

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное
образовательное учреждение высшего образования
«Национальный исследовательский университет ИТМО»
(Университет ИТМО)

Факультет систем управления и робототехники

ОТЧЕТ
по производственной (научно-исследовательской) практике:

МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ ДИНАМИКИ
БИОЛОГИЧЕСКИХ НЕЙРОНОВ

Студент:
Группа № R3338

A.A. Нечаева

Руководитель практики:
ассистент факультета СУиР, к. ф.-м. н.

Д.М. Семенов

Санкт-Петербург 2025

СОДЕРЖАНИЕ

1 ЦЕЛИ И ЗАДАЧИ РАБОТЫ	3
1.1 Цель работы	3
1.2 Задачи	3
2 МАТЕМАТИЧЕСКАЯ МОДЕЛЬ БИОЛОГИЧЕСКОГО НЕЙРОНА..	4
2.1 Модель Ижикевича	4
2.2 Параметры модели	4
2.3 Виды нейронов.....	5
2.3.1 Регулярно-спайковые (Regular spiking).....	5
2.3.2 Быстро-спайковые (Fast spiking)	5
2.3.3 Низкопороговые спайковые (Low-threshold spiking).....	6
2.3.4 Резонансные (Resonator)	6
2.3.5 Внутренне разрывные (Intrinsically bursting)	6
2.3.6 Чаттерные (Chattering)	6
3 МОДЕЛИРОВАНИЕ.....	7
3.1 Визуализация.....	7
3.2 Анализ результатов	16
3.3 Листинг кода	16
4 ПОСТРОЕНИЕ СЕТИ ИЗ ДВУХ СВЯЗАННЫХ НЕЙРОНОВ	20
5 ПРОВЕДЕНИЕ МОДЕЛИРОВАНИЯ ПРИ РАЗЛИЧНЫХ ЗНАЧЕНИЯХ СИЛЫ СВЯЗИ	21
5.1 Визуализация.....	21
5.2 Анализ результатов	22
5.3 Листинг кода	22
6 ЗАКЛЮЧЕНИЕ	25
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	26

1 ЦЕЛИ И ЗАДАЧИ РАБОТЫ

1.1 Цель работы

Выполнить математическое моделирование динамики отдельных нейронов и сети из двух нейронов.

1.2 Задачи

1. Выбрать модель биологического нейрона для выполнения математического моделирования.
2. Изучить параметры модели и их связь с электрическими процессами в биологических нейронах.
3. Провести численное моделирование выбранной модели при различных значениях параметров.
4. Построить сети из двух связанных нейронов.
5. Провести моделирования при различных значениях силы связи.

2 МАТЕМАТИЧЕСКАЯ МОДЕЛЬ БИОЛОГИЧЕСКОГО НЕЙРОНА

2.1 Модель Ижикевича

Нейронная модель Ижикевича описывает изменение электрического потенциала в мембране нейрона в зависимости от тока, протекающего через ионные каналы мембранны. Изменения электрического потенциала представлены следующими дифференциальными уравнениями

$$\begin{cases} \frac{dv}{dt} = 0.04v^2 + 5v + 140 - u + I, \\ \frac{du}{dt} = a(bv - u), \end{cases} \quad (1)$$

вспомогательный сброс

$$\text{если } v \geq 30 \text{ mV, то } \begin{cases} v \leftarrow c \\ u \leftarrow u + d, \end{cases} \quad (2)$$

где v – мембранный потенциал нейрона, u – вспомогательная переменная, которая восстанавливает мембранный потенциал: отвечает за активацию ионных токов K^+ и инактивацию ионных токов Na^+ и обеспечивает отрицательную обратную связь с v [1].

Спайк (или потенциал действия) – это кратковременное (1-2 мс) резкое увеличение мембранныго потенциала нейрона (v), которое возникает при достижении порогового значения и служит для передачи информации между нейронами. В модели Ижикевича спайки формализованы через пороговый сброс переменных v и u .

2.2 Параметры модели

Параметр a отвечает за временнной масштаб переменной восстановления u . Чем меньше значение a , тем медленнее восстановление. Значение по умолчанию $a = 0.02$ [1].

Параметр b характеризует чувствительность переменной восстановления u к подпороговым колебаниям мембранныго потенциала v . При увеличении значения параметра b возрастает и сила связи v и u , что приводит к

возможным подпороговым колебаниям и низкопороговой динамике спайков. Значение по умолчанию $b = 0.2$ [1].

Параметр c соответствует значению сброса мембранныго потенциала v после скачка, вызванного быстрыми высокопороговыми проводимостями K^+ . Значение по умолчанию $c = -65$ мВ [1].

Параметр d характеризует сброс переменной восстановления u после скачка, вызванный медленными высокопороговыми проводимостями Na^+ и K^+ . Значение по умолчанию $d = 2$ [1].

С помощью вариации параметров a, b, c, d можно добиться моделирования различных типов нейронов.

2.3 Виды нейронов

Рассмотрим модели нейронов, которые можно получить при определенных значениях параметров.

2.3.1 Регулярно-спайковые (Regular spiking)

Воссоздают динамику стандартных возбуждающихся нейронов, например, пирамидных нейронов коры. Параметры модели Ижикевича в данном случае: $a = 0.02$, $b = 0.2$, $c = -65$, $d = 8$. Возможные начальные условия: $v_0 = -65$, $u_0 = b \cdot v_0 = -13$, значение тока $I \in [5; 15]$ нА. При постоянном стимуле генерируют регулярные спайки с постепенным увеличением интервала между спайками.

2.3.2 Быстро-спайковые (Fast spiking)

Моделируют быстрые ГАМК-ergicические интернейроны, например, корзинчатые клетки. Параметры модели Ижикевича: $a = 0.1$, $b = 0.2$, $c = -65$, $d = 2$. Начальные условия: $v_0 = -70$, $u_0 = b \cdot v_0 = -14$, ток $I \in [10; 20]$ нА. В данном случае формируются высокочастотные спайки, причем увеличения интервала между спайками на происходит.

2.3.3 Низкопороговые спайковые (Low-threshold spiking)

Моделируют низкопороговые интернейроны, например, соматостатин-положительные. Параметры модели Ижикевича: $a = 0.02$, $b = 0.25$, $c = -65$, $d = 2$. Начальные условия: $v_0 = -70$, $u_0 = b \cdot v_0 = -17.5$, ток $I \in [5; 10]$ нА. С точки зрения динамики характерна высокая частота спайков в начале и быстрое увеличение интервала между спайками с ростом времени.

2.3.4 Резонансные (Resonator)

Симулируют нейроны с резонансными свойствами, например, таламические релейные нейроны. Параметры модели Ижикевича: $a = 0.1$, $b = 0.26$, $c = -65$, $d = 2$. Начальные условия: $v_0 = -65$, $u_0 = b \cdot v_0 = -16.9$, ток $I \in [5; 15]$ нА. Генерация спайков только на определенных частотах стимула.

2.3.5 Внутренне разрывные (Intrinsically bursting)

Нейроны с аутохтонными всплесками, например, в таламусе. Параметры модели Ижикевича: $a = 0.02$, $b = 0.2$, $c = -55$, $d = 4$. Начальные условия: $v_0 = -60$, $u_0 = b \cdot v_0 = -12$, ток $I \in [5; 15]$. Вначале высокочастотный участок спайков, затем формируются автоматические пачки спайков даже без стимула.

2.3.6 Чаттерные (Chattering)

Описываемая динамика близка быстроразряжающимся кортикальным нейронам, находящимся в зрительной коре и других областях неокортекса. Параметры модели Ижикевича: $a = 0.02$, $b = 0.2$, $c = -50$, $d = 2$. Начальные условия: $v_0 = -65$, $u_0 = b \cdot v_0 = -13$, ток $I \in [5; 10]$. Характерно формирование высокочастотных пачек спаек.

3 МОДЕЛИРОВАНИЕ

3.1 Визуализация

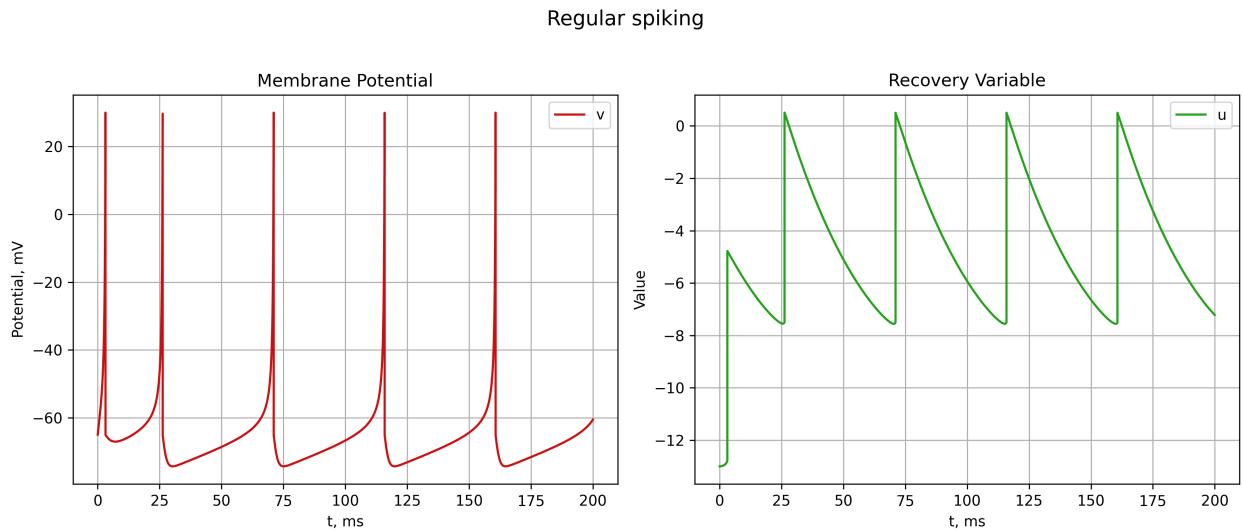


Рисунок 1 — Визуализация регулярно-спайкового нейрона при $I = 10$ нА.

