# Users' guide to *FAC-PACK*

## A software for the computation
## of multi-component factorizations
## and the area of feasible solutions
### Software version 1.1

Mathias Sawall[a] and Klaus Neymeyr[a,b]

[a] Institut für Mathematik,Universität Rostock, Ulmenstrasse 69, 18057 Rostock, Germany.

[b] Leibniz-Institut für Katalyse, Albert-Einstein-Strasse 60, 18069 Rostock, Germany.

Email: $\alpha$ =mathias.sawall and $\beta$ =klaus.neymeyr with $\{\alpha$ or $\beta\}$@uni-rostock.de

February 19, 2014

*FAC-PACK* is an easy-to-use software for the computation of nonnegative multi-component factorizations and for the numerical approximation of the area of feasible solutions (AFS). Revision 1.1 contains new functionalities for the reduction of the rotational ambiguity.

Important features of **FAC-PACK** are:

- a fast C program with a graphical user interface (GUI) in MatLab,
- baseline correction of spectral data,
- computation of a low-rank approximation of the spectral data matrix and initial nonnegative matrix factorization,
- computation of the spectral and concentrational AFS by the polygon inflation method,
- applicable to two- and three-component systems,
- live-view mode and factor-locking for a visual exploration of all feasible solutions,
- reduction of the AFS by factor-locking,
- simultaneous representation of the spectral and concentrational AFS,
- reduction of the rotational ambiguity by the complementarity theorem.

# Contents

# 1 Quick start

Download the **FAC-PACK** software from

$$\text{http://www.math.uni-rostock.de/facpack/}$$

Then extract the software package facpack.zip, open a MATLAB desktop window and start facpack.m. The user can now select one of the three modules of **FAC-PACK**.

| | |
|---|---|
| **Baseline correction** | Baseline correction for a series of spectra. |
| **AFS computation** | Computation of multi-component factorizations and the area of feasible solutions by the polygon inflation algorithm. |
| **Complementarity & AFS** | Simultaneous representation of the spectral and concentrational AFS and reduction of the rotational ambiguity by the complementarity theorem. |

## 1.1 Baseline correction module

Step 1: If the baseline needs a correction (e.g., if a background subtraction has partially turned the series of spectra negative), then the **Baseline correction** button can be pressed. This simple correction method only works for spectral data in which frequency windows with a distorted baseline are clearly separated from the main signal groups of the spectrum; see the example below.
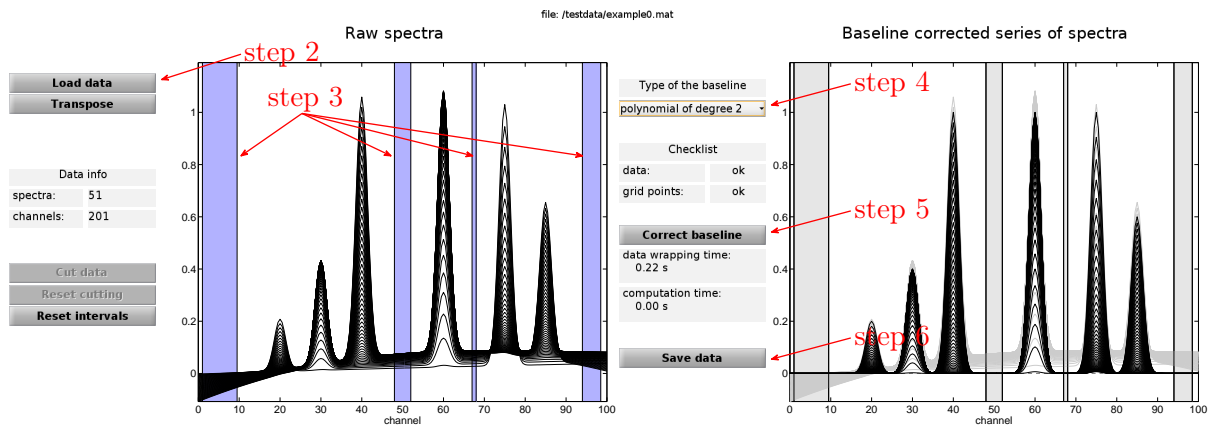


Step 2: Load the spectral data matrix. Sample data shown above is example0.

Step 3: One or more frequency windows should be selected in which more or less only the distorted background signal is present and which are clearly separated from the relevant spectral signals. In each of these frequency windows (marked blue) the series of spectra is corrected towards zero. However, the correction is applied to the full spectrum. In order to select these frequency windows click the left mouse button within the raw data window, move the mouse pointer and release the mouse button. This procedure can be repeated.

Step 4: Select the correction method, e.g. polynomial degree 2.

Step 5: Compute the corrected baseline.

Step 6: Save the corrected data.

## 1.2  AFS computation module

Step 1: Press the Button **AFS computation** in order to compute the AFS.
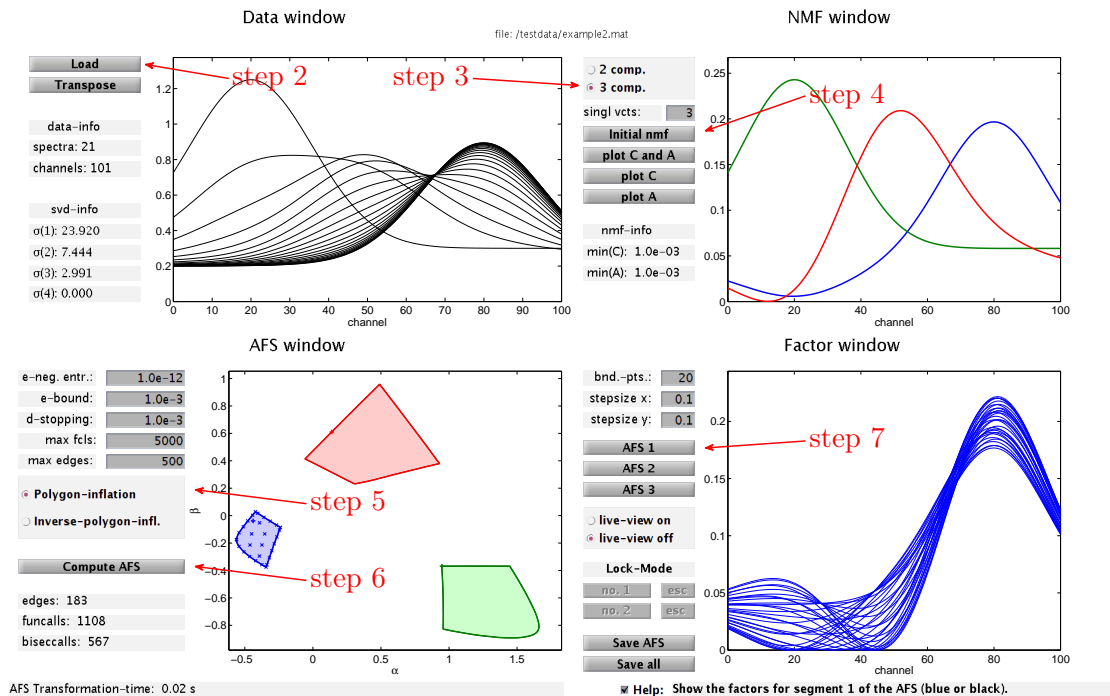


Step 2:  Select example2 as sample spectral data matrix.
Step 3:  Select 3 as the number of components.
Step 4:  Compute an initial nonnegative matrix factorization (NMF).
Step 5:  Choose Polygon inflation.
Step 6:  Compute the AFS which consists of three segments (a blue, red and green segment).
Step 7:  Plot the range of spectral factors associated with segment 1 (blue segment) of the AFS.

A second test problem:
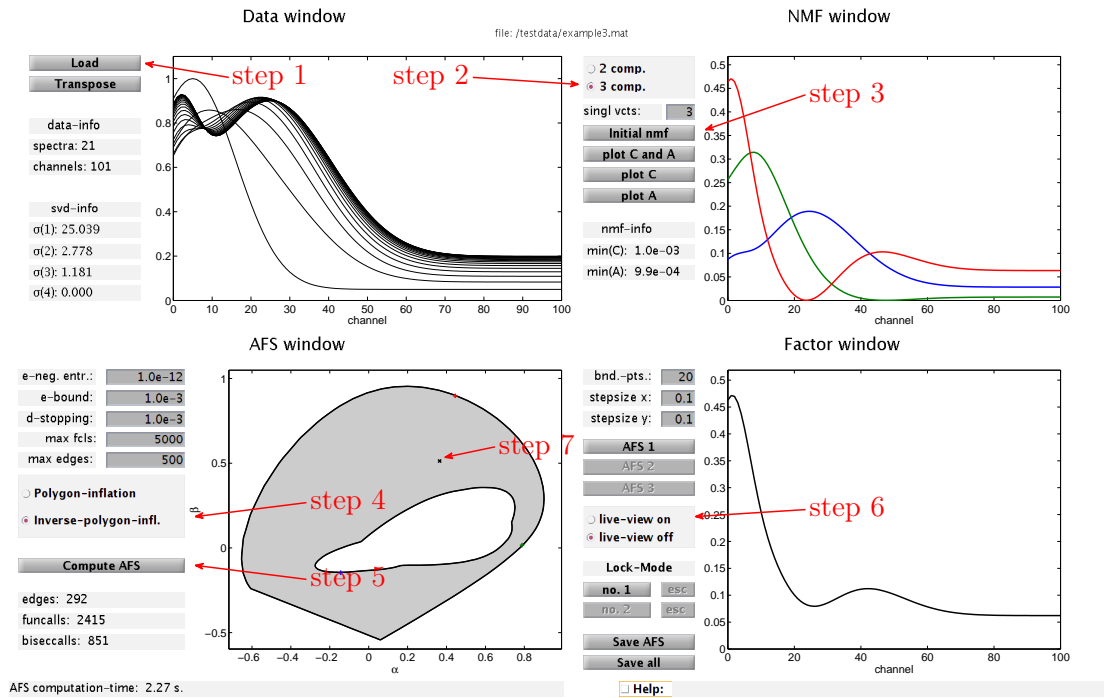


Figure 1: Quick start with live-view mode in 7 steps.

Step 1: Select example3 as the data matrix.
Step 2: Select 3 as the number of components.
Step 3: Compute an initial nonnegative matrix factorization (NMF).
Step 4: Choose Inverse polygon inflation. The AFS consists of only one segment with a hole.
Step 5: Compute the AFS.
Step 6: Select live-view on.
Step 7: Move the mouse pointer through the AFS and watch the interactively computed solutions.

## 1.3 Complementarity & AFS module

Step 1: Press the button **Complementarity & AFS** to activate this GUI.
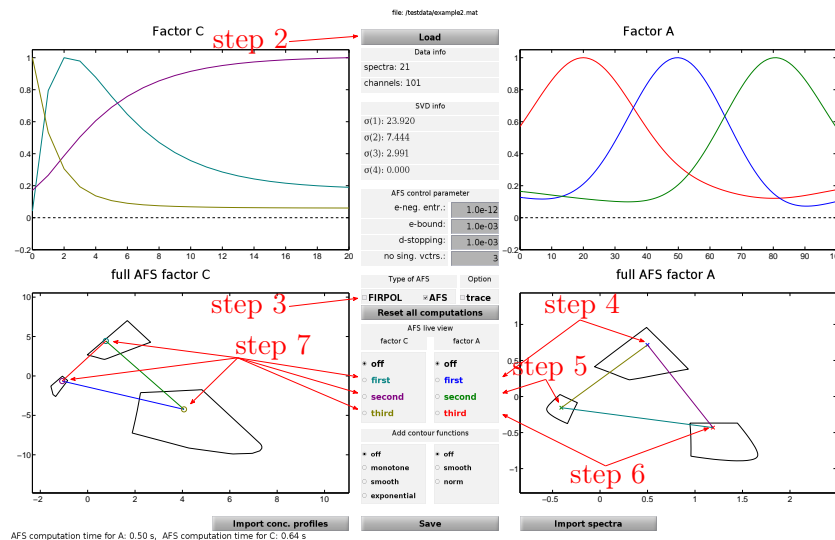


Step 2:  The sample data example2 can be selected.

