**MSRESOLVESG Manual**

Contents

[INTRODUCTION TO THE MSRESOLVE PROGRAM: 2](#_Toc519240321)

[Mass Spectrometry: 2](#_Toc519240322)

[MSRESOLVE 2](#_Toc519240323)

[MSRESOLVE Purpose 3](#_Toc519240324)

[MSRESOLVE Quickstart Tutorial 7](#_Toc519240325)

[Overview of Functions and Capabilities 7](#_Toc519240326)

[Miscellaneous: 7](#_Toc519240327)

[Preprocessing 8](#_Toc519240328)

[Data Analysis: 9](#_Toc519240329)

[Preprocessing 12](#_Toc519240330)

[Time Range: 12](#_Toc519240331)

[Background Mass Fragments 13](#_Toc519240332)

[Correction coefficients: 14](#_Toc519240333)

[Chosen Mass Fragments: 15](#_Toc519240334)

[Background Fragments Baseline: 16](#_Toc519240335)

[Data Range Specifier: 17](#_Toc519240336)

[Marginal Change Restrictor 18](#_Toc519240337)

[Reference Pattern Changer: 19](#_Toc519240338)

[Mass Fragmentation Threshold: 23](#_Toc519240339)

[Data Threshold Chooser: 25](#_Toc519240340)

[Data Smoothing: 27](#_Toc519240341)

[Raw Signal Thresholds: 29](#_Toc519240342)

[Negative Analyzer: 31](#_Toc519240343)

[Data Analysis: 34](#_Toc519240344)

[Inverse Method: 35](#_Toc519240345)

[Sequential Linear Subtraction Method: 36](#_Toc519240346)

[Finisher: 37](#_Toc519240347)

[Brute: 37](#_Toc519240348)

[Inverse: 37](#_Toc519240349)

[Converting Relative Signals to Concentrations: 38](#_Toc519240350)

[Signal Simulation 39](#_Toc519240351)

[Appendix 1: JDX Converter 40](#_Toc519240352)

[Appendix 2: Data Generation (Module) 43](#_Toc519240353)

# INTRODUCTION TO THE MSRESOLVE PROGRAM:

## The Need for MSRESOLVE and what it offers

During “solving” of collected mass spectrometry spectra to extract concentrations, there are several sources of challenges.

1. For time dependent signal collection, pre-processing may be required (such as baseline corrections, smoothing, etc.)
2. There may be overlapping signals, making solution difficult.
3. In many cases a particular molecule’s calibration may not be possible or practical at the instrument where the collection is being done. This creates two challenges. Firstly, that one may need to rely upon an externally collected reference pattern that may not exactly match how the molecule fragments in one’s own instrument, and Secondly that one requires a method for converting the signals for that molecule into (approximate) concentrations even in the absence of a calibration.

MSRESOLVE addresses each of the above challenges, as follows:

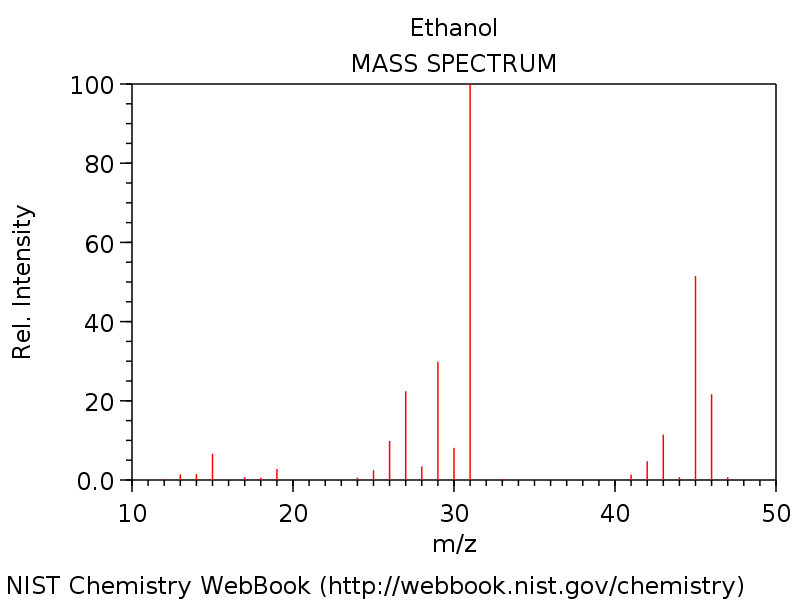
1. MSRESOLVE has various pre-processing functions, including baseline correction, and smoothing.
2. MSRESOLVE has several methods for resolving the signals: Sequential Linear Subtraction (SLS), the Matrix Inverse Method, and also brute force grid (regression).
   1. For the SLS and the inverse methods, it is possible to extract error bars for the final solved concentrations.
3. MSRESOLVE is able to convert mass spectrometry signals into a common concentration scale even for uncalibrated signals, provided that reference concentration patterns are available. Furthermore, MSRESOLVE can also correct for differences in mass spectrometers tuning.

## Suggested Procedure for Solving Time Series Data of Mass Signals

1. Create the files for the reference spectra and collected data – don’t delete any signals.
   1. When possible, use spectra directly calibrated on the instrument being used.
2. Create an MSRESOLVE run with SLS Unique and see what masse MSRESOLVE chooses, with ExportedSLSUniqueMassesUsedInSolvingMolecules.
3. If unsatisfied, start narrowing things down with chosen masses.
4. Also start using some reference file threshold filtering .
   1. UserChoices['minimalReferenceValue']['referenceValueThreshold'] = [1.0]  #this is what I am suggesting that you use.
5. If the application warrants doing so, include more sophisticated features of MSRESOLVE, such as mass spectrum tuning correction.
6. **Have Realistic expectations:** without direct calibrations of all molecules, the concentrations extracted are often simply semi-quantitative. In the absence of direct calibrations, finally extracted concentrations that are incorrect by 30% are very common due to differences in fragmentation patterns and tuning between mass spectrometers. MSRESOLVE is capable of accounting for some of these issues by mass tuning correction. It has been observed that masses higher than 50 can have responses that differ by a factor of 6 between mass spectrometers. Clearly, MSRESOLVE cannot account for this without a calibration file or without guidance from the user.

## Mass Spectrometry:

Mass Spectrometry is a widely used analytical technique in both academia and in research. Conceptually Mass Spectrometry functions by ionizing molecules and then, by taking advantage of the unique properties of ions, separating the newly form ions from the molecular flow and recording the molecular masses of these ions. When ionized, most molecules will fragment, and each formed fragment has a certain percent chance of retaining the ionizing charge and thus being recorded. This results in each molecule having a certain mass fragmentation pattern, the equivalent of a molecular fingerprint. That signifies the molecules presence. An example is shown below:



## MSRESOLVE

Mass Spectrometry can be used to monitor chemical reactions, to identify and characterize unknown compounds, and to determine the relative concentrations of molecular species. However, as the number of monitored molecules increases, greater overlap will occur among the mass fragmentation patterns increasing the difficulty and complexity of discerning individual molecules and their relative concentrations/signals from the overall raw mass signals. Efforts to do just this, analyze the mass signal generated by multiple molecules, is what led to the creation of the MSRESOLVE program. MSRESOLVE is designed to analyze and discern the molecules and their relative concentrations contained within raw mass signals. To do so MSRESOLVE requires reference mass fragmentation patterns and user inputted evaluation parameters.

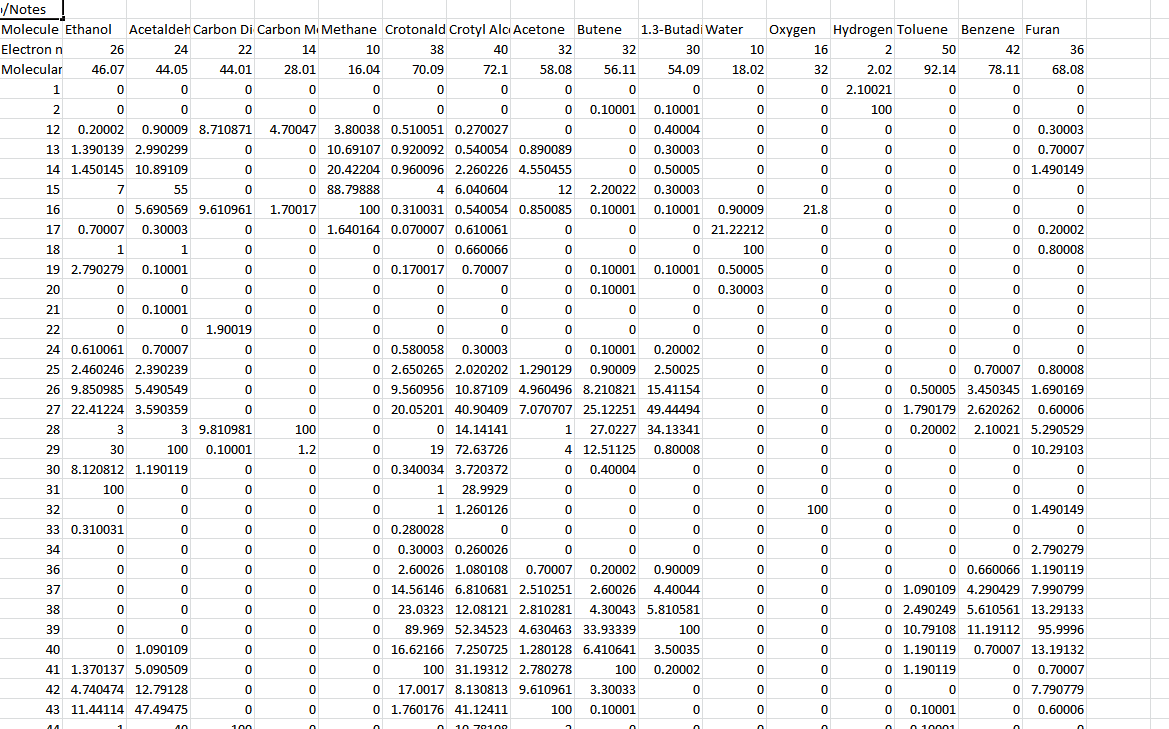
Along with the core MSRESOLVE program there are two auxiliary programs: the JDXConverter program which generates the required reference file and the Data Generation program which simulates the theoretical raw signals produce by molecules of various concretions. Neither program is required to use MSRESOLVE, but their use is suggested.

All programs are explained in the following pages including an extensive guide of the options and inputs of MSRESOLVE.

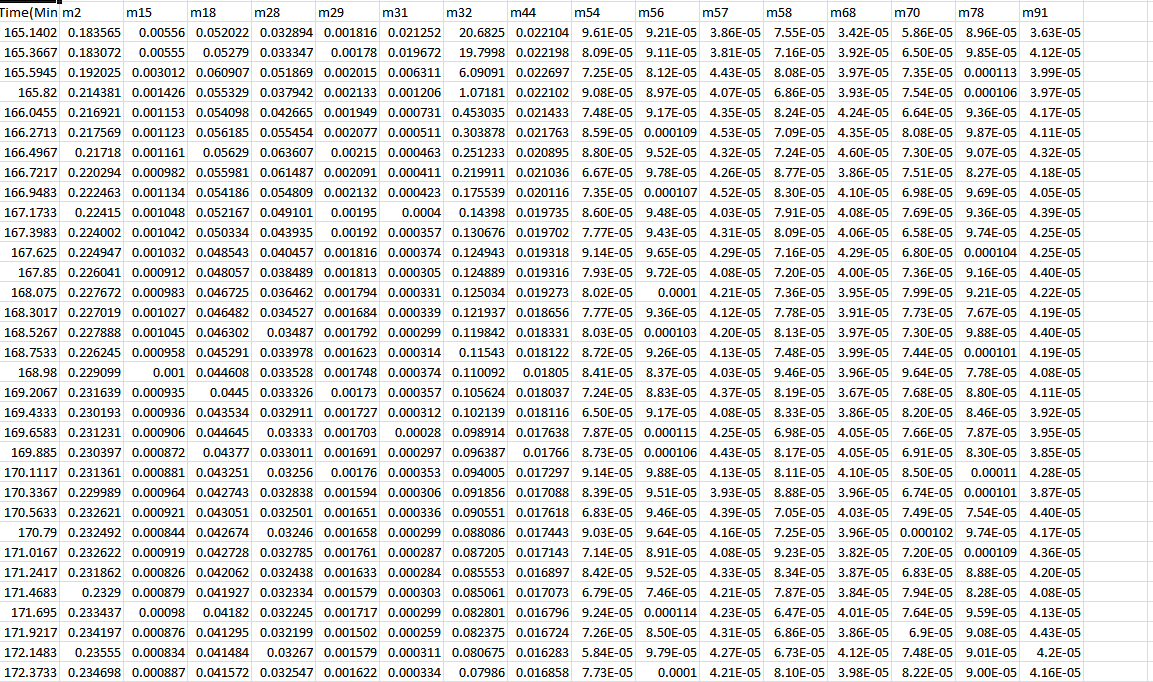
## MSRESOLVE Purpose

The MSRESOLVE program takes three inputs; a reference sheet (including reference fragmentation patterns), the raw signals from mass spectrometry and a text file detailing multiple options/choices for the MSRESOVLE program. The MSRESOLVE imports these three inputs and computes, according to the options chosen in the text file, the concentrations/relative signals of each molecule that was present in the Mass Spectrometer over the times the raw mass spectrometer signals were recorded.