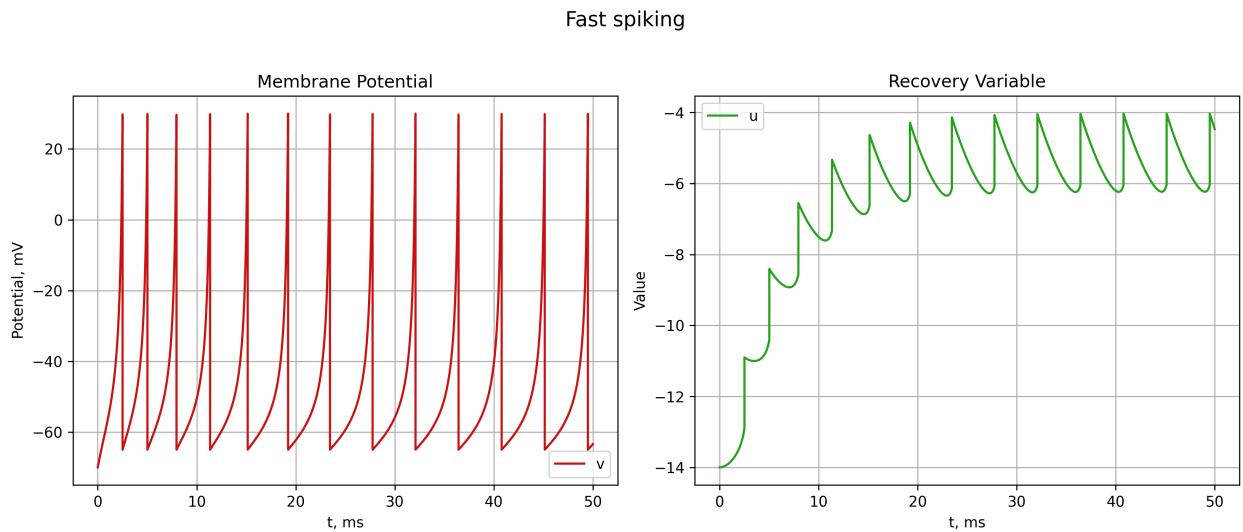


Рисунок 2 — Визуализация быстро-спайкового нейрона при $I = 15$ нА.

Low-threshold spiking

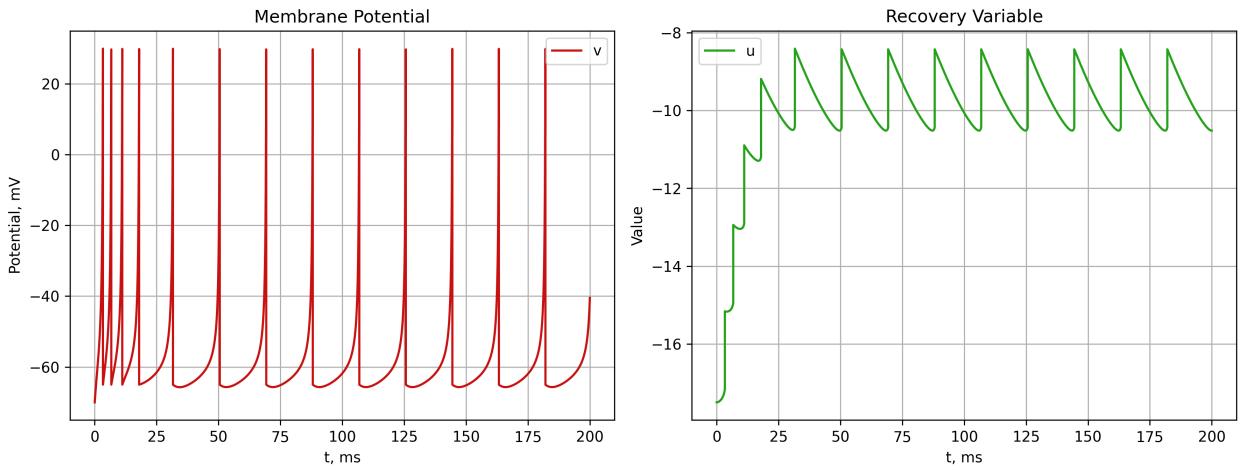


Рисунок 3 — Визуализация низкопорогового спайкового нейрона при $I = 7$ нА.

Resonator

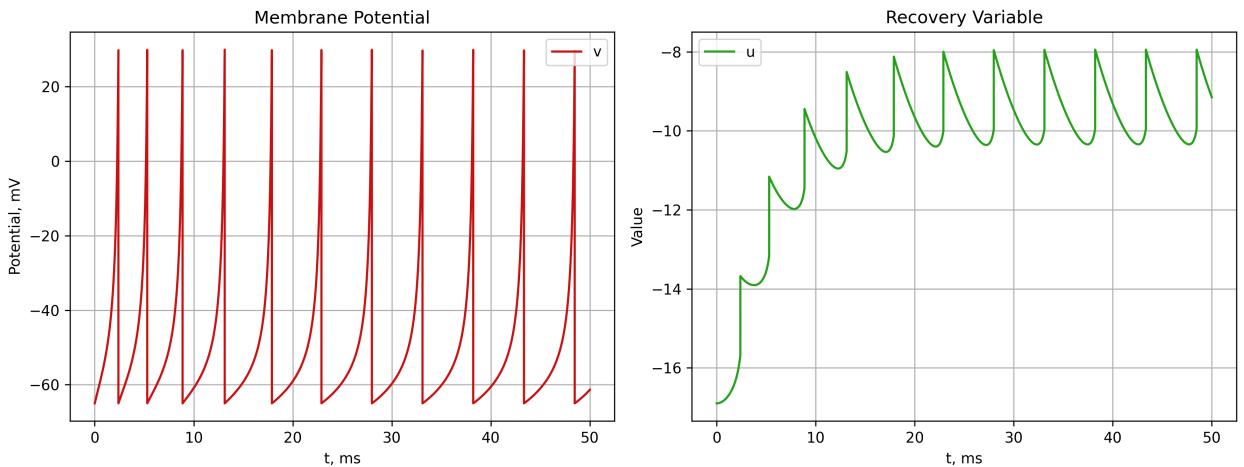


Рисунок 4 — Визуализация резонансного нейрона при $I = 10$ нА.

Intrinsically bursting

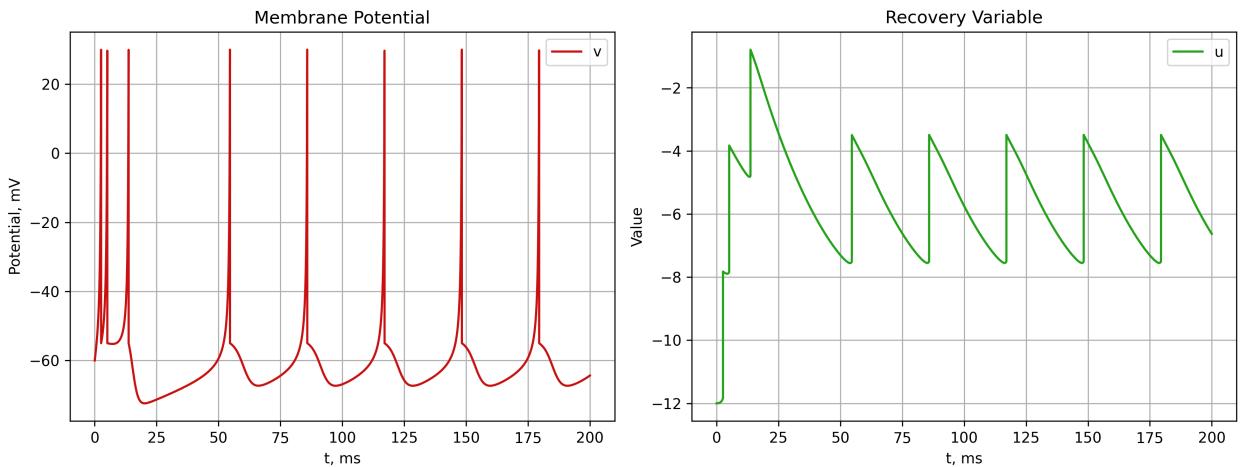


Рисунок 5 — Визуализация внутренне разрывного нейрона при $I = 10$ нА.

Chattering

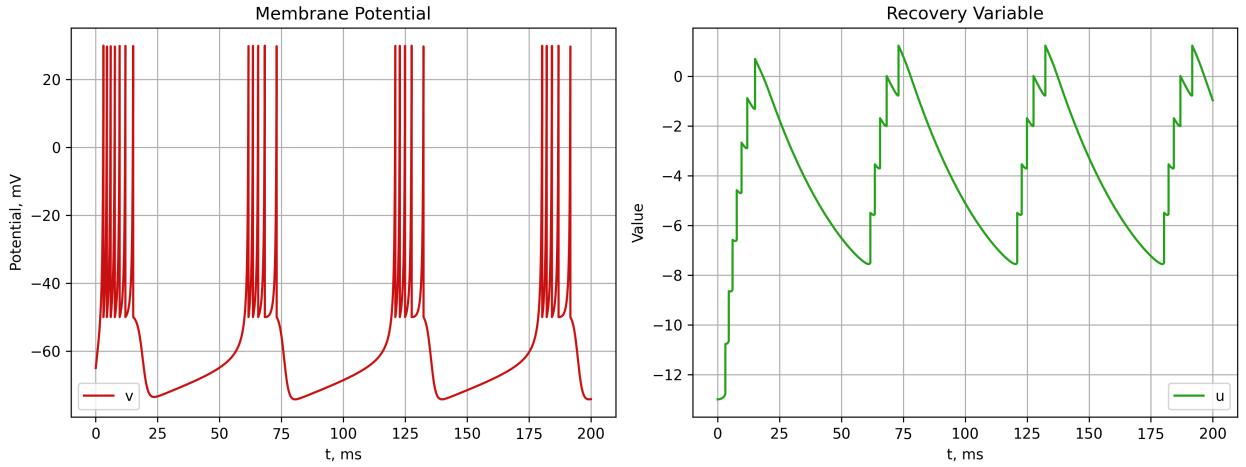


Рисунок 6 — Визуализация чаттерного нейрона при $I = 10$ нА.