Step 3:  Select FIRPOL to plot two supersets including the AFS for $C$ and $A$ and/or select full to show the AFS for the spectral and the AFS for concentrational factor. The sets FIRPOL can easily and quickly be computed; the computation of the AFS for $C$ and for $A$ may be time-consuming.

Step 4:  Click the button first and move the mouse pointer through the spectral AFS (factor $A$). A first spectrum $A(1,:)$ can be locked by clicking in the AFS. The associated spectrum is shown in the spectral factor window (right-upper window).

Step 5:  Click the button second and move the mouse pointer once again through the AFS for $A$. While moving the mouse pointer through the AFS a second spectrum $A(2,:)$ is shown in the spectral factor window together with the unique (by the complementarity theorem) concentration profile of the remaining third component. A certain second spectrum $A(2,:)$ can be locked by clicking in the AFS.

Step 6:  Finally, a third spectrum $A(3,:)$ can be selected by moving the mouse pointer through the spectral AFS. The resulting predictions on the spectral factor are shown interactively. This third spectrum can also be locked.

Step 7:  These last three buttons and also the buttons first, second, third can be clicked and then a spectrum or concentration profile can be modified by moving the mouse pointer through the AFS. This allows to modify the two triangles in the spectral and concentrational AFS which uniquely determine a feasible factorization of the given spectral data matrix.

If you like these appetizers you may want to read more in the following sections.

Enjoy working with **_FAC-PACK_**!

# 2 Introduction to *FAC-PACK*

*FAC-PACK* is a software for the computation of nonnegative multi-component factorizations and for the numerical approximation of the area of feasible solutions (AFS). Currently, the software can be applied to systems with $s = 2$ or $s = 3$ components.

Given a nonnegative matrix $D \in \mathbb{R}^{k \times n}$, which may even be perturbed in a way that some of its entries are slightly negative, a multivariate curve resolution (MCR) technique can be used to find nonnegative matrix factors $C \in \mathbb{R}^{k \times s}$ and $A \in \mathbb{R}^{s \times n}$ so that

$$D \approx CA. \tag{1}$$

Some references on MCR techniques are [6, 9–11, 14]. Typically the factorization (1) does not result in unique nonnegative matrix factors $C$ and $A$. Instead a continuum of possible solutions exists [7, 11, 21]; this non-uniqueness is called the rotational ambiguity of MCR solutions. Sometimes additional information can be used to reduce this rotational ambiguity, see [8, 15] for the use of kinetic models.

The most rigorous approach is to compute the complete continuum of nonnegative matrix factors $(C, A)$ which satisfy (1). In 1985 Borgen and Kowalski [4] found an approach for a low dimensional representation of this continuum of solutions by the so-called area of feasible solutions (AFS). For instance, for a three-component system ($s = 3$) the AFS is a two-dimensional set. Further references on the AFS are [1, 13, 20, 21]. For a numerical computation of the AFS two methods have been developed: the triangle-boundary-enclosure scheme [5] and the polygon inflation method [17]. *FAC-PACK* uses the polygon inflation method.

*FAC-PACK* works as follows: First the data matrix $D$ is loaded. The singular value decomposition (SVD) is used to determine the number $s$ of independent components underlying the spectral data in $D$. *FAC-PACK* can be applied to systems with $s = 2$ or $s = 3$ predominating components. Noisy data is not really a problem for the algorithm as far as the SVD is successful in separating the characteristic system data (larger singular values and the associated singular vectors) from the smaller noise-dependent singular values. Then the SVD is used to compute a low rank approximation of $D$. After this an initial nonnegative matrix factorization (NMF) is computed from the low rank approximation of $D$. This NMF is the starting point for the polygon inflation algorithm since it supplies two or three points within the AFS. From these points an initial polygon can be constructed, which is a first coarse approximation of the AFS. The initial polygon is inflated to the AFS by means of an adaptive algorithm. This algorithm allows to compute all three segments of an AFS separately. Sometimes the AFS is a single topologically connected set with a hole. Then an "inverse polygon inflation" scheme is applied. The program allows to compute from the AFS the continuum of admissible spectra. The concentration profiles can be computed if the whole algorithm is applied to the transposed data matrix $D^T$. Alternatively, the spectral and the concentrational AFS can be computed simultaneously within the "Complementarity & AFS" graphical user interface (GUI).

*FAC-PACK* provides a live-view mode which allows an interactive visual inspection of the spectra (or concentration profiles) while moving the mouse pointer through the AFS. Within the live-view mode the user might want to lock a certain point of the AFS, for instance, if a known spectrum has been found. Then a reduced and smaller AFS can

be computed, which makes use of the fact that one spectrum is locked. For a three-component system this locking-and-AFS-reduction can be applied to a second point of the AFS.

Within the GUI "Complementarity & AFS" the user can explore the complete factorization $D = CA$ simultaneously in the spectral and the concentrational AFS. The factorization $D = CA$ is represented by two triangles. The vertices of these triangles can be moved through the AFS and can be locked to appropriate solutions. During this procedure the program always shows the associated concentration profiles and spectra for all components. In this way **FAC-PACK** gives the user a complete and visual control on the factorization. It is even possible to import certain pure component spectra or concentration profiles in order to support this selection-and-AFS-reduction process within the "Complementarity & AFS" GUI.

# 3 Get ready to start

Please download **FAC-PACK** from

$$\text{http://www.math.uni-rostock.de/facpack/}$$

Then extract the file facpack.zip. Open a MATLAB desktop and run facpack.m.

This software is shareware that can be used by private and scientific users. In all other cases (e.g. commercial use) please contact the authors. We cannot guarantee that the software is free of errors and that it can successfully be used for a particular purpose.

## 3.1 Program structure

The current revision of **FAC-PACK** consists of three graphical user interface (GUI) windows to solve the following problems in multivariate curve resolution:

1. Correction of the baseline for given spectroscopic data or pre-processed data, e.g., after background subtraction,

2. Computation of the area of feasible solutions for two- and three-component systems,

3. Simultaneous representation of the spectral and concentrational AFS and interactive reduction of the rotational ambiguity up to uniqueness by means of the complementarity theorem [16, 19].

Once **FAC-PACK** is started it is necessary to select one of the three GUIs "Baseline correction", "AFS computation" or "Complementarity & AFS". No additional MATLAB toolboxes are required to run the software.

### 3.1.1 Baseline correction

This GUI has the following two windows:

1. Raw data window (Left window): The raw data window shows the $k$ raw spectra which are the rows of the data matrix $D \in \mathbb{R}^{k \times n}$.

2. Baseline corrected data window (Right window): This window shows the control parameters for the baseline computation together with the corrected spectra. Seven simple functions (polynomials, Gaussian- and Lorentz curves) can be used to approximate the baseline.

### 3.1.2 AFS computation

The GUI uses four windows:

1. Data window (Upper-left window): The data window shows the $k$ rows (spectra) of the data matrix $D \in \mathbb{R}^{k \times n}$ in a 2D-plot. The number of spectra $k$ is printed together with the number of spectral channels $n$. The four largest singular values of $D$ are shown. By transposing the data matrix $D$ it is possible to compute the AFS for the concentration factor $C$ (instead of the AFS for the spectral factor $A$).
2. NMF window (Upper-right window): This window allows to set the number of components to $s = 2$ or $s = 3$ and to compute an initial NMF. The smallest minimal components of $C$ and $A$ are printed. The figure shows the profiles of the so-called abstract matrix factors. The buttons allow to compute the profiles of $C$ and/or $A$.
3. AFS window (Lower-left window): Various control parameters allow to make certain settings for the AFS computation in order to control the approximation quality or maximal number of edges. Pressing the Compute AFS button starts the AFS computation. The live-view mode is active after the AFS has been drawn. Just move the mouse pointer to (and through) the AFS.
4. Factor window (Lower-right window): This window shows the spectral factors (or concentration profiles if the transpose option has been activated in the first window) for the grid points shown in the AFS window.

### 3.1.3 Complementarity & AFS

The GUI is built around four windows and a central control bar:

1. Pure factor windows (Upper row): These windows show three concentration profiles and three spectra of a factorization $D = CA$ which is computed by a step-wise reduction of the rotational ambiguity for a three-component system.
2. $C$, $A$ - AFS windows (Lower row): Here FIRPOL, a superset which includes the AFS, and/or the AFS can be shown for the factors $C$ and $A$. A feasible factorization $D = CA$ is associated with two triangles whose vertices represent the concentration profiles and spectra.
3. Control bar (Center): Includes all control parameters for the computations, the live-view mode, the import of pure component spectra or concentration profiles as well as the control buttons for adding contour plots on certain soft constraints.

## 3.2 Importing initial data

Spectral data are imported to **FAC-PACK** by means of a MAT-file *.mat. This file must contain the matrix $D \in \mathbb{R}^{k \times n}$ whose rows are the $k$ spectra. Each spectrum contains absorption values at $n$ frequencies. The file may also contain a vector $x$ with $n$ components representing the spectral wavenumbers/frequencies and a time-coordinates vector $t \in \mathbb{R}^k$.

The spectral data matrix $D$ is loaded by pressing one of the buttons (1)[1], (18) or (64) depending on the active GUI. The dimension parameters $k$ and $n$ are shown in the GUI. The four largest singular values are shown in the fields (22)-(25) or (71)-(74). These singular values allow to determine the numerical rank of the data matrix and are the basis for assigning the number of components $s$.

## 3.3 Sample data

**FAC-PACK** provides some sample data matrices:

| | |
|---|---|
| example0.mat: | $k = 51$ spectra, $n = 201$ channels, $s = 3$ components with a baseline turning negative to demonstrate the baseline correction. |
| example1.mat: | $k = 51$ spectra, $n = 101$ channels, $s = 2$ components. |
| example2.mat: | $k = 21$ spectra, $n = 101$ channels, $s = 3$ components, the AFS has three clearly separated segments. |
| example3.mat: | $k = 21$ spectra, $n = 101$ channels, $s = 3$ components, the AFS is one topologically connected set with a hole. |
| example4.mat: | some random noise has been added to data matrix $D$ given in example2.mat, the AFS has three clearly separated segments. |

## 3.4 External C-routine

All time-consuming numerical computations are externalized to a C-routine in order to accelerate **FAC-PACK**. This C-routine is AFScomputationSYSTEMNAME.exe, wherein SYSTEMNAME stands for your system including the bit-version. For example AFScomputationWINDOWS64.exe is used on a 64 bit Windows system. The external routine is called if any of the buttons **Initial nmf** (30), **Compute AFS** (43), **no. 1** (57) or **no. 2** (59) is pressed. Pre-compiled versions of the $C$-routine for the following systems are parts of the distribution:

- Windows 32/64 bit,
- Unix 32/64 bit and
- Mac 64 bit.

The execution of the external routine can always be stopped by CTRL + C in the command window. If the C-routine takes too much computation time, the reason for this can be large values for $k$, $n$, max fcls (40) or max edges (41) or too small values for $\varepsilon$-bound (38) or $\delta$-stopping (39).

---

[1]Red numbers refer to Section 7.