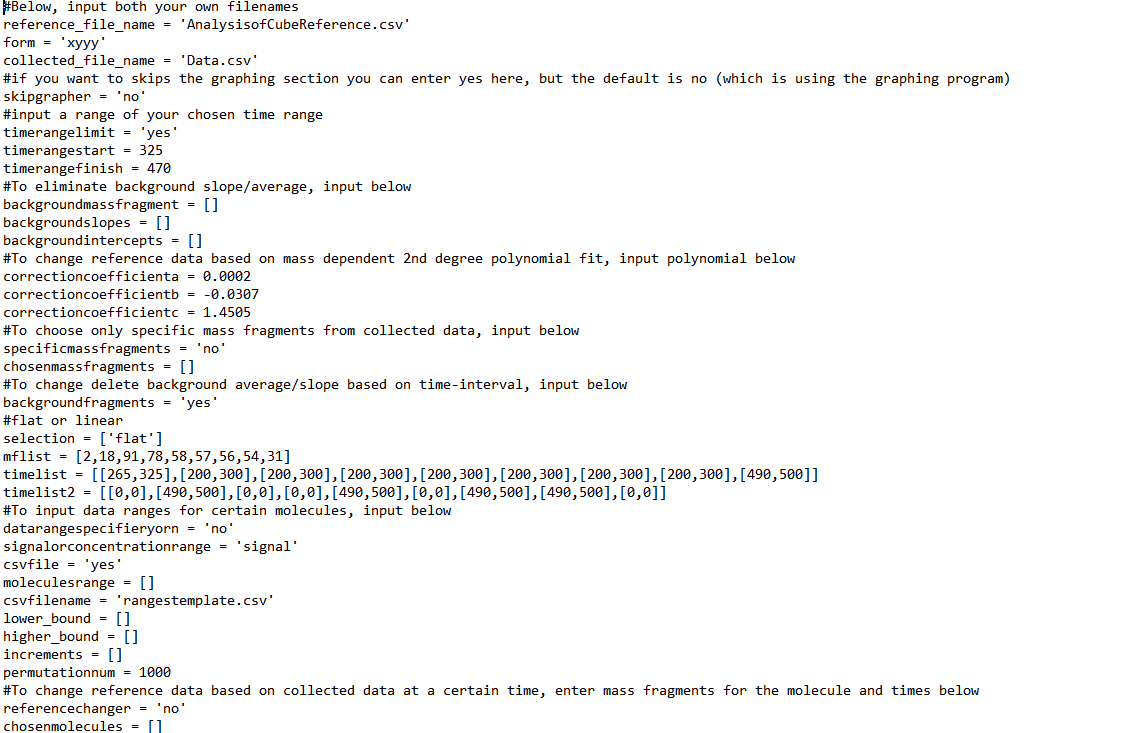
The input files used in the MSRESOVLVE program are displayed below:



(Left) The Reference File for an example experiment. Present are the mass fragmentation patterns, the electron numbers, and the molecular masses of each molecule included in the search..



(Left) An example of raw data from a Mass Spectrometer. The file displays the change of each measured mass fragment over time. This data is analyzed by the MSRESOLVE Program to determine the concentrations of each molecule at these corresponding times.

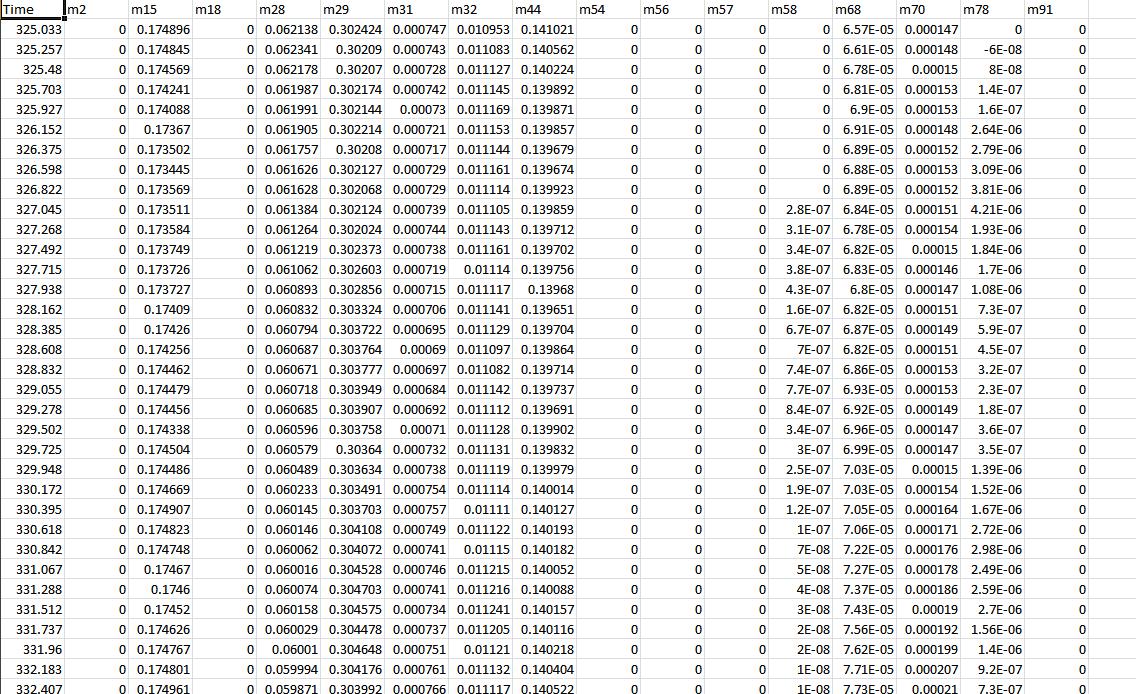


(Above) This is the UserInput File. In this file the User specifies the methods and parameters he/she wants to use to preprocess the raw data and to compute the concentration profiles. A more through introduction to the user input file will follow this section.

There are four possible output files from the MSRESOLVE program. The preprocessed data, the simulated signals output, and the concentration and signals output file-only one of which will be exported at a time.

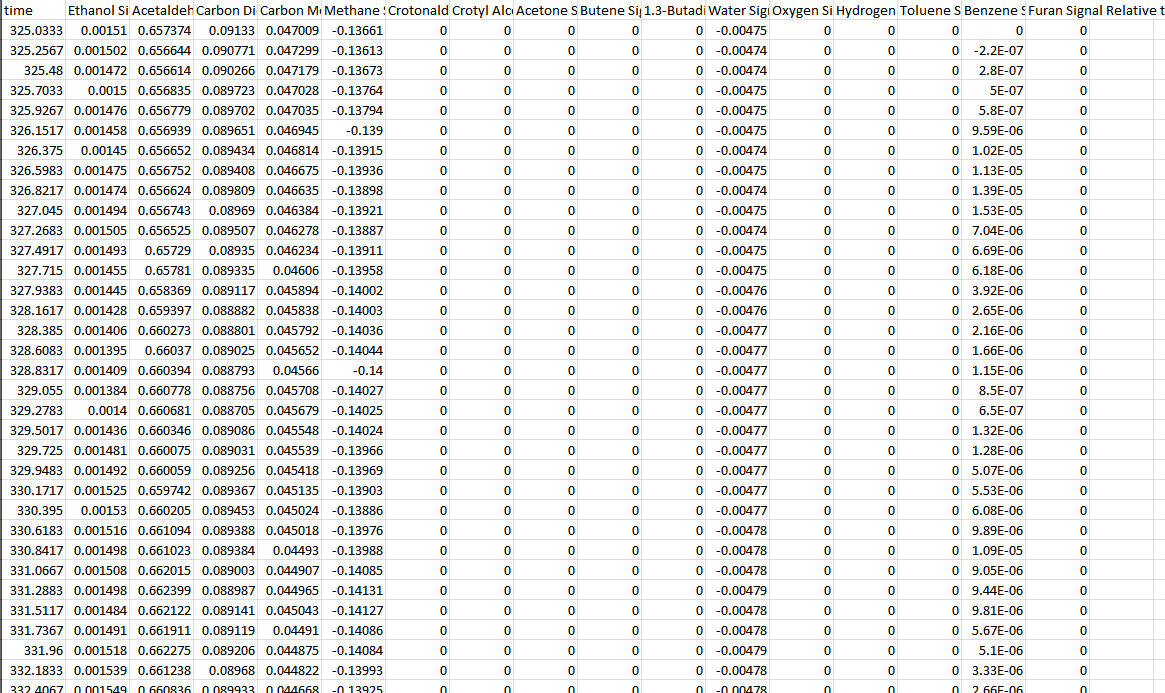
The preprocessed data displays the final iteration of raw signals before computing the concentrations/simulated signals but post all preprocessing and data smoothing. This is outputted to allow the user to determine the impact of the chosen preprocessing. The concentration/signals output file displays the computed concentrations or signals. Finally the simulated signals output file, similar to the data generation program, outputs simulated raw signals from the concentrations calculated. This file potentially could be compared to the initial input signals as a way to evaluate the validity of the calculated concentrations/signals.

Below are illustrations of all output files:

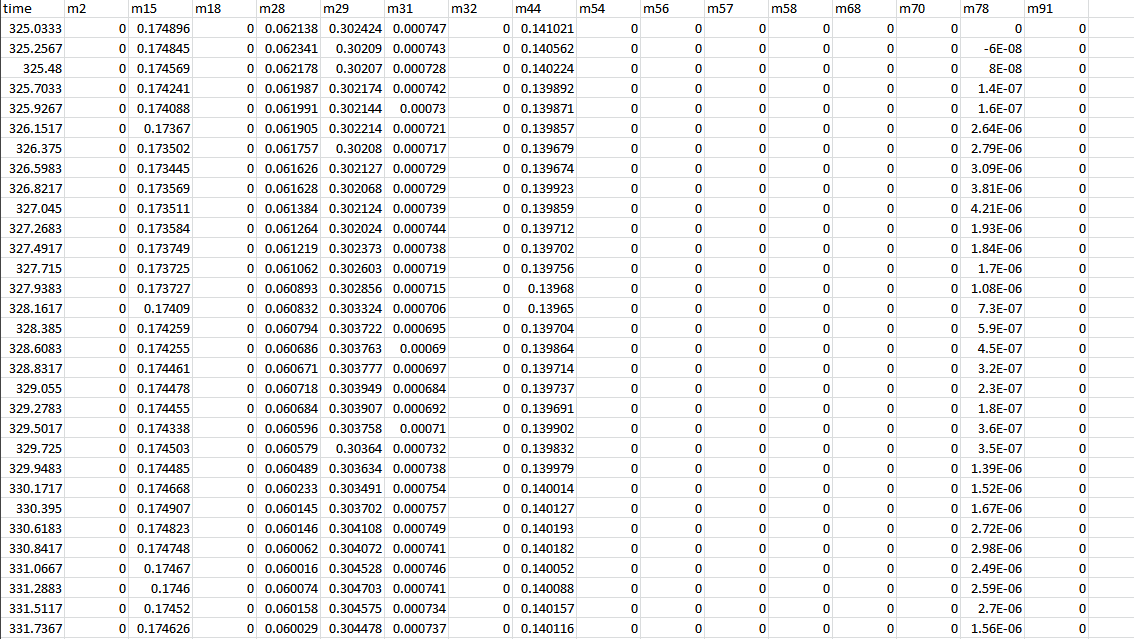


(Right) Preprocessed Data:

This sheet shows the results of the preprocessing on the raw data.

