

How to compute the Area of Feasible Solutions

A practical case study and users' guide to *FAC-PACK*

M. Sawall¹ and K. Neymeyr^{1,2},

¹ Institut für Mathematik, Universität Rostock, Ulmenstrasse 69, 18057 Rostock, Germany.

² Leibniz-Institut für Katalyse, Albert-Einstein-Strasse 60, 18069 Rostock, Germany.

Abstract

The area of feasible solutions (AFS) is a low dimensional representation of the set of all possible nonnegative factorizations of a given spectral data matrix. The AFS can be used to investigate the so-called rotational ambiguity of MCR methods in chemometrics, and the AFS is an excellent starting point for feeding in additional information on the chemical reaction system in order to reduce the ambiguity up to a unique factorization.

The *FAC-PACK* program package is a freely available and easy-to-use Matlab toolbox (with a core written in C and FORTRAN) for the numerical AFS computation for two- and three-component systems. The numerical algorithm of *FAC-PACK* is based on the inflation of polygons and allows a stable and fast computation of the AFS. In this contribution we explain all functionalities of the *FAC-PACK* software and demonstrate its application to a consecutive reaction system with three components.

1 Introduction

Multivariate curve resolution techniques in chemometrics suffer from the non-uniqueness of the nonnegative matrix factorization $D = CA$ of a given spectral data matrix $D \in \mathbb{R}^{t \times n}$. Therein $C \in \mathbb{R}^{n \times s}$ is the matrix of pure components concentration profiles and $A \in \mathbb{R}^{s \times n}$ is the matrix of the pure component spectra. Further, s denotes the number of independent chemical components. This non-uniqueness of the factorization is often called the rotational-ambiguity. The set of all possible factorizations of D into nonnegative matrices C and A can be presented by the so-called Area of Feasible Solutions (AFS).

The first graphical representation of an AFS goes back to Lawton and Sylvestre in 1971 [10]. They have drawn for a two-component system the AFS as a pie-shaped area in the plane; this area contains all pairs of the two expansion coefficients with respect to the basis of singular vectors, which result in nonnegative factors C and A . In 1985 Borgen and Kowalski [4] extended AFS computations to three-component systems by the so-called Borgen plots, see also [14, 22, 1, 2].

In recent years, methods for the numerical approximation of the AFS have increasingly gained importance. Golshan, Abdollahi and Maeder presented in 2011 a new idea for the numerical approximation of the boundary of the AFS for three-component systems by chains of equilateral triangles [6]. This idea has even been extended to four-component systems [7].

An alternative numerical approach for the numerical approximation of the AFS is the *polygon inflation method* which has been introduced in [18, 19, 20]. The simple idea behind this algorithm is to approximate the boundary of each segment of the AFS by a sequence of adaptively refined polygons. A numerical software for AFS computations with the polygon inflation method is called *FAC-PACK* and can be downloaded from

<http://www.math.uni-rostock.de/facpack/>

The current software revision 1.1 of *FAC-PACK* appeared in February 2014.

The aim of this contribution is to give an introduction into the usage of *FAC-PACK*. In Section 2 a sample model problem is introduced, the initial data matrix is generated in MATLAB and some functionalities of the software are demonstrated. In Section 3 this demonstration is followed by the Users' guide to *FAC-PACK*, which appears here for the first time in printed form.

2 A practical case study

2.1 Generation of test data

Let us consider the consecutive system of reactions $X \rightarrow Y \rightarrow Z$ with kinetic constants $k_1 = 0.75$ and $k_2 = 0.25$. The time interval for the concentration factor C is assumed to be $t \in [0, 20]$ with an equidistant discretization by $k = 21$ grid points. The spectral factor A is constructed on the wavelength interval $\lambda \in [0, 100]$ with an equidistant discretization which uses $n = 101$ points. The three Gaussian functions (plus a constant)

$$a_1(\lambda) = 0.95 \exp\left(-\frac{(\lambda - 20)^2}{500}\right) + 0.3,$$

$$a_2(\lambda) = 0.9 \exp\left(-\frac{(\lambda - 50)^2}{500}\right) + 0.25,$$

$$a_3(\lambda) = 0.7 \exp\left(-\frac{(\lambda - 80)^2}{500}\right) + 0.2$$

define the rows of A by their evaluation along the discrete wavelength axis. According to the bilinear Lambert-Beer model the spectral data matrix $D = CA$ for this reaction is a 21×101 matrix with the rank 3. This model problem is a part of the *FAC-PACK* software; see the data set `example2.mat` in Section 3.3.3. The pure component concentration profiles and spectra together with the mixture spectra for the sample problem are shown in Figure 1.

The MATLAB code to generate data matrix D reads as follows:

```
t = linspace(0,20,21)';
x = linspace(0,100,101)';
k = [0.75 0.25];
C(:,1) = exp(-k(1)*t);
C(:,2) = k(1)/(k(2)-k(1))*(exp(-k(1)*t)-exp(-k(2)*t));
C(:,3) = 1-C(:,1)-C(:,2);
A(1,:) = 0.95*exp(-(x-20).^2./500)+0.3;
A(2,:) = 0.9*exp(-(x-50).^2./500)+0.25;
A(3,:) = 0.7*exp(-(x-80).^2./500)+0.2;
D = C*A;
save('example2', 'x', 't', 'D');
```

2.2 AFS computation with *FAC-PACK*

Here we consider the *factorization problem* which is an inverse problem compared to the construction of D in Section 2.1. For given D we are looking for matrix factorizations $D = CA$ with nonnegative factors C and A . The pair of matrices (C, A) as constructed in Section 2.1 is one possible factorization of D

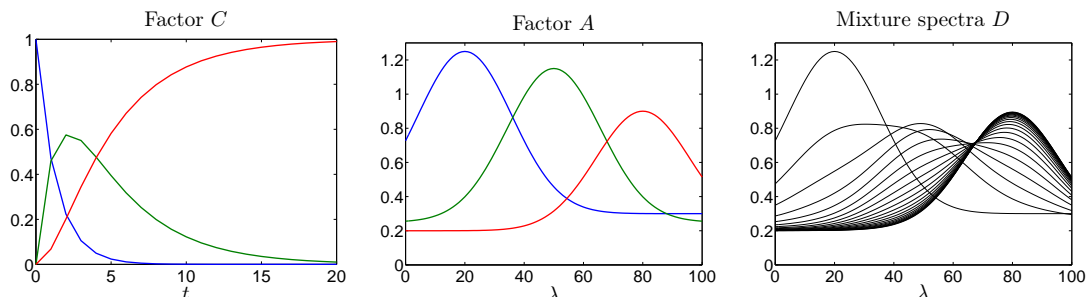


Figure 1: Pure component concentration profiles and spectra together with the mixture spectra for the sample problem presented in Section 2.1.

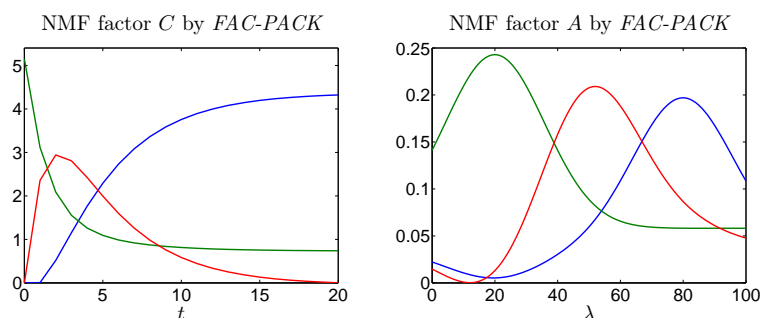


Figure 2: Initial nonnegative matrix factorization (NMF) for the model problem as computed by *FAC-PACK*.

within the set of all nonnegative matrix factorizations of D . The set of all nonnegative factorizations is described by the area of feasible solutions (AFS), which is a certain low dimensional representation of these feasible matrix pairs. For geometric techniques to compute the AFS see [4, 14] and the references therein. For numerical methods to compute the AFS see [1, 6, 2, 7, 18, 19].

In *FAC-PACK* the computation of the AFS is based on the polygon inflation algorithm, which is explained in [18, 19]. The users' guide to *FAC-PACK* is contained in Section 3; a quick introduction is given in Section 3.1 and a detailed introduction is given in Section 3.2. The program starts with the import of the spectral data, see Section 3.3.2. In the following we consider the model problem from Section 2.1. For this sample data there is no need for a baseline correction. Otherwise the baseline correction module as introduced in Section 3.1.1 can be used.

The first step of the polygon inflation algorithm is the computation of a singular value decomposition (SVD). The SVD is computed automatically after the data matrix is loaded. For the given sample problem the four largest singular values are

$$\sigma_1 = 23.920, \quad \sigma_2 = 7.444, \quad \sigma_3 = 2.991, \quad \sigma_4 = 0.000.$$

Since $\sigma_4 = 0$, the rank of D equals 3 and D represents a three-component system. After selecting 3 as the number of components in the AFS computation module, an initial nonnegative matrix factorization (NMF) can be computed, see Section 3.5.3. This initial NMF often has no physical or chemical meaning and is sometimes called an abstract factorization. However, this initial NMF provides the starting

point from which the polygon inflation algorithm inflates the polygons. The initial NMF for the model problem is presented in Figure 2.

