

Instructions for using the Mixing Polygon Simulation

updated 22/7/13

R users:

- Download the R script for the simulation. If your study is a two-isotope study (e.g. C^{13} and N^{15}), download *MixPolySim_2iso.r*; if your study is a three-isotope study (e.g. C^{13} , N^{15} , S^{34}), download *MixPolySim_3iso.r*.
- Prepare your Source data (these are the signatures of the prey items) as below, and save as a **.csv** file. The order should be C^{13} , followed by N^{15} , or the figure axes will be unconventional. Each source is one row, and has an average (dC_Av) and a standard deviation (dC_Sd) for each isotope. **NOTE:** each isotope must have a non-zero standard deviation for every source (a zero will cause a model error). Do not use row names, only column names.

The screenshot shows two Excel spreadsheets side-by-side. The left spreadsheet, titled 'Sources_example.csv', has columns A (d13C_Av), B (d13C_Sd), C (d15N_Av), and D (d15N_Sd). It contains 6 rows of data. A text box points to row 9, stating 'For two isotopes. This example has 5 sources.' The right spreadsheet, titled 'Sources_example_3i.csv', has columns A (d13C_Av), B (d13C_Sd), C (d15N_Av), D (d15N_Sd), E (d34S_Av), and F (d34S_Sd). It contains 10 rows of data. A text box points to row 12, stating 'For three isotopes. This example has 9 sources.'

- Prepare your mixture data (these are the signatures of the consumers) as below, and save as a **.csv** file. The column order of the isotopes must be the same as your source data (e.g. C^{13} first, then N^{15}). Do not use row names, only column names.

The screenshot shows two Excel spreadsheets side-by-side. The left spreadsheet, titled 'Mixture_example.csv', has columns A (d13C) and B (d15N). It contains 6 rows of data. A text box points to row 10, stating 'For two isotopes. This example has 6 consumers.' The right spreadsheet, titled 'Mixture_example_3i.csv', has columns A (d13C), B (d15N), and C (d34S). It contains 11 rows of data. A text box points to row 12, stating 'For three isotopes. This example has 10 consumers.'

- Prepare your trophic enrichment data (these are the trophic enrichment factors ‘TEF’, also known as fractionation factors) as below, and save as a **.csv** file. The column order must be the same as your source data. Each source must have its own TEF (i.e. the Sources and TEF files must have the same number of rows). Each isotope has an average TEF (dC_Av) and a standard deviation TEF (dC_Sd). **NOTE:** each isotope must have a non-zero standard deviation for every source (a zero will cause a model error). Do not use row names, only column names.

The image shows two Excel spreadsheets side-by-side. The left spreadsheet, titled 'TEF_example.csv', has 5 columns: A (d13C_Av), B (d13C_Sd), C (d15N_Av), D (d15N_Sd), and E. It contains 6 rows of data. A text box below it says 'For two isotopes. This example has 5 sources.' The right spreadsheet, titled 'TEF_example_3i.csv', has 7 columns: A (d13C_Av), B (d13C_Sd), C (d15N_Av), D (d15N_Sd), E (d34S_Av), F (d34S_Sd), and G. It contains 10 rows of data. A text box below it says 'For three isotopes. This example has 9 sources.'

	A	B	C	D	E
1	d13C_Av	d13C_Sd	d15N_Av	d15N_Sd	
2	1	0.3	3.4	0.6	
3	1	0.3	3.4	0.6	
4	1	0.3	3.4	0.6	
5	1	0.3	3.4	0.6	
6	1	0.3	3.4	0.6	

	A	B	C	D	E	F	G
1	d13C_Av	d13C_Sd	d15N_Av	d15N_Sd	d34S_Av	d34S_Sd	
2	1	0.2	3.4	0.5	1.5	0.2	
3	1	0.2	3.4	0.5	1.5	0.2	
4	1	0.2	3.4	0.5	1.5	0.2	
5	1	0.2	3.4	0.5	1.5	0.2	
6	1	0.2	3.4	0.5	1.5	0.2	
7	1	0.2	3.4	0.5	1.5	0.2	
8	1	0.2	3.4	0.5	1.5	0.2	
9	1	0.2	3.4	0.5	1.5	0.2	
10	1	0.2	3.4	0.5	1.5	0.2	

