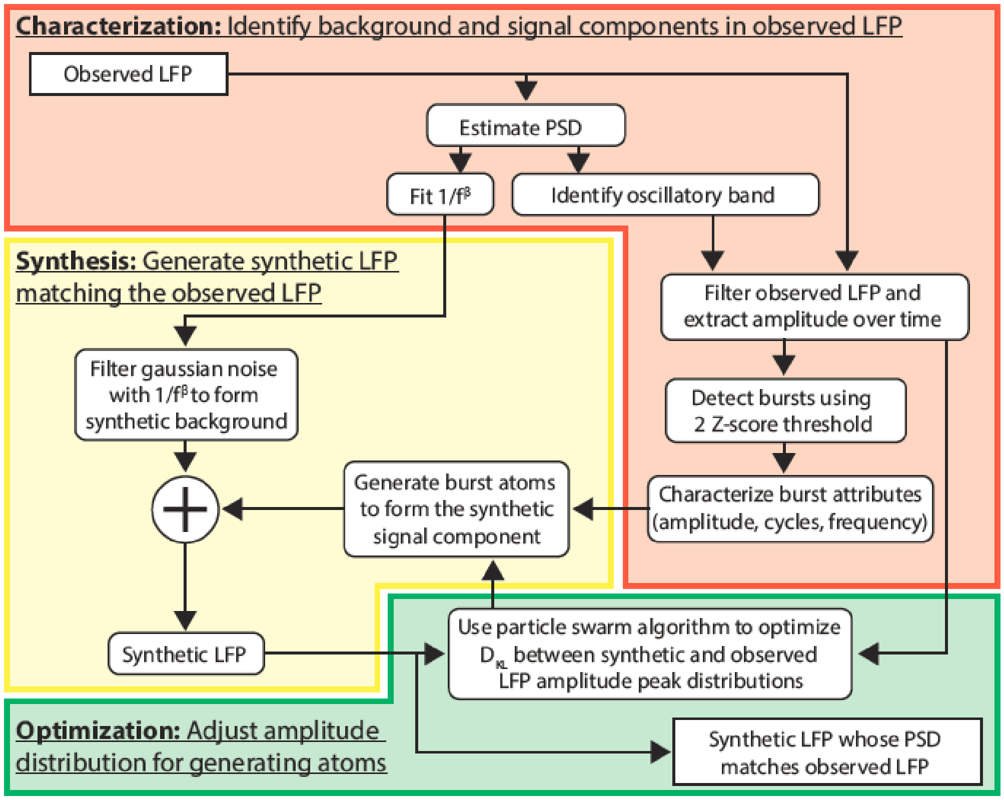
User's Manual

Contents

1 Introduction

Schematic of analysis and synthesis algorithm



The work flow has 3 major steps: 1. Characterization, 2. Synthesis, 3. Analysis, as follows:

1. Select local field potential (LFP) recording as input. Run the algorithm to fit power spectral density (PSD), decompose the PSD into signal and background components. Then characterize bursts properties, including statistics of amplitude peaks, number of cycles and burst frequency. Finally, save the characterization results.

2. Load results from the previous step. Specify parameters for synthesizing the background component and oscillatory signal component. Then perform an optimization for amplitude peak distribution parameters of burst atoms to be generated in the signal component. This step is essential for the synthetic data to reproduce the amplitude distribution of the observed data. There are three candidate types of parametric distribution for the burst atom amplitude: gamma, lognormal and exponential. Gamma distribution works well in general. The goodness of fit is indicated by Kullback–Leibler divergence (D_{KL}) or Jensen–Shannon divergence (D_{JS}). Lower divergence value indicates better fit. Then synthetic LFP is generated with optimized parameters and the resulting properties including PSD and burst statistics are visualized as in step 1. Finally, save the synthesis parameters and data.

3. Analyze the synthetic data generated from the previous step for evaluating detection. Define the ground truth with lower and upper bounds in the synthetic signal trace and evaluate detection performance on the composite trace using a receiver operating characteristic (ROC) curve. The relation between detection threshold and true/false positive rates will be shown.

2 Graphical User Interface Instructions

2.1 Tutorial

This tutorial demonstrates the usage of a graphical user interface of the toolbox using an example LFP data. The data is the example case for gamma oscillation illustrated in Fig.1, 2 and 3 of the associated paper *Approaches to characterizing oscillatory burst detection algorithms for electrophysiological recordings*.

2.1.1 Install and run the GUI

Make sure you have *Signal Processing Toolbox* and *Statistics and Machine Learning Toolbox* installed in your MATLAB. Download our MATLAB toolbox from the Github repository `https://github.com/chenziao/Matlab_Toolbox-Oscillatory_Burst_Detector.git`. Run MATLAB and install the App using the MATLAB App installer *"Oscillatory Burst Detector.mlappinstall"*. Run the App after installation completes and it will open a GUI.

2.1.2 Load example data

An example LFP data *"LFP_BLA_gamma.mat"* was provided in the folder *"example data"*.

First, load the data as a *MAT-file* into MATLAB workspace. Then, run the following line

```
LFP_seg = cellfun(@(x) scale*double(x),LFP_seg,'UniformOutput',false);
```

to scale it from digital value to microvolts (µV). The sampling frequency was 1000 Hz, also provided as the variable *fs*. Gamma oscillation was present in this data.

2.1.3 Step 1: Characterization

In panel *Select Data*, select the variable *LFP_seg* in *LFP Data*. Enter *fs* in *Sampling Rate*. In panel *Fit PSD*, use default settings, *i.e.*, Gamma oscillation frequency range 30 - 80 Hz for *Signal Frequency Band*, *Autofit* for *Frequency Range for Fitting* and *Default* for *Decibel Threshold*. Then click on *Fit* button. The figure for the PSD fit will display on the top right. And messages will print in *Results* at the bottom as you proceed.

After fitting the PSD successfully, use default settings in panel *Characterize Bursts and Fit Probability Distribution*, *i.e.*, 30 for number of histogram bins and microvolt for the unit of the LFP amplitude. Then click on *Run*. The figures for the histograms of the burst attributes will display on the bottom right. Select the tabs on the right of the figure to switch between different attributes (where AP, BF, CN stands for amplitude peak, burst frequency, cycle number, respectively) and select the tabs at the bottom of the figure to switch between linear and log scale for the histograms.

Now, you have got the characterization results and need to save it for later steps. Use the *Project Directory* panel at the top of the GUI to select or create a folder where all the results will be saved for this example LFP data. Then, enter the name of the data *e.g.*, *LFP_BLA_gamma*, in the text box in the bottom panel *Save Characterization Data*. Finally, click on *Save* button and the results will be saved to a *MAT-file* named *LFP_BLA_gamma_Charac.mat* under the selected folder. Check the box *Also Save Burst Statistics* if you want to also save the attribute values of each burst characterized from the data before hitting the *Save* button and you will get another *MAT-file* named *LFP_BLA_gamma_BurstStats.mat* in the same folder.

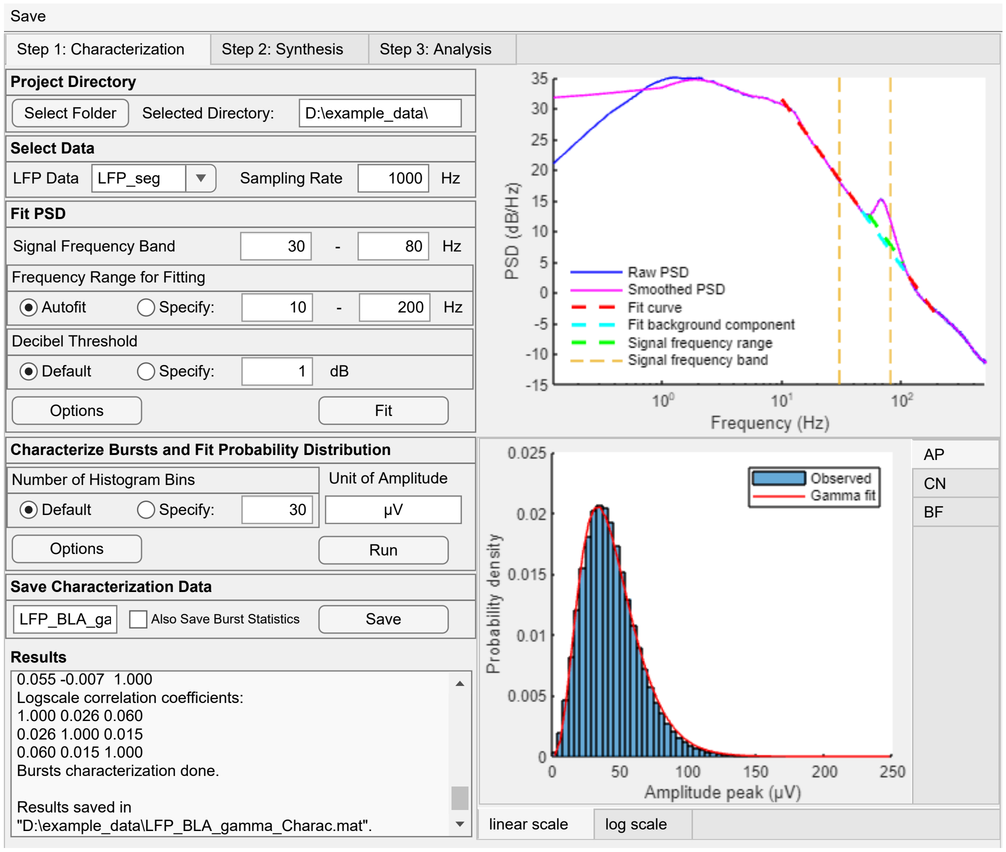
2.1.4 Step 2: Synthesis

Select the tab at the top of the GUI to switch to step 2. In the top panel *Characterization Data*, the characterization result file saved in step 1 should be present automatically. Or you can manually select the file using *Select File* button.