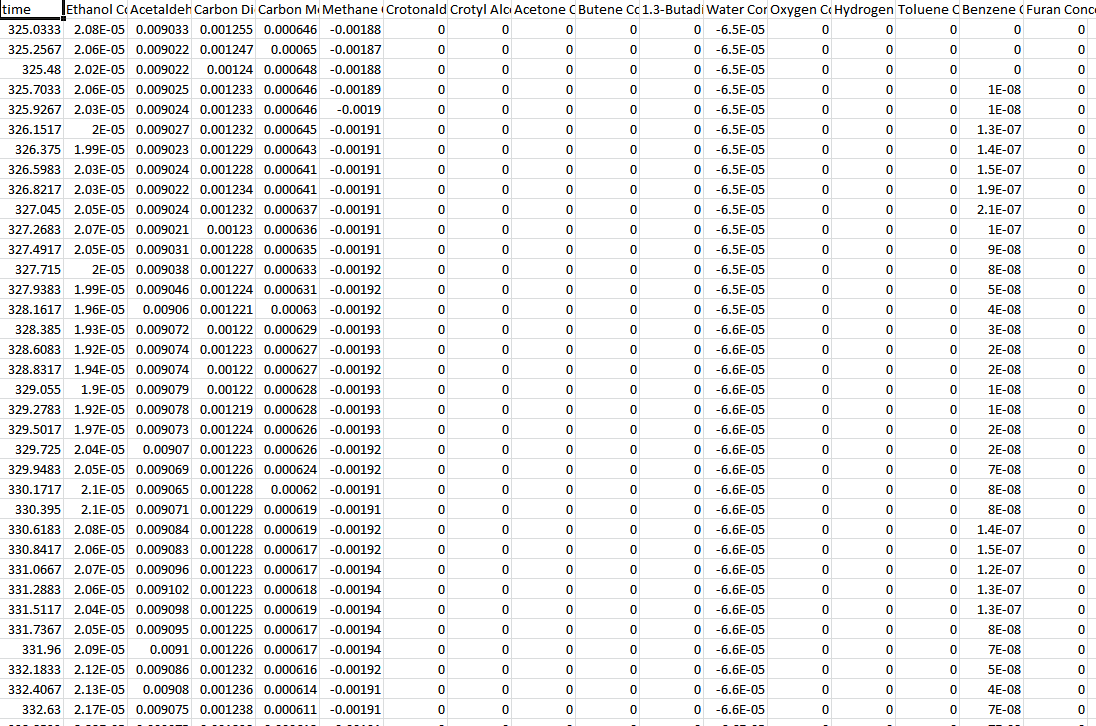


(Right) Signals Output: If the user searches for relative signals, this is the output. It shows the results of the calculations: the calculated signals of every molecule relative to each other.



(Right) Concentrations Output: If the user searches for concentrations, this is the output. It shows the calculated concentrations of every molecule.

(Right) Simulated Signals: This displays the Raw Simulated Signals generated from the earlier calculated concentrations/signals.



# MSRESOLVE Quickstart Tutorial

Included with the MSRESOLVE program is a “Tutorial” Directory. The word document in that directory will guide you through a first time quick start tutorial.

# Overview of Functions and Capabilities

The MSRESOLVE program can be divided into two distinct sections: data preprocessing and data analysis. Data preprocessing is where the raw mass spectrometry data and the reference data undergo a series of transformations and filters alleviating and removing random outliers, offsets, weak or disruptive signals, etc. Data analysis is where the calculation of relative signals and concentrations from the processed mass spectrometry data occurs. Preprocessing is essential for efficient and accurate data analysis.

In the data preprocessing sections and the data analysis section there are a myriad of capabilities and options all of which are controlled via the Input Text File. Thus an understanding and ease in using the Input File and the options it contains is a necessity to effectively deploy the MSRESOLVE program. The following is a step-by-step, function-by-function, guide of the Input File. The goal of which is instruct users on what each function does, briefly how it works, and, finally, how to use it.

## Miscellaneous:

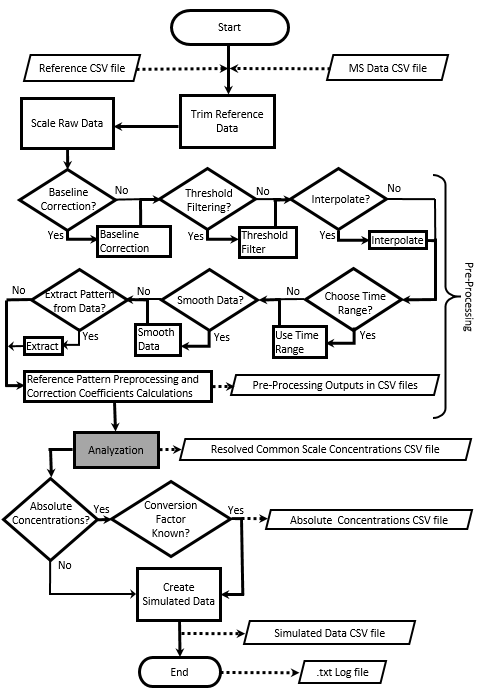
* Input Files: Select appropriate input files for experimental and reference data. The format of the experiment data must also be specified here.
* Program Execution: Here the user specifies which of the three primary sub-programs in MSRESOLVE should be executed. The sub-programs are pre-processing, data analysis and data simulation.
* Graphing: MSRESOLVE can produce plots of mass fragment signals and concentrations at various stages of execution. In this section the user can specify whether these plots should be created or not.

Preprocessing**:**

* Time Range: choose a specific range from the raw mass spectroscopy to analyze.
* Chosen Mass Fragments: choose which mass fragments, present in the mass spectroscopy data, should be used to calculate relative signals/concentrations.
* Baseline Correction
  + Linear Semiautomatic (Recommended): Here the user specifies only whether the corrections should be a linear or flat (slope of zero) correction; the mass fragments to be corrected and the time ranges in which the corrections should be applied. The user should specify *early* and *late* baseline times, and a baseline fit will be performed for all points in those ranges. The *late* baseline times can be omitted if the user only wishes to use one time range.
    - BaselineType: can be ‘flat’ or ‘linear’, a flat line means a baseline with a slope of 0, while a ‘linear’ baseline uses a fitted slope.
  + Linear Manual (Not Recommended): The user can manually specify baseline corrections for a selection of mass fragments. The user will need to specify the mass fragments to correct as well as the slope and intercept of the correction.
* Data Specifier: determine a maximum rate of change for mass spectrometer data.
* Data Solving Restrictions:
  + Marginal Change Restrictor: Limits the maximum change in the signal for a particular mass fragment between two data points.
    - Marginal Change Restriction: User specified factor that regulates the amount of change between the data at consecutive times. This defines the maximum accepted change.
    - Ignorable Delta Y Threshold: Threshold value that prevents evaluation of the marginal change restriction at and below this value.
  + Brute Solving Restrictions: Restricitons for the brute solcing method.
    - Lower/upperBound: prevent brute from searching above and below user specified bounds
    - Increments: sets the size of the increments for Brute (e.g., if we said 0.01 bar, it would make the separation between points 0.01 bar in the grid, for that axis).
* Set Scaling Factor: Gives the user the option to scale the data automatically or manually with specified scaling factor.
* Reference Correction Coefficients: experimentally determine coefficients that account for the bias of Mass Spectrometers for heavier molecules.
* Extract Reference Pattern from Data: Changes reference data for particular molecules based on collected data at a certain time, enter mass fragments for the molecule and times below.
* Reference Mass Fragmentation Threshold: removes all mass fragments below a certain numerical relevance from calculations.
* Data Threshold Filter: Evaluates all experimental data to determine if those expermental data are greater than a user chosen value (counts data smaller than that value as ‘0’, which makes the matrices easier to solve).
* Data Smoothing: Modifies mass spectroscopy data so that each point is modified by a line fit utilizing a user specified number of adjacent data points. The user can specify either a number of points (‘pointrange’ option) or a radius of time/temperature (‘timerange’ option) to be used in the data smoothing of a point.
* Raw Signal Threshold: eliminates recorded signals below a certain threshold or above another limit.

## Data Analysis:

* Negative Analyzer: prevents relative signals/concentrations from being calculated as negative.
* Data Analysis Methods:
* Inverse Method: method of solving for concentrations/ relative signals through matrix manipulation.
  + Distinguished: uses an algorithm to determine the best rows of a matrix from which to generate a smaller matrix
  + Combination: uses an average of all possible matrixes to generate a smaller matrix
* Sequential Linear Subtraction: method of solving for concentrations/relative signals by subtracting molecular signals from raw mass signals.
  + Unique: subtract one molecule’s signal at a time.
  + Common: subtract several molecules’ signal at a time.
* Brute: method of solving for concentrations/relative signals by using a grid search (numerical solver).
  + Sum of Square Residuals (ssr): minimizes the sum of square residuals.
  + Sum of Absolute Residuals (sar): minimizes the sum of absolute residuals.
  + Weighted: assigned error/magnitude to molecules to change value of the residuals.
* Concentration Finder: determines the conversion factor to convert simulated signals to concentrations.
* Output Files: In this section the use specifies the names for various output data file names.

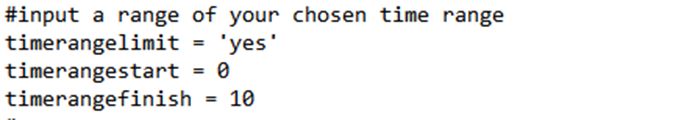


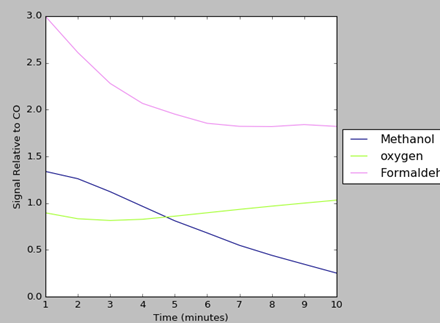
(Left) A Visual overview of the MSRESOLVE program, displaying the many possible functions contained within the preprocessing section.

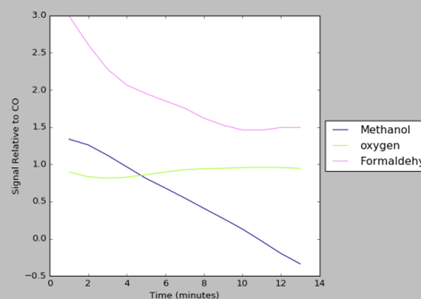
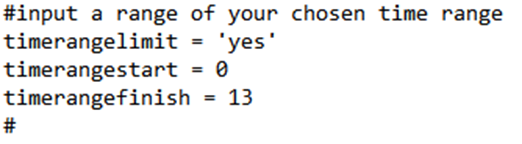
# Preprocessing

## Time Range:

The first option is to either analyze all of the raw signals or only the raw signals within a certain time range. The user should choose a time range that includes all relevant concentration changes without any data beyond the selected time range. This will allow the graphical representation generated by MSRESOLVE to scale to the significant data (the data the user wishes to see) and not the overall data set. Additionally a specific time range will minimize the data that undergoes preprocessing. This in turn will decrease the effects of non-relevant data on the outcome and the need for additional preprocessing.

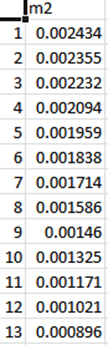




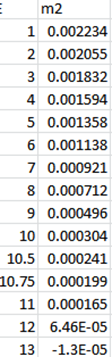


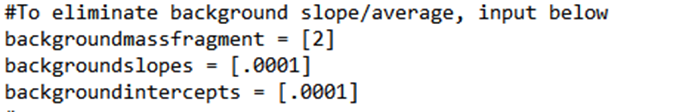
## Background Mass Fragments

Like many experimental techniques, Mass Spectroscopy is not without error or slight deviation from actuality. The magnitude of these errors may not be large but when they impact less important mass fragments or weaker signals, the distortion of results can be exaggerated. To rectify this situation, the BackGroundMassFragments method was instituted. The BackGroundMassFragments method is a form of preprocessing that modifies the raw signals to help eliminate the effect of error or lack of precision. A simple way to use this method is to graph a mass fragment, observe it, and determine a function/line that could replace/simulate/standardize it.