Regular spiking: Membrane Potential

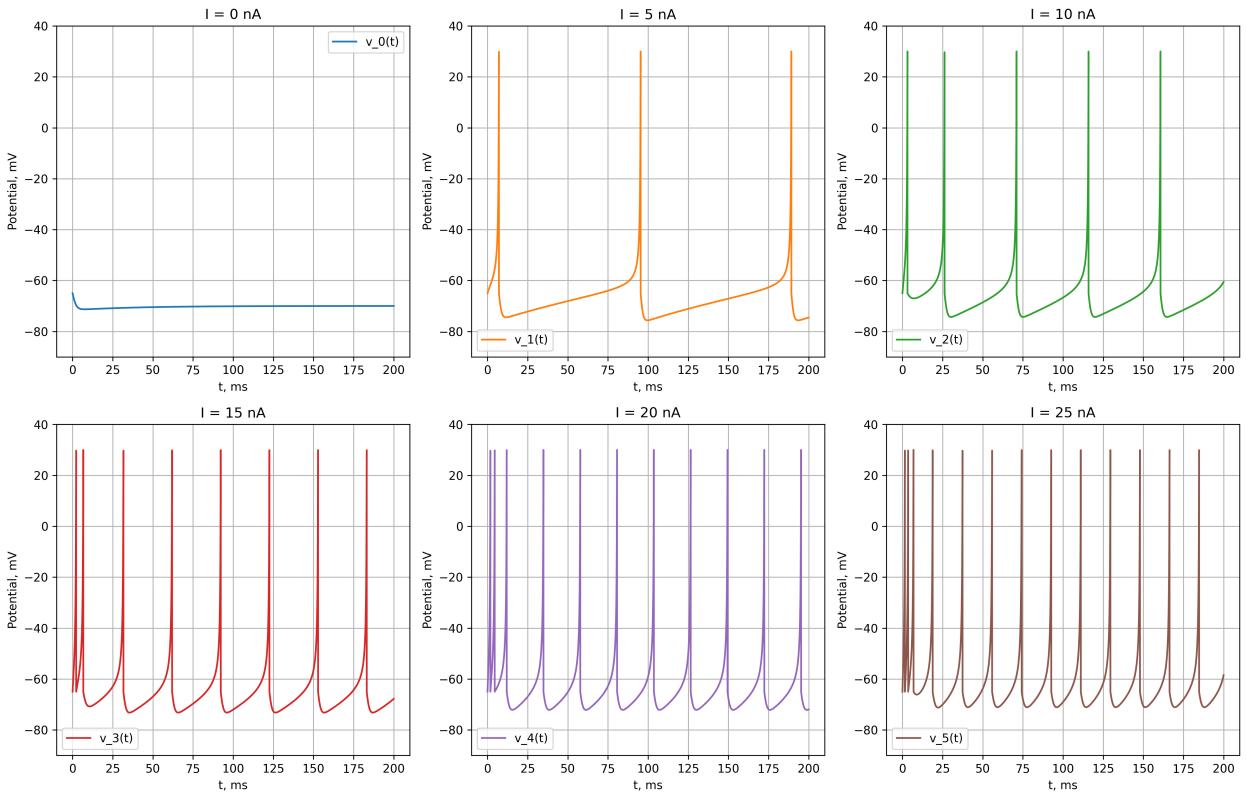


Рисунок 7 — Графики $v(t)$ регулярно-спайкового нейрона для разных значений I .

Regular spiking: Recovery Variable

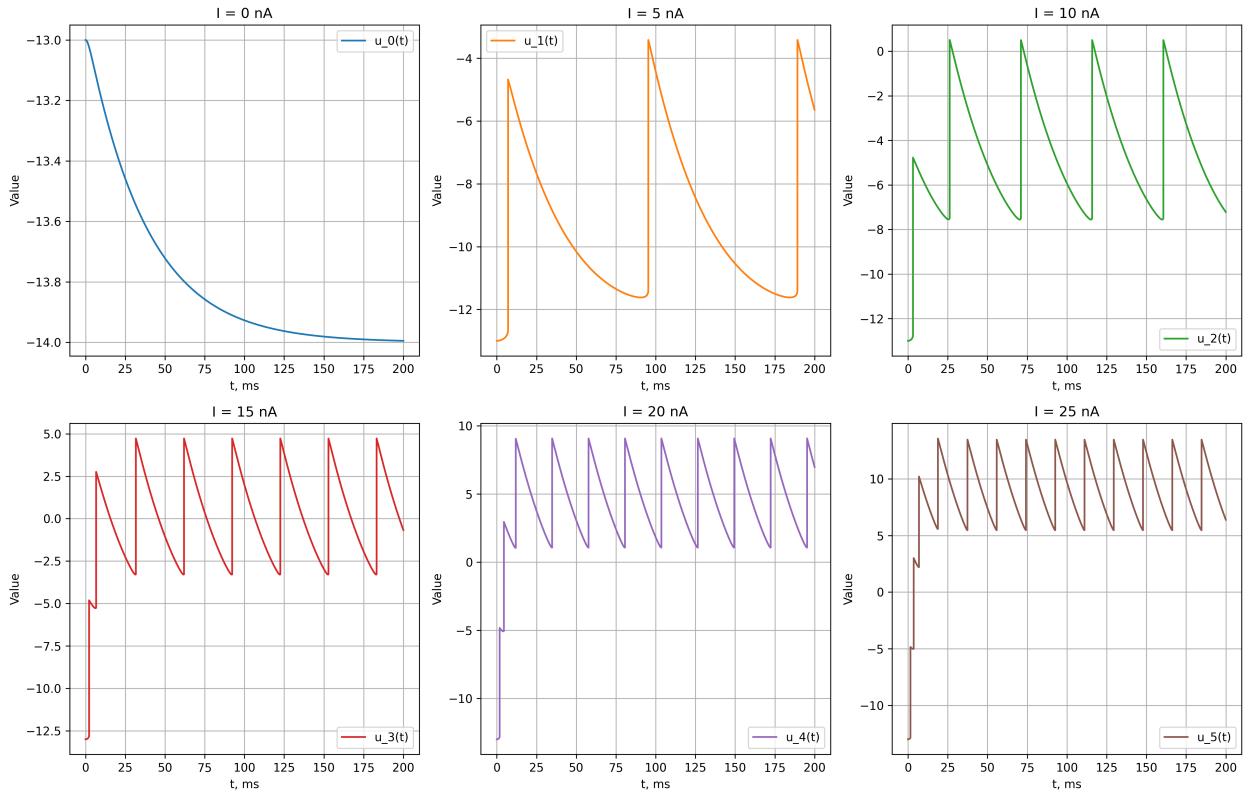


Рисунок 8 — Графики $u(t)$ регулярно-спайкового нейрона для разных значений I .

Fast spiking: Membrane Potential

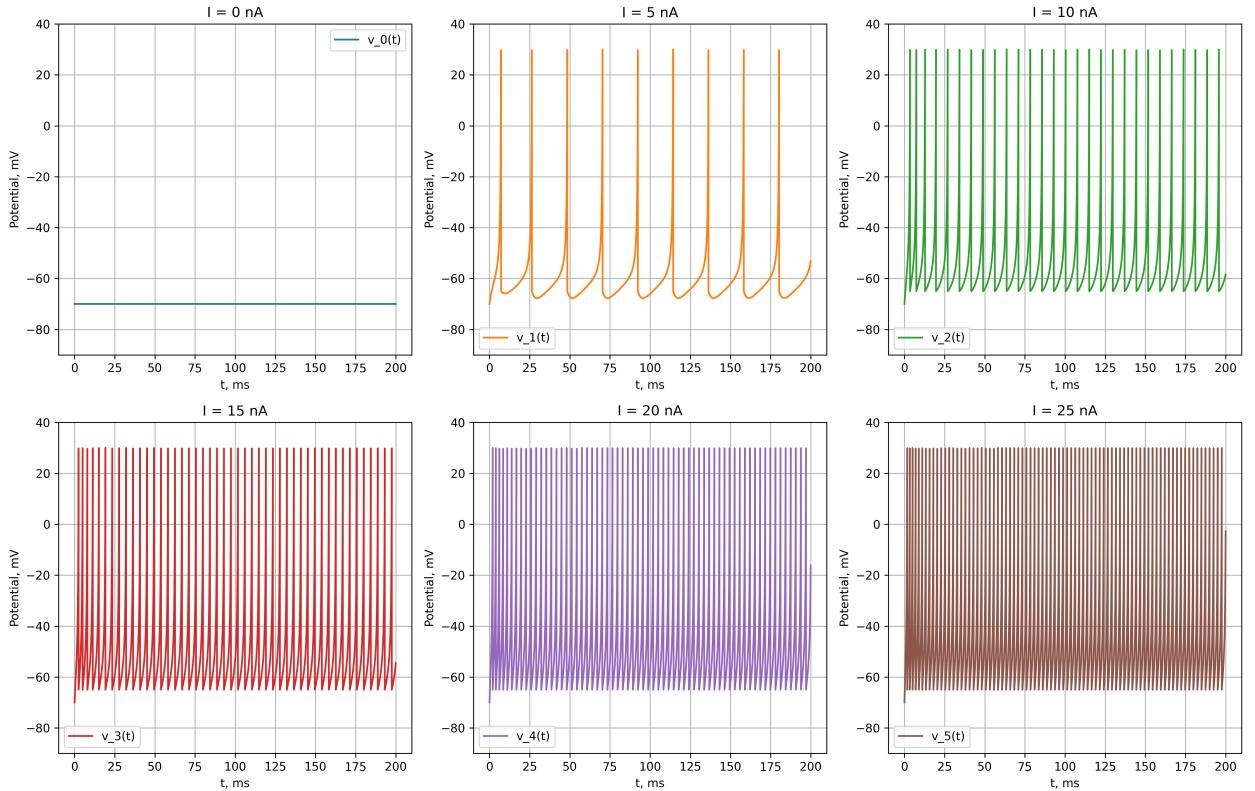


Рисунок 9 — Графики $v(t)$ быстро-спайкового нейрона для разных значений I .

