 0$ else. The interpolated residuals $\hat{\mathbf{d}} = [\hat{d}(x_1)\ldots\hat{d}(x_n)]$ are given by:

$$\hat{\mathbf{d}} = \mathbf{C}_{\hat{d}d}(\mathbf{C}_{\hat{d}\hat{d}} + \mathbf{C}_{ee})^{-1}\mathbf{d} \tag{33}$$

with $\mathbf{d} = [d(H_1^{**})\ldots d(H_n^{**})]$. The estimated errors $\hat{\mathbf{e}}$ of $\hat{\mathbf{d}}$ are given by the diagonal elements of the error matrix:

$$\mathbf{E} = \mathbf{C}_{\hat{d}\hat{d}} - \mathbf{C}_{\hat{d}d}(\mathbf{C}_{dd} + \mathbf{C}_{ee})^{-1}\mathbf{C}_{\hat{d}d}^{\mathsf{T}} \tag{34}$$

6.11. Filtering the residuals

The autocorrelation model of the residuals calculated by CODICA is usually accurate. However, it is very difficult to find a model that applies to all possible situations, and the human brain has a superior capacity in distinguishing a regular pattern from noise. Therefore, the user has the opportunity to correct the model proposed by CODICA to make the interpolated residuals $\hat{d}(x_k)$ closer to the actual residuals or, on the opposite, to make $\hat{d}(x_k)$ smoother. To do so, the user is asked to enter his "estimate" r_0^{usr} of the correlation length r_0. If the user has the feeling that $\hat{d}(x_k)$ should be closer to the actual residuals, he enters $r_0^{\mathrm{usr}} < r_0$. If the user has the feeling that $\hat{d}(x_k)$ still contain some noise, he enters $r_0^{\mathrm{usr}} > r_0$. Let $\Delta r = r - \hat{d}$ be the difference between the actual residuals r and their interpolation $\hat{d}$. If $r_0^{\mathrm{usr}} < r_0$, CODICA uses a least-squares collocation to obtain an interpolation $\Delta\hat{r}$ of Δr:

$$\Delta\hat{\mathbf{r}} = \mathbf{C}_{\hat{d}d}(\mathbf{C}_{\hat{d}\hat{d}} + \mathbf{C}_{ee})^{-1}\Delta\mathbf{r} \tag{35}$$

where the covariance matrices $\mathbf{C}_{\hat{d}d}$ and $\mathbf{C}_{dd}$ are calculated using equations (30,32) with r_0^{usr} instead of r_0. The "corrected" interpolation $\overline{\mathbf{d}}$ is then given by:

$$\overline{\mathbf{d}} = \hat{\mathbf{d}} + \Delta\hat{\mathbf{r}} \tag{36}$$

On the other hand, if $r_0^{\mathrm{usr}} > r_0$, CODICA uses a least-squares collocation to obtain a smooth version $\overline{\mathbf{d}}$ of $\hat{\mathbf{d}}$:

$$\overline{\mathbf{d}} = \mathbf{C}_{\hat{d}d}(\mathbf{C}_{\hat{d}\hat{d}} + \mathbf{C}_{ee})^{-1}\hat{\mathbf{d}} + \Delta\hat{\mathbf{r}} \tag{37}$$

with $\Delta\hat{\mathbf{r}}$ as in equation (35). The covariance matrices $\mathbf{C}_{\hat{d}d}$ and $\mathbf{C}_{dd}$ are calculated using equation (30,32) with r_0^{usr} instead of r_0.

The first derivative $\overline{\mathbf{d}}'$ of $\overline{\mathbf{d}}$ is needed to calculate the coercivity distribution, and is obtained from equations (33,35-37) by replacing $\mathbf{C}_{\hat{d}d}$ with its derivative $\mathbf{C}'_{\hat{d}d}$, given by:

$$[\mathbf{C}'_{\hat{d}d}]_{kj} = \partial_{x_k} C_d(x_k - H_j^{**}) \quad , k = 1\ldots N \ , \ j = 1\ldots n \tag{38}$$

The error estimate of $\overline{\mathbf{d}}$ is given by equation (34) and that of $\overline{\mathbf{d}}'$ by the diagonal elements of the error matrix:

$$\mathbf{E}' = \mathbf{C}''_{\hat{d}\hat{d}} - \mathbf{C}'_{\hat{d}d}(\mathbf{C}_{dd} + \mathbf{C}_{ee})^{-1}\mathbf{C}'^{\mathsf{T}}_{\hat{d}d} \tag{39}$$

with

$$[\mathbf{C}''_{\hat{d}d}]_{kj} = \partial_{x_k x_k} C_d(x_k - x_j) \quad , k, j = 1\ldots N \tag{40}$$

6.12. Filtered magnetization curve

The filtered magnetization curve $\overline{M}(H)$ is calculated from $\overline{\mathbf{d}}$ by inverting the steps described in sections 6.2, 6.5, 6.6, 6.9. Hence:

$$\begin{aligned} &\overline{M}(H) = S[\overline{d}(H^{**}) + P_{n+1}(H^*) + H^*] + U(H) \\ &H^* = \mathscr{P}^{*-1}(H^{**}) \ \ , H = \mathscr{P}^{-1}(H^*) \end{aligned} \tag{41}$$

The estimated error $\delta\overline{M}(H)$ of $\overline{M}(H)$ is:

$$\delta\overline{M}(H) = S'[\overline{d}(x) + P_{n+1}(H^*) + H^*]\delta\overline{d}(x) \tag{42}$$

where $\delta\overline{d}(x)$ is the estimated error of $\overline{d}(x)$.

6.13. Coercivity distributions

The coercivity distribution $\mathscr{M}_H(H)$ on a linear field scale is the derivative of equation (41) with respect to H:

$$\begin{aligned} \mathscr{M}_H(H) = {} & \mathscr{P}'(H)S'[\overline{d}(x) + P_{n+1}(H^*) + H^*][\overline{d}'(x)\mathscr{P}^{*\prime}(H^*) + P'_{n+1}(H^*) + 1] + \\ & \mathscr{P}'(H)U'(H) \end{aligned} \tag{43}$$

The estimated error $\delta\mathscr{M}_H(H)$ of $\mathscr{M}_H(H)$ is:

$$\delta\mathscr{M}_H(H) = \mathscr{P}'(H)S'[\overline{d}(x) + P_{n+1}(H^*) + H^*]\,\delta\overline{d}'(x)\mathscr{P}^{*\prime}(H^*) \tag{44}$$

The coercivity distribution $\mathscr{M}_{\log H}(H)$ and its error $\delta\mathscr{M}_{\log H}(H)$ on a $\log_{10}$ scale are given by:

$$\begin{aligned} &\mathscr{M}_{\log H}(\log H) = H\mathscr{M}_H(H)\ln 10 \\ &\delta\mathscr{M}_{\log H}(\log H) = H\delta\mathscr{M}_H(H)\ln 10 \end{aligned} \tag{45}$$

The coercivity distribution $\mathscr{M}_{H^p}(H)$ and its error $\delta\mathscr{M}_{H^p}(H)$ on a H^p-scale are given by:

$$\begin{aligned}\mathscr{M}_{H^p}(H^p) &= p^{-1}H^{1-p}\mathscr{M}_H(H)\\ \delta\mathscr{M}_{H^p}(H^p) &= p^{-1}H^{1-p}\delta\mathscr{M}_H(H)\end{aligned} \tag{46}$$

Appendix A

Tables of appropriate coercivity distribution units in the SI and the cgs system. $[M]$ is the magnetization unit (magnetic moment per unit volume or unit mass), $[H]$ is the magnetic field unit.

Magnetization, linear scale	Magnetization generic unit $[M]$	Magnetization SI unit: A/m	Magnetization SI unit: Am²/kg	Magnetization cgs unit: emu/g
Manetic field H generic unit $[H]$	$[M][H]^{-1}$			
Manetic field H SI unit: A/m		dimensionsless (eq. 1)	m^3/kg (eq. 1)	avoid this combination
Magnetic induction $\mu_0 H$ SI unit: T SI unit: mT		dimensionsless (eq. 1) × μ_0 (eq. 1) × $10^3 \mu_0$	m^3/kg (eq. 1) × μ_0 (eq. 1) × $10^3 \mu_0$	avoid this combination
Magnetic field H gcs unit: Oe		avoid this combination	avoid this combination	$\mathrm{emu\,Oe^{-1}g^{-1}}$

ARM susceptibility, linear scale	susceptibility generic unit $[M][H]^{-1}$	susceptibility SI unit: none	susceptibility SI unit: m³/kg	susceptibility cgs unit: emu/(Oe g)
Manetic field H generic unit $[H]$	$[M][H]^{-2}$			
Manetic field H SI unit: A/m		$\mathrm{A^{-1}m}$ (eq. 1)	$\mathrm{m^4kg^{-1}A^{-1}}$ (eq. 1)	avoid this combination
Magnetic induction $\mu_0 H$ SI unit: T SI unit: mT		$\mathrm{A^{-1}m}$ (eq. 1) × μ_0 (eq. 1) × $10^3 \mu_0$	$\mathrm{m^4kg^{-1}A^{-1}}$ (eq. 1) × μ_0 (eq. 1) × $10^3 \mu_0$	avoid this combination
Magnetic field H gcs unit: Oe		avoid this combination	avoid this combination	$\mathrm{emu\,Oe^{-2}g^{-1}}$

Magnetization, logarithmic scale	Magnetization generic unit $[M]$	Magnetization SI unit : A/m	Magnetization SI unit : Am²/kg	Magnetization cgs unit : emu/g
Manetic field H generic unit $[H]$	$[M]$			
Manetic field H SI unit: A/m		$\mathrm{A/m}$ (eq. 4)	$\mathrm{Am^2/kg}$ (eq. 4)	avoid this combination
Magnetic induction $\mu_0 H$ SI unit: T SI unit: mT		$\mathrm{A/m}$ (eq. 4) (eq. 4)	$\mathrm{Am^2/kg}$ (eq. 4) (eq. 4)	avoid this combination
Magnetic field H gcs unit: Oe		avoid this combination	avoid this combination	$\mathrm{emu\,g^{-1}}$

ARM susceptibility, logarithmic scale	susceptibility generic unit $[M][H]^{-1}$	susceptibility SI unit : none	susceptibility SI unit : m³/kg	susceptibility cgs unit : emu/(Oe g)
Manetic field H generic unit $[H]$	$[M][H]^{-1}$			
Manetic field H SI unit: A/m		dimensionsless (eq. 4)	$\mathrm{m^3kg^{-1}}$ (eq. 4)	avoid this combination
Magnetic induction $\mu_0 H$ SI unit: T SI unit: mT		dimensionsless (eq. 4) (eq. 4)	$\mathrm{m^3kg^{-1}}$ (eq. 4) (eq. 4)	avoid this combination
Magnetic field H gcs unit: Oe		avoid this combination	avoid this combination	$\mathrm{emu\,Oe^{-1}g^{-1}}$

Appendix B

Suggested empirical field correction table for AF demagnetization curves obtained with a 2G cryogenic magnetometer with build-in AF unit:

Range of AF fields (mT)	**Suggested correction factor** Δ $H_{\mathrm{corr}} = H + \Delta$
0 – 9.9	+0
10 – 29.9	+0.254
30 – 39.9	+0.465
40 – 49.9	–0.104
50 – 69.9	+0.104
70 – 79.9	+0.25
80 – 89.9	+0.10
90 – 109.9	+0.65
110 – 139.9	+0.95
140 – 149.9	+0.35
150 – 179.9	+0.55
180 – 300	+0.55

GECA

GEneralized **C**oercivity **A**nalyzer

Version 2.1

User's manual

Improtant notice

This version (2.1) of GECA (GEneralized Coercivity Analyzer) is almost identical with the previous version 2.0 that is described in this manual. Since GECA is going to be deeply revised in the near future, this manual has not been updated. Please consider the following differences of GECA 2.1 with respect to the manual:

- Since some users reported difficulties in using the characters "{" and "}" in the prompt windows, their need has been eliminated in GECA 2.1. Please ignore these characters in all examples provided in the manual. For example, if the manual tells you to enter {0,2}, type 0,2 instead. Accordingly, {} is replaced by no character (just click the "OK" button of the prompt window without typing anything).
- You can now model a coercivity distribution with a maximum number of 5 components, instead of 4.

Install GECA

Requirements

To run GECA 2.1 you need *Mathematica 5.0* and later versions installed on a Windows OS. At least 128 MB RAM and a 1 GHz CPU are recommended.

Install GECA 2.1

To install GECA 2.1, copy the source code file `MAG_MIX_1/GECA/Install/Geca.m` and the file `MAG_MIX_1/GECA/Install/components.txt` into the following directory:
`C:/.../Wolfram Research/Mathematica/5.0/AddOns/StandardPackages/Utilities`
whereby `C:/.../` depends on the installation of *Mathematica* on your computer.

WARNING

The packages `Codica.m` anf `Geca.m` are incompatible. Load only one of them in a *Mathematica* session. If you need both programs, load and use first CODICA. When you are finished with CODICA *close* the *Mathematica* session. Then, open the same or another *Mathematica* notebook and load GECA. This problem will be fixed with the next version of GECA.

Introduction

The program GECA (GEneralized Coercivity Analyzer) is part of the package MAG-MIX. GECA performs a component analysis based on special generalized functions which can fit the coercivity distributions of natural and artificial magnetic components particularly well. It also performs a Pearson's χ^2 goodness of fit test to evaluate the number of functions required to model a coercivity distribution. Finally, GECA performs an error estimation and calculate the cofidence limits for each model parameter.
Read carefully this manual to learn about GECA and take full advantage from the different possibilities offered by the program to perform a component analysis and verify its significance. This manual contains a theoretical part, which gives you the background to understand the basic ideas of GECA, and a practical part, which guides you through each step of the program. You can practice with the examples delivered together with this program. GECA is designed to work optimally on coercivity distributions calculated with CODICA and stored in files with extention `.slog`.
Click on the following topics to see the contents of this manual:

Theoretical background: coercivity distributions **2**
- Finite mixture models 2
- logarithmic Gaussian functions 2
- Skewed Generalized Gaussian functions (SGG) 3
- distribution parameters 3

Some aspects of component analysis **5**

Performing and testing a component analysis **8**
- merit function 8
- mean squared residuals 8
- Chi-square estimator 8
- local and global minima of the merit function 9
- Pearson´s Chi-square goodness of fit test 11

A program example **14**

Cautionary note **35**

Theoretical background: coercivity distributions

A group of magnetic grains with similar chemical and physical properties, distributed around characteristic values, is called a *magnetic component*. Examples of magnetic components are pedogenic magnetite (nanometric magnetite perticles with a wide grain size distribution), and magnetosomes (prismatic magnetite with a very narrow grain size distribution between 40 nm and 80 nm). Magnetic components have a simple-shaped, unimodal distribution of coercivities. Commonly, the coercivity distribution of a single magnetic component is modeled with a logarithmic Gaussian function:

$$G(x,\mu,\sigma) = \frac{1}{\sqrt{2\pi}\sigma x}\exp\left[-\frac{\log^2(x/\mu)}{2\sigma^2}\right] \tag{1}$$

In the literature, x is identified with the magnetic field H, μ is the *median destructive or acquisition field* $H_{1/2}$, called also MDF and MAD respectively, and σ the *dispersion parameter* DP. However, not all coercivity distributions can be modeled appropriately with (1). Experimental and theoretical coercivity distributions of single components are better described by distribution functions with four parameters. The two additional parameters control the skewness and the squareness of the distribution.

The coercivity distribution $f(H)$ of a mixture of different magnetic components may be considered as a linear combination of the coercivity distributions of the single components:

$$f(H) = \sum_{i=1}^{n} c_i M_{\mathrm{r}i} f_i(H \mid \theta_i) \tag{2}$$

where c_i and $M_{\mathrm{r}i}$ are the concentration and the saturated magnetization of the *i*-th component respectively, and $f_i(H \mid \theta_i)$ is the corresponding coercivity distribution with the parameters $\theta_i = (\theta_{i1},\ldots,\theta_{ik})$. Equation (2) is called a *finite mixture model*, and $f_i(H \mid \theta_i)$ are the so-called *end members*. Equation (2) assumes that the magnetization of all components adds linearly (*linear additivity*). This assumption does not hold in case of magnetic interactions between the magnetic grains of different components. However, magnetic interactions between different components are not likely to occur in natural samples, since each component is expected to have a different origin and to hold different places within a nonmagnetic matrix. On the other hand, magnetic interactions within the same component are possible, but they do not affect the linear additivity law.