(Right) This illustration displays the changes of the mass fragment: 2, over time.





The code (above) modifies the second mass fragment as shown in the accompanying illustration (right).

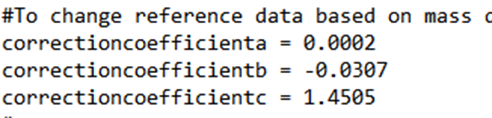
## Correction coefficients:

Mass Spectrometers innately favor heavier molecules over others. The computational methods in MSRESOLVE do not innately take the preferences of individual mass spectrometers into account and thus several correction factors are instituted to accommodate for this issue.

By graphing the mass fragments at known concentrations a 2nd degree polynomial can be determined that accounts for this issue. A 2nd degree polynomial was determine to be the most accurate possibility.

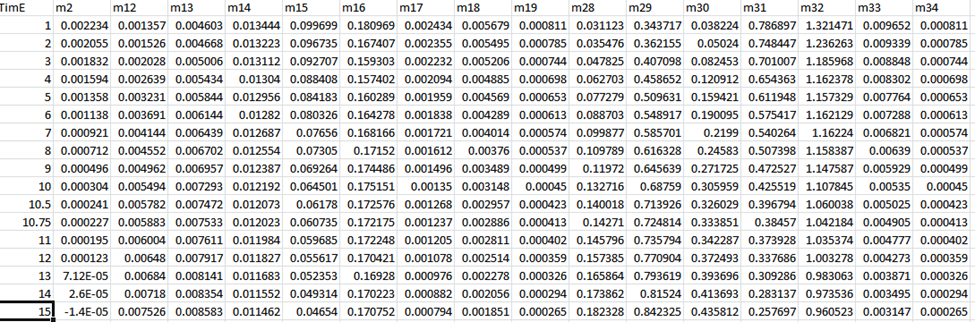
Code:

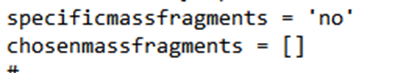
(Left) the correction coefficients correspond to the user determined 2nd degree polynomial from the form:



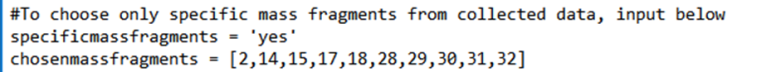
## Chosen Mass Fragments:

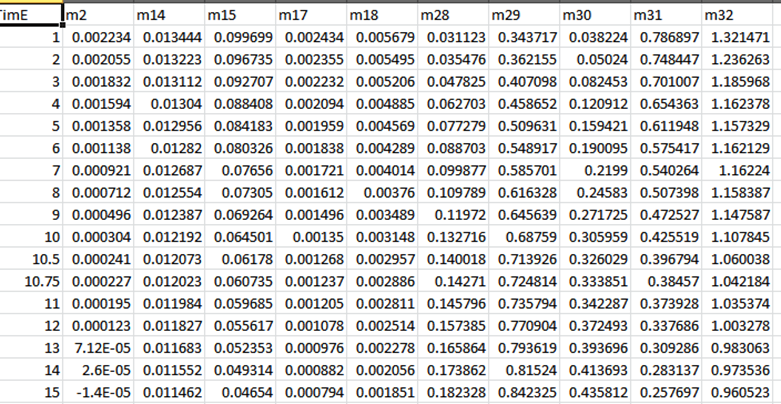
A user may not always wish to use all recorded mass fragments in calculating the mass spectroscopy computations. Some mass fragments may be of minimal importance and detract from the accuracy of concentration computations with their inclusion. Additionally some mass fragments may correlate with molecules that the user does not wish to evaluate or to display. Thus the user has the Chosen Mass Fragments option.





No fragments are selected in the code (above) so all fragments are displayed (right).



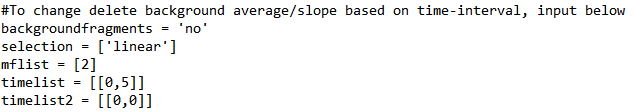
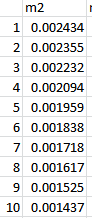


As shown in the code (above) the mass fragments chosen are the only mass fragments present the output file (right).

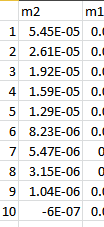
## Background Fragments Baseline:

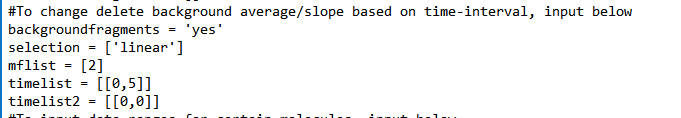
Raw data gathered from Mass Spectroscopy may show variations – in particular, offsets from actuality. To accommodate for these inaccuracies and specifically to correct any offsets, a user may use the Changing Background function. This function allows the user to modify the raw data of any mass fragment within two different time ranges. The underlying implication being that the user can correct/smooth any data offsets at the beginning and the end of the data signals. The user has two options for modifying the raw data, *Flat* or *Linear*. Flat approximates the data range with an average value, Linear approximates the data range with a Linear line.

Code: Results:



The original preprocessed data (right) without adjusting the background fragments as shown in the input text file code (left).





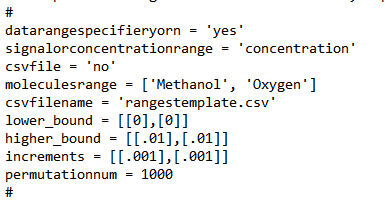
The new raw data (right) after the background corrections are performed (above).

## Data Range Specifier:

The Data Range Specifier enables the user to determine the maximum rate of change for mass spectrometer data. By default the MSRESOLVE program prevents mass spectrometer data from increasing by more than two fold. However there are situations where some users may wish for more specific limits. There are three parameters for the Data Range Specifier: lower bound, higher bound, and increments. The function works by limiting the data points from increasing/decreasing by more than their boundary value every increment.

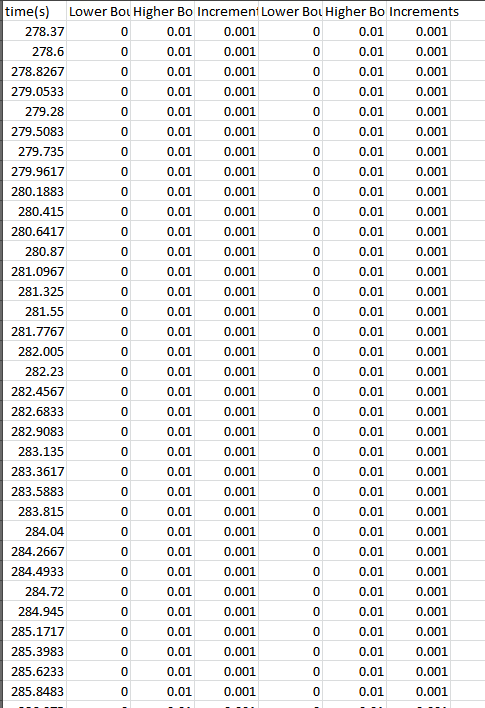
These parameters can be entered via the input text file (where the boundaries and increments are applied at all times to each molecule specified) or via a csv file.

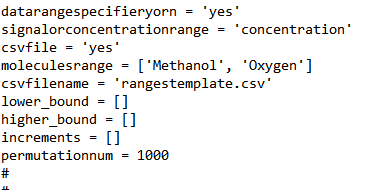
(left): Using the data range specifier to modify the spectroscopy data of Methanol and Oxygen.

Entering through text file:

(Bottom Left): Example of CSV file used by Data Range Specifier. The columns are attributed sequentially to the molecules contained within the molecules range. (Bottom Right) Utilizing the data range specifier with a csv file.

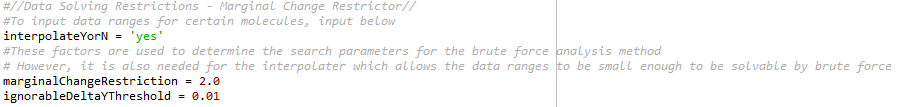
Entering through csv file:

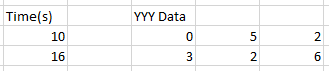




## Marginal Change Restrictor

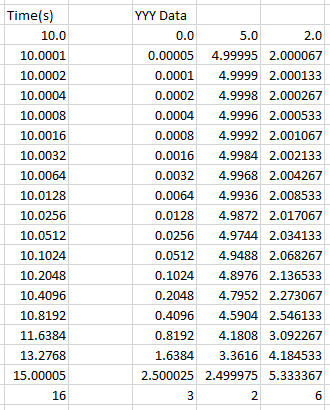
The marginal change restrictor prevents the signal intensities at each time from varying from the previous times’ signal intensity by more than a certain user-specified factor, the marginal change restriction. If there are fragment signals of consecutive times that vary by a factor larger than the marginal change restriction, the marginal change restrictor will insert interpolated data values and corresponding times between the two consecutive times that do not change by a factor larger than the marginal change restriction. To prevent the insertion of insignificant data values, the user can specify an ignorable delta Y threshold. MSRESOLVE will not evaluate consecutive data points having a difference below this threshold. Additionally, at YYY data values that are equal to 0, MSRESOLVE will insert new interpolated YYY data rows and the corresponding time values above and below the zero so that the values surrounding the zero are equal to half of the ignorable delta Y threshold





(Top) Specifying the marginal change restriction and ignorable delta Y threshold

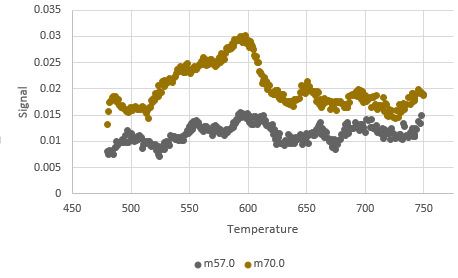
(Left) Example Time and YYY Data before the marginal change restrictor (top) and after (bottom)



## Reference Pattern Changer:

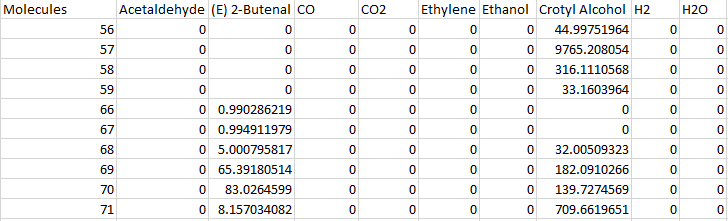
The reference pattern changer feature uses the function ExtractReferencePatternFromData to manipulate a specific mass fragment for a specific molecule in the reference fragmentation pattern. The function extracts average ratios of two or more intensities in experimental data in a user specified time range. These ratios are multiplied by the appropriate reference signal in the reference fragmentation pattern and the products are used as the new reference signals. This feature is useful when the ratio between two signals differ within different time ranges. The user can extract the ratio in one time range and apply it to the entire data set by manipulating that mass fragment’s reference signal.

Shown is an example using truncated data from a sample gathered during a reaction involving acetaldehyde, ethanol, and crotyl alcohol.

