

# **The GC×GC Elution Model: A Matlab code to simulate GC×GC retention times for a hydrocarbon structure library**

## **User Documentation**

Version 1.0

J. Samuel Arey (Oleolytics LLC) and the Concawe Environmental Management Group

September 2022

Please cite the following article when reporting any results obtained by use of this software:

J. Samuel Arey, Alberto Martin Aparicio, Eleni Vaiopoulou, Stuart Forbes, and Delina Lyon, “Modeling the GC×GC elution patterns of a hydrocarbon structure library to innovate risk assessments of petroleum substances”, *in preparation* (2022).

### **Contacts**

For questions, comments, or bug reports, please contact:

J. Samuel Arey: [sam@oleolytics.com](mailto:sam@oleolytics.com), [arey@alum.mit.edu](mailto:arey@alum.mit.edu)

Concawe EMG: [environment@concawe.eu](mailto:environment@concawe.eu)

### **I. Purpose of the GC×GC elution model software**

The GC×GC elution model is designed to simulate the GC×GC retention times of a user-provided hydrocarbon library. The hydrocarbon library is defined as a list of chemical structures which represent a hypothetical substance composition, described with the Simplified Molecular-Input Line-Entry System (SMILES) format. The elution model produces as output a set of predicted two-dimensional GC×GC retention times for these chemical structures. The model output depends on the GC×GC instrument conditions. The elution model can flexibly represent a range of possible GC×GC instrument methods through three tunable model parameters:  $\alpha_1$ ,  $\alpha_2$ , and  $\alpha_3$ . To adapt the model to a particular instrument method, the user must provide measured two-dimensional retention time data for a training set of known calibration analytes. The code then optimizes the three tunable parameters by regression of the model equations: this procedure is called model calibration. The user can then employ the calibrated model to predict GC×GC retention times of the hydrocarbon library, which assumes the same instrument method conditions as those used to measure the calibration analytes. The GC×GC elution model can simulate gas chromatography-amenable chemical structures containing a wide variety of functionalities: it has been tested for hydrocarbon analytes.<sup>1</sup> The elution model code is programmed in Matlab,<sup>2</sup> and it is freely distributed under an open-source license.<sup>3</sup> The User Documentation explains how to use the code. The User Documentation assumes that the reader has familiarity with basic operations in Matlab, as well as the creation and editing of ASCII-format text files. Further information on the theory, validation, and limitations of the GC×GC elution model are provided in the article by Arey and coworkers.<sup>1</sup>

## II. Getting started: what the user must plan before the GC×GC analysis

### A. Decide on the GC×GC instrument program

The designation of the GC×GC instrument program represents the first consideration for users of the GC×GC elution model. Each time the GC×GC elution model is called, it will optimize the tunable model parameters ( $\alpha_1$ ,  $\alpha_2$ ,  $\alpha_3$ ) to the measured GC×GC retention time data of a set of calibration analytes. The resulting calibration represents the elution model's representation of a particular instrument program and instrument column set. Therefore, the measured retention time data for the calibration analytes must originate from an analysis with a particular GC×GC instrument program (i.e., the oven temperature program, inlet gas pressure program, and inlet temperature program) and a particular instrument column set (i.e., the column types and column lengths from the injector inlet to the detector). If any aspect of the instrument program or column set are changed, the elution model must be recalibrated.

When choosing the GC×GC instrument program and column set, the following requirements and recommendations should be considered:

1. The elution model version 1.0 is hard-parameterized for the following stationary phases for the first- and second-dimension columns: 100% methyl polysiloxane stationary phase for column 1 (e.g., Rxi-1MS or equivalent); and methyl 50% phenyl polysiloxane stationary phase for column 2 (e.g., BPX-50 or equivalent). If the analyst chooses stationary phase types that are different than those indicated here, they will be required to update hard-coded parameters in the model. This update procedure is not very complicated, but it requires familiarity with mathematical programming in Matlab and it lies outside of the scope of this documentation.
2. Decide the GC×GC oven program with the following recommendations in mind. The elution model assumes that the first-dimension and second-dimension ovens both apply linear temperature programs (i.e., constant temperature ramps) throughout the analysis; therefore, we expect the elution model to perform best when such conditions are applied. We expect the elution model to perform best for a first-dimension temperature ramp that is not overly steep; we recommend the application of a first-dimension temperature ramp which does not exceed 3 °C min<sup>-1</sup>. The second-dimension oven program can differ from the first-dimension oven program in both the offset temperature and the ramp rate.
3. The elution model assumes that the second-dimension column experiences a constant gas hold-up time throughout the entire analysis. The inlet pressure program should be designed to meet this requirement, if possible.
4. The elution model version 1.0 has been validated for an elution window which spans *n*-decane (*n*-C<sub>10</sub>) to *n*-tetracosane (*n*-C<sub>24</sub>). This elution window is expected to envelope analytes having boiling points approximately between 174 and 391 °C. The GC×GC instrument program should be designed with these limitations in mind.

Further information about the assumptions, limitations, and domain of the GC×GC elution model are described in the article and Supporting Information of Arey and coworkers.<sup>1</sup>

## B. Choose a set of calibration analytes

To optimize the model for the particular GC×GC instrument program, the user must provide measured retention time data for a set of calibration analytes. The calibration analytes should be selected in a way that the model calibration procedure robustly constrains the three tunable model parameters,  $\alpha_1$ ,  $\alpha_2$ , and  $\alpha_3$ . This section describes the requirements and recommendations for choosing and measuring the calibration analytes.

1. For each calibration analyte, the user must be able to confidently assign a known chemical structure which can be represented as a SMILES string, and the user must be able to obtain measured two-dimensional retention times with the designated GC×GC instrument program and column set.
2. The calibration analytes must include a series of *n*-alkanes. The model is designed to simulate analytes falling the *n*-C<sub>10</sub> to *n*-C<sub>30</sub> elution range. The supplied *n*-alkane members do not need to span this entire range, nor must the *n*-alkanes form a contiguous or regular set. However, we consider the model predictions to be most valid within the elution window that is spanned by the measured *n*-alkanes.
3. The calibration analytes should span the modeled two-dimensional elution region as widely as possible, and the calibration analytes should also exhibit balance among the represent classes. Ideally, the calibration analytes should be well-distributed throughout the covered elution space. Finally, the set of calibration analytes should include a minimum of 15-20 members. For petroleum hydrocarbons, these recommendations can be achieved by injecting the following standards:
  - a. A series of *n*-alkanes.
  - b. A series of parent polycyclic aromatic hydrocarbons (PAHs).
  - c. A series of mono-aromatic analytes, such as the linear alkyl benzenes, and/or a series of mono-naphthenic analytes, such as the linear alkyl-cyclohexanes.
4. In selecting the members of the calibration analyte, achieving balanced chemical representation is more important than maximizing the size of the set. The term “balance” is used to mean that different chemical types are represented proportionately in the set. For example, it would be preferable to select a calibration set that has proportionate representation among different chemical types, rather than assigning a larger calibration set that is strongly weighted by a single chemical family or a localized sub-region of the chromatogram.
5. The retention time data for the calibration analytes can be obtained from injected standards, if these are available. Alternatively, the calibration analyte data can be obtained from a petroleum substance sample, if the analyst can confidently identify relevant analytes in the sample.
6. The user must record the measured two-dimensional retention time data of the calibration analytes. The two-dimensional retention time of each calibration analytes is represented by two floating point numbers: (1) a retention time in the first dimension, having units of minutes; and (2) a retention time in the second dimension, having units of seconds. These retention time data can be obtained from commercially available software or by a user-defined method. The required file formats to input the calibration analyte data into the elution model are described in the subsequent section.

