



Известия высших учебных заведений. Прикладная нелинейная динамика. 2024. Т. 32, № 1
Izvestiya Vysshikh Uchebnykh Zavedeniy. Applied Nonlinear Dynamics. 2024;32(1)

Научная статья
УДК 537.86

DOI: 10.18500/0869-6632-003079
EDN: VCXHMU

Моделирование динамики нейронных осцилляторов типа Ходжкина–Хаксли при помощи нейронной сети

П. В. Купцов^{1,2}✉, Н. В. Станкевич^{2,1}

¹Саратовский филиал Института радиотехники и электроники им. В. А. Котельникова РАН, Россия

²Национальный исследовательский университет «Высшая школа экономики», Нижний Новгород, Россия

E-mail: ✉kupav@mail.ru, stankevichnv@mail.ru

Поступила в редакцию 27.04.2023, принята к публикации 8.09.2023,

опубликована онлайн 10.12.2023, опубликована 31.01.2024

Аннотация. Цель настоящего исследования — представить подробное описание процедуры создания и обучения нейросетевого отображения на примере моделирования динамики нейронного осциллятора типа Ходжкина–Хаксли; показать, что нейросетевые отображения, обученные для одиночного осциллятора, можно использовать в качестве элементов связанной системы, моделирующей поведение связанных осцилляторов. **Методы.** В работе используется численный метод решения жёстких систем обыкновенных дифференциальных уравнений. Также применяется процедура обучения нейронных сетей на основе метода обратного распространения ошибки и алгоритма оптимизации Adam, который представляет собой модифицированный алгоритм градиентного спуска с автоматической подстройкой шага. **Результаты.** Показано, что построенные согласно описанной процедуре нейросетевые отображения с высокой точностью воспроизводят динамику одиночных нейронных осцилляторов. Кроме того, без дополнительного обучения эти отображения можно использовать как элементы связанной системы для моделирования динамики связанных систем нейронных осцилляторов. **Заключение.** Описанное нейросетевое отображение может рассматриваться как новая универсальная конструкция для моделирования сложной динамики. В отличие от моделей на основе разложения в ряды (степенные, тригонометрические), нейросетевое отображение не требует отбрасывания старших членов. Следовательно, оно позволяет моделировать процессы с произвольным порядком нелинейности, и по этой причине есть основания полагать, что в некоторых аспектах оно окажется более эффективным. Развитый в работе подход на основе использования нейросетевого отображения можно рассматривать в некотором смысле как альтернативу традиционным численным методам моделирования динамики. Актуальным этот подход делает бурное развитие в настоящее время технологий создания быстродействующего вычислительного оборудования, поддерживающего обучение и работу нейронных сетей.

Ключевые слова: нейросетевое отображение, нейронная сеть, набор данных, обучение нейронной сети, нейроморфная динамика, численное моделирование.

Благодарности. Исследование математических моделей было проведено в рамках проекта «Зеркальные лаборатории» НИУ ВШЭ (разделы 1, 2). Разработка и исследование нейросетевого отображения (разделы 3–5) выполнены при поддержке Российского научного фонда, проект 20-71-10048.

Для цитирования: Купцов П. В., Станкевич Н. В. Моделирование динамики нейронных осцилляторов типа Ходжкина–Хаксли при помощи нейронной сети // Известия вузов. ПНД. 2024. Т. 32, № 1. С. 72–95. DOI: 10.18500/0869-6632-003079. EDN: VCXHMU

Статья опубликована на условиях Creative Commons Attribution License (CC-BY 4.0).

Modeling of the Hodgkin–Huxley neural oscillators dynamics using an artificial neural network

P. V. Kuptsov^{1,2}, *N. V. Stankevich*^{2,1}

¹Saratov Branch of Kotelnikov Institute of Radioengineering and Electronics of the RAS, Russia

²HSE University, Nizhny Novgorod, Russia

E-mail: ✉kupav@mail.ru, stankevichnv@mail.ru

Received 27.04.2023, accepted 8.09.2023, available online 10.12.2023, published 31.01.2024

Abstract. The purpose of this study — to represent a detailed description of the procedure for creating and training a neural network mapping on the example of the dynamics modeling of a neural oscillator of the Hodgkin–Huxley type; to show that the neural network mappings trained for single oscillators can be used as elements of a coupled system that simulate the behavior of coupled oscillators. **Methods.** Numerical method is used for solving stiff systems of ordinary differential equations. Also a procedure for training neural networks based on the method of back propagation of error is employed together with the Adam optimization algorithm, that is a modified version of the gradient descent supplied with an automatic step adjustment. **Results.** It is shown that the neural network mappings built according to the described procedure are able to reproduce the dynamics of single neural oscillators. Moreover, without additional training, these mappings can be used as elements of a coupled system for the dynamics modeling of coupled neural oscillator systems. **Conclusion.** The described neural network mapping can be considered as a new universal framework for complex dynamics modeling. In contrast to models based on series expansion (power, trigonometric), neural network mapping does not require truncating of the series. Consequently, it allows modeling processes with arbitrary order of nonlinearity, hence there are reasons to believe that in some aspects it will be more effective. The approach developed in this paper based on the neural network mapping can be considered as a sort of an alternative to the traditional numerical methods of modeling of dynamics. What makes this approach topical is the current rapid development of technologies for creating fast computing equipment that supports neural network training and operation.

Keywords: neural network mapping, neural network, dataset, neural network learning, neuromorphic dynamics, numerical simulation.