## 3.5 Further included libraries

The C-routine AFScomputationSYSTEMNAME.exe includes the netlib library lapack and
the ACM routine nl2sol. Any use of **FAC-PACK** must respect
the lapack license, see http://www.netlib.org/lapack/LICENSE.txt, and
the nl2sol license, see http://www.acm.org/publications/policies/softwarecrnotice.

# 4 Baseline correction module

AFS computations depend sensitively on distorted baselines and negative components in the spectral data. Such perturbing signals can pretend further components in the reaction system and negative components are not consistent with the nonnegative factorization problem. The **Baseline correction** GUI tries to correct the baseline and is accessible by pressing the button **Baseline correction** on the *FAC-PACK* start window, see a screen shot on page 33. Data loading is explained in Section 3.2. The **Transpose** button (2) serves to transpose the data matrix $D$.

The **Baseline correction** GUI is a simple tool for spectral data preprocessing. It can be applied to series of spectra in which the distorted baseline in some frequency windows is well separated from the relevant signal. We call such frequency intervals, which more or less show the baseline, zero-intervals. The idea is to remove the disturbing baseline from the series of spectra by adapting a global correction function within in the zero-intervals to the baseline. See Figure 2 where four zero-intervals have been marked by blue columns. The zero-intervals are selected in the raw data window (8) by clicking in the window, dragging a blue zero-interval and releasing the mouse button. The procedure can be repeated in order to define multiple zero-intervals. The **Reset intervals** button (7) can be used in order to reset the selection of all zero-intervals.
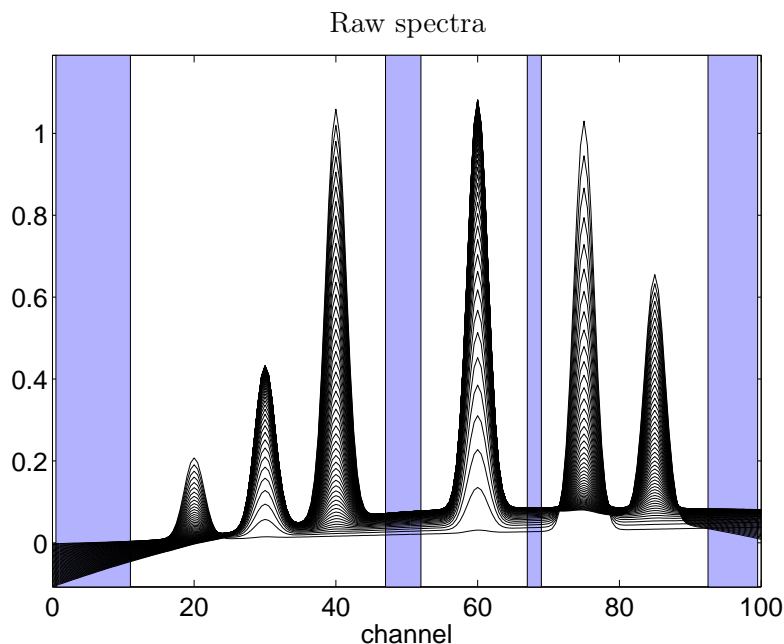


Figure 2: Four zero-intervals are marked by blue columns. In these intervals the spectral data is corrected towards zero. Data: example0.mat.

Mathematically every spectrum is treated separately. Let $a \in \mathbb{R}^n$ be a certain spectrum. Then the aim is to compute a baseline function $b \in \mathbb{R}^n$ so that the norm of the residuum $a - b$ is minimal within the selected intervals. The baseline-corrected spectrum is then $a_{new} = a - b$. Once again, this explains the naming zero-intervals, since there the spectrum is assumed to be zero.

The sum of the interval lengths of these zero-intervals should be as large as possible in order to ensure a reliable baseline approximation. However, the required number of frequency grid points depends on the degrees of freedom of the type of the baseline function. The text field (11) shows the information "ok"/"not ok" in order to indicate whether or not the selected intervals are large enough for the baseline approximation. Data cutting is a further functionality of the baseline correction GUI: If only a certain frequency subinterval is to be analyzed or the baseline correction is only to be applied to a subinterval, then this subinterval can be marked by mouse clicking and dragging and then the **Cut data**-button can be pressed. The user can always return to the initial data by pressing the **Reset cutting**-button (6).

## 4.1  Type of the baseline

Four types of baselines are available:

- Polynomials of order zero up to order four,
- a Gauss curve,
- or a Lorentz curve.

The most recommended baselines are polynomials with the degrees 0, 1 or 2. A polynomial of degree 0 is a constant function so that a constant is added or subtracted for each spectral channel of a spectrum.

The baselines "Gauss curve" and "Lorentz curve" can be used to remove single isolated peaks from the series of spectra. Once again, the curve profile is fitted in the least-squares sense to the spectroscopic data within the selected frequency interval. Then the fitted profile is subtracted from the spectrum.

## 4.2  Program execution and data export

The baseline correction by an external C-procedure is started by pressing the **Correct baseline**-button (12). The computation times for data wrapping and for solving the least-squares problem are shown in the text fields (13) and (14). The corrected series of spectra (in black) is shown in the window (16) together with the original data (in gray). The baseline-corrected data can be stored in a MATLAB file by pressing the **Save**-button (15).

By clicking the right mouse button in a figure a separate MATLAB figure opens. The figure can now be modified, printed or exported in the usual way. If for some reason (e.g. program runs too long) the external C-procedure for the baseline correction is to be stopped, then CTRL + C only works the MATLAB command window (and not in the GUI).
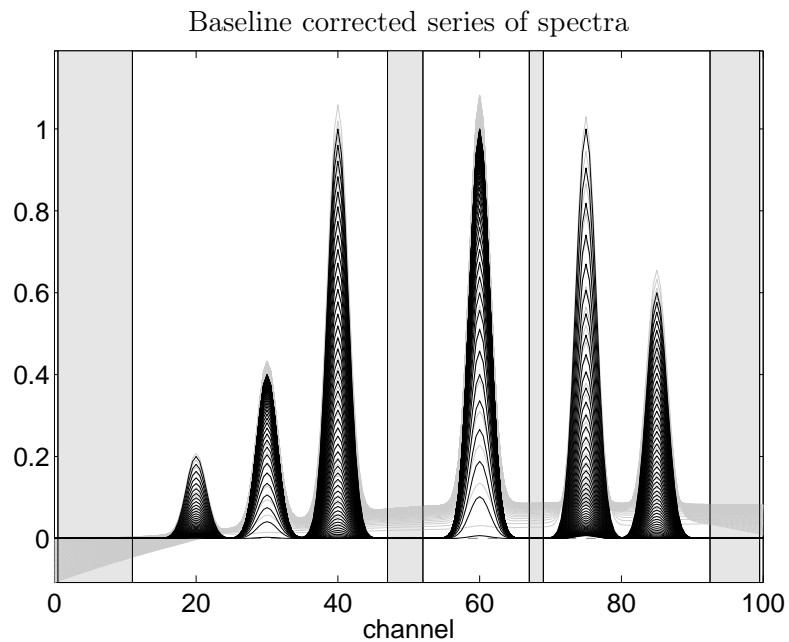
Figure 3: Sample problem example0.mat with corrected baseline by a 2nd order polynomial.

# 5 AFS computation module

This section describes how to compute the area of feasible solutions (AFS) for two- and three-component systems with *FAC-PACK*.

## 5.1 Initial steps

The AFS computation module can be started by pressing the **AFS computation** button on the *FAC-PACK* start window. Data loading by the **Load** button (18) is explained in Section 3.2. The loading process is followed by a singular value decomposition (SVD) of the data matrix. The computing time is printed in the field **Computing time info** (47). The dimensions of $D$ are shown in the data window together with the singular values $\sigma_1, \ldots, \sigma_4$. These singular values allow to determine the "numerical rank" of the data matrix and are the basis for assigning the number of components $s$ in the NMF window.

*FAC-PACK* provides some sample data matrices. These sample data sets are introduced in Section 3.3. Figure 4 shows the series of spectra from example2.mat. Three clearly nonnegative singular values indicate a three-component system.

## 5.2 Transposing the problem

*FAC-PACK* computes and displays the spectral matrix factor $A$, see (1), together with its AFS. In order to compute the first matrix factor $C$ and its AFS, press the **Transpose** button (19) to transpose the matrix $D$ and to interchange $x$ and $t$.

## 5.3 Initial NMF

To run the polygon inflation algorithm an initial NMF is required. First the number of components (either $s = 2$ or $s = 3$) is to be assigned. In the case of noisy data additional singular vectors can be used for the decomposition by selecting a larger number of **singl vcts** (29). For details on this option see [12], wherein the variable $z$ equals **singl vcts**.

The initial NMF is computed by pressing the button **Initial nmf** (30). The NMF uses a genetic algorithm and a least-squares minimizer. The smallest relative entries in the columns of $C$ and rows of $A$, see [17] for the normalization of the columns of $C$ and rows of $A$, are also shown in the NMF window, see (34) and (35). These quantities are used to define an appropriate noise-level $\varepsilon$; see Equation (6) in [17] for details.

The initial NMF provides a number of $s$ interior points of the AFS; see Equations (3) and (5) in [17].

The buttons **plot C and A** (31), **plot C** (32) and **plot A** (33) serve to display the factors $C$ and $A$ together or separately. Note that these abstract factors are associated with the current NMF. After an NMF computation the $A$ factor is displayed.

Figure 5 shows the factor $A$ for an NMF for example2.mat. The relative minimal components in both factors are greater than zero ($9.9 \cdot 10^{-4}$ and $10^{-3}$). So the noise-level **e-neg. entr.:** (37) can be set to the lower bound **e-neg. entr.:** $1 \cdot 10^{-12}$.
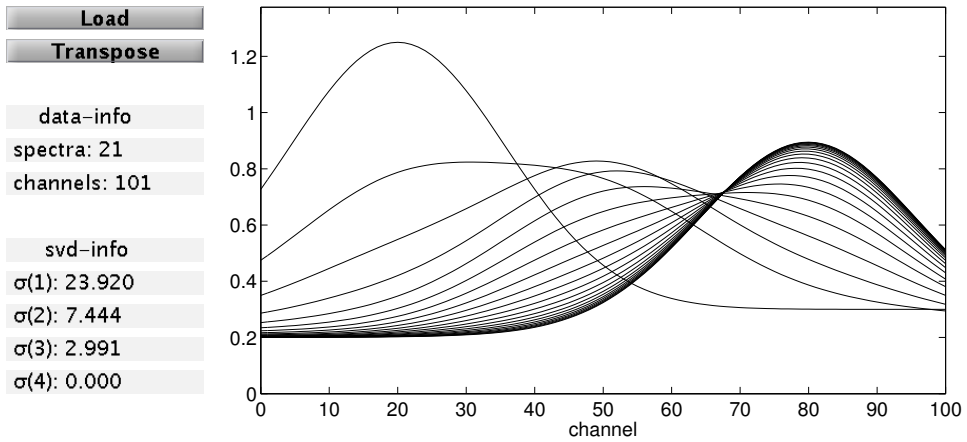
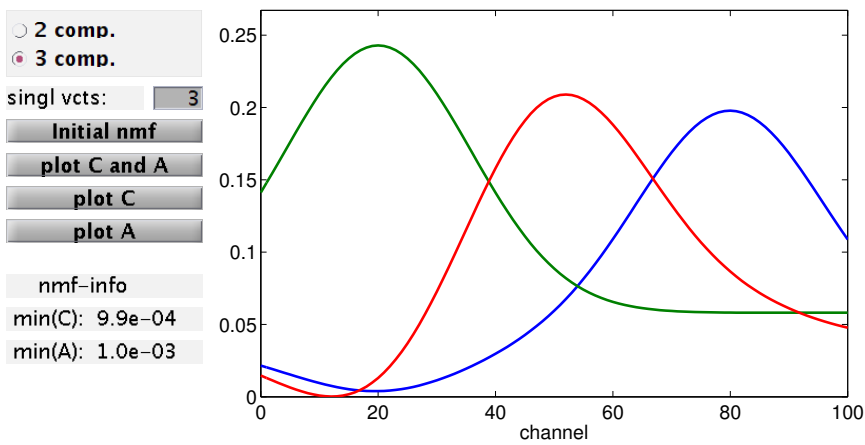Figure 4:  Test matrix example2.mat loaded by pressing the **Load** button (18).



