

This README file was generated on July 2, 2025 by Anastasia Bankowski. It introduced and explains the Python codes that were used in order to generate the parameter estimations and figures in the scientific paper entitled

Parameter Optimization for a Neurotransmission Recovery Model

by Ariane Ernst, Anastasia Bankowski, Meida Jusyte, Toluwani Okunola, Tino Petrov, Alexander M. Walter and Stefanie Winkelmann.

1 Multistart Optimization

1.1 `opt_DE_multistart.py`

This code runs the parameter optimization using amici and the differential evolution algorithm in a multistart manner. If it is the first time running the codes, please first run the sbml importer at the top of this file to import the `recovery_model_60_peaks_qssa_rates` python module (lines 54 to 64). For the settings of this optimization, check `opt_settings_multistart.py`. After the optimization, the results of all starts are saved. To extract the best parameters, please use `bestparams_meanvalue.py` before continuing with the plots in `plotting.py`.

1.2 `opt_settings_multistart.py`

These are the settings for the multistart optimization in `opt_DE_multistart.py`. Change here the type of the parameter estimation (I, II, III), the animal whose current is being fitted, the parameter bounds and the path where to save the results. Moreover, the settings for the differential evolution algorithm is located here.

1.3 `bestparams_meanvalue.py`

Evaluate the results of the `opt_DE_multistart.py` run here and find the best parameters of all runs of the multistart. This is used for the plots and you can calculate the mean values used for `baseline.py`.

2 Fixed Parameter Optimization

2.1 `opt_DE_fixed_params.py`

This code runs the parameter optimization using amici and the differential evolution algorithm in a multistart manner but where one parameter is fixed. If it is the first time running the codes, please first run the sbml importer at the top of this file to import the `recovery_model_60_peaks_qssa_rates` python module (lines 54 to 64). For the settings of this optimization, check `opt_settings_fixed_params.py`. After the optimization, the results of all starts are saved. To extract the best

parameters, please use `fixedparams_bestparams.py` before continuing with the plots in `objective_fct_profiles.py`.

2.2 `opt_settings_multistart.py`

These are the settings for the multistart optimization in `opt_DE_fixed_params.py`. The type of parameter estimation is II. Change here the parameter fixed, the range of the values, the animal whose current is being fitted, the parameter bounds and the path where to save the results. Moreover, the settings for the differential evolution algorithm is located here.

2.3 `fixedparams_bestparams.py`

With this code you can evaluate the results of `opt_DE_fixed_params.py`. You can find the best parameters for each fixed parameter of all runs of the multistart. This is used for the plots.

3 Plottings

3.1 `plotting.py`

This plots the best fit of all starts of the multistart optimization in `opt_DE_multistart.py` as well as the variability of the parameters over these starts. It produces Figures 3, 4, 5, 7, A2, A4, A6.

3.2 `optimal_parameters_compared_plot.py`

This plots the loss and the optimal parameters of all animals of parameter estimation III as scatterplot. It produces Figure 8.

3.3 `waterfallplots.py`

This plot the loss values of all starts of the multistart optimization in `opt_DE_multistart.py` and produces Figures A1, A3, A5.

3.4 `baseline.py`

Plots the baseline functions as estimated from Parameter Estimation II, as well as the mean of them and produces Figure 6.

3.5 `objective_fct_profiles.py`

Plots the objective function profiles of the fixed parameter optimization runs. Produces Figure 9 and 10.