

Advances in the PANDORA Matlab Toolbox for neural database analysis

Cengiz Günay

cgunay@ggc.edu

School of Science and Technology, Georgia Gwinnett College
(previous versions developed at Emory University)



EMORY
UNIVERSITY

CNS*2020 Software Showcase #3
July 18, 2020

- 1 Introduction to Pandora
- 2 Analyzing voltage trace data
 - Loading a membrane voltage trace
 - Analyzing a membrane voltage trace
- 3 Database analysis with Pandora
 - Creating a database from arbitrary data
 - Creating a database from analysis of traces
 - Multivariate analysis with database objects
- 4 Links

Why should I use the Pandora toolbox?

Use it if:

- ① You are already using Matlab
- ② Python and other environments are too complex or unsustainable in your lab

Why should I use the Pandora toolbox?

Use it if:

- ① You are already using Matlab
- ② Python and other environments are too complex or unsustainable in your lab

Seriously, why would anyone still use Matlab?

- ① If you're just starting in computational neuroscience, probably the best option is using Python (Jupyter notebooks, etc)
- ② However, even though Python and its modules have improved considerably, they still require a bit of maintenance
- ③ Matlab still has its audience in non-programmer, scientist communities (e.g. experimentalists)
- ④ Also many researchers can't quit Matlab because of inherited legacy code

Main features of the Pandora toolbox

Pandora has several independent features, which also work well together:

- ① Extracting electrophysiological properties from intracellular recordings
 - Can **find spikes** from a membrane voltage trace using multiple methods
 - Frequency **filtering** of data (lowpass, bandpass, highpass)
 - **Finding bursts** and analyzing their properties
 - You can **add any other custom measurement** yourself
 - Made to **process large number of files** and produce uniform database output

Main features of the Pandora toolbox

Pandora has several independent features, which also work well together:

- ① Extracting electrophysiological properties from intracellular recordings
 - Can **find spikes** from a membrane voltage trace using multiple methods
 - Frequency **filtering** of data (lowpass, bandpass, highpass)
 - **Finding bursts** and analyzing their properties
 - You can **add any other custom measurement** yourself
 - Made to **process large number of files** and produce uniform database output
- ② Analysis of model or experimental data using a *Dataframe*-like objects
 - Creating a database from any data for **querying** and **plotting**.
 - Putting results from analysis of voltage traces of multiple models into a database.
 - Advanced operations with a database: **statistics, multivariate analysis**, etc.

Main features of the Pandora toolbox

Pandora has several independent features, which also work well together:

- ① Extracting electrophysiological properties from intracellular recordings
 - Can **find spikes** from a membrane voltage trace using multiple methods
 - Frequency **filtering** of data (lowpass, bandpass, highpass)
 - **Finding bursts** and analyzing their properties
 - You can **add any other custom measurement** yourself
 - Made to **process large number of files** and produce uniform database output
- ② Analysis of model or experimental data using a *Dataframe*-like objects
 - Creating a database from any data for **querying** and **plotting**.
 - Putting results from analysis of voltage traces of multiple models into a database.
 - Advanced operations with a database: **statistics, multivariate analysis**, etc.
- ③ Improved plotting functions
 - Matlab's plotting functions are augmented
 - Can **stack subplots** that share same axes
 - **Control spacing** between subplots
 - Render plots based on export size to produce **publication-quality figures**

Newer features of the Pandora toolbox

① Simple model simulation and parameter fitting

- Can simulate simple neuronal structures such as single **ion channels, passive membrane**, etc
- Useful for fitting responses from **voltage and current clamp protocols**
- For instance, you can **compensate for series resistance artifacts**
- Packaged separately as param-fitter, but depends on Pandora objects