### III. Calibrate and apply the GC×GC elution model

Once the calibration analytes have been analyzed using a designated GC×GC instrument program, then the user is ready to calibrate and apply the GC×GC elution model.

#### A. Organization of the model file directory

The elution model is organized as follows. In the base directory, three folders exist, called `users/`, `model_code/`, and `model_parameters/`. These three folder names should not be changed.

```
~/GC×GC_elution_model/users/  
~/GC×GC_elution_model/model_code/  
~/GC×GC_elution_model/model_parameters/
```

To run the model, the user must provide user-defined parameter data and calibration analyte information within the `user/` directory. The user may choose to employ the default “mini” hydrocarbon library data provided in the `model_parameters/` folder, or the user can obtain the full hydrocarbon library from Concawe. Alternatively, the user can define a hydrocarbon library. During routine operation of the model, the user should not need to make any changes in the `model_code/` folder, including `main.m`, except in special circumstances that are explained below.

By default, sample input data and sample model parameters are already provided in the appropriate folders. Therefore, the user can run the model on a sample data set and view the outputs without needing to modify any files.

#### B. Set user-defined model parameters in the `user/` directory

The `users/` folder contains a text file named `GC×GC_elution_model.m`, as well as two sub-folders, `input/` and `output/`. All of the user-selected model parameters are contained in `GC×GC_elution_model.m`. The user must modify `GC×GC_elution_model.m` to specify the user-selected inputs that will be employed by the model, as follows. This file can be opened and edited within Matlab, or else with a common text editor program such as Notepad++.

Within the file `GC×GC_elution_model.m`, the user must define following parameters which describe model inputs and model calibration:

`input_path`

The `input_path` variable indicates the directory path location of the user-specified input files. The default value is:

```
input_path = 'user/input/';
```

The `input_path` variable assumes that the indicated directory exists.

`alkanes_data_file`

The `alkanes_data_file` variable indicates the name of the file that will contain quantitative data about the measured *n*-alkanes that are employed as calibration analytes. The file should reside in the folder designated by `input_path`. The default value is:

```
alkanes_data_file = 'alkane_data_progA.dat';
```

This sample file name can be interpreted to refer to the alkanes data for instrument program “A”. The user should update this information to a file name that contains appropriate data for the *n*-alkanes of the

calibration set. If left to the default value, the model will employ a sample data set which is contained in the input folder.

#### calibration\_data\_file

The calibration\_data\_file variable indicates the name of the file that will contain quantitative data about the entire set of calibration analytes, including *n*-alkanes. The file should reside in the folder designated by input\_path. The default value is:

```
calibration_data_file = 'calibration_data_progA.dat';
```

This sample file name can be interpreted to refer to the calibration analyte data for instrument program “A”. The user should update this information to a file name that contains appropriate data for the calibration set. If left to the default value, the model will employ a sample data set which is contained in the input folder.

#### calibration\_classes\_file

The calibration\_classes\_file variable indicates the name of the file that will contain text data which describe the chemical class information of the calibration analytes. The file should reside in the folder designated by input\_path. The default value is:

```
calibration_classes_file = 'calibration_classes_progA.dat';
```

This sample file name can be interpreted to refer to the class information of calibration analytes for instrument program “A”. The user should update this information to a file name that contains appropriate data for the calibration set. If left to the default value, the model will employ a sample data set which is contained in the input folder.

#### modulation\_period

The modulation\_period variable indicates the modulation period of the GC×GC instrument program, in units of seconds. The default value is:

```
modulation_period = 12.5;
```

The user should update this information to the modulation period that is used by the GC×GC instrument program for the calibration analytes. The default value (12.5 s) is appropriate for the sample calibration analyte files that are contained in the input folder (including the sample files with either appendix “progA” or appendix “progB”).

