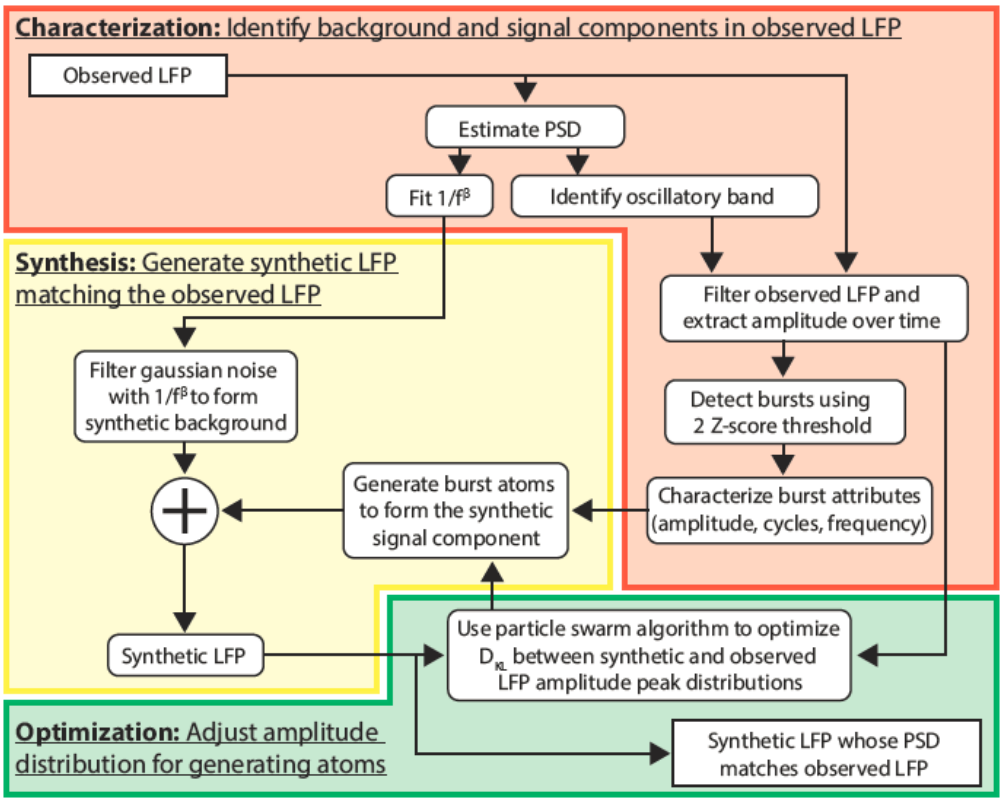
User's Manual

Contents

1 Introduction

Schematic of analysis and synthesis algorithm



The work flow has 3 major steps: 1. Characterization, 2. Synthesis, 3. Analysis. Each step does following things:

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 and calculate signal-to-noise ratio (SNR). Then characterize bursts properties, including amplitude peaks, number of cycles and burst frequency. Finally save the characterization results.