Newer features of the Pandora toolbox

① Simple model simulation and parameter fitting

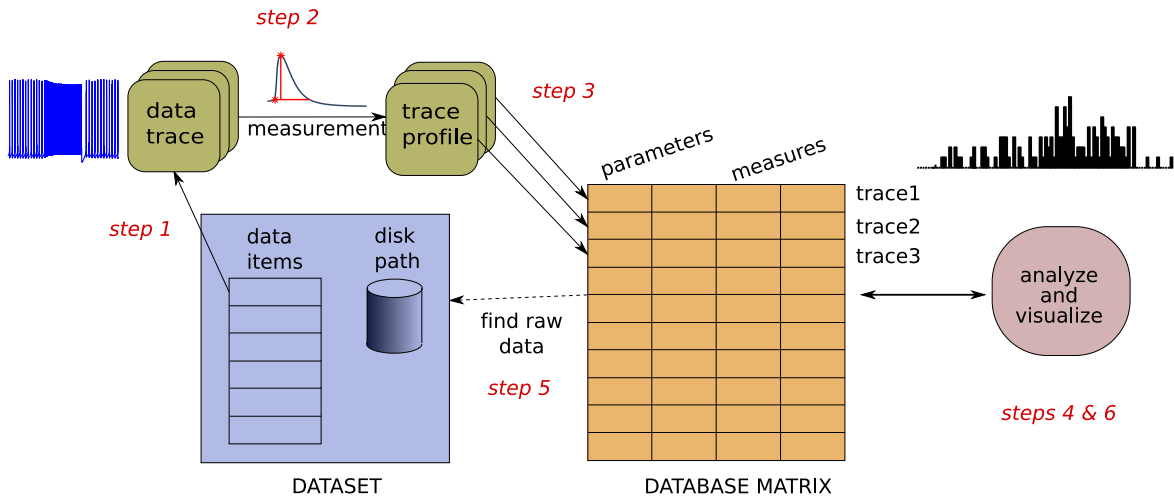
- Can simulate simple neuronal structures such as single **ion channels, passive membrane**, etc
- Useful for fitting responses from **voltage and current clamp protocols**
- For instance, you can **compensate for series resistance artifacts**
- Packaged separately as param-fitter, but depends on Pandora objects

② Model simulation parameter optimization

- Uses the GODLIKE toolbox that can run **multiple optimization algorithms** (multi-objective evolutionary algorithms, swarm, ...)
- Can **control running simulations** by calling an external simulator like Neuron, GENESIS, etc)
- Experimental feature, not included in main Pandora package

See [Günay et al. \(2009\) *Neuroinformatics*](#); also [Github](#) and [Mathworks File Exchange](#) pages.

The basic Pandora workflow



- 1 Introduction to Pandora
- 2 Analyzing voltage trace data
 - Loading a membrane voltage trace
 - Analyzing a membrane voltage trace
- 3 Database analysis with Pandora
 - Creating a database from arbitrary data
 - Creating a database from analysis of traces
 - Multivariate analysis with database objects
- 4 Links

Loading a membrane voltage trace

Pandora can read the file formats from:

① Simulators:

- Neuron
- Genesis

② Data acquisition programs:

- All NeuroShare-compatible acquisition devices (Alpha Omega, Cambridge Electronic Design, NeuroExplorer, Plexon, R.C. Electronics Inc., Tucker-Davis Technologies, and Cyberkinetics Inc., etc.)

③ Other:

- Hierarchical data format (HDF5)

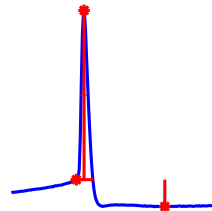
Tutorial demo on Github

Analyzing a membrane voltage trace

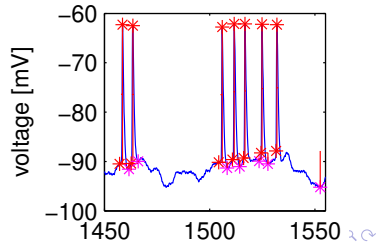
By extracting electrophysiological characteristics