The set of all nonnegative matrix factorizations of D is represented by the area of feasible solutions (AFS). For a three-component system the AFS is a subset of the two-dimensional plane, see Section 3.5.4 and [18] for the mathematical background of this low-dimensional representation. Section 3.5.4 additionally contains the setting options for the various control parameters of the polygon inflation algorithm and also the options on the algorithmic variants of the polygon inflation algorithm. The resulting area of feasible solutions (AFS) for the spectral factor A is denoted by \mathcal{M}_A and is shown in Figure 3. This AFS consists of three separated segments (isolated subsets) shown by a blue, a green and a red set.

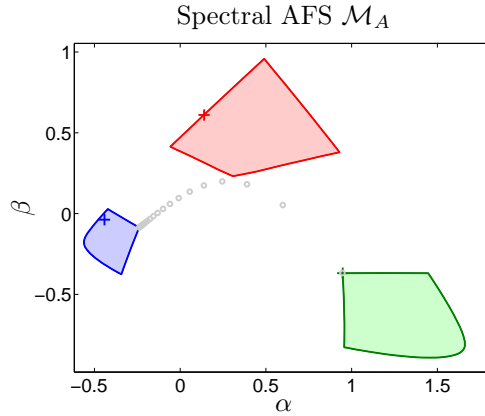


Figure 3: The spectral AFS \mathcal{M}_A for the model problem from Section 2.1. The three markers (+) in green, red and blue indicate the components of the initial NMF. These three points are the starting points for the polygon inflation method. Furthermore, the series of points marked by the symbol \circ represent the single spectra, which are rows of the spectral data matrix D .

Each of the three segments of the spectral AFS \mathcal{M}_A is approximated by an adaptively generated polygon. The AFS \mathcal{M}_C for the concentration factor C can be computed in a similar way; then the complete algorithm is applied to the transposed data matrix D^T . In some cases the AFS consists of only one segment with a hole or of more than three segments. Then the AFS is computed by a modification of the method which is called the inverse polygon inflation algorithm [19]. In this modified algorithm first the superset FIRPOL of the AFS is computed and in a second step points not belonging to the AFS are eliminated by inflation a second polygon in the interior of FIRPOL.

2.3 Reduction of the rotational ambiguity by the complementarity theorem

The *bilinear* Lambert-Beer model $D = CA$ results in *two* factors and so *two* areas of feasible solutions can be constructed. The spectral AFS \mathcal{M}_A has been considered in Section 2.2. The concentrational AFS \mathcal{M}_C can be constructed in the same way if the algorithm is applied to the transposed data matrix D^T . The simultaneous representation of \mathcal{M}_A and \mathcal{M}_C can be very advantageous. In this way restrictions can be analyzed of the partial knowledge of one factor on the other factor. This mutual

dependence of the factors is a consequence of the factorization

$$D = U\Sigma V^T = \underbrace{U\Sigma T^{-1}}_C \underbrace{TV^T}_A.$$

where partial knowledge of A implies some restrictions on the matrix elements of the $s \times s$ matrix T , this implies some restrictions on T^{-1} and so the factor C is to some extent predetermined. Therein s is the rank of D , which is the number of independent components. A systematic analysis of these restrictions appeared in [17] and has led to the complementarity theorem. The application of the complementarity theorem to the AFS is analyzed in [20], see also [3].

For the important case of an $s = 3$ -component system a given point in one AFS (either a given spectrum or concentration profile) is associated with a straight line in the other AFS. The form of this straight line is specified by the complementarity theorem. Only points which are located on this line and which are elements of the AFS are consistent with the pre-given information. In case of a four-component system a given point in \mathcal{M}_A (or \mathcal{M}_C) is associated with a plane in \mathcal{M}_C (or \mathcal{M}_A) and so on.

For three-component systems this simultaneous representation of \mathcal{M}_A and \mathcal{M}_C together with a representation of the restrictions of the AFS by the complementarity theorem is a functionality of the **Complementarity & AFS** module of *FAC-PACK*. This module, its options and results are explained in the users' guide in Section 3.1.3. This module gives the user the opportunity to fix a number of up to three points in the AFS. These points correspond to known spectra or known concentration profiles. For example such information can be available for the reactants or for the final products of a chemical reaction. Typically any additional information on the reaction system can be used in order to reduce the rotational ambiguity for the remaining components. Within the AFS computation module the locking of spectra (or concentration profiles) results in smaller AFS segments for the remaining components, see Section 3.5.6. Here we focus on the simultaneous representation of the spectral and the concentration AFS and the mutual restrictions of the AFS by adding partial information on the factors. Up to three points (for a three-component system) in an AFS can be fixed. These three points determine a triangle in this AFS and also a second triangle in the other AFS. These two triangles correspond to a factorization of D into nonnegative factors C and A . All this is explained in the users' guide in Section 3.6. The reader is invited to experiment with the MATLAB GUI of *FAC-PACK*, to move the vertices of the triangles through the AFS by using the mouse pointer and to watch the resulting changes for all spectra and concentration profiles. This interactive graphical representation of the system provides the user with the full control of the factors and supports the selection of a chemically meaningful factorization.

In Figure 4 the results of the application of the **Complementarity & AFS** module are shown for the sample problem from Section 2.1. The four rows of this figure show the following:

1. First row: A certain point in the AFS \mathcal{M}_A is fixed and the associated spectrum $A(1, :)$ is shown. This point is associated with a straight line in the concentrational AFS \mathcal{M}_C . This line intersects two segments of \mathcal{M}_C . The complementary concentration profiles $C(:, 2)$ and $C(:, 3)$ are restricted to the two continua of the intersection. Up to now no concentration profile is uniquely determined.
2. Second row: A second point in \mathcal{M}_A is fixed and this second spectrum $A(2, :)$ is shown (green color). A second straight line (green) is added to \mathcal{M}_C by the complementarity theorem. The intersection of these two lines in \mathcal{M}_C uniquely determines the complementary concentration $C(:, 3)$.
3. Third row: Three points are fixed in \mathcal{M}_A which uniquely determines the factor A . The triangle in \mathcal{M}_A corresponds with a second triangle in \mathcal{M}_C . All this determines a nonnegative factorization $D = CA$.

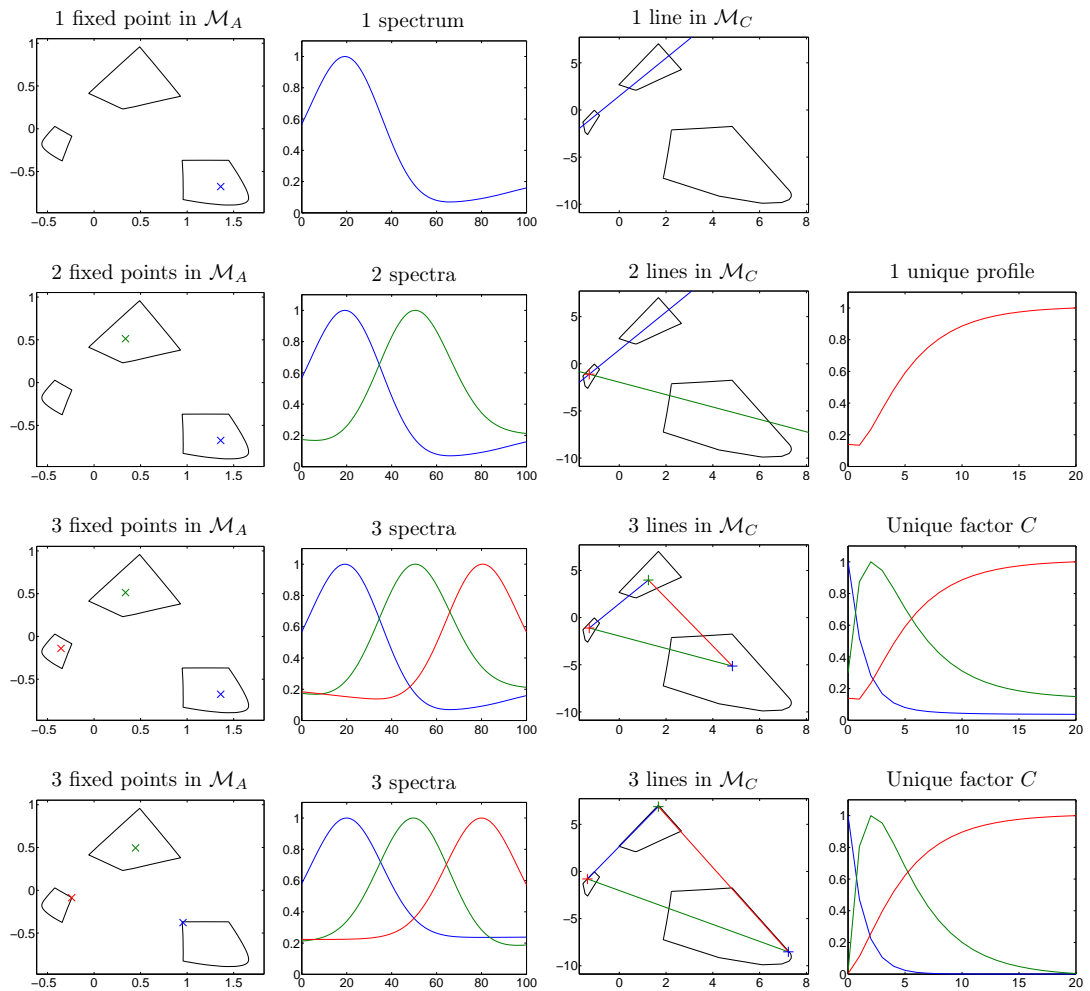


Figure 4: Successive reduction of the rotational ambiguity. First row: one fixed spectrum is associated with a straight line in \mathcal{M}_C . Second row: two fixed spectra in \mathcal{M}_A result in two straight lines in \mathcal{M}_C . The point of intersection uniquely determines one concentration profile. Third row: three fixed points in \mathcal{M}_A uniquely determine a complete factorization $D = CA$. Fourth row: This complete factorization reproduces the original factors from Section 2.1.