Fast spiking: Recovery Variable

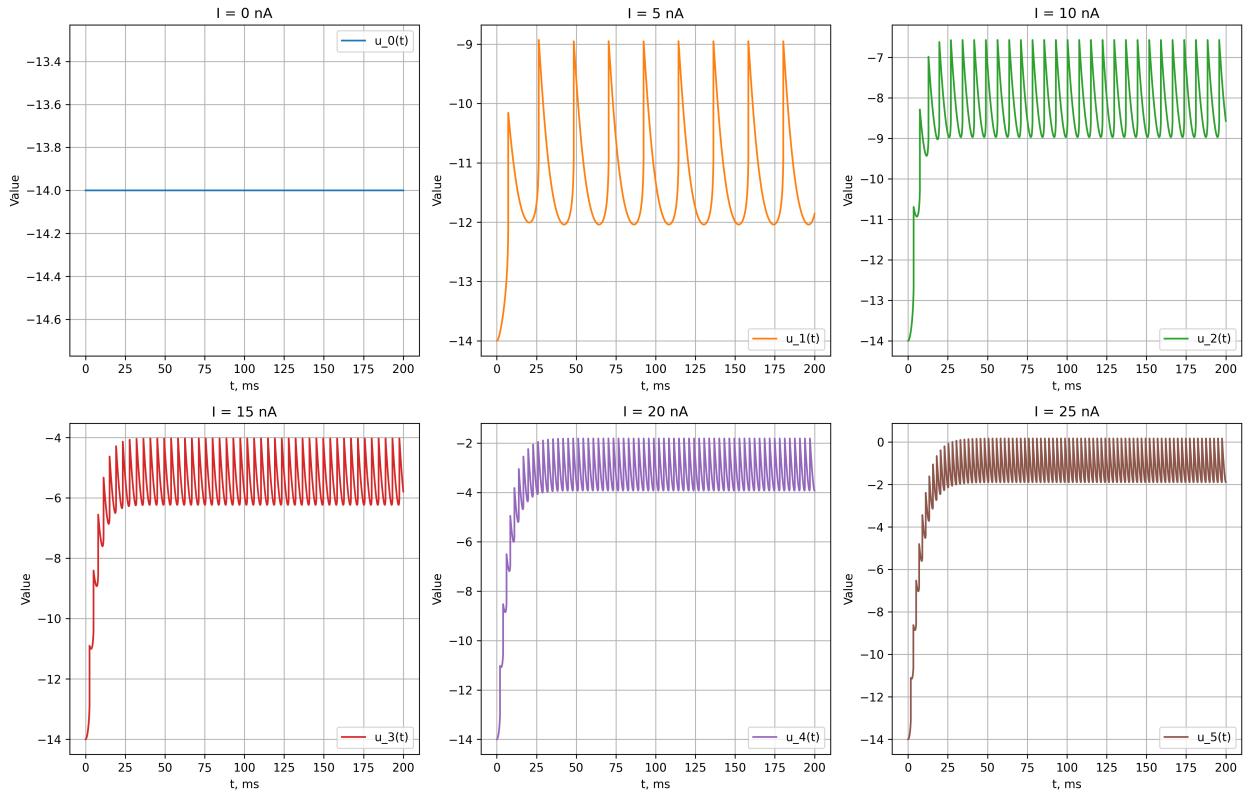


Рисунок 10 — Графики $u(t)$ быстро-спайкового нейрона для разных значений I .

Low-threshold spiking: Membrane Potential

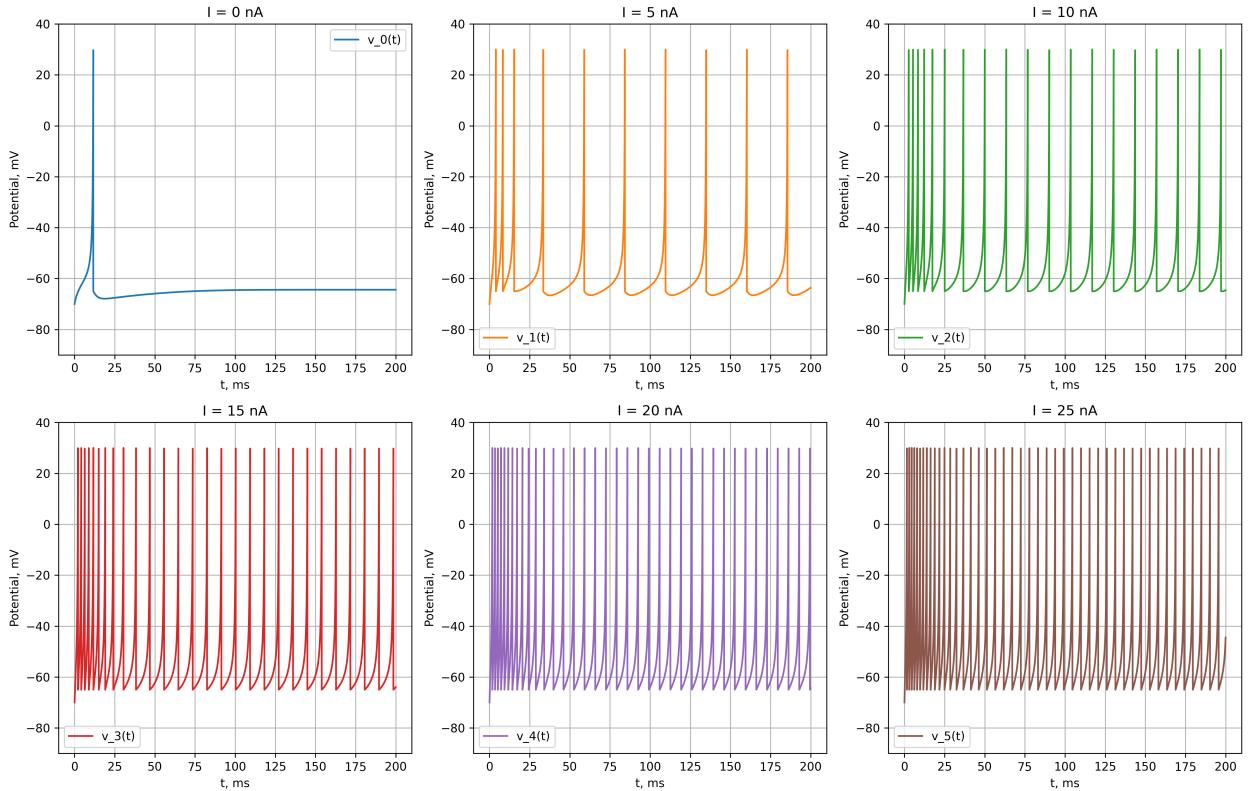


Рисунок 11 — Графики $v(t)$ низкопорогового спайкового нейрона для разных значений I .

Low-threshold spiking: Recovery Variable

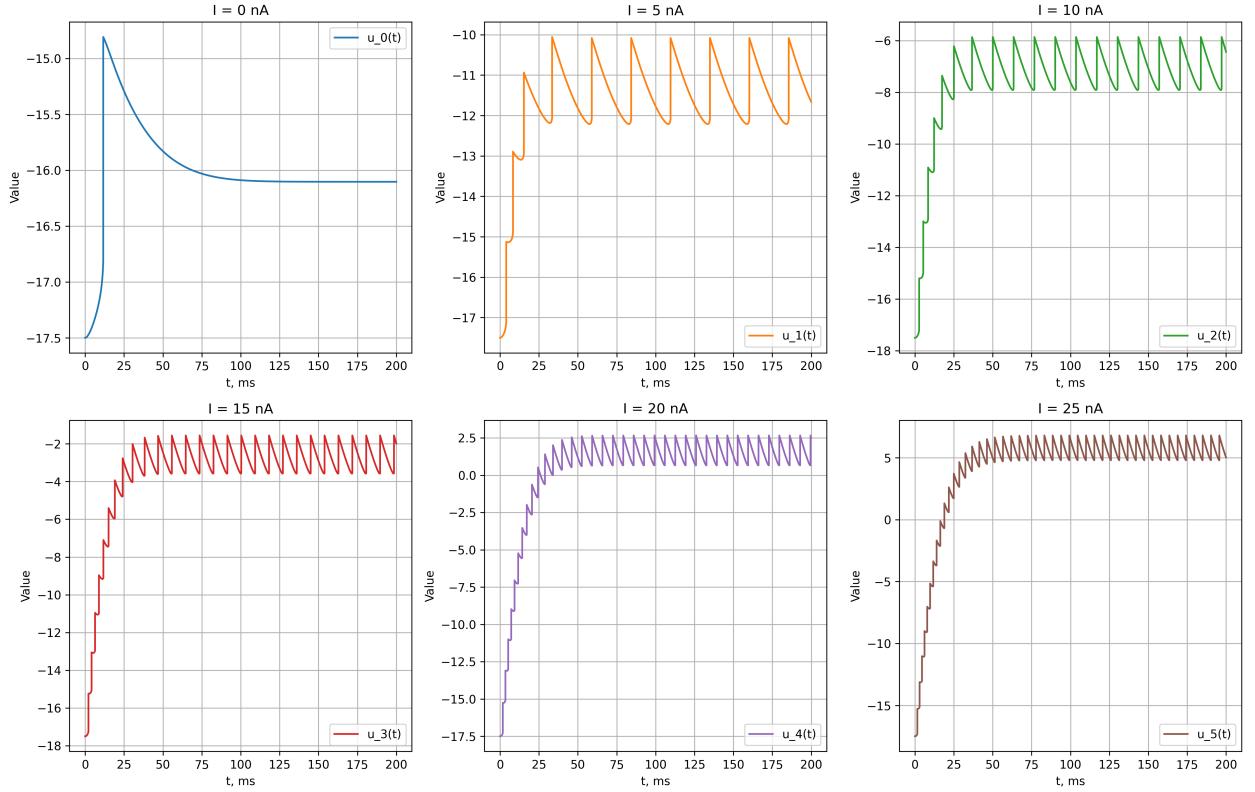


Рисунок 12 — Графики $u(t)$ низкопорогового спайкового нейрона для разных значений I .