Acknowledgements. The study of mathematical models was carried out within the framework of the Mirror Laboratories Project of HSE University (Sections 1, 2). Development and research of the neural network mapping (Sections 3–5) was supported by the Russian Science Foundation, 20-71-10048.

For citation: Kuptsov PV, Stankevich NV. Modeling of the Hodgkin–Huxley neural oscillators dynamics using an artificial neural network. *Izvestiya VUZ. Applied Nonlinear Dynamics*. 2024;32(1):72–95. DOI: 10.18500/0869-6632-003079

This is an open access article distributed under the terms of Creative Commons Attribution License (CC-BY 4.0).

Введение

Исследование динамики нелинейных систем в значительной степени сводится к обработке и анализу данных, генерируемых в ходе эволюции этих систем. По этой причине для обогащения научного инструментария нелинейной динамики и для расширения круга решаемых задач представляется естественным обращаться к опыту бурно развивающихся в настоящее время дисциплин из области машинного обучения и науки о данных.

Не претендуя на полноту, в качестве примера такого междисциплинарного обогащения можно указать несколько работ: применение нейронных сетей для адаптивного моделирования и управления системами [1], применение нейронных сетей для реконструкции аттрактора Эль-Ниньо [2], использование нейронных сетей для предсказания динамики на основе выполненной реконструкции пространства состояний [3], использование методов машинного обучения для построения моделей и анализа динамики биологических систем [4].

Один из важнейших элементов фундамента продуктивного взаимодействия нелинейной динамики с науками о данных и машинном обучении — серия теорем, с математической строгостью

обосновывающих возможность аппроксимации произвольных функций многих переменных при помощи нейронных сетей [5–10]. Из этих работ следует, что уже двухслойной полносвязной сети с сигмоидальной функцией активации достаточно, чтобы с заданной точностью аппроксимировать любую функцию многих переменных. Точность аппроксимации определяется размером слоёв сети, а также выбором эффективного алгоритма обучения. В самом общем виде это означает, что для решения задач, связанных с поиском или анализом функциональных зависимостей, можно привлекать нейронные сети. Ещё один элемент, уже не строго математический, а, скорее, эвристический — способность нейронных сетей к обобщению. При правильной организации процесса обучения сети, она не просто запоминает предъявляемые ей данные, а вычленяет из них наиболее существенные, наиболее общие их признаки. Впоследствии это позволяет уже обученной сети эффективно обрабатывать новые образцы данных, не предъявлявшиеся ей в процессе обучения [11].

В настоящей работе исследуется применимость нейронных сетей для моделирования нелинейной динамики: представляется интересным найти универсальную архитектуру, которая могла бы после соответствующего обучения с заданной точностью воспроизводить любое или почти любое поведение нелинейных систем. В пользу возможности построения такой архитектуры говорят упомянутые выше теоремы об аппроксимации и способность нейронных сетей к обобщению. Теоретическая мотивация такого исследования — получить ещё одну универсальную модель динамики, наряду с широко применяемыми сегодня моделями, например, на основе степенных или тригонометрических рядов. Можно ожидать, что применение нейросетевой модели будет в каких-то аспектах более эффективным по сравнению с моделями на основе разложений в ряды. Последние всегда учитывают ограниченное, часто небольшое число членов ряда и следовательно порядок моделируемой нелинейности также ограничен. В случае же нейронной сети это ограничение снимается. Кроме того, интерес вызывает способность нейронных сетей к обобщению — в идеале даже при наличии ограниченного количества данных можно надеяться, что сеть будет в состоянии извлечь из них информацию о динамических феноменах, недоступную для других методов моделирования. С практической точки зрения моделирование динамики при помощи нейронных сетей можно рассматривать как своего рода альтернативу численным методам решения модельных уравнений. Традиционные методы численного моделирования наилучшим образом реализуются на вычислительных машинах с классической архитектурой и зачастую даже их распараллеливание на несколько вычислительных ядер вызывает затруднения. При этом современная вычислительная техника развивается в сторону адаптации к поддержке работы именно нейронных сетей. В качестве примера можно привести игровые видеокарты и так называемые AI-ускорители [12–15]. Поэтому можно ожидать, что развитие методов моделирования динамики при помощи нейронных сетей будет иметь большое практическое значение.

Ранее в работе [16] было показано, что даже простейшую двухслойную сеть можно обучить воспроизводить достаточно разные виды динамики — системы Лоренца и Рёсслера, а также модель нейрона Хиндмарша–Роуза. Продемонстрировано хорошее соответствие бифуркационных картин, спектров Фурье и показателей Ляпунова. В работе [17] предложена более сложная структура сети, когда каждую из переменных моделирует отдельная подсеть. Такая сеть справляется с моделированием жёсткой динамики, когда переменные системы имеют разные масштабы времени. Было показано что сеть воспроизводит поведение модельного физиологического нейрона, заданного системой уравнений, сформулированных на основе формализма Ходжкина–Хаксли [18]. Продемонстрировано, что благодаря своей способности к обобщению, нейросетевая модель может успешно воспроизводить режим бистабильности даже когда в процессе обучения ей предъявлялась только одна из ветвей решения. Нейросетевая модель, обученная только на колебательном решении, находит также и сосуществующую с ним устойчивую неподвижную точку, и кроме того правильно воспроизводит зависимость её собственных чисел от параметров. В настоящей работе мы рассмотрим нейросетевую модель, предложенную в статье [17].

Цель исследования — продемонстрировать, что такие модели, обученные для одиночных систем, можно без дополнительного обучения использовать для воспроизведения динамики связанных систем. Будет показано, что наблюдается хорошее качественное и количественное соответствие различных динамических режимов связанных систем и их бифуркационных трансформаций.