- Measure spike shape and firing rate properties
- Measure sag, spike adaptation and current response properties
- Can be done repetitively for a large number of models
- Can be entered into a Matlab database

Tutorial demo on Github



annotated spike characteristics



- 1 Introduction to Pandora
- 2 Analyzing voltage trace data
 - Loading a membrane voltage trace
 - Analyzing a membrane voltage trace
- 3 Database analysis with Pandora
 - Creating a database from arbitrary data
 - Creating a database from analysis of traces
 - Multivariate analysis with database objects
- 4 Links

Creating a database from arbitrary data

Create a 2×2 database matrix:

	col1	col2
row1	1	2
row2	3	4

Creating a database from arbitrary data

Create a 2×2 database matrix:

	col1	col2
row1	1	2
row2	3	4

With Matlab code:

```
>> db_obj =  
    tests_db([1 2; 3 4],  
            {'col1', 'col2'},  
            {'row1', 'row2'}, 'a 2x2 DB')
```


Creating a database from arbitrary data

Create a 2×2 database matrix:

	col1	col2
row1	1	2
row2	3	4

With Matlab code:

```
>> db_obj =  
    tests_db([1 2; 3 4],  
             {'col1', 'col2'},  
             {'row1', 'row2'}, 'a 2x2 DB')
```

Can also import text files as database (e.g., Excel export).

Creating a database from analysis of traces

Using a dataset:

```
>> my_dataset_obj =  
    my_dataset_class('data/*.bin', arguments...)
```

Creating a database from analysis of traces

Using a dataset:

```
>> my_dataset_obj =  
    my_dataset_class('data/*.bin', arguments...)  
  
>> my_database_obj =  
    param_tests_db(my_dataset_obj)
```

Creating a database from analysis of traces

Using a dataset:

```
>> my_dataset_obj =  
    my_dataset_class('data/*.bin', arguments...)  
  
>> my_database_obj =  
    param_tests_db(my_dataset_obj)  
  
>> sorted_obj =  
    sortrows(my_database_obj, 'AP_amplitude')
```

Database analysis: Querying

```
>> db_obj2 =  
    db_obj(1:10, {'neuron_index', 'fire_rate'})
```

Database analysis: Querying

```
>> db_obj2 =  
    db_obj(1:10, {'neuron_index', 'fire_rate'})  
>> db_obj2 =  
    db_obj(db_obj(:, 'neuron_index') == 46, :)
```

Database analysis: Querying

```
>> db_obj2 =  
    db_obj(1:10, {'neuron_index', 'fire_rate'})  
>> db_obj2 =  
    db_obj(db_obj(:, 'neuron_index') == 46, :)  
>> db_obj2 =  
    db_obj(anyRows(db_obj(:, 'neuron_index'),  
                    [46; 56; 12])), :)
```

Database analysis: Querying

```
>> db_obj2 =  
      db_obj(1:10, {'neuron_index', 'fire_rate'})  
>> db_obj2 =  
      db_obj(db_obj(:, 'neuron_index') == 46, :)  
>> db_obj2 =  
      db_obj(anyRows(db_obj(:, 'neuron_index'),  
                     [46; 56; 12])), :)  
>> db_obj2 =  
      db_obj(db_obj(:, 'neuron_index') ~= 46 &  
             (db_obj(:, 'CIP') > 100 |  
              db_obj(:, 'rate') <= 50 ), :)
```