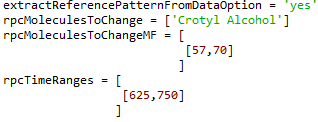


(Left) Graph showing preprocessed signals across temperature in experimental data for mass 57 and mass 70

(Below) Truncated Reference Data

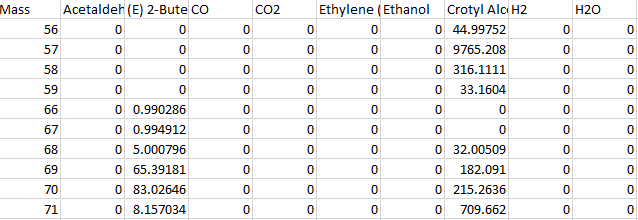


Notice the ratios between the two signals are changing from 475 to 625. This is probably due to there being more than one chemical species in the spectrum. From 625 to 750 the ratios appear to be steady and in this example will be interpreted to mean that there is probably only one chemical species present. In this example, we will take it as a given that crotyl alcohol is present. Since mass fragment 57 is unique to crotyl alcohol, we can assume crotyl alcohol is the only species present from 625 to 700, and that the peak of mass 70 from 475 to 625 represents the presence of some other species. Suppose the NIST reference files might not have the same response as our mass spectrometer. So we need to account for this difference. This ratio can be extracted and applied throughout the data set to account for crotyl alcohol’s contribution. After extracting this ratio, we can then remove crotyl alcohol’s contribution from all masses which, in this example, would enable the remaining masses (mass 70 in this case) to be representative of solely the other species present. So the user would input in the user input file as follows:



(Left) The syntax in the user input file shows the change of mass fragment 70 in the reference pattern for crotyl alcohol based on the ratios of intensities from the collected data at temperature range 625 to 750

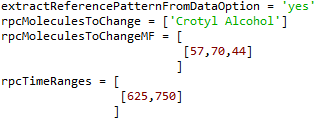
Now the reference data looks like this:

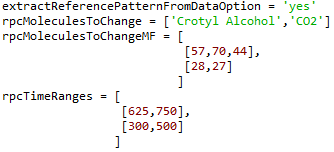


Note: mass 70 has been changed to about 215

If the user wishes to change two (or more mass fragments) with the base mass fragment then he/she should use the syntax on the top. If the user wishes to change more than one molecule they should use the syntax on the bottom.

(Left) Both reference signals for crotyl alcohol at mass fragments 70 and 44 would change based on the ratio of their average intensities to the average intensities of m75 from 625 to 750





(Left) As well as the changes occurring above, the reference signal at m28 for CO2 would also change based on the ratio of the average intensity of m27 to the average intensity of m28 from 300 to 500.

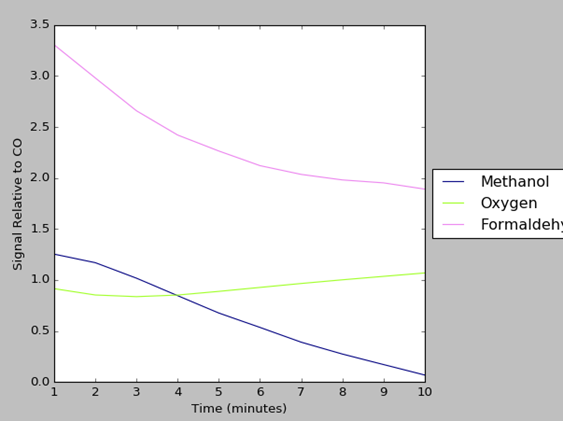
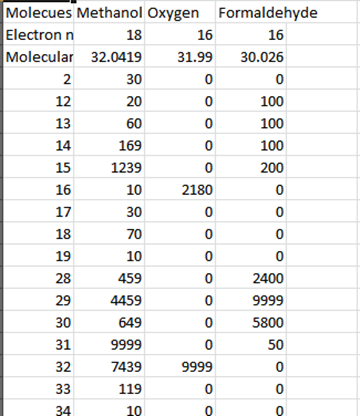
## Mass Fragmentation Threshold:

The Mass Fragmentation Threshold option removes all consideration of mass fragments that are below the entered numerical relevance for the evaluated molecules. By using a mass fragment threshold, the user can remove the consideration of minor, less important mass fragments, which can play an exaggerated role in the calculation of concentrations than their importance deserves.

The same data set discussed in the previous method discussion will be used as an example here:

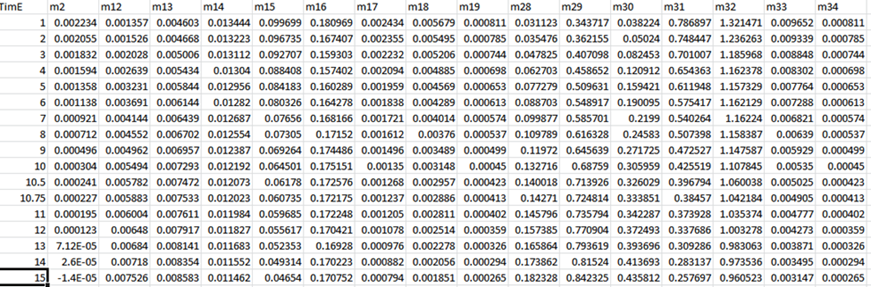
A theoretical data set modeling the reaction of methanol and oxygen (roughly twice as much initial methanol as oxygen) to generate formaldehyde is analyzed by the MSRESOLVE program. Initially, in the theoretical data, there is no formaldehyde present. With no preprocessing functions enabled, the initial profiles generated by the MSRESOLVE program are surprising. Formaldehyde is shown to have a higher concentration than any other molecule, and its concentration is shown to be decreasing.

After observing the involved molecules’ mass fragmentation patterns it seems reasonable to postulate that perhaps the many overlapping mass fragments — most of which are minor in value — are somehow distorting MSRESOLVE’s analysis. Especially since in this theoretical exercise, all mass fragments possible were recorded and utilized in the initial calculations.



(Right) The reference Fragment patterns for all species involved. (Below) the entered data includes all possible mass fragments (practically it is unlikely an experimenter would be able to record so many mass fragments-but this is a theoretical exercise).

(Left) the results from the first calculations.

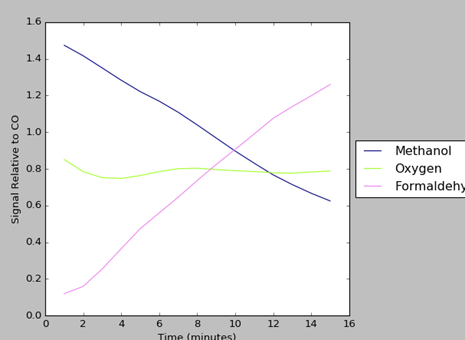
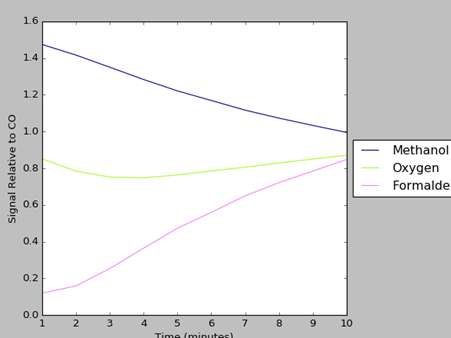


But what if all mass fragments were not used to determine the concentrations of the inolved molecules? This is the premise of the Mass Fragmentation Threshold. If, for this expermint, the Mass Fragmentation Threshold is set to 1000 then only fragments with a signal stregth greater than a 1000 are considered for calcluating each molecule’s concentration.

(Reft) code setting Mass Fragment threshold in user input



This threshold eliminates the majority of mass fragments from calculation considerations and leads to a much more accurate model. All experimental trends are present and are well represented.



(Above) When relative signals are calculated with a Mass Fragment Threshold, the results are much more accurate. (Above right) The same data set but extended to time 15.

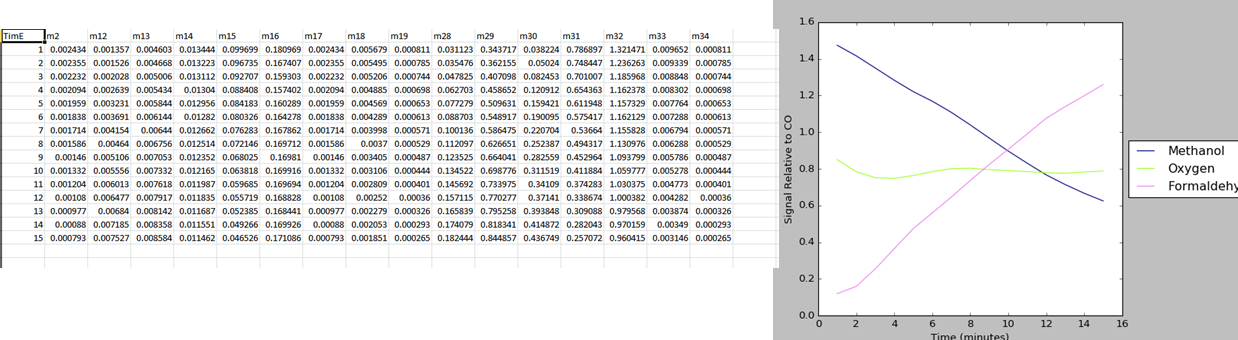
## Data Threshold Chooser:

The Data Threshold Chooser evaluates all experimental data to see if said experimental data is greater than a user chosen value, or threshold. If the experimental data are greater than the threshold value it remains unchanged, if the experimental data are less than the threshold value, the data is set to zero. This leads to more accurate calculations and decreased “noise” and influence from less important mass fragments.

The same data set discussed in the previous method discussion will be used as an example here (the preprocessing described previously is applied here):

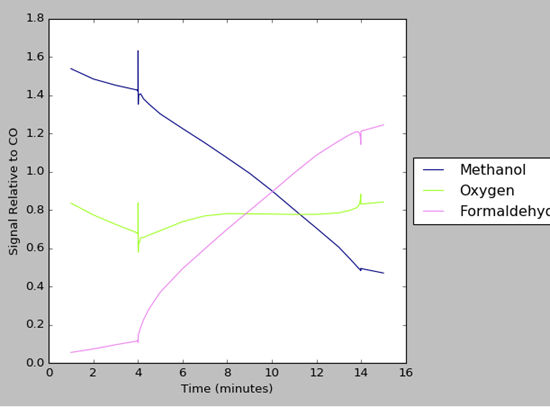
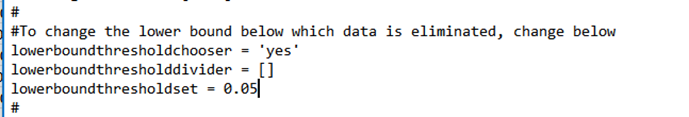
A theoretical data set modeling the reaction of methanol and oxygen (roughly twice as much initial methanol as oxygen) to generate formaldehyde is analyzed by the MSRESOLVE program. Initially, in the theoretical data, there is no formaldehyde present.

The initially calculated signals demonstrate the overall correct trends:

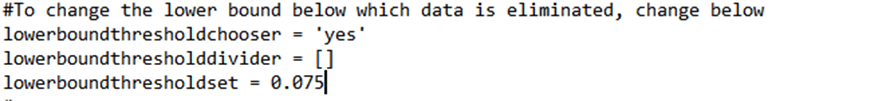


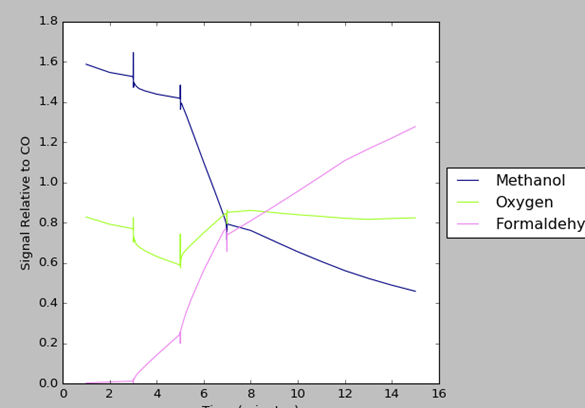
(above) The calculated signals from the initial run of MSRESOLVE

But this output lacks dynamism and is overly simplified. To gain a more realistic version the user can attempt to decrease the system noise and impact of minor, less important mass fragments by using the Data Threshold Method.



(above) setting the Data Threshold to .05 generates results (left) which demonstrates a higher level reaction order and is overall more representational of the reactions reality.





(above) Setting the Data Threshold to .075 generates results that illustrate more details about the reaction (left)

As seen in the illustrations above, by increasing the the value of the threshold chosen, the greater the impact of the significant mass fragments leads to more descriptive and more accurate concentration profiles. The spike in the reaction rate around four minutes and decreasing around seven minutes accurately models the theoretical data input. There are tradeoffs from increasing the threshold values. Important data points may be lost and a decrease in graphical quality can be observed.