1. Модельный нейрон на основе формализма Ходжкина–Хаксли

Мы рассматриваем модельный нейрон, уравнения которого получены на основе формализма Ходжкина–Хаксли [18]. Кроме оригинальной системы мы также рассмотрим её модифицированный вариант, предложенный в работе [19]. Исходная система в режиме колебаний имеет неустойчивую неподвижную точку. Модификация приводит к тому, что в пространстве параметров появляется область, в которой неподвижная точка устойчива, то есть в системе возникает бистабильность, характерная для нейронных моделей [20–22]. Так как отвечающая за это модификация активируется только в небольшой области вблизи неподвижной точки, это практически не оказывает видимого влияния на качественный характер колебательного решения.

$$\begin{aligned}\tau\dot{V} &= -I_{Ca}(V) - I_K(V, n) - I_{K2}(V) - I_S(V, S), \\ \tau\dot{n} &= \sigma[n_\infty(V) - n], \\ \tau_S\dot{S} &= S_\infty(V) - S.\end{aligned}\tag{1}$$

Здесь V , n и S — динамические переменные. Входящие в уравнения функции заданы формулами (2), а используемые числовые значения параметров сведены в табл. 1.

$$I_{Ca}(V) = g_{Ca} m_\infty(V) (V - V_{Ca}),\tag{2a}$$

$$I_K(V, n) = g_K n (V - V_K),\tag{2b}$$

$$I_S(V, S) = g_S S (V - V_K),\tag{2c}$$

$$I_{K2}(V) = g_{K2} p_\infty(V) (V - V_K),\tag{2d}$$

$$\omega_\infty(V) = \left(1 + \exp \frac{V_\omega - V}{\theta_\omega}\right)^{-1}, \quad \omega = m, n, S,\tag{2e}$$

$$p_\infty(V) = \left(\exp \frac{V - V_p}{\theta_p} + \exp \frac{V_p - V}{\theta_p}\right)^{-1}.\tag{2f}$$

Уравнения соответствуют оригинальной системе при $g_{K2} = 0$. Модифицированная система рассматривается при $g_{K2} = 0.12$. Переменные системы, а также входящие в уравнения

Таблица 1. Параметры системы (1)

Table 1. Parameters of the system (1)

$\tau = 0.02 \text{ s}$	$\tau_S = 35 \text{ s}$	$\sigma = 0.93$	
$g_{Ca} = 3.6$	$g_K = 10$	$g_S = 4$	$g_{K2} = 0.12$
$V_{Ca} = 25 \text{ mV}$	$V_K = -75 \text{ mV}$		
$\theta_m = 12 \text{ mV}$	$\theta_n = 5.6 \text{ mV}$	$\theta_S = 10 \text{ mV}$	$\theta_p = 1 \text{ mV}$
$V_m = -20 \text{ mV}$	$V_n = -16 \text{ mV}$	$V_S = -36 \text{ mV}$	$V_p = -49.5 \text{ mV}$

функции и параметры имеют биологическую интерпретацию, обсуждение которой можно найти в работах [18, 19].

На рис. 1, *a*, *b* и *c* показаны различные режимы динамики модифицированного варианта системы (1). Рис. 1, *a* и *b* иллюстрируют бистабильность: в зависимости от начальной точки, траектория выходит либо на пачечный аттрактор (см. рис. 1, *a*), либо на неподвижную точку (см. рис. 1, *b*). Видно, что в режиме пачечного аттрактора переменные V и n меняются значительно быстрее чем переменная S . Системы с таким видом поведения называют жёсткими [23]. Рис. 1, *c* демонстрирует другой режим системы, спайковый. В этом режиме все три переменные меняются с одинаковыми временными масштабами. Переход от пачечного аттрактора к спайковому