Resonator: Membrane Potential

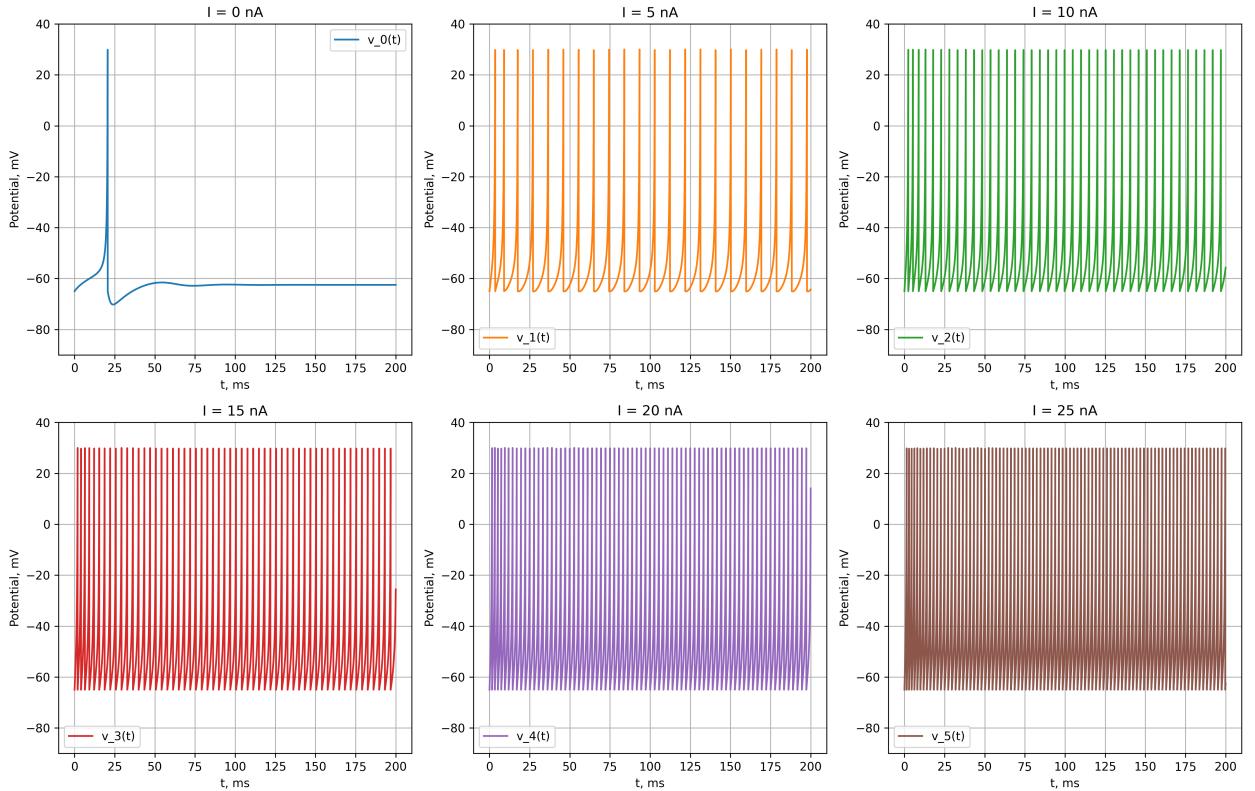


Рисунок 13 — Графики $v(t)$ резонансного нейрона для разных значений I .

Resonator: Recovery Variable

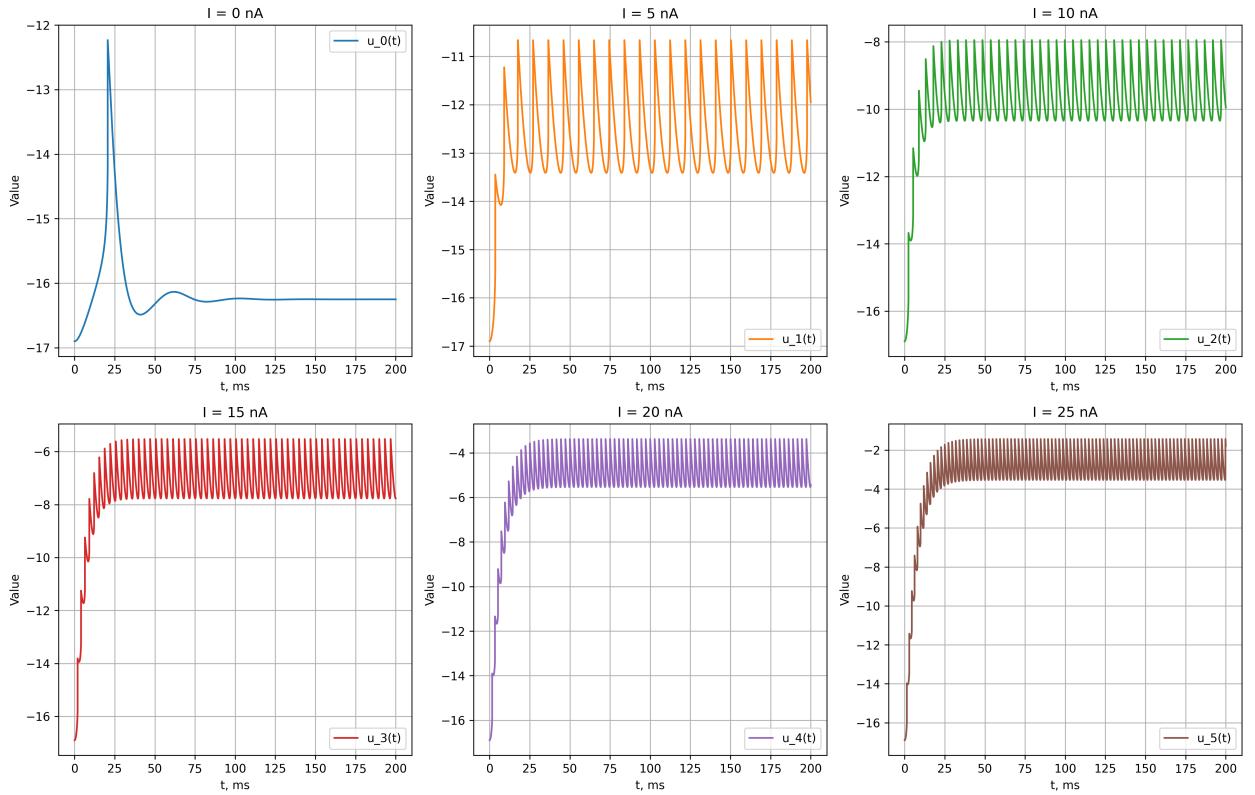


Рисунок 14 — Графики $u(t)$ резонансного нейрона для разных значений I .

Intrinsically bursting: Membrane Potential

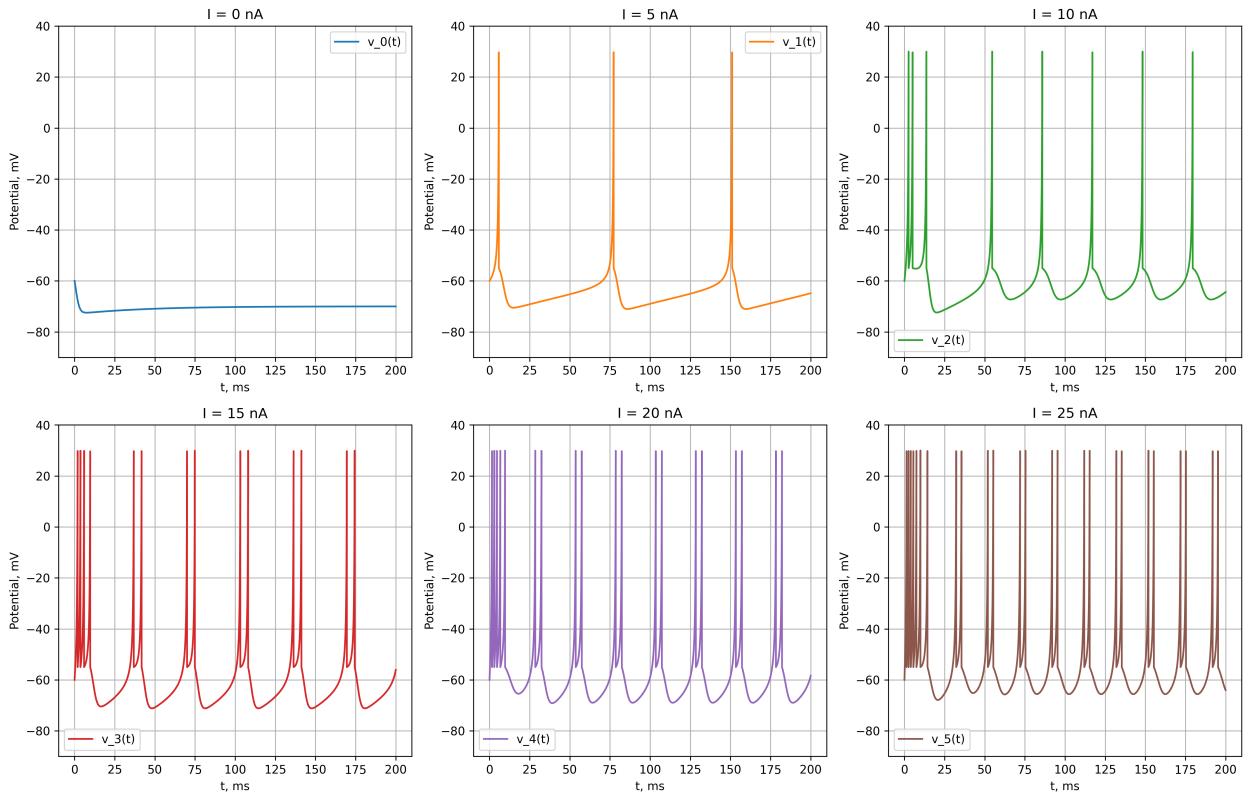


Рисунок 15 — Графики $v(t)$ внутренне разрывного нейрона для разных значений I .