Coercivity distributions of single magnetic components are described by *probability density functions* (PDF). The shape of a PDF is controlled by a set of *distribution centers* μ_n with related *dispersion parameters* σ_n, with $n \in \mathbb{N}$. Special cases are given when $n = 1$ (μ_1 is the median, σ_1 the mean deviation), $n = 2$ (μ_2 is the mean, σ_2 the standard deviation), and $n \to \infty$ (μ_∞ is the mid-range and σ_∞ the half-range). The *dispersion parameter* DP corresponds to σ_2 on a logarithmic field scale. The symmetry of a PDF is described by the *coefficient of skewness* s, where

$s = \sigma_3^3/\sigma_2^3$. Symmetric distributions are characterized by $s = 0$ and $\mu_n = \mu_2$. The curvature of a PDF is described by the *coefficient of excess kurtosis* k, where $k = \sigma_4^4/\sigma_2^4 - 3$. The Gaussian PDF is characterized by $k = 0$.

The description of non-Gaussian PDF involves the use of functions with more than two independent parameters. It is of great advantage if such functions maintain the general properties of a Gauss PDF: the n-th derivative should exist over $\mathbb{R}$ and $\sigma_n < \infty$ for all values of $n \in \mathbb{N}$. Furthermore, the Gaussian PDF should be a particular case of such functions. A good candidate is the *generalized Gaussian distribution GG*, known also as the *general error distribution*. The Gaussian PDF is a special case of *GG* distributions. Other special cases are the Laplace distribution and the box distribution. The *GG* distribution is symmetric: $s = 0$. In GECA, a particular set of *skewed generalized Gaussian distributions*, called SGG, is used to model single components. A SGG function is given by:

$$SGG(x,\mu,\sigma,q,p) = \frac{1}{2^{1+1/p}\sigma\,\Gamma(1+1/p)} \frac{\left|qe^{qx^*} + q^{-1}e^{x^*/q}\right|}{e^{qx^*} + e^{x^*/q}} \exp\left[-\frac{1}{2}\left|\ln\left(\frac{e^{qx^*} + e^{x^*/q}}{2}\right)\right|^p\right] \tag{3}$$

with $x^* = (x-\mu)/\sigma$, $x = \log H$, and $0 < |q| \le 1$. The *GG* distribution is a special case of (3) for $q = 1$, and the Gauss distribution is a special case of (3) with $q = 1$ and $p = 2$. The relation between the distribution parameters μ, σ, q, p and some statistical properties is given in Table 1.

Distribution properties	**Definition**	**Relation with the distribution parameters**	**Comments**
Median $x_{x/2}$	$\int_{-\infty}^{x_{1/2}} f(x)\,\mathrm{d}x = \frac{1}{2}$	$x_{1/2} = \mu$	$x_{x/2}$ is also called MDF or MAF
Mean μ_2	$\mu_2 = \int_{-\infty}^{+\infty} f(x)\,x\,\mathrm{d}x$	$\mu_2 \approx \mu + \frac{s}{6}(1+0.856k)$ for $q \to 1,\ p \to 2$	generally not used in the literature
Standard deviation σ_2	$\sigma_2^2 = \int_{-\infty}^{+\infty} f(x)(x-\mu_2)^2\,\mathrm{d}x$	$\sigma_2^2 = \sigma^2\,(1+0.856k)\left(1-\lvert s\rvert/3\right)$ for $q \to 1,\ p \to 2$	σ_2 is also called DP
Skewness s	$s = \sigma_3^3/\sigma_2^3$ $\sigma_3^3 = \int_{-\infty}^{+\infty} f(x)(x-\mu_2)^3\,\mathrm{d}x$	$s \approx -6\,\mathrm{sgn}\,q\,(1-q)^2\,(1+1.856k)$	$q > 0$: left skewed $q < 0$: right skewed
Kurtosis k	$s = \sigma_3^4/\sigma_2^4 - 3$ $\sigma_4^4 = \int_{-\infty}^{+\infty} f(x)(x-\mu_2)^4\,\mathrm{d}x$	$k \approx 2 - p$	$p > 2$: box-shaped $p < 2$: tip-shaped

Table 1: *Relation between statistical distribution properties and distribution parameters for a SGG function. Except for the median, the relations are not analytical; approximations are given in the case of small deviations from a Gaussian distribution.*

Examples of *SGG* functions with different parameters are given in Fig. 1. The parameters of the coercivity distribution of some calculated and measured coercivity distributions are plotted in Fig. 2.

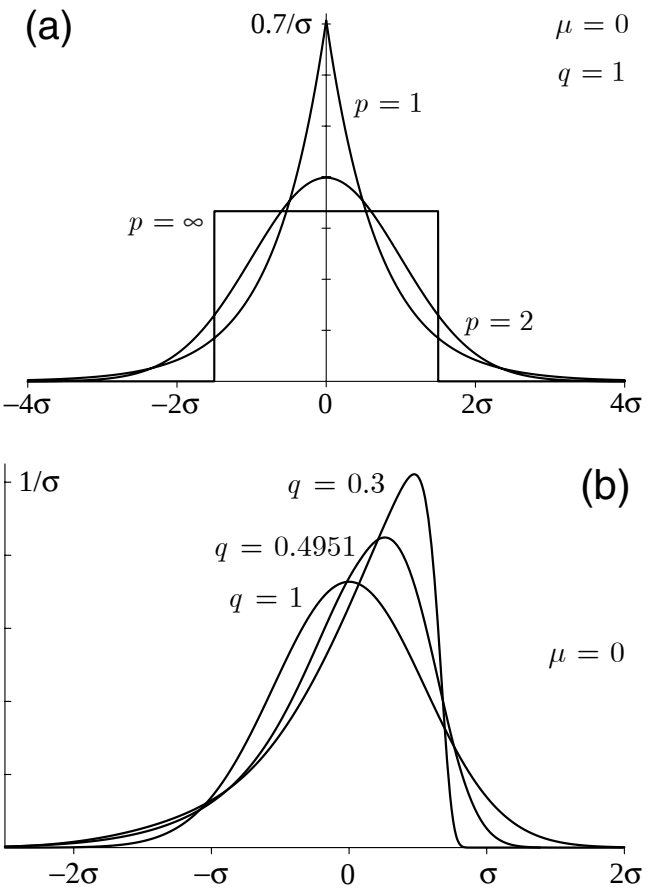


Fig. 1: *Examples of SGG distributions.* (a) *Some particular cases with* $\mu = 0$, $\sigma_2 = 1$ *and* $q = 1$ *are plotted. The skewness of all curves is zero. Furthermore,* $p = 1$ *for a Laplace distribution,* $p = 2$ *for a Gauss distribution and* $p = \infty$ *defines a box distribution.* (b) *Some left-skewed SGG distributions with* $\mu = 0$ *and* $\sigma_2 = 0.5484$ *are plotted. The SGG distribution with* $q =$ 0.4951 *is an excellent approximation of the logarithmic plot of a negative exponential distribution.*

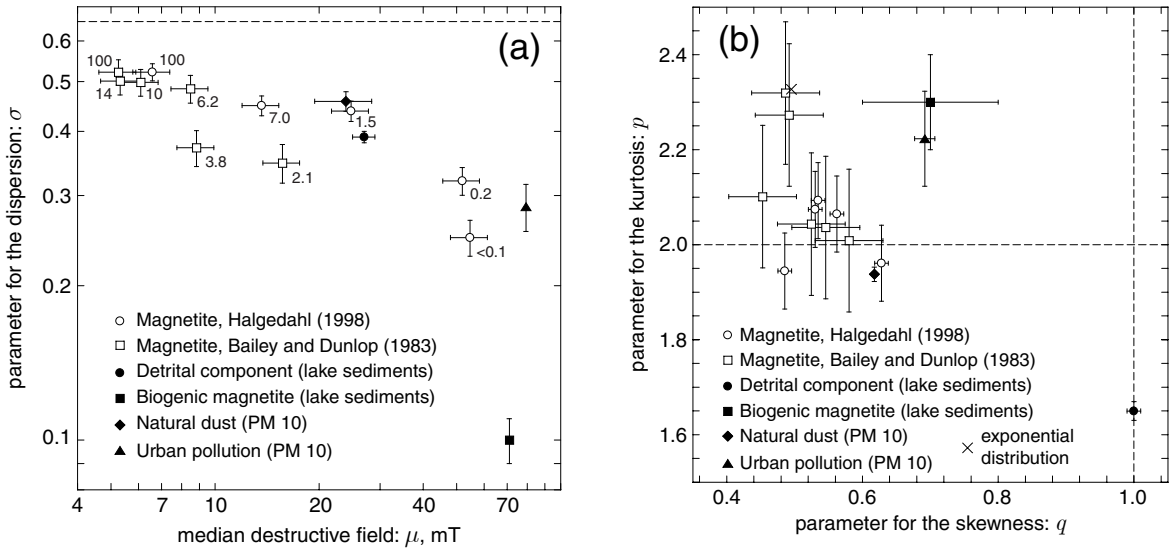


Fig. 2: *Coercivity distribution parameters* μ, σ, q *and* p *for the AF demagnetization of IRM in various synthetic and natural samples. Numbers beside the points indicate the grain size in* μm. (a) *Scatter plot of* μ *and* σ. *The dashed line indicates the value of* σ *for a negative exponential distribution.* (b) *Scatter plot of* q *and* p. *The cross point of the dashed lines corresponds to the values of* q *and* p *for a logarithmic Gaussian distribution. All samples show significant deviation form a logarithmic Gaussian distribution. All parameters of sized magnetite are intermediate between those of a logarithmic Gaussian distribution and those of an exponential distribution.*

Some aspects of component analysis

The result of a component analysis depends upon the PDF chosen to model the end-member coercivity distributions, and particularly on the number of parameters assigned to each PDF. Strong differences exist between the results obtained with a linear combination of Gaussian distributions on the one hand, and a linear combination of SGG distributions on the other. Since finite mixture models with non-Gaussian coercivity distributions have not been reported in the literature, it is not possible to decide from a-priori information which kind of PDF should be used as a basis for a finite mixture model. From the mathematical point of view, all PDFs are equivalent, since the goodness of fit which can be reached with a particular model depends only upon the total number of parameters assumed, regardless of how they are assigned to individual components. Generally, the use of few PDFs with more distribution parameters, instead of a large number of distributions with fewer distribution parameters leads to results of the fitting model which are more stable against measurement errors. The stable behavior of a fitting with SGG distributions can be explained by the fact that small deviations from an ideal coercivity distribution, which arise from measurement errors, are taken into account by variations in skewness and kurtosis, rather than by variations in the contributions of the single components. Obviously, the values obtained for skewness and kurtosis may not be significant at all. A similar stability can be obtained with Gaussian functions if some of them are grouped as if they were one component. However, it is not always evident which distributions group together, and multiple solutions are often possible. The aspects discussed above are illustrated with the examples of Fig. 3 and Fig. 4. Both figures show the results of a component analysis performed with GECA on the coercivity distribution of a sample of urban particulate matter. In Fig. 3, the component analysis is performed with logarithmic Gaussian functions. Four logarithmic Gaussian functions are needed to fit the measured data so that the misfit between model and data is compatible the measurement errors. However, it is impossible to identify these four distributions with an equivalent number of magnetic components. In Fig. 4, the component analysis is performed with SGG functions. The measurements are already well fitted with one SGG function, however, the measured and the modeled coercivity distributions differens significantly. This model could be adequate to describe low-precision measurements of the same sample. Two SGG functions fit the data within the margins given by the measurement errors. However, multiple solutions are possible, but only one solution minimize the difference between model and measurements. The other solutions imply rather uncommon shapes for the coercivity distribution of the individual components, which are not likely occur in natural samples.

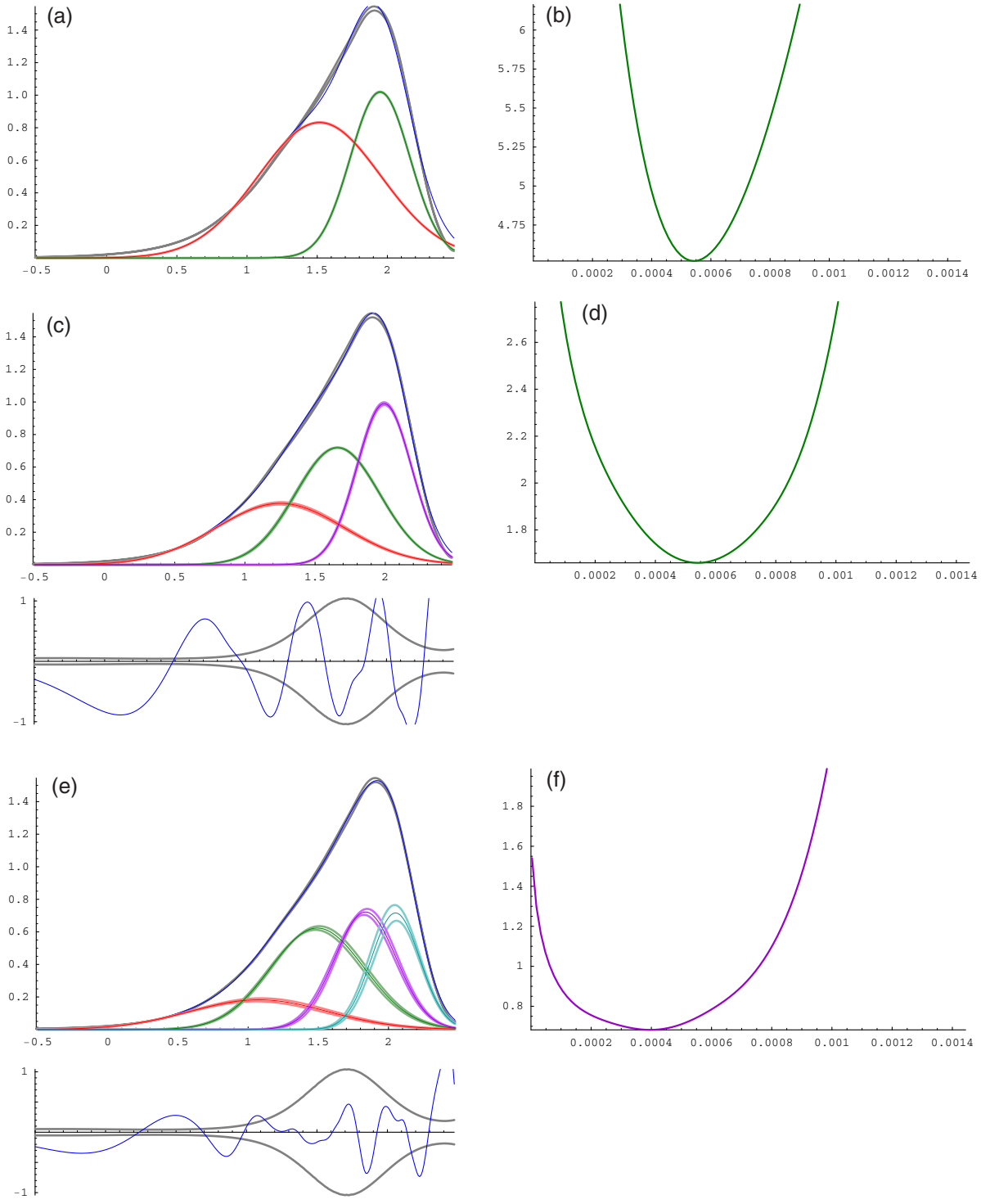


Fig. 3: *Component analysis on a sample of urban atmospheric dust collected in Zürich, Switzerland. The component analysis is performed with logarithmic Gaussian functions. Results of the component analysis are shown in* (a), (c) *and* (e). *The gray pair of line indicates the confidence limits of the measured coercivity distribution. The blue line is the modelled coercivity distribution, expressed as the sum of the logarithmic Gaussian distributions (red, green, violet and light blue). Confidence limits are plotted around each function. Below each plot, the difference between measured and modelled curve is drawn in blue; the gray pair of curves indicates the amplitude of the measurement errors. The mean quadratic residuals of each model are plotted in* (b), (d) *and* (f) *as a function of the amplitude of the logarithmic Gaussian function labeled with the same color. The solutions plotted in* (a), (c) *and* (e) *represents the absolute minima of* (b), (d) *and* (f).