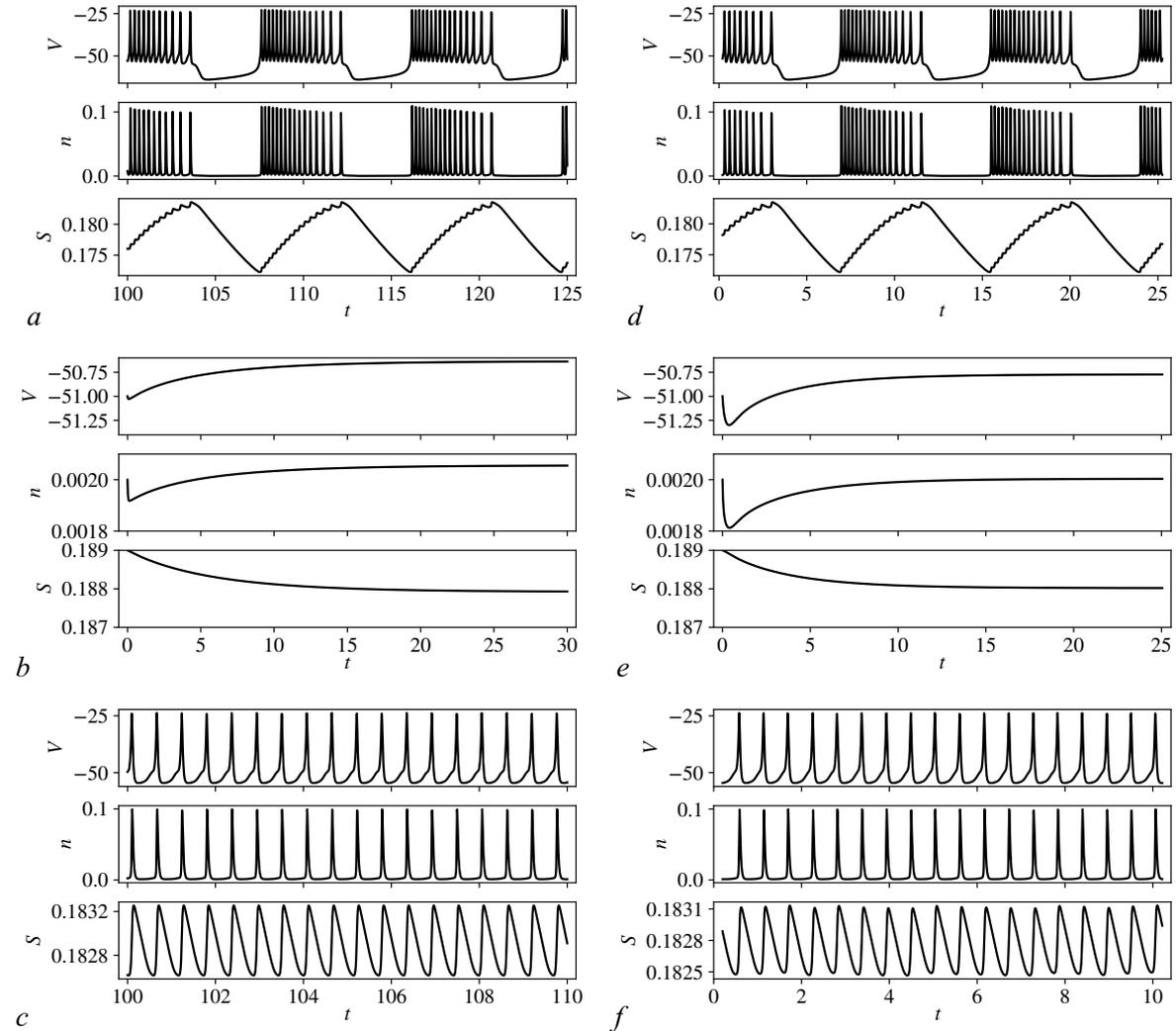


Рис. 1. Решения модифицированной системы (1), то есть при $g_{K2} = 0.12$ (*a*, *b*, *c*), и соответствующего нейросетевого отображения (5) (*d*, *e*, *f*). Диаграммы *a*, *d*, *b* и *e* иллюстрируют бистабильность при $V_S = -36$: *a*, *d* – пачечный (бёрстовый) режим при старте из точки $V_0 = -51$, $n_0 = 2 \cdot 10^{-3}$, $S_0 = 0.185$; *b*, *e* – выход на неподвижную точку при старте из точки $V_0 = -51$, $n_0 = 2 \cdot 10^{-3}$, $S_0 = 0.189$. Диаграммы *c*, *f* демонстрируют спайки при $V_S = -34$

Fig. 1. Solutions for modified system (1), i.e. e. at $g_{K2} = 0.12$ (*a*, *b*, *c*), and for the corresponding neural network mapping (5) (*d*, *e*, *f*). Diagrams *a*, *d*, *b* and *e* illustrate bistability at $V_S = -36$: *a*, *d* – bursts when starting from the point $V_0 = -51$, $n_0 = 2 \cdot 10^{-3}$, $S_0 = 0.185$; *b*, *e* – approach a fixed point when starting from the point $V_0 = -51$, $n_0 = 2 \cdot 10^{-3}$, $S_0 = 0.189$. Diagrams *c*, *f* demonstrate spikes at $V_S = -34$

происходит при увеличении параметра V_S через бифуркацию, называемую катастрофой голубого неба [19, 24]. Отметим, что пачечная или спайковая активность системы, переключение между которыми происходит при изменении параметров, называется нейроморфной динамикой [25].

На рис. 2 показаны фазовые портреты модифицированной системы в трёх измерениях. Рис. 2, *a* демонстрирует пачечный аттрактор. Хорошо видно, что в этом режиме аттрактор имеет характерную структуру из нескольких витков с замыкающей их петлёй. Спайковый аттрактор представлен на рис. 2, *b*. Так как все переменные в этом режиме колеблются с одинаковыми временными масштабами, аттрактор имеет вид предельного цикла.

Исходная версия системы также демонстрирует пачечный и спайковый режимы. Визуально они неотличимы от показанных на рис. 1 и 2 и поэтому не показаны на отдельном рисунке.

Чтобы наглядно демонстрировать изменения характера поведения системы в зависимости от параметров и от выбора начальных условий, будем вычислять скалярную характеристическую величину

$$Q = \sqrt{\frac{1}{T} \int_{t_0}^{t_0+T} S^2(t) dt}. \quad (3)$$

Здесь $S(t)$ — динамическая переменная системы (1), t_0 — некоторое, достаточно большое, время выхода системы на режим, T — время наблюдения. Мы используем $t_0 = 100$ и $T = 100$. Ниже, рассматривая систему связанных модельных нейронов (11), вычисляем величину $Q = (Q_1 + Q_2)/2$, где Q_1 и Q_2 получены по формуле (3) с заменой S на соответствующую величину S_1 или S_2 .

