

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ імені ІГОРЯ СІКОРСЬКОГО»
НАУКОВО-НАВЧАЛЬНИЙ ФІЗИКО-ТЕХНІЧНИЙ ІНСТИТУТ
КАФЕДРА ПРИКЛАДНОЇ ФІЗИКИ

На правах рукопису
УДК 57.085.23

«До захисту допущено»

Завідувач кафедри

_____ Г. Є. Монастирський
(підпис) (ініціали, прізвище)

“ ____ ” _____ 2022 р.

Магістерська дисертація
на здобуття ступеня магістра

за спеціальністю _____ 105 «Прикладна фізика та наноматеріали»
(код і назва)

на тему: Моделювання динаміки імпульсної активності нейронів

Виконав: студент 6 курсу, групи ФФ-01мн

_____ Руцький Костянтин Валерійович
(прізвище, ім'я, по батькові) (підпис)

Керівник: проф., д.б.н. Лук'янець О.О.
(посада, науковий ступінь, вчене звання, прізвище та ініціали) (підпис)

Рецензент: ст.наук. співр., канд. біол. наук Яворський В. А.
(посада, науковий ступінь, вчене звання, прізвище та ініціали) (підпис)

Засвідчую, що у цій магістерській дисертації
немає запозичень з праць інших авторів без
відповідних посилань.

Студент _____
(підпис)

Міністерство освіти і науки України
Національний Технічний Університет
України
«Київський політехнічний інститут імені Ігоря Сікорського»
Фізико-технічний інститут
Кафедра прикладної фізики

Рівень вищої освіти – другий (магістерський)
Спеціальність 105 Прикладна фізика та
наноматеріали

«ЗАТВЕРДЖЕНО»

Завідувач кафедри

_____ Г. Є. Монастирський
(підпис) (ініціали, прізвище)

«___» _____ 2022 р.

ЗАВДАННЯ

на магістерську дисертацію студенту

_____ Руцький Костянтин Валерійович _____

(прізвище, ім'я, по батькові)

(підпис)

1. Тема: Моделювання динаміки імпульсної активності нейронів

Керівник _____ проф., д.б.н. Лук'янець О.О.

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по університету « » травня 2022 р. № 1414-с.

2. Термін подання студентом дисертації
3. Об'єкт дослідження: нейронні моделі; модель Ходжкіна-Хакслі
4. Предмет дослідження: імплементація моделей динаміки імпульсної активності нейронів за допомогою мови програмування Python
5. Перелік завдань, які потрібно розробити: створити код, який базується на нейронних моделях, і симулює їхню нейронну динаміку представити цю динаміку у графіках
6. Орієнтовний перелік ілюстративного матеріалу: 15 рисунків
7. Консультанти розділів дисертації

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання видав
1	Ст. наук. співр., канд. біол. наук Яворський В.А.		
2	Ст. наук. співр., канд. біол. наук Яворський В.А.		
3	Ст. наук. співр., канд. біол. наук Яворський В.А.		

1. Дата видачі завдання

Календарний план

№ з/п	Назва етапів виконання дипломної роботи	Термін виконання етапів магістерської дисертації	Примітка
1.	Ознайомлення з типами нейронних моделей	31.01.2022 – 06.02.2022	
2.	Детальний аналіз нейронної моделі Ходжкіна-Хакслі	07.02.2022 - 13.02.2022	
3.	Спостереження за реальною нейронною динамікою, появою потенціалів дії	14.02.2022- 18.04.2022	
4.	Ознайомлення з бібліотекою Brian2 для мови програмування Python	18.02.2022- 20.04.2022	
5.	Написання коду для першої нейронної моделі	21.02.2022-27.02.2022	
6.	Огляд літературних джерел та написання першого розділу магістерської дисертації	28.02.2022-06.03.2022	
7.	Написання коду для моделі Ходжкіна-Хакслі та написання основної частини магістерської роботи	10.04.2022-01.06.2022	
8.	Написання висновків	02.06.2022-07.06.2022	
9.	Подання роботи на рецензування	08.06.2022-10.06.2022	
10.	Підготовка доповіді та презентації	10.06.2022-12.06.2022	
11.	Подання роботи до захисту	12.06.2022-13.06.2022	

Студент

████████████████████

(підпис)

К. В. Руцький

(ініціали, прізвище)

Науковий керівник дисертації

████████████████████

(підпис)

О. О. Лук'янець

(ініціали, прізвище)

РЕФЕРАТ

Магістерська дисертація має обсяг 63 сторінки, 18 рисунків і 3 додатки окрім цього використано 20 бібліографічних джерел.

Створено симуляції динаміки імпульсної активності нейронів декількох видів.

Метою роботи є розробка симуляцій які демонструють нейронну динаміку моделей нейронів.

У ході експерименту було розроблено код, який симулює нейронну динаміку моделей, зокрема моделі Ходжкіна-Хакслі.

Ключові слова: нейрон, нейронна динаміка, потенціал дії, потенціал спокою, модель Ходжкіна-Хакслі.

ABSTRACT

Master's thesis is presented in 63 pages, 18 figures, 3 appendixes in addition, 20 bibliographic sources were used.

Simulations of dynamics of impulse activity of neurons of several types are created.

The aim of the work is to develop simulations that demonstrate the neural dynamics of neuronal models.

During the experiment, code was developed that simulates the neural dynamics of models, in particular the Hodgkin-Huxley model.

Key words: neuron, neural dynamics, action potential, rest potential, Hodgkin-Huxley model.

ЗМІСТ

Перелік умовних позначень, символів, одиниць, скорочень і термінів	7
Вступ	8
Розділ 1 Літературний Огляд	9
1.1 Модель Ходжкіна-Хакслі	11
1.2 Моделі нейронів типу I та типу II	17
1.3 Узагальнені нейрони Integrate-and-Fire (I-F)	18
1.4 Модель Іжикевича	22
1.5 Варіативність послідовностей потенціалів дії і нейронне кодування	23
1.6 Середня швидкість збудження	26
1.7 Інтервальний розподіл і коефіцієнт варіації	28
1.8 Проблема нейронного кодування	29
Розділ 2 Матеріали I Методи Досліджень	37
2.1 Моделювання нейронної динаміки	37
Розділ 3 Результати Досліджень	40
3.1 Симуляція експоненційної моделі «Leaky Integrate-and-Fire»	40
3.2 Симуляція адаптивної експоненційної моделі	42
3.3 Симуляція моделі Ходжкіна-Хакслі	44
Висновки	49
Перелік джерел посилань	50
Додаток А	53
Додаток Б	56
Додаток В	58

**ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ,
СКОРОЧЕНЬ І ТЕРМІНІВ**

ПД – потенціал дії.

I-F – модель “Integrate-and-Fire”

LI-F – модель “Leaky Integrate-and-Fire”

ГАМК – γ -аміномасляна кислота

GABA_A і GABA_B – перший і другий типи гальмівних синапсів

AdEx – адаптивна експоненційна модель

ISI - розподіл інтервалу між потенціалами дії

ВСТУП

Однією з найважливіших функцій біологічних мембран є генерація та передача біологічних потенціалів. Це є основою регуляції внутрішньоклітинних процесів, збудливості клітин, роботи нервової системи, регуляції м'язового скорочення і рецепції. Мембранним потенціалом називають різницю потенціалів між цитоплазматичною та зовнішньою поверхнями мембрани. Мембранні потенціали бувають стаціонарні та потенціали збудження. До стаціонарних потенціалів відноситься потенціал спокою, до потенціалів збудження – потенціал дії. Потенціал спокою це стаціонарна різниця електричних потенціалів, що реєструється між внутрішньою та зовнішньою поверхнями мембрани у стані спокою. У тварин він визначається різницею концентрацій іонів по різні боки мембрани та напівпроникними властивостями мембрани, що призводять до дифузії певних іонів через мембрану. Якщо концентрація іона всередині клітини відрізняється від концентрації іона зовні і мембрана проникна для цього іона, виникає потік заряджених частинок через мембрану, внаслідок чого порушується електрична нейтральність системи, утворюється різниця потенціалів всередині та зовні клітини, яка перешкоджатиме подальшому переміщенню іонів через мембрану. При встановленні рівноваги вирівнюються значення електрохімічних потенціалів з різних боків мембрани.

РОЗДІЛ 1

ЛІТЕРАТУРНИЙ ОГЛЯД

З біофізичної точки зору потенціали дії є результатом струмів, які проходять через іонні канали в клітинній мембрані.

Нейронну динаміку можна уявити як процес підсумовування, або інколи його називають «процес інтеграції» у поєднанні з механізмом, який запускає потенціали дії вище деякої критичної напруги. В експериментах час збудження часто визначають як момент, коли потенціал мембрани досягає деякого порогового значення знизу. Для побудови феноменологічної моделі нейронної динаміки, описують критичну напругу для ініціації потенціалу дії формальним порогом ϑ . Якщо напруга $u_i(t)$ (яка містить підсумований ефект усіх входів) досягає ϑ знизу, говорять, що нейрон i запускає потенціал дії. Момент перетину порогу визначає час випалювання $t_i(f)$ [1].

Модель використовує той факт, що потенціали дії нейрона завжди мають приблизно однакову форму. Якщо форма потенціалу дії завжди однакова, то форма не може бути використана для передачі інформації: скоріше інформація міститься в наявності чи відсутності потенціалу дії. Тому потенціали дії зводяться до «подій», які відбуваються в певний момент часу.

Нейронні моделі, де потенціали дії описуються як так звані події називаються моделями «Integrate-and-Fire»(I-F). В таких моделях не робиться жодних спроб описати форму потенціалу дії. Такі моделі мають дві окремі компоненти, які необхідні для визначення їхньої динаміки: по-перше, рівняння, яке описує зміну з часом мембранного потенціалу $u_i(t)$; по-друге, механізм який генерує потенціали дії [4].

Найпростіша модель у класі моделей I-F побудована на наступних складових:

- лінійне диференційне рівняння для опису зміни з часом мембранного потенціалу;

- поріг для збудження потенціалу дії.

Ця модель називається моделлю «Leaky Integrate-and-Fire».

Вона дуже спрощена і нехтує багатьма аспектами нейронної динаміки. Наприклад вхідним сигналом, який може виникнути від пресинаптичних нейронів або від ін'єкції струму, інтегрується лінійно, незалежно від стану постсинаптичного нейрона. Крім того, після кожного вихідного стрибка мембранний потенціал скидається, щоб не збереглася пам'ять про попередні сплески [4].

Проста LI-F не враховує адаптацію. Однак, якщо динаміка напруги посилюється механізмами адаптації, то це може бути потужним інструментом для точного прогнозування часу стрибків кортикальних нейронів.

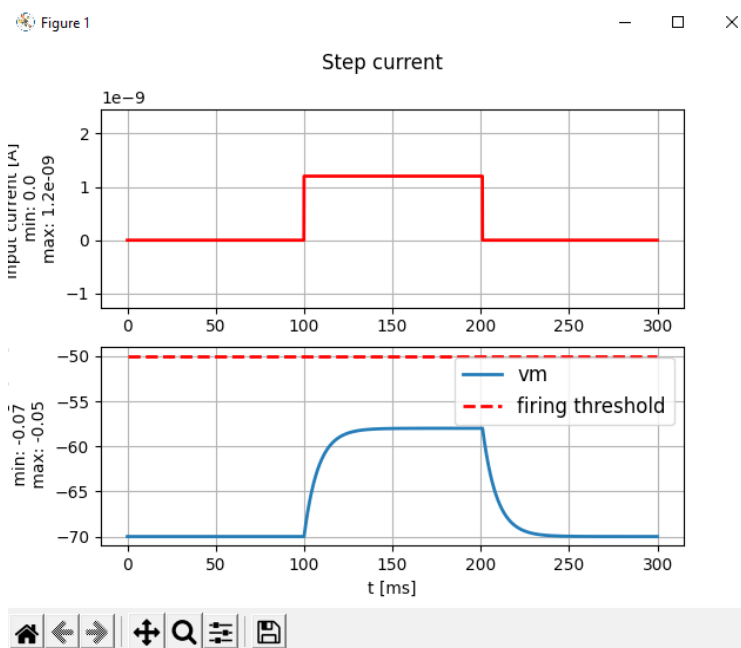


Рис. 1.1 – Базова модель типу «Leaky Integrate-and-Fire»

1.1 Модель Ходжкіна-Хакслі

У великій серії експериментів над гігантським аксоном кальмара Ходжкіну і Хакслі вдалося виміряти ці струми і описати їх динаміку в термінах диференціальних рівнянь. Вони опублікували статтю, де був представлений експеримент, а також представлена математична теорія, яка стала проривною в своїй галузі і за що вони отримали в кінцевому рахунку Нобелівську премію.

У своїх роботах Ходжкін і Хакслі показали, що провідність селективних каналів визначається станом їх так званих воріт, що активують і інактивують. Ворота, що активують, поступово відкриваються при підвищенні мембранного потенціалу, а інактивуючі ворота, навпаки, при цьому закриваються [10].

За припущенням Ходжкіна та Хакслі, динаміка активаційної та інактиваційної змінних описується диференціальними рівняннями першого порядку

$$dx / dt = (x_{st}(V) - x) / t_{st}(V) \quad (1)$$

x – вектор змінних, x_{st} , t_{st} – стаціонарні стани змінних і час релаксації процесів, які в загальному випадку залежать від величини мембранного потенціалу.