4. Fourth row: A second nonnegative factorization is shown, which exactly reproduces the original components of the model problem from Section 2.1.

2.3.1 Soft constraints and the AFS

Multivariate curve resolution methods often use soft-constraints in order to favor solutions which are particularly smooth, monotone, unimodal or that have other comparable properties. While moving vertices of triangle-factors interactively through the AFS the user might be interested to see where in the AFS solutions with certain properties can be found.

To this end the **Complementarity & AFS** module of *FAC-PACK* provides the option to plot the level set contours for certain constraint functions within the AFS window. Contour plots are available for estimating the

- monotonicity of the concentration profiles (in \mathcal{M}_C),
- the smoothness of spectra (in \mathcal{M}_A) or concentration profiles (in \mathcal{M}_C),
- the Euclidean norm of the spectra (in \mathcal{M}_A) in order to estimate the total absorbance of the spectra,
- and the closeness to an exponentially decaying function (in \mathcal{M}_C) in order to find concentration profiles of reactants which are degraded in a first order reaction.

In Figure 5 two contour plots are shown. First, in \mathcal{M}_A the closeness of a concentration profile to an exponentially decaying function is shown. The smallest distance, which is shown by the lightest grey has the coordinates $(\alpha, \beta) \approx (7.2714, -8.5516)$. This point is the same which is shown in the fourth row and third column of Figure 4 and which can be identified with the reactant X in the model problem from Section 2.1. Second, the contours of the Euclidean norm of the spectra within the \mathcal{M}_A window. These contours serve to estimate the total absorbance of the spectra and can for instance be used in order to identify spectra with few and narrow peaks.

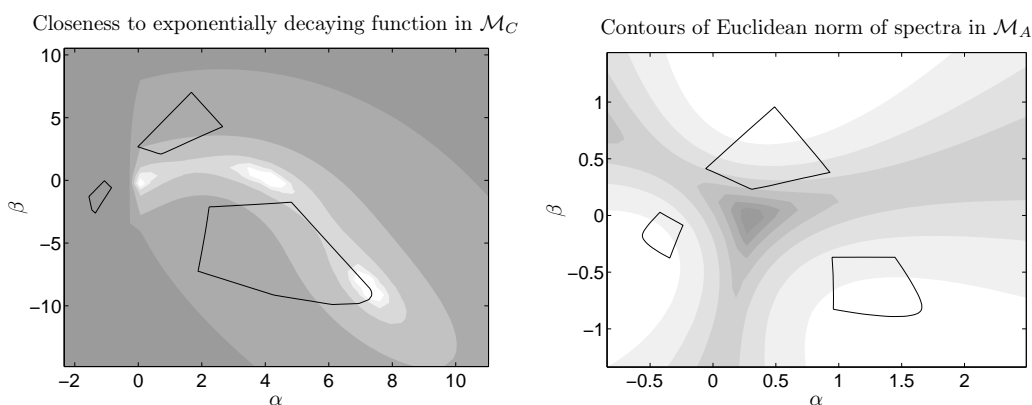


Figure 5: Contour plots within the (α, β) -plane close to the areas of feasible solutions \mathcal{M}_C and \mathcal{M}_A for the data from Section 2.1. Left: Contours of closeness of a concentration profile to an exponentially decaying function in the \mathcal{M}_C window. Right: The contour plot of the Euclidean norm (total absorbance) of the pure component spectra within the \mathcal{M}_A window.

3 Users' guide to *FAC-PACK*

Here the users' guide on the revision 1.1 of *FAC-PACK* follows.

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.

3.1 Quick start

Download the *FAC-PACK* software from

<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*, see Figure 6.

3.1.1 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 Figure 7.
- 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.

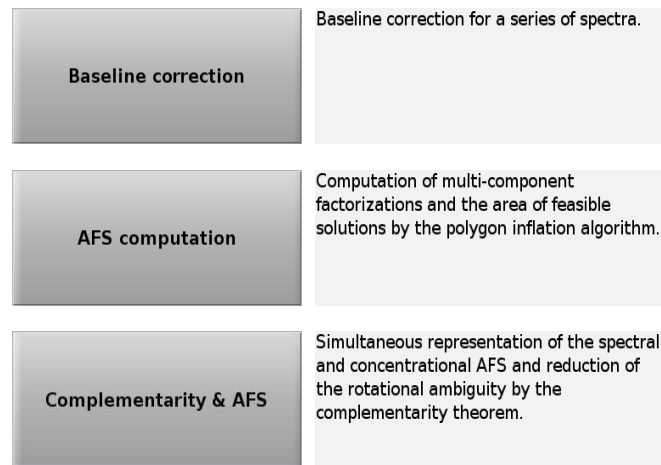


Figure 6: The start window of *FAC-PACK*

- Step 4: Select the correction method, e.g. polynomial degree 2.
 Step 5: Compute the corrected baseline.
 Step 6: Save the corrected data.

3.1.2 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 is shown in Figure 9:

- 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.

3.1.3 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

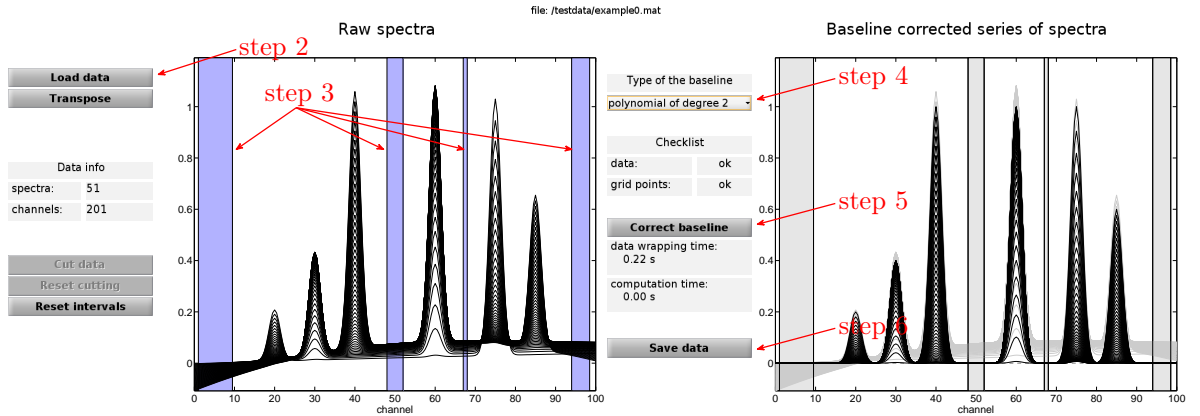


Figure 7: Baseline correction within 6 steps.

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.

In the following sections the functionalities of *FAC-PACK* are explained in detail.

3.2 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

$$D \approx CA. \quad (1)$$

Some references on MCR techniques are [10, 8, 12, 11, 15]. Typically the factorization (1) does not

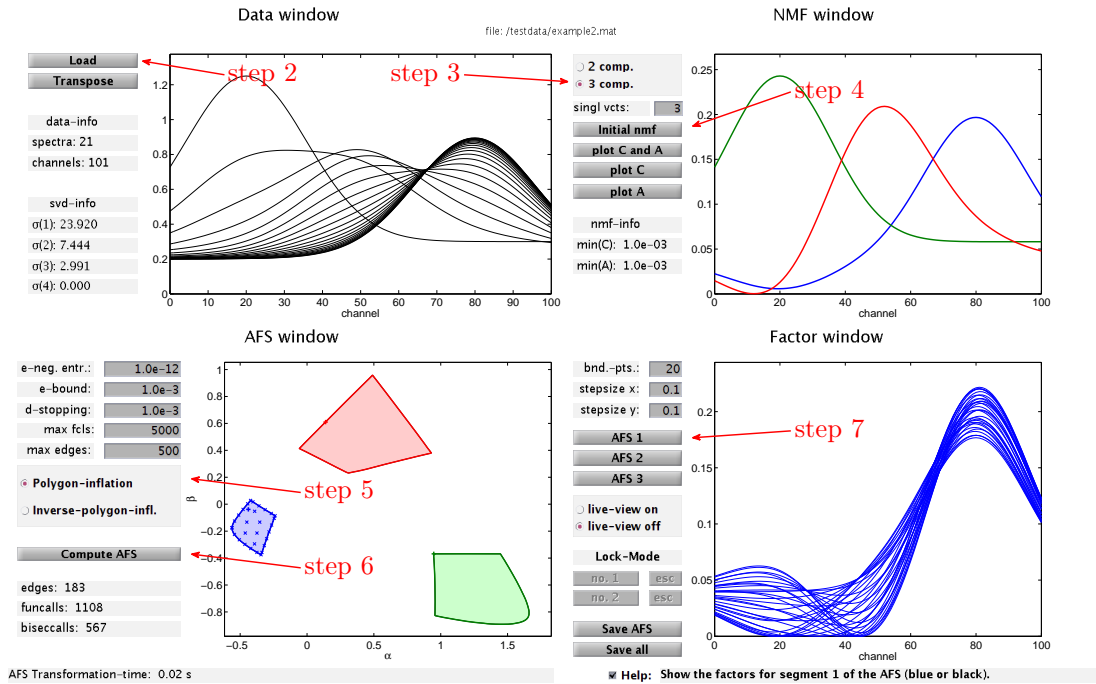


Figure 8: Quickstart in 6 steps.