Смысл формулы (3) — сопоставить скаляры инвариантным множествам в фазовом пространстве системы, чтобы иметь возможность компактно и наглядно визуализировать его структуру. Не имеет значения, как конкретно будет выглядеть зависимость Q от параметров. Нас интересуют перестройки режимов, поэтому важно, чтобы при изменении характера поведения системы значение Q менялось бы достаточно значительно. Теоретически разные множества могут описываться одним и тем же Q и быть неразличимы по этому признаку. Однако вероятность этого достаточно мала. Для ещё большего снижения этой вероятности $S(t)$ возводится в квадрат, чтобы в гипотетическом случае возникновения отрицательных значений на Q влияли бы только их абсолютные величины. Не имеет смысла конструировать более сложные формулы для Q , так как представление многомерного множества при помощи скалярной величины всегда приводит к потере той или иной информации и, следовательно, не может считаться исчерпывающим. Поэтому, используя Q ,

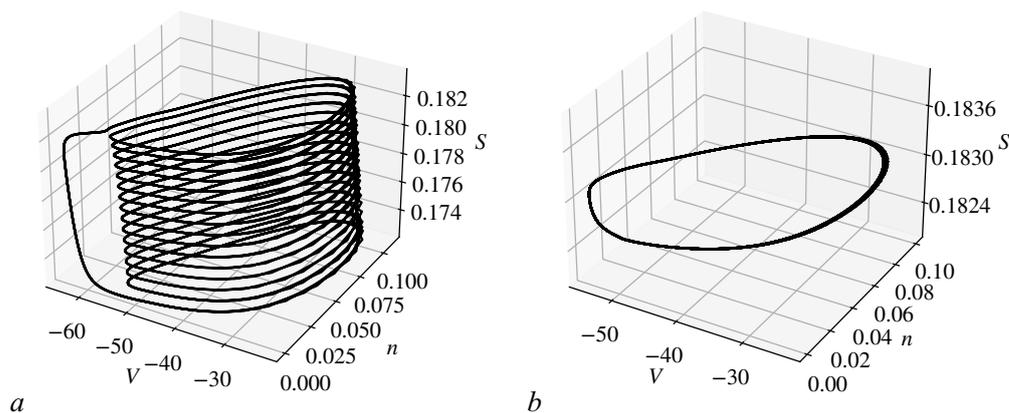


Рис. 2. Трёхмерные фазовые портреты модифицированной системы (1): *a* — пачечный режим при $V_S = -36$, *b* — режим спайков при $V_S = -34$

Fig. 2. Three-dimensional phase portraits of the modified system (1): *a* — bursts at $V_S = -36$, *b* — spikes at $V_S = -34$

следует соотносить его значения с информацией о характере динамики системы, полученной из других соображений. Применительно к изучаемой системе нам заранее известно, что в ней могут наблюдаться пачечный и спайковый режимы, и кроме этого может быть устойчивой неподвижная точка. Поэтому, вычисляя Q в зависимости от параметров, мы ожидаем, что будут встречаться три типичных значения этой величины.

На рис. 3, *a* и 3, *b* для исходной и модифицированной систем показано как меняется в зависимости от V_S распределение Q , получаемое при большом числе запусков системы со случайными начальными значениями V , n и S . Ансамбль начальных значений выбирался из области определения системы (см. ниже табл. 3 и раздел 4) следующим образом. Интервал значений параметра V_S разбивался на 300 равноотстоящих друг от друга точек, и в каждой генерировались по 3000 начальных значений V , n и S из указанных в табл. 3 диапазонов. Градации серого отображают в логарифмическом масштабе частоты появления решений, характеризуемых соответствующими значениями Q . Видно, что эти значения хорошо сгруппированы и образуют систему линий, по виду которых можно судить о происходящих в системах бифуркационных перестройках.

На рис. 3, *a* и 3, *b* линия в левой части представляет пачечный аттрактор. Вид соответствующего решения показан на рис. 1, *a* и 2, *a*. При движении в сторону увеличения V_S вблизи значения $V_S \approx -34$ у оригинальной системы (см. рис. 3, *a*) и при $V_S \approx -35$ у модифицированной (см. рис. 3, *b*) величина Q меняется скачком. Это соответствует переходу от пачечного аттрактора к спайковому (см. рис. 1, *c* и 2, *b*). На рис. 3, *b*, который построен для модифицированной системы, имеется ещё одна линия значений Q в диапазоне между $V_S \approx -37$ и $V_S \approx -35$. Эта линия соответствует неподвижной точке, которая устойчива в этом диапазоне значений параметра. Линия, представляющая неподвижную точку, заметно светлее линии для колебательного решения, существующего при тех же значениях V_S . Это говорит о том, что устойчивая неподвижная точка имеет небольшой по сравнению с колебательным решением размер бассейна притяжения, и поэтому вероятность попадания в него выбранных случайным образом начальных условий достаточно мала. Более подробно это обсуждается в работе [19].

