Documentation for Matlab code to draw mass loss tables

Version 1.1

J. Samuel Arey, Deedar Nabi, and Jonas Gros, EPFL, 2017

Please cite the following articles when publishing any results obtained by use of this software:

Nabi, D., Gros, J., Dimitriou-Christidis, P., Arey, J. S., "Mapping environmental partitioning properties of nonpolar complex mixtures by use of GC×GC". *Environmental Science & Technology* 2014, vol 48, p 6814-6826.

Arey, J. S., Nelson, R. K., Reddy, C. M., "Disentangling oil weathering using GC×GC 1. Chromatogram analysis", *Environmental Science & Technology* 2007, vol 41, p 5738-5746.

This code combines algorithms to estimate properties from GC×GC retention time data (Nabi et al. 2014) with codes to draw mass loss tables (MLTs), as defined by Arey et al. (2007). Please note that it is necessary to estimate properties from GC×GC retention time data in order to enable the drawing of MLTs and that these steps cannot be skipped.

Further discussion of these Matlab codes and their use is available in the book chapter of Swarthout et al.:

Swarthout, R. F., Gros, J., Arey, J. S., Nelson, R. K., Valentine, D. L., Reddy, C. M., "Comprehensive Two-Dimensional Gas Chromatography to Assess Petroleum Product Weathering", chapter in the book "Hydrocarbon and Lipid Microbiology Protocols", McGenity, T. J., Timmis, K. N., Nogales Fernández, B. (Eds.), Springer Protocols Handbooks, Springer: Berlin, **2017**.

I. Getting started. What you need to plan before GC×GC analysis.

A. Decide on the GC×GC instrument program.

Run all of the samples with the same GC×GC instrument program. Use an instrument program that leads to good chromatography for your samples, and bear in mind the instrument program requirements discussed in section 4.6 of Nabi et al., *ES&T* 2014. Chief considerations include the following:

1. Ensure that you use the following stationary phases for the 1st and 2nd dimension columns: 100% methyl polysiloxane stationary phase for column 1 (Rxi-1MS or equivalent); and methyl 50% phenyl polysiloxane stationary phase for column 2 (BPX-50 or equivalent).
2. Avoid using a 1st dimension temperature ramp that exceeds 3 °C min^{-1}.
3. Bear in mind that partitioning property predictions are considered valid only for non-polar chemicals having boiling point ≤402 °C. This includes analytes that elute earlier than

pentacosane (*n*-C_{25}) on the GC×GC chromatogram (see sections 4.5 and 4.6 of Nabi et al., *ES&T* 2014).

4. You will need to analyze a series of *n*-alkanes. The model is designed to estimate partitioning properties for the *n*-C_9 to *n*-C_{25} elution range, so ideally some or all of these *n*-alkanes should be included in the analysis. The supplied *n*-alkane members do not need to be a contiguous or regular set.
5. The 2nd dimension of the produced GC×GC chromatogram should have a "zero" retention time value when the GC×GC modulation occurs.

B. Choose a set of instrument calibration analytes.

To estimate partitioning properties with GC×GC, you will need to calibrate the model to the GC×GC instrument program. The calibration is represented by three model-fitted parameters named α_1, α_2, and α_3 (see section 4.3 of Nabi et al., *ES&T* 2014). Once you have fitted the three α parameters for a specific instrument program, the instrument program subsequently can be used to produce partitioning property predictions for as many samples as needed. However if you change any instrument program parameters that would lead to changes in analyte retention times (e.g., temperature, pressure, column length), then you must recalibrate the three α parameters. Drawing a MLT requires to use two chromatograms acquired with the same GC×GC instrument program.

To calibrate the three α parameters, you will need to know the GC×GC retention times of 15 or more identified non-polar analytes. These *instrument calibration analytes* can be compounds that are identified in the sample, or they can be separately run standards. What is important is to record the retention times of the instrument calibration analytes.

Guidelines for choosing the instrument calibration analytes are as follows. First, a minimum of 15 analytes is recommended, although more is better.