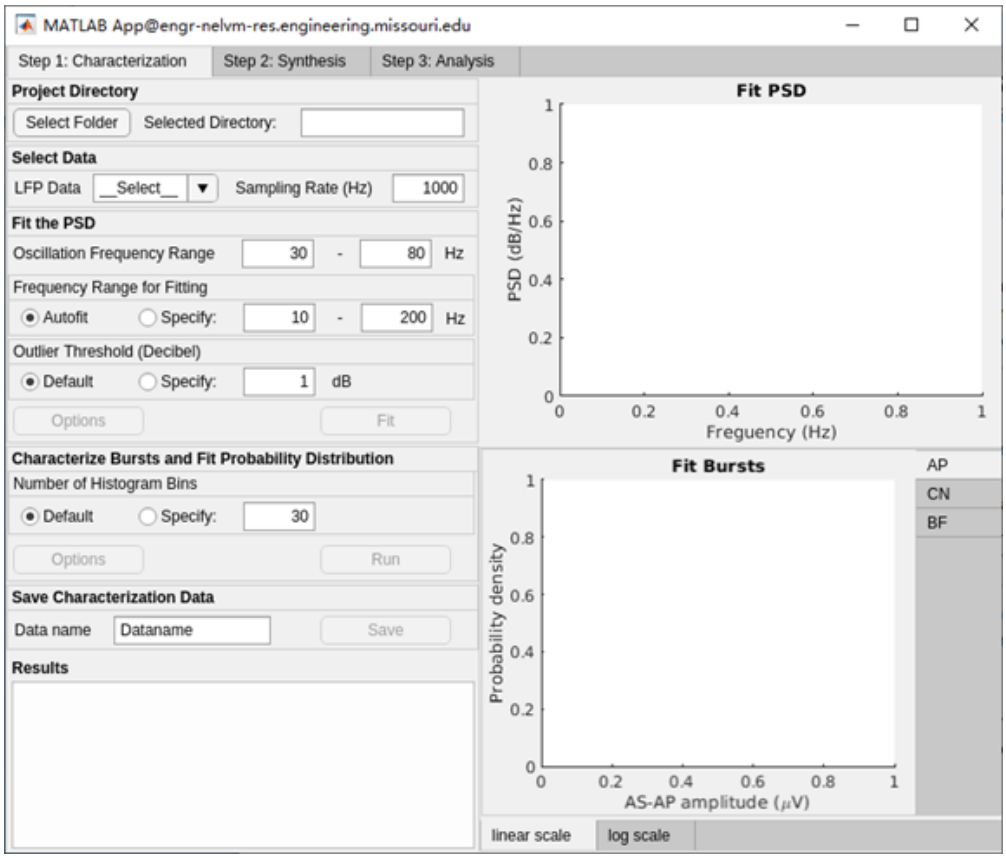
2. Load results from previous step. Specify parameters for synthesizing the background component and oscillatory signal component. Then run an optimization for amplitude peak distribution parameters of burst atoms to be generated in the signal component. This step is essential for the synthetic data to reproduce the amplitude peak distribution of the observed data. There are three candidate types of distribution for the burst atom amplitude peak: exponential, gamma, and lognormal. 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 previous step for the detection problem. Define the ground truth with lower and upper bounds in the synthetic signal trace and evaluate detection performance on the composite trace with receiver operating characteristic (ROC) curve. The relation between detection threshold and true/false positive rates will be shown.

2 Graphical user interface and instructions

2.1 Characterization

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 data, your LFP recording. Select a variable from your Matlab workspace or type in the variable name or an expression. The variable needs to be a ***vector*** of LFP data or a ***cell array*** where each element is a vector of LFP segments. The latter format is useful for segmentized data.
- *Sampling Rate* is the recording sampling rate in Hz.

Fit the PSD:

- *Oscillation Frequency Range* is the signal frequency band of the oscillation you are interested in.
- *Frequency Range for Fitting* is the range where the algorithm tries to fit a $1/f^{\beta}$ curve to the power spectral density. You can adjust the range to find a best fitting for the $1/f^{\beta}$ background component. Or you can also use the *Autofit* option to automatically find the range.

- *Outlier Threshold* is the decibel threshold in the PSD used to find a bump in the signal frequency band that indicates significant oscillatory signal power and to determine the signal frequency range. Try lowering this value if a bump is too small to be found.
- Click on *Fit* button to run. Fitting results will show up in the plot on the top right.

Characterize Bursts and Fit Probability Distribution:

- *Number of Histogram Bins* specifies the resolution of the distribution histogram. Adjust this number according to your data size.
- Click on *Run* button to run characterization. Burst statistics will show up in the plot on the bottom right. Switch the tabs to view different plots.