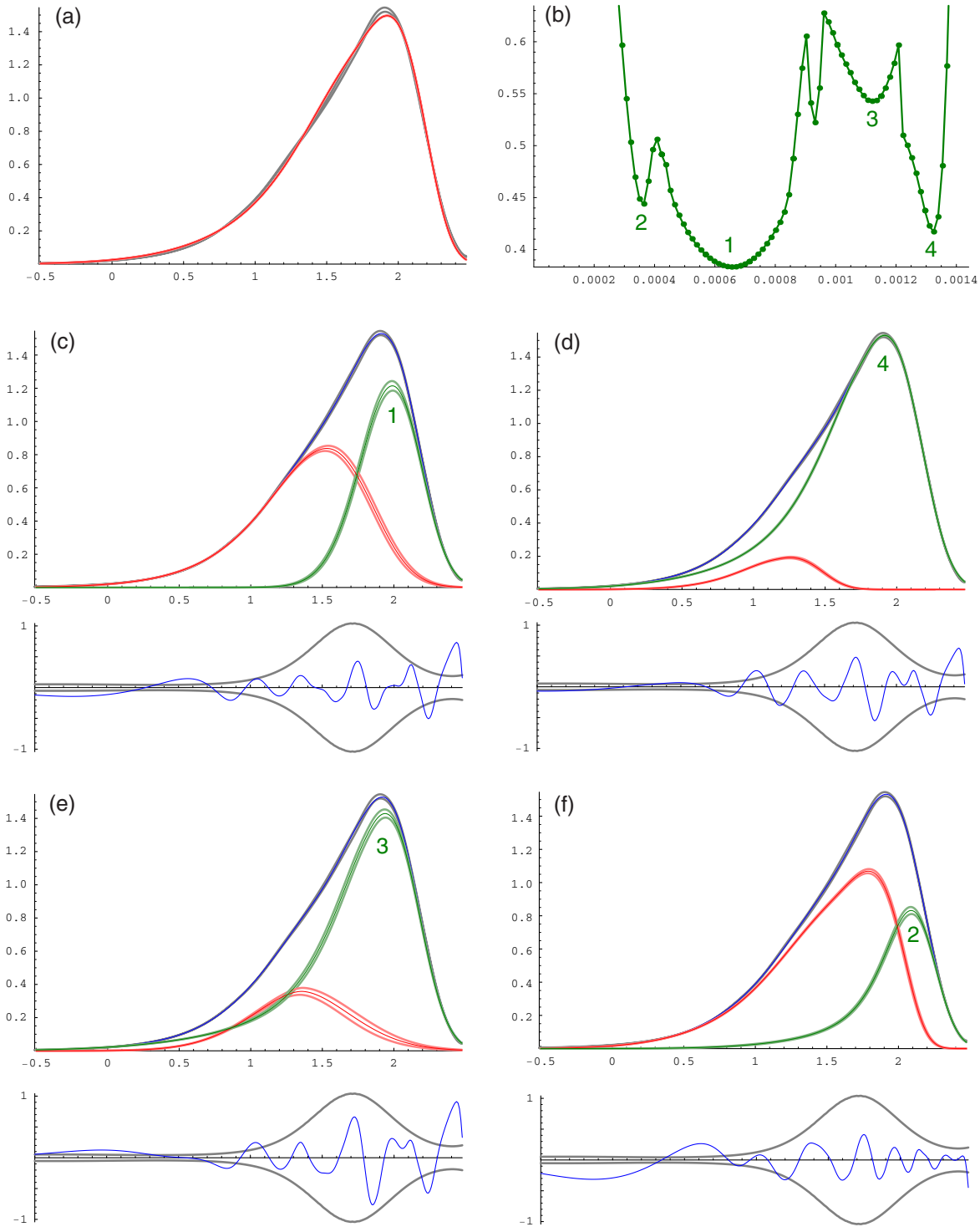


Fig. 4: *Component analysis of the same sample as in Fig. 3. The component analysis is performed with SGG functions. The same notation as in Fig. 3 is used for the plots.* (a) *Component analysis with one SGG function. The modeled coercivity distributions is significantly different from the measured distribution.* (b) *Mean quadratic residuals of a model with two SGG functions, plotted as a function of the amplitude of one function. Different local minima which correspond to stable solutions of the component analysis are labeled with numbers. The corresponding solutions are plotted in* (c), (d), (e) *and* (f). *The solution plotted in* (c) *corresponds to the global minimum of* (b) *and the resulting components are compatible with the coercivity distributions of natural dust (red), and combustion products of motor vehicles (green). The solutions corresponding to the other local minima of* (b) *are not realistic.*

The fundamental questions related to component analysis are:

- How many components are needed to fit a given coercivity distribution?
- Are multiple solutions possible? If yes, which solution is correct?

The answer to these questions is not simple. In the example of Fig. 4 the number of components and the identification of the correct solution among multipe solutions is evident. However, this is not always possible, especially if good measurements are not available, or if the coercivity distributions of individual components are too widely overlapped. In this case, some additional information is needed to put appropriate constraints to the number of end-members and to their distribution parameters.

Performing and testing a component analysis

When component analysis is performed, a modeled coercivity distribution $f(x \mid \boldsymbol{\theta})$ with parameters $\boldsymbol{\theta} = (\theta_1,\ldots,\theta_n)$ is compared with the measured coercivity distribution, given by a set of numerical values $(x_i, f_i \pm \delta f_i)$ with measurement errors δf_i. A solution of the component analysis is represented by a set of values of $\boldsymbol{\theta}$ which minimizes a so-called *merit function* $\varepsilon(\boldsymbol{\theta})$. The merit function is an estimation of the difference between the modeled and the measured curve: $\varepsilon = 0$ if the model is identical with the measurements. Examples of $\varepsilon(\boldsymbol{\theta})$ are the mean squared residual:

$$d^2(\boldsymbol{\theta}) = \sum_{i=1}^{N} [f(x_i \mid \boldsymbol{\theta}) - f_i]^2 \tag{4}$$

used for a least-squares fitting, and the χ^2 estimator:

$$\chi^2(\boldsymbol{\theta}) = \sum_{i=1}^{N} \left[\frac{f(x_i \mid \boldsymbol{\theta}) - f_i}{\delta f_i} \right]^2 \tag{5}$$

used for a minimum χ^2 fitting. GECA uses following weighted version of the χ^2 estimator:

$$w^2(\boldsymbol{\theta}) = \sum_{i=1}^{N} r_i^{-2} \left[\frac{f(x_i \mid \boldsymbol{\theta}) - f_i}{\delta f_i} \right]^2 \tag{6}$$

where $r_i = \delta f_i / f_i$ are the relative errors. In this case, measurement points affected by a large relative error are less considered for the component analysis. Equation (6) can be rewritten as:

$$w^2(\boldsymbol{\theta}) = \sum_{i=1}^{N} \left(\frac{\delta f_i}{\sqrt{f_i}} \right)^{-4} [f(x_i \mid \boldsymbol{\theta}) - f_i]^2 \tag{7}$$

If $f(x)$ originates from the sum of a finite number of elementary contributions, f_i is a Poisson distributed variable, and $(\delta f_i)^2 \propto f_i$. An experimental confirmation of this assumption is shown in Fig. 5. After these considerations, $w^2(\boldsymbol{\theta}) \propto d^2(\boldsymbol{\theta})$ and $d^2(\boldsymbol{\theta})$ is used by GECA as an improved merit function with respect to (6), since the randomizing effect of the measurement errors on the weighting factors r_i is removed.

Generally, the merit function $\varepsilon(\boldsymbol{\theta})$ has several local minima $\varepsilon_{\min} = \varepsilon(\boldsymbol{\theta}_{\min})$, which correspond to stable solutions $\boldsymbol{\theta}_{\min}$ of the component analysis. Among these minima, there is an absolute minimum $\varepsilon_{\mathrm{MIN}} = \varepsilon(\boldsymbol{\theta}_{\mathrm{MIN}})$. Depending on the starting values $\boldsymbol{\theta}_{\mathrm{ini}}$ of $\boldsymbol{\theta}$, one of these solutions is attained by GECA.

If the model used for component analysis is adequate and if there are no measurement errors, $\varepsilon_{\mathrm{MIN}} = 0$. Let n be the number of magnetic components and m the number of end-member functions used in the model. Then, $\varepsilon_{\mathrm{MIN}} > 0$ for $m < n$ and $\varepsilon_{\mathrm{MIN}} = 0$ for $m \geq n$, so that the number of components can be easily guessed (Fig. 6a). In case of an inadequate model, the end-member functions cannot reproduce exactly the coercivity distribution of all magnetic components, and $\varepsilon_{\mathrm{MIN}} > 0$, even without measurement errors.

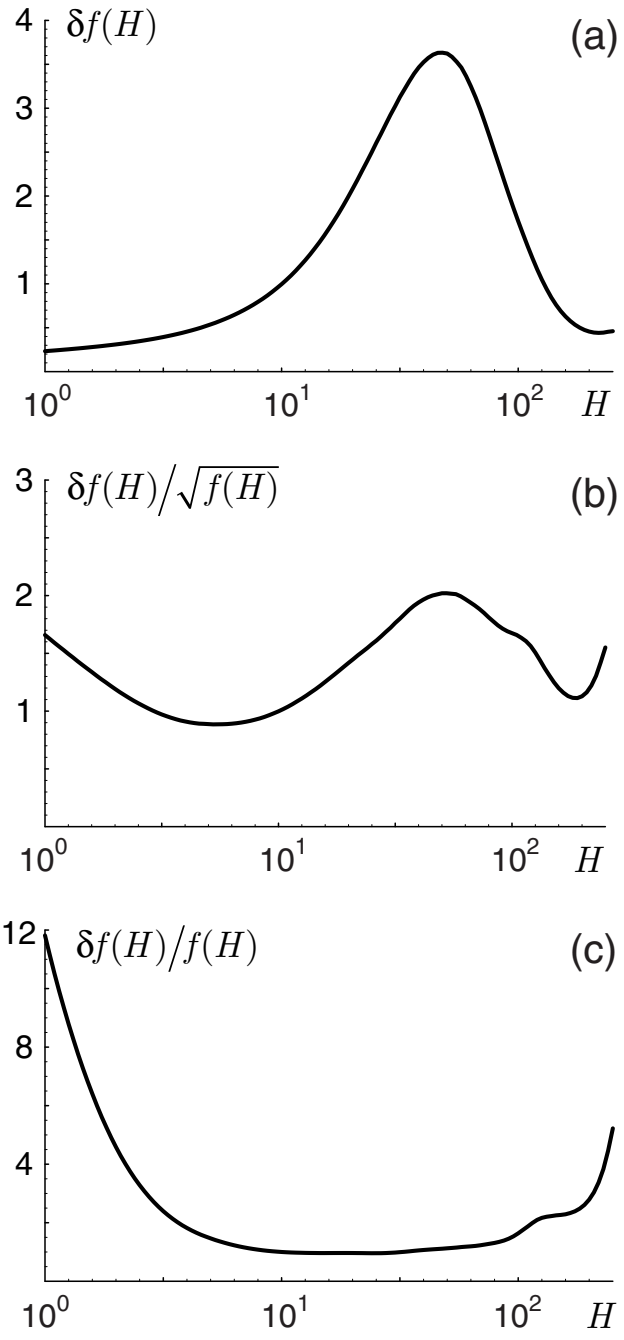


Fig. 5: *Mean measurement error of the coercivity distribution of six samples of loess, soil, lake sediments, marine sediments and atmospheric particulate matter. The absolute error* $\delta f(H)$ *and the relative error* $\delta f(H)/f(H)$ *are plotted in* (a) *and* (c)*, respectively. In* (b)*, the absolute error is normalized by the square root of* $f(H)$*. The field unit is* mT*. All curves are normalized by their value at* 10 mT.

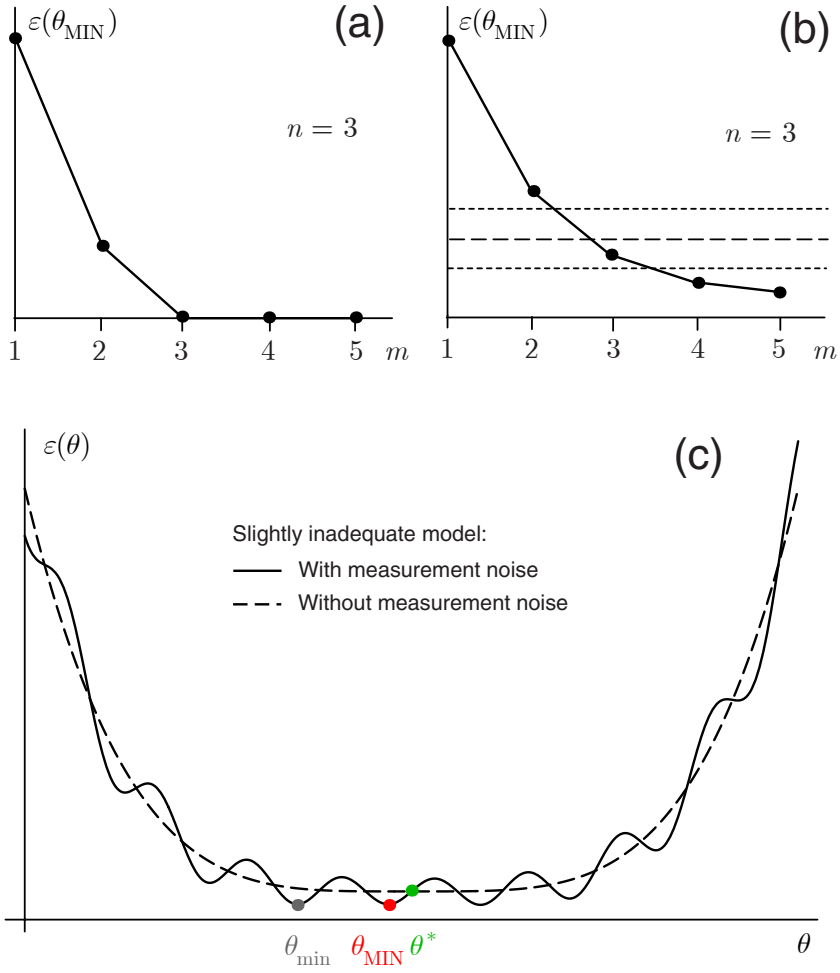


Fig. 6: *Dependence of the merit function* $\varepsilon(\boldsymbol{\theta})$ *on the parameters of the model chosen for fitting a coercivity distribution. In* (a) *a noise-free coercivity distribution with* $n = 3$ *magnetic components is fitted with an adequate model with* m *end-member functions. The functions are assumed to reproduce exactly the coercivity distribution of each component. If* $m < n$, *some components cannot be considered into the model and* $\varepsilon(\boldsymbol{\theta}) > 0$; *on the other hand* $\varepsilon(\boldsymbol{\theta}) = 0$ *for a given combination* $\boldsymbol{\theta} = \boldsymbol{\theta}_{\mathrm{MIN}}$ *of parameters when* $m > n$. *The number of components can be easily guessed. The situation becomes more complex in* (b), *where measurement errors are taken into account. In this case, there is always a misfit between model and measurements, and* $\varepsilon(\boldsymbol{\theta}_{\mathrm{MIN}})$ *decreases monotonically as the number of end-members taken into account by the model is increased. In this case, the number of components is guessed with the help of a Pearson's* χ^2 *test. According to this test,* $\varepsilon(\boldsymbol{\theta}_{\mathrm{MIN}})$ *is compared with the expected value of* ε *(dashed line). If* $\varepsilon(\boldsymbol{\theta}_{\mathrm{MIN}})$ *is compatible with the expected value within given confidence limits (dotted lines), the model is accepted. If* $\varepsilon(\boldsymbol{\theta}_{\mathrm{MIN}})$ *is too large, the modeled coercivity distribution is significantly different from the measured coercivity distribution and more parameters should be included in the model. On the other hand, if* $\varepsilon(\boldsymbol{\theta}_{\mathrm{MIN}})$ *is too small, the model fits the measured data unrealistically well and random effects produced by the measurement errors are included in some parameters which are not significant. The model is accepted if* $\varepsilon(\boldsymbol{\theta}_{\mathrm{MIN}})$ *belongs to the range of values given by the confidence limits. The complex dependence of the merit function on the model parameters is illustrated in* (c) *for the case of a model with a fixed number of end-member functions which approximately fit the coercivity distribution of all magnetic components. These end-member functions produce a small misfit between model and data, even is the measurement errors are not considered (dashed curve). Nevertheless, there is only one stable solution* $\boldsymbol{\theta}^*$ *of the component analysis (green point), which corresponds to an absolute minimum of* $\varepsilon(\boldsymbol{\theta})$. *If the measurement errors are taken into account, the shape of* $\varepsilon(\boldsymbol{\theta})$ *becomes rather complex, with numerous local minima* $\varepsilon(\boldsymbol{\theta}_{\mathrm{min}})$. *Some of these local minima represent possible solutions which fits the measurements as good as the absolute minimum* $\varepsilon(\boldsymbol{\theta}_{\mathrm{MIN}})$, *even if they do not model the coercivity distribution of the real components. The absolute minimum (red point) represents a solution* $\boldsymbol{\theta}_{\mathrm{MIN}}$ *which is still close to the realitiy. With larger measurement errors, this could not be the case, and a realistic solution may be given by a local minimum of* $\varepsilon(\boldsymbol{\theta})$.