Second, the instrument calibration analytes must have known Abraham solvation parameters. Many (but not all) non-polar compounds have known Abraham parameters. Additionally, Abraham parameter datasets have undergone "revisions", and recent compilations may lead to the most reliable and consistent sets of values. Ideally the supplied Abraham parameters should come from experimental data and not computed estimates. Refer to Sections 3.1-3.2 and Tables S1 and S4 of Nabi et al., *ES&T* 2014, as well as references 22-38 in that work.
(A table of values of the Abraham parameters for several hydrocarbons is provided in the file: '*Abraham parameters for hydrocarbons.xlsx*')

Third, the instrument calibration analytes should form a balanced chemical set and span the two-dimensional region of chromatogram for which you want to make property predictions. Ideally, the instrument calibration analytes should be well-distributed throughout the chromatogram. Also, it is more important that the instrument calibration analyte set is balanced rather than large in number. The term "balanced" is used to mean that different chemical types are represented proportionately in the set. It would be better to choose a smaller calibration set that has a reasonably equivalent distribution among several different chemical types, rather than choosing a large set that is strongly biased toward only one or two chemical families.

Finally, the instrument calibration analytes can include some *n*-alkanes – in fact this is a good idea – as long as the instrument calibration set remains balanced.

II. Calibrating and applying the partitioning property estimation model.

Once the instrument calibration analytes have been analyzed using a designated GC×GC instrument program, then you are ready to calibrate and apply the partitioning property estimation model.

A. Organization of the model file directory. Where to find what.

The model code is organized as follows. The base directory contains three folders called `users/`, `model_code/`, and `model_parameters/`.

```
~/.../users/
~/.../model_code/
~/.../model_parameters/
```

These three folder names should not be changed.

The user should only need to operate from within the folder called `users/`. Normally, nothing should be changed or adjusted in the `model_code/` and `model_parameters/` folders.

Within the folder called `users/`, the organization of folders and files is user-defined. The user can define directory paths with the following two model variables:

`input_path`. This variable indicates the directory path location of the input files. Example:

```
input_path = 'users/Columbia_input/';
```

`output_path`. This variable indicates the directory path location of the output files. Example:

```
output_path = 'users/Columbia_output/';
```

These path variables are set in the file called `main.m`, and it is assumed that `main.m` is located in the directory `~/.../users/`. The `input_path` and `output_path` variables also assume that the indicated directory exists.

B. Prepare the model input files.

The model requires four input files. The input files must be placed in the directory designated by `input_path` before you can run the model. The names and contents of the input files are explained below.

The Matlab code will assume that a given unique GC×GC instrument program is represented by a capital letter ranging from A to Z. Whenever you run the model, you will designate the GC×GC instrument program based on this assigned capital letter. This letter is assigned in the main.m file and all input files corresponding to a given instrument program will end with this capital letter as shown below.

The contents of the input files may be generated by hand, or you may copy/paste data into the input files directly from an Excel spreadsheet. Example input file names are given below for a program assigned the letter 'B'.

`retention_times_alkanes_progB.dat`
This file contains three columns of data describing the *n*-alkane series retention times in the

GC×GC program. These are 1) the carbon number (Nc, an integer), 2) first dimension retention time (rt1, units of minutes), and 3) second dimension retention time (rt2, units of seconds).

The *n*-alkane series is an important model input: the model is designed to calculate partitioning properties only for solutes that fall within the 1st dimension retention time span of the *n*-alkane series. No property predictions will be made for solutes that fall outside of this elution range. The input *n*-alkane members do not need to be a contiguous, incremental set of carbon numbers: the algorithm will use retention times for whichever alkanes are provided.

`retention_times_calibration_progB.dat`
This file contains 8 columns of data describing the instrument calibration analytes used in the GC×GC program. These are 1) first dimension retention time (rt1, units of minutes), second dimension retention time (rt2, units of seconds), and 3-8) the 6 Abraham parameters of the instrument calibration analytes in the sequence: $A\ B\ S\ E\ V\ L$.

`retention_times_test_progB.dat`
This file is optional and contains two columns of data. These are 1) first dimension retention time (rt1, units of minutes) and 2) second dimension retention time (rt2, units of seconds) of "test" analytes for which you want to make partitioning property predictions, separately from the calibration analytes. This file is not necessary for drawing MLTs.