Figure 5: The initial NMF for example2.mat with $s = 3$ components.

## 5.4  Computation of the AFS

For two-component systems ($s = 2$) the AFS consists of two real intervals. **FAC-PACK** represents the associated AFS by the two orthogonal sides of a rectangle. The sides are just the intervals of admissible values for $\alpha$ and $\beta$ where

$$T = \left( \begin{array}{cc} 1 & \alpha \\ 1 & \beta \end{array} \right)$$

is the transformation matrix which constitutes the rotational ambiguity. See [1, 2] for the AFS for the case $s = 2$. However in [1], see Equation (6), the entries of the second row of $T$ are interchanged.

For three-component systems ($s = 3$) the AFS is formed by all points ($\alpha, \beta$) so that

$$T = \left( \begin{array}{ccc} 1 & \alpha & \beta \\ 1 & s_{11} & s_{12} \\ 1 & s_{21} & s_{22} \end{array} \right)$$

is a transformation which is associated with nonnegative factors $C$ and $A$, see [17]. A permutation argument shows that with ($\alpha, \beta$) the points ($s_{11}, s_{12}$) and ($s_{21}, s_{22}$) belong to the AFS, too.

The initial NMF is the starting point for the numerical computation of the AFS since it provides first $\alpha$ and $\beta$ in the AFS.

Next various control parameters for the numerical computation of the AFS are explained. Further two ways of computing different kinds of the AFS are introduced.

### 5.4.1  Control parameters

Five control parameters are used to steer the adaptive polygon inflation algorithm. **FAC-PACK** aims at the best possible approximation of the AFS with the smallest computational costs. Default values are pre-given for these control parameters.

- The parameter **e-neg. entr.** (37) is the noise level control parameter which is denoted by $\varepsilon$ in [17]. Negative matrix elements of $C$ and $A$ do not contribute to the penalization functional if their relative magnitude is larger than $-\varepsilon$. In other words, an NMF with such slightly nonnegative matrix elements is accepted as valid.

- The parameter **e-bound** (38) controls the precision of the boundary approximation and is denoted by $\varepsilon_b$ in [17]. Decreasing the value $\varepsilon_b$ improves the accuracy of the positioning of new vertices of the polygon.

- The parameter **d-stopping** (39) controls the termination of the adaptive polygon inflation procedure. This parameter is denoted by $\delta$ in Section 3.5 of [17] and is an upper bound for change-of-area which can be gained by a further subdivision of any of the edges of the polygon.

- **max fcls.** (40) and **max edges** (41) are upper bounds for the number of function-calls and for the number of vertices of the polygon.

### 5.4.2 Type of polygon inflation

*FAC-PACK* offers two possibilities to apply the polygon inflation method for systems with three components ($s = 3$). The user should select the proper method according to the following explanations:

1. The "classical" version of the polygon inflation algorithm is introduced in [17] and applies best to an AFS which consists of three clearly separated segments. Select **Polygon inflation** by pressing the upper button (42). A typical example is shown in Figure 6 for the test problem example2.mat. In each of the segments an initial polygon has been inflated from the interior. Interior points are accessible from an initial NMF.

2. Alternatively, the **Inverse polygon inflation** procedure is activated by pressing the lower button in (42). This should be done if the AFS is only one topologically connected set (with a hole) or if the isolated segments of AFS are in close neighborhood. See Figure 7 for an example.

   The inverse polygon inflation method is more expensive than the classical version. First the complement of the AFS is computed and then some superset of the AFS is computed. A set subtraction results in the desired AFS. The details are explained in [19].

The user should always try the second variant of the polygon inflation algorithm if the results of the first variant are not satisfying or if something appears to be doubtful.
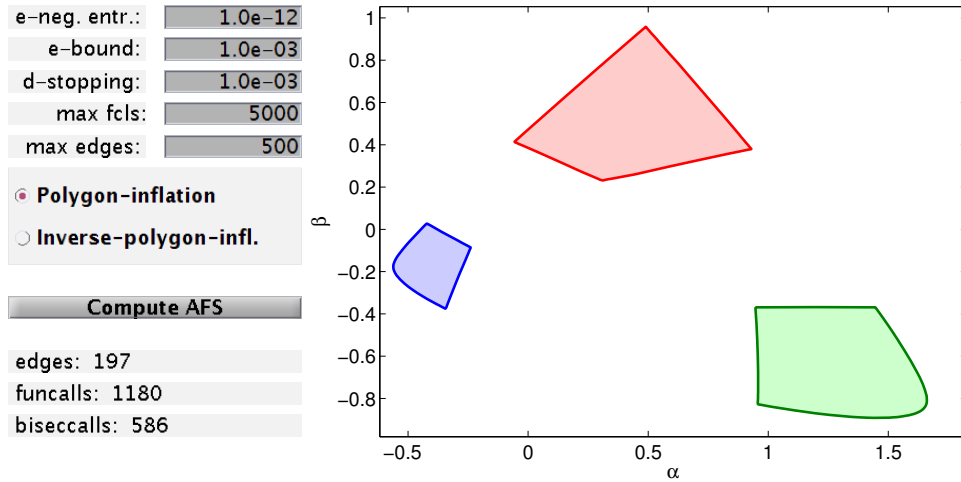
Figure 6:  An AFS with three clearly separated segments as computed by the "classical" polygon inflation algorithm. Data: example2.mat.
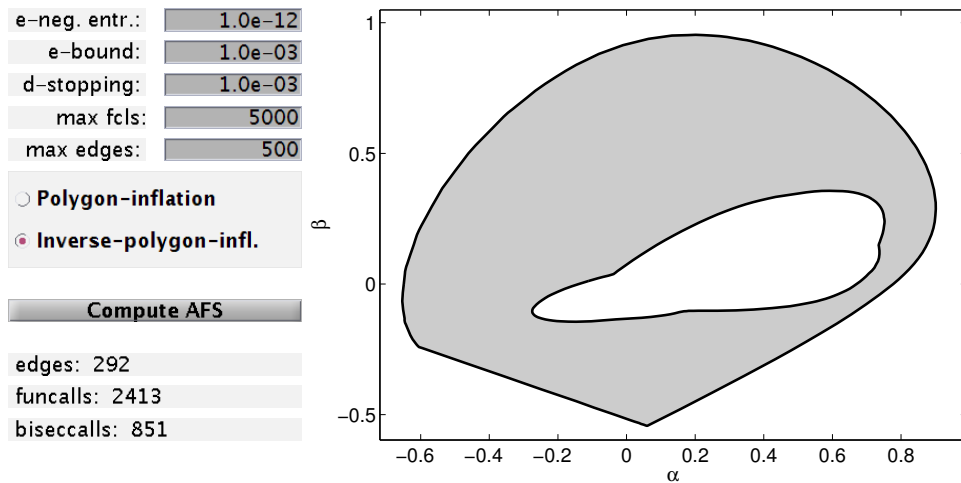


Figure 7:  An AFS which consists of only one segment with a hole. For the computation the inverse polygon inflation algorithm has been applied to example3.mat.

## 5.5 Factor representation & live-view mode

Next the plotting of the spectra and/or concentration profiles is described. These factors can be computed from the AFS by certain linear combinations of the right and/or left singular vectors. The live-view mode allows an interactive representation of the factors by moving the mouse pointer through the AFS.

### 5.5.1 Two-component systems

For two-component systems a full and simultaneous representation of the factors $C$ and $A$ is possible. In order to get an overview of the range of possible spectra one can discretize the AFS-intervals equidistantly. For each point of the resulting 2D discrete grid the associated spectrum is plotted in the factor window. The discretization parameter (subinterval length) is selected in the field (50) and/or (51). The plot of the spectra is activated by pressing button **AFS 1** (52) or button **AFS 2** (53).

<u>Live-view mode:</u> The live-view mode can be activated in the field (55). By moving the pointer through the AFS the associated solutions are shown interactively. Figure 8 shows a typical screen shot. The $y$-axis labels are turned off.

### 5.5.2 Three component systems

For three-component systems the range of possible solutions is presented for each segment of the AFS separately. The user can select with the buttons **AFS 1** (52), **AFS 2** (53) and **AFS 3** (54) a specific AFS which is then covered by a grid. First the boundary of the AFS segment (which is a closed curve) is discretized by a number of **bnd-pts** nodes. The discretization parameters for the interior of this AFS segment are **step size x** (50) and **step size y** (51); the interior points are constructed line by line. The resulting nodes are shown in the AFS window by symbols $\times$ in the color of the AFS segment. For each of these nodes the associated spectrum is drawn in the factor window. For the test problem example2.mat the bundle of spectra is shown in Figure 9. The resolution can be refined by increasing the number of points on the boundary or by decreasing the step size in the direction of $x$ or $y$.

If the admissible concentration profiles are to be printed, then the whole procedure is to be applied to the transposed data matrix $D$; activate the **Transpose** button (19) at the beginning and recompute everything.

<u>Live-view mode:</u> The live-view mode can be activated in the field (55). By moving the pointer through the AFS, the associated solutions are shown interactively. An example is shown in Figure 10.

Additionally, a certain point in the AFS can be locked by clicking the left mouse button; then the AFS for the remaining two components is re-computed. The resulting AFS is a smaller subset of the original AFS, which reflects the fact that some additional information is added by locking a certain point of the AFS. See Section 5.6 for further explanations.
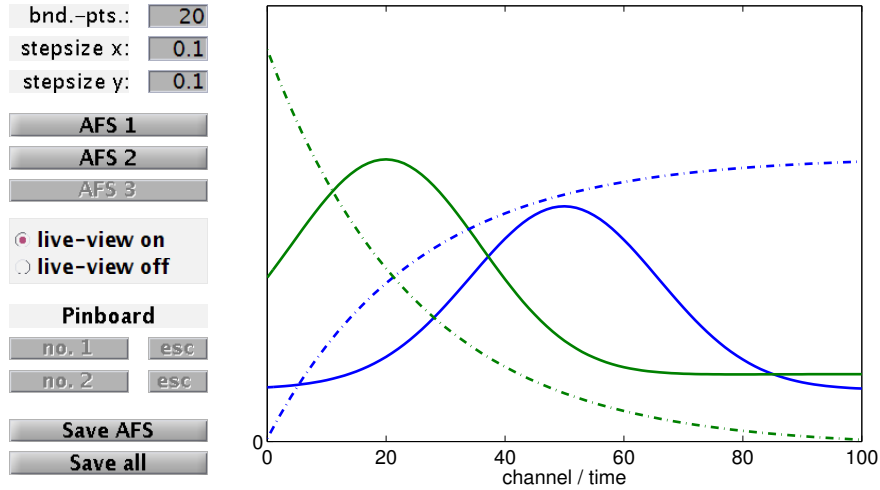
Figure 8:  The live-view mode for the two-component system example1.mat. By moving
the mouse pointer through the AFS the transformation to the factors $C$ and $A$
is computed interactively and all results are plotted in the factor window. The
spectra are drawn by solid lines and the concentration profiles are represented
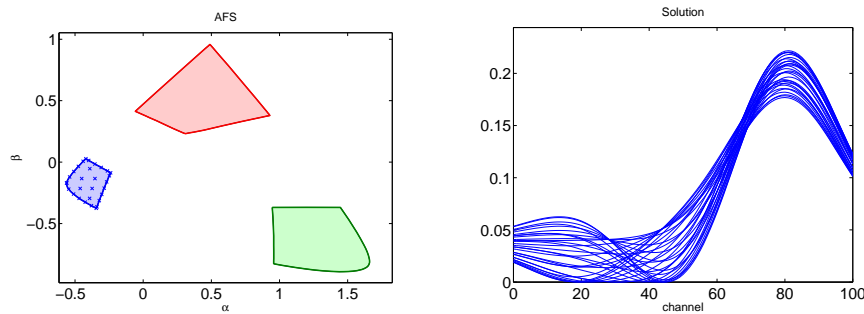by dash-dotted lines.