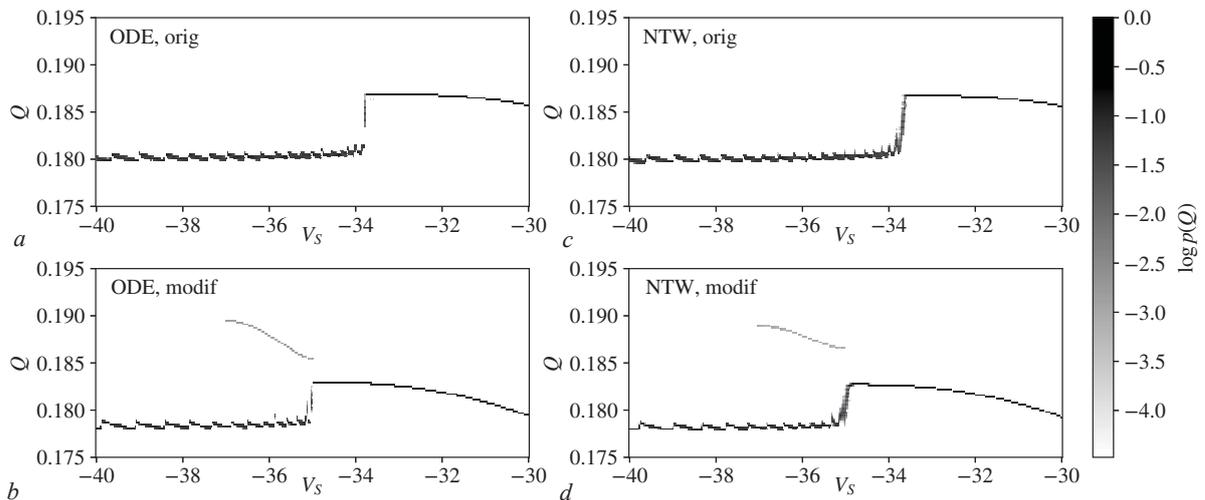


Рис. 3. Распределения Q , см. (3), в зависимости от V_S : *a* и *b* — исходная и модифицированная системы (1), соответственно; *c* и *d* — нейросетевые отображения (5), обученные для этих систем

Fig. 3. Distributions of Q , see Eq. (3), vs. V_S : *a* and *b* — original and modified systems (1), respectively; *c* and *d* — the neural network mapping (5) trained for these systems

На рис. 4, *a* показано, как изменяются одномерные сечения бассейнов притяжения различных инвариантных множеств в фазовом пространстве модифицированной системы в зависимости от V_S . Для построения этого рисунка сначала вычисляется неподвижная точка V_f, n_f, S_f системы при $V_S = -36$. Начальные значения переменных n и S всегда задаются равными $n_0 = n_f$ и $S_0 = S_f$, а начальное значение V_0 варьируется вблизи от V_f и эти значения откладываются на рисунке по вертикали. Параметр V_S варьируется около точки -36 , и эти значения откладываются по горизонтали. Для каждой пары V_S и V_0 вычисляется Q и её значения отображаются на рисунке при помощи градаций серого.

На рис. 4, *a* выделяются три области. Тёмный прямоугольник в центре — область, из которой траектории выходят на устойчивую неподвижную точку. Светло-серая область в левой части — это множество стартовых точек, из которых система выходит на пачечный аттрактор. Эти две области сосуществуют при одних и тех же значениях V_S , что соответствует режиму бистабильности в системе. Тёмно-серая область справа показывает точки, из которых система выходит на спайковый аттрактор.