Finally, a chromatogram of a reference (non-weathered) sample and a chromatogram of a weathered sample must be provided. The naming of these two files is totally free for the user and the names will have to be provided in the parameter settings (below). By default, these chromatograms should be csv files containing one long column vector of signal intensity values, separated by a comma (,) or semi-colon (;) (as can be exported from GC Image). The length of the file is assumed a multiple of the product of the sampling rate multiplied by the modulation period, and any last additional values would be ignored. Other file types can be used (two-dimensional csv files in the GC Image format, or multi-column csv files exported from ChromaTOF, in which case the column labeled "S1" is imported). The file type is automatically recognized by the codes, assuming the file follow one of the three supported formats listed above. Baseline correction is necessary prior to drawing MLTs. We highly recommend that these chromatograms be baseline-corrected using the algorithm of Reichenbach et al. (2003), which has been implemented in the GC Image software.

C. Adjust the parameter settings of the model.

Adjust the parameter settings that appear in the first 50-70 lines of `main.m`. This file can be read and modified from within Matlab or using a generic text editor. This is the only Matlab file that you need to adjust for normal use of the code.

Most of the parameters are self-explanatory. However some additional explanation is given below.

The `program_flag` indicates the instrument program, given by a capital letter (A, B, C, D, …), to be designated by you. Example:

```
program_flag = 'A';
```

The modulation period (in units of seconds), sampling rate (in units of Hertz), and the acquisition delay (in units of seconds) should be entered.

The `Reference_chromatogram_file` and `Weathered_chromatogram_file` indicate the file names for the chromatogram of the reference (unweathered) sample and the chromatogram of the weathered sample, respectively. These files must be situated within `input_path`.

The `Bleed_cutoff` is used to indicate pixels to set to zero. The earliest part of the second dimension of a GC×GC chromatogram contains only noise and column bleed signal. To avoid having the bleed line included in the bottom cells of the MLTs, where they could generate spurious results, the corresponding pixels are set to zero. The `Bleed_cutoff` is a time in seconds. All pixels having second dimension retention times lower or equal to `Bleed_cutoff` are set to zero. Set this parameter to a value of zero if you would not like to set pixels to zero. However, removing the bleed line is a necessary step for drawing MLTs.

The `Vol_Reference` and `Vol_Weathered` are used if the user needs to perform normalization of chromatograms. Normalization is a necessary step for drawing MLTs. Two possibilities exist for normalization:
(1) `Reference_chromatogram_file` and `Weathered_chromatogram_file` correspond to chromatograms that are already normalized. Then `Vol_Reference` and `Vol_Weathered` are both set to a value of 1 (no additional normalization performed).
(2) If `Reference_chromatogram_file` and `Weathered_chromatogram_file` correspond to chromatograms that are <u>not</u> normalized. Then `Vol_Reference` is the volume of the peak used for normalization in the chromatogram of the reference (unweathered) sample. `Vol_Weathered` is the volume of the same peak in the chromatogram of the weathered sample. The peak used for normalization has to correspond to a compound that is conserved; for example the C_{30} 17α(H),21β(H)-hopane has been used by several authors for this purpose. Please ensure to use a proper peak-quantification technique for the chosen normalization peak.

The `group_flag` parameter is used to adjust the training set that is used to fit the eq 5 coefficients (see section 4.2 of Nabi et al., *ES&T* 2014). If the `group_flag` parameter is set to a value of 0, the entire training set will be used to set the eq 5 coefficients (Table S4 of Nabi et al., *ES&T* 2014). If you are making predictions for hydrocarbons only, we have found that a hydrocarbons-only training set gives better regression statistics compared to a fit of the entire nonpolar compound set. With the `group_flag` parameter set to a value of 1, the model will use the hydrocarbons-only training set for the fitting of eq 5 coefficients. Training set regression statistics for all 11 partitioning properties are given as output in the file called `eq5_training_set_fit_statistics.dat`, discussed further below. A priori, `group_flag` should be set to 1 for oil spill chromatograms, and this is the default value of this parameter in `main.m`.

The `winP` and `winC` are used to indicate the width and height of the cells of the MLT, in units of log(vapor pressure) and log(solubility), respectively. The user is a priori advised to stick to default values.