In panel *Synthesis Settings*, select the default *Sampling Rate* for the synthetic signal which is the same as the LFP data. Enter any random seed number in *Rng Seed* *e.g.*, 0. Then, click on *Load* button. This procedure loads the results from step 1 and creates settings for generating the synthetic data.

Next step is to optimize the amplitude distribution in the synthetic data using particle swarm optimization (PSO). In panel *Optimize Amplitude Distribution Parameters*, select *Default* for *Synthetic Data Length* *i.e.*,

Step 1

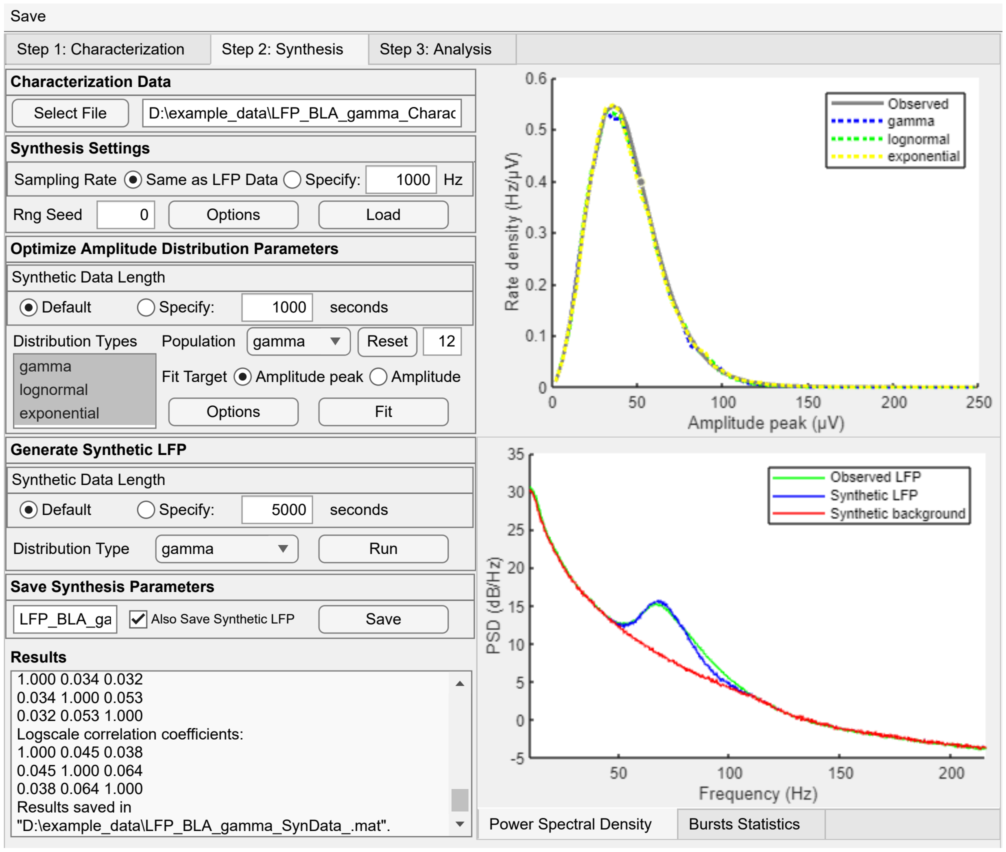


1000 seconds. Then, you can select the parametric distribution types used for the amplitude of the burst atoms to be generated. The default type is *gamma*. You can select multiple types using *Ctrl+Click* in the menu. You can also change the population size of the PSO algorithm by first selecting the distribution type in the drop-down menu and then entering the number on the right. Note that a larger population allows larger parts of the search space to be covered per iteration and leads to fewer iterations to converge, but increases the run time per iteration. Finally, there are two approaches for optimizing the amplitude distribution, with one considering only the peak of the amplitude while the other considering amplitude at all time points. Select *Amplitude peak* for *Fit Target* (the former approach) and click on *Fit* button to run the optimization. It may take several minutes to finish. After the optimization finishes, the amplitude distribution plot of the synthetic data and the observed data will display on the right together.

Next step is optional. It generates the synthetic LFP and shows its PSD and burst statistics for comparison with the observed data. You can choose the *Synthetic Data Length* in panel *Generate Synthetic LFP*. A longer synthetic data will improve the accuracy of the analysis in step 3. You can also choose the *Distribution Type* for the burst atom amplitude if you have run the optimization with multiple distribution types. By default, it is the type that achieves the best match among all selected types after the optimization. Click on *Run* button to generate and show the synthetic LFP.

Finally, you need to save the synthesis parameters for step 3. It is optional to enter a case name in case there are multiple sets of synthesis parameters you want to keep. Click on *Save* button to save a *MAT-file* (named *LFP_BLA_gamma_SynParam.mat* if case name is not specified) and it must be saved in the same directory of the characterization result file. You can check the box *Also Save Synthetic LFP* before hitting the *Save* button if the synthetic LFP has been generated previously. You will get a *MAT-file* named *LFP_BLA_gamma_SynData.mat* instead which could have large file size but saves time to regenerate the data in step 3.

Step 2



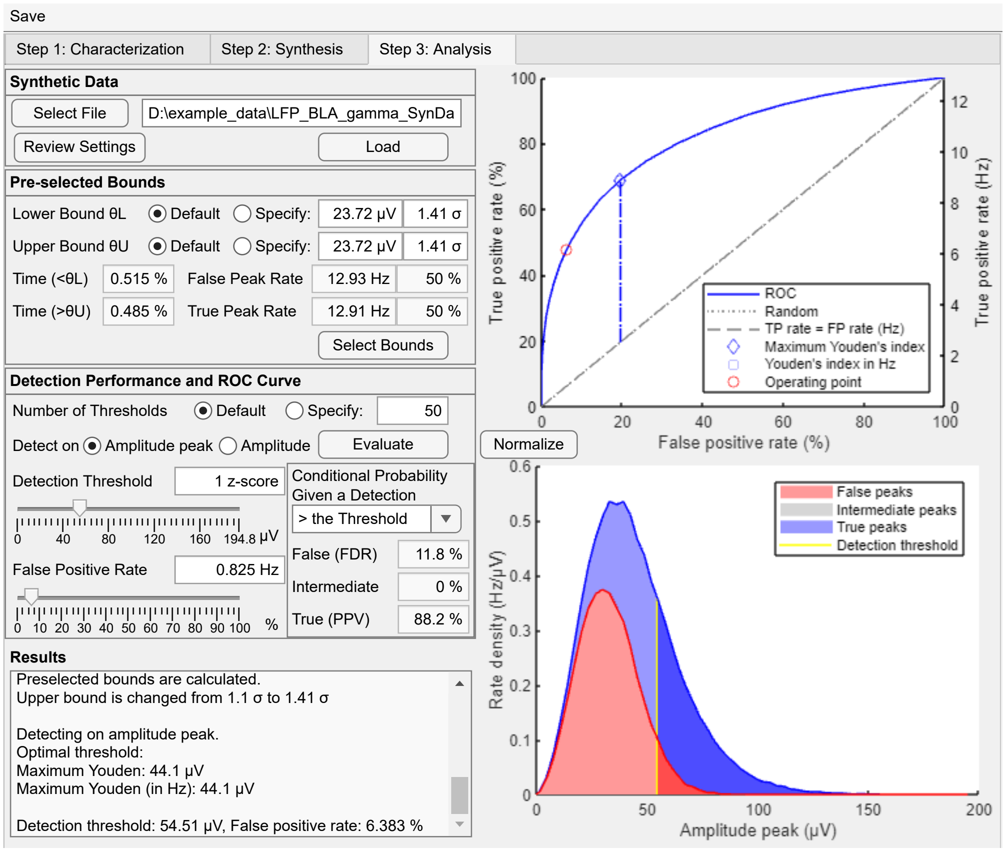
2.1.5 Step 3: Analysis

Select the tab at the top of the GUI to switch to step 3. In the top panel *Synthetic Data*, the file saved in step 2 should be present automatically. Or you can manually select the file using *Select File* button. You can review the optimization results and the synthesis parameters by clicking on *Review Settings* button and the GUI will switch back to step 2. You can change the settings or rerun optimization if needed and click on *Run* button in panel *Generate Synthetic LFP* to make the changes effective for step 3. Then, click on *Load* button in step 3 to load the synthetic data.