Save Characterization Data:

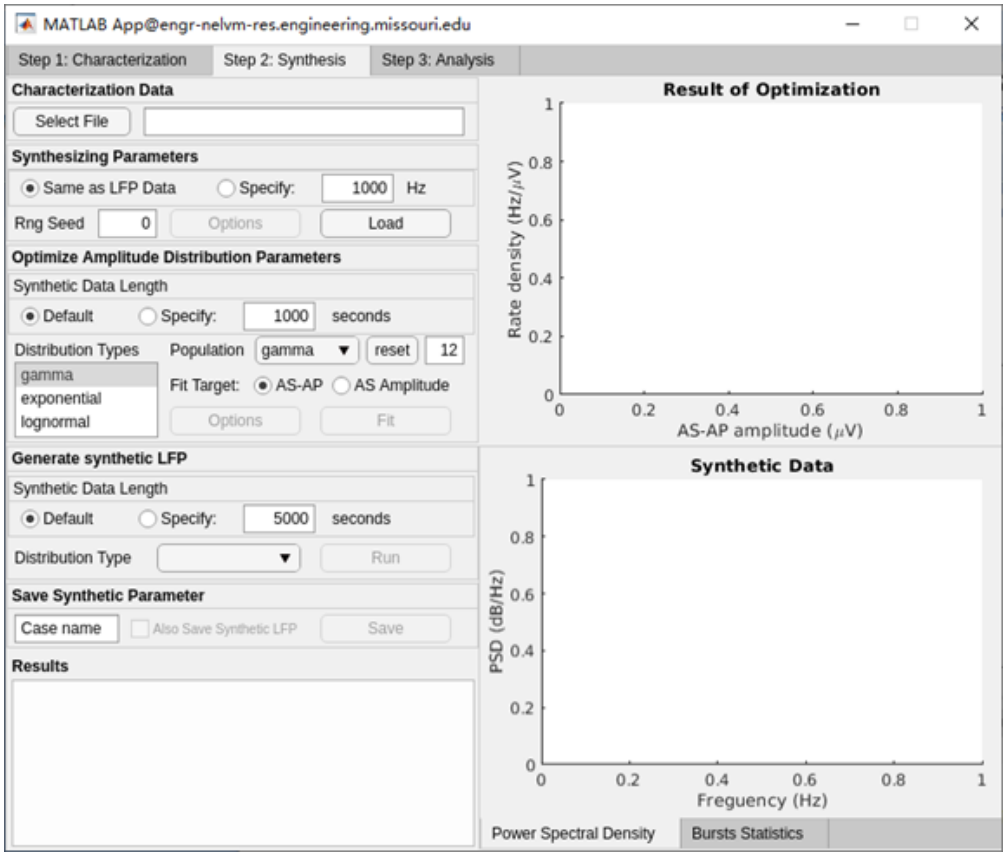
- Type a name for the characterization results and click on *Save* button. A dialog box will open and save the result file under the project directory.

Results:

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

2.2 Synthesis

Step 2: Synthesis



Characterization Data:

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

Synthesizing Parameters:

- *Sampling Rate* specifies the sampling rate of the synthetic signal. You can use the same rate as the LFP data which yields best reproduction of it, or you can specify a different one. ***Note:*** Large sampling rate will significantly slow down the process, especially during the optimization.
- *Rng Seed* specifies the seed for random number generator.
- When parameters are set, click on *Load* button before moving to next step.

Optimize Amplitude Distribution Parameters:

- ***Particle swarm optimization*** is used. It creates a population of models with different parameters and they will be attracted toward the point in the parameter space with minimal loss found after each iteration. The convergence criteria is that the number of stall iterations (consecutive iterations with no lower loss value found) reaches 5, or maximal number of iterations (30) is reached.
- *Synthetic Data Length* is the duration of generated signal. Shorter ones yield shorter runtime while longer ones yield lower variance in the resulting distribution meaning lower error in the loss.
- *Distribution Types* lists the available candidate distributions for the burst atom amplitude peak distribution that are used to generate the synthetic signal component. You can select multiple items by holding "Ctrl" / "Shift" + "Left Click".
- *Population* specifies the particle swarm population for each candidate distribution. Select the candidate in the dropdown menu and type in the number in the text box. Clicking on reset button will restore default number. Large population helps searching the parameter space but takes longer to simulate.
- *Loss Function* is the loss for the optimization. It can be either KL divergence or JS divergence which indicates the similarity between the amplitude peak distributions of the observed LFP and synthetic LFP.
- Click on *Fit* button to run. ***Note:*** The optimization may take several minutes to finish. The resulting amplitude peak distributions of synthetic data generated by optimized parameters of selected candidate distribution types will show up in the plot on the top right, together with the distribution of observed data.