The `bdP` and `bdC` are used to indicate respectively the limits of $\log_{10}$(vapor pressure) and $\log_{10}$(aqueous solubility) spanned by the MLT. Each of these two parameters is a vector that specifies the lower and upper bound spanned by the MLT. The user is a priori advised to stick to default values.

The `NoiseCutoff` is used to decide the limits for cells to be counted as containing mostly noise and be shaded black. Typical values are 1e-5 to 2e-4. Cells that contain a fraction of the total mass in

the MLT lower than `NoiseCutoff` are shaded black. Increase the value of `NoiseCutoff` to remove more cells, or decrease its value to keep more cells in the MLT.

D. Run the model.

The Matlab code is straightforward to use.

Make sure that the Matlab working directory points to `~/.../users`. Then at the Matlab prompt, type:

```
>> main
```

The model may require several seconds to run.

III. Interpreting the model output.

A. Output appearing in the Matlab console.

The output appearing in the Matlab console depends on the parameter setting you have chosen for the `prompt_output` parameter in the file main.m. The default parameter setting is `'normal'`, which will lead the model to produce the following output information in the Matlab console:

```
Fitted alpha_1 and alpha_2 values (eq 6) are:

    0.2360   -0.1614

Bootstrap uncertainty estimates of alpha_1 and alpha_2 are:
    0.0070    0.1033

The resulting r^2 and RMSE values of eq 6 fitted u_1 values are:
    0.9900    0.1176
```

The above results indicate that the model has determined values of $\alpha_1 = 0.236 \pm 0.007$ and $\alpha_2 = -0.16 \pm 0.10$ using the instrument calibration analyte set, based on a regression fit of eq 6 (see section 4.3 of Nabi et al., *ES&T* 2014).

Eq 6 produces GC×GC-estimated values of the parameter log L_1, which is equivalent to u_1, for each analyte. According to the output shown above, the regression fit has produced a correlation coefficient of $r^2 = 0.990$ and root-mean-squared-error of RMSE = 0.12, between the GC×GC-estimated u_1 values and the reference u_1 values, for the set of calibration analytes. The reference values are obtained using the Abraham solvation model for the stationary phase of the GC×GC 1st dimension column.

The user should inspect the above output results carefully to ensure that the regression fits of α_1 and α_2 are robust. In particular:

1. Good fit statistics for eq 6, ideally $r^2 \geq 0.98$ and RMSE ≤ 0.15.
2. Tolerable uncertainties for the α values, ideally an α_1 uncertainty ≤ 0.01 and an α_2 uncertainty ≤ 0.2.

Subsequently the model will output the following results.