In panel *Pre-selected Bounds*, the lower and upper amplitude bounds in the synthetic signal trace defines salient oscillatory events which is the ground truth for evaluating the detection. The bounds can be specified using either absolute amplitude value or a multiple of σ which is the square root of the average power in the noise trace. You can choose the *Default* bounds that are $\theta_L = \sqrt{2}\sigma$ and θ_U based on the dB threshold and the PSD (see Methods). Once the bounds are determined, click on *Select Bounds* button and the resulting ground truth information will be shown in terms of the time duration proportion, and the amplitude peak rate and its proportion for each category.

In panel *Detection Performance and ROC Curve*, use the default settings, *i.e.*, 50 for *Number of Thresholds* and *Amplitude peak* as the target for detection. Then click on *Evaluate* button. The ROC curve will display on the top right and the distribution of the ground truth will display on the bottom right. You can change the *Detection Threshold* by entering a Z-score of the amplitude or using the slider with absolute amplitude scale. The resulting *False Positive Rate* will display in the text box and in the slider below. The corresponding operating point will show on the ROC curve and the threshold will also be shown as a vertical line in the distribution plot. The panel *Conditional Probability Given a Detection* shows the probability of each true outcome given a detection with amplitude greater than the threshold. For example, using 1 Z-score detection threshold, if an amplitude peak above it is detected, it has 11.8% chance to be false positive and 88.2%

Step 3



chance to be true positive. They correspond to the areas under the distribution curves to the right of the detection threshold. You can select $=$ *the Threshold* in the drop-down to show the conditional probabilities given a detection with amplitude exactly equal to the detection threshold. They correspond to the proportion of each segment length along the vertical line of the detection threshold in the distribution plot. You can also set a *False Positive Rate* by entering the text box or moving the slider. The corresponding *Detection Threshold* will display above it and the figures will also update. *Maximum Youden's Index* is a method that determines an optimal detection threshold, which maximizes the difference between the true positive rate and the false positive rate. The corresponding operating point is marked on the ROC curve and the optimal detection threshold is printed in *Results* at the bottom.

Note that the same filter should be used when applying optimal detection threshold for actual data. Butterworth filter is used with parameters determined in step 1. The filter order is 6 by default (see 2.2.1). The cutoff frequency is the signal frequency range characterized in step 1. The filtering uses the second-order sections format (biquad filter, *i.e.*, the *sos* form in MATLAB) with zero-phase forward and backward filtering (*i.e.*, *filtfilt* in MATLAB).

2.1.6 Save Results

Finally, you can save all the figures using the option *Save Figures* in the top menu. You can also save all the output messages to a log file using the option *Save Output Log* in the menu.

Next time when you restart the GUI, you can start with any of the three steps as long as you have the files needed for that step.

2.2 GUI Components

2.2.1 Step 1: Characterization

Project Directory:

- Select the project directory where the result files will be saved. Use *Select Folder* button to select or directly type the path in the text box.

Select Data:

- *LFP Data* specifies the input LFP data. Select a variable from your MATLAB workspace in the drop-down menu or enter the variable name or an expression. The variable needs to be a ***vector*** of the input LFP data or a ***cell array*** with each element a vector of an LFP segment. The latter format is useful for segmentized data.

- *Sampling Rate* is the recording sampling rate in Hz. You can enter either a number or a variable name.

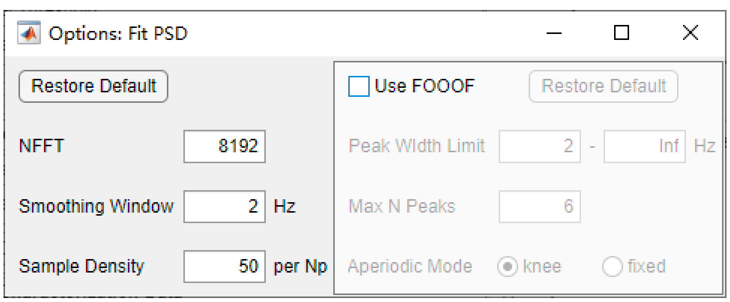
Fit PSD:

- *Signal Frequency Band* is the frequency range of the oscillation you want to detect. It tells the algorithm where to find the oscillatory signal component in the PSD. It does not need to be accurate.

- *Frequency Range for Fitting* is the frequency range where the algorithm fits an $1/f^{\beta}$ curve to the PSD. You can adjust the range to find a best fit for the $1/f^{\beta}$ background component. Or you can use the *Autofit* option to let the algorithm determine the range automatically which is suggested.

- *Decibel Threshold* is a threshold used to find significant oscillatory signal power above the background component in the signal frequency band. The signal frequency range is determined as the range where the power is certain decibels higher than the background component. Try to decrease the threshold if the power of the signal component is too small to be found.

- *Options*



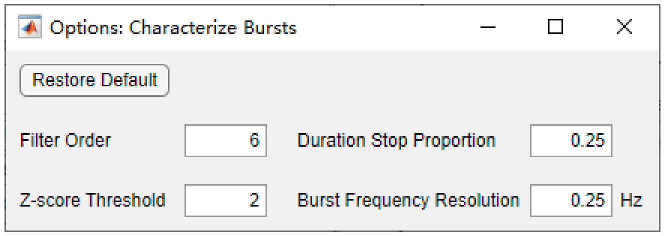
  - *NFFT* is the time window size for estimating the PSD.
  - *Smoothing Window* is the frequency window size (Hz) for smoothing the PSD.
  - *Sample Density* is the density of sample frequency points in natural logarithm per neper (Np).
  - Check the box *Use FOOOF* to use FOOOF algorithm which is an alternative method for fitting the PSD. Using FOOOF will enable the parameter settings for it in this panel. *Sample Density* will not be used. ***Note:*** You need to install the package first. See the documentation of FOOOF for the parameters.

- Click on *Fit* button to run the fit. Fitting results will display in the plot on the top right.

Characterize Bursts and Fit Probability Distribution:

- *Number of Histogram Bins* determines the number of bins for the distribution histogram of the burst attributes. The histogram for amplitude peak uses twice this number for better precision during the optimization in step 2. Adjust this number according to your data size.

- Enter the *Unit of Amplitude* of the LFP data which will be used in all displays.
- *Options*



  - *Filter Order* is the order of the Butterworth filter used to band-pass filter the LFP in the signal frequency range.
  - *Z-score Threshold* is the amplitude Z-score threshold used to detect salient bursts for characterization.
  - *Duration Stop Proportion* is the proportion of amplitude peak value that determines the burst duration. The burst duration stops on both sides of the amplitude peak when amplitude drops below this value.
  - *Burst Frequency Resolution* is the resolution (in Hz) for estimating the burst frequency, which determines the NFFT for the estimation.

- Click on *Run* button to run characterization. Histograms of burst attributes will display in the plot on the bottom right. Switch the tabs to view different plots for the attributes AP, BF, CN, which stands for amplitude peak, burst frequency, cycle number, respectively. The tabs at the bottom switches between linear and log scale for the histograms.

Save Characterization Data:

- Enter a name for the characterization results and click on *Save* button. A dialog box will open and save the result file under the project directory.
- Check the box *Also Save Burst Statistics* if you want to also save the attribute values of each burst characterized from the data before hitting the *Save* button. Another file will be saved under the same directory.

Results:

- A text box where information/message are shown during runtime.