Ходжкін та Хакслі встановили, що головну роль у процесі іонного транспорту відіграють іони натрію та калію. Саме натрієві та калієві канали детально описуються у запропонованій ними моделі, а внесок інших іонів характеризується загальним струмом витоку. Ходжкін і Хакслі експериментально показали, що для натрієвих каналів характерна наявність трьох активуючих та одних воріт, що інактивують ($p_{Na} = 3$, $q_{Na} = 1$), а для калієвих – наявність чотирьох активуючих воріт на канал ($p_K = 4$, $q_K = 0$). Дані числа підбираються емпірично так, щоб найкраще описати кінетику струмів. Функції $x_{st}(V)$ і $t_{st}(V)$ також підбираються емпірично. В результаті модель Ходжкіна-Хакслі набуває наступного вигляду [10]:

$$C \frac{du}{dt} = - \overbrace{g_{Na} m^3 h (u - E_{Na})}^{I_{Na}} - \overbrace{g_K n^4 (u - E_K)}^{I_K} - \overbrace{g_l (u - E_l)}^{I_{leak}} + \overset{\text{stimulus}}{\downarrow} I(t)$$

Рис. 1.2 – Математичне представлення моделі Ходжкіна-Хакслі

Чисельне інтегрування рівняння при біологічно релевантних параметрах показує, що модель адекватно відтворює динаміку мембранного потенціалу нейрона. Основною властивістю системи є генерація потенціалу дії, що відбувається згідно із законом «все або нічого».

Нейрони мають складну морфологію: центральною частиною клітини є сома, яка містить генетичну інформацію та велику частину молекулярного механізму. На сомі виникають розширення двох типів. Перші – дендрити; утворюють безліч різних за розміром гілок, на яких розташовані синапси. Синапси є контактними точками, куди надходить інформація від інших нейронів. Другий — аксон, теж починається з соми, який нейрон використовує для передачі потенціалів дії до своїх цільових нейронів. Традиційно вважається, що перехідна область між сомою і аксоном є вирішальною областю, де приймається рішення, посиляється потенціал дії чи ні.

Модель Ходжкіна-Хакслі, принаймні у своїй формі, представлений в загальному випадку, ігнорує цю просторову структуру і зводить нейрон до точкового генератора потенціалів дії – незважаючи на той факт, що точне просторове розташування нейрона потенційно може бути важливим для процесингу сигналу в мозку [10].

Третій тип іонних каналів — це іонні канали, що активуються передавачами, які беруть участь у синаптичній передачі і зазвичай активуються ззовні клітини. Активація пресинаптичного нейрона призводить до вивільнення нейромедіаторів у синаптичну щілину. Молекули-передавачі дифундують на інший бік щілини і активують рецептори, які розташовані в постсинаптичній

мембрані. Так звані іонотропні рецептори мають безпосередній вплив на стан асоційованого іонного каналу. Метаботропні рецептори контролюють стан іонного каналу за допомогою біохімічного каскаду G-білків і вторинних посередників. В обох випадках активація рецептора призводить до відкриття певних іонних каналів і, таким чином, до збудливого або гальмівного постсинаптичного трансмембранного струму[7].

Замість того, щоб розробляти математичну модель концентрації передавачів у синаптичній щілині, можна просто описати активовані передавачем іонні канали як явно залежну від часу провідність $g_{\text{syn}}(t)$, яка відкривається щоразу, коли надходить пресинаптичний потенціал дії. Струм, що проходить через синаптичний канал, як і раніше, залежить від різниці між його реверсивним потенціалом E_{syn} і фактичним значенням мембранного потенціалу.

$$g_{\text{syn}}(t) = \sum_f \bar{g}_{\text{syn}} e^{-(t-t^{(f)})/\tau} \Theta(t - t^{(f)}) \quad (2)$$

Гальмівні синапси

Вплив швидких гальмівних нейронів у центральній нервовій системі вищих хребетних майже виключно передається нейромедіатором, який називається γ -аміномасляною кислотою, або скорочено ГАМК. Характерною особливістю гальмівних синапсів є те, що потенціал реверсії E_{syn} знаходиться в діапазоні від -70 до -75 мВ. Таким чином, якщо потенціал нейронної мембрани вище потенціалу реверсії, прихід пресинаптичного потенціалу дії призводить до гіперполяризації нейрона, що робить менш імовірним створення потенціалу дії. Однак той самий пресинаптичний потенціал дії призведе до деполяризації мембрани, якщо мембранний потенціал нейрона становить -80 мВ або нижче.

Розрізняють два основних типи гальмівних синапсів, які називаються ГАВАА і ГАВАВ. Обидва типи синапсів використовують ГАМК як нейромедіатор. Канали ГАВАА є іонотропними і відкриті виключно для хлорид-іонів, тоді як

синапси GABA_B мають метаботропні рецептори, які запускають порівняно повільний сигнальний ланцюг, що в кінцевому підсумку призводить до відкриття K⁺-каналів[2].

Збуджувальні синапси

Більшість збудливих синапсів у центральній нервовій системі хребетних покладаються на глутамат як на нейромедіатор. Однак постсинаптичні рецептори можуть мати дуже різні фармакологічні властивості, і в одному синапсі можуть бути присутні різні типи одиниць глутаматних рецепторів. Ці рецептори класифікуються за допомогою штучних препаратів, які діють як селективні агоністи. NMDA (N-метил-D-аспаратат) зв'язується з каналами з рецепторами NMDA, але не з іншими глутаматними рецепторами[2].

Швидка синаптична динаміка

Параметри точки синаптичного контакту не є фіксованими, але можуть змінюватися в залежності від історії стимуляції. Короткочасну синаптичну пластичність можна виміряти, якщо стимулювати пресинаптичний нейрон, щоб створити послідовність імпульсів дії. Синаптична полегшення означає, що видима амплітуда постсинаптичного струму у відповідь на другий потенціал дії більша, ніж на перший.

Нейрони в корі та інших областях мозку часто демонструють високорозвинені дендритні дерева, які можуть поширюватися на кілька сотень мікрон. Синаптичний вхід до нейрона здебільшого розташований на його дендритному дереві. Електричні властивості точкових нейронів були описані як конденсатор, який заряджається синаптичними струмами та іншими поперечними іонними струмами через мембрану. Нерівномірний розподіл мембранного потенціалу на дендритному дереві та сомі викликає додатковий поздовжній струм уздовж дендрита[11].

$$\frac{\partial^2}{\partial x^2} u(t, x) = c r_L \frac{\partial}{\partial t} u(t, x) + r_L \sum_{ion} i_{ion}(t, x) - r_L i_{ext}(t, x). \quad (3)$$

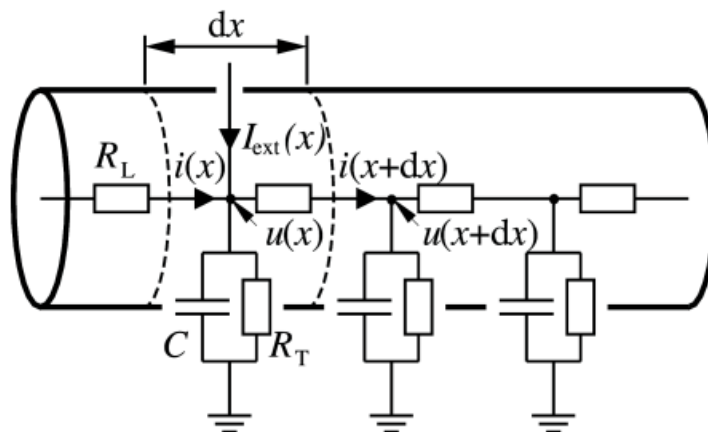


Рис.1.3 – Представлення кабельного рівняння дендритів

Будь-який нейрон має один аксон, який залишає сому для встановлення синаптичних контактів. Як і дендрити, аксони мають різноманітну морфологію. Деякі аксони проектуються в основному на нейрони поблизу. Це стосується нейронів у шарі 2-3 кори, їх аксони розгалужуються в усіх напрямках від соми, утворюючи зірчасту аксональну альтанку, яка називається «ромашка». Інші нейрони, такі як пірамідні нейрони, розташовані глибше в корі, мають аксони, які занурюються в білу речовину і можуть перетинати весь мозок, щоб досягти іншої ділянки мозку. Є ще довші аксони, які залишають центральну нервову систему і рухаються вниз по спинному мозку, щоб досягти м'язів[8].

Два типи аксонів: мієлінові та немієлінові аксони. Мієлін корисний для збільшення швидкості поширення в далекосяжних проєкціях. Це стосується кортикальних проєкцій, що проходять через білу речовину, або аксонів, що перетинають спинний мозок. З іншого боку, короткі проєкції використовують аксони, позбавлені мієліну.

Математичний опис мембранного потенціалу в аксоні ідентичний опису дендритів з активними іонними каналами. Немієлінові аксони містять натрієві та калієві канали, рівномірно розподілені по всій їх довжині.

Коли потенціал дії збуджується в початковому сегменті аксона, підвищений мембранний потенціал деполяризує сусідні аксональні сегменти. Натрієві канали далі від аксона, які раніше були закриті, почнуть відкриватися, тим

самим ще більше деполяризуючи мембрану. Потенціал дії поширюється шляхом активації натрієвих каналів уздовж кабелю, а не шляхом поширення зарядів, наприклад, у пасивному дендриті. Властивості іонних каналів сильно впливають на швидкість провідності. У немієлінових аксонах гіпокампу швидкість провідності аксонів становить 0,25 м/с.

Мієлінові аксони мають натрієві та калієві канали лише в обмежених сегментах, які називаються вузлами Ранв'є. Ці вузли утворюють лише 0,2% довжини аксона, решта вважається пасивною мембраною, яка загорнута в мієлінову оболонку. Мієлін в основному зменшує ємність мембрани C і підвищує опір R .

Іони утримуються мієліном, оскільки він не дає їм витікати за межі аксона або накопичуватися на мембрані. Замість цього іони входять і виходять з вузлів таким чином, що іон, що залишає вузол Ранв'є, змушує інший увійти в наступний вузол.

Для чисельного визначення мембранного потенціалу складного дендритного дерева використовуються компартментальні моделі, які є результатом дискретизації. Дендритне дерево розділене на невеликі циліндричні відсіки з приблизно рівномірним мембранним потенціалом. Кожен відсік характеризується своєю місткістю та поперечною провідністю. Суміжні відсіки з'єднані поздовжнім опором, який визначається їх геометричними властивостями. Після використання чисельних методів для визначення потенціалу мембрани вздовж дендритного дерева деякі або всі відсіки також можуть бути оснащені нелінійними іонними каналами. Таким чином можна вивчати ефекти нелінійної інтеграції синаптичного входу. Крім практичних проблем, які виникають через зростаючу складність основних диференціальних рівнянь, концептуальні проблеми пов'язані з різко зростаючою кількістю вільних параметрів. Щоб уникнути цих проблем, усі нелінійні іонні канали, що відповідають за генерацію імпульсів дії, зазвичай об'єднуються в сому, а дендритне дерево розглядається як пасивний кабель [12].

1.2 Моделі нейронів типу I та типу II

Нейронні моделі поділяються на два класи: моделі зі скінченною кривою частота-струм називаються типом I, тоді як моделі з неперервною кривою частота-струм називаються типом II. Для двовимірних моделей нейронів поведінку активації обох типів нейронів можна зрозуміти за допомогою аналізу фазової площини.

Моделі типу I

Математично біфуркація “saddle-node-onto-limit-cycle” зазвичай призводить до поведінки типу I. Для нульового вхідного та слабо позитивного входу припускається, що модель нейрона має три фіксовані точки в конфігурації. Стабільна нерухома точка (вузол), сідлоподібна точка і нестабільна нерухома точка. Якщо I збільшити, μ -нулькліналь рухається вгору, а стійка нерухома точка зливається з сідлом і зникає. Залишається нестабільна фіксована точка, навколо якої має бути граничний цикл, якщо потік обмежений. Якщо граничний цикл проходить через область, де зникли сідло і вузол, сценарій називається біфуркацією “saddle-node-onto-limit-cycle”(сідло-вузол на граничному циклі)[19].

Моделі типу II

Немає жодної фундаментальної причини, чому на біфуркації сідло-вузла має з'явитися граничний цикл. Дійсно, в одновимірних диференційних рівняннях можливі біфуркації сідло-вузла, але вони ніколи не призводять до граничного циклу. Більше того, якщо граничний цикл існує в двовимірній системі, немає причин, чому він повинен з'являтися безпосередньо в точці біфуркації – він також може існувати до досягнення точки біфуркації. У цьому випадку граничний цикл не проходить через руйни фіксованої точки і тому має кінцеву частоту. Це призводить до моделі нейронів типу II. Відповідну біфуркацію можна класифікувати як “Saddle-Node-off-Limit-Cycle”(сідло-вузол за межею граничного циклу)[19].

1.3 Узагальнені нейрони Integrate-and-Fire (I-F)

Нелінійні моделі

Детальні моделі нейронів на основі провідності можуть відтворювати електрофізіологічні вимірювання з високим ступенем точності, але через їхню внутрішню складність ці моделі важко аналізувати. З цієї причини прості феноменологічні моделі стрибкоподібних нейронів дуже популярні для досліджень нейронного кодування, пам'яті та динаміки мережі[4].