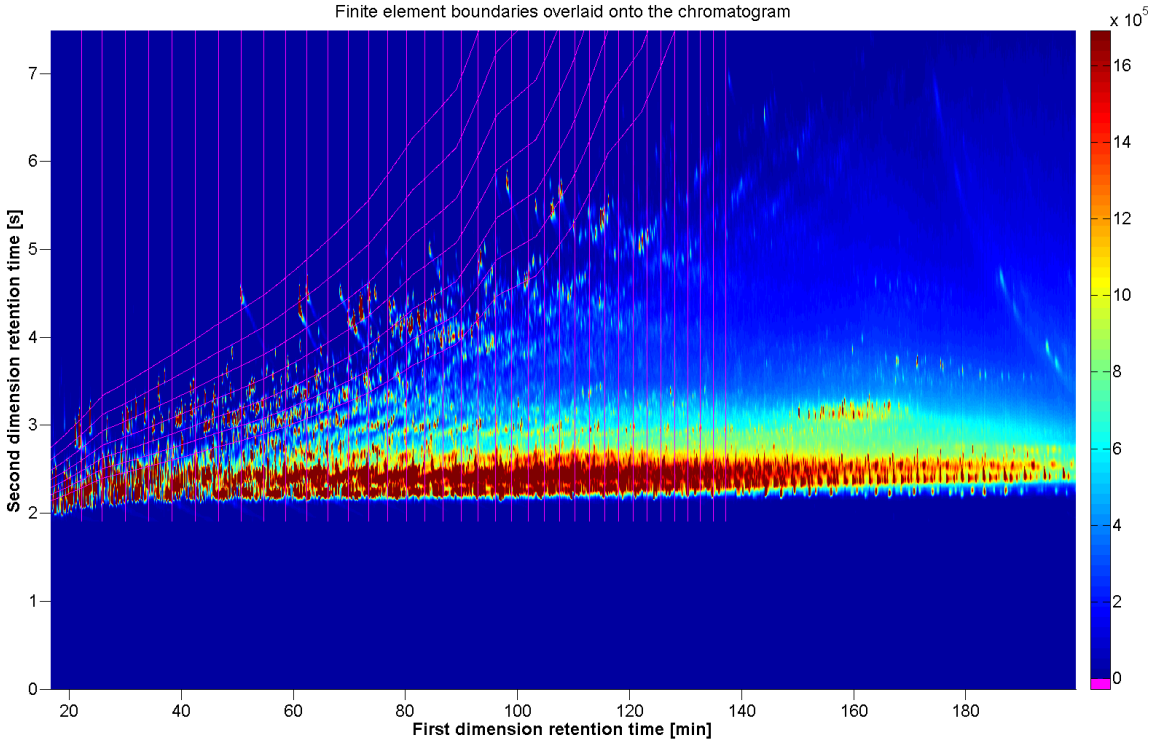
```
Now fitting alpha_3 with a nonlinear optimization of eq 7.

Local minimum found.

Optimization completed because the size of the gradient is less than
the default value of the function tolerance.

<stopping criteria details>

Conducting a bootstrap uncertainty analysis of alpha_3. This may take a minute.
The fitted alpha_3 value is:
    0.8026

The bootstrap uncertainty estimate of alpha_3 is:
    0.1001

The r^2 and RMSE values of eq 7 fitted u_2 values are:
    0.9004    0.0843
```

This means that a value of $\alpha_3 = 0.80 \pm 0.10$ has been assigned, based on a non-linear fit of eq 7. Fit statistics for eq 7 are shown, finding a $r^2 = 0.90$ and RMSE = 0.084 for u_2 values of the calibration set. As with the first two α values, the user should inspect the statistics of the eq 7 fit in order to ensure that α_3 is well-determined. You will want to see:

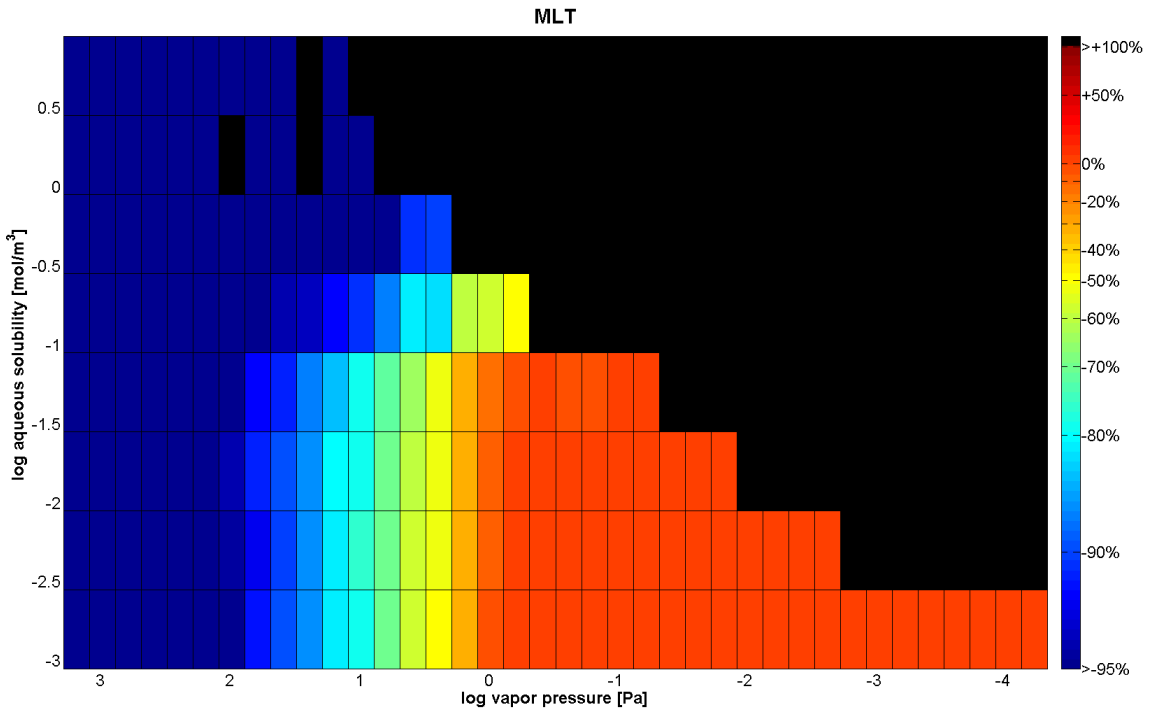
1. Good fit statistics for eq 7, ideally $r^2 \geq 0.85$ and RMSE ≤ 0.1.
2. An α_3 uncertainty ≤ 0.2.