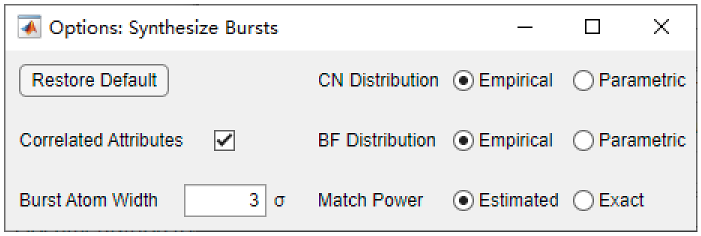
2.2.2 Step 2: Synthesis

Characterization Data:

- Select the file you saved from step 1 using the *Select File* button or enter the file path in the text box

Synthesis Settings:

- *Sampling Rate* specifies the sampling rate of the synthetic data. The default is the same as the observed LFP data which leads to best reproduction of it. However, you can still specify a different one. ***Note:*** Large sampling rate will significantly slow down the generation of synthetic data, especially during the optimization.
- *Rng Seed* specifies the seed for random number generator.
- *Options*



- *Correlated Attributes* is whether to generate correlated burst attributes (AP, CN, BF) for synthetic signal using the correlation characterized from the observed data.
- *Burst Atom Width* is multiple of σ of the Gaussian envelope, which is the length of each burst atom to be generated.
- *CN Distribution* determines whether the cycle number to be generated follows an empirical distribution or a parametric distribution characterized from the observed data.
- *BF Distribution* determines whether the burst frequency to be generated follows an empirical distribution or a parametric distribution characterized from the observed data.
- *Match Power* is the method that determines the number of burst atoms to be generated to match the burst power in the observed PSD. *Estimated* method determines the number by estimating the expectation of total power prior to generating the burst atoms. *Exact* method adds the generated burst atom until the total power matches. *Estimated* method is more efficient but leads to greater random error.

• When parameters are set, click on *Load* button before moving to next step.

Optimize Amplitude Distribution Parameters:

Particle swarm optimization is used to match the amplitude distribution to the observed data. It creates a population of models with different parameters and they will be attracted toward the point in the parameter space with minimal cost found after each iteration. The termination criteria is when either the number of stall iterations or the maximal number of iterations is reached.

- *Synthetic Data Length* is the duration of signal to be generated for optimization. ***Note:*** Shorter duration leads to shorter run time but results in higher variance in the empirical distribution and higher random error in the cost value for the optimization.

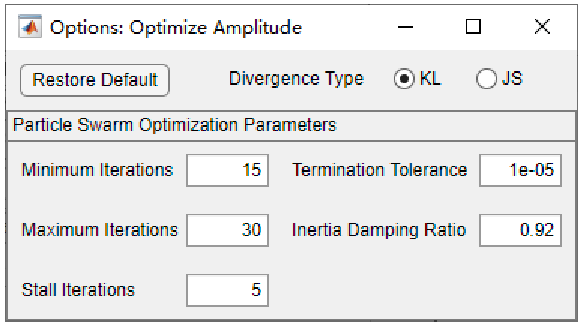
- *Distribution Types* lists the candidate parametric distribution types for the burst atom amplitude peak that are used to generate the synthetic signal component. You can select multiple of them using *Ctrl/Shift+Click*. An individual optimization will run for each distribution type selected.

- *Population* specifies the particle swarm size for each candidate distribution. Select the distribution type in the drop-down menu and enter the number in the text box. Clicking on *Reset* button will restore default number. ***Note:*** Larger population allows larger parts of the search space to be covered per iteration and leads to fewer iterations to converge, but increases the run time per iteration. Large population helps searching the parameter space but takes longer to simulate.

- *Fit Target* determines the target amplitude distribution for the optimization. *Amplitude peak* considers the distribution of only the peak of the amplitude and *Amplitude* considers the distribution of amplitude at all time points.

- *Options*

  - *Divergence Type* is the cost function for the optimization. It can be either KL divergence or JS divergence which indicates the similarity between the amplitude distributions of the observed data and synthetic data.
  - *Minimum Iterations* is the minimum number of iterations the optimization will run.



- *Maximum Iterations* is the maximum number of iterations the optimization will run when the number of stall iterations is never reached.
- *Stall Iterations* is the number of consecutive stall iterations for the termination criteria. Stall iterations are iterations with no better cost value found.
- *Termination Tolerance* is a tolerance value that determines whether an iteration achieves better cost. When the improvement of the lowest cost value found in an iteration compared to the lowest cost value ever found in all previous iterations, is lower than the tolerance value, the iteration is considered as a stall iteration.
- *Inertia Damping Ratio* is a factor used to determine the impact of each particle's previous position with respect to its new position in the next iteration.

- Click on *Fit* button to run. ***Note:*** The optimization may take several minutes to finish. The resulting amplitude distributions of the synthetic data generated by optimized parameters of selected candidate distribution types will display in the plot on the top right, together with the distribution of the observed data.

Generate Synthetic LFP:

- *Synthetic Data Length* is the duration of synthetic signal to be generated for latter analysis in step 3. It can be longer than that in the previous step since it will be generated only once. Longer synthetic data leads to lower variance for later analysis.
- *Distribution Type* is the amplitude distribution type selected from the ones that has been optimized and will be used to generate the synthetic data.
- Click on *Run* button to generate. The characterization results of the synthetic data similar to that of the observed data in step 1 will display in the plot on the bottom right.

Save Synthesis Parameters:

- Save the synthesis parameters obtained into a file. You can do it right after the optimization step. You can also check the box *Also Save Synthetic LFP* after you run the *Generate Synthetic LFP* step. But the file size could be large. Saving the synthetic LFP is not required for step 3 because the synthetic data can be reproduced by the synthesis parameters saved. Data generated by one set of parameters will be saved in one individual file. You can rerun *Generate Synthetic LFP* with another parameters and save to another file.
- Enter a case name in the text box in case there are multiple sets of synthesis parameters or generated data to be save. The case name will be added as a suffix to the file name.
- Click on *Save* button, a dialog box will show up and save the file in the project directory. ***Note:*** The synthesis parameter file has to be in the same directory as the characterization result file in order to work properly.

Results:

- A text box where information/message are shown during runtime.

2.2.3 Step 3: Analysis

Synthetic Data:

- Select the synthesis parameter file you saved from step 2 using the *Select File* button or enter the file path in the text box.
- Click on *Review Settings* to review the optimization results and synthesis parameters. The GUI will switch back to step 2. You can change the settings or rerun optimization if needed. You need to click on *Run* button in panel *Generate Synthetic LFP* to make the changes effective for step 3.
- Once you confirm with the settings, click on *Load* button to load the synthetic data. If the synthetic data was not saved in the synthesis parameter file, it will take a while to generate the data. The synthetic LFP and its two components, background and signal, are band-pass filtered with the signal frequency range.

Pre-selected Bounds:

- *Lower Bound* and *Upper Bound* specify the amplitude bounds θ_L and θ_U in the signal trace that are used to define the ground truth with three categories in the synthetic LFP, ***true***, ***false*** bursts, and an ***intermediate*** category. You can select default to use automatically determined bounds. Or you can specify the bounds using absolute amplitude or a multiple of the standard deviation of the background trace (σ_b, which is also the square root of the average power).
- Click on *Select Bounds* to evaluate the bounds and create the ground truth categories. If $\theta_U < \theta_L$, which could happen in some cases when using default bounds, θ_U will be set to equal θ_L.
- *Time ($< \theta_L$) / Time ($> \theta_U$)* displays the percentage of duration in the signal trace where the amplitude is lower than the lower bound (for false bursts), or higher than the upper bound (for true bursts).
- The resulting *True/False Peak Rate* peak rate in the ground truth data is shown in both Hz and percentage of total amplitude peaks in the synthetic LFP. You can adjust the bounds to obtain desired true/false peak rate according to experience.

Detection Performance and ROC curve:

- *Number of Thresholds* specifies the number of detection threshold to be evaluated. Adjusting this number will change the resolution of the ROC curve and distribution plot.
- *Detect on* determines the approach for the detection. *Amplitude peak* applies the detection threshold only on the peak of the amplitude and *Amplitude* applies the threshold on the amplitude at all time points.
- Click on *Evaluate* button to evaluate the detection. The ROC curve will display on the top right and the distribution of ground truth categories will display on the bottom right.
- When detecting on *Amplitude peak*, you can click on *Normalize* button in the ROC curve plot to toggle the false positive rate scale between the percentage of total false peaks and the ratio to total true peaks. The diagonal line toggles between the dotted line and the dashed line, respectively, each representing where the false positive equals the true positive measured in percentage and in Hz, respectively. ***Note:*** The diamond marker (Maximum Youden's index) indicates the point on the ROC curve that maximizes the difference from the dotted line. While the square marker (Youden's index in Hz) indicates the point that maximizes the difference from the dashed line.
- *Detection Threshold* specifies a particular detection threshold to be evaluated. You can either enter in the text box with the Z-score of the amplitude of the synthetic composite trace, or use the slider to change the absolute amplitude threshold. A marker will show up on the ROC curve to indicate the corresponding operating point and a vertical line will show up in the distribution plot to indicate the detection threshold.

- The resulting *False Positive Rate* will display in the text box and in the slider given the particular detection threshold. You can also enter a desired false positive rate in the text box or use the slider to specify it. The corresponding detection threshold will display above it and the plots will also update. This is known as the ***constant false alarm rate*** method for selecting a detection threshold.