Форма потенціалу дії даного нейрона досить стереотипна з дуже невеликими змінами між одним спайком і наступним. Таким чином, форма потенціалу дії, що рухається вздовж аксона до постсинаптичного нейрона, не може бути використана для передачі інформації; скоріше, з точки зору приймаючого нейрона, потенціали дії — це «події», які повністю характеризуються часом приходу потенціалу дії в синапс. Потенціали дії різних типів нейронів можуть мати різну форму, а тривалість і форма імпульсу дії дійсно впливають на вивільнення нейромедіатора; але всі потенціали дії, які надходять до даного синапсу, походять від одного пресинаптичного нейрона, і — якщо нехтувати ефектами втоми іонних каналів в аксоні — можна припустити, що його часовий курс завжди однаковий. Потенціали дії розглядаються як події, що характеризуються часом їхнього спрацьовування, і завдання полягає в пошуку моделі, щоб надійно передбачити час імпульсів дії[4].

В узагальнених моделях I-F стрибки генеруються щоразу, коли мембранний потенціал u перетинає деякий поріг θ_{reset} знизу. Момент перетину порогу визначає час збудження $t^{(f)}$.

$$t^{(f)} : u(t^{(f)}) = \theta_{\text{reset}} \quad \text{and} \quad \left. \frac{du(t)}{dt} \right|_{t=t^{(f)}} > 0 \quad (4)$$

На відміну від двовимірних нейронних моделей, відсутня змінна релаксації, яка б дозволяла описати повернення мембранного потенціалу до стану спокою. У моделях I-F зниження потенціалу дії замінюється алгоритмічним скиданням мембранного потенціалу до нового значення u_r кожного разу, коли досягається поріг θ_{reset} . Тривалість потенціалу дії іноді, але не

завжди, замінюється (dead-time) Δ^{abs} після кожного стрибка, перш ніж динаміка напруги перезапуститься з $u=u_r$ як початковою умовою.

У загальній нелінійній моделі I-F з однією змінною u , мембранний потенціал змінюється відповідно

$$\tau \frac{d}{dt} u = f(u) + R(u) I \quad (5)$$

Якщо не вказано інше, завжди припускається постійний вхідний опір $R(u)=R$ незалежно від напруги. Порівняння рівняння вище з рівнянням стандартної Leaky I-F

$$\tau \frac{d}{dt} u = -(u - u_{\text{rest}}) + R I; \quad (6)$$

показує, що нелінійну функцію $R(u)$ можна інтерпретувати як залежний від напруги вхідний опір, тоді як $f(u)$ замінює термін витоку $-(u-u_{\text{rest}})$. Деякі добре відомі приклади нелінійних моделей I-F включають експоненціальну модель I-F і квадратичну модель I-F.

Якщо нелінійну модель I-F стимулювати струмами різної форми, можна визначити, з моделювання, поріг генерації потенціалів дії. Шукається максимальна напруга, яка може бути досягнута до того, як модель спрацює[6].

Експоненційна модель I-F

В експоненціальній моделі диференціальне рівняння для мембранного потенціалу задається у вигляді

$$\tau \frac{d}{dt} u = -(u - u_{\text{rest}}) + \Delta_T \exp\left(\frac{u - \vartheta_{rh}}{\Delta_T}\right) + R I \quad (7)$$

Момент, коли мембранний потенціал досягає чисельного порогу θ_{reset} , визначає час збудження $t^{(f)}$. Після збудження мембранний потенціал скидається до u_r , і інтеграція перезапускається в момент часу $t^{(f)} + \Delta_{\text{abs}}$, де Δ_{abs} — це абсолютний час рефрактерності, який зазвичай обирається в діапазоні $0 < \Delta_{\text{abs}} < 5$ мс. Якщо числовий поріг обраний досить високим, $\theta_{\text{reset}} \gg \vartheta + \Delta T$, його точне значення не грає ніякої ролі. Причина в тому, що підйом потенціалу дії для $u \gg \vartheta + \Delta T$ настільки швидкий, що йде до нескінченності за неймовірно короткий час. Порогове значення θ_{reset} введено в основному для чисельної зручності. Для формального математичного аналізу моделі поріг можна розсунути до нескінченності[4].

Квадратична модель I-F

Конкретним екземпляром нелінійної моделі I-F є квадратична модель.

$$\tau \frac{d}{dt} u = a_0 (u - u_{\text{rest}}) (u - u_c) + RI \quad (8)$$

з параметрами $a_0 > 0$ і $u_c > u_{\text{rest}}$. Для $I=0$ та початкового стану $u < u_c$ напруга спадає до потенціалу спокою u_{rest} . Для $u > u_c$ він збільшується так, що запускається потенціал дії. Таким чином, параметр u_c можна інтерпретувати як критичну напругу для ініціювання імпульсу дії коротким імпульсом струму. Квадратична модель I-F тісно пов'язана з так званим Θ -нейроном, канонічної моделлю нейронів типу I[4].

Для чисельних реалізацій моделі інтегрування рівняння припиняється, якщо напруга досягає числового порогу θ_{reset} , і перезапускається зі значенням скидання u_r як новою початковою умовою. Однак для математичного аналізу моделі стандартним припущенням є $\theta_{\text{reset}} \rightarrow \infty$ і $u_r \rightarrow -\infty$.

Стандартна модель Leaky I-F досить обмежена за обсягом, оскільки має один універсальний поріг напруги. Однак нелінійні нейрони I-F можуть пояснити той факт, що в реальних нейронах ефективний поріг напруги для

повторного спрацьовування відрізняється від порогу напруги, знайденого для коротких імпульсів струму. Ці два пороги напруги, які пов'язані з мінімумом нелінійності $f(u)$ і з нестабільною фіксованою точкою, відповідно, є невід'ємними ознаками нелінійних моделей інтегрування та вогонь. Як тільки мембранний потенціал перевищує внутрішній поріг, починається підйом мембранного потенціалу. Інтеграція зупиняється на чисельному порозі θ_{reset} , який набагато вищий і концептуально дуже відрізняється від внутрішнього порогу спрацьовування моделі. Насправді, точне значення числового порогу не має значення, оскільки без такого порогу потенціал мембрани йшов би до нескінченності за скінченний час[4].

Можна уявити багато різних форм нелінійності. Однак виявляється, що багато нейронів добре описуються лінійним терміном («leak») у поєднанні з експоненціальним терміном («activation»). Таким чином, експоненціальна модель I-F має «правильну» нелінійність, тоді як квадратична модель I-F занадто нелінійна в підпороговому режимі і занадто повільна в надпороговому режимі.

Як експоненціальна, так і квадратична модель I-F показують криву частотно-струмового типу I. Дійсно, близько до точки біфуркації обидві моделі стають ідентичними і можуть бути відображені на канонічну модель типу I.

Адаптивна експоненційна I-F

Одного рівняння недостатньо, щоб описати різноманітність моделей збудження, які нейрони демонструють у відповідь на ступінчастий струм. Тому ми зв'язуємо рівняння напруги з абстрактними змінними струму w_k , кожна з яких описується лінійним диференціальним рівнянням.

$$\begin{aligned} \tau_m \frac{du}{dt} &= f(u) - R \sum_k w_k + RI(t) \\ \tau_k \frac{dw_k}{dt} &= a_k (u - u_{\text{rest}}) - w_k + b_k \tau_k \sum_{t^{(f)}} \delta(t - t^{(f)}) \end{aligned} \quad (9)$$

Зв'язок напруги зі струмом адаптації w_k реалізується параметром a_k і розвивається з постійною часу τ_k . Адаптаційний струм повертається до рівняння напруги з опором R . Як і в інших моделях I-F, змінна напруги u скидається, якщо потенціал мембрани досягає чисельного порогу Θ_{reset} . Момент $u(t)=\Theta_{\text{reset}}$ визначає час збудження $t(f)=t$. Після збудження інтегрування напруги знову починається при $u=u_r$. δ -функція в рівняннях w_k вказує, що під час збудження адаптаційні струми w_k збільшуються на величину b_k . Параметри b_k є «стрибком» адаптації, що спрацьовує потенціалом дії. Одна з можливих біофізичних інтерпретацій збільшення полягає в тому, що під час потенціалу дії кальцій потрапляє в клітину, так що амплітуда кальцій залежного калієвого струму збільшується[17].

1.4 Модель Іжикевича

У той час як модель AdEx демонструє нелінійність експоненціальної моделі I-F, модель Іжикевича використовує квадратичну модель I-F для першого рівняння

$$\begin{aligned} \tau_m \frac{d u}{d t} &= (u - u_{\text{rest}}) (u - \vartheta) - R w + R I(t) \\ \tau_w \frac{d w}{d t} &= a (u - u_{\text{rest}}) - w + b \tau_w \sum_{t^{(f)}} \delta(t - t^{(f)}) \end{aligned} \quad (9)$$

Якщо $u=\Theta_{\text{reset}}$, напруга скидається до $u=u_r$, а змінна адаптації w збільшується на величину b . Зазвичай b є додатним, але також можливо $b<0$ [11].

“Leaky” модель з адаптацією

Змінні адаптації w_k також можна об'єднати зі стандартною Leaky I-F моделлю.

$$\begin{aligned} \tau_m \frac{d u}{d t} &= - (u - u_{\text{rest}}) - R \sum_k w_k + R I(t) \\ \tau_k \frac{d w_k}{d t} &= a (u - u_{\text{rest}}) - w_k + b_k \tau_k \sum_{t^{(f)}} \delta(t - t^{(f)}) \end{aligned} \quad (10)$$

У момент збудження, визначеного пороговою умовою $u(t(f)) = \theta_{\text{reset}}$, напруга скидається до $u = u_r$ і змінні адаптації w_k збільшуються на величину b_k . В моделі Leaky I-F числовий поріг θ_{reset} збігається з порогом напруги ϑ , який можна було б знайти при коротких імпульсах вхідного струму[11].

1.5 Варіативність послідовностей потенціалів дії і нейронне кодування

Моделі нейронів, описані попередньо, є детермінованими і для більшості параметрів генерують серії імпульсів дії, які виглядають регулярними, коли є постійний стимул. Проте записи нейронної активності *in vivo* характеризуються високим ступенем нерегулярності. Серія імпульсів дії окремого нейрона далека від періодичності, а кореляції між часами потенціалів дії сусідніх нейронів слабкі. Якщо електричну активність, яку вловлює позаклітинний електрод, можна почути через гучномовець, то в реальності чути лише шум. На питання, чи це справді просто шум чи, радше, високоефективний спосіб кодування інформації, не можна легко відповісти. Справді, прослуховування комп'ютерного модему чи факсимільного апарату також може створити враження, що це просто шум. Можливість вирішити, чи спостерігається нейронна активність, яка лежить в основі композиції, а не просто безглуздий флікер, є однією з найбільш гострих проблем у нейронауці[10].

Для вирішення цієї проблеми було проведено кілька експериментів. Здається, що нейрон *in vitro*, після того, як він ізольований від мережі, може дуже надійно та відтворюваним чином реагувати на коливання вхідного струму, а також нейрони в сенсорній корі *in vivo*, коли керуються сильним залежним від часу сигналом. З іншого боку, нейрони виробляють нерегулярні серії потенціалів дії за відсутності будь-яких тимчасово структурованих стимулів. Нерегулярна спонтанна активність, тобто активність, яка ніяким чином не пов'язана із зовнішньою стимуляцією, і варіації нейрональних реакцій від проби до проби часто розглядаються як шум[11].

Походження цієї нерегулярності нейронної динаміки *in vivo* недостатньо вивчене. Таким чином, у моделях I-F шум часто явно додається до нейронної динаміки, щоб імітувати непередбачуваність нейронних записів.

Якщо моделі нейронів, такі як модель Ходжкіна-Хакслі або модель I-F, керуються досить сильним постійним струмом, вони генерують регулярну послідовність імпульсів дій. У нейронних моделях з адаптаційними струмами на початку може бути коротка перехідна фаза, але потім усі інтервали між потенціалами дій є постійними. Серії типових нейронів *in vivo* демонструють набагато більш нерегулярну поведінку[13].

Багато експериментів *in vivo* показують шумну поведінку нейронів кори. Наприклад, активність нейронів зорової кори можна зафіксувати, коли на екрані в полі зору тварини повільно рухається смуга. Як тільки смужка потрапляє в сприйнятливий поле нейрона, швидкість спрацьовування зростає. Однак, коли один і той самий експеримент повторюють кілька разів, від випробування до випробування значно змінюється послідовність. Аналогічно, нейрони в області сенсорної кори щурів або мишей систематично реагують на рухи вусів, але реакція дещо різниться між одним випробуванням і наступним. Крім того, той самий нейрон час від часу випромінює спонтанний стрибок, навіть якщо не застосовується зовнішній стимул. Під час спонтанної активності траєкторія напруги значно коливається, і інтервали між одним сплеском і наступним демонструють великий ступінь мінливості[13].

У цілому і неушкодженому мозку нейрони набагато надійніше реагують на зовнішній подразник, що швидко змінюється, ніж на подразники, що постійні або повільно рухаються.

Джерелом шуму, який є буквально всюдисущим, є тепловий шум. Через дискретну природу носіїв електричного заряду напруга u на будь-якому електричному резисторі R коливається при кінцевій температурі (шум Джонсона). Дисперсія флуктуацій у стані спокою дорівнює

$$\langle \Delta u^2 \rangle \propto RkTB \quad (11)$$

де k — постійна Больцмана, T — температура, а B — смуга пропускання системи. Оскільки динаміка нейронів описується еквівалентним електричним ланцюгом, що містить резистори, потенціал нейронної мембрани також коливається. Однак флуктуації, викликані шумом Джонсона, мають незначне значення в порівнянні з іншими джерелами шуму в нейронах.