Figure 9:  The AFS for the problem example2.mat consists of three separated segments.
The blue segment of the AFS is covered with grid points (dark blue points).
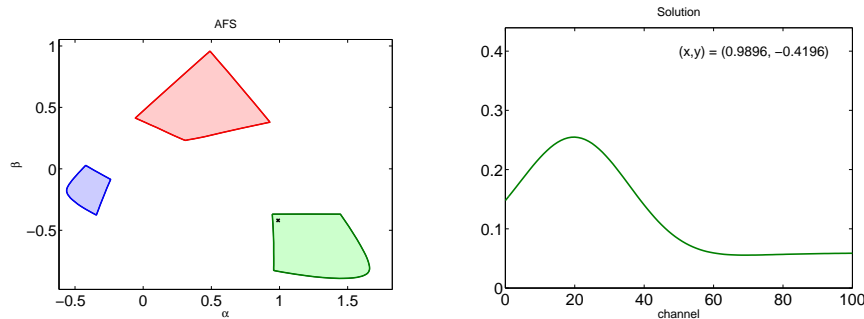For each of these points the associated spectrum is plotted in the right figure.



Figure 10: The live-view mode is active for the 3 component system example2.mat. In the
green AFS-segment the mouse pointer is positioned in the left upper corner at
the coordinates $(0.9896, -0.4196)$. The associated spectrum is shown in the
right figure together with the mouse pointer coordinates.

## 5.6  Reduction of the rotational ambiguity

With the computation of the AFS **FAC-PACK** provides a continuum of admissible matrix factorizations. All these factorizations are mathematically correct in the sense that they represent nonnegative matrix factors, whose product reproduces the original data matrix $D$. However, the user aims at the *one* solution which is believed to represent the chemical system correctly. Additional information on the system can help to reduce the so-called rotational ambiguity. **FAC-PACK** supports the user in this way. If, for instance one spectrum within the continuum of possible spectra is detected, which can be associated with a known chemical compound, then this spectrum can be locked and the resulting restrictions can be used to reduce the AFS for the remaining components.

### 5.6.1  Locking a first spectrum

After computing the AFS, activate the **live-view** mode (55). While moving the mouse pointer through any segment of the AFS, the factor window shows the associated spectra. If a certain (known) spectrum is found, then the user can lock this spectrum by clicking the left mouse button. A $\times$ is plotted into the AFS and the button **no. 1** (56) becomes active. By pressing this button a smaller subset of the original AFS is drawn by solid black lines in the AFS window. This smaller AFS reflects the fact that some additional information on the system has been added. The button **esc** (57) can be used for unlocking any previously locked point.

### 5.6.2  Locking a second spectrum

Having locked a first spectrum and having computed the reduced AFS the **live-view** mode can be reactivated. Then a second point within the reduced AFS can be locked. By pressing button **no. 2** (58) the AFS for the remaining component is reduced for a second time and is shown as a black broken line. Once again, the button **esc** (59) can be used for unlocking the second point.

Figure 11 shows the result of such a locking-and-AFS-reduction procedure for the test problem example2.mat. For this problem the AFS consists of three clearly separated segments and the final reduction of the AFS is shown by the black broken line in the red segment of the AFS. The live-view mode allows to display the possible spectra along this line. The resulting restrictions on the complementary concentration factor $C$ can be computed by the module AFS & Complementarity, see Section 6.
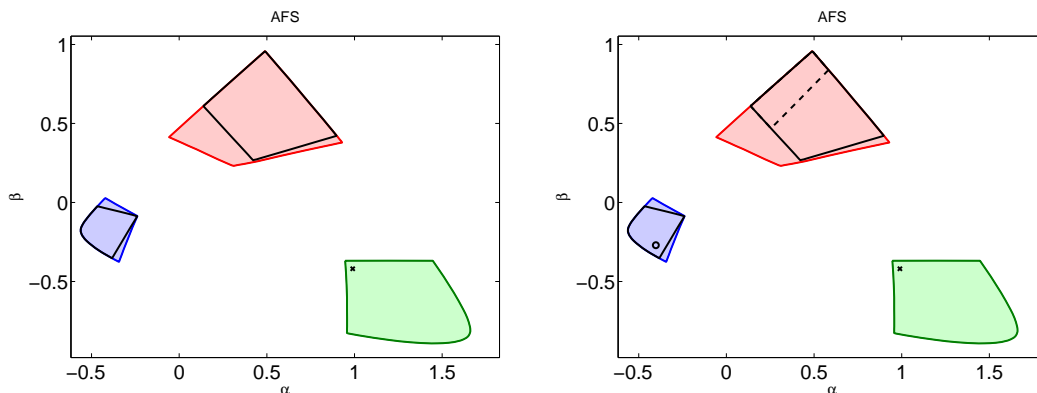
Figure 11:  Reduction of the AFS for the test problem example2.mat. Left: the reduced
           AFS is shown by black solid lines after locking a first spectrum (marker $\times$).
           Right: The AFS of the remaining third component is shown by a black broken
           line.

## 5.7  Save data and extract axes

To save the results in a MATLAB *.mat file press either **Save AFS** (60) or **Save all**
(61). A proper file name is suggested.

**Save AFS** saves the data $D$, the factors $U$, $S$, $V$ of the singular value decomposition,
the factors $Cinit$, $Ainit$ of the initial NMF and the AFS. The AFS has the data format
of a structure which contains the following variables:

- For $s = 2$ the AFS consists of two segments: $AFS\{1\} \in \mathbb{R}^2$ and $AFS\{2\} \in \mathbb{R}^2$
  with $\alpha \in [AFS\{1\}(1),\ AFS\{1\}(2)]$ and $\beta \in [AFS\{2\}(1),\ AFS\{2\}(2)]$ and vice
  versa.

- For $s = 3$ and an AFS consisting of three segments: $AFS\{i\} \in \mathbb{R}^{m_i \times 2}$ with $i =$
  $1, 2, 3$ is a polygon whose $x$-coordinates are $AFS\{i\}(:, 1)$ and whose $y$-coordinates
  are $AFS\{i\}(:, 2)$.

- For $s = 3$ and an AFS consisting of one segment with a hole: $AFS\{1\} \in \mathbb{R}^{m \times 2}$ is the
  outer polygon ($AFS\{1\}(:, 1)$ the $x$-coordinates and $AFS\{1\}(:, 2)$ the $y$-coordinates)
  and $AFS\{2\} \in \mathbb{R}^{\ell \times 2}$ is the inner polygon surrounding the hole.

If the lock-mode has been used, see Section 5.6, then the results are saved in $AFSLockMode1$
and $LockPoint2$ (case of one locked spectrum) as well as $AFSLockMode2$ and $LockPoint1$
(case of two locked spectra).

The user can get access to all figures.  Therefore click the right mouse button in the
desired figure. Then a separate MATLAB figure opens. The figure can now be modified,
printed or exported in the usual way.

## 5.8  Cancellation of the program

Whenever the buttons **Initial nmf** (30), **Compute AFS** (43), **no. 1** (57) or **no. 2**
(59) are pressed, an external C-routine is called, see Section 3.4.  The GUI does not

respond to any activities during the execution of the C-routine. Therefore, in case of any problems or in case of too long program runtimes, the C-routine cannot be canceled from the GUI. The termination can be enforced by typing CTRL + C in the MATLAB command window. Time consuming processes can be avoided if all the parameters are adjusted to reasonable (default) values.

## 5.9  How to get help

If the **Help** box (62) is checked and the mouse pointer is moved over a button, a text field or an axis, then a short explanation appears right next to the **Help** field. Further, a small pop-up window opens if the mouse pointer rests for more than one second on a button.

# 6 Complementarity & AFS module

This **FAC-PACK** module simultaneously shows the AFS for the spectral factor and the AFS for the concentration factor. It demonstrates how the rotational ambiguity for a three-component system can be reduced by means of the complementarity theory. For the complementarity theorem see [18, 19] and for comparable results see [3]. Solutions can be selected and modified within a live-view mode. Any changes in the spectral AFS are immediately mapped to changes in the concentrational AFS and vice versa. This gives the user the full control over and a visualization of a the complete factorization $D = CA$.

## 6.1 Initial steps

The Complementarity & AFS module module can be started by pressing the **Complementarity & AFS** button on the **FAC-PACK** start window. Data loading by the **Load** button (18) is explained in Section 3.2. The loading process is followed by a singular value decomposition (SVD) of the data matrix. The dimensions of $D$ are shown in the fields (69) and (70) together with the four largest singular values in the fields (71-74). For this module the fourth singular value should be "small" compared to the first three singular values so that the numerical rank of $D$ equals 3. For noisy data one might try to use a larger number of vectors in the field (78), see also Section 5.3.

In Figure 4 we consider the sample data example2.mat, cf. Section 3.3. The parameters in the fields (75)-(77) control the AFS computation. The meaning of these parameters is explained in Section 5.4.1.

## 6.2 AFS selection

In this module both the spectral AFS $\mathcal{M}_A$ and the concentrational AFS $\mathcal{M}_C$ are computed. It is also possible to compute FIRPOL which includes the AFS. The user can select to compute

1. the two FIRPOL sets which contain $\mathcal{M}_A$ and $\mathcal{M}_C$ as subsets. For the spectral factor FIRPOL is the set $\mathcal{M}_A^+ = \{(\alpha, \beta) : (1, \alpha, \beta) \cdot V^T \geq 0\}$; see also Equation (6) in [18]. Points in $\mathcal{M}_A^+$ represent only one nonnegative spectrum $A(1,:)$. The concentrational set FIRPOL $\mathcal{M}_C^+$ can be described in a similar way. The FIRPOL computation is computationally cheaper and faster since no optimization problems have to be solved.

2. the spectral AFS $\mathcal{M}_A$ and the concentrational AFS $\mathcal{M}_C$. These computations are often time-consuming.

FIRPOL as well as the AFS are computed by the polygon inflation method [17, 18]. Numerical calculations are done by the C-routine AFScomputationSYSTEMNAME, see Section 3.4, immediately after a selection is made in the checkbox (79). The results are shown in the windows (65) and (67).