- *Conditional Probability Given a Detection* panel shows the conditional probability of the category of each true outcome given the particular detection threshold. When you select option *> the Threshold*, given a detection with amplitude above the threshold, the probability of false positive is also known as the ***false discovery rate*** (FDR), and the probability of true positive is also known as the ***positive predictive value*** (PPV). The conditional probabilities correspond to the areas under the distribution curves to the right of the detection threshold vertical line. When you select option *= the Threshold* instead, the conditional probability of each category given a detection with amplitude exactly equal to the detection threshold will be shown instead. The conditional probabilities correspond to the proportion of each segment length along the vertical line of the detection threshold in the distribution plot.

Results:

- A text box where information/message are shown during runtime.

2.2.4 Menu

Save:

The *Save* menu provides several options to save results from the GUI.

- *Save Figures* - A pop-up window allows you to select and save the figures generated in the GUI.
- *Save Output Log* - Save all the messages printed in the *Results* panels of all three steps into a text file.
- *Screenshot* - Take a screenshot of the GUI at the current step and save it to an image file.

3 Video Instructions

Video - How to Use the Oscillatory Burst Detector Toolbox

4 Materials and Methods

4.1 The characterization and synthesis algorithm

We propose a two-part algorithm that first decomposes and characterizes the LFP's signal and background components (Fig. 1A, red), and then uses their properties to produce a synthetic LFP (Fig. 1A, yellow). The first part of the algorithm separates the power spectral density (PSD) of an observed LFP into signal and background components. The background component in an LFP exhibits a $1/f^{\beta}$ characteristic in the frequency domain with the exponent β that characterizes the 'color' in $1/f^{\beta}$-type signals [1]. A robust frequency restricted deviation from this is the signal component, comprised of oscillatory bursts whose properties are derived from a filtered version of the observed LFP. In the second part of the algorithm (Fig. 1A, yellow), attributes of the background and oscillatory bursts are used to generate synthetic burst (signal) and background traces. The synthetic background and signal components are then combined to form a synthetic LFP whose PSD will match the PSD of the LFP. This 'ground-truth' signal can then be used to evaluate the performance of an oscillatory burst detection algorithm using receiver operating characteristic (ROC) analysis (Fig. 1A, green).

4.1.1 Separating background and oscillatory signal components

We hypothesize that the LFP consists of the signal components X and the background component B, *i.e.*, $Y(t) = X(t) + B(t)$. We estimate the PSD $S_Y(f)$ of the discrete-time LFP signal $Y(n/f_s)$ with sampling frequency f_s (*e.g.*, = 1000 Hz) using Welch's method with a Hamming window of size N (*e.g.*, = 8192) and 50% overlap between windows. The PSD is then smoothed using a moving average filter with a 2 Hz sliding rectangular window (Fig. 1B). The smoothed PSD $\hat{S}_Y(f)$ is then fit by a straight line in log-log scale $\ln \alpha - \beta \ln f$ to obtain a fit of the background component $\hat{S}_B(f) = \alpha/f^\beta$ using an algorithm. Two versions of this algorithm are provided. Version-1 requires manual specification of the frequency range for fitting $[f_-, f_+]$, while version-2 determines this range automatically based on deviations from the fit. Both require the following prespecified settings:

(i) $[f^-, f^+]$ - signal frequency band, e.g. 30-100 Hz for gamma oscillation;

(ii) t_{dB} - decibel threshold in PSD which is equivalent to $t_{dB} \ln 10/10$ in natural logarithm scale, 0.95 dB by default (approximately 1.24 in linear scale);

(iii) *sample density* - density of sample frequency points in natural logarithm of Hz, 50 per neper (Np) by default.

In the pre-processing part of both versions, the sample points for fitting are chosen and evenly-spaced to the extent possible in log scale to avoid bias toward high frequency samples. A set of evenly spaced points is first generated in the interval $[\ln f_-, \ln f_+]$ with the density in the settings. Then, from the linearly spaced PSD frequency points $\{f_i = if_s/N\}_{i=1}^{\lfloor N/2 \rfloor}$, find the closest one for each generated point to form the set of sample points without repeated value $\{f_i\}_{i \in I}, I = \{i_1, \ldots, i_m\} \subset \{1, \ldots, \lfloor N/2 \rfloor\}$ where I is an increasing sequence of the m chosen indices.

Version-1 then iteratively fits $\hat{S}_B$ to $\hat{S}_Y$ on the samples with indices I using least squares, and finds the upward outliers with error $e_i = \ln \hat{S}_Y(f_i) - \ln \hat{S}_B(f_i)$ above the dB threshold (above the green line in Fig. 1B). Of these, the target outliers, defined as the segments of outliers which overlap with the signal frequency band $[f^-, f^+]$, are removed from the sample points for fitting in the next iteration, and the process continues until no target outliers remain. We identify the outliers by a set of indices $J \subset I$ and partition it into p ordered segments $J = I_{k_1^-}^{k_1^+} \cup \cdots \cup I_{k_p^-}^{k_p^+}$ which exists and is unique. A segment is defined as $I_{k^-}^{k^+} = \{i_{k^-}, i_{k^-+1}, \ldots, i_{k^+-1}, i_{k^+}\}$ where k^-, k^+ indicate the lower and upper bounds of the segment, and the partition must satisfy $k_1^- \le k_1^+ < k_2^- \le k_2^+ < \cdots < k_p^- \le k_p^+$ and $k_l^+ + 2 \le k_{l+1}^-$ for any l. We identify the signal frequency band by $R \subset I$, the indices of frequency points within it.

Version-2 determines the frequency range for fitting automatically starting from the full frequency range $[f_-, f_+] = [f_1, f_{\lfloor N/2 \rfloor}]$ for fitting. First, it fits like version-1 but for only till the second iteration. Then, target outlier segments are identified and the other segments of outliers including downward outliers that do not overlap with the signal frequency band $[f^-, f^+]$ are marked as undesired outliers which are classified as being on the lower or upper side of the range. Note that any downward outlier segment that overlap with the range is neither undesired nor target. The range for fitting $[f_-, f_+]$ is reduced on one side per iteration according to the following conditions: If one of the lowermost and the uppermost undesired outlier segments occurs on the edge, it is excluded first from the range for fitting. If neither of them is on the edge but they are on the same side, the distance from the edge to the closet one of them is reduced by half. If they are on both edges, or if neither of them is on the edge but they are on different sides, the range is reduced on the side of the segment in which higher absolute error occurs. The whole process repeats until there is no undesired outlier segment and the range for fitting is then fixed at this value for the final fit. After that step, only target outliers are removed step-wise each time till no target outliers remain, the same as version-1 where the range for fitting is prespecified.

After either version of the fit algorithm, the signal frequency range $[f_L, f_U]$ is determined by the range of the target outliers J (range of the green line denoting 'signal frequency range' in Fig. 1B). Random fluctuations in the PSD curve with respect to the $1/f^\beta$ fit necessitate specification of the dB threshold for robust determination of the signal frequency range. Deviations from the fitted background $\hat{S}_B(f) = \alpha/f^\beta$ in the signal frequency range reflect the putative oscillatory signal component.

Algorithm: version-1

Input: $\{f_i\}_{i=1}^{\lfloor N/2 \rfloor}$, $\hat{S}_Y$, $[f^-, f^+]$, t_{dB}, $[f_-, f_+]$, *sample density*
Output: α, β, $[f_L, f_U]$

Get indices $I = \{i_1, \dots, i_m\}$ of sample points $\{f_i\}$ in range $[f_-, f_+]$ with *sample density*
$R \leftarrow \{i \in I : f^- \leq f_i \leq f^+\}$ // Get indices within the signal frequency band
$\alpha, \beta, J \leftarrow \textsc{fit}(m)$ // Set max iteration to the number of elements in I

procedure $\textsc{fit}$(*max iteration*)
  Initialize $J \leftarrow \emptyset$, $J' \leftarrow \emptyset$
  repeat
    $(\alpha, \beta) \leftarrow \arg\min_{(\alpha,\beta)} \sum_{i \in I \setminus J} e_i^2$ // Fit $\hat{S}_B$ to $\hat{S}_Y$
    $J \leftarrow \{i \in I : e_i > t_{dB} \ln 10/10\}$ // Find upward outliers
    $I_{k_1^-}^{k_1^+} \cup \cdots \cup I_{k_p^-}^{k_p^+} \leftarrow J$ // Find segments
    for $l \leftarrow 1$ to p **do**
      if $I_{k_l^-}^{k_l^+} \cap R = \emptyset$ **then**
        $J \leftarrow J \setminus I_{k_l^-}^{k_l^+}$
      end if
    end for
    changed $\leftarrow J \neq J'$
    $J' \leftarrow J$
  until not *changed* or *max iteration* reached
  return α, β, J
end procedure

$f_L \leftarrow f_{\min(J)}$, $f_U \leftarrow f_{\max(J)}$ // Get signal frequency range

Algorithm: version-2

Input: $\{f_i\}_{i=1}^{\lfloor N/2 \rfloor}$, $\hat{S}_Y$, $[f^-, f^+]$, t_{dB}, *sample density*
Output: α, β, $[f_L, f_U]$, $[f_-, f_+]$