Інше джерело шуму, характерне для нейронних клітин і присутнє в ізольованому нейроні, виникає через кінцеву кількість іонних каналів у ділянці нейрональної мембрани. Більшість іонних каналів мають лише два стани: вони або відкриті, або закриті. Електропровідність ділянки мембрани для іонів типу i пропорційна кількості відкритих іонних каналів. При заданому постійному мембранному потенціалі u частка $P_i(u)$ іонного каналу i типу в середньому відкрита. Фактична кількість відкритих каналів коливається навколо $N_i P_i(u)$, де N_i — загальна кількість іонних каналів типу i в цьому ділянці мембрани[13].

Формулювання рівнянь Ходжкіна-Хакслі в термінах провідності іонних каналів неявно засноване на припущенні про велику кількість іонних каналів, тому флуктуаціями можна знехтувати. Оскільки насправді N_i має скінченне значення, провідність коливається. Якщо мембранний потенціал близький до порогу, шум каналу може бути критичним для генерації потенціалів дії. Моделі, які враховують кінцеву кількість іонних каналів, можуть відтворити спостережувану мінливість реальних нейронів при внутрішньоклітинній стимуляції. Зокрема, вони показують невелике стрибкове тремтіння, якщо вхідний струм швидко змінюється, але менш надійні, якщо вхідний струм є постійним[13].

Шум від мережі

Крім джерел внутрішнього шуму на рівні окремого нейрона існують також джерела шуму, які зумовлені передачею сигналу та мережевими ефектами

(зовнішній шум). Наприклад, збої в синаптичній передачі, схоже, накладають істотне обмеження на передачу сигналу в нейронній мережі. Експерименти із записом подвійних електродів від двох синаптично з'єднаних нейронів показують, що лише 10–30 відсотків пресинаптичних потенціалів дій генерують постсинаптичну відповідь.

Нарешті, важлива частина нерегулярності нейронних стрибків під час спонтанної активності, пов'язана з властивостями мережі – навіть якщо сама мережа повністю детермінована. Дослідження моделі показують, що мережі збуджуючих і гальмівних нейронів із фіксованою випадковою зв'язністю можуть створювати дуже нерегулярні серії імпульсів дій – навіть за відсутності будь-якого джерела шуму. В результаті активності мережі кожен нейрон отримує на вхід неправильну послідовність потенціалів дії, яку можна описати як стохастичне прибуття імпульсу дії. Тому різницю між великою мінливістю нейронів *in vivo* порівняно з мінливістю під час внутрішньоклітинної стимуляції *in vitro* можна, принаймні частково, віднести до мережевих ефектів[6].

1.6 Середня швидкість збудження

Найбільш часто використовуване визначення швидкості збудження відноситься до середнього за часом. Експерименталіст спостерігає у дослідженні k потенціали дії даного нейрона. Швидкість спрацьовування у експерименті k — це кількість потенціалів дії n_k^{sp} в інтервалі тривалості T , поділена на T .

$$v_k = \frac{n_k^{sp}}{T} \quad (12)$$

Довжина T часового вікна встановлюється експериментатором і залежить від типу нейрона та стимулу. На практиці, щоб отримати розумні середні значення, протягом періоду часу має відбутися кілька збуджень. Типовими

значеннями є $T=100$ мс або $T=500$ мс, але тривалість також може бути більшою або меншою[6].

Це визначення швидкості успішно використовується в багатьох препаратах, зокрема в експериментах на сенсорних або моторних системах.



Рис. 1.4 – Середня швидкість збудження

Середня швидкість збудження як популяційна активність (середнє за кількома нейронами)

Кількість нейронів у мозку величезна. Часто багато нейронів мають схожі властивості і реагують на однакові подразники. Наприклад, нейрони первинної зорової кори кішок і мавп розташовані в стовпчиках клітин зі схожими властивостями. Якщо ідеалізувати ситуацію і розглянути популяцію нейронів з однаковими властивостями, то всі нейрони в популяції повинні мати однакову схему вхідних і вихідних зв'язків. Потенціали дії нейронів у популяції m відправляються до іншої популяції n . У ідеалізованій картині кожен нейрон у популяції n отримує вхідні дані від усіх нейронів у популяції m . Відповідна

величина, з точки зору приймального нейрона, — це частка активних нейронів у пресинаптичній популяції m . Формально визначається активність популяції[14].

1.7 Інтервальний розподіл і коефіцієнт варіації

Оцінка розподілів інтервалу між потенціалами дії (ISI) за експериментальними даними є звичайним методом дослідження мінливості нейронів за певного стаціонарного входу. У типовому експерименті реєструють послідовність потенціалів дії одного нейрона (наприклад, нейрона в зоровій корі) під час дії постійного стимулу. Стимулом може бути зовнішній вхід, застосований до системи (наприклад, візуальна контрастна решітка, що рухається з постійною швидкістю); або це може бути внутрішньоклітинний постійний керуючий струм. Аналізується послідовність потенціалів дії і на гістограмі наноситься розподіл інтервалів s_k між двома наступними імпульсами. Для достатньо довгої послідовності гістограма забезпечує хорошу оцінку розподілу ISI, який позначають як $P_0(s)$. Інтервальний розподіл можна інтерпретувати як умовну густину ймовірності[15].

$$P_0(s) = P(t^{(f)} + s | t^{(f)}) \quad (13)$$

Середній інтервал:

$$\langle s \rangle = \int_0^{\infty} s P_0(s) ds \quad (14)$$

А середня швидкість збудження є оберненою величиною, тому:

$$\nu = \frac{1}{\langle s \rangle} = \left[\int_0^{\infty} s P_0(s) ds \right]^{-1} \quad (15)$$

Розподіли між потенціалами дії інтервалами $P_0(s)$, отримані від серії імпульсів дії у стаціонарних умовах, можуть бути широкими або різко

піковими. Щоб кількісно визначити ширину інтервального розподілу, нейробіологи часто оцінюють коефіцієнт варіації, C_V , який визначається як відношення стандартного відхилення та середнього[15].

$$C_V^2 = \frac{\langle \Delta s^2 \rangle}{\langle s \rangle^2} \quad (16)$$

1.8 Проблема нейронного кодування

Експериментатор як зовнішній спостерігач може оцінити та класифікувати активацію нейронів за допомогою вимірювання кількості потенціалів дії – але чи дійсно це код, який використовується нейронами в мозку? Іншими словами, чи кортикальний нейрон, який отримує сигнали від сенсорного нейрона, лише дивиться і реагує на кількість стрибків, які він отримує, у часовому вікні, наприклад, 500 мс[20]?

З точки зору швидкості кодування, потенціали дії є просто зручним способом передачі аналогової вихідної змінної v на великі відстані. Насправді, найкращою схемою кодування для передачі значення швидкості v буде звичайна послідовність імпульсів дії з інтервалами $1/v$. У цьому випадку швидкість можна було б надійно виміряти лише після двох стрибків. З точки зору швидкості кодування, нерівності, які зустрічаються в реальних ланцюгах нейронів кори, слід розглядати як шум.

Щоб позбутися шуму і отримати надійну оцінку швидкості, експериментатору доводиться усереднювати більшу кількість скачків.

Час до першого стрибка: код затримки

Якщо розглянути нейрон, який раптово отримує новий постійний вхідний струм у момент часу t_0 . Наприклад, нейрон може керуватися зовнішнім стимулом, який раптово вмикається в момент t_0 . Це здається дещо штучним, але навіть у реалістичній ситуації різкі зміни у вхідних даних досить поширені.

Дивлячись на картинку, погляд стрибає з однієї точки на іншу. Після кожної саккади фоторецептори сітківки отримують новий візуальний вхід. Інформація про початок сакад повинна бути легко доступною в мозку і може служити внутрішнім еталонним сигналом. Тоді ми можемо уявити код, де для кожного нейрона час першого сплеску після опорного сигналу містить всю інформацію про новий стимул. Нейрон, який спрацьовує незабаром після опорного сигналу, інтерпретується як сильна стимуляція цього нейрона, тоді як спрацьовування дещо пізніше буде сигналізувати про більш слабку стимуляцію[20].

У чистій версії цієї схеми кодування кожному нейрону для передачі інформації потрібен лише один потенціал дії. Якщо він випромінює кілька імпульсів, зараховується лише перший імпульс після опорного сигналу. Усі наступні стрибки не мають значення. Щоб реалізувати чисту версію такої схеми кодування, уявляють, що кожен нейрон відключається гальмуванням, як тільки він запускає потенціал дії. Гальмування закінчується з появою наступного стимулу (наприклад, після наступної саккади). Після звільнення від гальмування нейрон готовий випромінювати свій наступний потенціал дії, який тепер передає інформацію про новий стимул. Оскільки кожен нейрон у такому сценарії передає рівно один потенціал дії на стимул, зрозуміло, що інформацію передає лише час, а не кількість потенціалів дії.

Шумовий вхідний сигнал

Нейрони в мозку отримують інформацію від тисяч інших пресинаптичних нейронів, які випромінюють потенціали дії і посилають свої потенціали дії до постсинаптичних цілей. З точки зору постсинаптичного нейрона, який отримує шквал імпульсів дії, час прибуття потенціалу дії може виглядати абсолютно випадковим, навіть якщо припустити, що пресинаптичні нейрони генерують свої потенціали дії за допомогою детермінованого процесу. Внутрішні джерела шуму клітини, такі як спонтанне відкриття іонних каналів, не враховують усієї мінливості послідовностей потенціалів дії, які зустрічаються у тварин, що вільно поводяться *in vivo*. Швидше за все, цілком імовірно, що велика частка

видимої мінливості генерується мережею. Дослідження моделювання підтверджують, що мережі з фіксованою випадковою зв'язністю можуть призвести до хаосу на мікроскопічному рівні, так що час надходження стрибків здається випадковим, навіть якщо він генерується детермінованою мережею[14].

Нейрони керуються вхідним струмом $I(t)$, який підсумовує ефект синаптичного введення від інших нейронів мережі *in vivo* або струму, введеного експериментатором у клітину *in vitro*. Моделювання шумності входу зводиться до розбиття вхідного струму на дві складові, детерміновану та стохастичну.

$$I(t) = I^{\text{det}}(t) + I^{\text{noise}}(t) \quad (17)$$

де детермінований I^{det} підсумовує частину струму, яка відома або, принаймні, передбачувана, тоді як стохастичний I^{noise} є непередбачуваною або шумовою частиною струму.

Наприклад, під час дослідження *in vitro* з внутрішньоклітинною ін'єкцією струму, I^{det} буде струмом, який встановлюється на комутаційному щиті генератора струму, але фактичний струм коливається навколо попередньо встановленого значення через кінцеву температуру. У нейронному записі під час візуального психофізичного експерименту *in vivo* частина вхідного струму, яка не змінюється під час випробувань з одним і тим же зоровим стимулом, буде підсумована як I^{det} , тоді як усі інші вхідні сигнали до нейрона, які змінюються від одного випробування до дослідження, наступні розглядаються як шум і підсумовуються як I^{noise} .

Для моделювання доданок шуму просто додається в правій частині диференційного рівняння напруги. Наприклад, нелінійна модель I-F із шумовим входом має рівняння напруги.

$$\tau_m \frac{d}{dt} u = f(u) + RI^{\text{det}}(t) + RI^{\text{noise}}(t) \quad (18)$$

Якщо u досягає порогового значення θ_{reset} , інтеграція припиняється, а потенціал мембрани скидається до u_r . Процедура додавання вхідного шуму повністю аналогічна для біофізичних моделей типу Ходжкіна-Хакслі або моделей I-F з адаптацією[14].

Білий шум

Стандартна процедура реалізації шуму RI^{noise} в диференціальному рівнянні напруги на мембрані полягає в тому, щоб сформулювати його як «білий шум», $RI^{\text{noise}}(t) = \xi(t)$. Білий шум ξ — це випадковий процес, що характеризується його очікуваним значенням

$$\langle \xi(t) \rangle = 0 \quad (19)$$

і автокореляцією

$$\langle \xi(t) \xi(t') \rangle = \sigma^2 \tau_m \delta(t - t') \quad (20)$$

де σ — амплітуда шуму (в одиницях напруги), τ_m — постійна часу диференціального рівняння. Точне знання значення ξ в момент t не дозволяє передбачити його значення в будь-який інший момент часу $t' \neq t$. Перетворення Фур'є автокореляційної функції дає спектр потужності. Спектр потужності білого шуму плоский(? - flat), тобто шум однаково сильний на всіх частотах.

Якщо додати член білого шуму в правій частині, ми отримаємо стохастичне диференціальне рівняння, тобто рівняння для стохастичного процесу,

$$\tau_m \frac{d}{dt} u(t) = f(u(t)) + R I^{\text{det}}(t) + \xi(t) \quad (21)$$

також відомий як рівняння Ланжевена[14].

Кольоровий шум

Термін шуму зі спектром потужності, який не є плоским, називається кольоровим шумом. Кольоровий шум $I^{\text{noise}}(t)$ може бути створений з білого шуму шляхом відповідної фільтрації. Наприклад, фільтрація низьких частот

$$\tau_s \frac{dI^{\text{noise}}(t)}{dt} = -I^{\text{noise}}(t) + \xi(t) \quad (22)$$

де $\xi(t)$ – процес білого шуму, визначений вище, дає кольоровий шум зі зниженою потужністю на частотах вище $1/\tau_s$.

Спектр потужності:

$$\langle I^{\text{noise}}(t) I^{\text{noise}}(t') \rangle = a \exp\left(-\frac{|t-t'|}{\tau_s}\right) \quad (23)$$

a - коефіцієнт амплітуди.

Отже, знання вхідного струму в момент t дає нам підказку про вхідний струм незабаром після цього, поки $|t'-t| \ll \tau_s$.