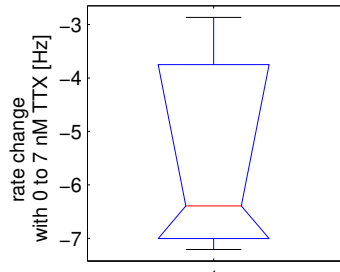
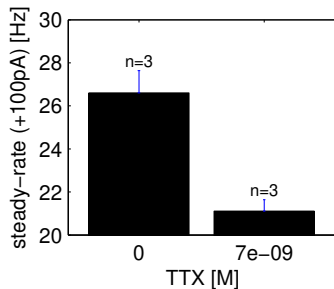
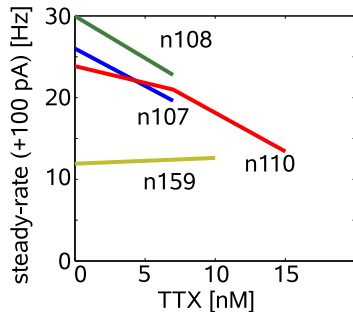

Database analysis: Querying

```
>> db_obj2 =  
    db_obj(1:10, {'neuron_index', 'fire_rate'})  
>> db_obj2 =  
    db_obj(db_obj(:, 'neuron_index') == 46, :)  
>> db_obj2 =  
    db_obj(anyRows(db_obj(:, 'neuron_index'),  
                    [46; 56; 12])), :)  
>> db_obj2 =  
    db_obj(db_obj(:, 'neuron_index') ~= 46 &  
            (db_obj(:, 'CIP') > 100 |  
             db_obj(:, 'rate') <= 50 ), :)  
>> db_obj2 =  
    model_db_obj(anyRows(model_db_obj(:, 'rate'),  
                          neuron_db_obj(:, 'rate'))), :)
```

Tutorial demo on Github

Multivariate analysis with non-grid data

Tetrodotoxin block effects on firing rate of globus pallidus neurons with current injection



Multivariate analysis (I)

Sifting the database to find effects of parameters

Sample with 3 Neurons:

PicroTx	0.0001	0.0001	0.0001	0.0001	0.0001	0.0001
KynAcid	0.001	0.001	0.001	0.001	0.001	0.001
TTX	0	$7e-09$	0	$7e-09$	0	$7e-09$
Apamin	0	0	0	0	0	0
drug 4AP	0	0	0	0	0	0
NeuronId	107	107	108	108	110	110
D100pA steady rate	25.9982	19.6056	29.9673	22.7628	23.8443	20.9744

Focus on changes with TTX:

	Page 1		Page 2		Page 3	
TTX	0	$7e-09$	0	$7e-09$	0	$7e-09$
D100pA steady rate	25.9982	19.6056	29.9673	22.7628	23.8443	20.9744
RowIndex	1	2	3	4	5	6

Multivariate analysis (II)

Processing database contents

Change in rate (Δ) between successive TTX levels:

d1_2	-6.3926	-7.2045	-2.8699
PageIndex	1	2	3

Regrouping to find average values for each TTX level:

	Page 1			Page 2		
TTX	0	0	0	$7e-09$	$7e-09$	$7e-09$
D100pA steady rate	25.9982	29.9673	23.8443	19.6056	22.7628	20.9744
RowIndex	1	3	5	2	4	6

DEMO

- 1 Introduction to Pandora
- 2 Analyzing voltage trace data
 - Loading a membrane voltage trace
 - Analyzing a membrane voltage trace
- 3 Database analysis with Pandora
 - Creating a database from arbitrary data
 - Creating a database from analysis of traces
 - Multivariate analysis with database objects
- 4 Links

Testing and feedback

How to access Pandora:

- Main publication: [Günay et al. \(2009\) *Neuroinformatics*](#)
- Downloads and documentation on [Github](#) and [Mathworks File Exchange](#) pages

How to give feedback/ask questions:

- Open issues on [Github](#)
- Email: cgunay AT ggc.edu
- Fill survey (TBA)

- 1 Introduction to Pandora
- 2 Analyzing voltage trace data
 - Loading a membrane voltage trace
 - Analyzing a membrane voltage trace
- 3 Database analysis with Pandora
 - Creating a database from arbitrary data
 - Creating a database from analysis of traces
 - Multivariate analysis with database objects
- 4 Links