If an adequate model is used to fit data affected by measurement errors, $\varepsilon_{\mathrm{MIN}} > 0$, and $\varepsilon_{\mathrm{MIN}} \to 0$ for $m \to \infty$ (Fig. 6b).

Two fundamental questions arise at this point:

1) How many end-member distributions should be considered for a component analysis?

2) Is a particular solution $\hat{\boldsymbol{\theta}}$ close to the (unknown) real solution $\boldsymbol{\theta}^*$?

These questions can be easily answered only if the model chosen for the component analysis is adequate and the measurement errors are sufficiently small. The first condition can be approximatively attained by using a set of SGG functions to model the coercivity distributions of the magnetic components. SGG functions are able to reproduce all fundamental characteristics of the coercivity distribution of a single component (median, dispersion parameter, skewness and kurtosis).

If the measurement errors are small enough, the solution $\boldsymbol{\theta}_{\mathrm{MIN}}$ which corresponds to a global minimum of $\varepsilon(\boldsymbol{\theta})$ is close to the real solution $\boldsymbol{\theta}^*$ (Fig. 6c). In case of large measurement errors, the real solution $\boldsymbol{\theta}^*$ may be close to one or more a local minima of $\varepsilon(\boldsymbol{\theta})$. In this case, additional independent information are needed to individuate the correct solution among all possible solutions $\boldsymbol{\theta}_{\mathrm{min}}$.

The problem of the number of end-members to consider for a component analysis is evaluated with a *Pearson's* χ^2 *goodness of fit test*. To perform this test, the statistical distribution of the χ^2 estimator given in equation (5) is considered. The χ^2 estimator is a statistical variable which is distributed according to a χ^2 distribution with $N-k-1$ degrees of freedom, being N the number of *independent* points to fit with a given model, and k the number of model parameters. The expected value of the χ^2 estimator is $N-k-1$. The confidence limits at a confidence level α (generally $\alpha = 0.95$) are given by $\chi^2_{N-k-1;\alpha}$ and $\chi^2_{N-k-1;1-\alpha}$, with:

$$\int_{\chi^2_{N-k-1;p}}^{\infty} \chi^2_{N-k-1}(t)\,\mathrm{d}t = p \tag{8}$$

If $\chi^2 > \chi^2_{N-k-1;1-\alpha}$, the model differs significantly from the measurements. The model should be refined by adding new parameters, eventually by considering an additional end-member function. If $\chi^2 < \chi^2_{N-k-1;\alpha}$ the differences between model and measurements are unrealistically small. An excessive number of parameters allow the model to include random effects of the measurement errors. Consequently, some of these parameters are not significant. The model should be revised to include a smaller number of parameters, eventually by reducing the number of end-members or by keeping some parameters fixed. If $\chi^2_{N-k-1;\alpha} \le \chi^2 \le \chi^2_{N-k-1;1-\alpha}$ the model is acceptable.

To calculate the χ^2 estimator with equation (5) some knowledge about the measurement errors δf_i and the number of independent data points is necessary. The measurement errors are automatically estimated with CODICA, when a coercivity distribution is calculated from an acquisition/demagnetization curve. The number of independent data points is more difficult to estimate. It is identical with the number of measurements if the measurement errors are equivalent to an ergodic noise signal, that is, when the autocorrelation of the noise signal is equivalent to a Dirac δ-function.

This is often not the case with real measurements, where entire groups of measured points are affected by the same error. Furthermore, the coercivity distributions calculated by CODICA are low-pass filtered, and an autocorrelation of the remaining measurement errors is unavoidable. GECA estimates the degrees of freedom of the fitting model by evaluating the residuals curve which results from the difference between the model and the measurements. The residuals curve contains a certain number of random oscillations around a mean value of zero. To reproduce these oscillations a minimum number l of points is necessary, whose spacing defines the Nyquist frequency of the signal. GECA sets $l-1$ equal to the number of zero crosses of the residuals. Obviously, the shape of the residuals curve depends on the model chosen for component analysis.

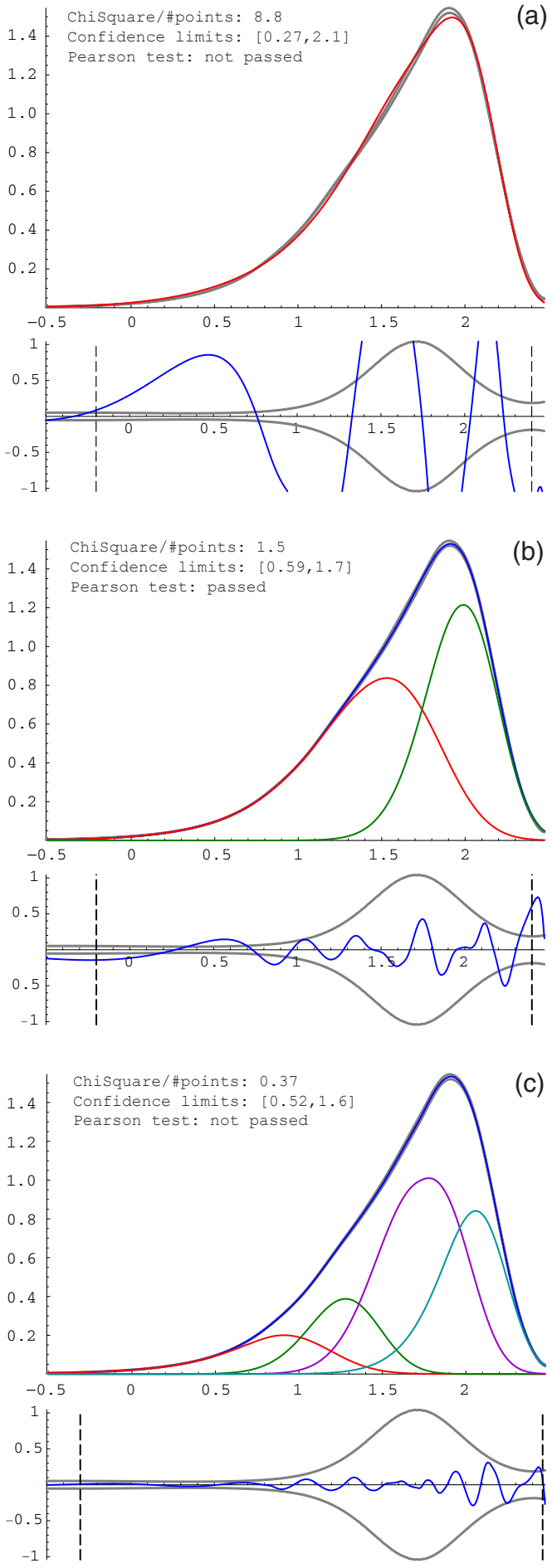


Fig. 7: *Examples of Pearson's χ^2 test on the component analysis of a sample of urban atmospheric particulate matter. The gray and the blue curves are the measured and the modelled coercivity distributions, respectively. Curves labeled with other colors represent the coercivity distributions of individual end-members. Below each plot, the difference between model and measurements is plotted (blue line) together with the measurement errors (pair of gray lines). In* (a)*, a model with one SGG function is evaluated. The differences between model and measurements are too large, and the model is rejected. In* (c) *the a model with four SGG functions is rejected for the opposite reason: the model fits the data unrealistically well for the given measurement errors. A model with two SGG functions is represented in* (b)*. In this case, the χ^2 statistics is compatible with the expected value within at a 95% confidence level, and the model is accepted.*

A model with a small number of parameters produces a residuals curve with few, large oscillations. The more parameters are included in the model, the more oscillations characterize the residuals and the confidence limits of the χ^2 estimator become closer to the expected value. Consequently, models with a too large number of parameters are rejected. An example of Pearson's χ^2 test is shown in Figure 7 with the example of a sample of urban atmospheric particulate matter. In Fig. 7a, the coercivty distribution is fitted with one SGG function. The residuals curve has 5 zero crosses in the range considered for fitting, and GECA assumes $l = 6$ degrees of freedom for the χ^2 distribution. The confidence limits of χ^2/l are 0.27 and 2.1, while $\chi^2/l = 8.8$ for that model, which is rejected. In Fig. 7b, two SGG functions are used for the component analysis. Now, $l = 12$, and the confidence limits of χ^2/l are 0.44 and 1.8. With $\chi^2/l = 1.5$ this model is acceptable. With four SGG functions (Fig. 7c), $l = 18$ and the confidence limits of χ^2/l are 0.52 and 1.6, while $\chi^2/l = 0.37$ for that model, which is rejected.

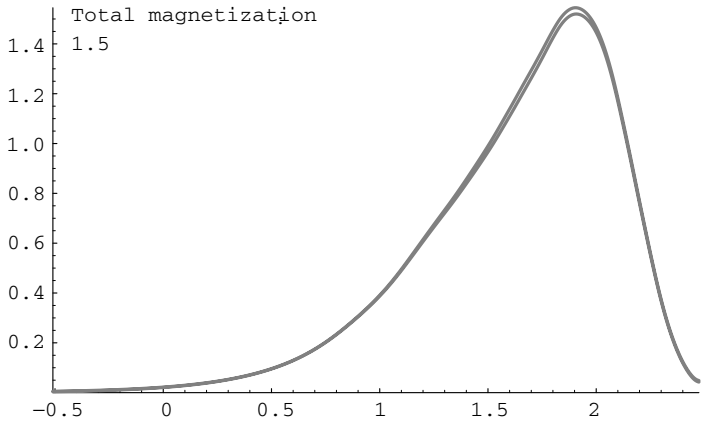
A program example

In[1]:= **<<Utilities`Geca`** Load the program

```
GECA v.2.1 for Mathematica 5.0 and later versions.
Distributed with the package MAG-MIX release 1, 04/04/2005.
Copyright 2005 by Ramon Egli. All rights reserved.
```

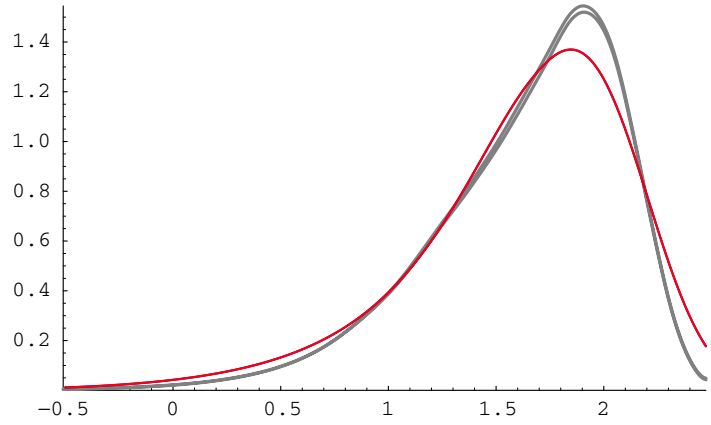
In[2]:= **Geca** Start the program

```
Data from file                                          Enter file name
C:/users/ramon/papers/fitting/WDKarm.slog

Checking the coercivity distribution...

Confidence limits of the coercivity distribution:       Plot the distribution
```




```
Coercivity distribution is significant between -0.5 and 2.474

Fitting is performed in the range between -0.1957 and 2.396   Set the fitting range
```

Enter initial parameters (1)



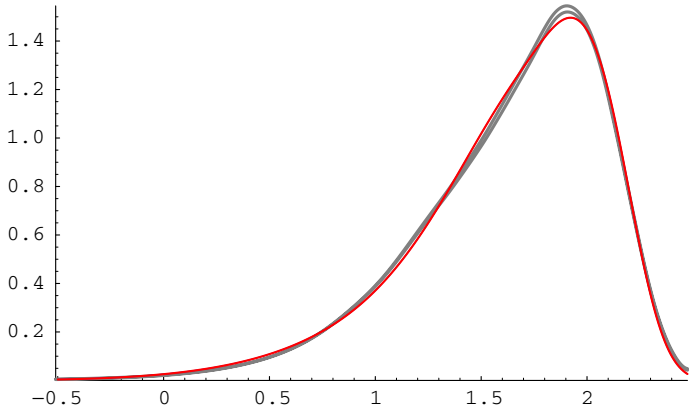
Initial distribution parameters:

a1 = 1.5
m1 = 1.7 s1 = 0.6
q1 = 0.5 p1 = 2.2

Optimizing the distribution parameters. Please wait...

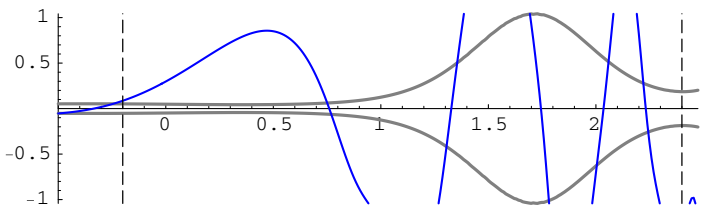
Perform a component analysis (1)

{a1=0.0015, m1=1.7, s1=0.6, q1=0.5, p1=2.2}



Optimized distribution parameters:

a1 = 1.454
m1 = 1.714 s1 = 0.6875
q1 = 0.4256 p1 = 3.076



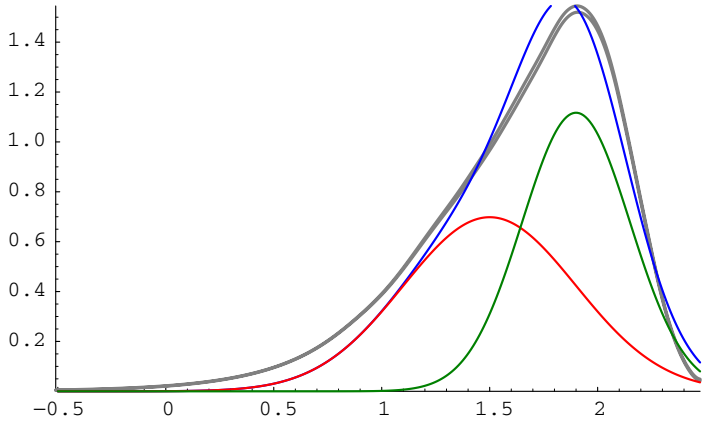
Residuals (modelled - measured) and measurement errors
in % of the maximum value of
the coercivity distribution.

Dashed lines delimitate the interval considered for the component analysis.

ChiSquare/#points: 8.8
Confidence limits: [0.52,1.9]

Model and data are significantly different. Refine your model.

Enter initial parameters (2)



Initial distribution parameters:

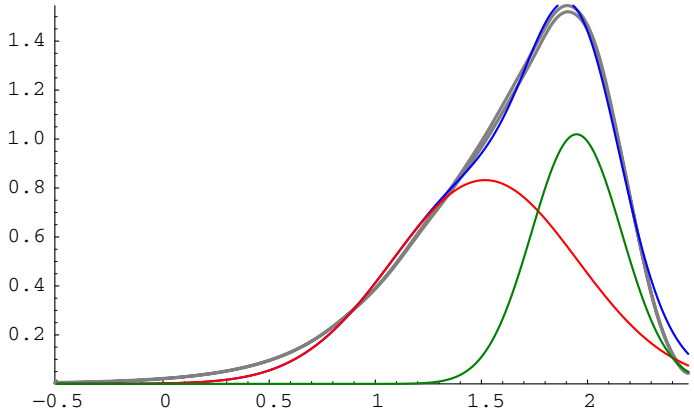
a1 = 0.7
m1 = 1.5 s1 = 0.4
q1 = 1. p1 = 2.

a1 = 0.7
m1 = 1.9 s1 = 0.25
q1 = 1. p1 = 2.

Optimizing the distribution parameters. Please wait...

Perform a component analysis (2)

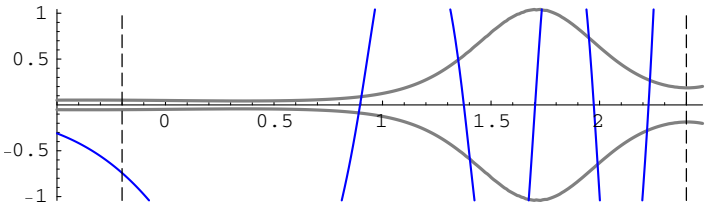
{a1=0.0007,m1=1.5,s1=0.4,a2=0.0007,m2=1.9,s2=0.25}



Optimized distribution parameters:

a1 = 0.9101
m1 = 1.517 s1 = 0.4363
q1 = 1. p1 = 2.

a1 = 0.5428
m1 = 1.948 s1 = 0.2124
q1 = 1. p1 = 2.