Generate Synthetic LFP:

- *Synthetic Data Length* is similar to that in previous step. It can be longer since it runs only once. Longer synthetic data yields lower variance in statistics for latter analysis.
- *Distribution Type* is the one among the optimized ones you will use 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 show up in the plot on the bottom right.

Save Synthetic Parameters:

- Save the synthesizing 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 data file size could be large. Saving the data is not essential for step 3 because the synthetic data can be reproduced solely by the synthesizing parameters saved. Data generated by only one *Distribution Type* will be saved in one file. You can rerun *Generate Synthetic LFP* with another distribution type and save the data to another file.
- Type in the case name you want for the file in the text box. The case name will be a suffix following the characterization data file name.

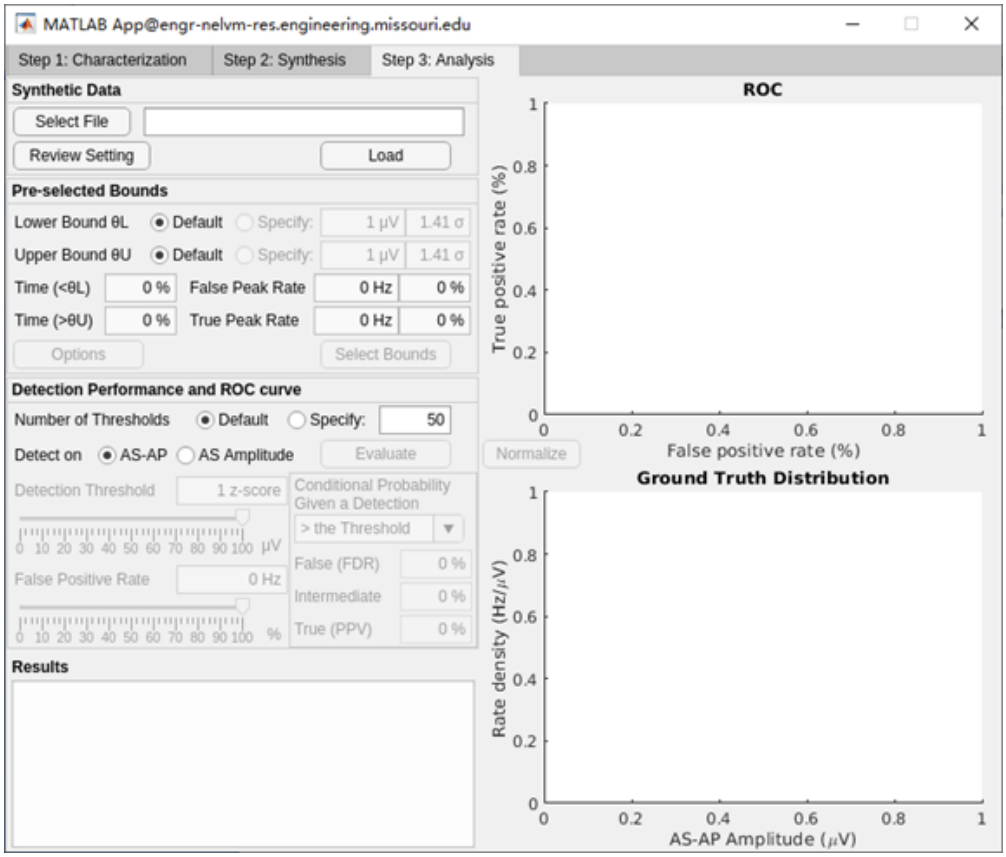
- Click on *Save* button, a dialog box will show up and save the file in the project directory. ***Note:*** The synthesizing parameter file has to be in the same directory as the characterization data file in order to work properly.

Results:

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

2.3 Analysis

Step 3: Analysis



Synthetic Data:

- Select the synthesizing parameter file you saved from step 2 using the *Select File* button or type in the file path in the text box.
- Click on *Review Setting* to review the synthesizing parameters. The GUI will switch to step 2 and the setting for the synthesizing parameters will be shown on the panels.
- Once you confirm with the setting, you can click on the *Load* button to load the synthetic data. If the synthetic data was not saved in the synthesizing parameter file, it will take a while to generate the data. The synthetic LFP and its two components, background and signal, will be 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 of three categories in the synthetic LFP, ***true***, ***false*** bursts, and an

intermediate category. You can select default to use recommended bounds. Or you can specify the bound in unit of amplitude or in multiples of the standard deviation of the background trace (σ_b, also the square root of power).