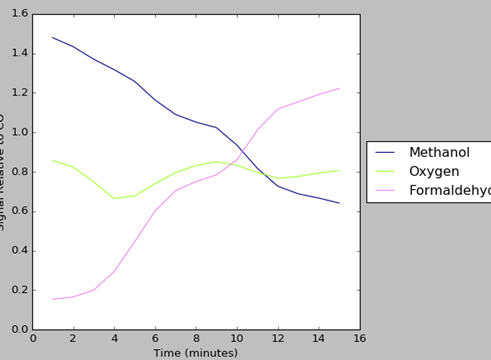
## Data Smoothing:

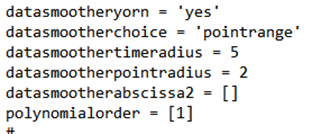
The Data Smoothing function is a function that modifies acquired mass spectroscopy data so that each data point of the acquired mass spectroscopy data is modified by a line fit utilizing a user specified number of the surrounding data points. This prevents outliers or errors from impacting calculations and creates a “smoother” graphical representation. This is one of the few data proccessing functions enabled by default.

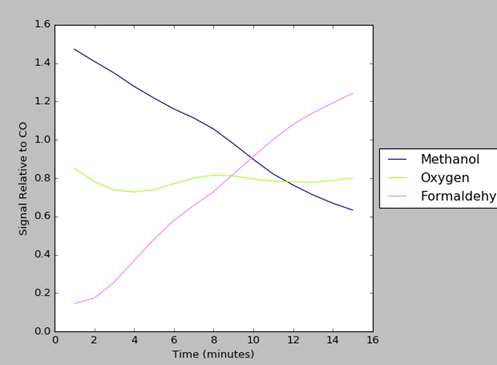
Below various configurations of the datasmoothing function and their effects are displayed. The same data set is discussed in the previous method and this discussion is used as an example here. (The preprocessing described previously is applied here.)

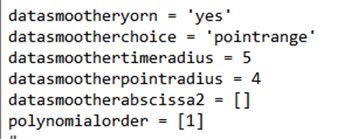
A theoretical data set modeling the reaction of methanol and oxygen (roughly twice as much initial methanol as oxygen) to generate formaldehyde is analyzed by the MSRESOLVE program. Initially, in the theoretical data, there is no Formaldehyde present.

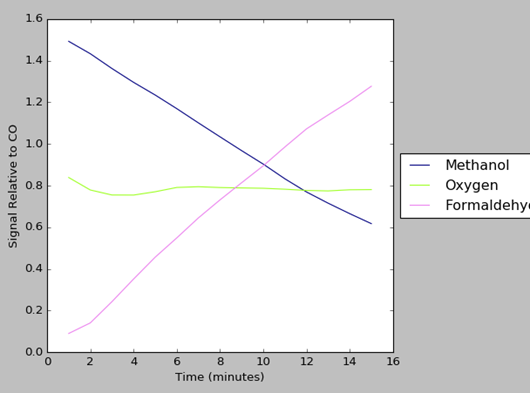
There are two ways to determining the local size for line fit adjustment *point range* or *time range*. *Point range* will use the specified number of points in point radius to generate the line fit. *Time range* will use a certain time period, determined by the time radius, to generate the line fit.

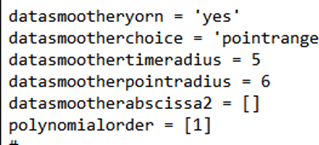








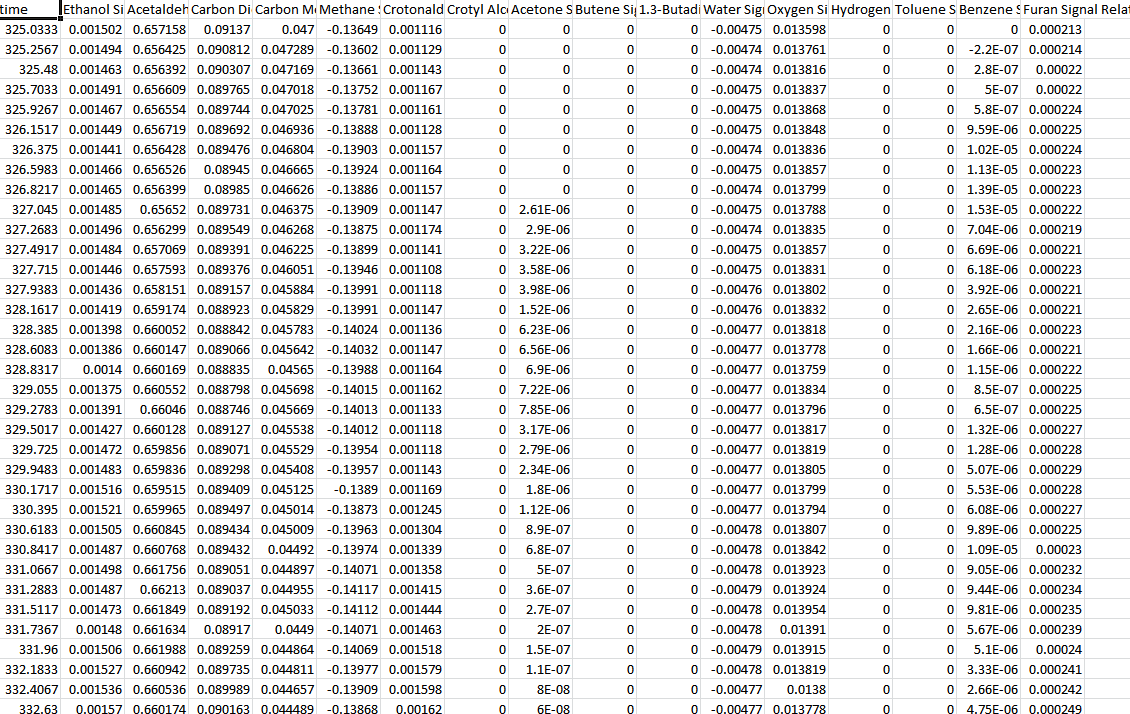




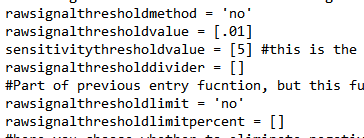
As demonstrated in the above illustrations, the greater the range of data points being used for line fit, the more standardized as a whole all data points become. The graphical representation of calculated signals/concentrations displays the increasingly linear nature or “smoother” nature of the data. With smaller data sets (such as this one which only has 15 total data points), high levels of smoothing can greatly impact the output data. However data smoothing is a very useful and needed function, especially when working with larger experimental data set.

## Raw Signal Thresholds:

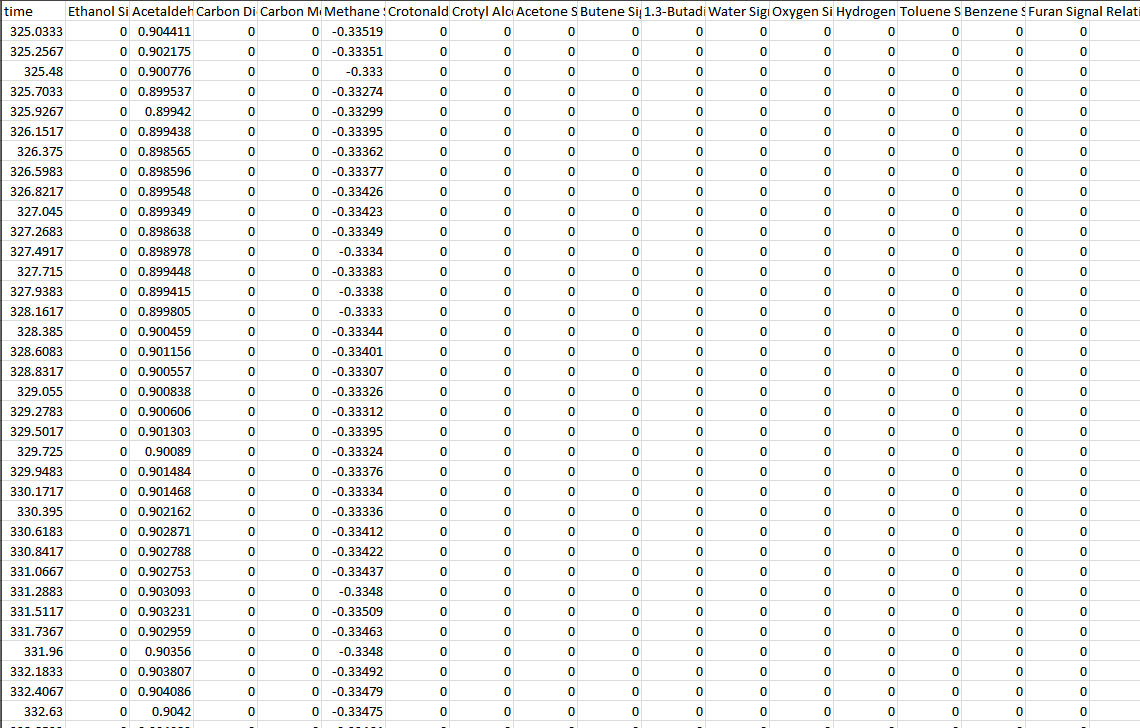
The Raw Signal Threshold eliminates any recorded signals below a certain threshold. This will eliminate any unwanted minor signals. Signals above a certain threshold can also be deleted. A user may want to delete a large mass fragment signal if it overlaps and thus obscures the mass fragments signals of other molecules.



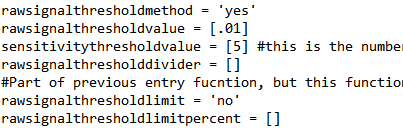
(right): the calculated relative signals with no Raw Signal Thresholds. (Below): the input text file specifying the section detailing the Raw Signal Threshold



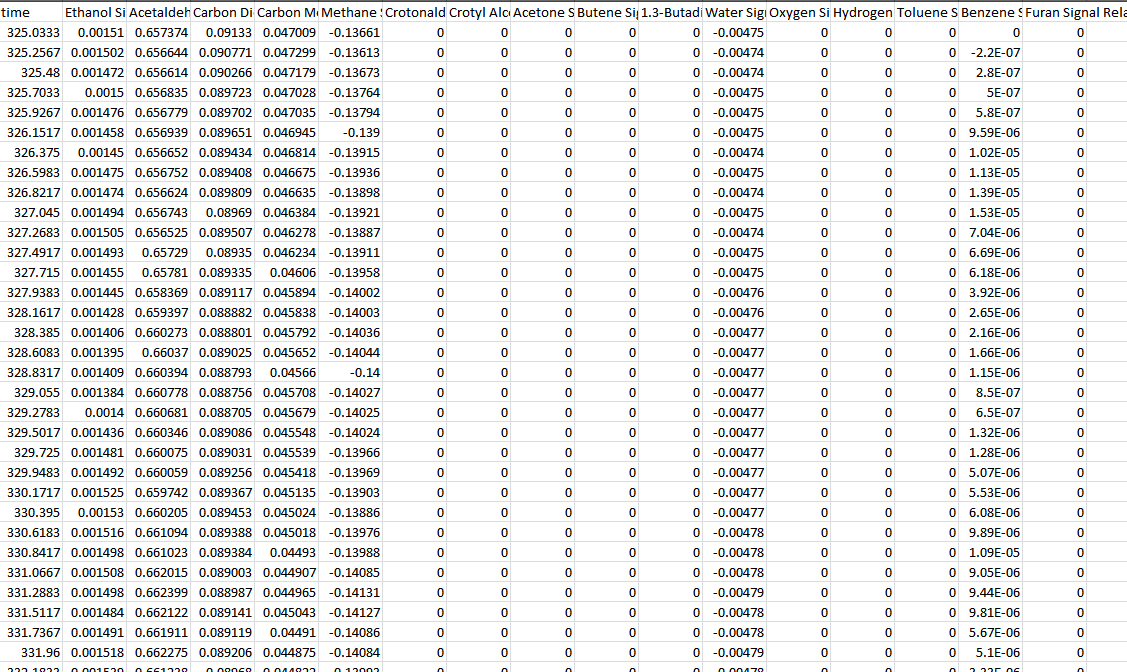
There are two options to utilize the signal threshold. The user can choose a raw threshold value that is then compared to all signals. The user also can choose a raw signal threshold divider which will divide all signals before being compared to the raw threshold value.



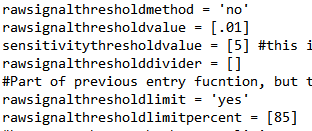
(right): the calculated relative signals displaying the effects of using the Raw Signal Threshold. (Below):the input text file demonstrating the use of the



The threshold limit is based on the percentage a molecule contributes to a mass fragments signal. Thus if the threshold limit is set to 80% and a molecule contributes 83% of a mass fragment’s raw signal then that molecule’s contributions are set to zero.