Спектр шуму є перетворенням Фур'є. Він є плоским для частот $\omega \ll 1/\tau_s$ і спадає при $\omega > 1/\tau_s$. Іноді $1/\tau_s$ називають частотою зрізу[3].

Стохастичне прибуття потенціалу дії

Типовий нейрон, наприклад, пірамідна клітина в корі хребетних, отримує вхідні потенціали дії від тисяч інших нейронів, які, у свою чергу, отримують вхідні дані від своїх пресинаптичних нейронів і так далі. Хоча неможливо об'єднати мільйони нейронів I-F у величезну мережеву модель, часто доцільно зосередити зусилля з моделювання на певній підгрупі нейронів, наприклад, і описати вхідні дані від інших частин мозку як стохастичну фонову активність[9].

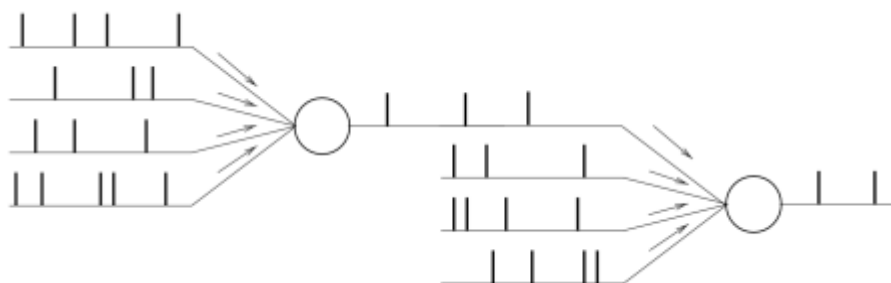


Рис. 1.5 Схема прибуття потенціалу дії

Кожен нейрон отримує вхідні потенціали дії від великої кількості пресинаптичних нейронів. Лише невелика частина вхідних даних надходить від нейронів в рамках моделі мережі; інший вхід описується як стохастичний стрибок.

Якщо розглянути нелінійний нейрон I-F з індексом i , який є частиною великої мережі. Його вхід складається з зовнішнього входу $I^{\text{ext}}_i(t)$; вхідних потенціалів дії $t^{(f)}_j$ від інших нейронів j мережі; і стохастичного потенціалу дії, що надходить $t^{(f)}_k$ через фонову активність в інших частинах мозку. Мембранний потенціал u_i розвивається відповідно до

$$\frac{d}{dt}u_i = \frac{f(u_i)}{\tau_m} + \frac{1}{C}I^{\text{ext}}(t) + \sum_j \sum_{t^{(f)}_j} w_{ij} \delta(t - t^{(f)}_j) + \sum_k \sum_{t^{(f)}_k} w_{ik} \delta(t - t^{(f)}_k) \quad (24)$$

де δ — δ -функція Дірака, а w_{ij} — сила зв'язку пресинаптичних нейронів j у мережі до нейрона i . Вхідні дані від фонових нейронів зважуються коефіцієнтом w_{ik} . У той час як часи спрацьовування $t^{(f)}_j$ генеруються пороговими перетинами пресинаптичних нейронів I-F, часи спрацьовування $t^{(f)}_k$ фонового нейрона k генеруються за допомогою процесу Пуассона із середньою швидкістю ν_k .

Для полегшення аналізу приймають декілька спрощень. По-перше, зосереджуються на нейроні Leaky I-F та зміщують напругу так, щоб потенціал спокою дорівнював нулю. Отже, можна встановити $f(u)=-u$. По-друге, концентруються на одному нейроні, який отримує стохастичний вхід від фонових нейронів. Отже, можна відкинути суму за j , яка представляє вхід з мережі, а також відкинути індекс i конкретного нейрона. Тому приходять до

$$\frac{d}{dt}u = -\frac{u}{\tau_m} + \frac{1}{C}I^{\text{ext}}(t) + \sum_k \sum_{t_k^{(f)}} w_k \delta(t - t_k^{(f)}) \quad (25)$$

Потенціал мембрани скидається до u_r щоразу, коли він досягає порога ϑ . Даний вираз називається моделлю Штейна

У моделі Штейна кожен вхідний потенціал дії генерує постсинаптичний потенціал $\Delta u(t)=w_k\epsilon(t-t_k^{(f)})$ з $\epsilon(s)=e^{-s/\tau_m}\Theta(s)$, тобто потенціал стрибає після надходження потенціалу дії на величину w_k і після цього експоненційно спадає. Інтегрування рівняння дає

$$u(t|\hat{t}) = u_r \exp\left(-\frac{t-\hat{t}}{\tau_m}\right) + \frac{1}{C} \int_0^{t-\hat{t}} \exp\left(-\frac{s}{\tau_m}\right) I(t-s) ds + \sum_{k=1}^N \sum_{t_k^{(f)}} w_k \epsilon(t - t_k^{(f)}) \quad (26)$$

для $t > \hat{t}$ де \hat{t} — час останнього спрацьовування нейрона. Можна узагальнити модель, щоб враховувати синаптичну постійну часу та працювати з довільними

постсинаптичними потенціалами $\epsilon(s)$, які генеруються прибуттям стохастичних потенціалів дії.

Кожен потенціал дії, що надходить у синапс, викликає зміну мембранного потенціалу постсинаптичного нейрона. Якщо імпульси дії надходять стохастично, мембранний потенціал експонує коливання навколо середньої траєкторії. Якщо флуктуації залишаються в підпороговому режимі, коли властивості мембрани можна апроксимувати за допомогою лінійного рівняння, середнє значення та стандартне відхилення траєкторії можна розрахувати аналітично, враховуючи параметри стохастичного процесу, які характеризують надходження імпульси дії. За наявності порогу спрацьовування флуктуації мембранного потенціалу, викликані приходами стохастичних імпульсів дії, можуть спричинити збудження нейрона, навіть якщо середня траєкторія ніколи не досягне порогу спрацьовування[9].

Оскільки швидкість надходження потенціалів дії у збудливі та гальмівні синапси висока, тоді як кожен потенціал дії викликає лише невеликий стрибок мембранного потенціалу, синаптичне бомбардування можна апроксимувати сумою двох доданків: середній вхідний струм і гаусівський білий шум. Білий шум призводить до «дифузії» траєкторії мембранного потенціалу навколо середньої траєкторії. Еволюція розподілу ймовірностей $p(u,t)$ мембранного потенціалу в часі описується рівнянням Фоккера-Планка. Для моделі Leaky I-F і стаціонарного входу рівняння Фоккера-Планка можна розв'язати аналітично. Для нелінійних нейронів I-F можливі чисельні рішення, що залежать від часу[9].

РОЗДІЛ 2

МАТЕРІАЛИ І МЕТОДИ ДОСЛІДЖЕНЬ

2.1 Моделювання нейронної динаміки

Для моделювання було використано мову програмування Python та кілька бібліотек.

- Brian 2 - це симулятор нейронних мереж. Цей симулятор повинен економити час процесорів, і тих, хто ним користується. Brian2 дуже гнучкий і легко розширюваний.
- NumPy - бібліотека, яка вводить і полегшує роботу з математичним апаратом.
- SciPy - бібліотека, схожа на NumPy, є бібліотекою високоякісних наукових інструментів.
- Matplotlib - використовується для візуалізації результатів.

Було написано код, що спрощує роботу з вхідними даними, він був написаний так, щоб кожна симуляція нейронів його використала.

```
def step_curr(t_start, t_end, u_t, amp, append_zero=True):  
  
    if not isinstance(t_start, int) or not isinstance(t_end, int)\  
        or not b2.units.fundamentalunits.have_same_dimensions(amp, b2.amp):  
        return AssertionError("There is problem with input data or dimensions of them")  
  
    temp_size = 1 + t_end  
    if append_zero:  
        temp_size += 1  
    temp = np.zeros((temp_size, 1)) * b2.amp  
    temp[t_start: t_end + 1, 0] = amp  
    current = b2.TimedArray(temp, dt=1. * u_t)  
    return current
```

Рис. 2.1 – Фрагмент коду з реалізацією східчастого струму

```

def sinus_curr(t_start, t_end, u_t,
               amp, frq, dir_curr, offset=0.,
               append_zero=True):

    if not isinstance(t_start, int) or not isinstance(t_end, int)\
        or not b2.units.fundamentalunits.have_same_dimensions(amp, b2.amp)\
        or not b2.units.fundamentalunits.have_same_dimensions(dir_curr, b2.amp):
        return AssertionError("There is problem with input data or dimensions of them")

    temp_size = 1 + t_end
    if append_zero:
        temp_size += 1
    tmp = np.zeros((temp_size, 1)) * b2.amp
    if t_end > t_start:
        phi = range(0, (t_end - t_start) + 1)
        phi = phi * u_t * frq
        phi = phi * 2. * math.pi + offset
        c = numpy.sin(phi)
        c = (dir_curr + c * amp)
        tmp[t_start: t_end + 1, 0] = c
    curr = b2.TimedArray(tmp, dt=1. * u_t)
    return curr

```

Рис.2.2 – Фрагмент коду з реалізацією синусоїдального струму

Крім цього було створено додатково код, який допомагає у візуалізації результатів, який полегшував побудову залежностей для кожного виду моделі. Використовувалась в більшості бібліотека matplotlib.

```

def volt_and_curr_plotting(volt_mon, curr, title=None, f_threshold=None, legend_pos=0):

    t_val = volt_mon.t / b2.ms

    axis_c = plt.subplot(211)
    c = curr(volt_mon.t, 0)
    max_curr = max(curr(volt_mon.t, 0))
    min_curr = min(curr(volt_mon.t, 0))
    mrg = 1.05 * (max_curr - min_curr)
    plt.plot(volt_mon.t / b2.ms, c, "r", lw=2)
    if mrg > 0.:
        plt.ylim((min_curr - mrg) / b2.amp, (max_curr + mrg) / b2.amp)
    plt.ylabel("Input current \n min: {} max: {}".format(min_curr, max_curr))
    plt.grid()
    axis_v = plt.subplot(212)
    plt.plot(t_val, volt_mon[0].v / b2.mV, lw=2)
    if f_threshold:
        plt.plot(
            (volt_mon.t / b2.ms)[[0, -1]],
            [f_threshold / b2.mV, f_threshold / b2.mV],
            "g--", lw=2
        )
    max_val = max(volt_mon[0].v)
    if f_threshold is not None:
        max_val = max(max_val, f_threshold)
    min_val = min(volt_mon[0].v)
    mrg = 0.05 * (max_val - min_val)
    plt.ylim((min_val - mrg) / b2.mV, (max_val + mrg) / b2.mV)
    plt.xlabel("t [ms]")
    plt.ylabel("Membrane voltage [mV]\n min: {} max: {}".format(min_val, max_val))
    plt.grid()

    if f_threshold:
        plt.legend(["vm", "firing threshold"], fontsize=12, loc=legend_pos)

    if title:
        plt.suptitle(title)
    return axis_c, axis_v

```

Рис.2.3 – Фрагмент коду, який спрощує та генералізує побудову залежностей з напругою та струмом

Щодо вхідних даних, то використовувались ті, що були отримані експериментально раніше в інших дослідках, вони були конвертовані спеціально під бібліотеку Brian2.

РОЗДІЛ 3

РЕЗУЛЬТАТИ ДОСЛІДЖЕНЬ

Результатами досліджень є код, який симулює нейронну динаміку з різних моделей. Власне запуск цього коду будує нейронну динаміку, яка показує поведінку нейронів.

На початку було показано приклад результату побудови моделі «Leaky Integrate-and-Fire», розглянемо тепер його наступну варіацію.

3.1 Симуляція експоненційної моделі «Leaky Integrate-and-Fire»

Короткий імпульсний струм тривалістю 2 мс вводиться в нейрон. Амплітуди струму достатньо, щоб викликати потенціал дії.

Для моделювання використовується рівняння (7), будується функція, фрагмент якої буде показано нижче. Для побудови необхідно задати вхідні аргументи

- постійна часу мембрани
- опір мембрани
- потенціал спокою
- поріг скидання
- поріг напруги для появи потенціалу дії
- гострота експоненційного члена
- вхідний струм
- тривалість симуляції

```

def int_and_fir_neuron_sim(
    v_rest=V_REST,
    simulation_time=200 * b2.ms):

    neuron = b2.NeuronGroup(1, model=eqs, reset="v=v_reset", threshold="v>v_spike", method="euler")
    neuron.v = v_rest
    voltage_monitor = b2.StateMonitor(neuron, ["v"], record=True)
    spike_monitor = b2.SpikeMonitor(neuron)

    net = b2.Network(neuron, voltage_monitor, spike_monitor)
    net.run(simulation_time)

    return voltage_monitor, spike_monitor

input_current = input_factory.get_step_current(
    t_start=20, t_end=120, unit_time=b2.ms, amplitude=0.8 * b2.namp)

st_mon, sp_mon = int_and_fir_neuron_sim(
    simulation_time=200*b2.ms)

plot_factory.plot_voltage_and_current_traces(
    st_mon, input_current, title="step current",
    firing_threshold=FIRING_THRESHOLD_v_spike)

plt.show()

```

Рис. 3.1 – Фрагмент коду з функцією, що симулює нейронну динаміку типу експоненційної «Leaky Integrate-and-Fire»

Використовуючи matplotlib можна побудувати поведінку з часом, де видно потенціал дії.

