Computational Lab Notebook

AdEx implementation in NetPyNe

Javier Palma Espinosa

Last Revision: 2022-03-01

Contents

1 Information

1.1 Project name

Adaptative Exponential Integrate and Fire (AdEx) implementation in Netpyne

1.2 Project date

This project started in June 2021

1.3 Motivation

The Kerr Lab implemented a previous version of the AdEx model. The basic idea here was to implement an AdEx Class.

Should netpyne provide a basic NEURON models, such as AdEx, izhikevic, integrate and fire?

Does NEURON provide those models?

To solve those questions, I propose to implement the AdEx model in NEURON and then, plug it into NetPyNe, as a new functionality.

1.4 Publications

1. NetPyNe Paper (Dura-Bernal et al., 2019)

1.5 Digital verification

not yet

1.6 Links

- The NetPyNe project is available at http://www.netpyne.org/
- Dura-Bernal Laboratory could be reached at http://dura-bernal.org/
- The whole project is archived in github as digital repository. It may be found in the following link: https://github.com/jpalma-espinosa/netpyne

1.7 Additional Credits

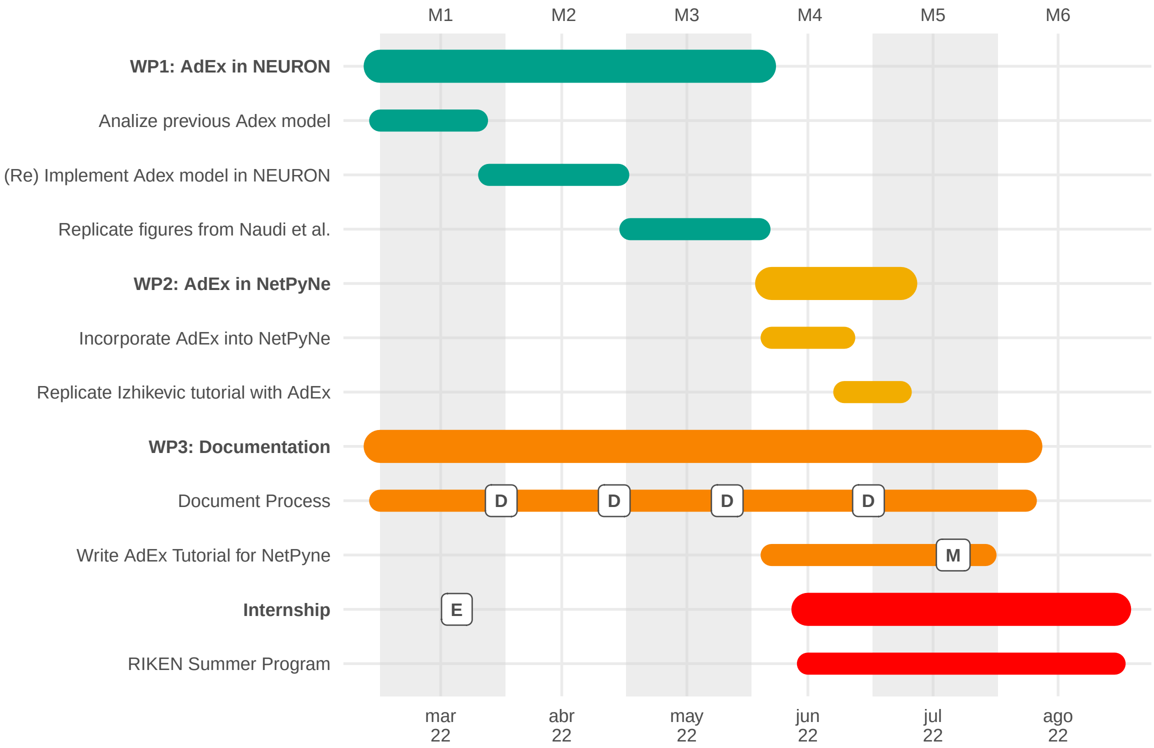
- This document was done with the Eisvogel Template, by Pascal Wagler

2 Changelog

Date	Commit	log
2022-03-01 05:32:33	6206ebe	Lab notebook actualization and project folder structure improved.
2022-02-27 02:48:10	c4bc781	added gantt chart to labNotebook
2022-02-26 20:47:25	1abe955	Computational Lab Notebook File updated
2022-02-26 20:29:36	cb63c70	update in project structure. New documentation available
2021-07-20 20:44:24	10a7bcb	Pseudo working example of Adex2021b
2021-07-07 05:46:44	4601b03	Changed typo in README
2021-07-07 05:35:18	be151ad	Upload new documents. Thoughts and ideas in README
2021-07-01 06:40:27	026e684	Notes name changed
2021-07-01 06:30:44	ffda310	Ball-Stick class is created (not working).

3 Work Plan (Gantt Chart)

wp	activity	start_date	end_date
WP1: AdEx in NEURON	Analize previous Adex model	22-03-01	22-03-25
WP1: AdEx in NEURON	(Re) Implement Adex model in NEURON	22-03-28	22-04-29
WP1: AdEx in NEURON	Replicate figures from Naudi et al.	22-05-02	22-06-03
WP2: AdEx in NetPyNe	Incorporate AdEx into NetPyNe	22-06-06	22-06-24
WP2: AdEx in NetPyNe	Replicate Izhikevic tutorial with AdEx	22-06-24	22-07-08
WP3: Documentation	Document Process	22-03-01	22-08-08
WP3: Documentation	Write AdEx Tutorial for NetPyne	22-06-06	22-07-29
Internship	RIKEN Summer Program	22-06-15	22-08-30



3.1 WP1: AdEx in NEURON

1. I will first *analyze the AdEx.mod file/model* based on what Kerr Lab did previously. This particular file has the characteristic that it kindda implement some Point Process that could might be useful for the Adex.
2. After the code is understood, I will re-implement the model and make some parametrization and test the firing in NEURON.
3. With the previous **working** adex model, I will replicate the different behavior that is shown in the Naud paper(Naud et al., 2008)

3.2 WP2: AdEx in NetPyNe

1. Assuming that adex.mod is working in NEURON, I will then plug it into NetPyNe. First, I will do a simple single spiking neuron. Then, I will test a network of AdEx neurons.
2. Finally, and for a proper implementation and documentation of the model, I will replicate and write a new tutorial, based on the Izhikevic one

3.3 WP3: Documentation

During the whole project, I will be writting and documenting every step. This process will be shown in this Lab Notebook.

