Processing Data How-To

Nicholas Garcia

2017-07-20

1 Save LabChart Data

The first step is to save the data recorded in LabChart to a MatLab compatible .mat file. Remove all of the blocks that don't contain relavent data, and saveas a MatLab file. You will be given the choice of which channels to export. Only two channels are needed: the AC-coupled intracellular trace (as of writing channel 4), and the AC-coupled extracellular trace (as of writing channel 2). Move this to a folder in your MatLab path. In addition, make sure the scripts run in later steps are in your path.

2 Convert Data

The next step is to convert the data to a usable form. Open the script named $file\_processing\_v3.m$. Fill in the user configured variables: the channel id's for the intracellular and extracellular traces, the threshold value to be used for the intracellular trace, and the name of the file you just saved. If you do not have intracellular data, use the same channel id for the intracellular channel variable. Once you run the script there will be a file named as the file you specified, with *_processed* on the end. This file contains a 1xn *data* vector of all of the extracellular blocks for analysis in Wave_clus, and a cell array with all of the data for a block on each row (see script $file\_converter$ or $intracellular\_processing$ for details).

3 Run Wave_clus

Make sure the *wave_clus* folder is in your MatLab path, and run $wave\_clus.m$. If you'd like to change any of the parameters you can change them in $set\_parameters.m$ (NOTE: this should be done before launching Wave_clus). Load the processed file using the appropriate button, and adjust the temperature until you're satisfied. You can then save the clusters using the *save_clusters* button. This save the file as $time\_ < file\_name >$, where file_name is the name of the file you loaded into wave_clus. The only relavent contents of the file is the *cluster_class* variable, which contains the cluster id and time of each spike (in ms).

4 Processing the $Wave\_clus$ Cluster Data

The next step is to process the *cluster_class* variable. We want to separate each cluster into it's own place, convert the ms data to indices, and group the spikes of each cluster by block. Open *convert_clusters_v2.m* and *process_clusters_v4.m*. Put the name of the *wave_clus* output file into the appropriate place in the *convert_clusters* script, and run the program. Then put the name of the first processed file (from step 2) into the *process_clusters* script, and run it.

5 Graph The Results

Now we can open *quick_graph_results_v2.m*, and fill in the file name of the processed file, the cluster, and the block we'd like to look at. Then run the script!