Initialize $f_- \leftarrow f_1$, $f_+ \leftarrow f_{\lfloor N/2 \rfloor}$
Get indices $I = \{i_1, \dots, i_m\}$ of sample points $\{f_i\}$ in range $[f_-, f_+]$ with *sample density*
$R \leftarrow \{i \in I : f^- \leq f_i \leq f^+\}$ // Get indices within the signal frequency band
$i^- \leftarrow \min(R)$, $i^+ \leftarrow \max(R)$ // Get indices of the bounds of R
repeat
  $\alpha, \beta, J \leftarrow \textsc{fit}(2)$ // Fit for 2 iterations
  $K_u \leftarrow \{i \in I : e_i > t_{dB} \ln 10/10\} \setminus J$ // Find undesired upward outliers
  $K_d \leftarrow \{i \in I : e_i < -t_{dB} \ln 10/10\}$ // Find undesired downward outliers
  $I_{k_1^-}^{k_1^+} \cup \cdots \cup I_{k_p^-}^{k_p^+} \leftarrow K_u$ // Find segments
  $I_{k_{p+1}^-}^{k_{p+1}^+} \cup \cdots \cup I_{k_q^-}^{k_q^+} \leftarrow K_d$
  reduce $\leftarrow q > 0$ // Break if no undesired outliers found
  if *reduce* **then**
    reduce $\leftarrow$ REDUCTION RULE
    $f_- \leftarrow f_{i_1}$, $f_+ \leftarrow f_{i_m}$ // Update the frequency range for fitting
  end if
until not *reduce*
$\alpha, \beta, J \leftarrow \textsc{fit}(m)$ // Set max iteration to the number of elements in I
$f_L \leftarrow f_{\min(J)}$, $f_U \leftarrow f_{\max(J)}$ // Get signal frequency range

Algorithm: version-2 (`Reduction Rule`)

procedure REDUCTION RULE
  // The reduction of range for fitting acts on one side per iteration
  // $\{-,+\}$ are used as indices to indicate lower or upper side
  $k_-^- \leftarrow \min_l\{k_l^-\}$, $k_-^+ \leftarrow \min_l\{k_l^+\}$ // Lower and upper bounds of the lowermost segment
  $k_+^- \leftarrow \max_l\{k_l^-\}$, $k_+^+ \leftarrow \max_l\{k_l^+\}$ // Lower and upper bounds of the uppermost segment
  $j_- \leftarrow 1$, $j_+ \leftarrow m$ // Fisrt and last index in I
  Find j^-, j^+ such that $i_{j^-} = i^-, i_{j^+} = i^+$ // Indices in I of the bounds of R
  // Determine the condition out of three cases and choose the side to reduce the range
  for s in $\{-,+\}$ **do**   // Identify condition for each side using numbers and assign to C_-, C_+
    if $sk_s^{-s} \leq sj^s$ **then**
      $C_s \leftarrow 0$ // No segment is on the s side of range R
    else if $k_s^s \neq j_s$ **then**
      $C_s \leftarrow 1$ // No segment is on the edge of the s side
    else
      $C_s \leftarrow 2$ // A segment is on the edge of the s side
    end if
  end for
  $C^* \leftarrow \max_s\{C_s\}$ // Prioritize the condition with greater number
  if $C^* = 0$ **then**
    return False   // Break if no undesired segment on either side
  end if
  if $C_- = C_+$ **then**   // Determine the side on which to reduce range for fitting
    $s^* \leftarrow \arg\max_{s\in\{-,+\}} \left\{ \max_{i \in I_s} |e_i|, \text{where } I_s = I_{k_s^-}^{k_s^+} \right\}$ // side of the segment with greater error
  else
    $s^* \leftarrow \arg\max_s\{C_s\}$ // side of greater condition number
  end if
  // Apply reduction on side s^* based on condition C^*
  if $C^* = 2$ **then**
    $j_{s^*} \leftarrow k_{s^*}^{-s^*} - s^* \cdot 1$ // Reduce the range to exclude the segment on the edge
  else
    $j_{s^*} \leftarrow s^* \left\lfloor s^* \left(j_{s^*} + k_{s^*}^{s^*} \right) / 2 \right\rfloor$ // Reduce the distance from the edge to the segment by half
  end if
  $I \leftarrow \{i_{j_-}, \ldots, i_{j_+}\}$ // Update indices of the reduced range
  $\{i_1, \ldots, i_m\} \leftarrow I$ // Relabel the indices in I
  return True
end procedure

4.1.2 Characterizing oscillatory bursts

The observed LFP is bandpass filtered by a zero-phase 6-th order Butterworth filter with cutoff frequencies corresponding to the boundaries of the signal frequency range $[f_L, f_U]$ obtained from the PSD (Fig. 1C). Taking the magnitude of the analytic signal obtained by Hilbert transform yields an amplitude time series. To extract the properties of oscillatory bursts, periods where the amplitude exceeded 2 Z-score were deemed significant. These periods were used to obtain the following attributes. Amplitude peak is defined as the peak value of the amplitude of a burst. The burst duration is defined as a continuous period when the amplitude stays above 25% of the amplitude peak. Burst main frequency is defined as the frequency at which the maximum peak occurs in the discrete Fourier transform (DFT) magnitude of a burst within the signal frequency range. A DFT (with length of 4096 for $fs = 1000$ Hz) of a burst is obtained from the raw LFP within the burst duration using a Tukey window of the same size as the duration. Zero padding is used if the duration is shorter than the length of DFT and the duration is truncated if the opposite. The number of cycles is defined as the duration multiplied by the burst main frequency. When the durations of two bursts overlap, the one with lower amplitude peak is omitted. This can be identified as the duration of the burst with the lower amplitude includes that of the other. A burst is also omitted when there is no peak in its DFT magnitude within the signal frequency range. The empirical distributions of number of cycles per burst and burst main frequency are obtained during significant periods (Fig. 1D). To obtain the distribution of amplitude peaks, all peaks in the amplitude time series were included. We found this provided a better bound for the scheme used to optimize the distribution parameters later. The correlation coefficients between the three attributes in logarithm scale of the bursts in significant periods were also obtained.

4.1.3 Generating a synthetic LFP

In the second part of the algorithm, the synthetic signal and background components of the LFP are generated (Fig. 1A, yellow). These are then summed to form a synthetic LFP whose PSD matches that of the observed LFP.

4.1.4 Synthetic background component

We first determine the desired PSD of the synthetic background to be

$$S_b(f) = \begin{cases} \hat{S}_B(f) = \dfrac{\alpha}{f^\beta} & \text{if } f_a < f < f_b \\ \hat{S}_Y(f) & \text{otherwise} \end{cases}$$

where $f_a < f_L$ and $f_b > f_U$ are the frequencies at the two intersection points between the smoothed PSD curve $\hat{S}_Y(f)$ and the fit curve $\hat{S}_B(f)$ near the boundaries of the signal frequency range $[f_L, f_U]$ (blue line 'fit background component' in Fig. 1B). If the smoothed PSD and the fit curve does not intersect on one side, take the point with minimum difference between them instead. The fitted curve is used to replace the smoothed PSD within the two intersection points.

A data sequence of Gaussian white noise $w(n)$ is generated, *i.e.*, $w(n) \sim \mathcal{N}(0, 1)$ and $\mathbb{E}[w(m)w(n)] = \delta_{mn}$ for any m, n. Then it is passed through a Type I linear phase FIR filter (MATLAB's fir2 function) whose frequency-magnitude characteristics matched the desired background PSD $S_b(f)$ by linearly interpolating the desired frequency response onto a dense grid and then using the inverse Fourier transform and a Hamming window to obtain the filter coefficients. The length of the FIR filter is selected to achieve a frequency resolution of 1 Hz. The output is then scaled to match the background power in the frequency range $[f_a, f_b]$ as follow to obtain the synthetic background b.

$$b(n) = \left(\frac{\sum_{k=0}^{N/2} |G(k)|^2}{\sum_{k=\lceil f_a N/f_s \rceil}^{\lfloor f_b N/f_s \rfloor} |G(k)|^2} \int_{f_a}^{f_b} \hat{S}_B(f) df \right)^{1/2} Z(w * g)(n)$$

where G is the N-point DFT of the filter impulse response g, and $Z(w * h) = (w * h - \mu_{w*h})/\sigma_{w*h}$ is the Z-score transform of the filtered sequence $w * g$ where μ_{w*h} and σ_{w*h} are the mean and standard deviation of the sequence.