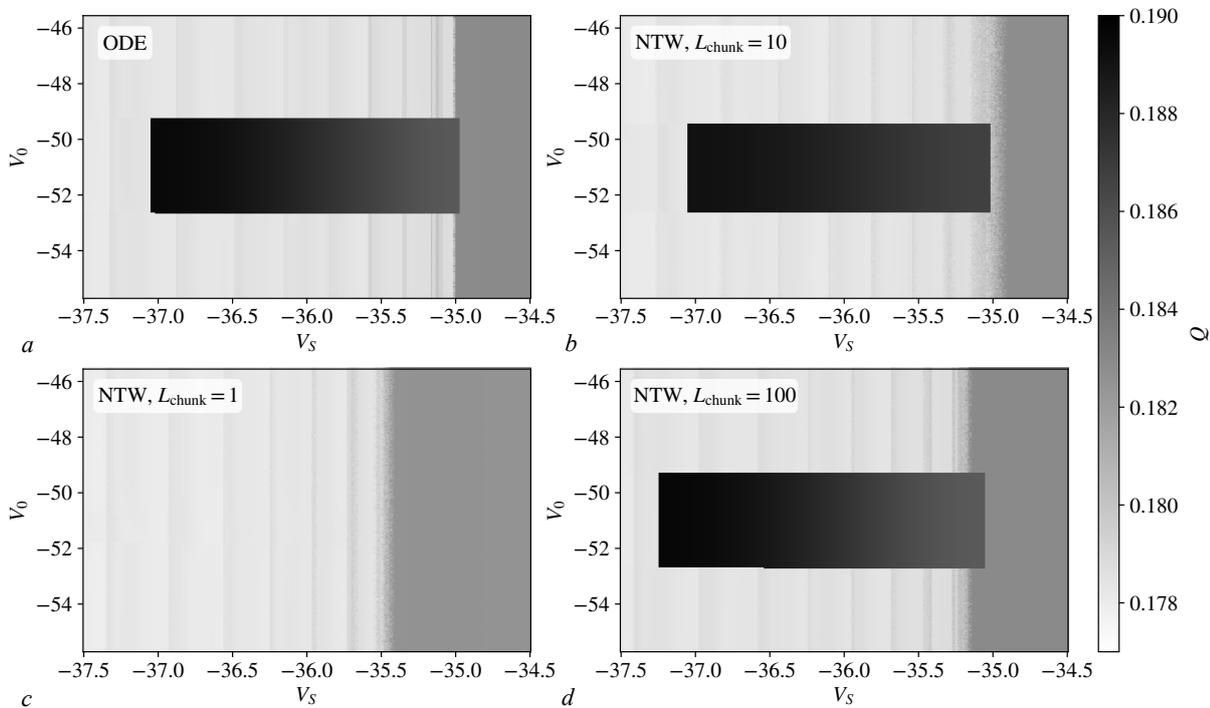


Рис. 4. Одномерные сечения бассейнов притяжения различных режимов динамики: *a* — модифицированная система (1); *b*, *c* и *d* — нейросетевое отображение (5), обученное на наборах данных, построенных из отрезков траекторий длины $L_{\text{chunk}} = 10$, $L_{\text{chunk}} = 1$ и $L_{\text{chunk}} = 100$, соответственно. Диаграммы построены в окрестности точки $V_f = -50.6357$, $n_f = 2.05598 \times 10^{-3}$, $S_f = 0.187922$, которая представляет собой устойчивую неподвижную точку (1) при $V_S = -36$. Начальное значение V_0 , отложенное по вертикали, варьируется в пределах $V_f \pm 0.1V_f$, стартовые значения переменных n_0 и S_0 берутся равными n_f и S_f , соответственно. Для каждой траектории вычисляется Q , значения которой представляются оттенками серого

Fig. 4. One-dimensional sections of attraction basins of different dynamic regimes: *a* — the modified system (1); *b*, *c* and *d* — the neural network mapping (5) trained on datasets, consisting of trajectory segments of lengths $L_{\text{chunk}} = 10$, $L_{\text{chunk}} = 1$ and $L_{\text{chunk}} = 100$, respectively. Diagrams are constructed in the vicinity of the point $V_f = -50.6357$, $n_f = 2.05598 \times 10^{-3}$, $S_f = 0.187922$ that is a stable fixed point of (1) at $V_S = -36$. The initial value of V_0 , vertical axis, is varied within the range $V_f \pm 0.1V_f$, and starting values for variables n_0 and S_0 are taken equal to n_f and S_f , respectively. For each trajectory Q is calculated and its values are represented via gray scale

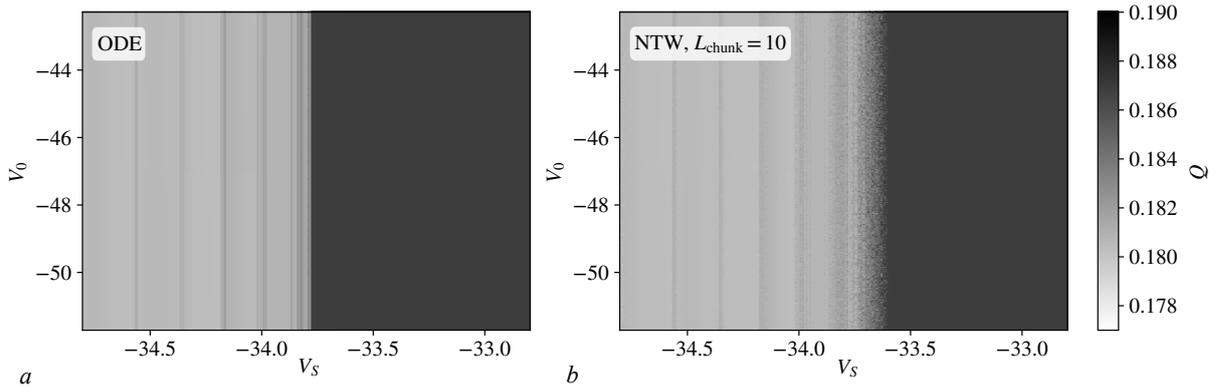


Рис. 5. Одномерные сечения бассейнов притяжения: a – оригинальная система (1); b – нейросетевое отображение, обученное на наборе данных с $L_{\text{chunk}} = 10$. Диаграммы построены в окрестности точки $V_f = -46.9978$, $n_f = 3.92943 \times 10^{-3}$, $S_f = 0.210855$ – неустойчивая неподвижная точка оригинальной системы при $V_S = -33.8$

Fig. 5. One-dimensional sections of basins of attraction: a – original system (1); b – the neural network mapping trained on a dataset with $L_{\text{chunk}} = 10$. The diagrams are built in the vicinity of the point $V_f = -46.9978$, $n_f = 3.92943 \times 10^{-3}$, $S_f = 0.210855$. This is an unstable fixed point of the original system at $V_S = -33.8$

Аналогичное построение для исходной системы представлено на рис. 5, a . Здесь имеются только две области: более светлая в левой части отвечает выходу на пачечный аттрактор, а более тёмная справа – на спайковый.

2. Связанные системы

Рассмотрим теперь две системы вида (1) и введём между ними связь, как это обсуждается в работе [26]. Уравнение для подсистемы 1 имеет вид:

$$\begin{aligned} \tau \dot{V}_1 &= -I_{Ca}(V_1) - I_K(V_1, n_1) - I_{K2}(V_1) - I_S(V_1, S_1) + g_{c,V}(V_1 - V_2), \\ \tau \dot{n}_1 &= \sigma [n_\infty(V_1) - n_1], \\ \tau_S \dot{S}_1 &= S_{\infty,1}(V_1) - S_1. \end{aligned} \quad (4)$$

Уравнения для второй системы получаются заменой индексов $1 \leftrightarrow 2$. Параметр $g_{c,V}$ отвечает за силу связи. Заметим, что в уравнениях (4) функция $S_{\infty,1}$ снабжена индексом «1». Это сделано потому, что в неё входит параметр V_S , см. (2e). Мы будем рассматривать подсистемы 1 и 2 с одинаковыми параметрами за исключением V_S , который будет задаваться разным для каждой из подсистем как $V_{S,1}$ и $V_{S,2}$, соответственно.