result in unique nonnegative matrix factors C and A . Instead a continuum of possible solutions exists [12, 22, 9]; this non-uniqueness is called the rotational ambiguity of MCR solutions. Sometimes additional information can be used to reduce this rotational ambiguity, see [5, 16] 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 [21, 14, 22, 1]. For a numerical computation of the AFS two methods have been developed: the triangle-boundary-enclosure scheme [6] and the polygon inflation method [18]. *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

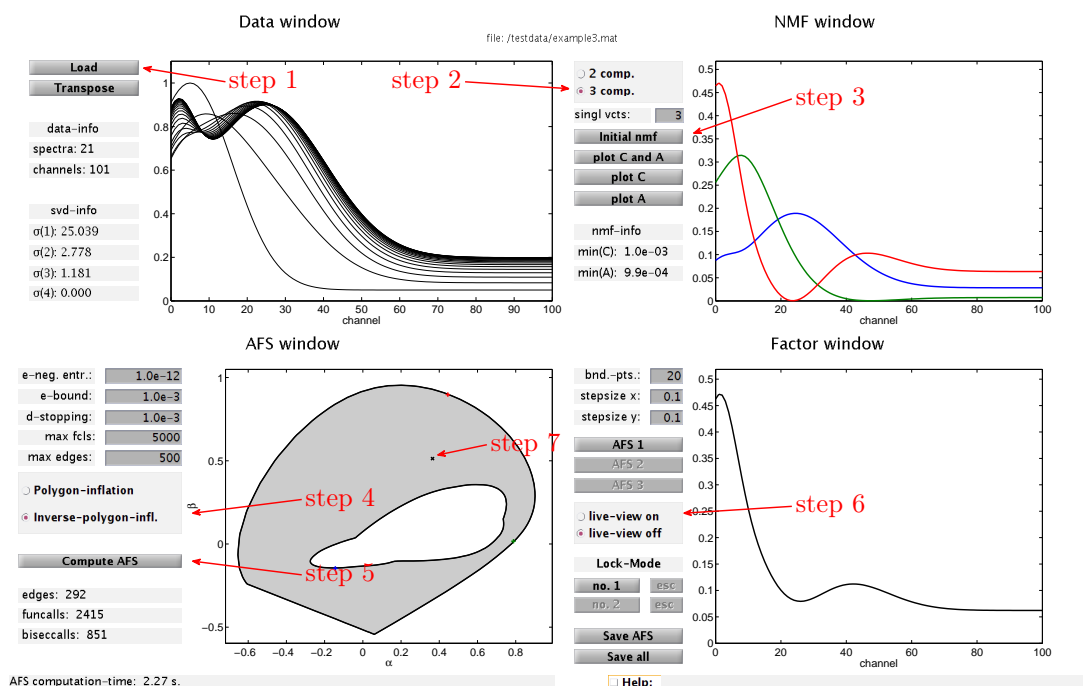


Figure 9: Quick start with live-view mode in 7 steps.

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.

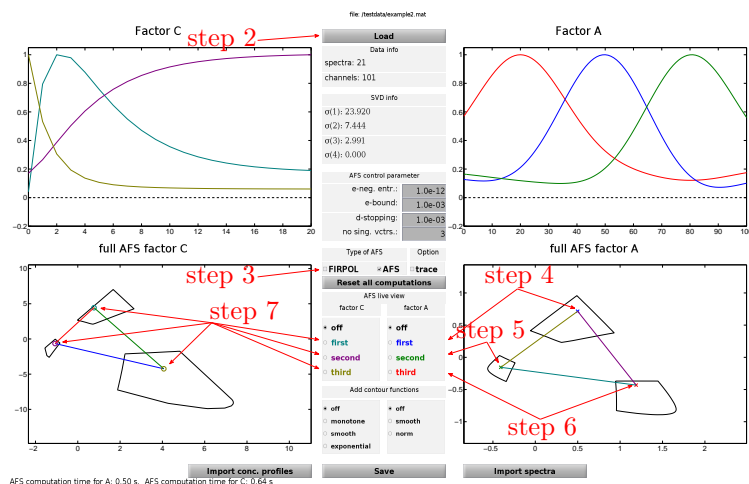


Figure 10: The complementarity & AFS module.

3.3 Get ready to start

Please download *FAC-PACK* from

<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.

3.3.1 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 [17, 20].

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.

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.

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.

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.

3.3.2 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 .

¹Red numbers refer to Section 3.7.

3.3.3 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.

3.3.4 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 ε -bound (38) or δ -stopping (39).

3.3.5 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>.

3.4 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 32. Data loading is explained in Section 3.3.2. 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 11 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.

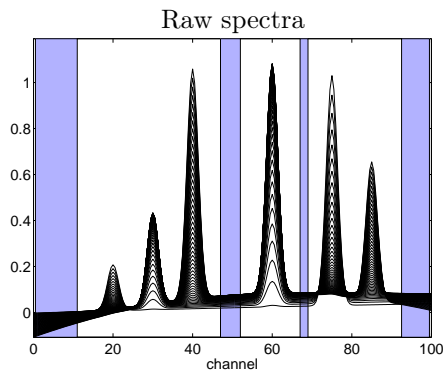


Figure 11: Four zero-intervals are marked by blue columns. In these intervals the spectral data is corrected towards zero. Data: `example0.mat`.

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 $a_{\text{new}} = a - b$. 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).

3.4.1 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.

3.4.2 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).

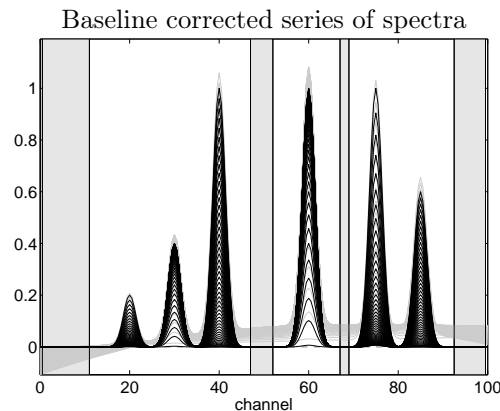


Figure 12: Sample problem example0.mat with corrected baseline by a 2nd order polynomial.

3.5 AFS computation module

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

3.5.1 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 3.3.2. 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 $\sigma_1, \dots, \sigma_4$. 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 3.3.3. Figure 13 shows the series of spectra from `example2.mat`. Three clearly nonnegative singular values indicate a three-component system.

3.5.2 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 .

3.5.3 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 [13], 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 [18] 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 ε ; see Equation (6) in [18] for details.

The initial NMF provides a number of s interior points of the AFS; see Equations (3) and (5) in [18].

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 14 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.:** $1 \cdot 10^{-12}$.

3.5.4 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 α and β where

$$T = \begin{pmatrix} 1 & \alpha \\ 1 & \beta \end{pmatrix}$$

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 (α, β) so that

$$T = \begin{pmatrix} 1 & \alpha & \beta \\ 1 & s_{11} & s_{12} \\ 1 & s_{21} & s_{22} \end{pmatrix}$$

is a transformation which is associated with nonnegative factors C and A , see [18]. A permutation argument shows that with (α, β) the points (s_{11}, s_{12}) and (s_{21}, s_{22}) belong to the AFS, too.

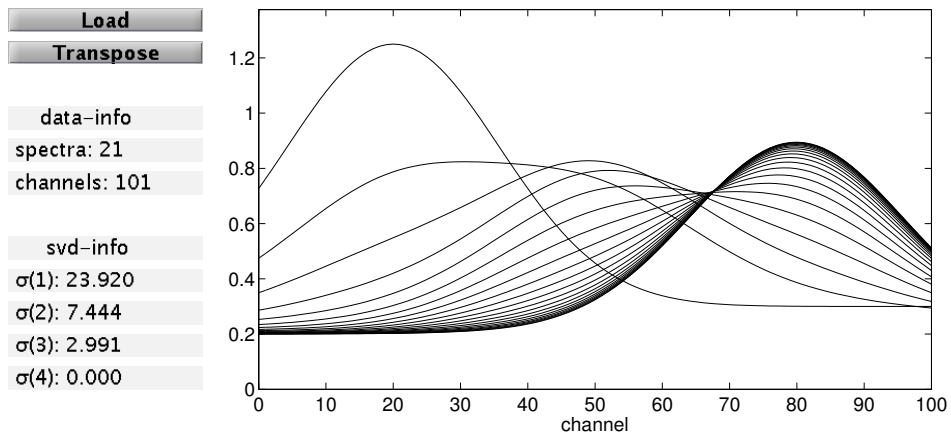


Figure 13: Test matrix example2.mat loaded by pressing the **Load** button (18).