- You may now open and begin using the script, but first:
 - Specify the location of your working directory using ‘setwd’ (e.g. C:/R_scripts/MixPolySim). This is where you have saved the R script and your data files.
 - Install the necessary R packages using ‘install.packages()’. Installation only needs to be done once, and these lines of the script can then be hidden by placing ‘#’ at the beginning. The packages still need to be loaded each time R is opened using ‘library()’.
 - Specify the names of your 3 data files (e.g. “Sources_example.csv”), which must be saved in your working directory.
 - Input the number of iterations (“its”) to run; 1500 is usually sufficient, but assess this using Figure 1 – if the variance of the polygon area has not flattened out, increase this number.
 - For the two-isotope script, you must input the limits of the Mixing Region figure (Figure 3; i.e. “min_C” and “max_C” are the limits of the x-axis). If you do not know what limits to expect, run the script, and see if the axes in Figure 3 are suitable.
 - For the two-isotope script, input the resolution (“res”) of the Mixing Region figure (Figure 3). This is the density of points that is tested for point-in-polygon. A higher “res” means a more precise figure, but will greatly increase the run time of the simulation. For quick scoping simulations, use a low “res” (e.g. 100); this can be increased for high quality images (e.g. 250). **NOTE:** this does not influence the quality of the probability data, only the visualization of the mixing region. **NOTE:** this visualization is not possible for three-isotope studies.

- You may now step through the simulation. Pause after the loop (line 67 in *MixPolySim_2iso.r*, and line 52 in *MixPolySim_3iso.r*) to let the iterations run. A counter is displayed in the R window. Click on this window during simulation to see the current iteration.
- Once the iterations have finished, you can create the figures:
 - Figure 1 will show the variance of the mixing polygon area (or mixing polyhedron volume for 3 isotopes) against the number of iterations. Increase the iterations if the variance does not flatten out, or if vastly different results are recorded each time the simulation is run (a small difference is OK and expected).
 - Figure 2 will show the probabilities for each consumer. The consumers are numbered along the x-axis according to their row number in the data file. If there are a large number of consumers, it is probably wise to run them through the simulation in groups. The exact probabilities are also printed out in the R window using ‘print()’, and will be saved in the “Consumer_Probabilities.csv” file.
 - Figure 3 is created for two-isotope data, and shows the mixing region. The consumers (black dots) and average source signatures (white crosses) are placed on a heatmap. Probability contours indicate how often a mixing polygon enclosed an area. If a consumer lies outside the outermost contour (the 5% contour), then this consumer is unlikely to be explained by the data, and should be excluded from the subsequent mixing model (e.g. SIAR). The colourbar for Figure 3 is generated separately, and should be resized before saving. It should be presented with to the main Mixing Region figure.
 - Figure 4 is new to version 1.1 (2-isotopes only), and is simply a black & white (printer friendly) version of Figure 3.
- The script then saves the data into two files: “Consumer_Probabilites” records whether point-in-polygon was satisfied for each consumer for every iteration, and the final row is the proportion of the total iterations in which point-in-polygon was satisfied (the Probabilities). “Parameter_values” records the source and TEF data from each iteration of the simulation, plus the polygon area (or polyhedron volume), iteration count, and variance of the polygon area (or volume).
- You’re done! If none of your consumers are outside the 5% contour, then continue to the mixing model analysis (such as SIAR), and buy yourself a hot chocolate. If there are consumers inside fewer than 5% of mixing polygons, then you have a decision to make. See the paper by Smith *et al.* (2013) for a discussion of the options. You should probably still buy yourself a hot chocolate.