4.1.5 Synthetic signal component

We generate individual bursts as Gabor atoms $a_i(n) = A_i \sin(2\pi n F_i/f_s + \varphi_i) \exp\left(-\frac{1}{2}\left(\frac{n-\tau_i}{f_s D_i}\right)^2\right)$ for the i-th atom (sinusoids modulated by Gaussian envelopes) and add them up to form the synthetic signal $x(n) = \sum_i a_i(n)$. The variables A_i, F_i, φ_i and τ_i denote the amplitude peak, main frequency, phase and peak time of each atom, respectively. The width parameter of the Gaussian envelope $D_i = C_i/(2dF_i)$ determines the burst atom duration and depends on the number of cycles C_i and the burst main frequency, and $d \approx 1.665$ is a constant such that $exp(-d^2/2) = 0.25$ which follows the definition of burst duration at 25% amplitude peak cutoff for characterization of the observed bursts.

The attribute triplet (A_i, C_i, F_i) of each Gabor atom is generated in two steps. First, use the empirical correlation coefficient matrix (in log-scale) of the observed burst attributes to generate a three-dimensional Gaussian copula. This is performed in two steps. First, sample from a three-dimensional normal distribution with the given covariance matrix, and then transform the samples using the cumulative distribution function on each component, whose underlying marginal distribution is uniform on the interval $[0, 1]$. Second, transform the random samples from the Gaussian copula to those following a joint distribution with the desired marginal distributions by applying the inverse cumulative probability function on each component. Empirical distributions are used for the marginal distributions of cycle number and burst frequency, while the marginal for amplitude peak has a parametric form, typically a gamma distribution, with probability density function $f(x; k, \theta) = \frac{1}{\Gamma(k)\theta^k} x^{k-1} e^{\frac{x}{\theta}}$ where $\Gamma(k) = \int_0^\infty t^{k-1} e^{-t} dt$ is the gamma function.

Since individual bursts are assumed to occur independently of each other and of the background, they are added to the background trace with τ_i's independently and uniformly distributed on $\{1, \ldots, M\}$ with $T = M/f_s$ being the total duration of the synthetic signal to be generated. The phase φ_i's are also assumed to independently follow uniform distribution. Our empirical checks indicate that the signal amplitude and phase are independent, and the phase at amplitude peak was found to follow uniform distribution, i.e., without preference. Bursts are accumulated until the total power equals that in the signal portion of the observed LFP PSD. This can be verified by calculating the sum of individual burst energy. The energy of each individual burst can be calculated as $E_i = \frac{\sqrt{\pi}}{2} A_i^2 D_i R_i$ where R_i is the proportion of energy remaining after bandpass filtered in $[f_L, f_U]$. Since the Fourier transform magnitude of a Gabor atom is a Gaussian function, we can estimate it by

$$R_i = \frac{1}{\sqrt{2\pi}\sigma_i} \int_{f_L}^{f_U} \exp\left(-\frac{1}{2}\left(\frac{f - F_i}{\sigma_i}\right)^2\right) df = \frac{1}{2}\left(\operatorname{erf}(2\pi D_i(f_U - F_i)) - \operatorname{erf}(2\pi D_i(f_L - F_i))\right)$$

where $\sigma_i = (2\sqrt{2}\pi D_i)^{-1}$ and $\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt$ is the error function. Then the condition for matching the signal power is $\sum_i E_i = T \int_{f_L}^{f_U} (\hat{S}_Y(f) - \hat{S}_B(f)) df$.

The resulting trace $y(n) = x(n) + b(n)$ is then filtered by the same bandpass filter used in characterization of observed signal and termed the synthetic composite signal. However, this synthetic signal is biased because the amplitude peak distribution of the observed LFP is influenced by the $1/f^\beta$ background, which contains an overabundance of small amplitude peaks. When these bursts are generated in the synthetic trace, together with the background trace, they skew the amplitude peak distribution to the low end (Fig. 1E, red).

To address this problem, we optimize the match between the synthetic composite and observed amplitude peak distributions. Specifically, we optimize the parameters for the amplitude peaks distribution used to generate synthetic burst atoms so that the synthetic composite amplitude peak distribution matches the observed. Formally, we define the synthetic burst atom amplitude distribution parameters as independent variables and the Kullback–Leibler divergence measure D_{KL} [3] between the observed and synthetic composite amplitude peak distributions as the cost. Since amplitude peak is a continuous random variable, the histogram of it with 60 bins is used to approximate its distribution using discrete distribution. The count in each bin of the histogram is added by one to avoid zero probability. If p_i and q_i are the probability of the i-th bin of the observed and synthetic composite amplitude peak, respectively, then $D_{KL}(p||q) = \sum_i p_i \ln(p_i/q_i)$.

Optimization is performed using the particle swarm algorithm [2]. Bounds of the parameters for the optimization are set to some multiples (0.2 for the lower bound and 3 for the upper bound) of the parameters estimated from the observed amplitude peak distribution. To avoid multiple solutions that may exist due to degrees of freedom in the background signal distribution, we assume that the background signal follows a Gaussian distribution.

4.2 The detection algorithm – Using the synthetic ground truth to evaluate the ROC of oscillatory burst detection

The characterization algorithm generated a synthetic LFP that matched features of the observed LFP. Consequently, the synthetic LFP can serve as a ground truth for evaluating the detection of oscillatory bursts using ROC analysis. The detection problem has two parts. The first is to use the characteristics of the synthetic background and signal traces to determine appropriate limits for the classification of salient oscillatory bursts in the synthetic composite LFP. With this 'ground truth', the second task is to determine the optimal detection threshold using ROC analysis.

4.2.1 Detection of salient burst peaks in the synthetic signal trace.

Since the synthesis algorithm provides separate background and signal traces, these can be used to generate ground truth data for evaluating the detection of salient oscillatory burst peaks in the composite trace. By salient, we are referring to oscillatory events in the synthetic signal trace whose amplitude deviates from the synthetic background trace by some predefined degree.

One may wonder why salient oscillatory events must be defined, given that the synthetic signal trace was constructed using burst atoms with known peak times and amplitudes. At first glance, just knowing these burst times should provide sufficient information for determining whether an oscillatory burst was present or not. However, the synthetic signal trace often contains many overlapping low amplitude burst atoms, obscuring their peak in the signal trace. Due to such overlap, every burst will not have its own distinct amplitude peak in the synthetic signal trace. This arises from the fact that at any given time n, the signal trace can be modeled as $x(n) = \sum_i a_i(n)$ where each $a_i(n)$ is a burst atom with amplitude peak A_i. If all A_i's are small and there is significant overlap at n, $x(n)$ will approach a Gaussian distribution (by the central limit theorem), with constructive and destructive interference of the burst amplitude peaks. To address this, we define a low amplitude bound, θ_L, in the synthetic signal trace below which bursts are not distinguishable from background.

If the signal trace is composed of sparsely distributed high amplitude bursts, then it may be desirable to define a high amplitude bound, θ_U, to detect these events specifically. Consider the case where there is one burst atom $a_k(n)$ having a large amplitude A_k. We should be able to detect the amplitude of the k-th atom, even in the presence of other atoms $\sum_{i\neq k} a_i(n)$. Extending this idea, in general, if we use a high amplitude bound θ_U in the signal trace to define the true burst activity, the occurrence of the burst atoms with significant amplitude peaks higher than θ_U will be sparse and hardly overlap.

4.2.2 Selection of the lower bound θ_L in the synthetic signal trace

We hypothesize that oscillatory bursts in the signal trace with expected instantaneous power lower than the average power in the background trace are difficult to distinguish, highlighting the need to define a lower bound θ_L (Fig. 2A). The average power of the background trace with standard deviation σ_b is σ_b^2. Equating the expectation of instantaneous power in the signal trace to average background power leads to the value of the bound as $\theta_L = \sqrt{2}\sigma_b$ as follows: Denote the analytic signal of the signal trace as $x_a(n) = x(n) + j\hat{x}(n)$, where $\hat{x}$ is the Hilbert transform of the signal trace x and j is the imaginary unit. Since the phase φ_i is independent of amplitude A_i for burst atoms, the phase $\arg(x_a(n))$ and the amplitude $|x_a(n)|$ of the analytic signal are uncorrelated, so the real part $x(n)$ and the imaginary part $\hat{x}(n)$ follow identical distribution. Then, if the amplitude $|x_a(n)|$ is given, the conditional expectation of amplitude over random phase is $|x_a(n)|^2 = \mathbb{E}|x_a(n)|^2 = \mathbb{E}[x(n)^2 + \hat{x}(n)^2] = 2\mathbb{E}[x(n)^2]$, where $\mathbb{E}[x(n)^2]$ is the expectation of the instantaneous power. If $\mathbb{E}[x(n)^2] = \sigma_b^2$, then $|x_a(n)| = \sqrt{2}\sigma_b$.

The analytic signal amplitude of the background trace follows the Rayleigh distribution $f(x;\sigma_b) = \frac{x}{\sigma_b^2}e^{-x^2/2\sigma_b^2}$ with parameter σ_b. The probability of the background amplitude below θ_L is 0.632. Or, equivalently, the background trace is below θ_L for 63.2% of the total duration. Consequently, only bursts in the signal trace with amplitude above θ_L are defined as salient.