(right) The resolved signals generated from using a signal threshold limit of 85%. (below) Implementing the threshold limit.

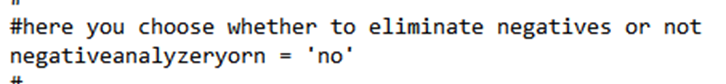
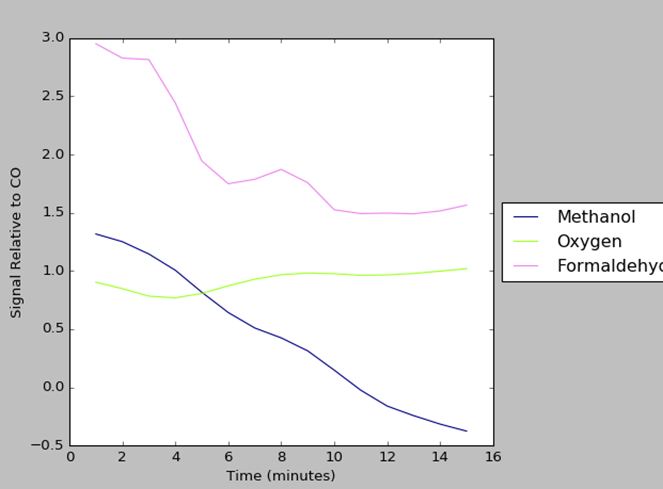


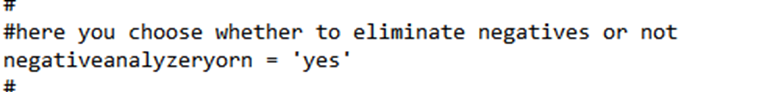
## Negative Analyzer:

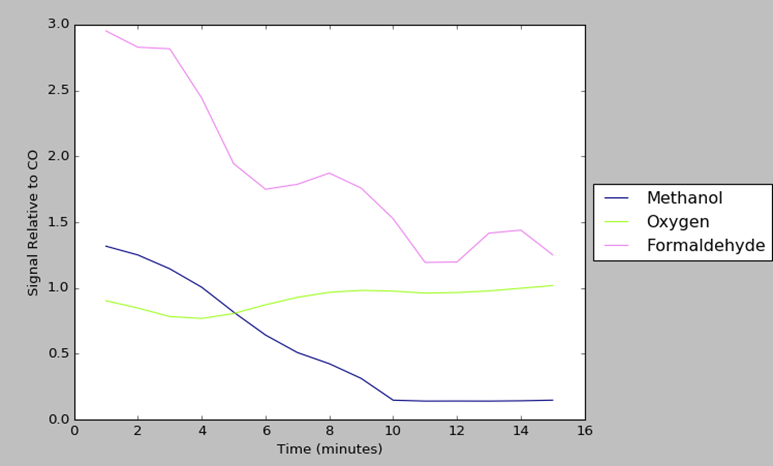
The Negative Analyzer function prevents negative values from being generated by MSRESOLVE when determining signals or concentrations profiles.

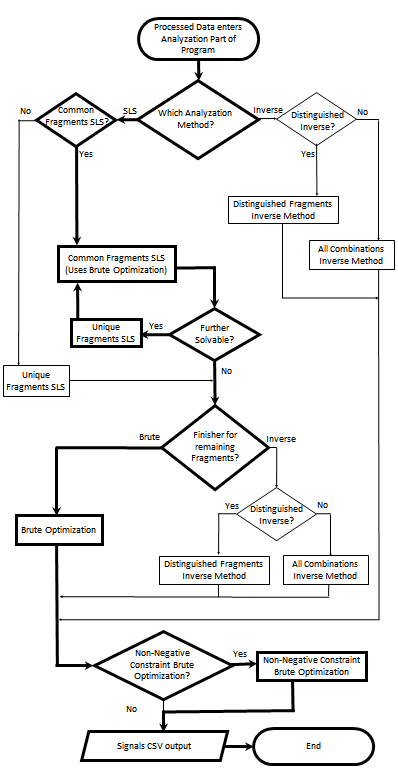
Below is a simple example displaying the effects of the Negative Analyzer. The same data set discussed in the previous method discussion is used as an example here. (There is NO preprocessing.)

A theoretical data set modeling the reaction of methanol and oxygen (roughly twice as much initial methanol as oxygen) to generate formaldehyde is analyzed by the MSRESOLVE program. Initially, in the theoretical data, there is no formaldehyde present.









(left) A Visual overview of the Data analysis section. The bold line represents the suggested path.

# Data Analysis:

After processing the raw mass spectroscopy data, a set of equations can be found that includes the correction factors determined via Madix & Ko method, the processed Raw Signals, and finally the ,currently unknown, calculated signals relative to CO. The point of the data analysis section is to solve this system of equations.

Raw Signal

Signal Relative to CO

C = Correction Value

(Eqn. 1)

The correction values are calculated using the following equation from Madix and Ko:

(Eqn. 2)

Eqn. 2 here is a rewritten form of the Madix and Ko equation that denotes the dependences of each variable in the function. The original Madix and Ko publication can be found using the doi: 10.1016/0021-9517(80)90454-6. In addition, the documentation folder contains an Excel spreadsheet with an example of the correction value equation applied to a data set.

There are two distinct methods for data analysis: the Inverse Method and the Sequential Linear Subtraction Method (SLS Method). Both routes have comparative advantages and disadvantages. The SLS method is more accurate and is able to accommodate larger ranges of data. However the Inverse Method is notably faster. Both routes are fully explored below.

## Inverse Method:

Inverse:

The inverse Method computes the relative signals by computing an Inverse Matrix and solving. The Numpy library is used to facilitate this operation.

(left) Choosing the inverse method in User Input



If there are more mass fragments (from Mass Spectroscopy data) than there are relative signals being solved for, then a new matrix must be created using some method to determine the construction of the new inverse scaling factor array.

Distinguished Inverse vs Combination Inverse:

When using the inverse method, if there are more mass fragments then molecules being searched for, then a new matrix must be created that has equal rows to the number of molecules. MSRESOLVE has two methods to accomplish this necessary matrix consolidation; the distinguished inverse method and the combination inverse method.

(left) ‘Yes’ chooses distinguished method and ‘no’ chooses combination method.



The distinguished method uses an algorithm that picks the “best” rows. This algorithm functions by multiplying each value in a column by the sum of the ratios minus one of all other numbers in the values row. The function then sorts all values by magnitude and picks, for each column, the row with the highest value (if a row is already chosen by another column then it will select the row with the next highest value and so on).

If the combination method is used, then all possible combinations of the rows are computed and the average from this different combination is found and used as the new matrix.

## Sequential Linear Subtraction Method:

SLS:

The SLS Method works by subtracting the signal of a molecule(s) from the raw signal and creating a smaller localized matrix that can then be easier solved.

(left) Choosing sls as the data analysis path.



Unique & Common SLS:

There are two approaches to SLS, unique and common. In unique SLS only one molecule is subtracted at a time. This allows solving by simple division, where the molecule’s contribution to the mass fragment raw signal is divided by the inverse scaling value. It should be apparent that there are many cases where this approach will not be able to resolve all of the signals. In common SLS several molecules with overlapping mass fragments are subtracted out. These removed fragments are then solved via brute optimization. As with unique SLS it is likely that not all of the mass specs data will be resolved. A secondary method or a “finisher” must be used.



(left) An illustration of choosing common for the sls approach.

## Finisher:

When using the SLS method to resolve mass fragment data into relative signals and concentrations there will likely be some unresolved equations or mass fragment data that the SLS method fails to resolve. Thus MSRESOLVE uses a secondary cycle of processing, a ‘Finisher’ to better resolve these signals.

### Brute:

Used in the earlier common SLS method, Brute optimization also can be used as an overall Finisher for the generalized SLS method. Brute optimization is a grid search styled method that numerically solves any system of equations. Brute optimization is the most effective method for resolving equations, but it also has the greatest time cost of any method (this is why Brute optimization is not offered along with SLS or inverse as an initial option to resolve the relative signals/concentrations). There are four variations of Brute option:

Sum of Square Residuals (ssr): goal is to minimize the sum of square residuals of each molecule in grid search.

Sum of Absolute Residuals (sar): goal is to minimize the sum of the absolute residuals of each molecule in grid search.

Weighted Sum of Square Residuals (WeightedSSR): goal is to minimize the sum of square residuals after they have been adjusted by the relative sizes of the signals

Weighted Sum of Absolute Residuals (WeightedSAR): goal is to minimize the sum of absolute residuals after they have been adjusted by the relative sizes of the signals

(left) choosing the ssr option for the sls finisher.



### Inverse:

The alternative to using the Brute option is to use the Inverse method which is the same method described earlier in this unit. As stated earlier, there are two versions of the Inverse method for the user to choose from here: the distinguished Inverse method and the combination inverse method.

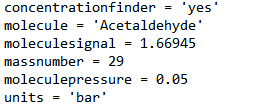


(left) Choosing the inverse method as the finisher for the sls method.

Note: If not using the sls method, the user must still mark distinguished as no or yes.

## Converting Relative Signals to Concentrations:

To convert the relative signals to concentrations, a conversion factor needs to be determined. To acquire this conversion factor, the user will have to run a molecule through the Mass Spectrometer at constant pressure and record the mass spec signal. Once this information is attained then the user simply needs to enter it into the input file and enable the concentration finder.



# Signal Simulation

The program is also able to simulate the experimental signals from the solved concentrations after the analysis has been completed. This feature can be useful to determine the accuracy of the programs analysis by comparing the signal simulation directly to the experimental data.

To select this option:

Example of the User Input File with the data simulation enabled. The output name determines where the simulated signals will be stored.





# Appendix 1: JDX Converter

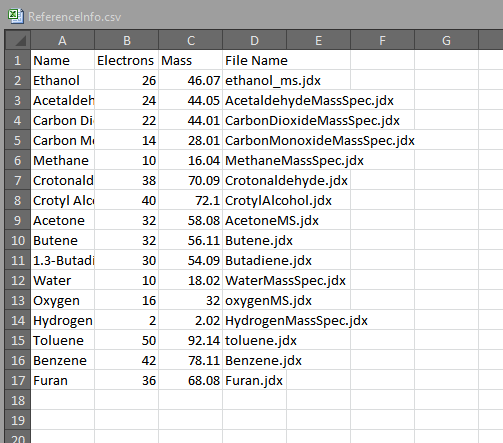
**Overview:**

The JDX Converter is a program that generates the standard Reference Fragmentation pattern csv file used in the MSRESOLVE program. An MSRESOLVE User does not have to use the JDXconverter to generate his or her reference data but it is highly suggested. The program is composed of two python files - JDXConverter.py and JDXConverterUserInterface.py - and is compiled and run from the command line. The JDX program has two possible input files: 1) the jdx files of all molecules that are desired; and 2) an optional csv sheet that details the molecules desired and some additional information. Without this file, the user will have to manually enter necessary information about each molecule. In this set of documentation, the output file is titled *simulatedData.csv* and the optional csv file is titled *ReferenceInfo.csv*.