Figure 12 shows FIRPOL $\mathcal{M}_A^+$ and the AFS $\mathcal{M}_A$ for the sample data **example2**. The trace of $D$, see Section 6.2.1, is also shown. The triangle in $\mathcal{M}_A$ represents a complete feasible factor $A$.
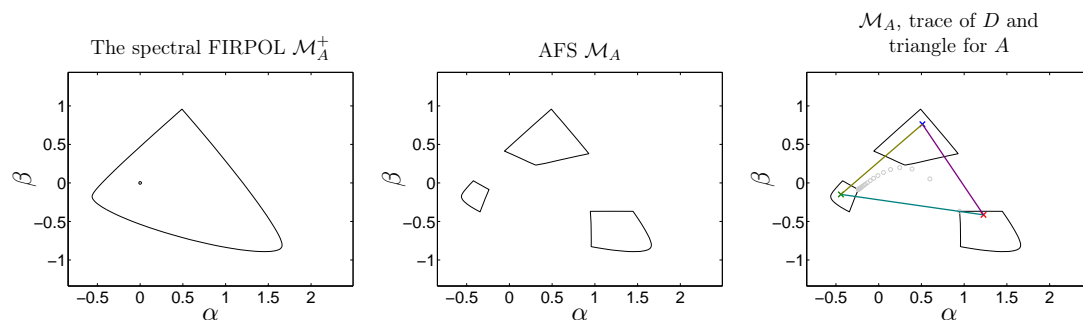
Figure 12: Sample data **example2**. Left: FIRPOL $\mathcal{M}_A^+$. Center: $\mathcal{M}_A$. Right: $\mathcal{M}_A$ with the trace of $D$ marked by gray circles and a triangle representing a feasible factor $A$.

### 6.2.1  Traces of the spectral data matrix $D$

The traces of spectral data matrix $D$ can be drawn in $\mathcal{M}_A$ and $\mathcal{M}_C$. The traces in $\mathcal{M}_A$ are the normalized expansion coefficients of the rows of $D$ with respect to the right singular vectors $V(:,2)$ and $V(:,3)$. The normalization is that the expansion coefficient for $V(:,1)$ equals 1. Thus the trace $\mathcal{M}_A$ is given by the $k$ points

$$w_i = \left( \frac{(DV)_{i2}}{(DV)_{i1}}, \frac{(DV)_{i3}}{(DV)_{i1}} \right) = \left( \frac{D(i,:) \cdot V(:,2)}{D(i,:) \cdot V(:,1)}, \frac{D(i,:) \cdot V(:,3)}{D(i,:) \cdot V(:,1)} \right), \quad i = 1, \ldots, k.$$

Analogously, the traces of $D$ in $\mathcal{M}_C$ are the normalized expansion coefficients of the columns of $D$ with respect to the scaled left singular vectors $\sigma_2^{-1} U(:,2)$ and $\sigma_3^{-1} U(:,3)$

$$u_j = \left( \frac{(\Sigma^{-1} U^T D)_{2j}}{(\Sigma^{-1} U^T D)_{1j}}, \frac{(\Sigma^{-1} U^T D)_{3j}}{(\Sigma^{-1} U^T D)_{1j}} \right) = \left( \frac{\sigma_1 U(:,2)^T \cdot D(:,j)}{\sigma_2 U(:,1)^T \cdot D(:,j)}, \frac{\sigma_1 U(:,3)^T \cdot D(:,j)}{\sigma_3 U(:,1)^T \cdot D(:,j)} \right)$$

for $j = 1, \ldots, n$. If the checkbox **trace** (80) is activated, then the traces of $D$ in $\mathcal{M}_A$ and $\mathcal{M}_C$ are plotted by small gray circles.

## 6.3  Live-view mode

The AFS can be explored in a live-view mode which allows an interactive representation of the factors by moving the mouse pointer through the AFS. Points in $\mathcal{M}_A$ and or $\mathcal{M}_C$ can be locked by clicking the left mouse button in the AFS. Any points can be unlocked and afterwards moved in the AFS. The modification of a certain point in one AFS results in changes for the components in the other AFS. All this is shown interactively. A unique factor $C$ or $A$ is represented by a triangle in the AFS. The vertices of these triangles can be unlocked by the buttons (82)-(84) and (86)-(88). By a step-by-step selection and modification of three vertices in the AFS for $A$ (or three vertices in the AFS for $C$) the user can construct two triangles and thus a complete factorization $D = CA$. Next an initial triangle is constructed in $\mathcal{M}_A$; the whole procedure can also be applied to $\mathcal{M}_C$.

### 6.3.1  Initialization: Selection of a first point

By activating the first-radio button (83) a first spectrum/point of the spectral AFS can be selected. If the mouse pointer is moved through the AFS window (67), then the associated

spectra are shown in the window (66) for the factor $A$. The complementary concentration profiles are restricted by the complementarity theorem to a one-dimensional affine space; the associated straight line is shown in the concentrational AFS simultaneously. By pressing the left mouse button a certain point in the AFS can be locked. All this is illustrated in Figure 13.
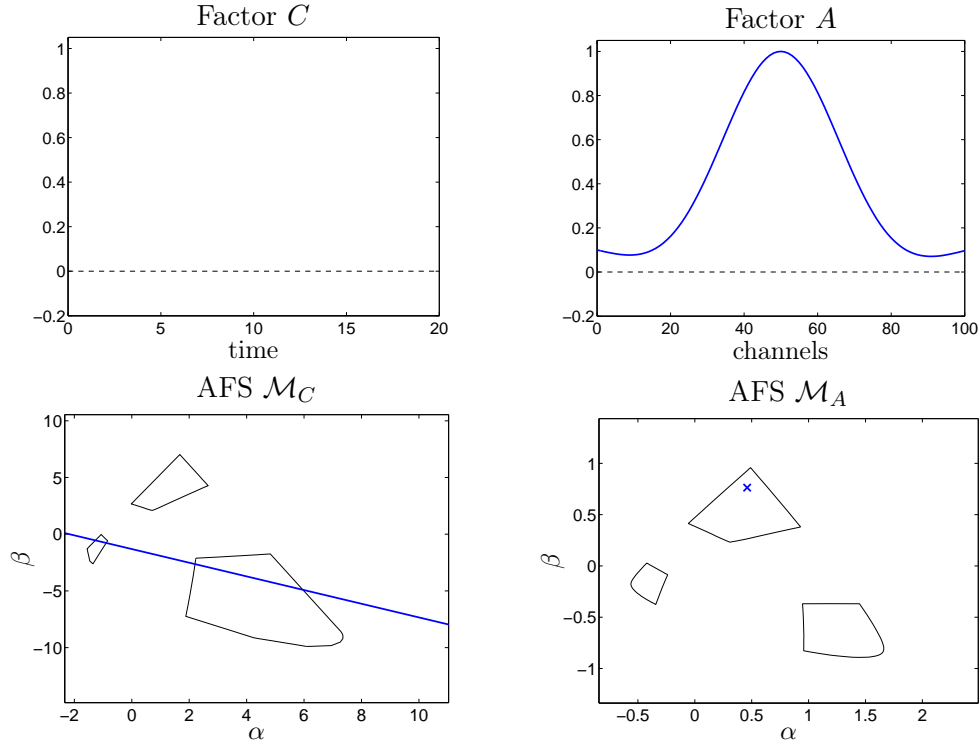


Figure 13: A first point, marked by ×, is locked in the AFS $\mathcal{M}_A$ (lower-right). This results in: Upper-left: no concentration profile has uniquely been defined up to now. Upper-right: a single pure component spectrum has been fixed. Lower-left: the blue straight line is a restriction on the complementary concentration profiles $C(:, 2:3)$ in $\mathcal{M}_C$.

### 6.3.2  Initialization: Selection of a second point

Next a second point can be selected in $\mathcal{M}_A$. Therefore the second-radio button (84) can be activated and the mouse pointer can be moved through the AFS $\mathcal{M}_A$. The following items are plotted simultaneously:

- the two selected points in the AFS $\mathcal{M}_A$,

- the two associated spectra in the factor window (66),

- the two complementary straight lines and the point of intersection in the AFS $\mathcal{M}_C$ (65),

- the associated concentration profile for the uniquely determined complementary concentration profile in the factor window (64).

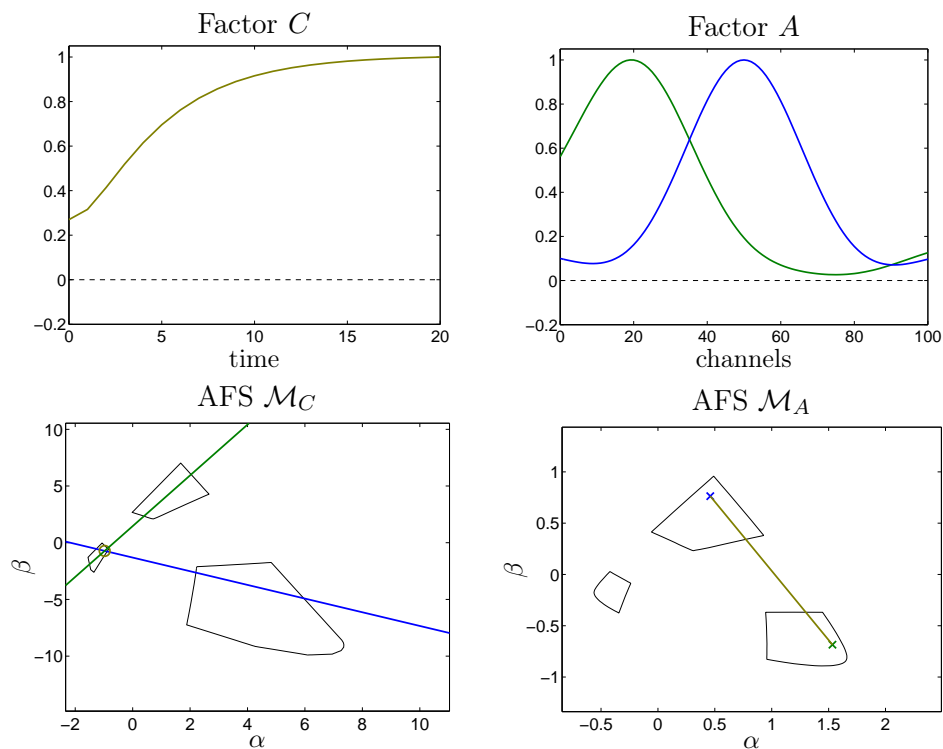By pressing the left mouse-button a certain point can be locked. The result is illustrated in Figure 14.



Figure 14: A second point, marked by ×, is locked in the AFS $\mathcal{M}_A$ (lower-right). This results in: Upper-left: one complementary concentration profile is uniquely determined. Upper-right: the two locked pure component spectra. Lower-left: the blue straight line restricts the complementary concentration profiles $C(:,2:3)$ and the green straight line restricts the complementary concentration profiles $C(:,1)$ and $C(:,3)$ in $\mathcal{M}_C$. One concentration profile marked by ○ is now uniquely determined.

### 6.3.3 Initialization: Selection of a third point

A third point can be added by activating the red third-radio button (85) and moving the mouse pointer through the AFS $\mathcal{M}_A$. The following items are plotted simultaneously:

- a triangle in the AFS $\mathcal{M}_A$ whose vertices uniquely determine the factor $A$,
- the three associated spectra in the factor window (66),
- the three complementary straight lines which define a triangle in the AFS $\mathcal{M}_C$ (65),
- the associated concentration profiles in the factor window $C$ (64).