4.2.3 Selection of an upper bound θ_U in the synthetic signal trace

An additional, more stringent, threshold can be used for defining salient bursts. Two approaches are proposed for the selection of the upper bound θ_U for amplitude peaks in the synthetic signal trace (Fig. 2B). The first determines θ_U using a data-driven criterion. In the first approach, the value of θ_U is set so that the average power of the signal trace over the duration with amplitude below θ_U equals the power of the signal component within the signal frequency range below the dB threshold in PSD. We divided the power in the time domain using θ_U in the synthetic signal (Fig. 2B) and in frequency domain in the PSD of the observed data (Fig. 1B), hypothesizing that the period below θ_U contributes to the power below the dB threshold in the PSD which is within the fluctuation level. If this value of $\theta_U < \theta_L$, it is set to θ_L. To account for a small random error between the total power of the synthetic signal trace and the corresponding total power in the signal component in the PSD of the observed data, instead of equating the powers, we equate the proportion of powers for a better estimate of θ_U, as follows

$$\frac{\frac{1}{|T_U^c|}\sum_{n\in T_U^c}|x_a(n)|^2}{\frac{1}{M}\sum_{n=1}^{M}|x_a(n)|^2} = \frac{\int_0^{f_s/2}|H(f)|^2\left(\min\left(10^{r/10}\hat{S}_B(f),\hat{S}_Y(f)\right)-\hat{S}_B(f)\right)df}{\int_0^{f_s/2}|H(f)|^2\left(\hat{S}_Y(f)-\hat{S}_B(f)\right)df}$$

where $T_U^c = T_L \cup T_I = \{n : |x_a(n)| \leq \theta_U\}$ is the period when the amplitude is below θ_U (formal definition of T_L, T_U, T_I is in the next section), $|T_U^c|$ is the number of time points in the period, M is the number of discrete time points over the duration T of the synthetic data, H is the frequency response the bandpass filter and r is the dB threshold. The left hand side is the ratio of the power in period T_U^c to the total power in the synthetic signal trace. The right hand side is ratio of the power above the background and below the dB threshold (the area between the blue and green line in Fig. 1B), to the total power in the signal component in the PSD of the observed data. The magnitude response $|H(f)|$ is used to correct the effect of power reduction by the filter. Only the numerator on the left hand side depends on θ_U, which is found by searching over the cumulative sum of sorted $|x_a(n)|^2$.

A second approach uses an expected rate of occurrence of oscillatory burst events in the observed data. This rate can be represented either as a burst rate (in Hz) in either the signal or the composite trace, or as the percent time spent in a burst state. The algorithm will then pick a θ_U that results in the expected rate of oscillatory bursts.

When using the bounds θ_L and θ_U, bursts can be categorized as follows for developing an ROC: those above θ_U are salient, those below θ_L are insignificant, and those between the two are intermediate (*i.e.*, indeterminant), which will be ignored in the ROC analysis.

4.2.4 Identifying true and false peaks in the synthetic composite signal

With the specification of upper and lower bounds in the synthetic signal trace, we propose an approach to define true and false peaks as follows (Fig. 2C). The discrete time points are partitioned into three subsets $T_L = \{n : |x_a(n)| < \theta_L\}$, $T_U = \{n : |x_a(n)| > \theta_U\}$ and $T_I = \{n : \theta_L \leq |x_a(n)| \leq \theta_U\}$. The set T_L denotes the period when the signal trace amplitude is below θ_L and T_U the period when the amplitude is above θ_U. The remaining period is T_I with I denoting 'intermediate'.

Consider now the synthetic composite trace obtained by adding the background trace to the signal trace. An amplitude peak $|y_a(n)|$ in the synthetic composite signal can be classified as either *true*, *false*, or *intermediate*, depending on whether its time n belongs to the T_U, T_L, or T_I time period, respectively (Fig. 2C). Since these periods are defined with respect to the signal trace, but the amplitudes are evaluated on the composite trace, the distribution of amplitudes labeled as salient, insignificant, and intermediate will overlap

(Fig. 2D). Then, peaks above the detection threshold are labeled *positive*, or *negative* if below. From this, ROC curves for the detection of bursts can be calculated.

4.2.5 ROC analysis of burst detection threshold

By systematically varying the detection threshold, it is possible to optimize the detection of oscillatory bursts using ROC analysis. In general, ROC analysis systematically varies the detection threshold while tracking the rate of True Positives (TP) and False Positives (FP, Fig. 2E). When detection is random, the TP rate (TPR) equals the FP rate (FPR), while an optimal detection threshold maximizes TP and minimizes FP. Here $\mathrm{TPR} = \mathrm{TP}/(\mathrm{TP} + \mathrm{FN})$ and $\mathrm{FPR} = \mathrm{FP}/(\mathrm{TN} + \mathrm{FP})$. Specific to the current task, detection of oscillatory bursts can be carried out in two ways. One is to detect their peak times, which is appropriate for offline analysis of pre-recorded LFP data. The second is to detect moments of elevated instantaneous power in the burst frequency range, which is useful for real-time processing applications where the burst peak is ambiguous.

4.2.6 Detection of salient oscillatory peaks

The first approach characterizes detection of salient burst amplitude peaks in the synthetic composite trace. The upper and lower bounds, θ_L and θ_U, applied to the synthetic signal trace determine whether a peak in the synthetic composite trace is a TP or FP.

4.2.7 Detection of salient oscillatory burst periods

The second approach detects salient burst periods when the synthetic composite trace instantaneous amplitude exceeds the detection threshold, as opposed to the detection of only peaks as in the first approach.

4.2.8 Criteria for optimal burst detection thresholds

Once the ROC curve is calculated a criterion is used to determine the optimal detection threshold. One wants to maximize the TP rate, while minimizing the FP rate. The Youden index $J = \mathrm{TPR} - \mathrm{FPR}$ identifies this by finding this point on the ROC curve that maximizes J, which geometrically corresponds to the point where the ROC curve maximally deviates chance, the line where $\mathrm{TPR} = \mathrm{FPR}$ (Fig. 2E).

5 Limitations

The observed amplitude peak distribution being more skewed to the right is indicative of a high fraction of low-amplitude peaks. In such cases, the match between the observed and synthetic amplitude peak distributions appears to be not as good after the D_{KL} optimization converges. This usually happens when working with the ripple waves, but not with the gamma or beta waves which do not have high skewness. However, the PSD plots between the observed and synthetic LFP showed very good match for the ripple case, as for the other frequency bands. Future research could explore what factors may explain the mismatch between the observed and synthetic amplitude distributions. Such factors, perhaps also interacting with each other, could include (i) low burst rate seemly associated with high skewness of the amplitude distribution, (ii) overestimation of the $1/f^{\beta}$ background power, and (iii) the assumption of Gaussian distributed noise in the generation of the background component.

A possible solution for this issue has been implemented in the code but not enabled by default since it has not been validated. For this case, we assumed that the power in the background may be estimated incorrect. To adjust for this, we included a factor between 0 and 1 to scale the background. This will increase the power in the synthetic signal component so that it contributes more to the composite signal so that the amplitude peak distribution can be matched. This additional parameter is used to run a second optimization when $D_{KL}(p||q) > 0.04$.

6 Source Code

The source codes are in the folder *"source"* of the Github repository `https://github.com/chenziao/Matlab_Toolbox-Oscillatory_Burst_Detector.git`.

Most of the functions used by the GUI have descriptive comments inside their source codes. To run the algorithm without using the GUI, run the example scripts in the following order:

1. *"CharaterizeInVivo_example.m"* - Step 1 in the GUI.
   "CharaterizeInVivo_fooof_example.m" - An alternative method for fitting the PSD in step 1 using FOOOF algorithm. You need to install the package first. This script runs FOOOF using a MATLAB wrapper.

2. *"AmpDistParamOptimization_example.m"* - Step 2 in the GUI.
   "GenSynthetic_example.m" - (Optional) Generates and characterizes the synthetic data.

3. *"Synthetic_Analysis_example.m"* - Step 3 in the GUI. This script can be adapted for customized detection method (see the example in a commented block in the script).

The class ***SynthParam*** defined in *"SynthParam.m"* plays a central role in generating the synthetic data.

References

[1] Biyu J He, John M Zempel, Abraham Z Snyder, and Marcus E Raichle, *The temporal structures and functional significance of scale-free brain activity*, Neuron **66** (2010), no. 3, 353–369.

[2] James Kennedy and Russell Eberhart, *Particle swarm optimization*, Proceedings of icnn'95-international conference on neural networks, 1995, pp. 1942–1948.

[3] Solomon Kullback and Richard A Leibler, *On information and sufficiency*, The annals of mathematical statistics **22** (1951), no. 1, 79–86.