- Click on *Select Bounds* to evaluate the bounds and create ground truth data. If $\theta_U < \theta_L$, which could happen in some cases when using default bounds, θ_U will be set to equal θ_L. The percentage of duration where the signal trace have amplitude lower than the lower bound ($< \theta_L$), or higher than the upper bound ($> \theta_U$), will be shown. The resulting ***true*/*false*** peak rate in the ground truth data will be 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.
- You can choose the target of detection. *AS-AP* option detects only the amplitude peaks in the analytic signal. *AS amplitude* option detects the analytic signal amplitude at all time points.
- Click on *Evaluate* button to evaluate the detection. The ROC curve will show up on the top right and the distribution of ground truth will show up on the bottom right.
- When detecting on *AS-AP*, you can click on *Normalize* button to toggle the false positive rate scale between the percentage of total false peaks and ratio to total true peaks.
- *Detection Threshold* specifies a particular detection threshold you want to evaluate. You can either type in the text box using z-score of the amplitude of the synthetic composite trace, or use the slider to change the amplitude threshold. A marker will show up on the ROC curve indicating current operation point and a line will show up in the distribution plot indicating the threshold location.
- *False positive rate* shows the resulting false positive rate in the text box and the slider given the specified detection threshold. You can also type in the text box or use the slider to specify a desired false positive rate and the corresponding detection threshold will be shown in the slider and text box above. This is known as the ***constant false alarm*** method for selecting a detection threshold.
- *Conditional Probability given a Detection* panel shows the conditional probability of each category a detection belongs to given the selected detection threshold. When you select option *$\geq$ the Threshold*, given each detection with amplitude ***above*** the threshold, the probability of a true positive is also known as the ***positive predictive value*** (PPV), and the probability of a false positive detection is also known as the ***false discovery rate*** (FDR). The conditional probabilities correspond to the areas under the distribution curves to the right of the detection threshold line. When you select option *= the Threshold* instead, given each detection with amplitude ***equal to*** the threshold, the probability of each category that the detection belongs to is shown. The conditional probabilities correspond to the length proportions of each category along the detection threshold line in the distribution plot.

Results:

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

2.4 Example data

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

After loading it as a *"MAT-file"* into Matlab workspace, run the following line

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

to scale it from digital signal to microvolts. The variable *LFP_seg* should be selected as the *LFP Data* in step 1.

The sampling frequency was 1000 Hz, also provided as the variable *fs*.

Gamma oscillation was present in this data. Oscillation frequency range of 30 to 80 Hz is recommended for characterizing the PSD.

This data was the example case of gamma oscillation illustrated in the paper, corresponding to Fig.1, 2 and 3.

3 Video instructions

TBA. We will be adding link to a video which will walk the user through an example case.

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 unit 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.

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

procedure 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

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.

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 $I = \{i_1, \ldots, i_m\}$ 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
  $\alpha, \beta, J \leftarrow$ 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*
$\alpha, \beta, J \leftarrow$ 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

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.

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.

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

4.1.3 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

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

where $\sigma_i = (2\sqrt{2}\pi D_i)^{-1}$ and $\mathrm{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt$ 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 TPR = TP/(TP + FN) and FPR = FP/(TN + 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 $J = \text{TPR} - \text{FPR}$ 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 TPR = 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 happens in the ripple case, but not the gamma or beta cases 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 as 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 Documentation

The source code can be found in the Github repository `https://github.com/chenziao/Matlab_Tool-Design_of_Brainwaves_Detection.git`. It contains a MATLAB App installer *"Design of Brainwaves Detection.mlappinstall"* and the source codes are in the folder *"source"*.

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.
2. *"AmpDistParamOptimization_example.m"* - Step 2 in the GUI.
3. *"Synthetic_Analysis_example.m"* - Step 3 in the GUI.

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.