Once again a third point can be locked by clicking the left mouse button. Figure 15 illustrates this.
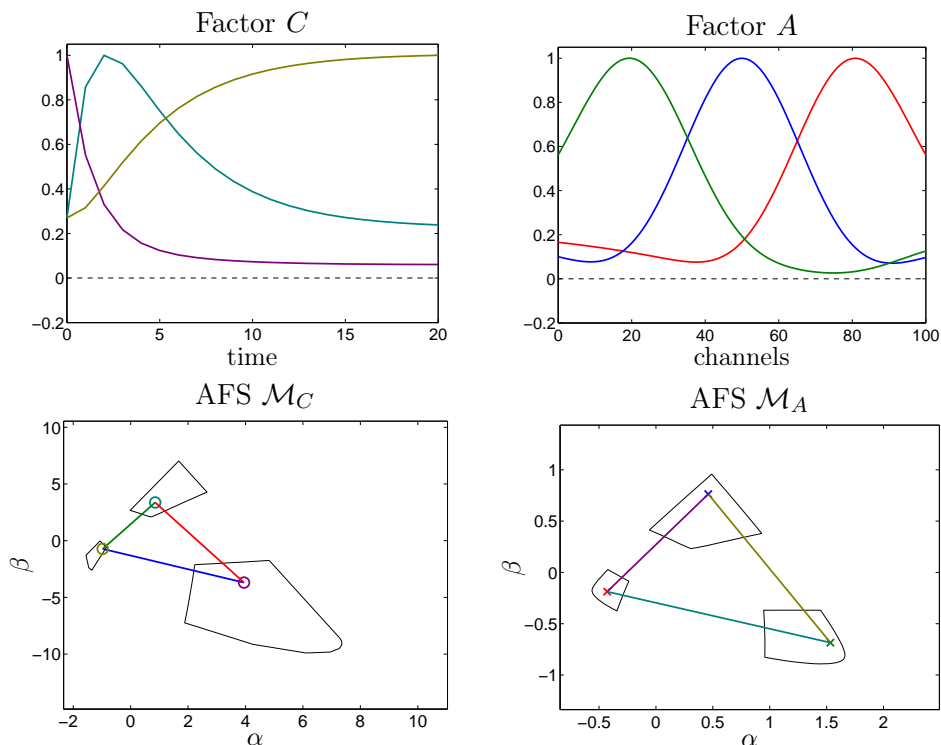
Figure 15: Three locked points in $\mathcal{M}_A$ uniquely define the factor $A$ represented by a feasible triangle (lower-right). This results in: Upper-left: three complementary concentration profiles. Upper-right: three pure component spectra. Lower-left: a triangle in the AFS $\mathcal{M}_C$.

## 6.4  Modification of the solution

Once an initial solution has been selected, one can now modify the vertices of the triangles. This can be done by activating the buttons

- (80)-(82) to modify the triangle in $\mathcal{M}_C$,
- (84)-(86) to modify the triangle in $\mathcal{M}_A$.

The selection and re-positioning of the vertices has to follow the rule that all six vertices are within the segments of $\mathcal{M}_A$ and $\mathcal{M}_C$. Otherwise, negative components can be seen in the factor windows (64) and (66). In general a vertex in one AFS is associated with a line segment in the other AFS; see [16] for the complementarity theorem and [3, 18, 19] for applications to three-component systems. Any vertex in one AFS and the associated line segment in the other AFS are plotted in the same color in order to express their relationship. By changing one vertex of a triangle, two vertices of the other triangle are affected.

## 6.5  Contour plots on factor properties

The Complementarity & AFS module can help the user to select pure components in the AFS which have certain desired properties. Sometimes it might help to know where in an

AFS monotone concentration profiles or exponentially decaying profiles (of a reactant) can be found. Smoothness of a pure component profile or a small integral of a spectrum might also be valuable selection criteria. In the Complementarity & AFS module any point in the $(\alpha, \beta)$-plane can be evaluated with respect to various constraint functions which estimate the smoothness, monotonicity, exponential decay and so on. The result in form of a contour plot is then shown in the AFS.

By using the radio buttons (90)-(92) for the factor $C$ and by using the radio buttons (94) and (95) for the factor $A$ the contour plots are shown. These plots can be removed from the AFS by clicking the buttons (89) and (93). The numerical evaluation of the constraint function may take some time.

Contour plots for the following constraint functions are available:

- **monotone** concentration profiles (90): monotonously increasing or decreasing profiles are favored and are shown in light gray or white in the contour plot,

- **smooth** concentration profiles (91): profiles with a small Euclidean norm of the discrete second derivative are favored (91) (shown white in the contour plot),

- **exponentially** decaying concentration profiles (92): for every point of a grid in the $(\alpha, \beta)$-plane the approximation error for an exponentially decaying function (with an optimized decay constant) is calculated. This allows to favor reactants decaying exponentially, which can be found in the light gray or white areas in $\mathcal{M}_C$,

- **smooth** spectra (94): spectra with a small Euclidean norm of the discrete second derivative are favored,

- small **norm** spectra: spectra with a small Euclidean norm of the representing vector are favored. Thus spectra with few isolated and narrow peaks are shown in light gray compared to those with wide absorbing peaks (shown in darker gray).
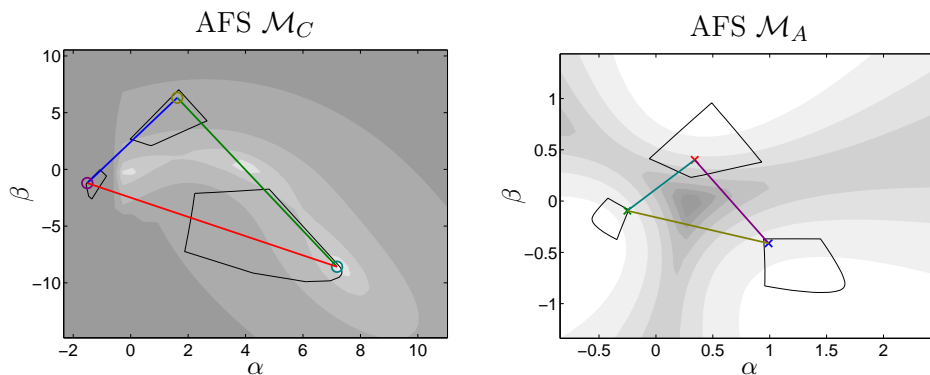
Figure 16 shows two examples.



Figure 16: Contour plots for two soft constraints. Left: Contour evaluating the closeness to an exponentially decaying function. The right lower vertex (marked by ∘) represents an exponentially decaying concentration profile. Right: Contour plot for the Euclidean norm of the spectrum for example2.mat.

## 6.6 Import of spectra and concentration profiles

If the user can provide one or more pure component spectra or certain concentration profiles for the reaction system under consideration, then these data can be loaded into the Complementarity & AFS module. After pressing the **Import spectra**-button (98) one, two or three pure component spectra can be loaded. These should be stored in a MATLAB variable named $a$ being an $\ell \times n$ vector in case of $\ell$ pre-given spectra. (Linear interpolation should be used if a known spectrum does not fit the dimension $n$.) Any loaded spectrum is marked in the spectral AFS $\mathcal{M}_A$ by a $*$.

Analogously, known concentration profiles can be loaded by pressing the **Import concentration profiles** (96) button. The data must be provided in a $*$.mat file containing the variable $c$; columns with $k$ components determine pre-given concentration profiles.

## 6.7 Save data and extract axes

In order to export the solutions press the **Save**-button (97). Then

- the variables determining the geometry of FIRPOL and of the AFS,

- the initial spectral data, the SVD,

- and, if available, the locked points in AFS $\mathcal{M}_A$ and $\mathcal{M}_C$, the associated spectra plus concentration profiles and transformation matrix $T$ determining a unique factorization $CA$

are saved in a MATLAB file.

Any figure window in the GUI can be accessed by clicking the right mouse button in the desired figure. Then a separate MATLAB figure opens. The figure can now be modified, printed or exported in the usual way.

## 6.8 Program abortion

The AFS & complementarity module calls an external C-routine if FIRPOL or the AFS is computed through the check fields (79). The GUI does not respond to any commands during the execution of the C-routine. In case of any problems or in case of too long program runtimes, the C-routine cannot be canceled from the GUI. Program abortion can be enforced by typing CTRL + C in the MATLAB command window.
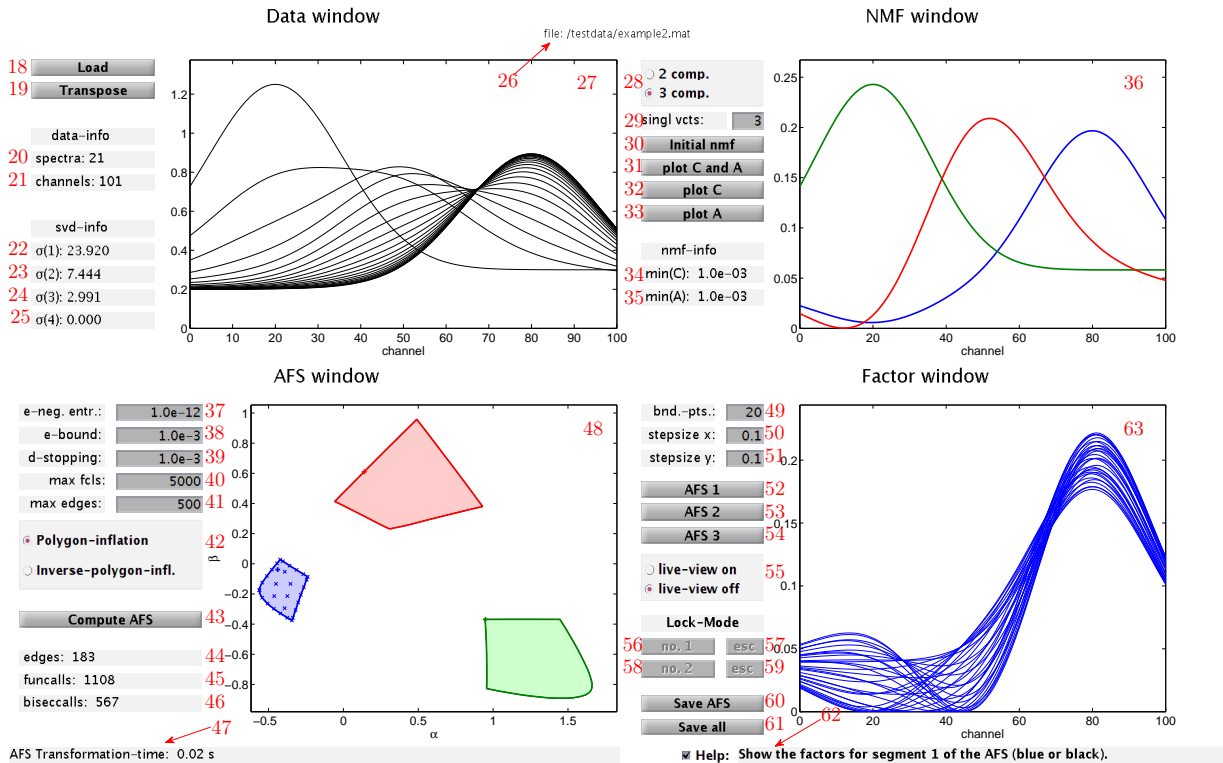
# 7 Screen shots & description of operation panels
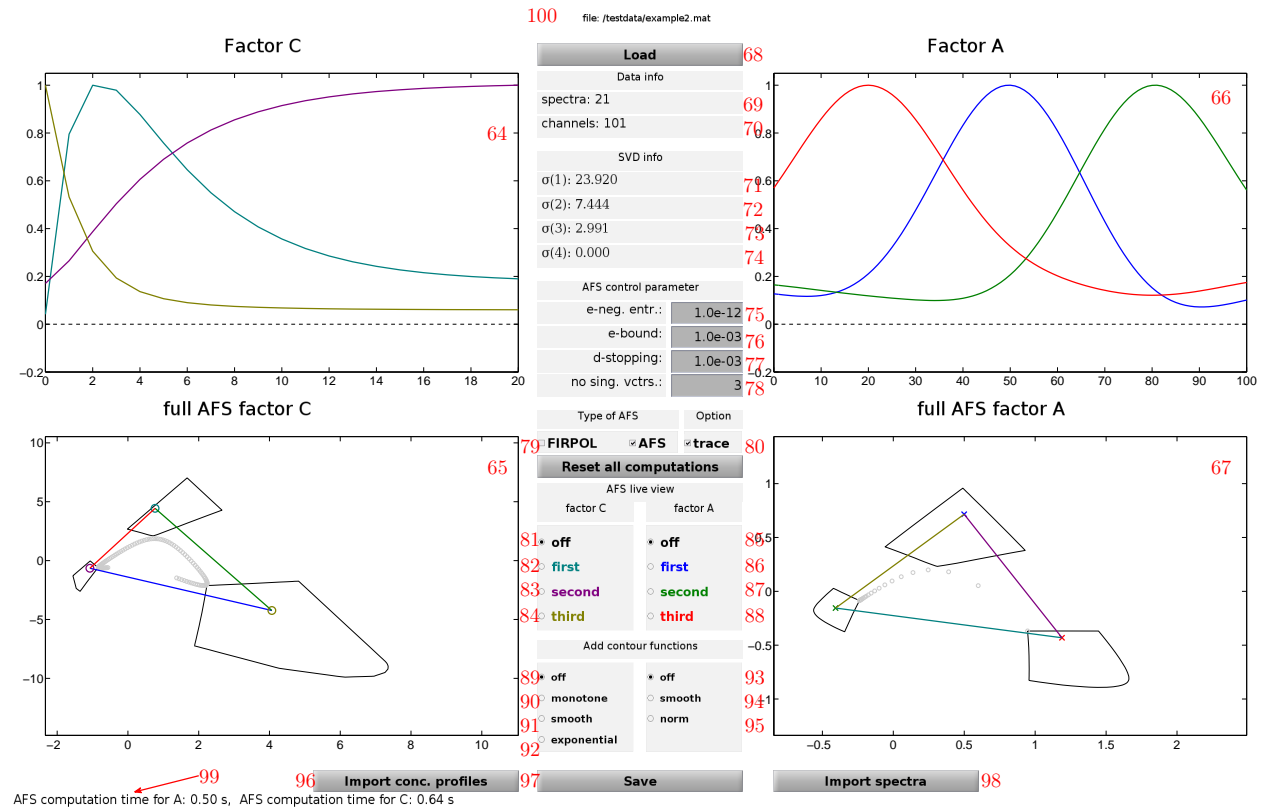
I: Screen shot of the **Baseline-correction** module:
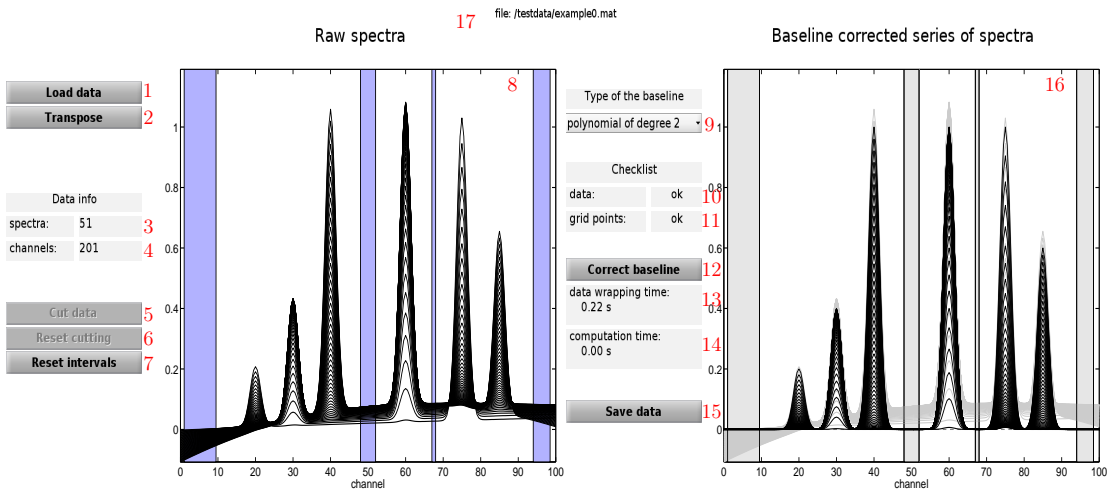