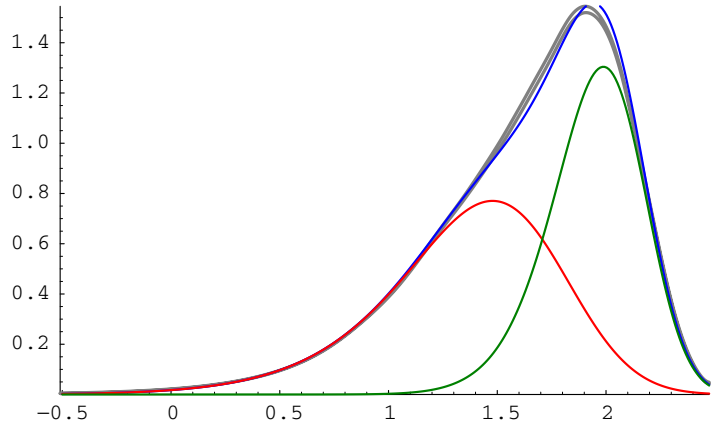
Residuals (modelled - measured) and measurement errors
in % of the maximum value of the coercivity distribution.

Dashed lines delimitate the interval considered for the component analysis.

ChiSquare/#points: 27.
Confidence limits: [0.52,1.9]

Model and data are significantly different. Refine your model.

Enter initial parameters (3)



Initial distribution parameters:

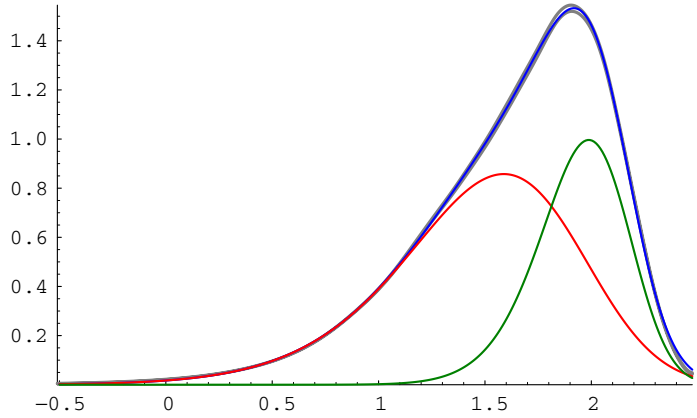
a1 = 0.75
m1 = 1.4 s1 = 0.45
q1 = 0.6 p1 = 2.

a1 = 0.7
m1 = 1.957 s1 = 0.235
q1 = 0.663 p1 = 2.

Optimizing the distribution parameters. Please wait...

Perform a component analysis (3)

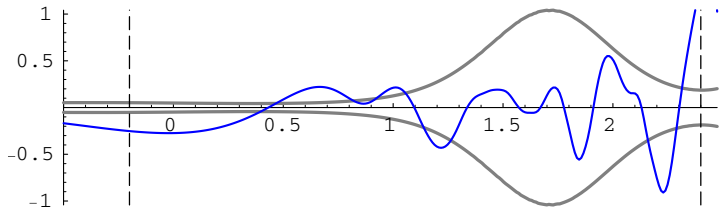
{a1=0.00075,m1=1.4,s1=0.45,q1=0.6,p1=2.,a2=0.0007}



Optimized distribution parameters:

a1 = 0.9217
m1 = 1.509 s1 = 0.4882
q1 = 0.6235 p1 = 2.023

a1 = 0.5348
m1 = 1.957 s1 = 0.235
q1 = 0.663 p1 = 2.



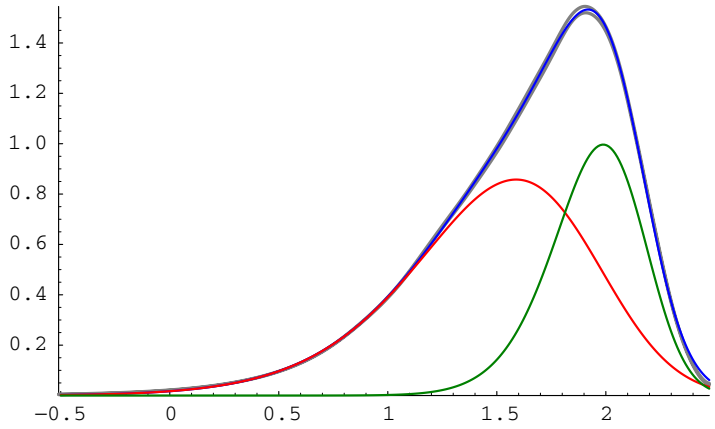
. Residuals (modelled - measured) and measurement errors
in % of the maximum value of the coercivity distribution.

Dashed lines delimitate the interval considered for the component analysis.

ChiSquare/#points: 2.8
Confidence limits: [0.63,1.6]

Model and data are significantly different. Refine your model.

Enter initial parameters (4)



Initial distribution parameters:

a1 = 0.9217
m1 = 1.509 s1 = 0.4882
q1 = 0.6235 p1 = 2.023

a1 = 0.5348
m1 = 1.957 s1 = 0.235
q1 = 0.663 p1 = 2.

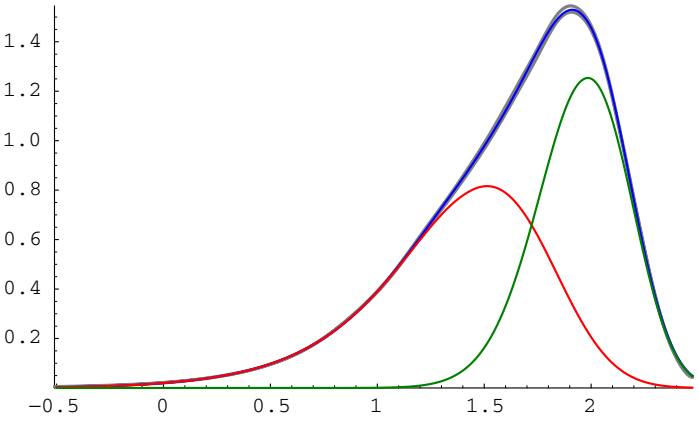
Optimizing the distribution parameters. Please wait...

Perform a component analysis (4)

{a1=0.0009217,m1=1.509,s1=0.4882,q1=0.6235,p1=2.023,
a2=0.0005348,m2=1.957,s2=0.235,q2=0.663,p2=2.}

FindMinimum::fmlim: The minimum could not be bracketed in 50 iterations.

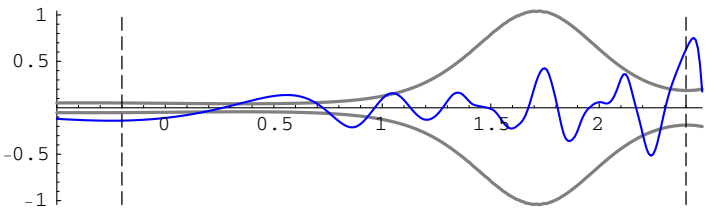
{a1=0.0006243,m1=1.311,s1=0.4483,q1=0.4878,p1=2.171,
a2=0.0008625,m2=1.963,s2=0.2331,q2=0.7765,p2=2.107}



Optimized distribution parameters:

a1 = 0.7699
m1 = 1.412 s1 = 0.4699
q1 = 0.5537 p1 = 2.106

a1 = 0.6847
m1 = 1.962 s1 = 0.2329
q1 = 0.7221 p1 = 2.053



Residuals (modelled - measured) and measurement errors
in % of the maximum value of the coercivity distribution.

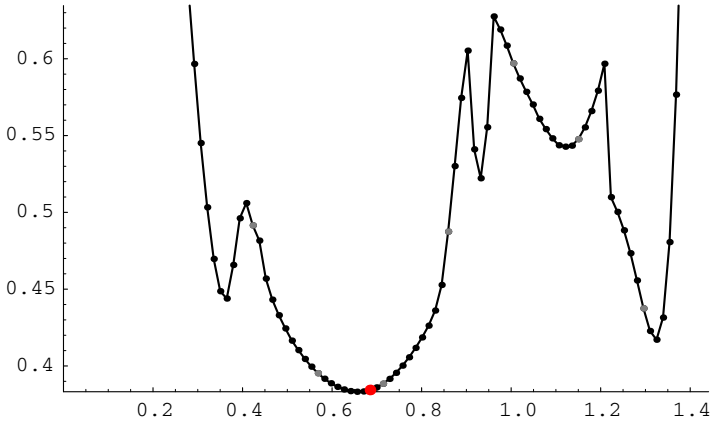
Dashed lines delimitate the interval considered for the component analysis.

ChiSquare/#points: 1.5
Confidence limits: [0.59,1.7]

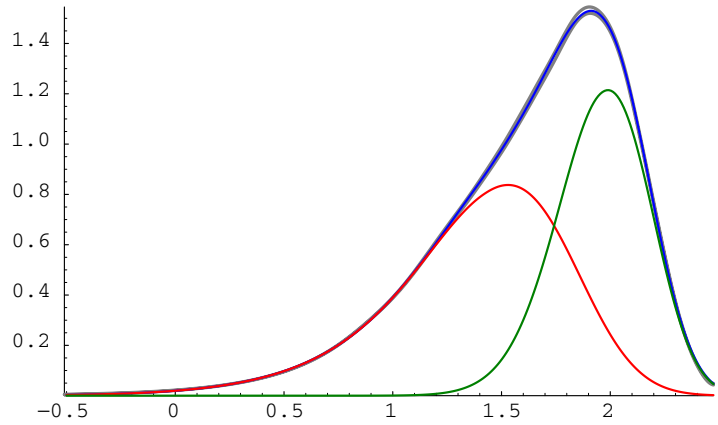
Model and data are compatible. You may accept this component analysis.

Systematic solution search

```
Perform an automatic variation of the contribution of component #2:
This process takes several minutes. Please wait...

Decreasing contribution of component #2...

Increasing contribution of component #2...

Residuals as a function of the contribution of component #2
(Every 10th point in gray, first point is #21, red point is the starting solution)
```



Choose initial parameters



Initial distribution parameters:

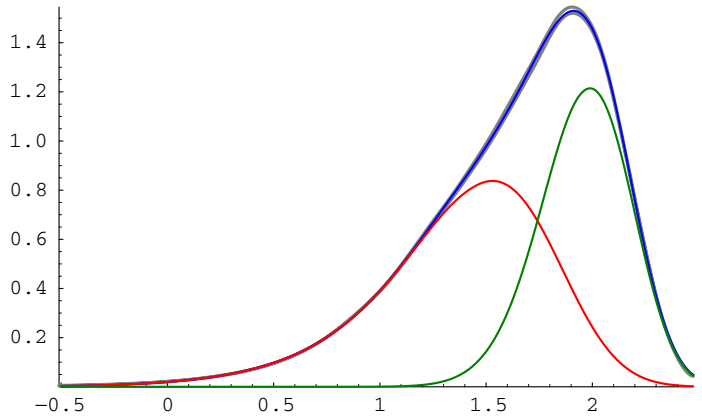
a1 = 0.7989
m1 = 1.428 s1 = 0.4739
q1 = 0.5562 p1 = 2.108

a1 = 0.6556
m1 = 1.967 s1 = 0.2297
q1 = 0.7265 p1 = 2.052

Optimizing the distribution parameters. Please wait...

Perform a component analysis (5)

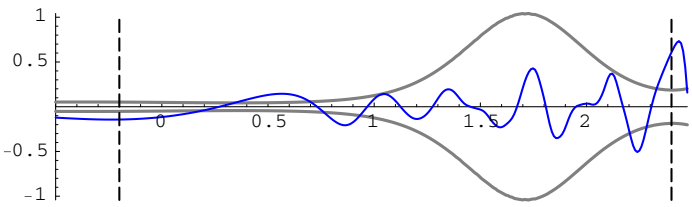
{a1=0.0007989,m1=1.428,s1=0.4739,q1=0.5562,p1=2.108,a2=0.0006556,
m2=1.967,s2=0.2297,q2=0.7265,p2=2.052}



Optimized distribution parameters:

a1 = 0.7989
m1 = 1.428 s1 = 0.4739
q1 = 0.5562 p1 = 2.108

a1 = 0.6556
m1 = 1.967 s1 = 0.2297
q1 = 0.7265 p1 = 2.052



Residuals (modelled - measured) and measurement errors
in % of the maximum value of
the coercivity distribution.

Dashed lines delimitate the interval considered for the component analysis.

ChiSquare/#points: 1.5
Confidence limits: [0.59,1.7]

Model and data are compatible. You may accept this component analysis.

Calculating statistical parameters of the distributions...

Perform an error estimation

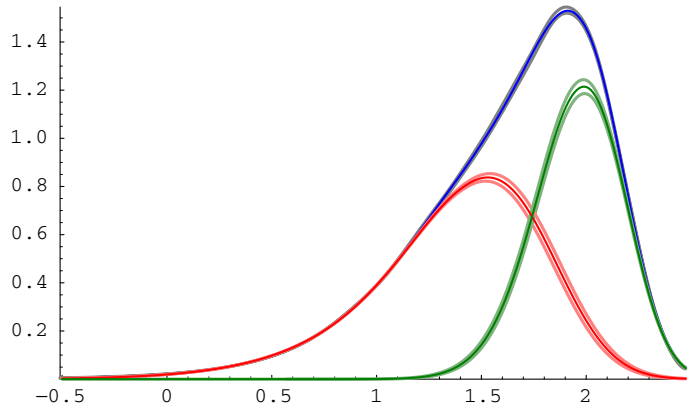
Perform an error estimation of the distribution parameters with 64 error simulations.
Accuracy of the error estimation: 12.%
This process takes several minutes time. Please wait...

Error estimation of the statistical parameters. Please wait...

Calculating the confidence limits of the components. Please wait...

Parameters of component #1:	Parameters of component #2:
a = 0.7989 ± 0.021	a = 0.6556 ± 0.021
μ = 1.428 ± 0.011	μ = 1.967 ± 0.0038
σ = 0.4739 ± 0.0029	σ = 0.4739 ± 0.0029
q = 0.5562 ± 0.0047	q = 0.7265 ± 0.0065
p = 2.108 ± 0.019	p = 2.052 ± 0.0082
MDF = 1.428 ± 0.011	MDF = 1.967 ± 0.0038
mean = 1.374 ± 0.012	mean = 1.957 ± 0.0044
DP = 0.4255 ± 0.0027	DP = 0.2165 ± 0.0024
skewness = -0.8043 ± 0.013	skewness = -0.2647 ± 0.015
kurtosis = 1.149 ± 0.026	kurtosis = 0.1122 ± 0.024

Result of the component analysis:



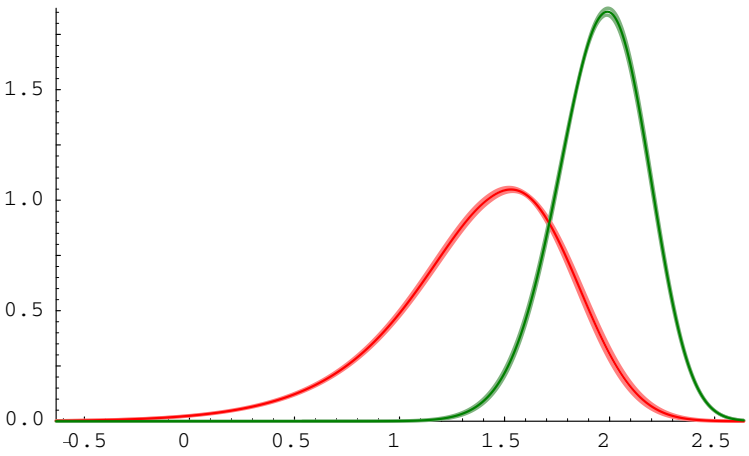
Total magnetization: 1.455 ± 0.029

a1 = 0.7989
MDF1 = 1.428 DP1 = 0.4255
sk1 = -0.8043 ku1 = 1.149

a2 = 0.6556
MDF2 = 1.967 DP2 = 0.2165
sk2 = -0.2647 ku2 = 0.1122

Calculating the confidence limits of each component. Please wait...

