

Tutorial of Asymmetric Logistic Peak Adjustment (ALPA): a Matlab program for the multivariate resolution of highly asymmetric voltammetric signals

Supplementary Material of the work:

Asymmetric logistic peak as a suitable function for the resolution of highly asymmetric voltammograms in non-bilinear systems

by M.Kooshki, J.M. Díaz Cruz*, H. Abdollahi, C. Ariño and M. Esteban

* Departament de Química Analítica. Universitat de Barcelona
Martí i Franquès, 1-11, E – 08028 – Barcelona (Spain)
E-mail: josemanuel.diaz@ub.edu

FILES INVOLVED

In order to be able to run **alpa** and/or **peakmaker** from Matlab, the following files must be copied to a directory included in the Matlab path:

peakmaker.m Matlab program for the visual initial estimation of pure voltammograms.

alpa.m Matlab program for the fitting of experimental data matrices by means of ALPA

alpacalc1.m set of Matlab programs which are used by **alpa.m**
alpacalc2.m
integrator.m

SAMPLE DATA

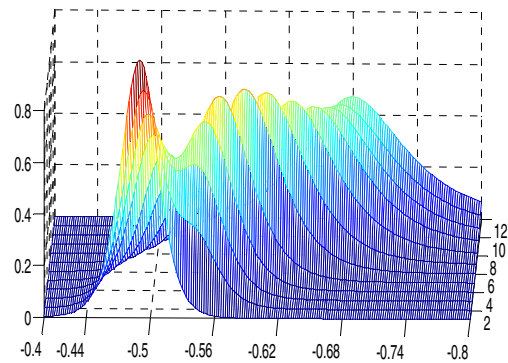
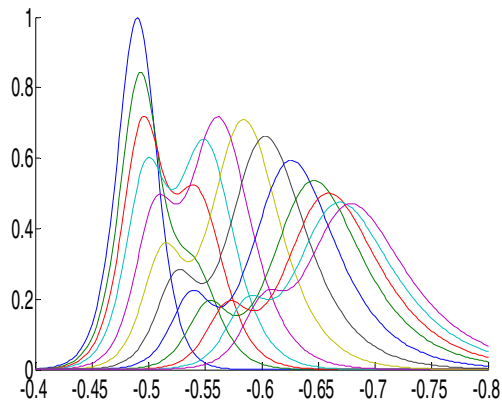
The workspace Matlab files **sample.mat** and **sample2.mat** contain simulated data to practise some aspects of the ALPA procedure according to the present tutorial.

ANALYSIS OF SAMPLE DATA

When you load the file **sample.mat** in Matlab you can find the following matrices:

I	(201 rows, 12 columns)	current data matrix including 12 voltammograms simulated for 201 potential values
E	(201 rows, 1 column)	column vector with the potential values
n	(12 rows, 1 column)	column vector with the number of the voltammograms

You can look at the data matrix in both 2D and 3D ways by means of the commands
`plot (E, I)` `mesh (n, E, I)`



Before starting the analysis of the **I** matrix, it is necessary to provide a reference pure signal for every component. This can be done in many ways. In the sample file, the reference signal matrix **esv** has been obtained from the visual inspection of the experimental matrix using the **peakmaker** program. This program can be invoked in the way:

```
esv=alpeakmaker(I);
```

Then, you get the mask:

```
INITIAL ESTIMATION OF PEAK-SHAPED FUNCTIONS
```

You are asked about the symmetry of the peak (1 for symmetric, 0 for asymmetric) and after selecting it you get the message:

```
Symmetric peak? yes (1), no (0) 1
```

You get a Figure with a cyan 2D plot of the experimental matrix. Then, you have to click with mouse the top of every component you want to consider, which defines height of the peaks.

Select maxima of the peak with mouse and press enter

Then, you have to click with mouse the top of every component you want to consider, which defines height of the peaks.