MATLAB users:

- To run this version of the simulation you will need the MATLAB software. This requires a licence. If you're here, you probably already know this. If you have access to MATLAB, I recommend using this script instead of the R script. The graphics are nicer (I think), and the 3-isotope script runs much quicker.
- Download the MATLAB script for the simulation. If your study is a two-isotope study (e.g. C¹³ and N¹⁵), download *MixPolySim_2iso.m*; if your study is a three-isotope study (e.g. C¹³, N¹⁵, S³⁴), download *MixPolySim_3iso.m*.
- Prepare your source data the same as for R users (see above) but save as an **.xls** file. NOTE: each isotope must have a non-zero standard deviation for every source (a zero will cause a model error). Do not use row names, only column names.
- Prepare your mixture data the same as for R users (see above) but save as an **.xls** file. The column order of the isotopes must be the same as your source data (e.g. C¹³ first, then N¹⁵). Do not use row names, only column names.
- Prepare your trophic enrichment data (TEF) the same as for R users (see above) but save as an **.xls** file. The column order must be the same as your source data. The Sources and TEF files must have the same number of rows. NOTE: each isotope must have a non-zero standard deviation for every source (a zero will cause a model error). Do not use row names, only column names.
- You may now open and begin using the script, but first:
 - Specify the names of your 3 data files (e.g. "Sources_example.xls"), which must be saved in the same directory as the MATLAB script.
 - Input the number of iterations ("its") to run; 1500 is usually sufficient, but assess this using Figure 1 – if the variance of the polygon area has not flattened out, increase this number.
 - For the two-isotope script, you must input the limits of the Mixing Region figure (Figure 3; i.e. "min_C" and "max_C" are the limits of the x-axis). If you do not know what limits to expect, run the script, and see if the axes in Figure 3 are suitable.
 - For the two-isotope script, input the resolution ("res") of the Mixing Region figure (Figure 3). This is the density of points that is tested for point-in-polygon. A higher "res" means a more precise figure, but will greatly increase the run time of the simulation. For quick scoping simulations, use a low "res" (e.g. 100); this can be increased for high quality images (e.g. 250). NOTE: this does not influence the quality of the probability data, only the visualization of the mixing region. NOTE: this visualization is not possible for three-isotope studies.

- You may now run the simulation. The iteration number is displayed in the MATLAB window.
- Once the iterations have finished, you may create the figures:
 - Figure 1 will show the variance of the mixing polygon area (or mixing polyhedron volume for 3 isotopes) against the number of iterations. Increase the iterations if the variance does not flatten out, or if vastly different results are recorded each time the simulation is run (a small difference is OK and expected).
 - Figure 2 will show the probabilities for each consumer. The consumers are numbered along the x-axis according to their row number in the data file. If there are a large number of consumers, it is probably wise to run them through the simulation in groups. The exact probabilities are also printed out in the MATLAB window using `Probabilities = sum(p)/its`, but this can be suppressed by adding `;` to the end of that line. The probabilities are also saved as the last row of the “Consumer_Probabilities.xls” file.
 - Figure 3 is created for two-isotope data, and shows the mixing region. The consumers (black dots) and average source signatures (white crosses) are placed on a heatmap. Probability contours indicate how often a mixing polygon enclosed an area. If a consumer lies outside the outermost contour (the 5% contour), then this consumer is unlikely to be explained by the data, and should be excluded from the subsequent mixing model (e.g. SIAR).
 - Figure 4 is new to version 1.1 (2-isotopes only), and is simply a black & white (printer friendly) version of Figure 3.
- The script saves the data into two files: “Consumer_Probabilites” records whether point-in-polygon was satisfied for each consumer for every iteration, and the final row is the proportion of the total iterations in which point-in-polygon was satisfied (the Probabilities). “Parameter_values” records the source and TEF data from each iteration of the simulation, plus the polygon area (or polyhedron volume), iteration count, and variance of the polygon area (or volume). It is good practice to rename these files, or to delete the previous version, each time the script is run.
- You’re done! If none of your consumers are outside the 5% contour, then continue to the mixing model analysis (such as SIAR), and celebrate with a rest in a hammock. If there are consumers inside fewer than 5% of mixing polygons, then you have a decision to make. See the paper by Smith *et al.* (2013) for a discussion of the options. You should probably still have a rest in a hammock.