Normalized components with confidence limits:



Distribution parameters:

MDF1 = 1.428 DP1 = 0.4255
sk1 = -0.8043 ku1 = 1.149

MDF2 = 1.967 DP2 = 0.2165
sk2 = -0.2647 ku2 = 0.1122

```
Preparing data to an export format. Please wait...
```

Save results to a log file

```
Printing results to:  components.dat
```

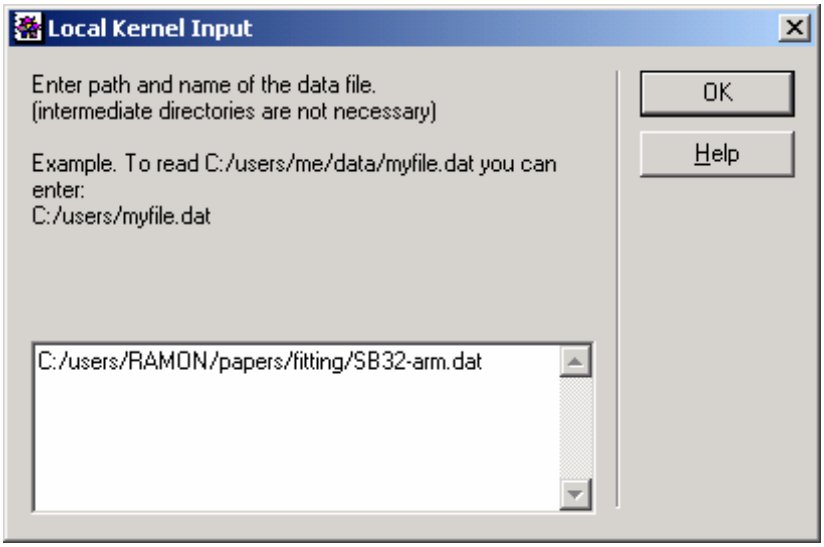
Save end-members

```
Saving the coercivity distributions to WDKarm.comp :
Column #1: magnetic field,
Column #2: component #1    Column #3: error of component #1
Column #4: component #2    Column #5: error of component #2
END
```

Loading CODICA and running GECA

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

Enter the name of the data file

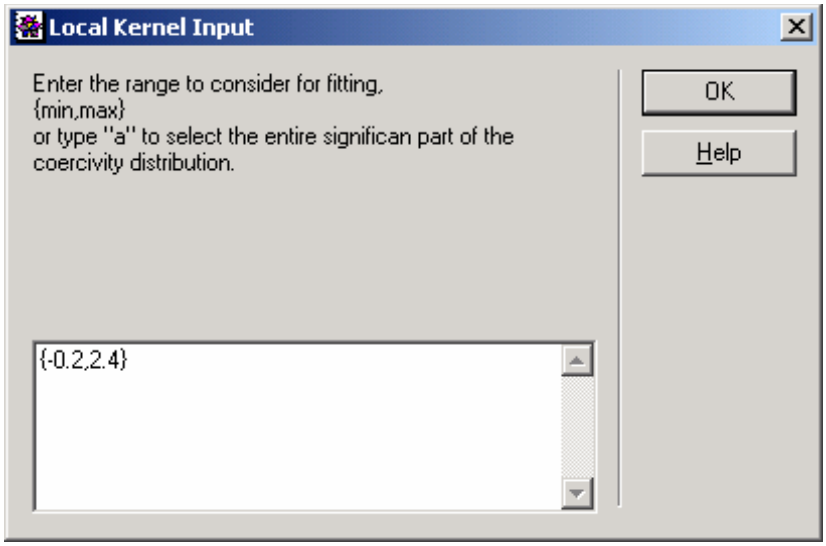


The prompt window on the right asks you to enter the name on the file which contains the coercivity distribution data. Type the path of the data file. You can skip intermediate directories if other files with the same name are not stored. The data file should be an ASCII file with three columns of numbers separated by spaces or tabulators. The file should not contain comment lines or text in general. The first column is the scaled or unscaled field, the second column is the value of the coercivity distribution for the corresponding field. The third column is the relative error of the second column; 0.1 means 10% error. Output files of CODICA with extensions `.slin`, `.slog` and `.spow` are automatically accepted. It is strongly recommended to run GECA only on CODICA output files with extension `.slog`.
Back to the program example

Plot the coercivity distribution

The coercivity distribution is plotted together with the confidence limits given by the error estimation stored in the file. If the maximal measurement error is less than 5% of the peak value of the coercivity distribution, only the confidence limit are plotted as a pair of gray lines. With errors larger than 5%, the coercivity distribution is plotted as a black line, together with the confidence limits. Within the plot, an estimation of the total magnetization is given. This estimation is obtained by integrating the coercivity distribution over the field range given by the data stored in the file. If saturation is not reached within this range, the calculated value is an underestimation of the total magnetization. You can use the estimation of the total magnetization as a reference when you enter the initial distribution parameters.
Back to the program example

Set the fitting range



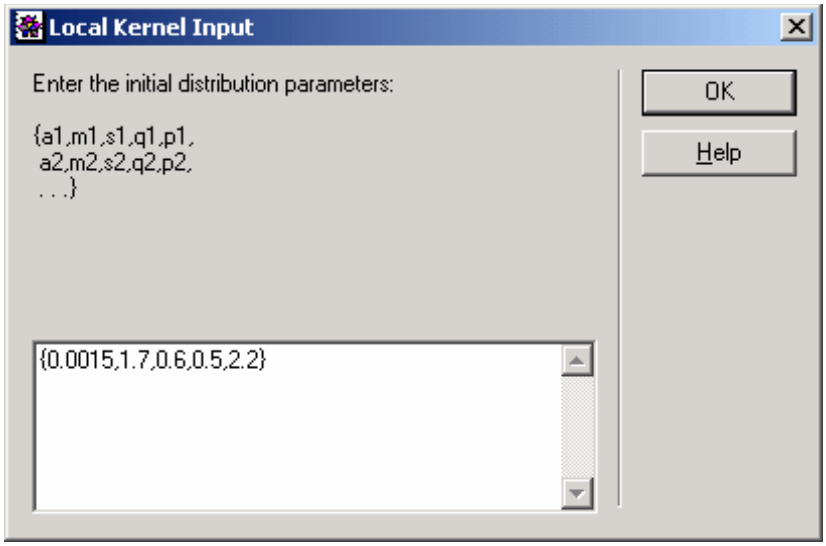
GECA estimates a field range where the values of the coercivity distribution are significant. As a significance limit, a maximum relative error of 50% has been chosen for the values of the coercivity distribution. You can enter a different range with the prompt window displayed on the right. If the coercivity distribution was calculated from a demagnetization curve, it is recommended to discard the data near the right end of the field range, because they could be affected by truncation effects. Data outside the range you entered are displayed but are not considered for further calculations.

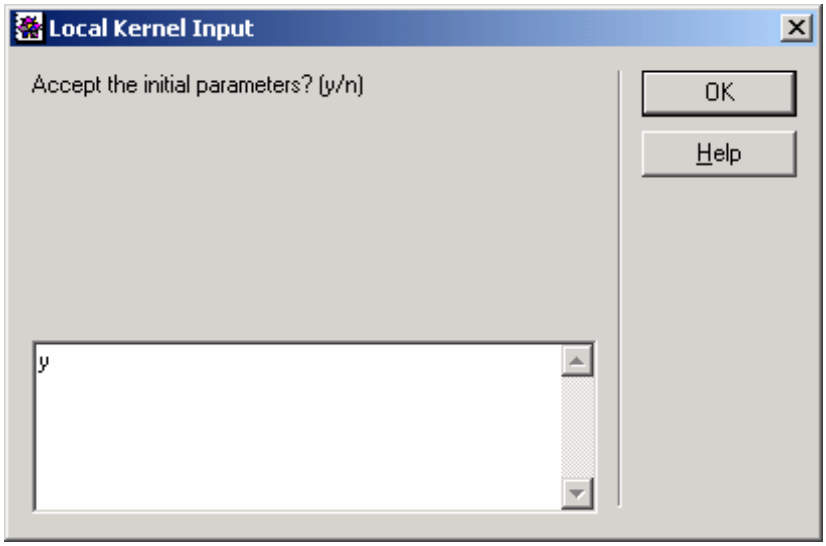
Back to the program example

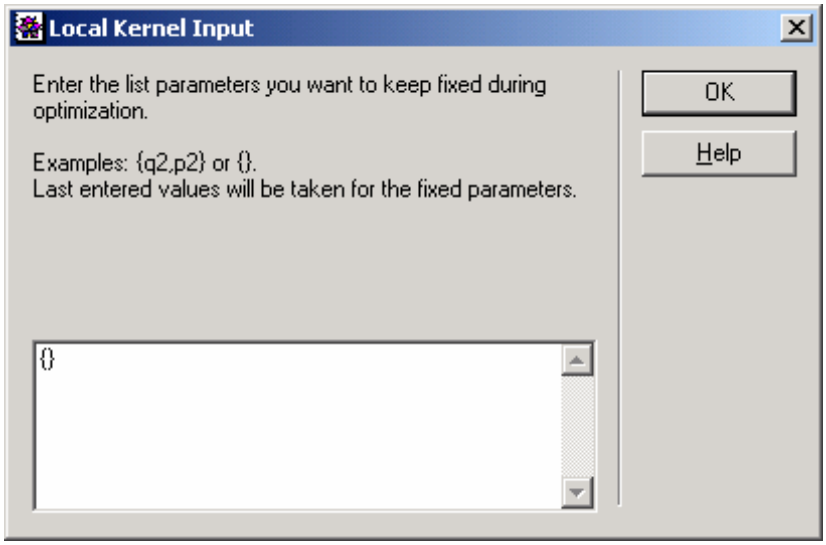
Enter initial distribution parameters

You are asked to enter initial values for the parameters of the finite mixture model that will be used for the component analysis. GECA uses a set of one to four SGG functions to fit the measured coercivity distribution. Each SGG function is characterized by following five parameters:

- *amplitude* a: the area under the SGG function, which is equivalent to the magnetization of a component whose coercivity distribution is represented by this function.
- *median* μ: this parameter corresponds to the median value of the function, also called median destructive field (MDF) or median acquisition field (MAF).
- *parameter for the standard deviation* σ: this is the principal parameter which controls the standard deviation of the SGG function, also called the dispersion parameter DP.
- *parameter for the skewness q*: this is the principal parameter which controls the skewness of a SGG function, with $-1 \leq q \leq 1$. Positive values of q generate left skewed functions, negative values of q generate right skewed functions. Symmetrical functions are characterized by $q = \pm 1$. Generally, real coercivity distributions are characterized by $0.5 \leq |q| \leq 1$. If you do not have independent informations about the starting parameters, set a value of q near 1.
- *parameter for the kurtosis p*: this is the principal parameter which controls the kurtosis of a SGG function. A logarithmic Gaussian distribution is characterized by $p = 2$. More squared functions are generated with $p > 2$, less squared distributions by $p < 2$. Common values for real coercivity distributions are given by $1.6 < p < 2.4$. If you do not have independent information about the starting parameters, set $p = 2$.







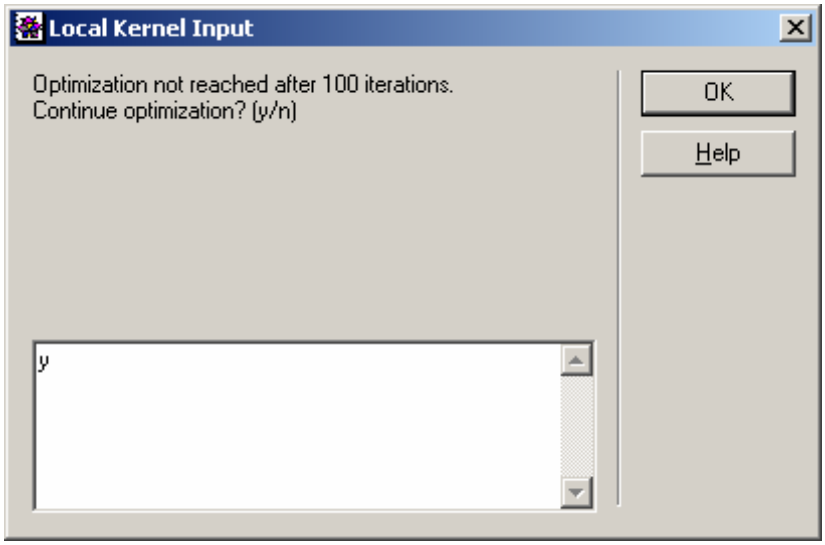
Enter the parameters as an ordered list: a, μ, σ, q and p of the first component, a, μ, σ, q and p of the second component, and so on, as in the example given in the prompt window shown to the right. The end-member distributions defined by the initial parameters you entered are plotted with different colors (red, green, violet and light blue). The modeled coercivity distribution is given by the sum of all end-members and is plotted in blue, together with the measured coercivity distribution (black/gray). The initial parameters should be chosen so, that the modeled coercivity distribution is as close to the measured coercivity distribution as possible. You can enter the initial parameters either with some knowledge about the magnetic components which are contributing to the measured distribution, or by try and error. In this last case you can reenter new initial parameters until you get a satisfactorily result. After entering the initial parameters, you are asked to keep some parameters fixed during the optimization. If you want to optimize all parameters, type "{}". Otherwise, enter the symbols for the fixed parameters in the next promt window. For example, if you want to use a logarithmic Gaussian function for the second end-member, set $q = 1$ and $p = 2$ as initial values for the corresponding SGG function, and keep these parameters fixed by entering "{q2,p2}" in the following prompt window. You can choose every combination of parameters to keep fixed. If you exactly know the parameters of one magnetic component, you can model this component by an end-member function with fixed values of μ, σ, q and p. Then, only the magnetic contribution of this component, given by a, will be optimized.

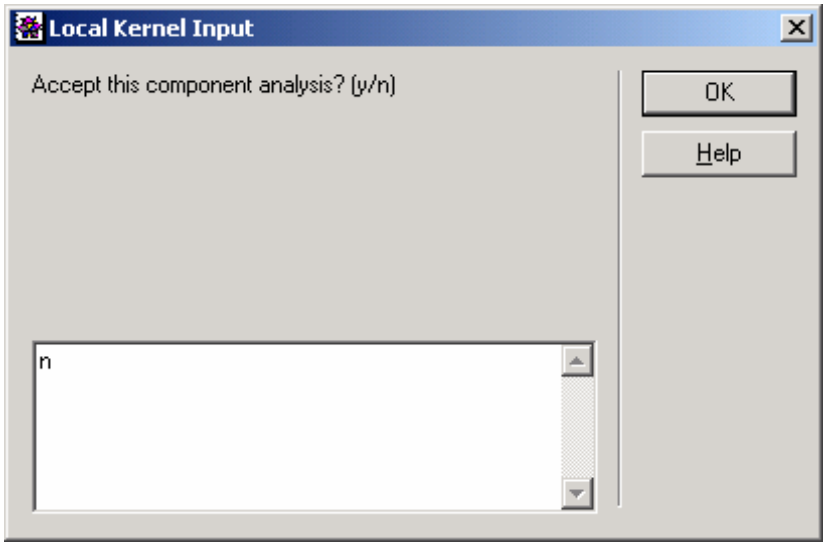
It is recommended to start with a small number of end-members and a small number of parameters, and to use independent information about the number of magnetic components and their properties. You can then progressively increase the complexity of your model. Keep in mind that the complexity of a model increases exponentially with the number of parameters to optimize. If you

want to perform a component analysis with three SGG functions, you have to deal with a solution space in 15 dimensions. You will not have the possibility to perform a systematic solution search is such a space: if you try 5 initial values for each parameter, you should perform $5^{15} = 3 \times 10^{10}$ optimizations! You would probably find several stable solutions, but only one among them is correct and has a physical meaning. The parameters have a hierarchical structure: a controls the amplitude of an end-member, μ the "position" along the field axis, and σ the "width", q the symmetry and p the curvature. The amplitude is the most important parameter, the curvature is the less important. You can start with fixed values of p, or fixed values of q and p. Use logarithmic Gaussian functions to model magnetic components which are not saturated in the field range of the measured coercivity distribution.
In the program example, the measured coercivity distribution is similar to an asymmetric unimodal probability density function. There is no direct evidence for more than one magnetic component. Therefore, initial parameters for one SGG function have been entered in the program example. Since only 5 parameters have to be optimized, the component analysis is relatively simple and only one stable solution is expected. Therefore, it is not necessary to start with a modeled coercivity distribution which is very close to the measured data.
Back to the program example

Perform a component analysis

GECA performs a component analysis by optimizing the initial parameters in a way that minimizes the squared residuals between model and measurements by using a Levenberg-Marquardt algorithm. The parameters to optimize are displayed together with the corresponding initial values. If the initial values were carefully chosen, the search for a solution is performed in a reasonable time with no more than 100 iterations. Otherwise, the search will take more than 100 iterations or it will converge to an absurd solution. If a global or a local minimum of the squared residuals is not reached within 100 iterations, a warning message appears and you will be asked to continue the search or stop it and plot the solution given by the last iteration. It is recommended to perform at least 200 iterations. The numerical values of the parameters are shown every 100 iterations and you can check how they change and if they converge to a meaningful result.





If a convergence to a stable solution cannot be obtained, interrupt the search for a solution by typing “n” in the prompt window and choose other initial parameters.