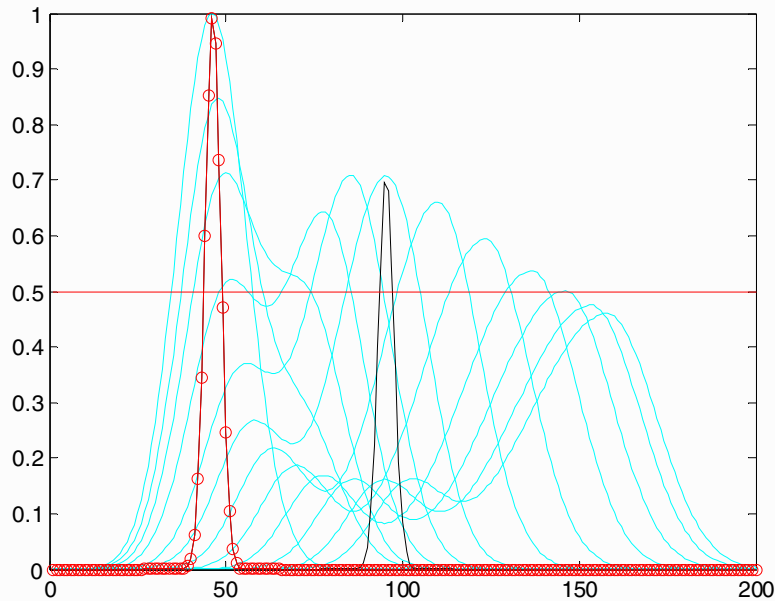
Then you are asked to press enter and select the width of each component.

Select peak width with mouse and press enter

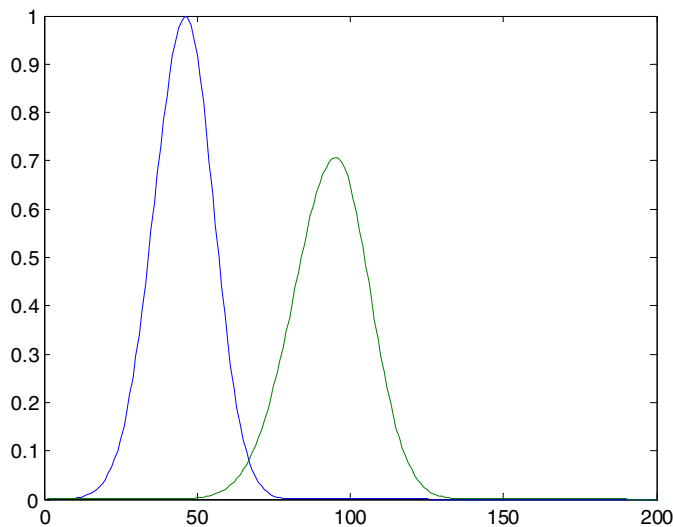
You have to click with mouse first in the starting point of the peak of every component peak and second in the ending point of peak of every component you want to consider, which defines width of the peaks.

When you have completed the selection, just press enter and a first estimation of all peaks will be displayed over imposed to the cyan-coloured plot of the matrix. In the case of the sample data, 2 components should be defined, which produces a graph like this (it can change, depending on where you arbitrarily choose the position and the height of the peaks, and the

best is to try in these parts of the experiment where the peaks are less overlapping to other signals, but not too far from the region where they appear for the first time):



In this graph, peaks are shown in black, except for the first one, which is red and denotes the experimental points with circles. A horizontal line is drawn at the half height of the peak. Then you have to select with mouse the x-axis position where you expect the peak to cross such line. In this way you are selecting the width of the symmetric peak, which is plotted now in black, while the red colour and the circles move to the next peak in order to repeat the same operation. If you are working with asymmetric peaks (i.e., you have selected this option in the beginning of the program) you have to click twice around every peak, to define the left and right boundaries and, hence, the asymmetry). At the end of the process, you should obtain an **esv** matrix similar to that provided in the file (with 200 rows and 2 columns):



Besides the **esv** matrix, to run **alpa** it is necessary to provide some information inside the **isshift** matrix. In the sample file, the elements of **isshift** are:

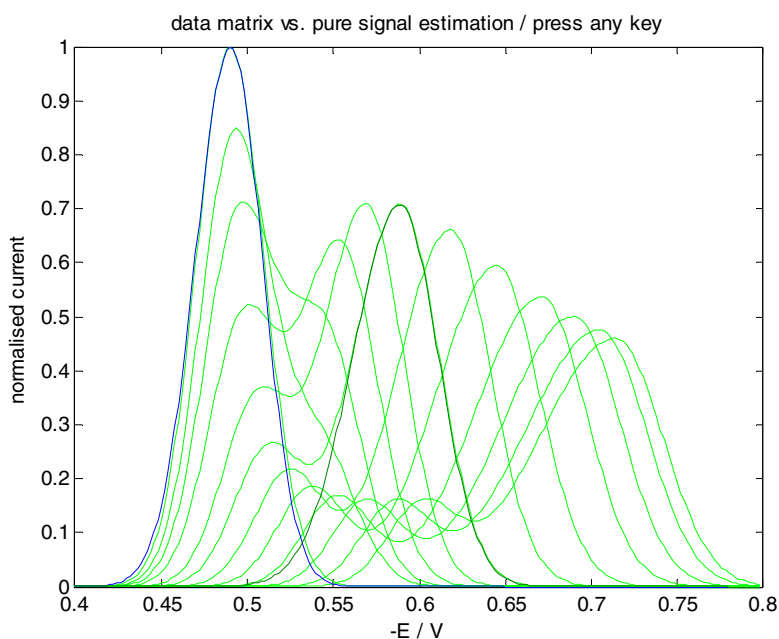
1	1
0	1
1	2
12	12

The first row indicates the movement of every peak (the two ones mean that both peaks move) and the second row indicates if there is any variation of peak width ("0" and "1" indicate that the first component has a fixed width and the second not). Inside the column corresponding to each component, the third and fourth rows define the existence range (i.e., the scan numbers when the peak appears and disappears). For instance, component 2 appears in voltammogram number 2 and exists until voltammogram 12. The numbers in the last two rows have to be deduced by a careful inspection of the data matrix, voltammogram by voltammogram.

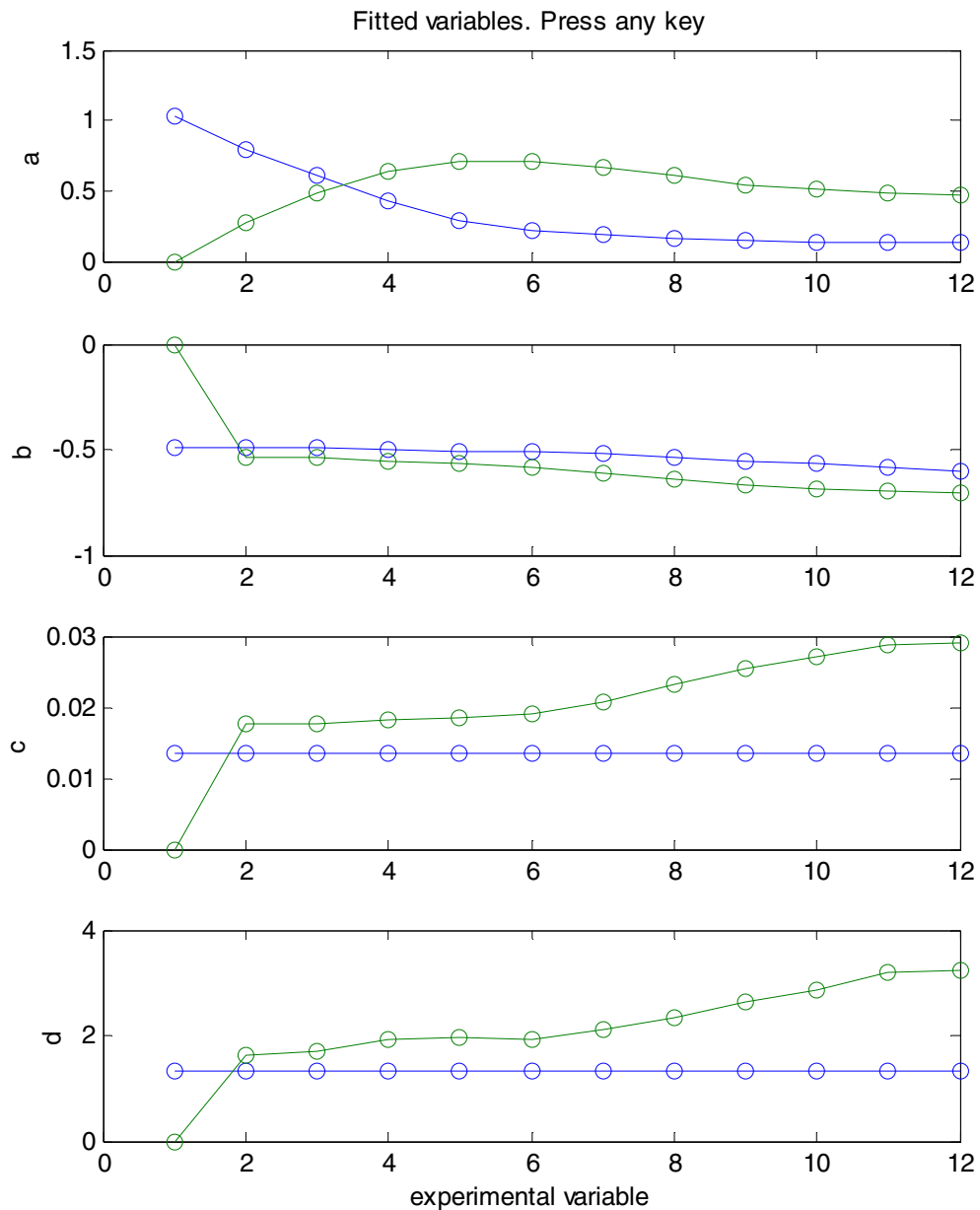
Once **esv** and **isshift** are ready, **alpha** program can be launched with the command:

```
[Epeak, Wpeak, Irep, Icor, Ierr, a, b, c, d, C, Apeak] = alpha(I, esv, E, isshift, n);
```