#### beta\_input

The beta\_input variable indicates the value of  $\beta$  applied in the relationship,  $\Delta \log L_{21} = \log L_1 - \beta \log L_2$ , described by Arey and coworkers.<sup>1</sup> The default value is 1.07, which was determined in that work by Gram-Schmidt orthogonalization of a large petroleum hydrocarbon library containing 15,495 hydrocarbon structures:

```
beta_input = 1.07;
```

During normal application of the model, we do not expect the user to change this default value. However, if the user wishes to supply a tailored hydrocarbon library in the model\_parameters folder, they have the option to update this parameter by indicating:

```
beta_input = 0;
```

If a value of zero is input, the elution model code will update the  $\beta$  parameter self-consistently by Gram-Schmidt orthogonalization of the  $\log L_1$  and  $\log L_2$  values of the hydrocarbon library. The new beta parameter will be displayed in the program output.

#### library\_ESABVL\_file

The library\_ESABVL\_file variable indicates the name of the file that contains text data which describe the Abraham parameter information of the hydrocarbon library structures. The file should reside in the folder designated by model\_parameters/. The default value is:

```
library_ESABVL_file = 'mini_constituent_library_ESABVL.dat';
```

The file mini\_constituent\_library\_ESABVL.dat contains Abraham parameters for an abridged hydrocarbon library containing 971 hydrocarbon structures, which is a shortened version of the full hydrocarbon library of 15,495 structures described by Arey and coworkers.<sup>1</sup> To obtain the full hydrocarbon library, the user should contact Concawe at the correspondence address provided in this document (see page 1). The file format and other information about the library\_ESABVL\_file is provided in section V, below.

#### library\_class\_CN\_file

The library\_class\_CN\_file variable indicates the name of the file that contains text data which describe the chemical class and carbon number information of the hydrocarbon library structures. The file should reside in the folder designated by model\_parameters/. The default value is:

```
library_class_CN_file = 'mini_constituent_library_carbon_number_class.dat';
```

The file mini\_constituent\_library\_carbon\_number\_class.dat contains chemical class and carbon number data for an abridged hydrocarbon library containing 971 hydrocarbon structures, which is a shortened version of the full hydrocarbon library of 15,495 structures described by Arey and coworkers.<sup>1</sup> To obtain the full hydrocarbon library, the user should contact Concawe at the correspondence address provided in this document (see page 1). The file format and other information about the library\_class\_CN\_file is provided in section V, below.

Within the file GCxGC\_elution\_model.m, the user must also define the following parameters which describe model outputs:

#### output\_path

This variable indicates the directory path location of model-generated output files. The default value is:

```
output_path = 'user/output/';
```

The output\_path variable assumes that the indicated directory exists.

#### generic\_plot\_flag

By setting this variable to a value of 1, the user queries a set of plots to visualize basic output information from the model. A value of 0 will suppress the plots. The default value is:

```
generic_plot_flag = 1;
```

#### single\_CN\_plot\_flag

By setting this variable to a value of 1, the user queries a plot of simulated retention times for only the members of the hydrocarbon library having one carbon number (the value of which is specified by the next parameter). A value of 0 will suppress the plots. The default value is:

```
single_CN_plot_flag = 1;
```

#### single\_CN\_value

If the single\_CN\_plot\_flag variable has been set to a value of 1, the user must indicate a carbon number value. For the supplied hydrocarbon library, valid values can range from 10 to 30. The provided value is:

```
single_CN_value = 20;
```

## B. Prepare the model input files

The model requires the following input files. The input files must be placed in the directory designated by `input_path` before the user can run the model. All input files are provided in ASCII text format. The names and contents of the input files are explained below.

### `alkanes_data_file`

This file contains quantitative data about the measured *n*-alkanes that are employed as calibration analytes. The file should reside in the folder designated by `input_path`. This input file contains three columns of information, which are tab-separated:

- *n*-alkane carbon number provided as an integer (first column)
- first-dimension retention time in units of minutes (second column)
- second-dimension retention time in units of seconds (third column).

The data for each analyte must be provided as a successive row. Two example files are provided in the input folder: `alkane_data_progA.dat` and `alkane_data_progB.dat`.