Intrinsically bursting: Recovery Variable

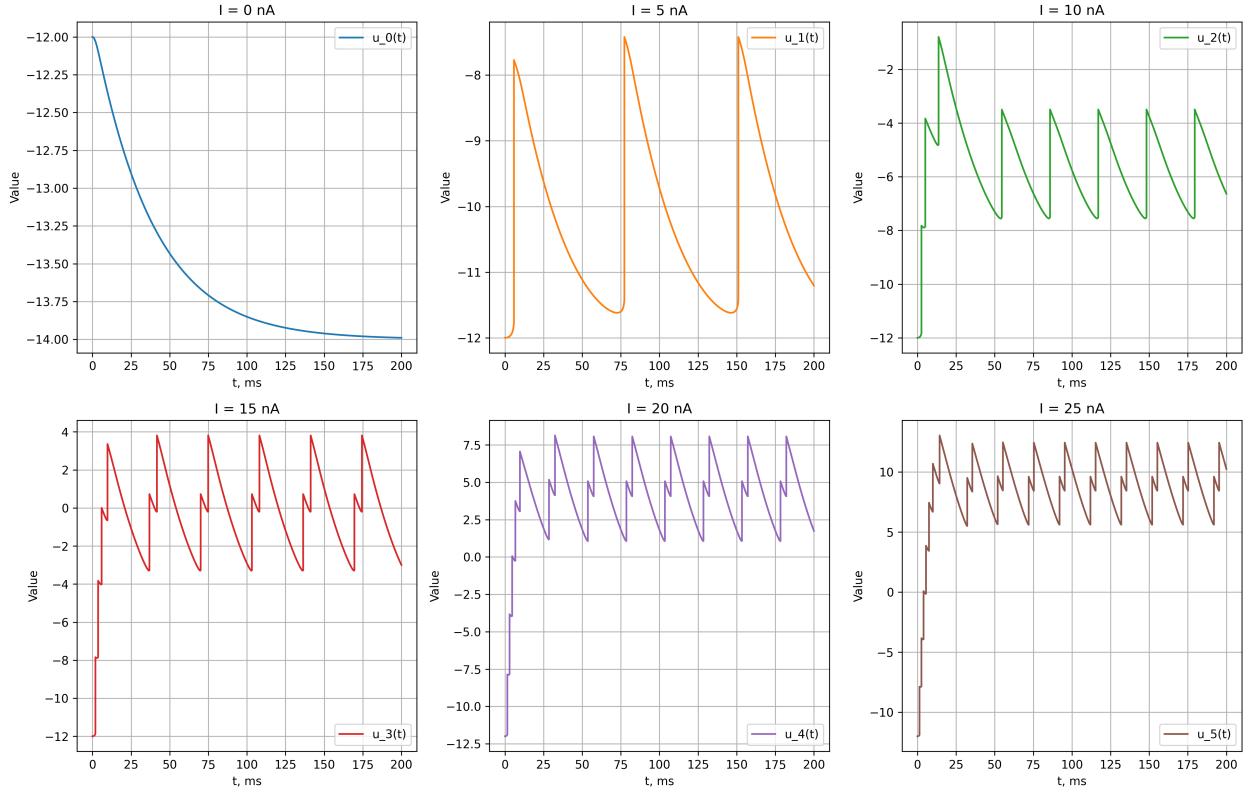


Рисунок 16 — Графики $u(t)$ внутренне разрывного нейрона для разных значений I .

Chattering: Membrane Potential

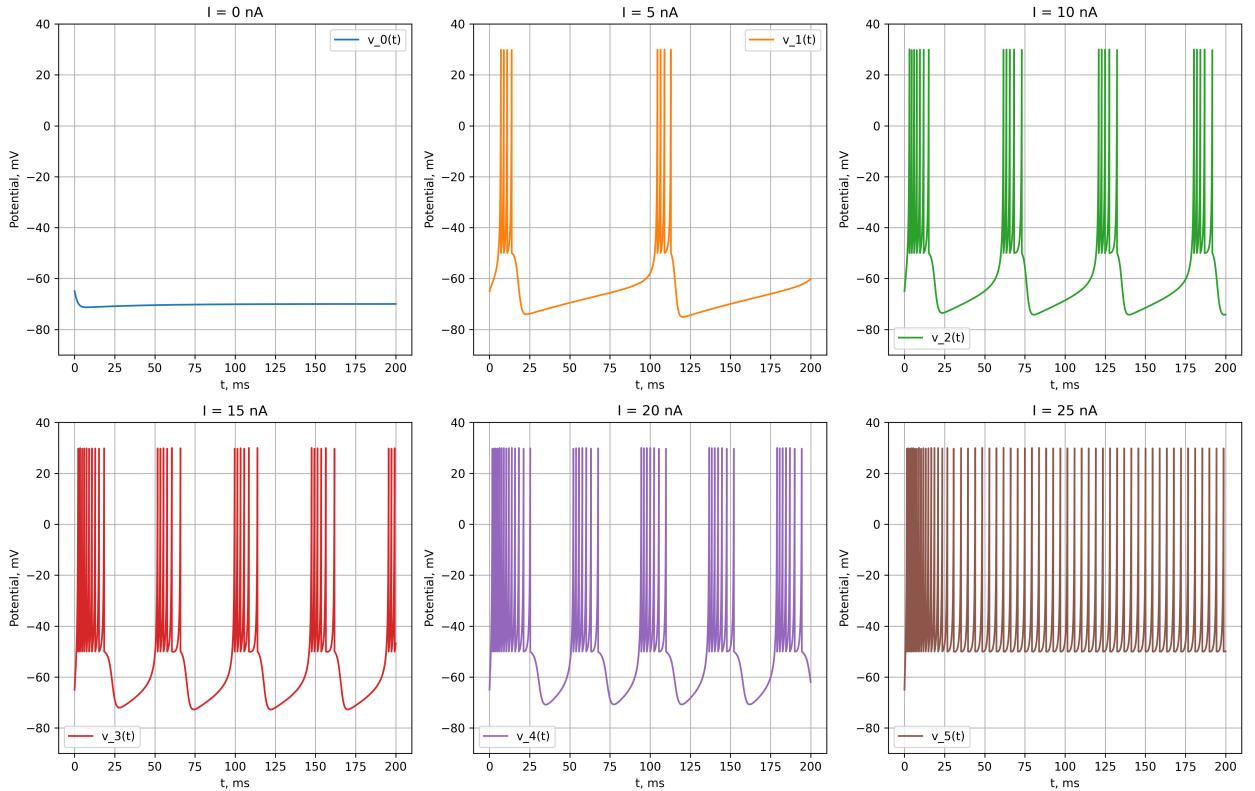


Рисунок 17 — Графики $v(t)$ чаттерного нейрона для разных значений I .

Chattering: Recovery Variable

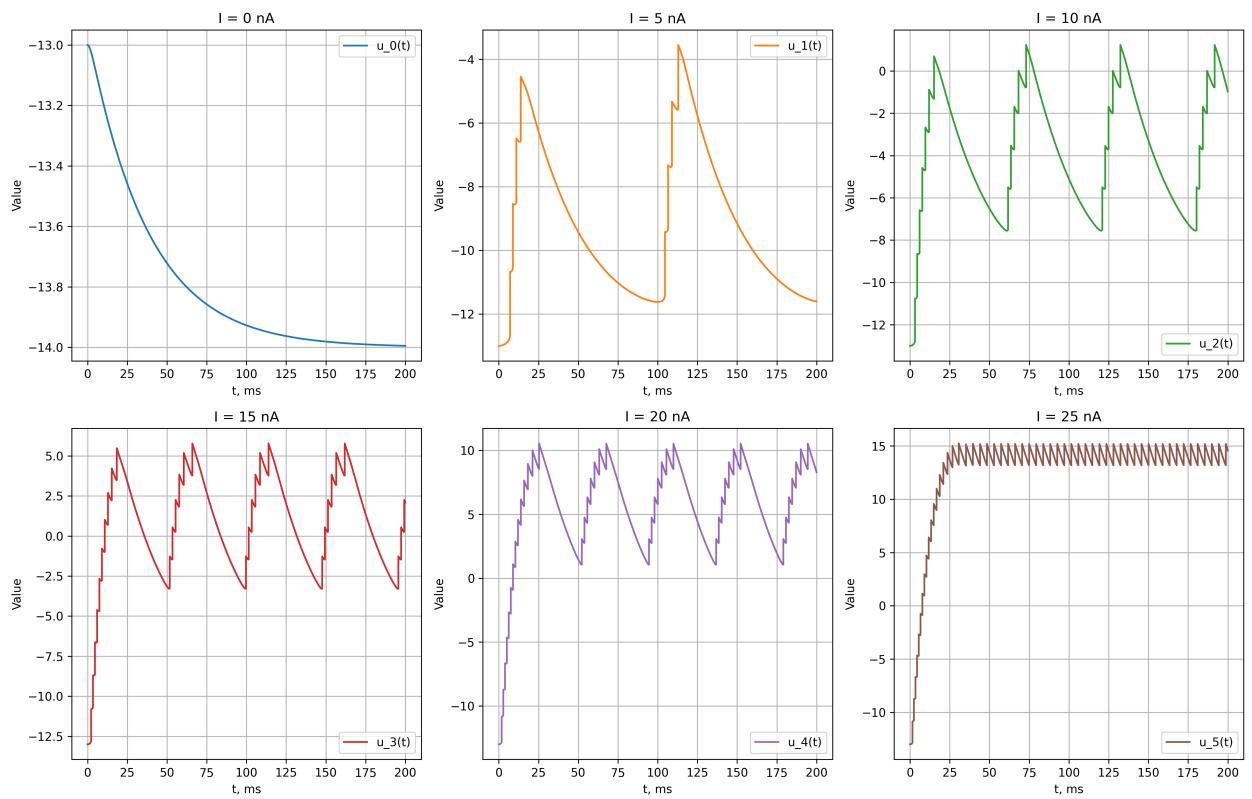


Рисунок 18 — Графики $u(t)$ чаттерного нейрона для разных значений I .

3.2 Анализ результатов

В случае регулярно-спайкового нейрона наблюдаются спайки с интервалом, который увеличивается с течением времени (рисунок 1). Для быстрого спайкового нейрона характерна постоянная высокая частота возникновения спаек (рисунок 2), особенно разница заметна при сравнении с регулярно-спайковым нейроном. В результате моделирования низкопорогового нейрона отчетливо видно постепенное снижение частоты спаек, в начале наблюдаются высокочастотные спайки (рисунок 3). Моделирование резонансного нейрона демонстрирует небольшое повышение частоты спаек в начале с последующим увеличением интервала между спайками (рисунок 4). Визуализация динамики внутренне разрывного нейрона показывает, что частота спаек в начале выше, а дальше образуется высокочастотные участки спаек с относительно большими интервалами между ними (рисунок 5). Для чаттерного нейрона характерно образование высокочастотных участков спаек с интервалом между ними (рисунок 6).