After executing, a figure compares the data matrix (in green) with the initial estimations of voltammograms in **esv**:



After pressing a key, the iterative process starts and, when finished, a plot is shown for the optimised a,b,c,d parameters as functions of the experimental variable (n in our case). The first component is shown in blue and the second in green:



After pressing a key, the program asks for the number of experiment to be taken as a reference for every component in order to build the corrected matrix:

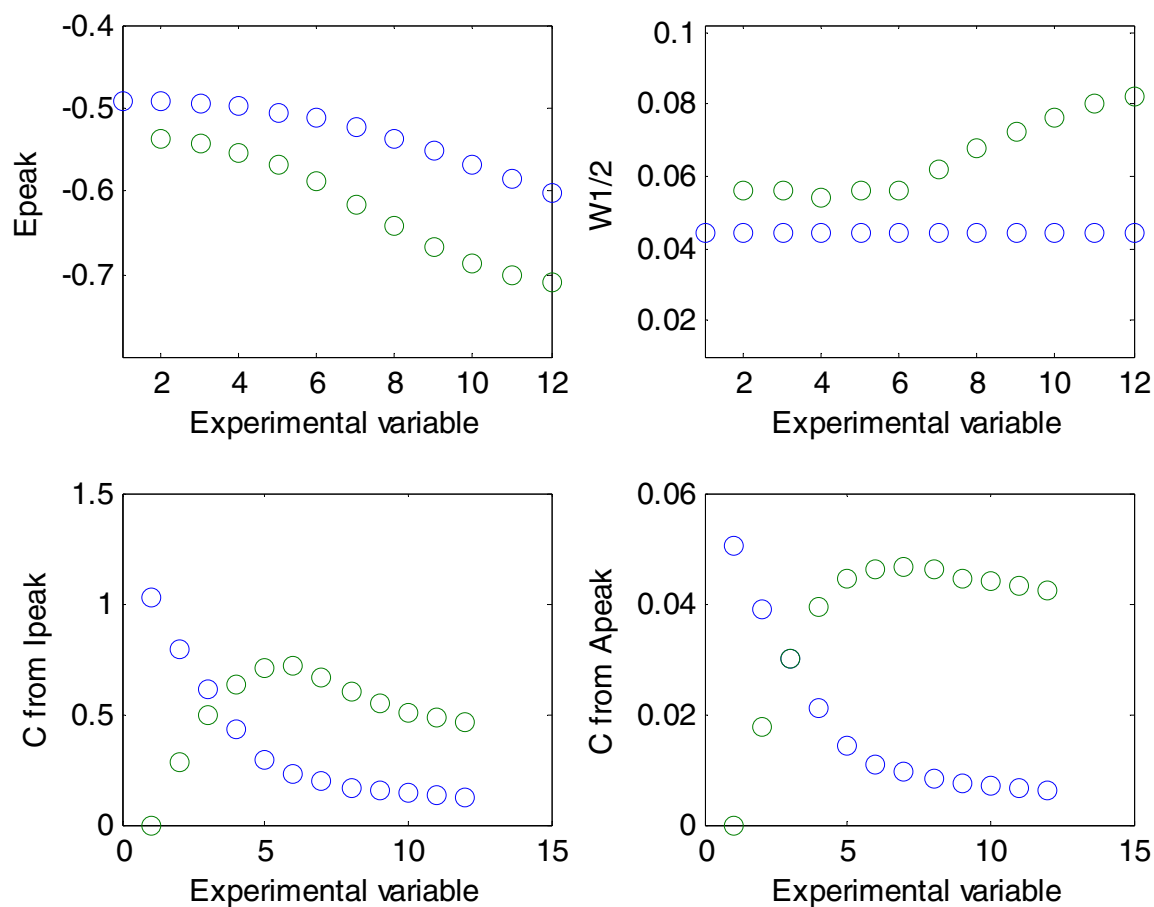
Insert the number of experiment for every reference voltammogram
 For instance [1 23 8]