The GUI buttons, data fields and figures:

1  Load data matrix $D$
2  Transpose $D$
3  # rows of $D$
4  # columns of $D$
5  Cut spectra as marked by mouse
6  Reset spectral cutting
7  Reset selected intervals
8  Raw spectra window
9  Select type of baseline correction

10  Status of data consistency
11  Status of band selection consistency
12  Apply baseline correction
13  Computing time for data wrapping
14  Computing time for baseline correction
15  Save button
16  Corrected spectra window
17  Name of data file

II: Screen shot of the **AFS computation** module:



The GUI buttons, data fields and figures:

18 Load button

19 Transpose button

20 # rows of data matrix $D$

21 # columns in $D$

22 Singular value 1 of $D$

23 Singular value 2 of $D$

24 Singular value 3 of $D$

25 Singular value 4 of $D$

26 File name

27 Initial spectra window

28 Input # components

29 # of singular values used for initial NMF

30 Compute NMF

31 Plot factors $C$ and $A$

32 Plot factor $C$

33 Plot factor $A$

34 Minimal matrix elements of $C$ in the NMF

35 Minimal matrix elements of $A$ in the NMF

36 NMF window

37 Acceptable negativeness for the NMF

38 Precision at the boundary

39 Stopping precision

40 Maximal # calls of the target function

41 Maximal # of edges

42 Select type of polygon inflation

43 Compute AFS

44 # edges

45 # calls of target function

46 No. used function calls for the boundary-precision

47 Output on the numerical computation

48 AFS window

49 # of boundary points for the factor representation

50 Step size $x$ direction for the factor representation

51 Step size $y$ direction for the factor representation

52 Plot factors for segment 1 of the AFS

53 Plot factors for segment 2 of the AFS

54 Plot factors for segment 3 of the AFS

55 Activate live-view mode

56 Compute AFS for 1 selected point

57 Reset 1. restricted AFS

58 Compute AFS for 2 selected points

59 Reset 2. restricted AFS

60 Save button (AFS)

61 Save button (all)

62 Help field

63 Factor window

III: Screen shot of the **Complementarity & AFS** module:



The GUI buttons, data fields and figures:

64 Factor $C$
65 AFS for $C$
66 Factor $A$
67 AFS for $A$
68 Load spectral data matrix $D$
69 # rows of $D$
70 # columns of $D$
71 Singular value 1 of $D$
72 Singular value 2 of $D$
73 Singular value 3 of $D$
74 Singular value 4 of $D$
75 Acceptable negativeness for the NMF
76 Precision at the boundary
77 Stopping precision
78 # of singular vectors used for initial NMF

79 Select AFS (FIRPOL and/or full AFS)
80 Mark mixed spectra (rows of $D$) in the AFS
81 Live-view mode $C$ off
82 Live-view mode $C(:, 1)$ on
83 Live-view mode $C(:, 2)$ on
84 Live-view mode $C(:, 3)$ on
85 Live-view mode $A$ off
86 Live-view mode $A(1, :)$ on
87 Live-view mode $A(2, :)$ on
88 Live-view mode $A(3, :)$ on
89 Contour mode $C$ off

90 Contour mode for $C$ - monotone profiles
91 Contour mode for $C$ - smooth profiles
92 Contour mode for $C$ - exponential decay
93 Contour mode for $A$ off
94 Contour mode for $A$ - smooth spectra
95 Contour mode for $A$ - spectra with a small norm
96 Import external concentration profiles
97 Save the results
98 Import external spectra
99 Computing time
100 Name of data file

# References

[1] H. Abdollahi, M. Maeder, and R. Tauler, *Calculation and Meaning of Feasible Band Boundaries in Multivariate Curve Resolution of a Two-Component System*, Analytical Chemistry **81** (2009), no. 6, 2115–2122.

[2] H. Abdollahi and R. Tauler, *Uniqueness and rotation ambiguities in Multivariate Curve Resolution methods*, Chemometrics and Intelligent Laboratory Systems **108** (2011), no. 2, 100–111.

[3] S. Beyramysoltan, R. Rajkó, and H. Abdollahi, *Investigation of the equality constraint effect on the reduction of the rotational ambiguity in three-component system using a novel grid search method*, Analytica Chimica Acta **791** (2013), no. 0, 25–35.

[4] O.S. Borgen and B.R. Kowalski, *An extension of the multivariate component-resolution method to three components*, Analytica Chimica Acta **174** (1985), 1–26.

[5] A. Golshan, H. Abdollahi, and M. Maeder, *Resolution of Rotational Ambiguity for Three-Component Systems*, Analytical Chemistry **83** (2011), no. 3, 836–841.

[6] J.C. Hamilton and P.J. Gemperline, *Mixture analysis using factor analysis. II: Self-modeling curve resolution*, J. Chemometrics **4** (1990), no. 1, 1–13.

[7] J. Jaumot and R. Tauler, *MCR-BANDS: A user friendly MATLAB program for the evaluation of rotation ambiguities in Multivariate Curve Resolution*, Chemometrics and Intelligent Laboratory Systems **103** (2010), no. 2, 96–107.

[8] A. Juan, M. Maeder, M. Martínez, and R. Tauler, *Combining hard and soft-modelling to solve kinetic problems*, Chemometr. Intell. Lab. **54** (2000), 123–141.

[9] W.H. Lawton and E.A. Sylvestre, *Self modelling curve resolution*, Technometrics **13** (1971), 617–633.

[10] M. Maeder and Y.M. Neuhold, *Practical data analysis in chemistry*, Elsevier, Amsterdam, 2007.

[11] E. Malinowski, *Factor analysis in chemistry*, Wiley, New York, 2002.

[12] K. Neymeyr, M. Sawall, and D. Hess, *Pure component spectral recovery and constrained matrix factorizations: Concepts and applications*, J. Chemometrics **24** (2010), 67–74.

[13] R. Rajkó and K. István, *Analytical solution for determining feasible regions of self-modeling curve resolution (SMCR) method based on computational geometry*, J. Chemometrics **19** (2005), no. 8, 448–463.

[14] C. Ruckebusch and L. Blanchet, *Multivariate curve resolution: A review of advanced and tailored applications and challenges*, Analytica Chimica Acta **765** (2013), 28–36.

[15] M. Sawall, A. Börner, C. Kubis, D. Selent, R. Ludwig, and K. Neymeyr, *Model-free multivariate curve resolution combined with model-based kinetics: algorithm and applications*, J. Chemometrics **26** (2012), 538–548.

[16] M. Sawall, C. Fischer, D. Heller, and K. Neymeyr, *Reduction of the rotational ambiguity of curve resolution technqiues under partial knowledge of the factors. Complementarity and coupling theorems*, J. Chemometrics **26** (2012), 526–537.

[17] M. Sawall, C. Kubis, D. Selent, A. Börner, and K. Neymeyr, *A fast polygon inflation algorithm to compute the area of feasible solutions for three-component systems. I: Concepts and applications*, J. Chemometrics **27** (2013), 106–116.

[18] M. Sawall and K. Neymeyr, *A fast polygon inflation algorithm to compute the boundary of the area of feasible solutions for three-component systems. II: Theoretical foundation, inverse polygon inflation and FAC-PACK implementation*, Accepted for J. Chemometrics, Technical Report, Universität Rostock, 2014.

[19] ———, *On the area of feasible solutions and its reduction by the complementarity theorem*, Technical report, Universität Rostock, 2014.

[20] R. Tauler, *Calculation of maximum and minimum band boundaries of feasible solutions for species profiles obtained by multivariate curve resolution*, J. Chemometrics **15** (2001), no. 8, 627–646.

[21] M. Vosough, C. Mason, R. Tauler, M. Jalali-Heravi, and M. Maeder, *On rotational ambiguity in model-free analyses of multivariate data*, J. Chemometrics **20** (2006), no. 6-7, 302–310.