Finally, α_3 has the physical interpretation of (approximately) representing the second dimension hold-up time. This equivalence is not exact, due to the presence of inactive column sections in the instrument. Nonetheless, the α_3 value should have a physically reasonable value. For example, α_3 should not have a negative (<0) or imaginary value. Typically expected values of α_3 would be between 0 and 1 s. If the user observes an unreasonable value for α_3, this is a sign that something else is likely wrong.

The code then produces a series of figures. The first two figures show the correlation of the fitted Abraham L_1 and L_{12} predicted by equations 6 and 7, respectively, to the known values for the calibration analytes. The next two figures show how the GC×GC chromatogram was divided into vapor pressure and solubility cells. The first of these figures is entitled "Checkerplot heatmap of finite element weighting" and shows how heavily each pixel was weighted in calculating the mass present in each cell. The second figure is entitled "Finite element boundaries overlaid onto the chromatogram", which shows the contours of the MLT cells (pink) overlaid onto the chromatogram of the reference (unweathered) sample:



The last figure is entitled "MLT" and contains the MLT. The color bar is labeled so that negative values indicate mass losses, whereas positive values indicate mass gains:



B. Names and contents of the output files.

The code will create several output files in the directory `output_path`. The naming convention for output files is:

"output_file_progB.dat"

for output data that you have generated from instrument program 'B'. The model will overwrite existing files, if they have the same names as the target output names of the model.

These output files are not necessary for drawing MLTs.

The output files are:

`predicted_properties_test_progB.dat`
This file contains the GC×GC-predicted partitioning properties for the analyte test set. Each row corresponds to a chemical for which retention time data was provided in the input file `retention_times_test_progB.dat`. Each column corresponds to a partitioning property, according to the sequence shown in section II.C above. This file can be imported directly into Excel or copy/pasted into Excel.

`ASM_predicted_properties_calib_progB.dat`
This file contains the partitioning property predictions given by the Abraham solvation model for the instrument calibration analytes. Each row corresponds to a chemical for which information was provided in the input file `retention_times_calibration_progB.dat`. Each column corresponds to a partitioning property, according to the sequence shown in section II.C above. This file can be imported directly into Excel or copy/pasted into Excel.

`eq5_training_set_fit_statistics.dat`

This file contains the eq 5 fitted λ coefficient values, the uncertainties assigned to each λ coefficient, and eq 5 regression fit statistics for the training set, for each partitioning property. Each row corresponds to a partitioning property, according to the sequence shown in section II.C above. The columns follow the format:

λ_1 λ_2 λ_3 σ_{λ_1} σ_{λ_2} σ_{λ_3} RMSE r^2

where $\lambda_j \pm \sigma_{\lambda_j}$ refers to the 95% confidence interval of λ_j.

Contacts

For any questions, comments, or bug reports, please contact:
J. Samuel Arey: arey@alum.mit.edu
Deedar Nabi: deedarnabi@gmail.com
Jonas Gros: gros.jonas@gmail.com

Acknowledgements

Special thanks to Bob Swarthout (Appalachian State University) for agreeing to beta-test the code during the development of this documentation.