3.4 Project's key activities

Deliverables: Montly meeting update.

Milestone: Full project presentation (labmeeting).

Event: Result from RIKEN

4 Daily Report

4.1 Feb 24th - Mar 03rd, 2022

During this week, I devoted to create this lab notebook, build the gantt chart, clean and update the project's github, and previous work wrap up.

4.2 Jul 19th, 2021

After the meeting with Salvador, on Jul 6th, and by following his advices, I replicated what was developed in the izhikevich model. In particular the b part. Briefly, the models could be sumarized as:

Characteristic	Izhi2003a	Izhi2003b	Izhi2007a	Izhi2007b
Kind	P.Proc.	P.Proc.	P.Proc.	P.Proc.
Section	Dummy	Regular	Dummy	Regular
Synaptic input	yes	no	yes	yes
Synaptic method	$g'_{syn} = -g_{syn}/\tau_g$	—	AMPA/NMDA/GABA dynamics	cell dependent
Implemented in Netpyne	no	no	yes	no

I focused on replicating the Izhi2007b.

Results:

1. I was able to build and compile Adex2021b (I am keeping the name scheme).
2. I was able to replicate the Izhikevic tutorial, but now using Adex (adex.ipynb).

Drawbacks:

1. my neuron does not fire, even further, I get an error

```
See http://neuron.yale.edu/neuron/credits

loading membrane mechanisms from
 ↪  /home/javier/Neuroscience/netpyne/AdEx/x86_64/.libs/libnrnmech.so
Additional mechanisms from files
 "./mod/adex.mod" "./mod/izhi2007b.mod"
```

```
nrniv: unable to open font "*helvetica-medium-r-normal*--14*", using "fixed"
oc>     -65
Segmentation violation
Backtrace:
terminate called after throwing an instance of 'std::regex_error'
  what():  regex_error
Aborted (core dumped)
```

I need to debug the .mod file, but I don't know how

4.3 Jul 6th, 2021

I only read a couple of documents from Neuron tutorial and from a MIT tutorial on Neuron. The important part here was to examinate how to properly define the puntual neuron AdEx. It seems that my model needs to considerate an external current *FROM* an external point mechanism. For this, I will need to re-study the integrate and fire model that is proposed in the Neuron Github page

4.4 Jun 30th, 2021

I was on halt because I had to deal with my master thesis. I am now a Master of Science :D.

Because the previous implementation wasn't sucessful, I asked wheter AdEx should be defined as a mechanism or a point neuron (see De Schutter book, Ch. 7). The way that NEURON is implemented, makes logical to define AdEx as a point process and define it as ARTIFICIAL_CELL. To do this, I have to understand how `NET_RECEIVE(w)` process works.

This code block is better defined in the Neuron Book (Ch. 10)

4.5 Jun 7th, 2021

I was able to run the izhikevic tutorial. Also, I wrote the Adex.mod file, by replicating what was done with izhi2007b.mod. However, I am still not able to produce a spike in the Adex model. The izhikevic one has some strange way of calculating the derivative states. What is the difference between those two forms of calculation? Also, how can I incorporate the synapses in the Adex neuron?

The izhikevic (and adex) is implemented as a POINT PROCESS (see also NEURON documentation), contrary to the HH model.

5 Preliminary Results

6 Meetings

6.1 Thursday, 23th feb. 2022

- **Hour:** 18:00 - 18:20 GMT-03
- **Reason:** Project's redefinition and continuity.

This meeting was mainly to update Salvador about my performance on the last semester. I explained him why I set aside this project, what were my interests and motivations. We decided that I should continue with this idea.

In order to do it, I commited to come along (on Thu, 03rd march) with a gantt chart and a proper task definition for achieving the project. The gantt chart is on the workplan page

7 Computing tools, code snippets, and tips.

7.1 CODE: Fast prototipying in code (2021-12-27)

One of my biggest mistakes is to try to build, at first, a very complicated piece of software, which is amendable for programmer, but also efficient as hell. **I must focus on building a working piece of software and then improve it!**

7.2 CODE: git store credential

general formula:

```
// local
git config credential.helper store
// global
git config --global credential.helper store
```

```
$ git config credential.helper store
$ git push http://example.com/repo.git
Username: <type your username>
Password: <type your password>
```

Notice that **Password** is the code obtained from the github access token

several days later

```
$ git push http://example.com/repo.git
[your credentials are used automatically]
```

8 Papers summary

References

Dura-Bernal, S., Suter, B. A., Gleeson, P., Cantarelli, M., Quintana, A., Rodriguez, F., et al. (2019). NetPyNE, a tool for data-driven multiscale modeling of brain circuits. *Elife* 8, e44494.

Naud, R., Marcille, N., Clopath, C., and Gerstner, W. (2008). Firing patterns in the adaptive exponential integrate-and-fire model. *Biological cybernetics* 99, 335–347.