### `calibration_data_file`

This variable indicates the name of the file that will contain quantitative data about the entire set of calibration analytes, including *n*-alkanes. There are two general types of information included in this file. The first two columns contain measured retention time data of the calibration analytes. The last six columns contain the Abraham solute parameters provided by the UFZ-LSER prediction tool.<sup>4</sup> The file should reside in the folder designated by `input_path`. This input file contains eight columns of information, which are tab-separated:

- first-dimension retention time in units of minutes (first column)
- second-dimension retention time in units of seconds (second column)
- the E parameter provided by the UFZ-LSER prediction tool (third column)
- the S parameter provided by the UFZ-LSER prediction tool (fourth column)
- the A parameter provided by the UFZ-LSER prediction tool (fifth column)
- the B parameter provided by the UFZ-LSER prediction tool (sixth column)
- the V parameter provided by the UFZ-LSER prediction tool (seventh column)
- the L<sub>16</sub> parameter provided by the UFZ-LSER prediction tool (eighth column)

The data for each analyte must be provided as a successive row. Two example files are provided in the input folder: `calibration_data_progA.dat` and `calibration_data_progB.dat`. Note that information about the *n*-alkanes is provided in both of the input files, `alkanes_data_file` and `calibration_data_file`.

### `calibration_classes_file`

This variable indicates the name of the file that will contain text data which describe the chemical class information of the calibration analytes. The file should reside in the folder designated by `input_path`. For each calibration analyte listed in the `calibration_data_file`, the `calibration_classes_file` must contain a string which indicates the class type. Take note that the ordering of the class type data in the `calibration_classes_file` must match the ordering of the analyte data listed in the `calibration_data_file`. The class types indicated in `calibration_classes_file` should match the class types provided in the hydrocarbon library file in the `model_parameters/` folder. For the default (provided) hydrocarbon library, these class types are:

nP	normal paraffins
iP	iso-paraffins
mN	mono-cyclic naphthenes

dN	di-cyclic naphthenes
polyN	polycyclic naphthenes having $\geq 3$ rings
mAr	mono-cyclic aromatics
dAr	di-cyclic aromatics
polyAr	polycyclic aromatics having $\geq 3$ rings
NmAr	naphthenic mono-cyclic aromatics
NdAr	naphthenic di-cyclic aromatics
NpolyAr	naphthenic polycyclic aromatics

Two example files are provided in the input folder: calibration\_classes\_progA.dat and calibration\_classes\_progB.dat.

### C. Run the model

The file GC×GC\_elution\_model.m represents the file that will be called by the user to run the elution model within the Matlab program. To run the model, first ensure that Matlab is pointed to the user/ directory (using, for example, the cd() function). Then, the user can enter GC×GC\_elution\_model at the Matlab prompt.

```
>> GC×GC_elution_model
```

### IV. Access and interpret the model output

When GC×GC\_elution\_model is executed, several output results will be displayed in the Matlab interface, and several plotted results will be displayed in graphical windows, by default. The user can verify results as follows.

The elution model should adequately constrain the three  $\alpha$  parameters. The elution model code provides several output statistics in the Matlab interface, and it also displays three plots showing graphical results of the model calibration. The adequacy of the model calibration should be judged on a case-by-case basis, depending on the instrument program, the modeled elution window, and the nature of the set of calibration analytes. However, we can provide the following general recommendations:

- The model-fitted  $\log L_1$  values should exhibit a good correlation with the LSER-determined values for the calibration analytes. For example, the user can verify that the reported squared correlation coefficient is  $r^2 \geq 0.95$ .
- The model-fitted  $\Delta \log L_{21}$  values should exhibit a good correlation with the LSER-determined values for the calibration analytes. For example, the user can verify that the reported squared correlation coefficient is  $r^2 \geq 0.90$ .
- The fitted  $\alpha_1$  value should substantially exceed the bootstrap-estimated uncertainty of  $\alpha_1$ .
- The bootstrap-estimated uncertainty of the fitted  $\alpha_2$  value should not substantially exceed 0.1 (log units).
- The bootstrap-estimated uncertainty of the fitted  $\alpha_3$  value should not substantially exceed 0.2 (units of seconds).
- The user should verify that the reported statistics are satisfactory for the fitted retention times for the calibration analytes. By default, the elution model code displays the root-mean-squared-error (RMSE) and squared correlation coefficient ( $r^2$ ) values of fitted versus measured values of both the first- and second-dimension retention times of the calibration analytes.