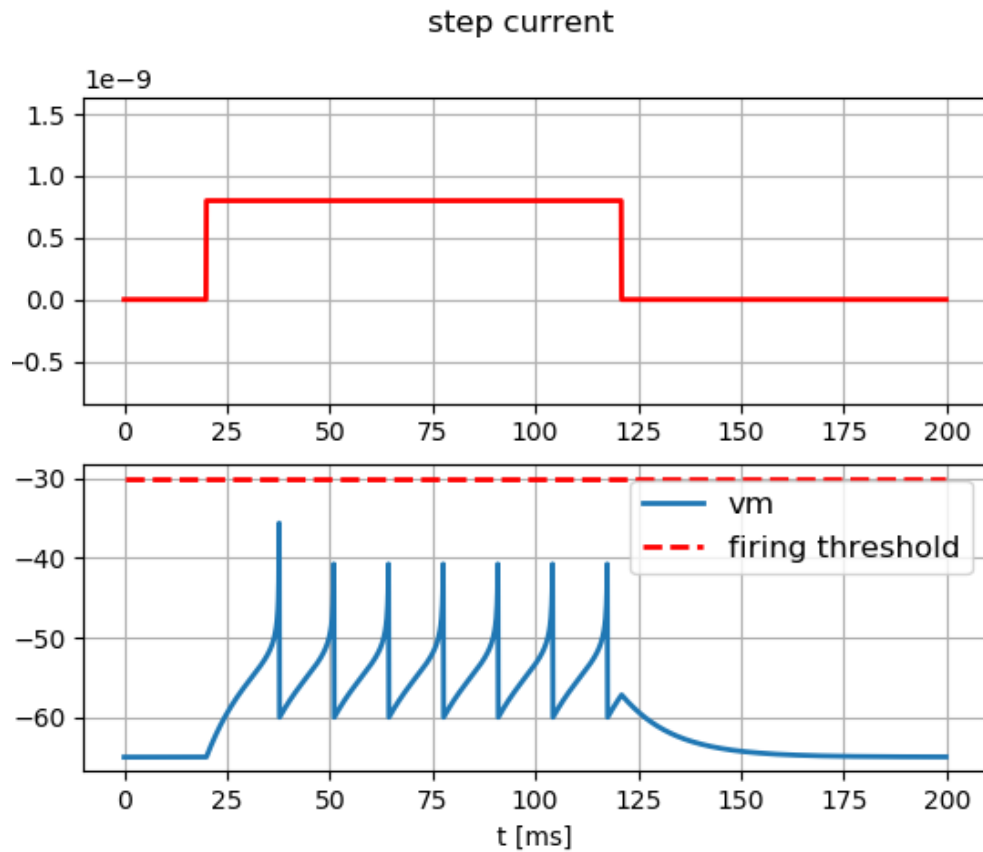


Рис. 3.2 – Нейронна динаміка моделі «Leaky Integrate-and-Fire»

3.2 Симуляція адаптивної експоненційної моделі

Для симуляції такої моделі застосовується система рівнянь (9), де використовуються ті самі вхідні дані з декількома додатками.

- зв'язок адаптації-напруги
- струм адаптації, викликаний потенціалом дії (приріст після кожного стрибка)

```

def sim_ad_ex_neuron(
    v_rest=V_REST,
    v_spike=FIRING_THRESHOLD_v_spike,
    simulation_time=200 * b2.ms):

    threshold = "v>{:f}*mvolt".format(v_spike / b2.mvolt)

    neuron = b2.NeuronGroup(1, model=eqs, threshold=threshold, reset="v=v_reset;w+=b", method="euler")

    neuron.v = v_rest
    neuron.w = 0.0 * b2.pA

    st_mon = b2.StateMonitor(neuron, ["v", "w"], record=True)
    sp_mon = b2.SpikeMonitor(neuron)

    # running simulation
    b2.run(simulation_time)
    return st_mon, sp_mon

```

Рис. 3.3 – Фрагмент коду з функцією, що симулює нейронну динаміку моделі адаптивної експоненційної «Leaky Integrate-and-Fire»

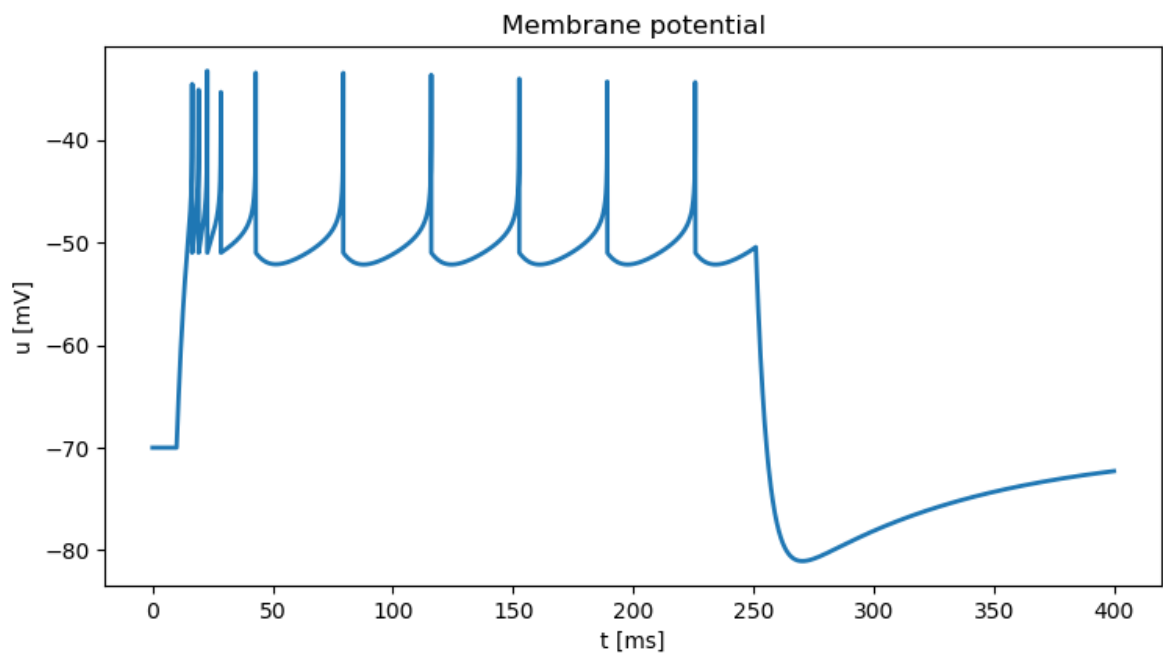


Рис. 3.4 – Потенціал мембрани, поведінка при адаптивній експоненційній моделі «Leaky Integrate-and-Fire»

3.3 Симуляція моделі Ходжкіна-Хакслі

Як було згадано вище, модель Ходжкіна-Хакслі є базовою, вона описувала поведінку аксона гігантського кальмару при генерації і передачі сигналу.

На рисунку 1.2 детально описана модель за допомогою диференційного рівняння, вона й була використана для симуляції.

Моделі типу Ходжкіна-Хакслі представляють біофізичні характеристики клітинних мембран. Ліпідний бішар представлений у вигляді ємності (C_m). Іонні канали, що керуються напругою, та іонні канали витоку представлені нелінійною (g_n) та лінійною (g_L) провідністю відповідно. Електрохімічні градієнти, що стимулюють потік іонів, представлені акумуляторами з ЕРС (E), а іонні насоси представлені джерелами струму (I_p)[13].

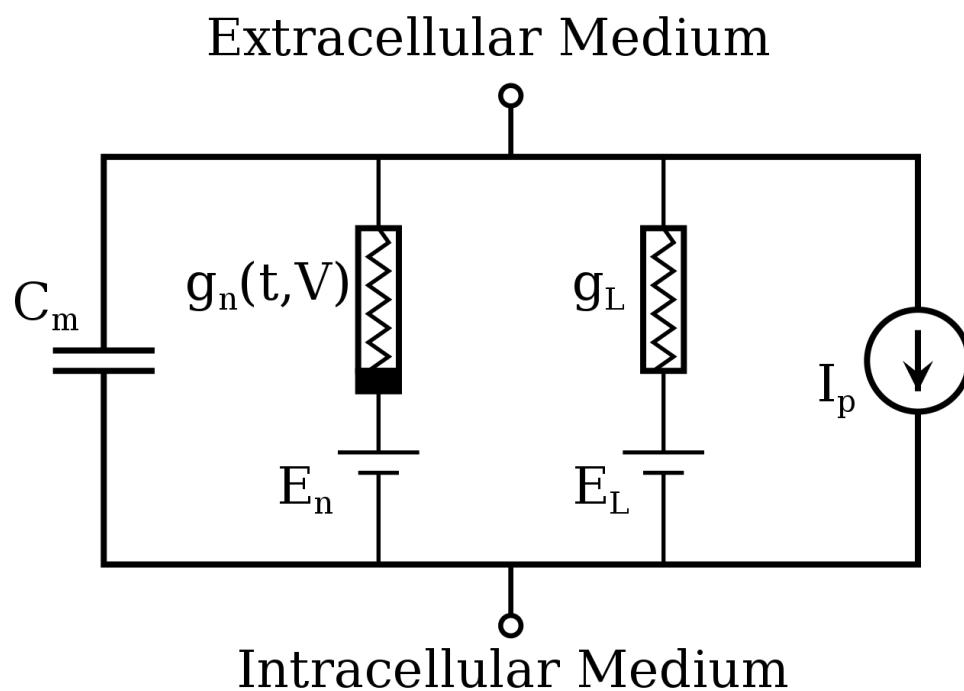


Рис. 3.5 – Базові компоненти моделі Ходжкіна-Хакслі

```
# нейронні параметри
E1 = 10.6 * b2.mV
EK = -12 * b2.mV
ENa = 115 * b2.mV
g1 = 0.3 * b2.msiemens
gK = 36 * b2.msiemens
gNa = 120 * b2.msiemens
C = 1 * b2.ufarad
```

Рис. 3.6 – Задання нейронних параметрів у кодi

Ці нейронні параметри безпосередньо використовуються у диференційному рівнянні, на основі якого будується функція симуляції.

```
def sim_hh_neuron(simulation_time):

    neuron = b2.NeuronGroup(1, eqs, method="exponential_euler")

    neuron.vm = 0
    neuron.m = 0.05
    neuron.h = 0.60
    neuron.n = 0.32

    st_mon = b2.StateMonitor(neuron, ["vm", "I_e", "m", "n", "h"], record=True)

    hh_net = b2.Network(neuron)
    hh_net.add(st_mon)
    hh_net.run(simulation_time)

    return st_mon
```

Рис. 3.7 – Фрагмент коду з функцією симуляції моделі Ходжкіна-Хакслі

Figure 1



HH Neuron, step current

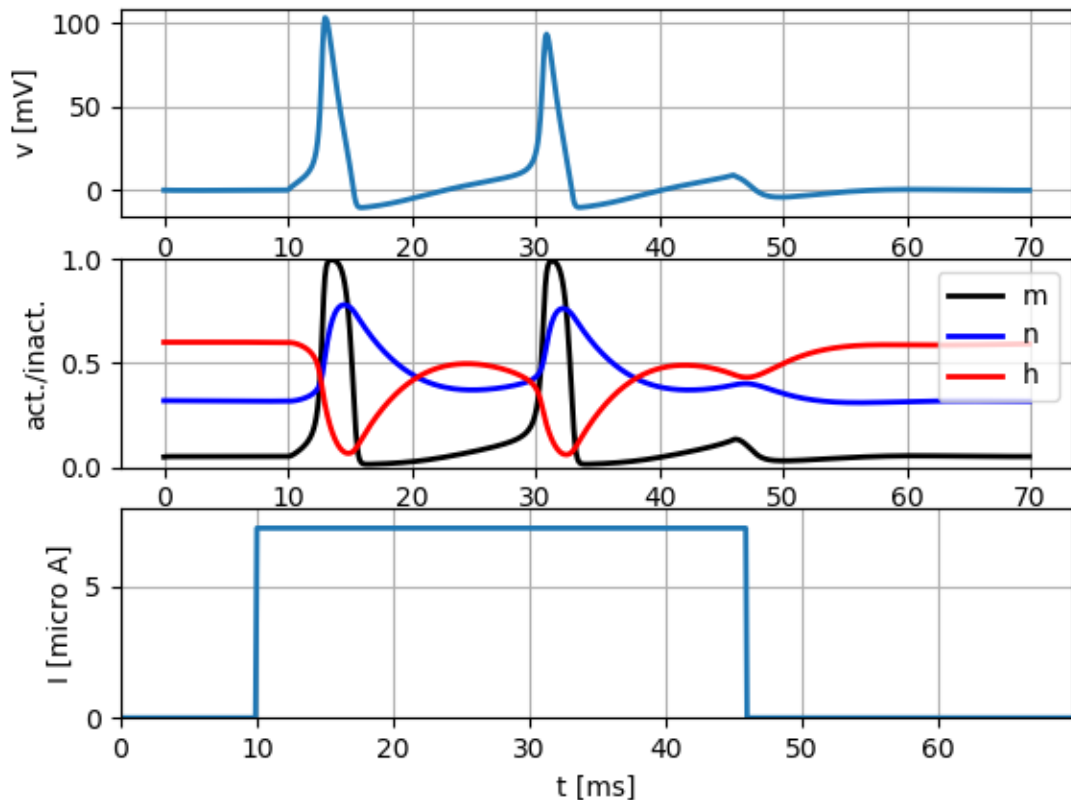


Рис. 3.8 – Нейронна динаміка моделі Ходжкіна-Хакслі

n та m – активаційні ворітні змінні для калієвих та натрієвих каналів;
 h – інактиваційна ворітна змінна для натрієвих каналів.

