INST-FLUX documentation

INST-FLUX is a software package written in MATLAB which implements the Elementary Metabolites Framework, to perform 13^{13}C INST-MFA, as described by [2, 1].

Download and Setup

All the required files are available at: `https://github.com/ApostolisZ/INST-FLUX`

Download and unpack INST-FLUX. Begin a new MATLAB® session, change working directory to the INST-FLUX directory and run the following command:

≫ setup

To test if the installation was successful, a test folder is included and

Create a new dataset

INST-FLUX requires 4 input .csv files. The names of the four files are "network.csv", "labeling.csv", "element_comp.csv", "pool.csv", "substrate.csv". These files need to be located in a subfolder, where the MATLAB®functions are located (*master_dir/analysis_name/*).

network.csv includes the information of the metabolic network and the atom transitions. It also includes the values for any measured metabolic fluxes. If there are no flux measurements, a flux value needs to be set and the other flues will be scaled compared to that. A condensation reaction of a 2-carbon metabolite (A) and an 1-carbon metabolite (B) to a 3-carbon metabolite (C) with flux 100 units should be written as shown in figure 2b:

labeling.csv includes the experimentally determined MIDs. For all labeled metabolites the time points need to be the same. The data input can be the raw experimental data (e.g. area under the peak or peak heights coming from a chromatogram). For

two time-series experiments measuring metabolites A and B, data should be written in "labeling.csv" as described in figure 2a

element_comp.csv contains information regarding the metabolites of the network. The metabolite names are entered in the first column, and the chemical composition is entered in the second column. Fig. 1a shows the way that INST-FLUX recognizes the chemical composition of the metabolites.

pool.csv includes the measured concentrations of metabolites. Ideally, the concentrations of labeling metabolites are included in this file, and INST-FLUX will calculate the dilution factors of these based on the concentrations. Fig. 1b show how the correct syntax to describe pool sizes. These values need to be in the same units (e.g. $nmol/gDW$) and the values of the output concentrations will be in units of the input.

substrate.csv includes information about the network substrate. In this file, the fraction of the labeled states of the substrate are described in this file. Figure **reffig** shows how the user needs to describe the 3-carbon substrate A if 50% of it has the first and third carbons labeled, while the rest is naturally labeled.

Finding the optimal fluxes

After the network is defined and all the required data are input in the respective files and in the correct format, the user needs to type the following command:

≫ INST_emu(analysis_name,nloop,init_fracs_zero,seed)

This command will calculate nloop local minima using algorithm **??**, and will generate a multiple (nloop) result Matlab® files in the folder
master_dir/results/analysis_name.
These results will include information about the local minima calculated, and will be MATLAB®structures. The fields are explained in table 1.

	A	B
1	Metabolite	Chemical Formula
2	A	C2H2O2
3	B	C2H2O5P

(a) Input data describing the elemental composition of Metabolites A and B

	A	B
1	B	1
2	C	3
3		
4		

(b) Input data describing the pool size measurements of Metabolites A and B

	A	B
1	Isotopomer	Fraction
2	A_10	0.25
3	A_11	0.25

(c) Input data describing the substrate of the network. In this case, the network substrate is only the two-carbon metabolite A. 25% of A is uniformly labeled, 25% is labeled only on the first carbon and the rest (50% in this case) is treated as unlabel substrate

To run the computation in a parallel manner, use

≫ INST_emu(analysis_name,nloop,init_fracs_zero,seed)

This command implements parallelization using the Matlab® parfor environment.

Processing the results

To process the results, the used needs to call the function "process_results", by typing in the MATLAB®command window:

≫ process_results(analysis_name,figures,write,calc_conf_int)

The arguments `figures, write, estimate_conf`, define whether the function will generate figures, a results file in `master_dir/analysis_name`, and calculates the confidence intervals are discussed in section **??**. The default values of the optional arguments are `true, false, false`.

	A	B	C	D	E	F	G
1	Ion	EMU	Time (mir	0	0.1	1	2
2	C	C_1234	M0	9579	9560	9520	9464
3			M1	414	423	442	468
4			M2	7	17	37	67
5			M3	0	0	1	1
6			M4	0	0	0	0

(a) Input of a time-series experiment, measuring the labeling of metabolite C for the starting point of the experiment and 0.1, 1 and 2 minutes later.

	A	B
1	A (ab) -> B (ab)	
2	B (ab) + E (cd) -> C (abcd)	
3	C (abcd) <-> D (bcd) + F (a)	
4	D (abc) -> E (ab) + H (c)	
5	D (abc) <-> B (ba) + J (c)	
6	F (a) -> G (a)	
7	-> A	1
8	G ->	
9	H ->	
10	J (a) ->	

(b) Contents of the `network.csv` file. The reaction, including the atom transitions, are written on the first column. In the second column flux measurements are placed.

Table 1: fields of result structures generated by the optimization routine

Field	Values
emumodel	MATLAB structure containing information about metabolic reactions and EMU reactions
cdata	MATLAB structure containing information about the measured pool sizes of metabolites
nz_isos	MATLAB structure containing information about the network substrate
x	values of decision variables at the local minimum
fval	Objective function at the local minimum
runtime	time needed to calculate the local minimum

Calculating the confidence intervals In order to calculate the confidence intervals, multiple optimization routines are called, with randomly perturbed experimental measurements.

≫ calc_ci_samp(analysis_name, result_name, nloop)

Does exactly that, `nloop` times. The string `result_name` contains the name of the result we want to calculate the confidence intervals around. This is usually the best result calculated from

≫ INST_emu(analysis_name,nloop,init_fracs_zero,seed)

This process need to be done prior to running

≫ process_results(analysis_name,figures,write,calc_conf_int)

with `calc_conf_int` set to true.

References

[1] Lara J. Jazmin et al. "Isotopically Nonstationary MFA (INST-MFA) of Autotrophic Metabolism". In: Humana Press, Totowa, NJ, 2014, pp. 181–210. DOI: 10.1007/978-1-62703-688-7_12. URL: http://link.springer.com/10.1007/978-1-62703-688-7%7B%5C_%7D12.

[2] Jamey D Young et al. "An Elementary Metabolite Unit (EMU) Based Method of Isotopically Nonstationary Flux Analysis". In: *Biotechnology and Bioengineering* 99 (3 2007). DOI: 10.1002/bit.21632. URL: https://lamp.vanderbilt.edu/younglab/pdf/young08a.pdf.