To further facilitate the interpretation of the model calibration results, three figures are plotted by the elution model code. The first figure displays the model-fitted  $\log L_1$  values against the corresponding LSER-determined values for the calibration analytes. The second figure displays the model-fitted  $\Delta \log L_{21}$  values against the corresponding LSER-determined values for the calibration analytes. The third figure displays the model-fitted two-dimensional retention times (circles) against with measured retention times (triangles) of the calibration analytes, together with a connecting line segment that shows the deviation of the fitted value from the measured value for each calibration analyte.

Once the user has verified the adequacy of the model calibration, attention can be turned to the simulation output for the hydrocarbon library. The elution model code displays several plotted results which are explained in further detail by Arey and coworkers.<sup>1</sup>

The elution model code also produces several output files, which are contained in the folder specified by the user-defined path, `output_path` (the default value is `user/output/`). These files are explained as follows.

`logL1_DeltalogL21_calibration.dat`

This file contains the LSER-determined  $\log L_1$  values of the calibration analytes (first column), and the LSER-determined  $\Delta \log L_{21}$  values of the calibration analytes (second column), in scientific format.

`logL1_DeltalogL21_fitted.dat`

This file contains the model-fitted  $\log L_1$  values of the calibration analytes (first column), and the model-fitted  $\Delta \log L_{21}$  values of the calibration analytes (second column), in scientific format.

`logL1_DeltalogL21_fitted_test.dat`

This file should contain contents that are identical to the file, `logL1_DeltalogL21_fitted.dat`. The file appended with “test” represents an internal check on the determination of the fitted values.

`t1_t2_fitted.dat`

This file contains the model-fitted first-dimension retention time values (units of minutes) of the calibration analytes (first column), and the model-fitted second-dimension retention time values (units of seconds) of the calibration analytes (second column), in scientific format.

`logL1_DeltalogL21_library.dat`

This file contains the LSER-determined  $\log L_1$  values of the library constituents (first column), and the LSER-determined  $\Delta \log L_{21}$  values of the library constituents (second column), in scientific format.

`logL1_logL2_library.dat`

This file contains the LSER-determined  $\log L_1$  values of the library constituents (first column), and the LSER-determined  $\log L_2$  values of the library constituents (second column), in scientific format.

`t1_t2_library.dat`

This file contains the simulated first-dimension retention time values (units of minutes) of the hydrocarbon library (first column), and the simulated second-dimension retention time values (units of seconds) of the hydrocarbon library (second column), in scientific format.

## V. Description of the files contained in the `model_parameters/` folder

The publicly available model is provided together with an abridged (“mini”) hydrocarbon library containing 971 hydrocarbon structures: it represents a shortened version of the hydrocarbon library containing 15,495

hydrocarbon structures described by Arey and coworkers.<sup>1</sup> To obtain the full hydrocarbon library, the user should contact Concawe at the correspondence address provided in this document (see page 1). If the user intends to apply the elution model to simulate the abridged hydrocarbon library, then they should make no changes to any files contained in the `model_parameters/` folder.

If the user wishes to apply the elution model to a user-defined hydrocarbon library, then the following section describes the input files necessary to fully construct the user-defined hydrocarbon library. By default, the library is fully defined by the data in two files: `mini_constituent_library_carbon_number_class.dat` and `mini_constituent_library_ESABVL.dat`, both of which are contained in the `model_parameters/` folder.