У моделі Ходжкіна-Хакслі передбачається, що ворітні змінні, що лежать в інтервалі від 0 до 1, відображають рух в мембрані спеціальних ворітних частинок, які рухаються незалежно один від одного і можуть призводити до закриття або відкриття іонного каналу.

Далі можна побачити поведінку моделі ще за двох умов, а саме при мінімальному струмі, а також взагалі при відсутності струму.

Figure 1

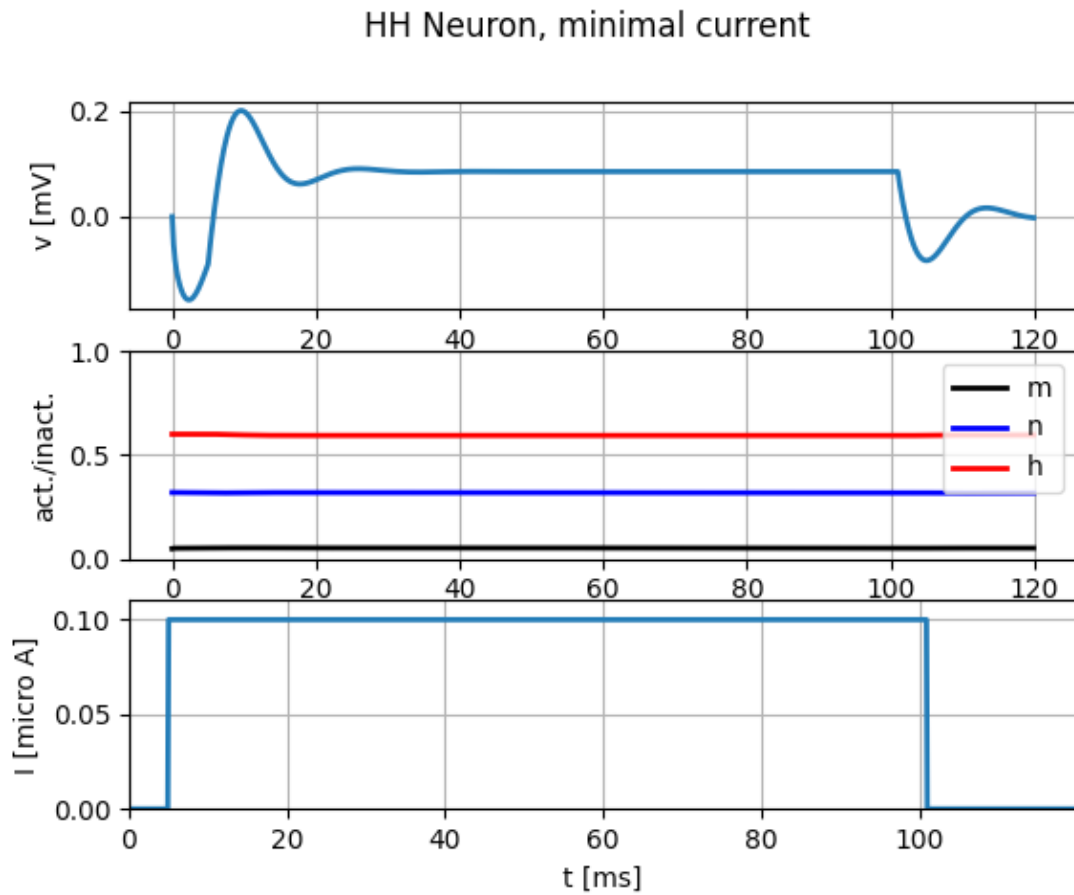


Рис. 3.9 – Нейронна динаміка моделі Ходжкіна-Хакслі при вхідному струмі 0.1 мкА

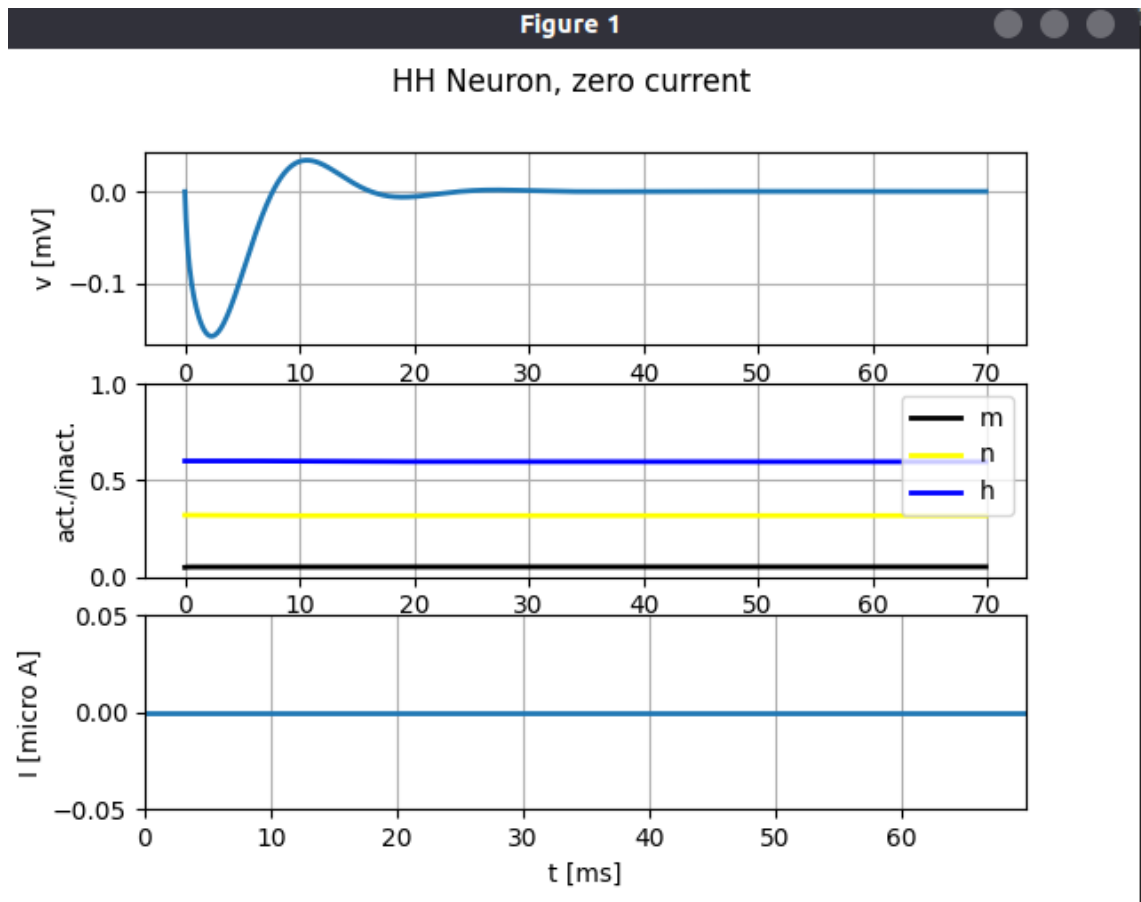


Рис. 3.10 – Нейронна динаміка моделі Ходжкіна-Хакслі при нульовому вхідному струмі

ВИСНОВКИ

В даній роботі було створено, за допомогою мови програмування Python, код, що симулює нейронну динаміку. Симуляції використовують існуючі моделі, наприклад, модель Ходжкіна-Хакслі. Базуючись на математичному апараті цих моделей, було імплементовано ці моделі в коді. Використовуючи існуючі, отримані раніше з експериментів, чисельні дані, було показано їхню нейронну динаміку, а саме виникнення потенціалів дії з усіма включними мікропроцесами. Також активацію та інактивацію каналів і залежності струму та напруги. Всі результати представлені графіками цих залежностей від часу і побудовані за допомогою бібліотеки `matplotlib`.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Amari S.-I., Yoshida K., Kanatani K.-I. A mathematical foundation for statistical neurodynamics. *SIAM journal on applied mathematics*. 1977. Vol. 33, no. 1. P. 95–126. Режим доступу: <https://doi.org/10.1137/0133008>.
2. Avery R. B., Johnston D. Multiple channel types contribute to the low-voltage-activated calcium current in hippocampal CA3 pyramidal neurons. *The journal of neuroscience*. 1996. Vol. 16, no. 18. P. 5567–5582. Режим доступу: <https://doi.org/10.1523/jneurosci.16-18-05567.1996>.
3. Ekerot C.-F., Jörntell H. Synaptic integration in cerebellar granule cells. *The cerebellum*. 2008. Vol. 7, no. 4. P. 539–541. Режим доступу: <https://doi.org/10.1007/s12311-008-0064-6>.
4. Extracting non-linear integrate-and-fire models from experimental data using dynamic I–V curves / L. Badel et al. *Biological cybernetics*. 2008. Vol. 99, no. 4-5. P. 361–370. Режим доступу: <https://doi.org/10.1007/s00422-008-0259-4>.
5. Frequency-dependent disynaptic inhibition in the pyramidal network: a ubiquitous pathway in the developing rat neocortex / T. K. Berger et al. *The journal of physiology*. 2009. Vol. 587, no. 22. P. 5411–5425. Режим доступу: <https://doi.org/10.1113/jphysiol.2009.17655>.
6. Fuortes M. G. F., Mantegazzini F. Interpretation of the repetitive firing of nerve cells. *Journal of general physiology*. 1962. Vol. 45, no. 6. P. 1163–1179. Режим доступу: <https://doi.org/10.1085/jgp.45.6.1163>.
7. Gerstner W. Time structure of the activity in neural network models. *Physical review E*. 1995. Vol. 51, no. 1. P. 738–758. Режим доступу: <https://doi.org/10.1103/physreve.51.738>.
8. Gerstner W., van Hemmen J. L., Cowan J. D. What matters in neuronal locking?. *Neural computation*. 1996. Vol. 8, no. 8. P. 1653–1676. Режим доступу: <https://doi.org/10.1162/neco.1996.8.8.1653>.
9. Hille B. *Ionic channels of excitable membranes*. 2nd ed. Sunderland, Mass : Sinauer Associates, 1992. 607 p.

10. Hodgkin A. L., Huxley A. F. Currents carried by sodium and potassium ions through the membrane of the giant axon of *Loligo*. *The journal of physiology*. 1952. Vol. 116, no. 4. P. 449–472. Режим доступа: <https://doi.org/10.1113/jphysiol.1952.sp004717>.
11. Hoppensteadt F. C., Izhikevich E. M. Weakly connected oscillators. *Weakly connected neural networks*. New York, NY, 1997. P. 247–293. Режим доступа: https://doi.org/10.1007/978-1-4612-1828-9_9.
12. Introduction. *Dynamical systems in neuroscience*. 2006. Режим доступа: <https://doi.org/10.7551/mitpress/2526.003.0002>.
13. Ion channels and the Hodgkin–Huxley model / W. Gerstner et al. *Neuronal dynamics*. Cambridge. P. 28–57. Режим доступа: <https://doi.org/10.1017/cbo9781107447615.003>.
14. Kass R. E., Ventura V. A spike-train probability model. *Neural computation*. 2001. Vol. 13, no. 8. P. 1713–1720. Режим доступа: <https://doi.org/10.1162/08997660152469314>.
15. Kepler T. B., Abbott L. F., Marder E. Reduction of conductance-based neuron models. *Biological cybernetics*. 1992. Vol. 66, no. 5. P. 381–387. Режим доступа: <https://doi.org/10.1007/bf00197717>.
16. Layer-Specific properties of the persistent sodium current in sensorimotor cortex / P. Aracri et al. *Journal of neurophysiology*. 2006. Vol. 95, no. 6. P. 3460–3468. Режим доступа: <https://doi.org/10.1152/jn.00588.2005>.
17. Prut Y., Abeles M. Time structure of cortical activity. *Central auditory processing and neural modeling*. Boston, MA, 1998. P. 199–210. Режим доступа: https://doi.org/10.1007/978-1-4615-5351-9_18.
18. Solari H. G., Natiello M. A., Mindlin G. B. Bifurcations. *Nonlinear dynamics*. 2019. P. 95–133. Режим доступа: <https://doi.org/10.1201/9780203745618-7>.
19. Yuste R., Konnerth A., Masters B. R. Imaging in neuroscience and development, A laboratory manual. *Journal of biomedical optics*. 2006. Vol. 11, no. 1. P. 019902. Режим доступа: <https://doi.org/10.1117/1.2150472>.

20. Zagrean L. Principles of neural science. Acta endocrinologica (bucharest). 2014. Vol. 10, no. 3. P. 529. Режим доступа: <https://doi.org/10.4183/aeb.2014.529>.