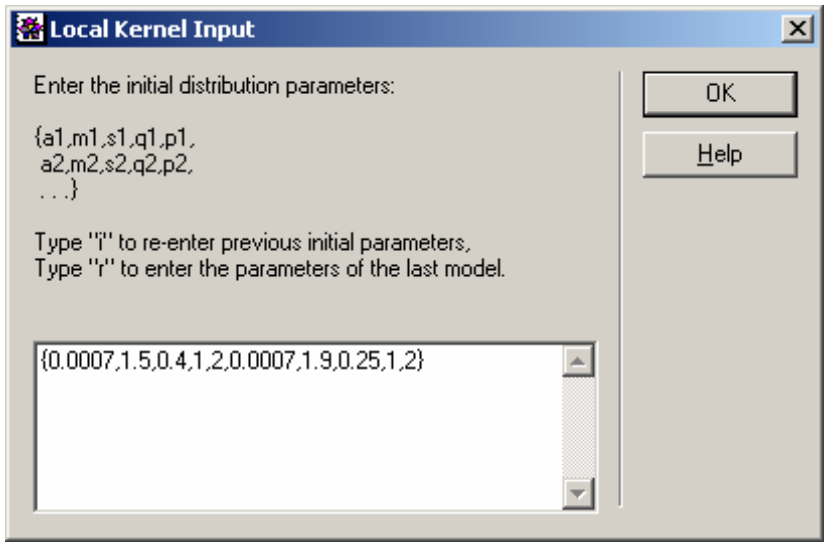
The result of the component analysis is displayed exactly like the initial model. The same colors are used to label the end-members. Additionally, the difference between the model and the measurements (blue line) is plotted below the result of the component analysis, together with the measurement errors (pair of gray lines). The difference between model and measurements (called *misfit* in the following) should be of the same order of magnitude as the estimated measurement error. If the misfit is much larger than the estimated measurement error, the model is not able to account for the measurements: other parameters should be added to reduce the misfit. If the misfit is much smaller than the measurement error, the model chosen for the component analysis is able to fit the measurements very well but it is not significant: some of the model parameters do not have any physical meaning. In this case you should decrease the number of model parameters by reducing the number of end-members or by keeping some parameters fixed. If the misfit has the same amplitude as the estimated measurement error, the model may be adequate. Nevertheless, more than one solution witch satisfy this condition may exist.

An adequate parameter to test the significance of a component analysis is the χ^2 statistics. GECA gives an estimation of χ^2/l, where l represents the degrees of freedom of the model. For a correct model, $\chi^2_{l;\alpha} \leq \chi^2 \leq \chi^2_{l;1-\alpha}$, where $\chi^2_{l;\alpha}$, $\chi^2_{l;1-\alpha}$ are the confidence limits at a given confidence level (usually 95%). GECA calculates the confidence limits with a 95% confidence level and displays them together with the estimation of χ^2/l. If $\chi^2 > \chi^2_{l;1-\alpha}$, the model is significantly different from the measured data, and GECA suggests you to refine it by adding more parameters. If $\chi^2 < \chi^2_{l;\alpha}$, not all model parameters are significant and a warning message is displayed. In this case you should reduce the number of parameters. If $\chi^2_{l;\alpha} \leq \chi^2 \leq \chi^2_{l;1-\alpha}$, GECA suggests to accept the model.

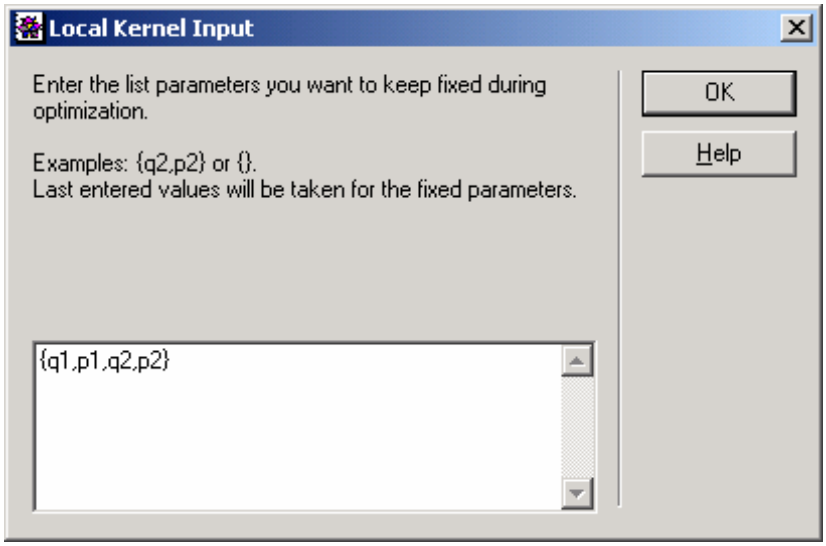
In the program example, the component analysis with one SGG function is inadequate to model the measurements within the given error margins.

Back to the program example

Enter initial distribution parameters (2 logarithmic Gaussian functions)



It is possible to run a new component analysis with different initial parameters. After obtaining your first result, the input prompt on the right asks you for one of the following actions: (1) type new initial parameters, (2) re-enter the initial parameters of the previous component analysis, (3) enter the result of previos component analysis as a set of initial parameters.
Since the component analysis with one component was not adequate, a more complex model with two logarithmic Gaussian functions is used. The initial values for skewness and kurtosis are setted to zero by entering $q = 1$ and $p = 2$ for both components. These parameters are kept fixed by entering "{q1,p1,q2,p2}" in the second prompt window. Initial parameters for a, μ and σ are guessed until a relatively good agreement with the measured data is obtained.
Back to the program example



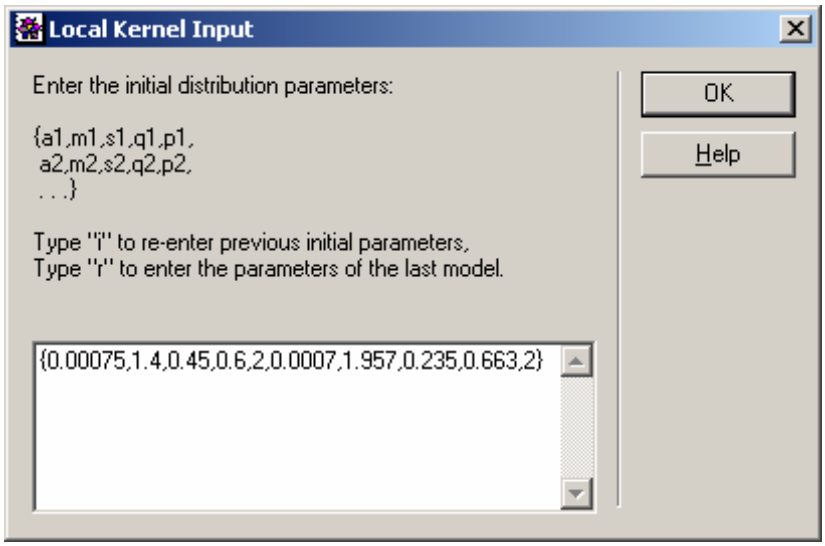
Perform a component analysis (2 logarithmic Gaussian functions)

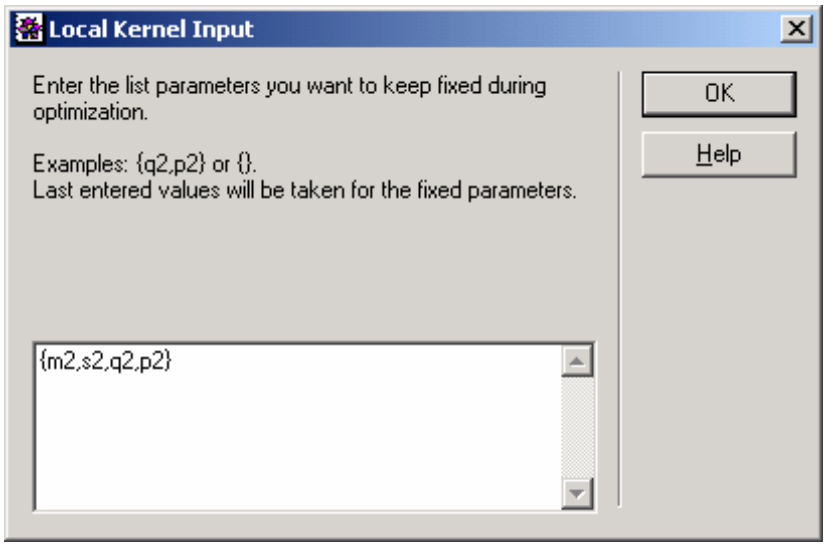
The component analysis with two logarithmic Gaussian functions is inadequate to model the measurements within the given error margins. The misfit between model and measurements is larger than that obtained with one SGG function, even if there is one more parameter to optimize. This example shows that logarithmic Gaussian functions are generally not suitable for modelling the coercivity distribution of single magnetic components. A good agreement between measurements and model is achieved only with 4 logarithmic Gaussian functions (Figure 3). Unfortunately, these functions cannot be related to the coercivity distributions or real magnetic components.
Back to the program example

Enter initial distribution parameters (one component is known)

The different sources of magnetic minerals for the sample taken as example are known from independent investigations on urban atmospheric dust samples collected in the same region. The two main sources are given by natural dust and by the products of combustion processes, mainly from motor vehicles and from waste incineration. The coercivity distribution of the combustion products can be modelled by a SGG function with $\mu = 1.96$, $\sigma = 0.235$, $q = 0.66$, $p = 2$. These parameters are kept fixed during the component analysis. Only the magnetization of the combustion products (given by a) is unknown and is optimized, together with the unknown parameters of natural dust.

Back to the program example





Perform a component analysis (one component is known)

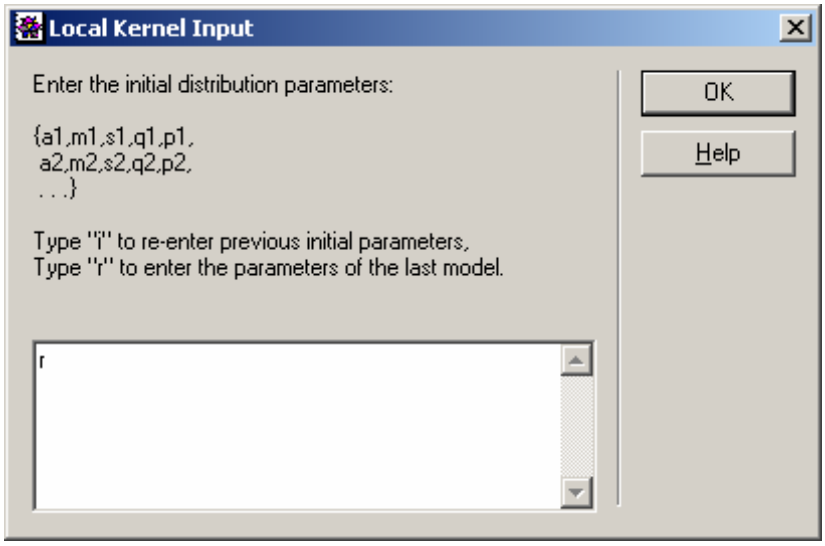
This component analysis is characterized by a much better agreement with the measurements, if compared to the previous results. Six distribution parameters have been optimized. The same number of parameters has been used to perform a component analysis with two logarithmic Gaussian functions: nevertheless, χ^2/l was almost one order of magnitude larger! This model is still significantly different from the measurements. A reason for that could arise from small variations in the properties of the same magnetic component collected from different places.

Back to the program example

Enter initial distribution parameters (2 SGG functions)

To take into account small variations in the magnetic properties of combustion products, the results of the previous component analysis are taken as initial values for a new component analysis where all 10 distribution parameters are optimized. This is done by typing “r” to recall the result of the last component analysis.

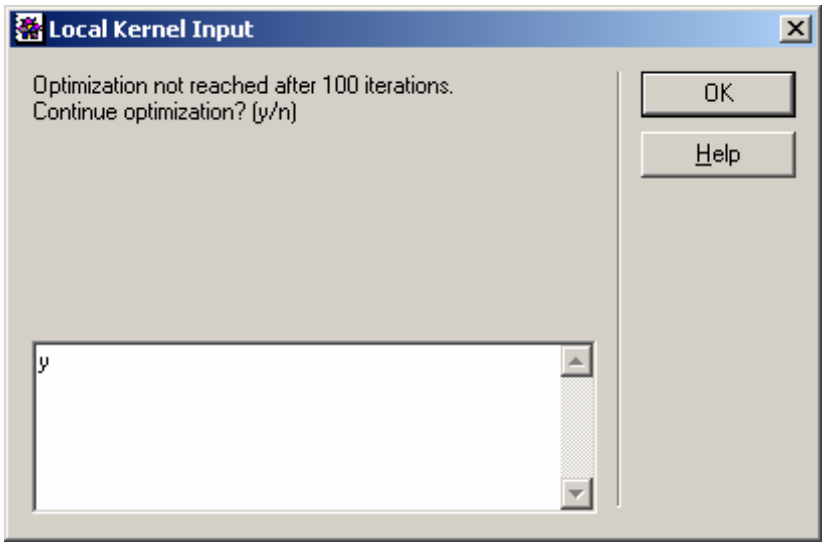
Back to the program example

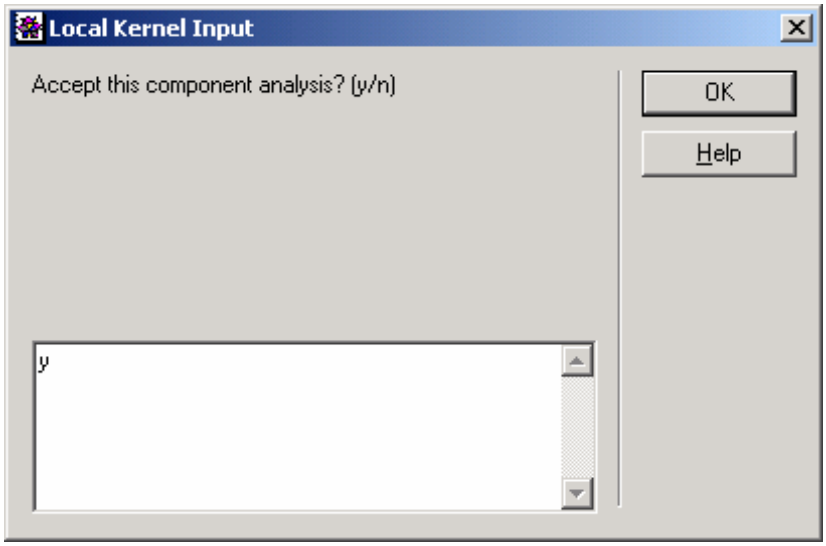


Perform a component analysis (2 SGG functions)

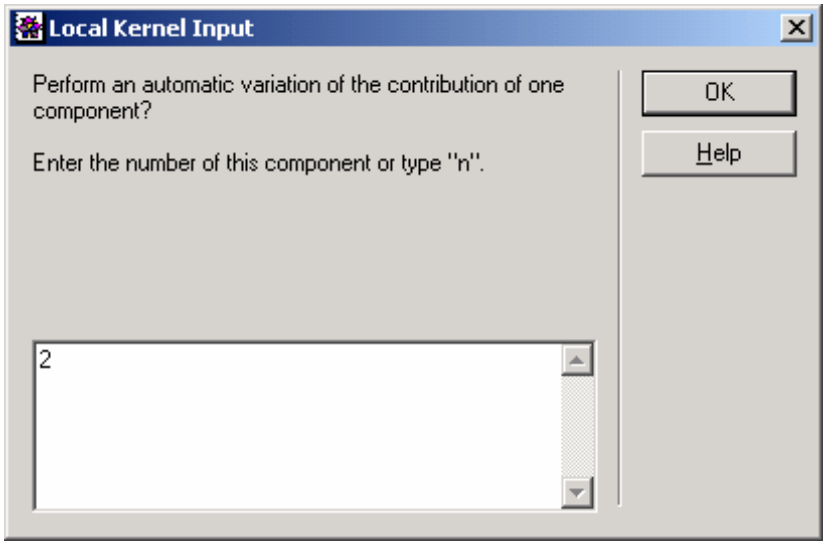
The model is now much more complex and the search for a stable solution needs more than 50 iterations. A warning message appears and you are asked to stop or continue for other 50 steps. Finally a stable solution is reached. The distribution parameters of the combustion product did not change more than 10% with respect to the initial values, and the model is now compatible with the measurements within the measurement errors. The solution of this component analysis is accepted, since it is compatible with the magnetic properties assumed initially for the combustion products.