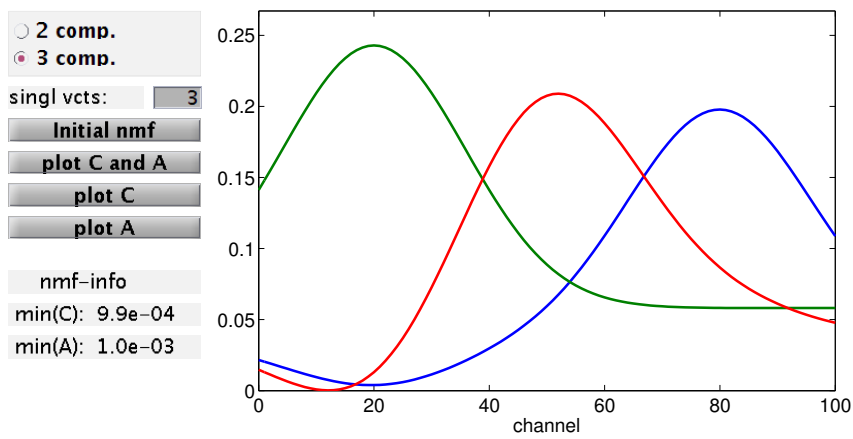


Figure 14: The initial NMF for example2.mat with $s = 3$ components.

The initial NMF is the starting point for the numerical computation of the AFS since it provides first α and β 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.

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 ε in [18]. 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 ε_b in [18]. Decreasing the value ε_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 δ in Section 3.5 of [18] 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.

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 [18] 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 15 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 16 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 [20].

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.

3.5.5 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.

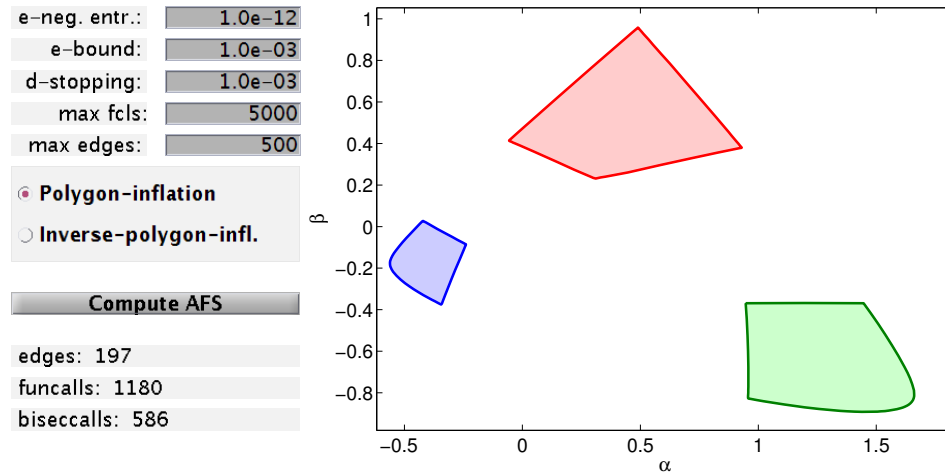


Figure 15: An AFS with three clearly separated segments as computed by the "classical" polygon inflation algorithm. Data: example2.mat.

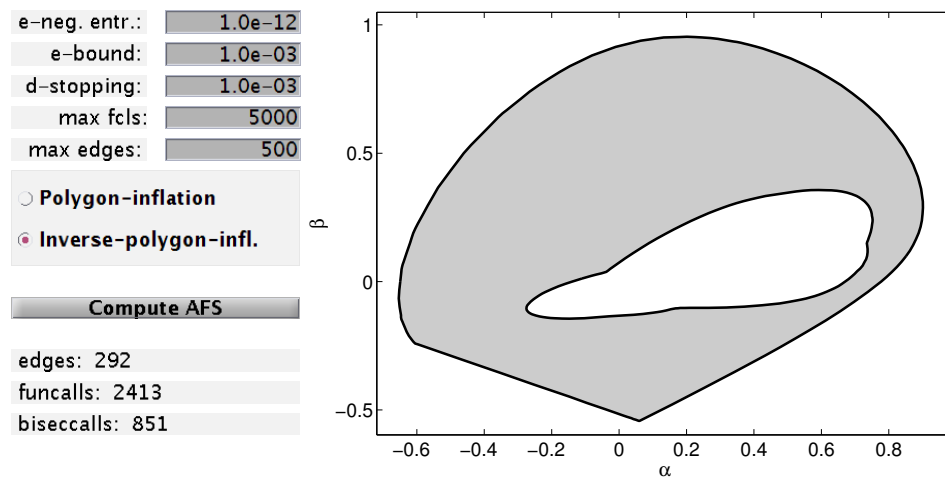


Figure 16: An AFS which consists of only one segment with a hole. For the computation the inverse polygon inflation algorithm has been applied to example3.mat.

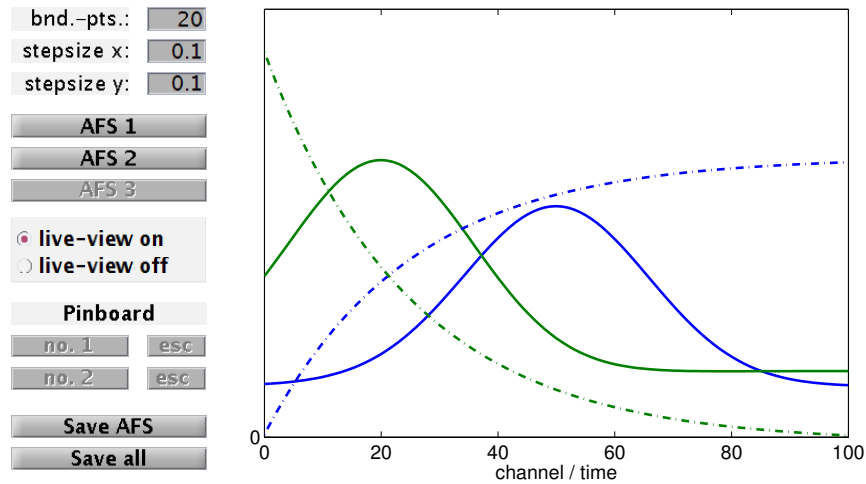


Figure 17: The live-view mode for the two-component system `example1.mat`. By moving the mouse pointer through the AFS the transformation to the factors C and A is computed interactively and all results are plotted in the factor window. The spectra are drawn by solid lines and the concentration profiles are represented by dash-dotted lines.

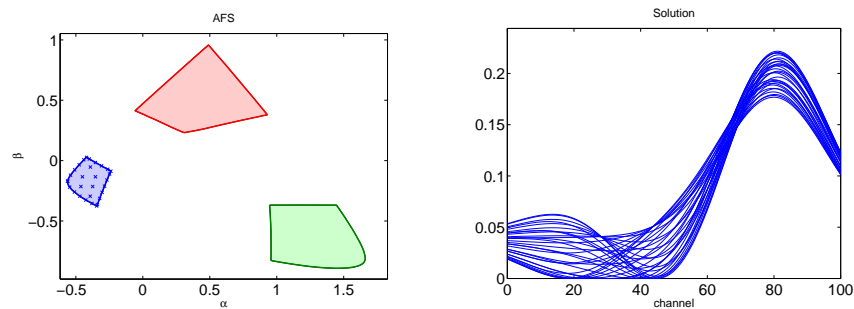


Figure 18: The AFS for the problem `example2.mat` consists of three separated segments. The blue segment of the AFS is covered with grid points (dark blue points). For each of these points the associated spectrum is plotted in the right figure.

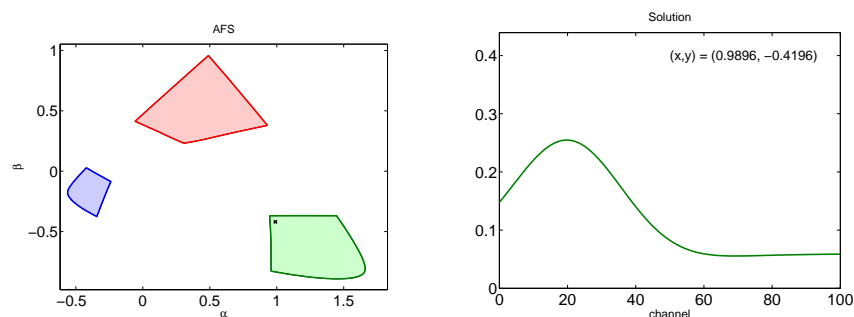


Figure 19: The live-view mode is active for the 3 component system `example2.mat`. In the green AFS-segment the mouse pointer is positioned in the left upper corner at the coordinates $(0.9896, -0.4196)$. The associated spectrum is shown in the right figure together with the mouse pointer coordinates.

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).

Live-view mode: 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 17 shows a typical screen shot. The y -axis labels are turned off.

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 18. 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.

Live-view mode: 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 19.

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 3.5.6 for further explanations.

3.5.6 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.