Выполненное моделирование подтверждает характеристики динамики нейрона Ижикевича с определенными наборами параметров, описанными в предыдущей главе.

Повышение значений тока I для всех рассмотренных моделей приводит к росту частоты спаек (рисунки 7-18), а в случае чаттерного нейрона еще и к стиранию интервала между высокочастотными участками спаек.

3.3 Листинг кода

```
1 import numpy as np
2 import matplotlib.pyplot as plt
3 from scipy.integrate import odeint
4 from typing import Tuple, Dict
5
6
7 regular_spiking_params = {
8     'a': 0.02,
9     'b': 0.2,
10    'c': -65,
11    'd': 8,
12    'v0': -65,
13    'I': 10,
14    'tt': 'Regular spiking'
15 }
16
17 fast_spiking_params = {
18     'a': 0.1,
19     'b': 0.2,
```

```

20      'c': -65,
21      'd': 2,
22      'v0': -70,
23      'I': 15,
24      'tt': 'Fast spiking'
25  }
26
27 low_threshold_spiking_params = {
28     'a': 0.02,
29     'b': 0.25,
30     'c': -65,
31     'd': 2,
32     'v0': -70,
33     'I': 7,
34     'tt': 'Low-threshold spiking'
35  }
36
37 resonator_params = {
38     'a': 0.1,
39     'b': 0.26,
40     'c': -65,
41     'd': 2,
42     'v0': -65,
43     'I': 10,
44     'tt': 'Resonator'
45  }
46
47 intrinsically_bursting_params = {
48     'a': 0.02,
49     'b': 0.2,
50     'c': -55,
51     'd': 4,
52     'v0': -60,
53     'I': 10,
54     'tt': 'Intrinsically bursting'
55  }
56
57 chattering_params = {
58     'a': 0.02,
59     'b': 0.2,
60     'c': -50,
61     'd': 2,
62     'v0': -65,
63     'I': 10,
64     'tt': 'Chattering'
65  }
66
67
68 def izhikevich_model(state, t, params):
69     v, u = state
70     a, b, I = params['a'], params['b'], params['I']
71
72     dvdt = 0.04 * v**2 + 5 * v + 140 - u + I
73     dudt = a * (b * v - u)
74     return [dvdt, dudt]
75
76
77 def simulate_izhikevich(params, t_max=200, dt=0.001):
78
79     a, b, c, d = params['a'], params['b'], params['c'], params['d']
80     v0, u0 = params['v0'], params['b'] * params['v0']
81     I = params['I']
82
83     t = np.arange(0, t_max, dt)
84     n_steps = len(t)
85
86     v = np.zeros(n_steps)
87     u = np.zeros(n_steps)

```

```

88     v[0], u[0] = v0, u0
89
90     for i in range(1, n_steps):
91         state = [v[i-1], u[i-1]]
92         new_state = odeint(izhikevich_model, state, [0, dt], args=(params,))[1]
93
94         v[i], u[i] = new_state
95
96         if v[i] >= 30:
97             v[i] = c
98             u[i] += d
99
100    return t, v, u
101
102
103 t, v, u = simulate_izhikevich(chattering_params)
104
105
106 DARK_RED = '#c91010'
107 DARK_GREEN = '#23a118'
108
109 plt.figure(figsize=(12, 5))
110 plt.suptitle(chattering_params['tt'], fontsize=14, y=1.02)
111
112 plt.subplot(1, 2, 1)
113 plt.plot(t, v, color=DARK_RED)
114 plt.title('Membrane Potential')
115 plt.legend('v(t)')
116 plt.xlabel('t, ms')
117 plt.ylabel('Potential, mV')
118 plt.grid(True)
119
120 plt.subplot(1, 2, 2)
121 plt.plot(t, u, color=DARK_GREEN)
122 plt.title('Recovery Variable')
123 plt.legend('u(t)')
124 plt.xlabel('t, ms')
125 plt.ylabel('Value')
126 plt.grid(True)
127
128 plt.tight_layout()
129 plt.savefig('chattering.png', dpi=300, bbox_inches='tight')
130 plt.show()
131
132 from google.colab import files
133 files.download('chattering.png')
134
135 I_values = [0, 5, 10, 15, 20, 25]
136 colors = ['#1f77b4', '#ff7f0e', '#2ca02c', '#d62728', '#9467bd', '#8c564b']
137
138 plt.figure(figsize=(15, 10))
139
140 for idx, I in enumerate(I_values):
141     params = regular_spiking_params.copy()
142     params['I'] = I
143     params['tt'] = f'I = {I}'
144
145     t, v, u = simulate_izhikevich(params)
146
147     plt.subplot(2, 3, idx+1)
148     plt.plot(t, v, color=colors[idx])
149     plt.title(f'I = {I} nA')
150     plt.xlabel('t, ms')
151     plt.ylabel('Potential, mV')
152     plt.legend([f'v_{idx}(t)'])
153     plt.grid(True)
154     plt.ylim(-90, 40)
155

```

```

156
157 plt.suptitle(f"{{regular_spiking_params['tt']}]: Membrane Potential", fontsize=14,
158     y=1.02)
158 plt.tight_layout()
159 plt.savefig('rs_different_I_potentials.png', dpi=300, bbox_inches='tight')
160
161 files.download('rs_different_I_potentials.png')
162
163
164 plt.figure(figsize=(15, 10))
165
166 for idx, I in enumerate(I_values):
167     params = regular_spiking_params.copy()
168     params['I'] = I
169     params['tt'] = f'I = {I}'
170
171 t, v, u = simulate_izhikevich(params)
172
173 plt.subplot(2, 3, idx+1)
174 plt.plot(t, u, color=colors[idx])
175 plt.title(f'I = {I} nA')
176 plt.xlabel('t, ms')
177 plt.ylabel('Value')
178 plt.legend([f'u_{idx}(t)'])
179 plt.grid(True)
180
181 plt.suptitle(f"{{regular_spiking_params['tt']}]: Recovery Variable", fontsize=14,
182     y=1.02)
182 plt.tight_layout()
183 plt.savefig('rs_different_I_recovery.png', dpi=300, bbox_inches='tight')
184 files.download('rs_different_I_recovery.png')
185
186 plt.show()

```

Листинг 1 — Исходный код программы для моделирования динамики отдельных нейронов.

4 ПОСТРОЕНИЕ СЕТИ ИЗ ДВУХ СВЯЗАННЫХ НЕЙРОНОВ

Будем рассматривать гетерогенную сеть из двух нейронов: одного регулярно-спайкового(RS) и одного быстро-спайкового(FS). Введем переменную σ , характеризующую силу связи между нейронами сети. Запишем дифференциальные уравнения для моделирования динамики связанных нейронов

$$\begin{cases} \frac{dv_1}{dt} = 0.04v_1^2 + 5v_1 + 140 - u_1 + I_1 + \sigma(v_2 - v_1), \\ \frac{du_1}{dt} = a(bv_1 - u_1), \\ \frac{dv_2}{dt} = 0.04v_2^2 + 5v_2 + 140 - u_2 + I_2 + \sigma(v_1 - v_2), \\ \frac{du_2}{dt} = a(bv_2 - u_2), \end{cases} \quad (3)$$

вспомогательный сброс

$$\text{если } v_i \geq 30 \text{ mV, то } \begin{cases} v_i \leftarrow c_i \\ u_i \leftarrow u_i + d_i, \end{cases} \quad (4)$$

Под синхронизацией в данной задачей будем понимать согласованное во времени функционирование двух объектов (нейронов сети)[2]. Для того, чтобы оценить уровень синхронизации в зависимости от заданной силы связи σ введем функцию $s(\sigma)$, значение которой вычисляется по формуле