ДОДАТОК А

```
import brian2 as b2
import numpy as np
import matplotlib.pyplot as plt
import math
import numpy
```

```
def step_curr(t_start, t_end, u_t, amp, append_zero=True):
```

```
    if not isinstance(t_start, int) or not isinstance(t_end, int)\
        or not b2.units.fundamentalunits.have_same_dimensions(amp, b2.amp):
        return AssertionError("There is problem with input data or dimensions of
them")
```

```
    temp_size = 1 + t_end
    if append_zero:
        temp_size += 1
    temp = np.zeros((temp_size, 1)) * b2.amp
    temp[t_start: t_end + 1, 0] = amp
    current = b2.TimedArray(temp, dt=1. * u_t)
    return current
```

```
def ramp_curr(t_start, t_end, u_t, amp_s, amp_f, append_zero=True):
```

```
    if not isinstance(t_start, int) or not isinstance(t_end, int)\
        or not b2.units.fundamentalunits.have_same_dimensions(amp_s, b2.amp)\
        or not b2.units.fundamentalunits.have_same_dimensions(amp_f, b2.amp):
        return AssertionError("There is problem with input data or dimensions of
them")
```

```
    temp_size = 1 + t_end # +1 for t=0
    if append_zero:
        temp_size += 1
    temp = np.zeros((temp_size, 1)) * b2.amp
    if t_end > t_start: # if deltaT is zero, we return a zero current
        slope = (amp_f - amp_s) / float((t_end - t_start))
```

```
ramp = [amp_s + t * slope for t in range(0, (t_end - t_start) + 1)]
temp[t_start: t_end + 1, 0] = ramp
```

```
current = b2.TimedArray(temp, dt=1. * u_t)
return current
```

```
def sinus_curr(t_start, t_end, u_t,
              amp, frq, dir_curr, offset=0.,
              append_zero=True):

    if not isinstance(t_start, int) or not isinstance(t_end, int)\
    or not b2.units.fundamentalunits.have_same_dimensions(amp, b2.amp)\
    or not b2.units.fundamentalunits.have_same_dimensions(dir_curr, b2.amp):
    return AssertionError("There is problem with input data or dimensions of
them")
```

```
temp_size = 1 + t_end
if append_zero:
temp_size += 1
tmp = np.zeros((temp_size, 1)) * b2.amp
if t_end > t_start:
phi = range(0, (t_end - t_start) + 1)
phi = phi * u_t * frq
phi = phi * 2. * math.pi + offset
c = numpy.sin(phi)
c = (dir_curr + c * amp)
tmp[t_start: t_end + 1, 0] = c
curr = b2.TimedArray(tmp, dt=1. * u_t)
return curr
```

```
def zero_curr():
    return step_curr(0, 0, b2.ms, 0 * b2.amp, append_zero=False)
```

```
def sp_curr(t_sp, unit_time, ampl, append_zero=True):
```

```
    if not isinstance(t_sp, list)\
```

```
or not b2.units.fundamentalunits.have_same_dimensions(ampl, b2.amp):  
return AssertionError("There is problem with input data or dimensions of  
them")
```

```
nr_sp = len(t_sp)  
t_max = max(t_sp)  
temp_size = 1 + t_max  
if append_zero:  
temp_size += 1  
tmp = np.zeros((temp_size, nr_sp)) * b2.amp  
for i in range(nr_sp):  
tmp[t_sp[i], i] = ampl  
curr = b2.TimedArray(tmp, dt=1. * unit_time)  
return curr
```

ДОДАТОК Б

```

import matplotlib.pyplot as plt
import brian2 as b2
import numpy

def volt_and_curr_plotting(volt_mon, curr, title=None, f_threshold=None,
legend_pos=0):

    t_val = volt_mon.t / b2.ms

    axis_c = plt.subplot(211)
    c = curr(volt_mon.t, 0)
    max_curr = max(curr(volt_mon.t, 0))
    min_curr = min(curr(volt_mon.t, 0))
    mrg = 1.05 * (max_curr - min_curr)
    plt.plot(volt_mon.t / b2.ms, c, "r", lw=2)
    if mrg > 0.:
        plt.ylim((min_curr - mrg) / b2.amp, (max_curr + mrg) / b2.amp)
        plt.ylabel("Input current \n min: {} max: {}".format(min_curr, max_curr))
    plt.grid()
    axis_v = plt.subplot(212)
    plt.plot(t_val, volt_mon[0].v / b2.mV, lw=2)
    if f_threshold:
        plt.plot(
            (volt_mon.t / b2.ms)[[0, -1]],
            [f_threshold / b2.mV, f_threshold / b2.mV],
            "g--", lw=2
        )
    max_val = max(volt_mon[0].v)
    if f_threshold is not None:
        max_val = max(max_val, f_threshold)
    min_val = min(volt_mon[0].v)
    mrg = 0.05 * (max_val - min_val)
    plt.ylim((min_val - mrg) / b2.mV, (max_val + mrg) / b2.mV)
    plt.xlabel("t [ms]")
    plt.ylabel("Membrane voltage [mV]\n min: {} max: {}".format(min_val,
max_val))

```

```
plt.grid()
```

```
if f_threshold:
```

```
plt.legend(["vm", "firing threshold"], fontsize=12, loc=legend_pos)
```

```
if title:
```

```
plt.suptitle(title)
```

```
return axis_c, axis_v
```

ДОДАТОК В

```

import brian2 as b2
import matplotlib.pyplot as plt
import input_factory
import plot_factory
MEMBRANE_TIME_SCALE_tau = 12.0 * b2.ms
MEMBRANE_RESISTANCE_R = 20.0 * b2.Mohm
V_REST = -65.0 * b2.mV
V_RESET = -60.0 * b2.mV
RHEOBASE_THRESHOLD_v_rh = -55.0 * b2.mV
SHARPNESS_delta_T = 2.0 * b2.mV

FIRING_THRESHOLD_v_spike = -30. * b2.mV
eqs = """
dv/dt = -(v-v_rest) + delta_T*exp((v-v_rheobase)/delta_T)+ R * I_stim(t,i)/(tau) :
volt
"""

def int_and_fir_neuron_sim(
    v_rest=V_REST,
    simulation_time=200 * b2.ms):

    neuron = b2.NeuronGroup(1, model=eqs, reset="v=v_reset",
threshold="v>v_spike", method="euler")
    neuron.v = v_rest
    voltage_monitor = b2.StateMonitor(neuron, ["v"], record=True)
    spike_monitor = b2.SpikeMonitor(neuron)

    net = b2.Network(neuron, voltage_monitor, spike_monitor)
    net.run(simulation_time)

    return voltage_monitor, spike_monitor

input_current = input_factory.step_curr(
    t_start=20, t_end=120, unit_time=b2.ms, amplitude=0.8 * b2.namp)

st_mon, sp_mon = int_and_fir_neuron_sim(

```

```

simulation_time=200*b2.ms)

plot_factory.volt_and_curr_plotting(st_mon, input_current, title="step current",
f_threshold=FIRING_THRESHOLD_v_spike)

plt.show()

import brian2 as b2
import matplotlib.pyplot as plt
import plot_factory
import input_factory
b2.defaultclock.dt = 0.01 * b2.ms

MEMBRANE_TIME_SCALE_tau_m = 5 * b2.ms
MEMBRANE_RESISTANCE_R = 500 * b2.Mohm
V_REST = -70.0 * b2.mV
V_RESET = -51.0 * b2.mV
RHEOBASE_THRESHOLD_v_rh = -50.0 * b2.mV
SHARPNESS_delta_T = 2.0 * b2.mV
ADAPTATION_VOLTAGE_COUPLING_a = 0.5 * b2.nS
ADAPTATION_TIME_CONSTANT_tau_w = 100.0 * b2.ms
SPIKE_TRIGGERED_ADAPTATION_INCREMENT_b = 7.0 * b2.pA

FIRING_THRESHOLD_v_spike = -30. * b2.mV
eqs = """
    dv/dt = (-(v-v_rest) +delta_T*exp((v-v_rheobase)/delta_T)+ R * I_stim(t,i) - R
* w)/(tau_m) : volt
    dw/dt=(a*(v-v_rest)-w)/tau_w : amp
    """

def sim_ad_ex_neuron(
    v_rest=V_REST,
    v_spike=FIRING_THRESHOLD_v_spike,
    simulation_time=200 * b2.ms):

    threshold = "v>{:f}*mvolt".format(v_spike / b2.mvolt)

```

```

neuron = b2.NeuronGroup(1, model=eqs, threshold=threshold,
reset="v=v_reset;w+=b", method="euler")

neuron.v = v_rest
neuron.w = 0.0 * b2.pA

st_mon = b2.StateMonitor(neuron, ["v", "w"], record=True)
sp_mon = b2.SpikeMonitor(neuron)

# running simulation
b2.run(simulation_time)
return st_mon, sp_mon

def plot_adex_state(adex_state_monitor):
    """
    Visualizes the state variables: w-t, v-t and phase-plane w-v
    Args:
    adex_state_monitor (StateMonitor): States of "v" and "w"
    """
    plt.subplot(2, 2, 1)
    plt.plot(adex_state_monitor.t / b2.ms, adex_state_monitor.v[0] / b2.mV, lw=2)
    plt.xlabel("t [ms]")
    plt.ylabel("u [mV]")
    plt.title("Membrane potential")
    plt.subplot(2, 2, 2)
    plt.plot(adex_state_monitor.v[0] / b2.mV, adex_state_monitor.w[0] / b2.pA,
lw=2)
    plt.xlabel("u [mV]")
    plt.ylabel("w [pAmp]")
    plt.title("Phase plane representation")
    plt.subplot(2, 2, 3)
    plt.plot(adex_state_monitor.t / b2.ms, adex_state_monitor.w[0] / b2.pA, lw=2)
    plt.xlabel("t [ms]")
    plt.ylabel("w [pAmp]")
    plt.title("Adaptation current")
    plt.show()

```

```

current = input_factory.step_curr(10, 200, 1. * b2.ms, 65.0 * b2.pA)
state_monitor, spike_monitor = sim_ad_ex_neuron(simulation_time=300 * b2.ms)
plot_factory.volt_and_curr_plotting(state_monitor, current)
plot_adex_state(state_monitor)

```

```

import brian2 as b2
import matplotlib.pyplot as plt
import numpy as np
import matplotlib
import input_factory
import tkinter

```

```

matplotlib.use('TkAgg')

```

```

# нейронні параметри

```

```

E1 = 10.6 * b2.mV

```

```

EK = -12 * b2.mV

```

```

ENa = 115 * b2.mV

```

```

g1 = 0.3 * b2.msiemens

```

```

gK = 36 * b2.msiemens

```

```

gNa = 120 * b2.msiemens

```

```

C = 1 * b2.ufarad

```

```

input_current = input_factory.step_curr(10, 45, b2.ms, 0 * b2.uA)

```

```

# forming HH model with differential equations

```

```

eqs = """

```

```

I_e = input_current(t,i) : amp

```

```

membrane_Im = I_e + gNa*m**3*h*(ENa-vm) + \
    g1*(E1-vm) + gK*n**4*(EK-vm) : amp

```

```

alphah = .07*exp(-.05*vm/mV)/ms : Hz

```

```

alphan = .1*(25*mV-vm)/(exp(2.5-.1*vm/mV)-1)/mV/ms : Hz

```

```

alphan = .01*(10*mV-vm)/(exp(1-.1*vm/mV)-1)/mV/ms : Hz

```

```

betah = 1./(1+exp(3.-.1*vm/mV))/ms : Hz

```

```

betam = 4*exp(-.0556*vm/mV)/ms : Hz

```

```

betan = .125*exp(-.0125*vm/mV)/ms : Hz

```

```

dh/dt = alphah*(1-h)-betah*h : 1

```

```

dm/dt = alphan*(1-m)-betam*m : 1

```

```

dn/dt = alphan*(1-n)-betan*n : 1

```

```
dvm/dt = membrane_Im/C : volt
```

```
"""
```

```
def plot_data(state_monitor, title=None):
```

```
    """Plots the state_monitor variables ["vm", "I_e", "m", "n", "h"] vs. time.
```

```
    Args:
```

```
    state_monitor (StateMonitor): the data to plot
```

```
    title (string, optional): plot title to display
```

```
    """
```

```
    plt.subplot(311)
```

```
    plt.plot(state_monitor.t / b2.ms, state_monitor.vm[0] / b2.mV, lw=2)
```

```
    plt.xlabel("t [ms]")
```

```
    plt.ylabel("v [mV]")
```

```
    plt.grid()
```

```
    plt.subplot(312)
```

```
    plt.plot(state_monitor.t / b2.ms, state_monitor.m[0] / b2.volt, "black", lw=2)
```

```
    plt.plot(state_monitor.t / b2.ms, state_monitor.n[0] / b2.volt, "yellow", lw=2)
```

```
    plt.plot(state_monitor.t / b2.ms, state_monitor.h[0] / b2.volt, "blue", lw=2)
```

```
    plt.xlabel("t (ms)")
```

```
    plt.ylabel("act./inact.")
```

```
    plt.legend(("m", "n", "h"))
```

```
    plt.ylim((0, 1))
```

```
    plt.grid()
```

```
    plt.subplot(313)
```

```
    plt.plot(state_monitor.t / b2.ms, state_monitor.I_e[0] / b2.uamp, lw=2)
```

```
    plt.axis((
```

```
    0,
```

```
    np.max(state_monitor.t / b2.ms),
```

```
    min(state_monitor.I_e[0] / b2.uamp) * 1.1,
```

```
    max(state_monitor.I_e[0] / b2.uamp) * 1.1
```

```
    ))
```

```
    plt.xlabel("t [ms]")
```

```
plt.ylabel("I [micro A]")  
plt.grid()
```

```
if title is not None:  
plt.suptitle(title)
```

```
plt.show()
```

```
def sim_hh_neuron(simulation_time):
```

```
    neuron = b2.NeuronGroup(1, eqs, method="exponential_euler")
```

```
    neuron.vm = 0
```

```
    neuron.m = 0.05
```

```
    neuron.h = 0.60
```

```
    neuron.n = 0.32
```

```
    st_mon = b2.StateMonitor(neuron, ["vm", "I_e", "m", "n", "h"], record=True)
```

```
    hh_net = b2.Network(neuron)
```

```
    hh_net.add(st_mon)
```

```
    hh_net.run(simulation_time)
```

```
    return st_mon
```

```
state_monitor = sim_hh_neuron(70 * b2.ms)
```

```
plot_data(state_monitor, title="HH Neuron, step current")
```

```
plt.show()
```