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

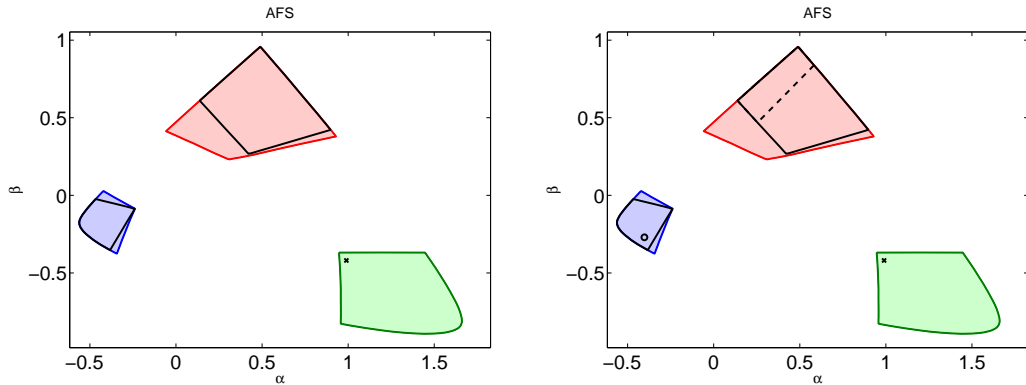


Figure 20: Reduction of the AFS for the test problem `example2.mat`. Left: the reduced AFS is shown by black solid lines after locking a first spectrum (marker \times). Right: The AFS of the remaining third component is shown by a black broken line.

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.

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 20 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 3.6.

3.5.7 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 C_{init} , A_{init} 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 3.5.6, 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.

3.5.8 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 3.3.4. 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.

3.5.9 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.

3.6 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 [19, 20] 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 $D = CA$.

3.6.1 Initial steps

The Complementarity & AFS 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 3.3.2. 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 3.5.3.

In Figure 13 we consider the sample data *example2.mat*, cf. Section 3.3.3. The parameters in the fields (75)-(77) control the AFS computation. The meaning of these parameters is explained in Section 3.5.4.

3.6.2 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 [19]. Points in \mathcal{M}_A^+ represent only one nonnegative spectrum $A(1, \cdot)$. 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 [18, 19]. Numerical calculations are done by the C-routine `AFScomputationSYSTEMNAME`, see Section 3.3.4, immediately after a selection is made in the checkbox (79). The results are shown in the windows (65) and (67).

Figure 21 shows FIRPOL \mathcal{M}_A^+ and the AFS \mathcal{M}_A for the sample data **example2**. The trace of D , see Section 3.6.2, is also shown. The triangle in \mathcal{M}_A represents a complete feasible factor A .

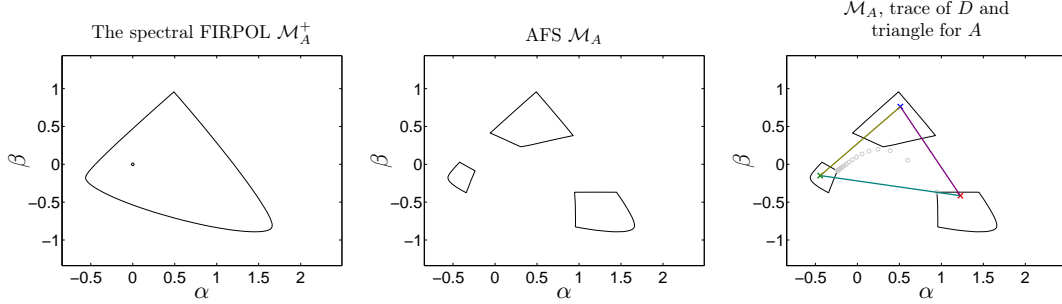


Figure 21: Sample data **example2**. Left: FIRPOL \mathcal{M}_A^+ . Center: \mathcal{M}_A . Right: \mathcal{M}_A with the trace of D marked by gray circles and a triangle representing a feasible factor A .

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

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

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 $j = 1, \dots, n$. 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.

3.6.3 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 .

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 22.

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 23.

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 24 illustrates this.

3.6.4 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

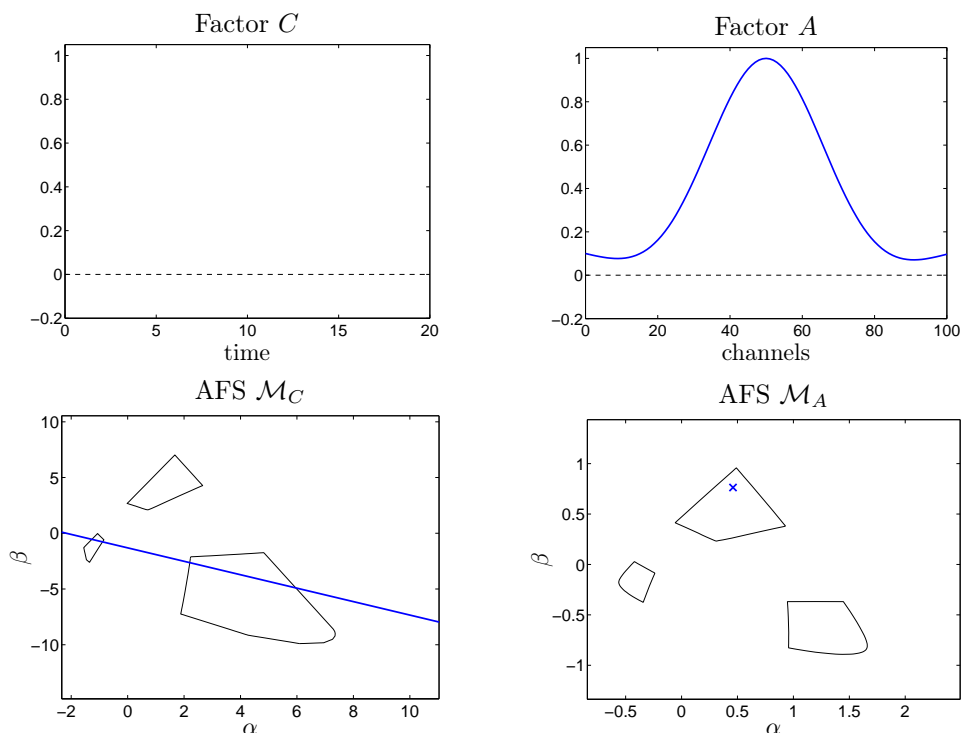


Figure 22: A first point, marked by \times , is locked in the AFS \mathcal{M}_A (lower-right). This results in: Upper-left: no concentration profile has uniquely been defined up to now. Upper-right: a single pure component spectrum has been fixed. Lower-left: the blue straight line is a restriction on the complementary concentration profiles $C(:, 2 : 3)$ in \mathcal{M}_C .

- (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 [17] for the complementarity theorem and [3, 19, 20] 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.

3.6.5 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

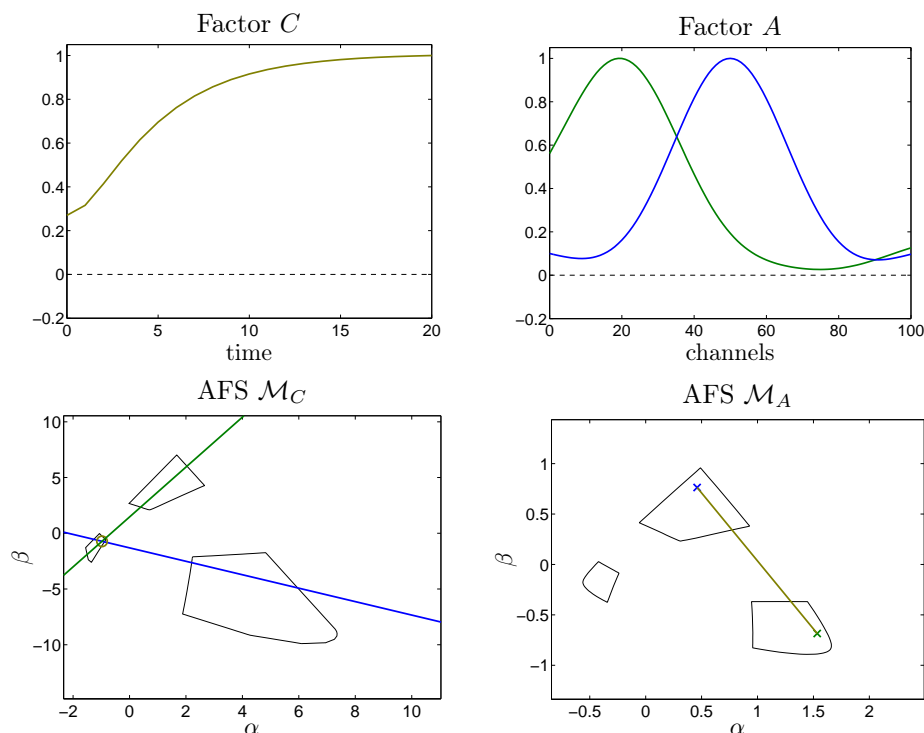


Figure 23: A second point, marked by \times , is locked in the AFS \mathcal{M}_A (lower-right). This results in: Upper-left: one complementary concentration profile is uniquely determined. Upper-right: the two locked pure component spectra. Lower-left: the blue straight line restricts the complementary 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 \circ is now uniquely determined.

the Complementarity & AFS module any point in the (α, β) -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 (α, β) -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