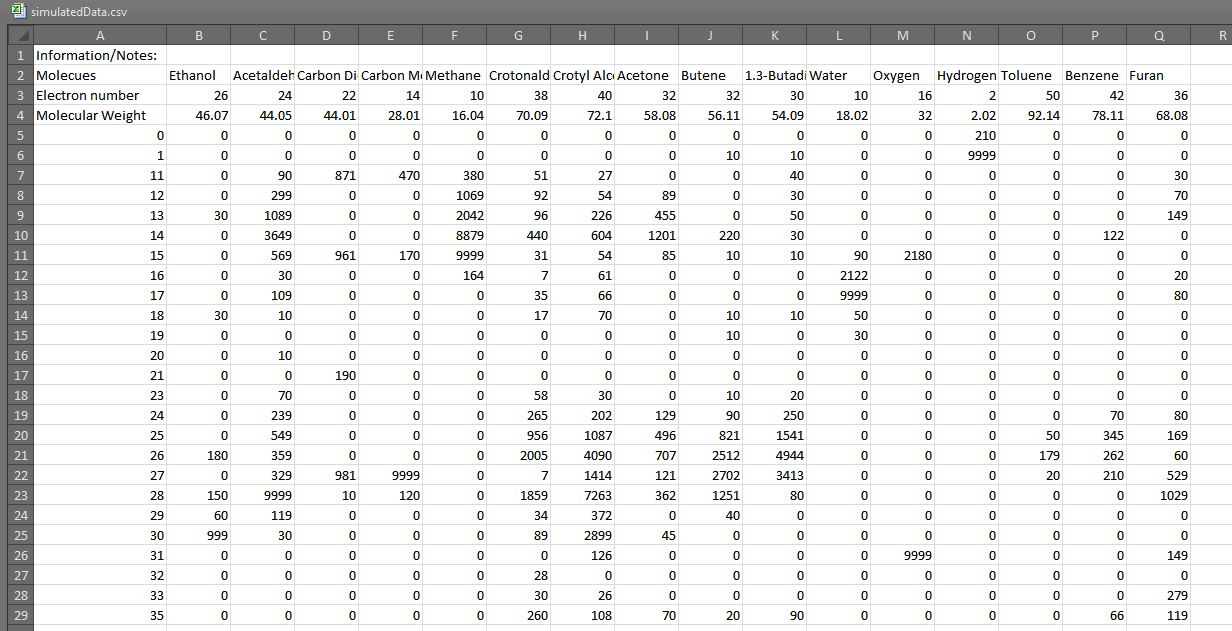
Note:

One may not be familiar with jdx files. jdx files are the standard format Mass Spectrometry data as stored by the NIST Chemical Inventory. It is recommended that all users attain their literature from this source.

Below are illustrations of both the *ReferenceInfo* sheet and the *simulatedData* sheet. Refer to these illustrations for aid in formatting.



(Left) the layout of the reference info file



(Right) An example of a generated reference file

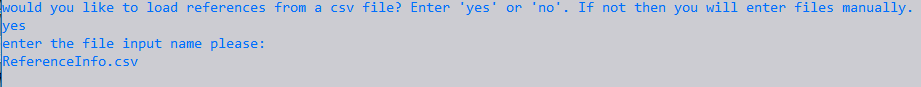
**Operation:**

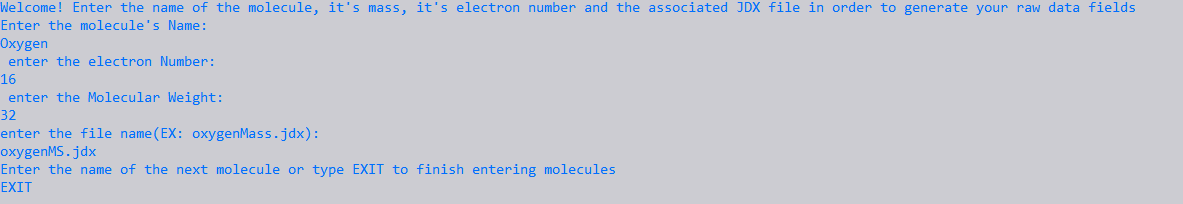
The JDX Converter program must be run from a command line that contains Anaconda. Before compiling, move all jdx files to the same location as the JDXConverter files. Begin the program by running the JDXConverterUserInterface:



Upon initialization, the user is faced with a choice: To load a csv file (such as the ReferenceInfo.csv file above) or to enter the files manually.

If the user chooses yes he/she will type in the csv file’s name and the program will generate a reference fragment sheet and terminate.

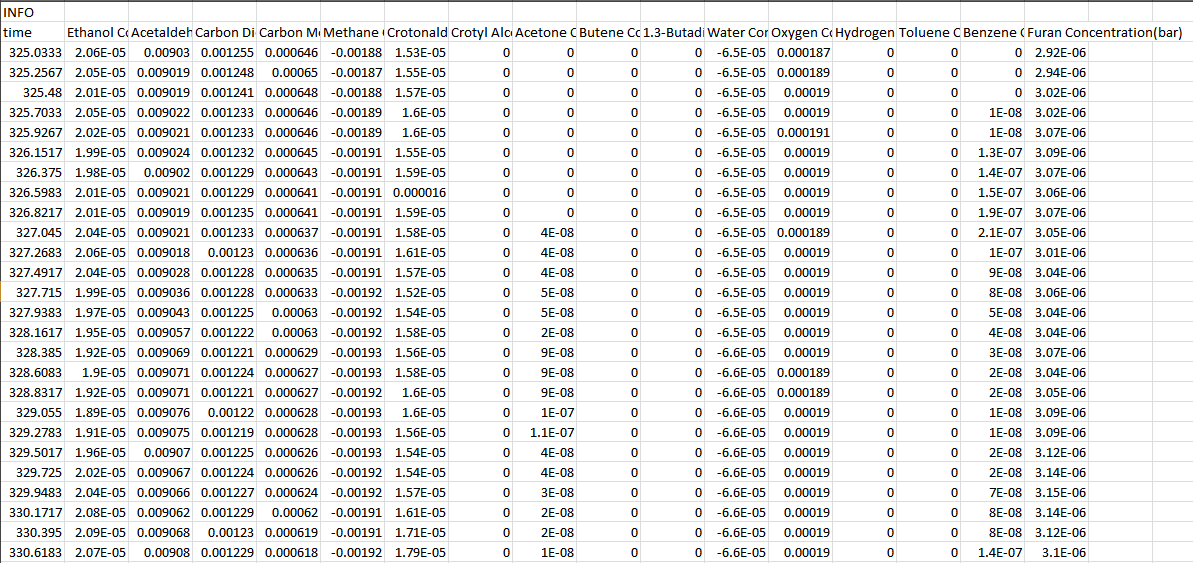


If the user does not wish to submit a csv file with the molecules (formatted as is demonstrated above) he /she will type no and then will be prompted for information about the molecule. This system of prompts will continue until it is manually terminated by the user typing EXIT.

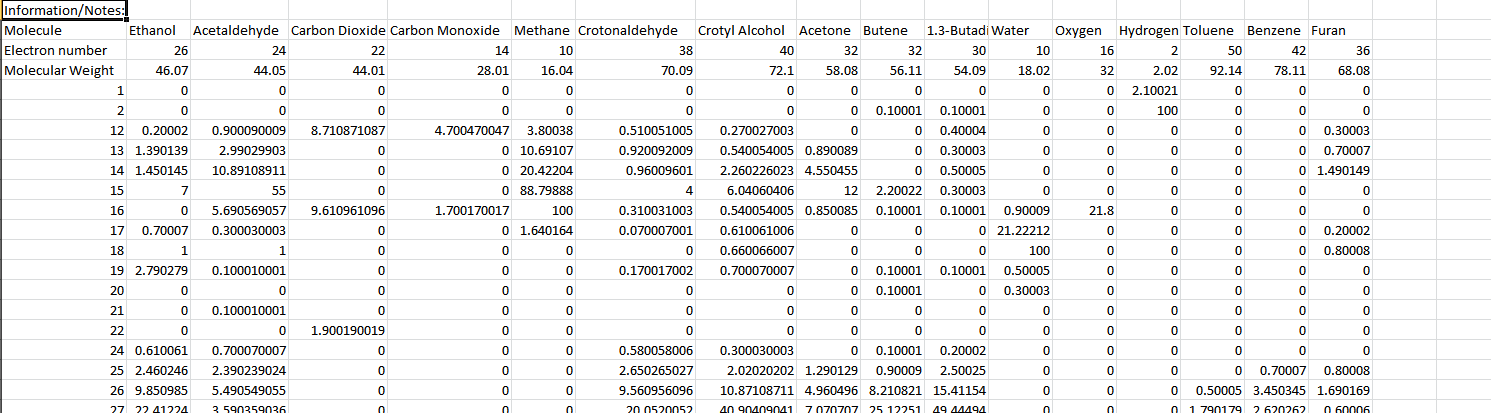
# Appendix 2: Data Generation (Module)

The Data Generation Program simulates raw data signals from concentration profiles and reference data. These simulated signals are graphed and exported to a csv file. By utilizing this program, users will have the ability to see the potential raw signals generated by various compositions before ever beginning experimentation.

The program requires two csv files as input: one file detailing concentration/signals over time and another csv file detailing relevant reference data. An illustration of both files is displayed below:

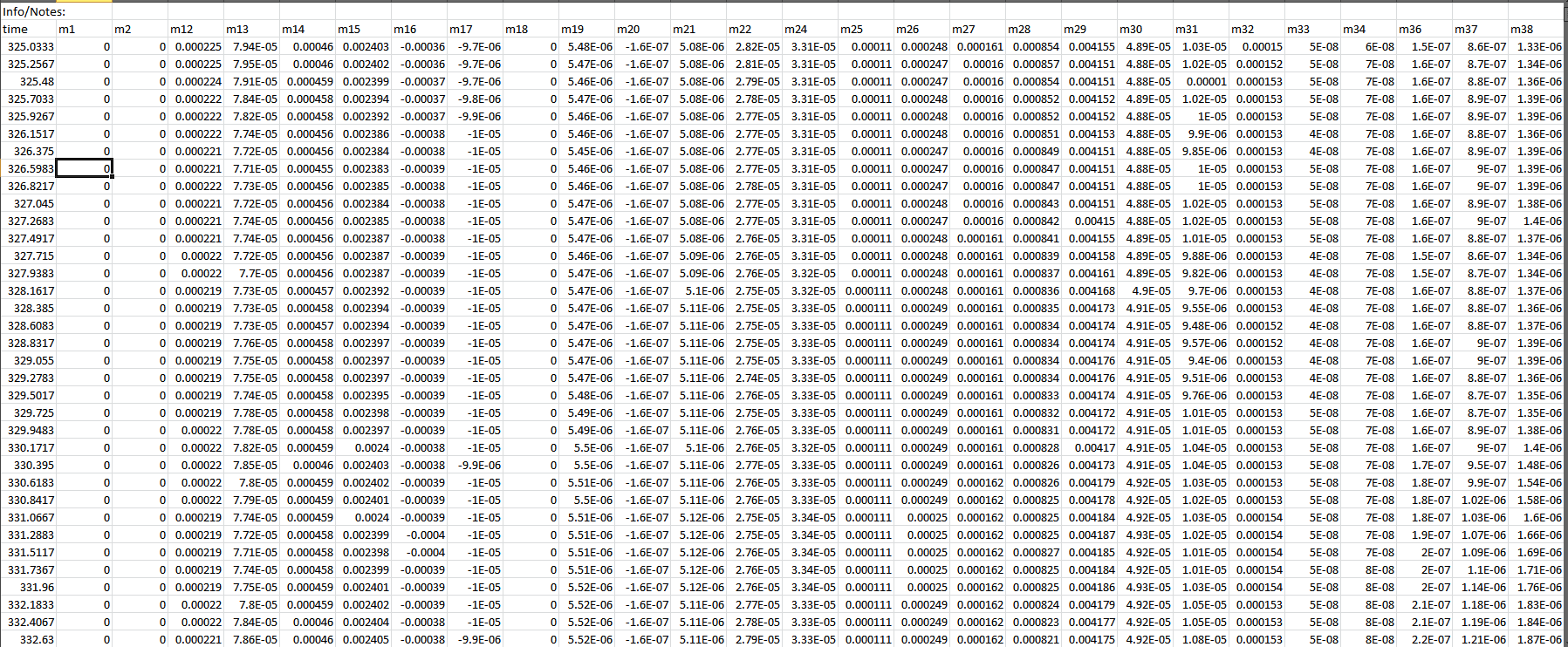


(Right): The format of the concentration/signal profile sheet.



(Right): The format of the reference sheet.

The output file of the Data Generation Program is shown below:



(Right) An example output of the Simulated Signals.

**Operation:**

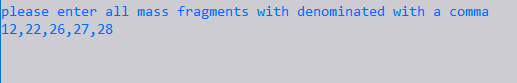
As with the JDXConverter program the Data Generation Program must be compiled and ran from the Anaconda command line. Make sure all files are located in the same folder before compiling.

Command to Compile and Run:

After compilation, the user will be prompted to either simulate all possible mass fragments or selected mass fragments. A user may only want to simulate the selected mass fragments that correlate to the mass fragments measured in an experiment.



If the user selects ‘yes’ then all mass fragments are simulated and exported to the simulated signals csv file. If the user selects ‘no’ then they are further prompted to enter the mass fragments he/she wishes to simulate.



After entering all desired mass fragments (delimited with a comma) the program simulates the signals for all desired mass fragments and concludes.