$$s(\sigma) = \frac{1}{n} \sum_{t=t_1}^{t_n} (v_1(t) - v_2(t))^2 \quad (5)$$

5 ПРОВЕДЕНИЕ МОДЕЛИРОВАНИЯ ПРИ РАЗЛИЧНЫХ ЗНАЧЕНИЯХ СИЛЫ СВЯЗИ

5.1 Визуализация

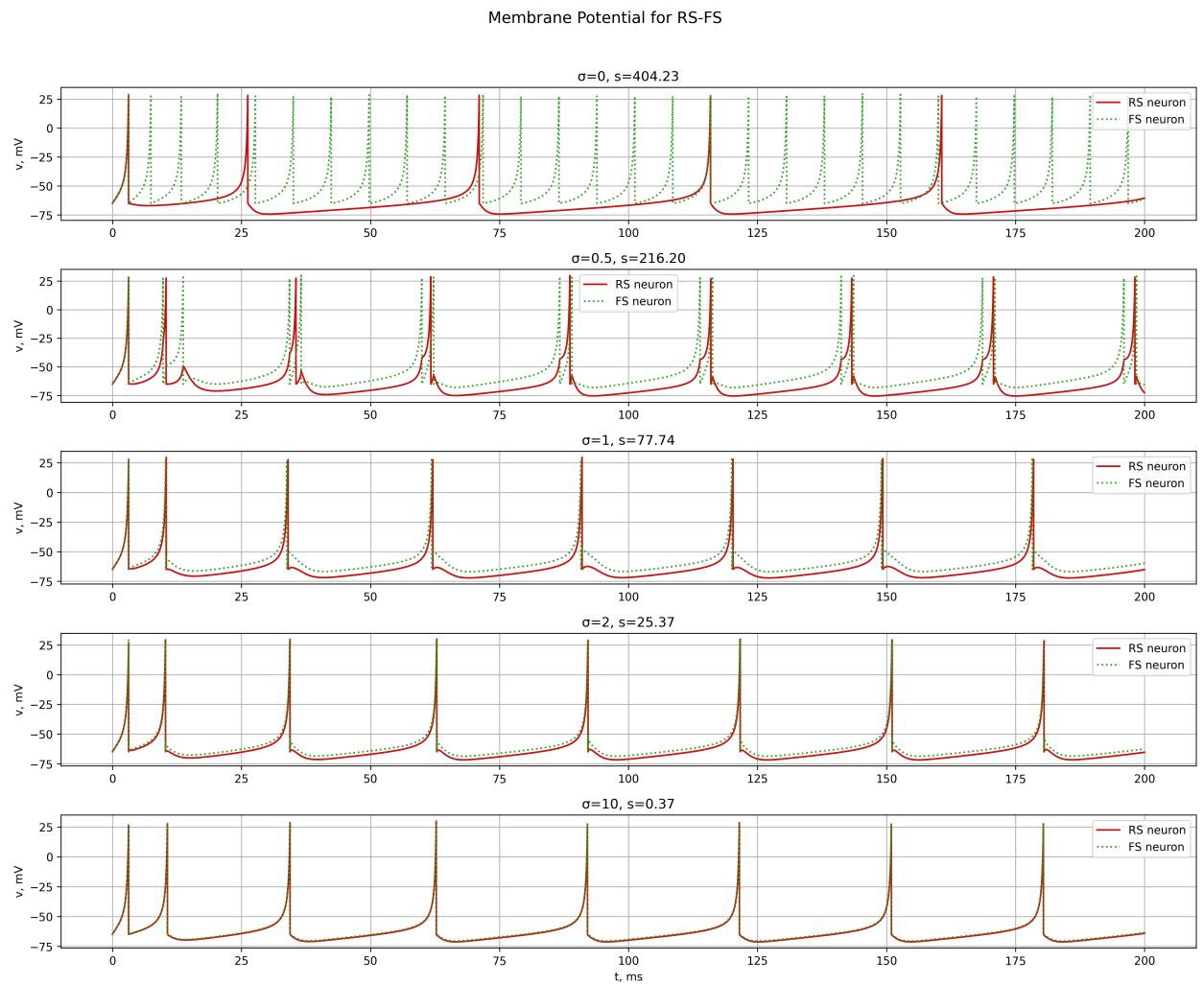


Рисунок 19 — Графики мембранныного потенциала $v_i(t)$ нейронов сети для различных значений силы связи.

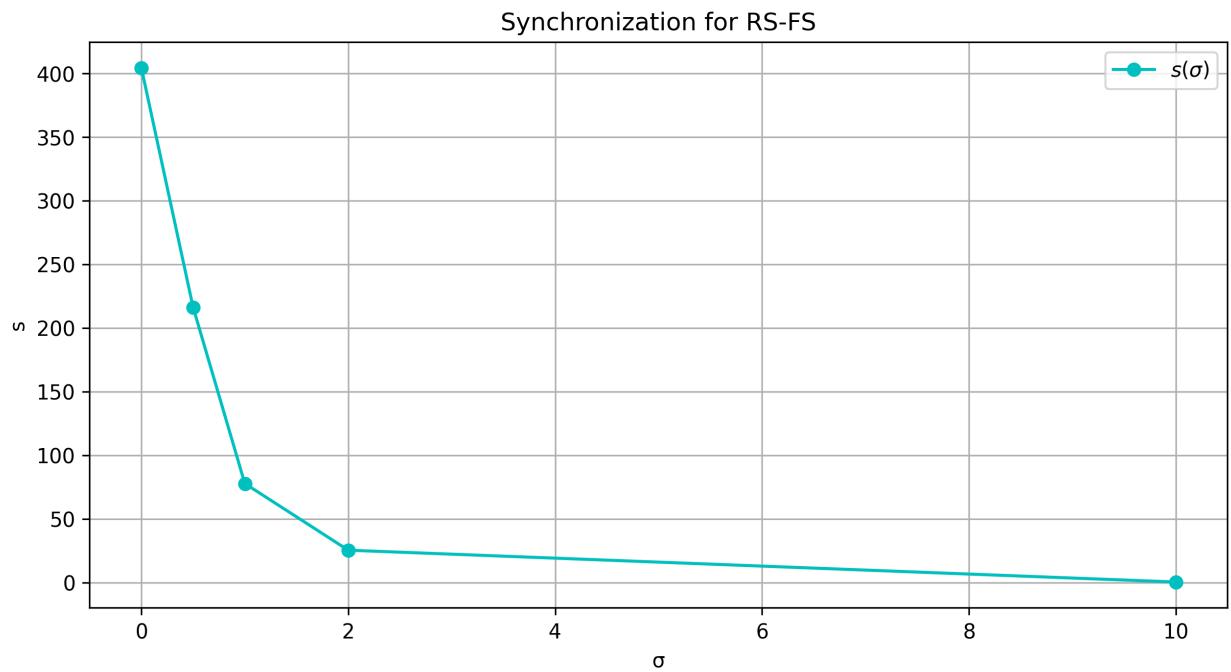


Рисунок 20 — График синхронизации $s(\sigma)$ нейронов сети для различных значений силы связи.

5.2 Анализ результатов

Заметим, что при увеличении значения силы связи возрастает и уровень синхронизации нейронов сети (рисунок 20). Кроме того, с увеличением силы связи динамика мембранных потенциалов обоих нейронов меняется, для RS-нейрона характерно повышение частоты спаек, для FS-нейрона – падение частоты (рисунок 19). В начале временного интервала заметно повышение частоты спаек с последующим увеличением временного интервала между спайками.

5.3 Листинг кода

```

1 import numpy as np
2 import matplotlib.pyplot as plt
3 from scipy.integrate import odeint
4 from google.colab import files
5
6
7 rs_params = {
8     'a': 0.02,
9     'b': 0.2,
10    'c': -65,
11    'd': 8,
12    'v0': -65,
13    'I': 10
14 }
```

```

15 fs_params = {
16     'a': 0.1,
17     'b': 0.2,
18     'c': -65,
19     'd': 2,
20     'v0': -65,
21     'I': 10
22 }
23
24
25 sigmas = [0, 0.5, 1, 2, 10]
26
27 def izhikevich_model_net(state, t, a1, b1, I1, a2, b2, I2, sigma_):
28     v1, u1, v2, u2 = state
29     dv1dt = 0.04*v1**2 + 5*v1 + 140 - u1 + I1 + sigma_*(v2 - v1)
30     du1dt = a1*(b1*v1 - u1)
31
32     dv2dt = 0.04*v2**2 + 5*v2 + 140 - u2 + I2 + sigma_*(v1 - v2)
33     du2dt = a2*(b2*v2 - u2)
34     return [dv1dt, du1dt, dv2dt, du2dt]
35
36 def simulate_izhikevich_net(sigma_, t_max=200, dt=0.01):
37     t = np.arange(0, t_max, dt)
38     state = np.array([
39         rs_params['v0'],
40         rs_params['b'] * rs_params['v0'],
41         fs_params['v0'],
42         fs_params['b'] * fs_params['v0']
43     ])
44
45     v_rs = np.zeros_like(t)
46     v_fs = np.zeros_like(t)
47
48     for i in range(len(t)):
49         new_state = odeint(
50             izhikevich_model_net,
51             state,
52             [0, dt],
53             args=(
54                 rs_params['a'], rs_params['b'], rs_params['I'],
55                 fs_params['a'], fs_params['b'], fs_params['I'],
56                 sigma_
57             )
58         )[-1]
59
60         if new_state[0] >= 30:
61             new_state[0] = rs_params['c']
62             new_state[1] += rs_params['d']
63         if new_state[2] >= 30:
64             new_state[2] = fs_params['c']
65             new_state[3] += fs_params['d']
66
67         v_rs[i], v_fs[i] = new_state[0], new_state[2]
68         state = new_state
69
70     return t, v_rs, v_fs
71
72
73
74 def sync_level(v1, v2):
75     return np.mean((v1 - v2)**2)
76
77 DARK_RED = '#c91010'
78 DARK_GREEN = '#23a118'
79
80 plt.figure(figsize=(15, 12))
81
82 for i, sigma_ in enumerate(sigmas):

```

```

83     t, v_rs, v_fs = simulate_izhikevich_net(sigma_)
84     sync = sync_level(v_rs, v_fs)
85
86     plt.subplot(len(sigmas), 1, i+1)
87     plt.plot(t, v_rs, color=DARK_RED, label='RS neuron')
88     plt.plot(t, v_fs, color=DARK_GREEN, label='FS neuron', linestyle=':')
89     plt.title(f'={sigma_}, s={sync:.2f}')
90     plt.ylabel('v, mV')
91     plt.legend()
92     plt.grid(True)
93     plt.yticks([-75, -50, -25, 0, 25])
94
95     plt.xlabel('t, ms')
96     plt.suptitle(f"Membrane Potential for RS-FS", fontsize=14, y=1.02)
97     plt.tight_layout()
98     plt.savefig('v_rsfs.png', dpi=300, bbox_inches='tight')
99     files.download('v_rsfs.png')
100    plt.show()
101
102
103    sync_values = [sync_level(*simulate_izhikevich_net(sigma_)[1:]) for sigma_ in
104                  sigmas]
105
106    plt.figure(figsize=(10, 5))
107    plt.plot(sigmas, sync_values, 'co-', label=r'$s(\sigma)$')
108    plt.legend()
109    plt.xlabel(' ')
110    plt.ylabel('s')
111    plt.title('Synchronization for RS-FS')
112    plt.grid(True)
113    plt.savefig('sync.png', dpi=300, bbox_inches='tight')
114    files.download('sync.png')
115    plt.show()

```

Листинг 2 — Исходный код программы для моделирования динамики сети нейронов.

6 ЗАКЛЮЧЕНИЕ

В ходе выполнения работы была изучена математическая модель нейрона Ижикевича. Рассмотрено влияние параметров модели на характеристики моделируемого нейрона. Выполнено численное моделирование изменения мембранныго потенциала нейрона, а также вспомогательной переменной, отвечающей за восстановление мембранныго потенциала. Теоретические данные были подтверждены проведенным моделированием, кроме того, было выяснено, что увеличение значения входного тока приводит к росту частоты спаек во всех рассмотренных наборах параметров модели.

В последних главах работы была построена гетерогенная сеть из двух связанных нейронов. Выполнено моделирование динамики нейронов сети при различных значениях силы связи, совместно с определением уровня синхронизации нейронов сети. Было выяснено, что рост силы связи приводит к увеличению степени синхронизации динамики нейронов в сети.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. *Izhikevich E.M.* Simple Model of Spiking Neurons // IEEE TRANSACTIONS ON NEURAL NETWORKS. — 2003. — Vol. 14, no. 6. — P. 1569–1572.
2. Семенов Д.М. Управление синхронизацией двух связанных неидентичных систем Хиндмарш-Роуз // Управление большими системами. — 2018. — № 75. — С. 30—49.