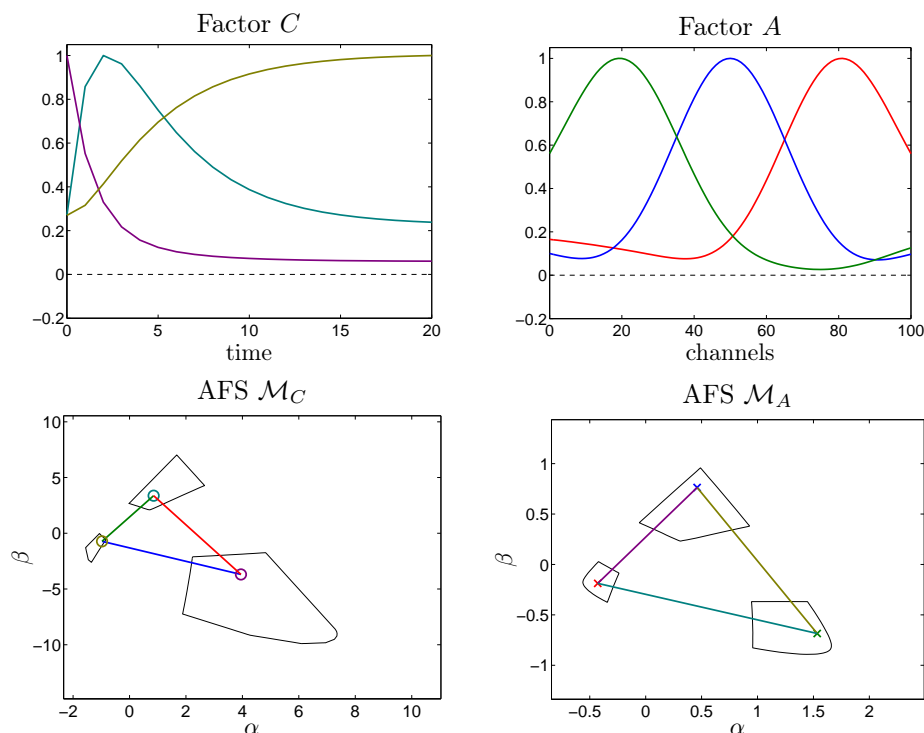


Figure 24: Three locked points in \mathcal{M}_A uniquely define the factor A represented by a feasible triangle (lower-right). This results in: Upper-left: three complementary concentration profiles. Upper-right: three pure component spectra. Lower-left: a triangle in the AFS \mathcal{M}_C .

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 25 shows two examples.

3.6.6 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 ℓ 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 $*$.

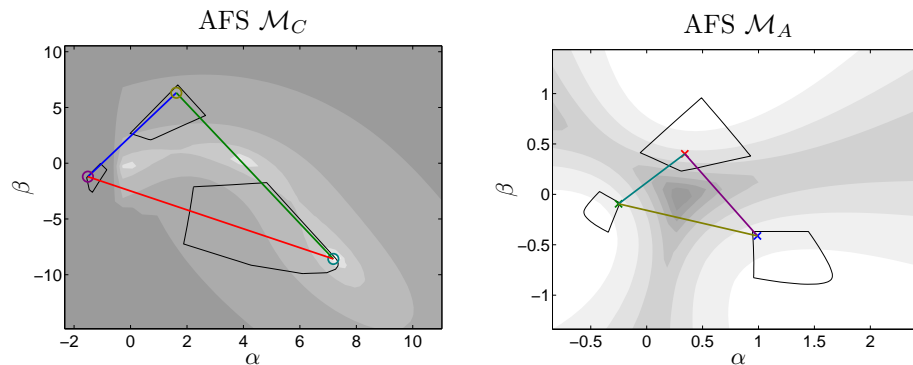


Figure 25: Contour plots for two soft constraints. Left: Contour evaluating the closeness to an exponentially decaying function. The right lower vertex (marked by \circ) represents an exponentially decaying concentration profile. Right: Contour plot for the Euclidean norm of the spectrum for `example2.mat`.

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.

3.6.7 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 \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.

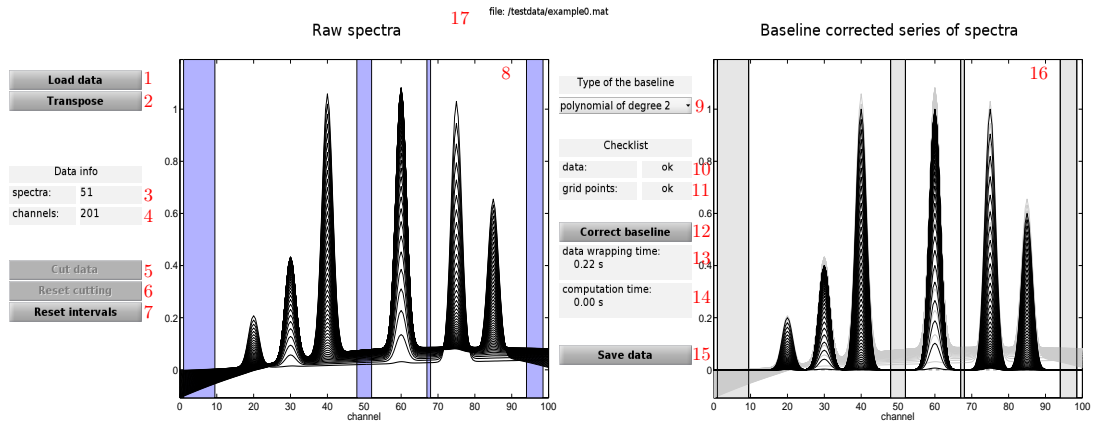
Any figure window in the GUI can be accessed by clicking the right mouse button in the desired figure. Then a separate MATLAB figure opens. The figure can now be modified, printed or exported in the usual way.

3.6.8 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.

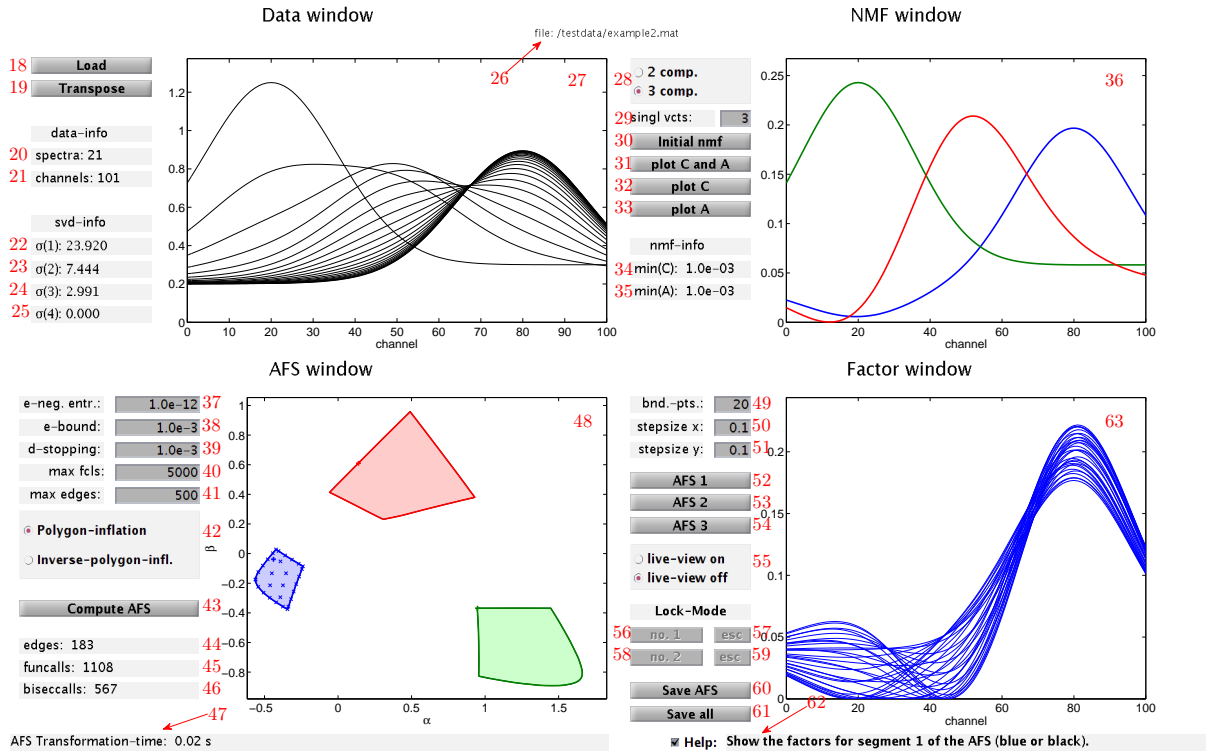
3.7 Screen shots & description of operation panels

I: Screen shot of the **Baseline-correction** module:



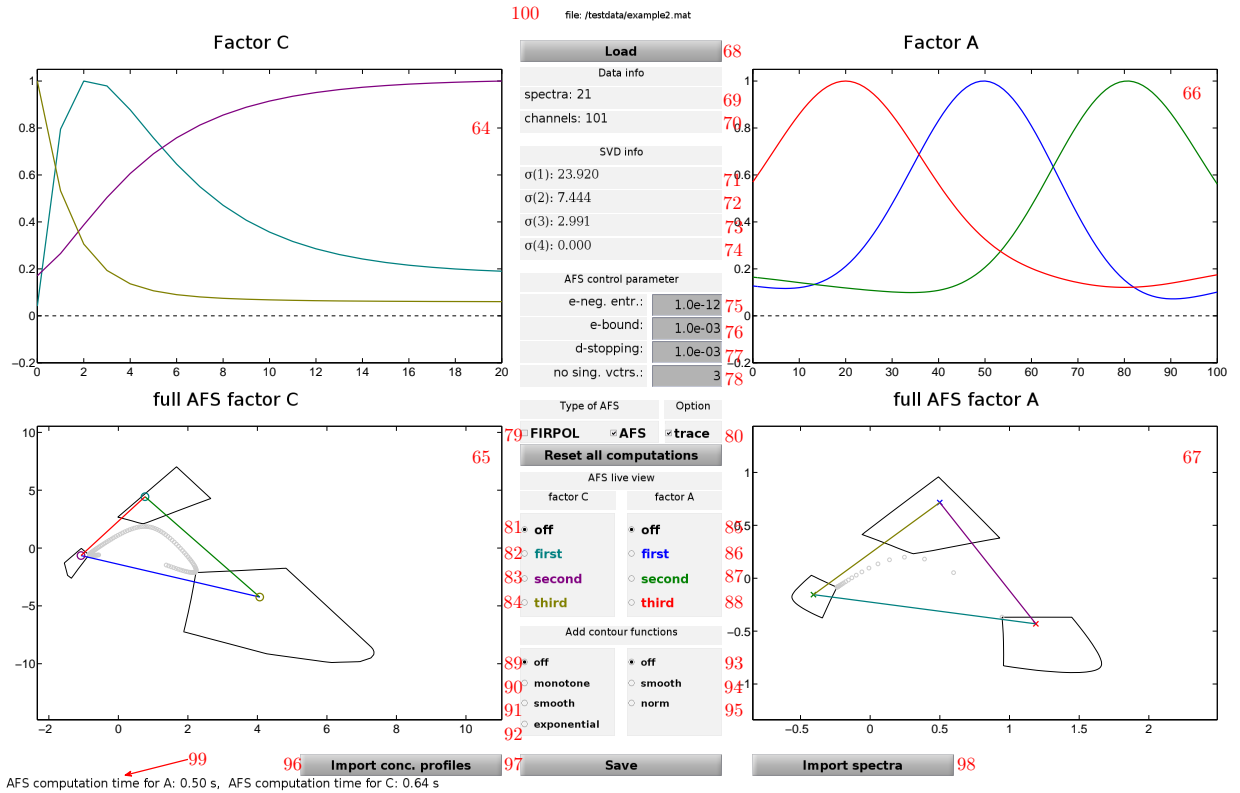
The GUI buttons, data fields and figures:

- | | |
|----------------------------------|---|
| 1 Load data matrix D | 10 Status of data consistency |
| 2 Transpose D | 11 Status of band selection consistency |
| 3 # rows of D | 12 Apply baseline correction |
| 4 # columns of D | 13 Computing time for data wrapping |
| 5 Cut spectra as marked by mouse | 14 Computing time for baseline correction |
| 6 Reset spectral cutting | 15 Save button |
| 7 Reset selected intervals | 16 Corrected spectra window |
| 8 Raw 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 | 37 Acceptable negativeness for the NMF | 51 Step size y direction for the factor representation |
| 19 Transpose button | 38 Precision at the boundary | 52 Plot factors for segment 1 of the AFS |
| 20 # rows of data matrix D | 39 Stopping precision | 53 Plot factors for segment 2 of the AFS |
| 21 # columns in D | 40 Maximal # calls of the target function | 54 Plot factors for segment 3 of the AFS |
| 22 Singular value 1 of D | 41 Maximal # of edges | 55 Activate live-view mode |
| 23 Singular value 2 of D | 42 Select type of polygon inflation | 56 Compute AFS for 1 selected point |
| 24 Singular value 3 of D | 43 Compute AFS | 57 Reset 1. restricted AFS |
| 25 Singular value 4 of D | 44 # edges | 58 Compute AFS for 2 selected points |
| 26 File name | 45 # calls of target function | 59 Reset 2. restricted AFS |
| 27 Initial spectra window | 46 No. used function calls for the boundary-precision | 60 Save button (AFS) |
| 28 Input # components | 47 Output on the numerical computation | 61 Save button (all) |
| 29 # of singular values used for initial NMF | 48 AFS window | 62 Help field |
| 30 Compute NMF | 49 # of boundary points for the factor representation | 63 Factor window |
| 31 Plot factors C and A | 50 Step size x direction for the factor representation | |
| 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 | | |

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

The GUI buttons, data fields and figures:

- | | | |
|---|---|---|
| 64 Factor C | 79 Select AFS (FIRPOL and/or full AFS) | 90 Contour mode for C - monotone profiles |
| 65 AFS for C | 80 Mark mixed spectra (rows of D) in the AFS | 91 Contour mode for C - smooth profiles |
| 66 Factor A | 81 Live-view mode C off | 92 Contour mode for C - exponential decay |
| 67 AFS for A | 82 Live-view mode $C(:, 1)$ on | 93 Contour mode for A off |
| 68 Load spectral data matrix D | 83 Live-view mode $C(:, 2)$ on | 94 Contour mode for A - smooth spectra |
| 69 # rows of D | 84 Live-view mode $C(:, 3)$ on | 95 Contour mode for A - spectra with a small norm |
| 70 # columns of D | 85 Live-view mode A off | 96 Import external concentration profiles |
| 71 Singular value 1 of D | 86 Live-view mode $A(1, :)$ on | 97 Save the results |
| 72 Singular value 2 of D | 87 Live-view mode $A(2, :)$ on | 98 Import external spectra |
| 73 Singular value 3 of D | 88 Live-view mode $A(3, :)$ on | 99 Computing time |
| 74 Singular value 4 of D | 89 Contour mode C off | 100 Name of data file |
| 75 Acceptable negativeness for the NMF | | |
| 76 Precision at the boundary | | |
| 77 Stopping precision | | |
| 78 # of singular vectors used for initial NMF | | |

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*, *Anal. Chem.* **81** (2009), no. 6, 2115–2122.
- [2] H. Abdollahi and R. Tauler, *Uniqueness and rotation ambiguities in Multivariate Curve Resolution methods*, *Chemom. Intell. Lab. Syst.* **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*, *Anal. Chim. 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*, *Anal. Chim. Acta* **174** (1985), 1–26.
- [5] A. de Juan, M. Maeder, M. Martínez, and R. Tauler, *Combining hard and soft-modelling to solve kinetic problems*, *Chemometr. Intell. Lab.* **54** (2000), 123–141.
- [6] A. Golshan, H. Abdollahi, and M. Maeder, *Resolution of Rotational Ambiguity for Three-Component Systems*, *Anal. Chem.* **83** (2011), no. 3, 836–841.
- [7] A. Golshan, M. Maeder, and H. Abdollahi, *Determination and visualization of rotational ambiguity in four-component systems*, *Anal. Chim. Acta* **796** (2013), no. 0, 20–26.
- [8] J.C. Hamilton and P.J. Gemperline, *Mixture analysis using factor analysis. II: Self-modeling curve resolution*, *J. Chemom.* **4** (1990), no. 1, 1–13.
- [9] J. Jaumot and R. Tauler, *MCR-BANDS: A user friendly MATLAB program for the evaluation of rotation ambiguities in Multivariate Curve Resolution*, *Chemom. Intell. Lab. Syst.* **103** (2010), no. 2, 96–107.
- [10] W.H. Lawton and E.A. Sylvestre, *Self modelling curve resolution*, *Technometrics* **13** (1971), 617–633.
- [11] M. Maeder and Y.M. Neuhold, *Practical data analysis in chemistry*, Elsevier, Amsterdam, 2007.
- [12] E. Malinowski, *Factor analysis in chemistry*, Wiley, New York, 2002.
- [13] K. Neymeyr, M. Sawall, and D. Hess, *Pure component spectral recovery and constrained matrix factorizations: Concepts and applications*, *J. Chemom.* **24** (2010), 67–74.
- [14] R. Rajkó and K. István, *Analytical solution for determining feasible regions of self-modeling curve resolution (SMCR) method based on computational geometry*, *J. Chemom.* **19** (2005), no. 8, 448–463.
- [15] C. Ruckebusch and L. Blanchet, *Multivariate curve resolution: A review of advanced and tailored applications and challenges*, *Anal. Chim. Acta* **765** (2013), 28–36.
- [16] 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. Chemom.* **26** (2012), 538–548.
- [17] M. Sawall, C. Fischer, D. Heller, and K. Neymeyr, *Reduction of the rotational ambiguity of curve resolution techniques under partial knowledge of the factors. Complementarity and coupling theorems*, *J. Chemom.* **26** (2012), 526–537.
- [18] 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. Chemom. **27** (2013), 106–116.
- [19] M. Sawall and K. Neymeyr, *A fast polygon inflation algorithm to compute the area of feasible solutions for three-component systems. II: Theoretical foundation, inverse polygon inflation, and FAC-PACK implementation*, J. Chemom. **28** (2014), 633–644.
- [20] ———, *On the area of feasible solutions and its reduction by the complementarity theorem*, Anal. Chim. Acta **828** (2014), 17–26.
- [21] R. Tauler, *Calculation of maximum and minimum band boundaries of feasible solutions for species profiles obtained by multivariate curve resolution*, J. Chemom. **15** (2001), no. 8, 627–646.
- [22] M. Vosough, C. Mason, R. Tauler, M. Jalali-Heravi, and M. Maeder, *On rotational ambiguity in model-free analyses of multivariate data*, J. Chemom. **20** (2006), no. 6-7, 302–310.