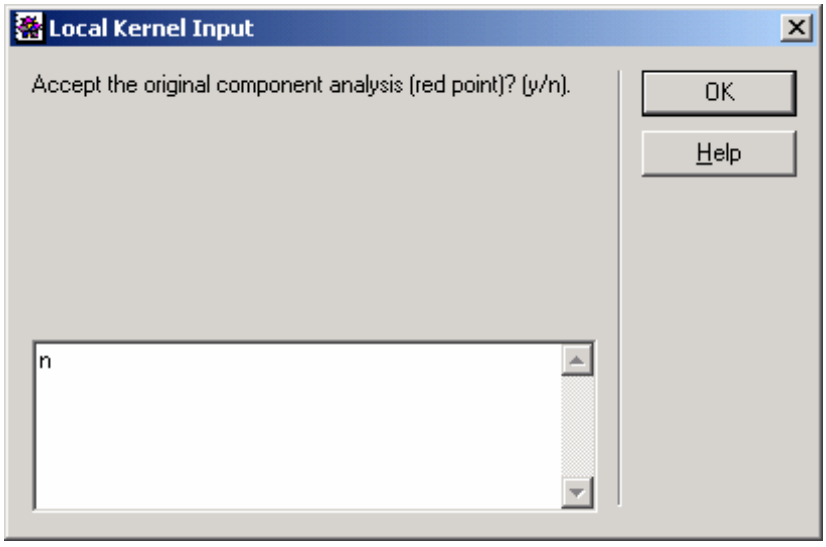
Back to the program example





Perform a systematic solution search

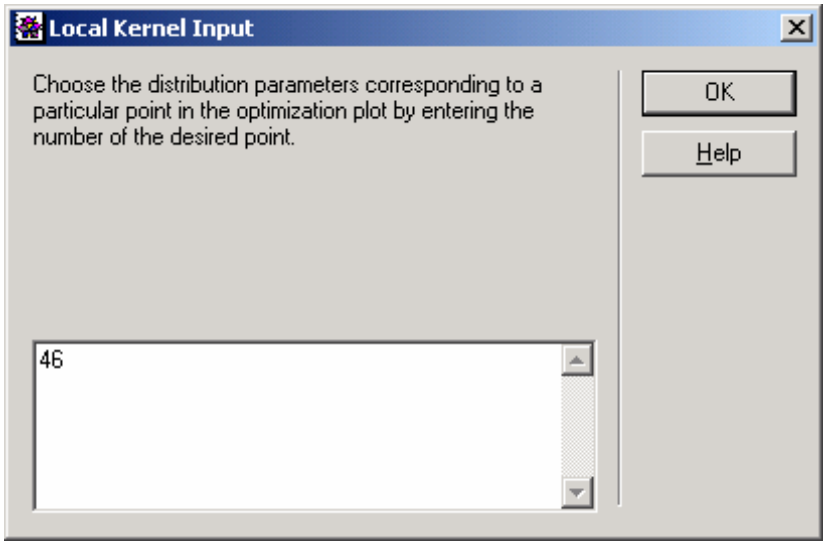




Stable solutions of the component analysis correspond to local minima of the merit function. The merit function can have several local minima for a given finite mixture model. One among them is a global minimum as well, and is usually considered as the acceptable solution. A solution which corresponds to a global minimum of the merit function is attained if the initial model is chosen to be close enough to the acceptable solution. Since this solution is usually unknown in advance, a sufficient number of initial models has to be tested in order to ensure that at least one will converge to a global minimum of the merit function. If you try 5 initial values for each parameter of a model with two SGG functions, you should perform $5^{10} = 10^7$ optimizations! GECA performs a selected search for a global minimum of the merit function, starting with the result of the last component analysis as initial model. You can select one end-member function, whose amplitude a will be increased and decreased in steps of 1/100 of the total sample magnetization, starting form the solution of the last component analysis. After each step, the new solution is taken as an initial model for the next component analysis. As a result, the merit function is plotted for all possible amplitudes of the selected SGG function. In the program example, the last solution of the component analysis (red point) is close to the global minimum of the merit function. The sharp steps of the merit function are an effect of the sudden convergence of some distribution parameters to a different local minimum. You are asked to accept the solution of the last component analysis, indicated by a red point, if it corresponds to a global minimum of the merit function.

Back to the program example

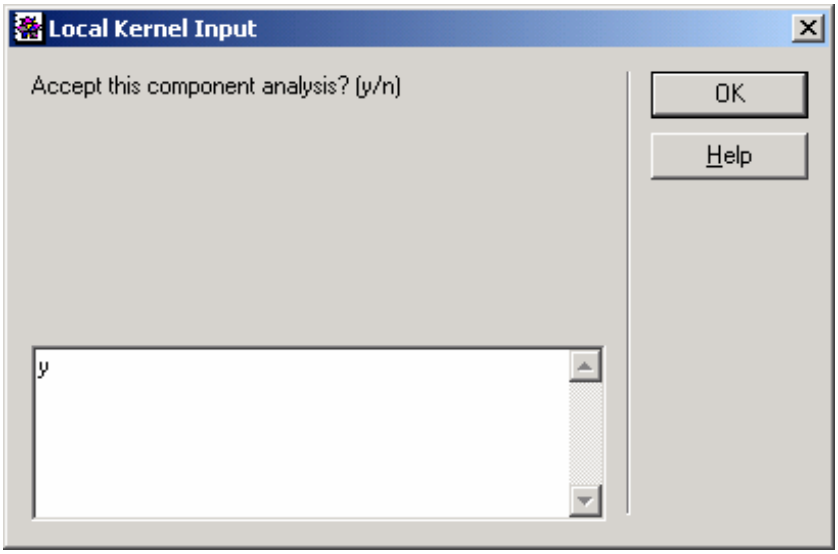
Choose initial distribution parameters from the systematic solution search



You can check the distribution parameters which corresponds to various values of the merit function previously plotted. The merit function was calculated for 100 points (black dots in the last plot, every 10th point is gray). You can enter the number of the point which corresponds to a particular value of the merit function you are interested in. In this way you can explore the solutions which correspond to various local minima and to the global minimum of the merit function. This option is particularly useful in the case that several local minima exist, which are close to the global minimum. Due to the measurement errors, a meaningful solution could be given by one of these local minima. You may evaluate different solutions with some independent information about the coercivity distribution expected for the individual magnetic components. In the program example, point number 46 is entered, which corresponds exactly to the global minimum of the merit function.

Back to the program example

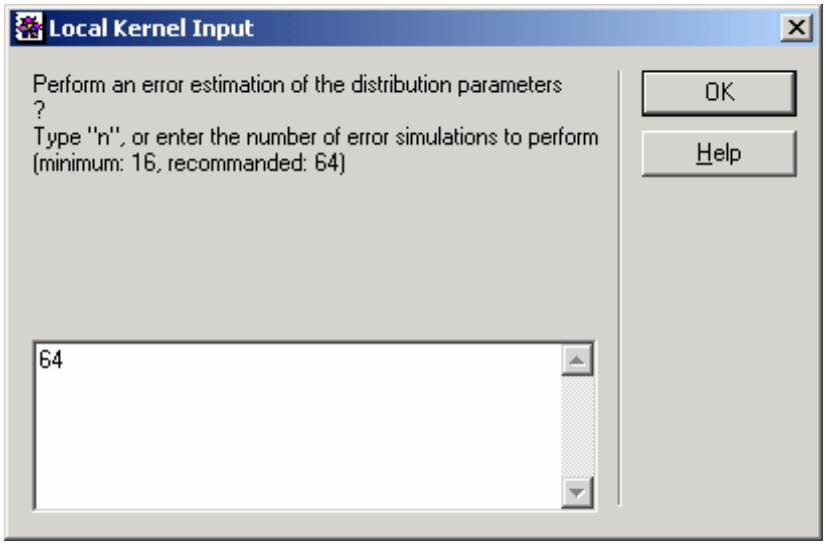
Perform a component analysis (representing a global minimum)



The set of distribution parameters which corresponds to a global minimum of the merit function is taken as initial model for the last component analysis. In the program example, this is the final solution, which represents a finite mixture model which is compatible with the measurements and with independent informations about the properties of the individual magnetic components. In other cases you may not accept this solution and enter a set of initial parameters which corresponds to other values of the merit function, until a satisfactory result is obtained.

Back to the program example

Perform an error estimation

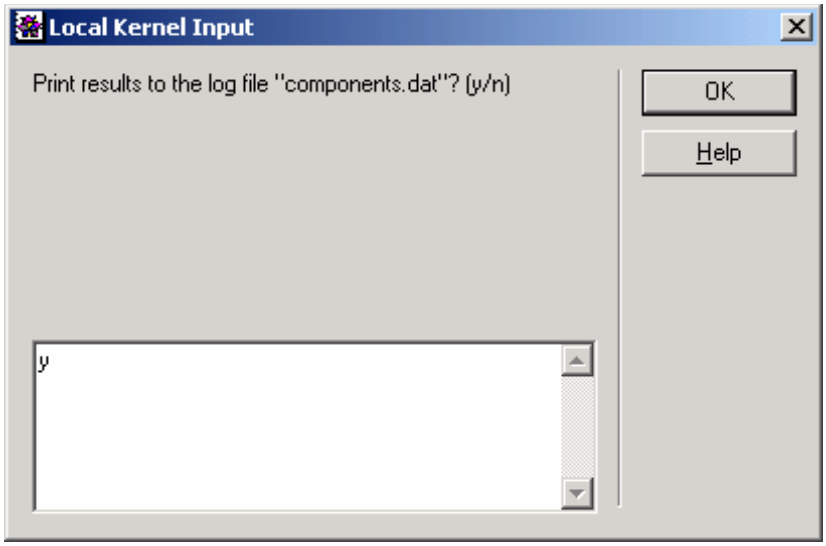


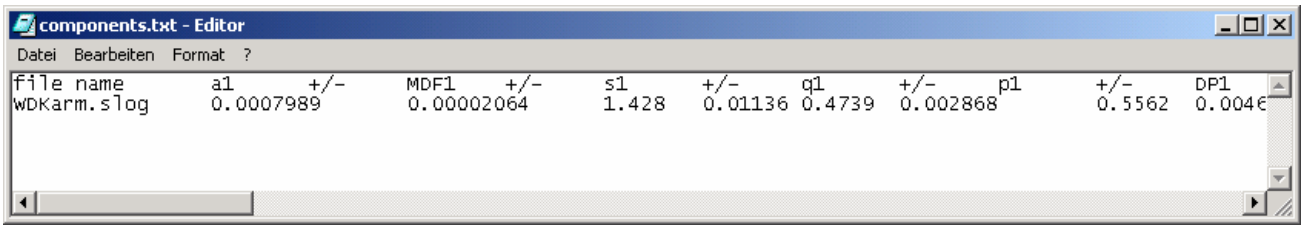
You can choose to perform an error estimation of the last component analysis. GECA will perform the error estimation by adding a random noise signal to the measured coercivity distribution. The standard deviation of the noise signal is chosen to be identical with the estimated measurement error for each value of the field. The new “noisy” coercivity distribution is fitted with the same set of end-member functions used for the last component analysis, whose solution is taken as the initial model. The result of the component analysis performed on the “noisy” coercivity distribution differs slightly from the result of the component analysis performed on the original coercivity distribution. The process of adding and adequate noise component to the original data and fitting the resulting coercivity distribution is repeated several times. GECA calculates the standard deviation of the component analysis results for each distribution parameter. These standard deviations are taken as an error estimation. At the same time, some statistical properties of the end-member functions are calculated as well, together with the related errors. The error estimation performed by GECA is quite time consuming, therefore you can choose the number of iterations to perform. With 64 iterations, an accuracy of 12% is expected for the error estimation. The relative accuracy of the error estimation, expressed in %, is given by $100/\sqrt{n}$, where n is the number of iterations used. The error estimation performed by GECA takes into account the effect of the measurement errors on a set of distribution parameters which is related to a particular local minimum of the merit function. The effect of measurement errors on the convergence of the component analysis to parameters which correspond to other local minima of the merit function is not considered. Therefore, the error estimation performed by GECA has to be considered as a lower limit for the real error of each parameter. Finally, all distribution parameters are displayed together with the estimated errors. Additionally, statistical parameters like the dispersion parameter DP, the mean, the skewness and the kurtosis are displayed with the related errors. The result of the component analysis is plotted again, together with the confidence limits of each end-member function. Finally, the normalized end-member distributions are plotted, together with their confidence limits. The area under the curve of each end-meber distribution is equal to one in this last plot.

Back to the program example

Save the component analysis results in a log file

You can save the component analysis results to the log file `components.txt`. You will find this file in the same folder where the program package CODICA is installed. The log file contains the results of all component analysis you decided to save, in form of a list of distribution parameters and statistics for each end-member distribution. The error estimation of each parameter is stored as well, if you decided to run an error estimation with GECA. An example of the content of the log file is displayed below.



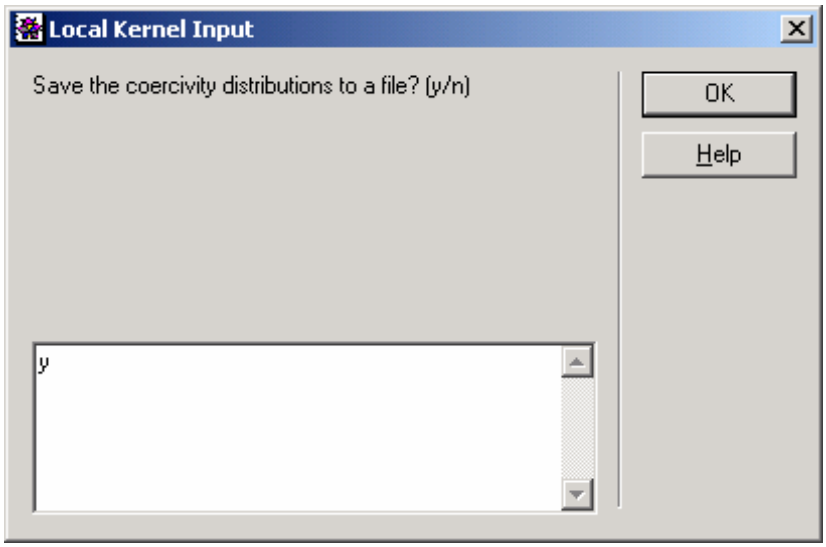


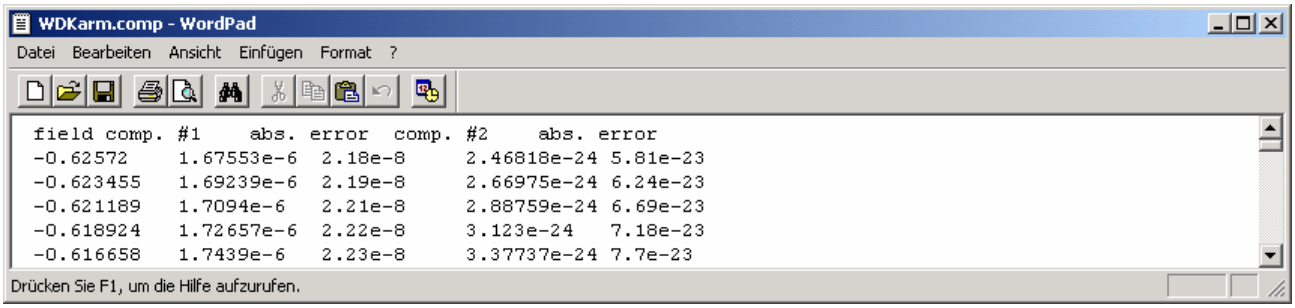
Back to the program example

Save the end-members to a file

The end-member distributions, together with their confidence limits, can be stored in a separated file as a list of columns with the numerical values of each function. The file will have the same name as the file where the original data for the coercivity distribution were stored, with extention `.cum`. This file will be stored in the same directory as the data file. An example is given below.

Back to the program example





Cautionary note

GECA 2.1 has been tested more than 500 times with coercivity distributions of various artificial and natural samples and the most different combinations of initial parameters. Nevertheless, there is a remote possibility that particular uncommon 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 GECA at any time just by typing "abort" in any input prompt window.
GECA 2.1 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 GECA, wich is expected to run on later versions. Use GECA 1.1 on previous versions of *Mathematica 5.0.*
In case of problems, write to the author (Ramon Egli) at the address given at the beginning of the source code file `Geca.m`. Please save and send a copy of the Mathematica session you were using when a problem arises, together with the data file you analyzed with GECA.

References

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

Heslop, D., M. J. Dekkers, P. P. Kruiver and I. M. H. Oorschot, Analysis of isothermal remanent magnetization acquisition curves using the expectation-maximization algorithm, *Geophys. J. Int.*, *148*, 58-64, 2002.

Kruiver, P. P., M. J. Dekkers, and D. Heslop, Quantification of magnetic coercivity components by the analysis of acquisition curves of isothermal remanent magnetization, *Earth Planet. Sci. Lett.*, *189*, 269-276, 2001.