If we introduce as references [1 8] and press enter,

Then, the lack of fit appears:

```
lack of fit (%) :
3.1104
```

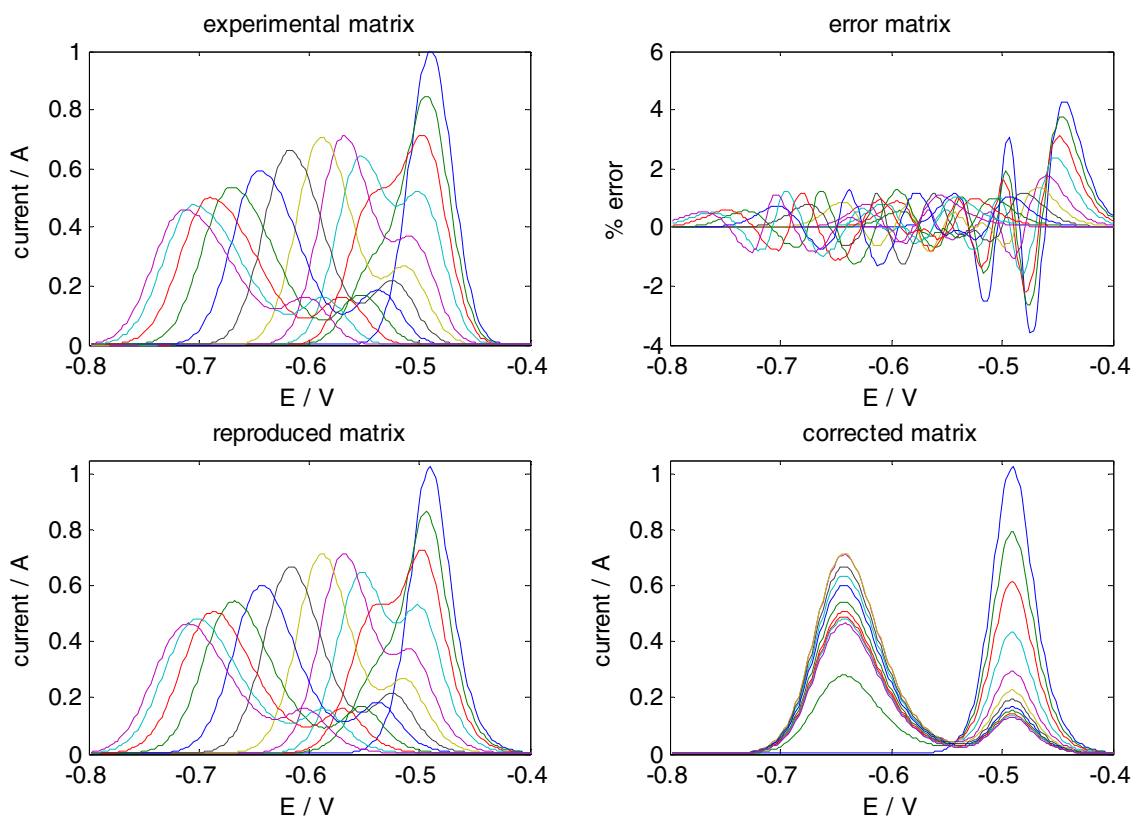
After pressing a key again, a new graph appears with the analytical parameters computed



Later, the program asks if you want to compute the concentration profiles from peak currents or peak areas. Then select the right number (e.g. 1) and press enter

Do you want to estimate concentration profiles from peak currents (1) or from peak areas (2)?

Then, you get a graph which compares the experimental (**I**) and reproduced (**I_{rep}**) matrices and also shows the error matrix and the corrected matrix (**I_{cor}**, that which would be observed if all peaks would keep the same position and width as the selected signals along all the experimental run).



The program generates the following matrices:

Epeak (12 x 2):	peak potential for every component
Wpeak (12 x 2):	half peak width for every component
Irep (12 x 200):	reproduced matrix
Icor (12 x 200):	matrix corrected for potential shifts
Ierr (12 x 200):	matrix of error
a (12 x 2):	fitted "a" values for each component at each voltammogram
b (12 x 2):	fitted "b" values for each component at each voltammogram
c (12 x 2):	fitted "c" values for each component at each voltammogram
d (12 x 2):	fitted "d" values for each component at each voltammogram
C (12 x 2):	concentration profiles for every component
Apeak (12 x 2):	peak potential E for every component

In **sample2.mat** file you have all these matrices as computed in the present tutorial. You should not get very different values unless you select initial values or restrictions (**esv**, **isshift**) very different from these employed here.