`mini_constituent_library_carbon_number_class.dat`

This file contains information about the carbon number and chemical class of each constituent in the library. This input file contains two columns of information, which are tab-separated:

- *n*-alkane carbon number provided as an integer (first column)
- a string which defines the chemical class assignment of the library constituent (second column)

The data for each library constituent must be provided as a successive row. The chemical class type assignments can be user-defined, as long as each class type can be represented by a single string (i.e., a sequence of consecutive alphanumeric characters that are unbroken by white space and do not contain special characters). The class types in the library file, `mini_constituent_library_carbon_number_class.dat`, must be consistent with the class types that are indicated in the input file, `calibration_classes_file`, of the calibration analytes.

If the user wishes to provide user-defined chemical class types, they must also update the function `create_class_index.m`, which is contained in the `model_code/` folder. This Matlab function assigns an index to the class types. The index starts at “1” and goes up to “11”, because there are eleven class types in the default hydrocarbon library. The user must update the string values of each class type in the `create_class_index.m` file, so that they are consistent with the `calibration_classes_file` in the `model_parameters/` folder. Take note that the index value of 11 is reserved for nP class (*n*-alkanes), due to the way that the plotting functions are currently coded. Therefore, the user should avoid changing the index assignment for nP class, if possible, because this would necessitate additional changes at other places in the model code.

`mini_constituent_library_ESABVL.dat`

This file contains information about the Abraham solute parameters provided by the UFZ-LSER prediction tool, for each constituent in the library. This input file contains six columns of information, which are tab-separated:

- the E parameter provided by the UFZ-LSER prediction tool (first column)
- the S parameter provided by the UFZ-LSER prediction tool (second column)
- the A parameter provided by the UFZ-LSER prediction tool (third column)
- the B parameter provided by the UFZ-LSER prediction tool (fourth column)
- the V parameter provided by the UFZ-LSER prediction tool (fifth column)
- the  $L_{16}$  parameter provided by the UFZ-LSER prediction tool (sixth column)

The data for each library constituent must be provided as a successive row. The ordering of the constituent data in the `mini_constituent_library_ESABVL.dat` file must match the ordering of the carbon number and class type data listed in the `mini_constituent_library_carbon_number_class.dat` file.

Two additional files reside in the `model_parameters/` folder which are used by the elution model code: `colorlist_class.dat` and `logL2_alkane_Nstar_parms.dat`. These two files are described briefly below.

The file, `colorlist_class.dat`, contains a list of 11 integers: these integers specify the sequence in which the chemical class information is plotted, according to the RGB array shown in the Matlab function, `colororder`.

The ordering of integers in the file corresponds to the ordering of class types provided in the function, `create_class_index.m`, which is contained in the `model_code/` folder. The values of the integers range from 1 to 7: the value corresponds to the index in `colororder`.

The file, `logL2_alkane_Nstar_parms.dat`, contains the two numerical parameters that are reported in equation 5 of Arey and coworkers.<sup>1</sup> The user should not change these values unless this update is merited by a re-determination of the parameters of equation 5.

## References

- (1) Arey, J. S.; Aparicio, A. M.; Vaiopoulou, E.; Forbes, S.; Lyon, D. Modeling the GC×GC elution patterns of a hydrocarbon structure library to innovate risk assessments of petroleum substances. *in preparation* **2022**.
- (2) *MATLAB version 9.11.0 (R2021b)*; The MathWorks Inc.: Natick, MA, 2021.
- (3) Arey, J. S. *GCxGC elution model*; Concawe, 2022. <https://github.com/concawe>.
- (4) Ulrich, N.; Endo, S.; Brown, T. N.; Watanabe, N.; Bronner, G.; Abraham, M. H.; Goss, K. U. *UFZ-LSER database v 3.2.1*; Helmholtz Centre for Environmental Research-UFZ, Leipzig, Germany, 2017. <http://www.ufz.de/lserd>.