Представление об общей картине поведения связанных систем дают рис. 6, $a-d$, рис. 7, $a-c$ и рис. 8, $a-c$. На рис. 6, аналогично тому, как это сделано на рис. 3, полутонами показаны распределения характеристической величины $Q = (Q_1 + Q_2)/2$, где Q_1 и Q_2 вычислены по формуле (3) для подсистемы 1 и 2, соответственно. По горизонтали отложены значения параметра $V_{S,1}$, а $V_{S,2}$ равен $V_{S,1} + 0.1$. Режимы поведения двух связанных оригинальных систем в значительной мере повторяют режимы одиночной системы (сравни рис. 6, a и 3, a). В левой части диаграммы, то есть при меньших значениях $V_{S,1}$ и $V_{S,2}$ имеет место пачечный режим. Это иллюстрирует рис. 7, a . Справа, при больших $V_{S,1}$ и $V_{S,2}$ возникает спайковый режим (рис. 7, c). Отличие от одиночной системы проявляется в том, как происходит переход от одного режима к другому. Вместо строго определённой точки бифуркации, у связанных систем возникает переходная область, в которой наблюдаются пачечные колебания с нерегулярной длиной. По мере увеличения параметра средняя

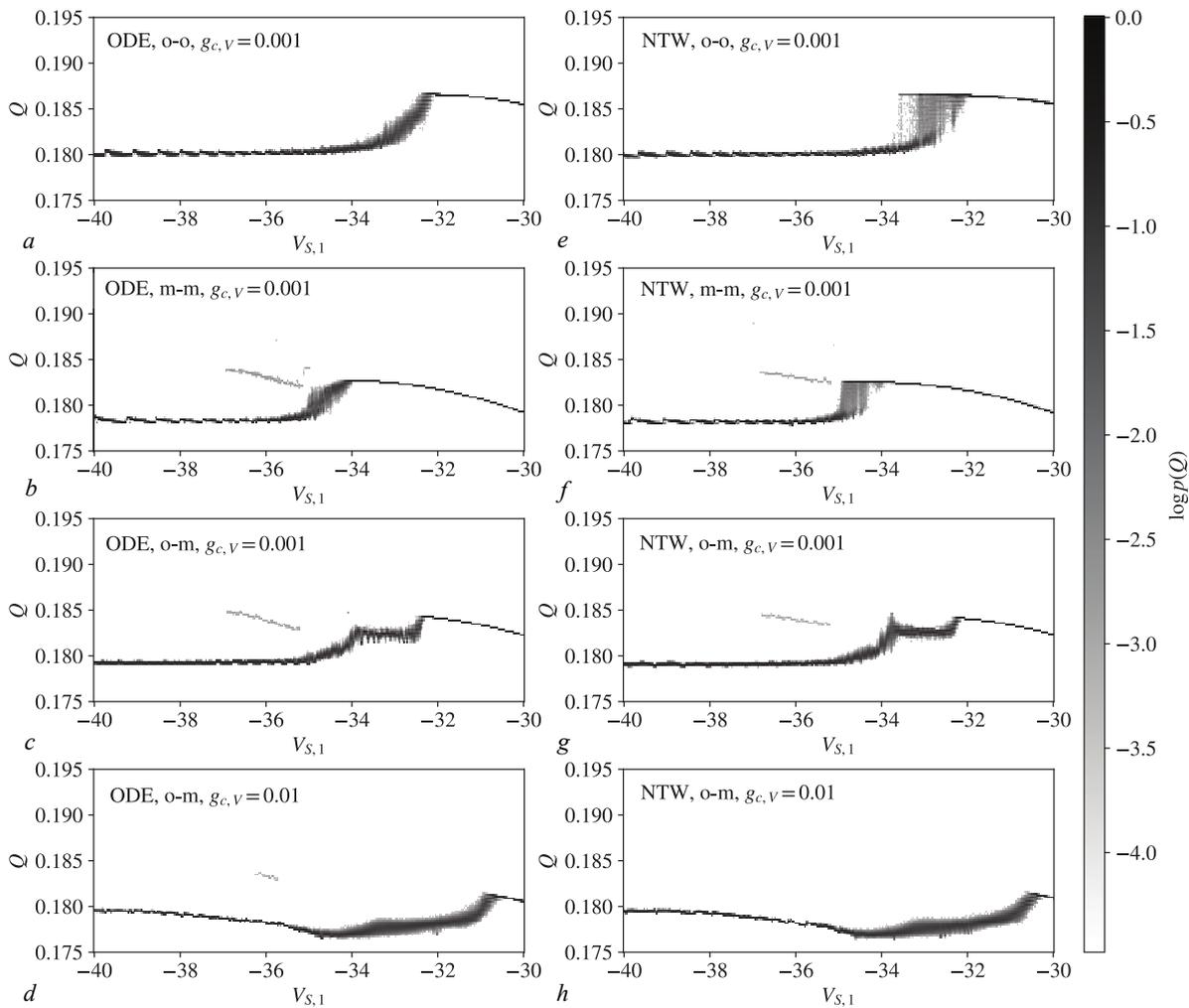


Рис. 6. Распределения $Q = (Q_1 + Q_2)/2$ в зависимости от $V_{S,1}$ при $V_{S,2} = V_{S,1} + 0.1$: $a-d$ – система уравнений (4); $e-h$ – нейросетевое отображение (11). Представлены следующие комбинации подсистем: a и e – две оригинальные системы ($g_{K2} = 0$); b и f – две модифицированные системы ($g_{K2} = 0.12$); c, d, g и h – оригинальная и модифицированная системы. Значения параметра связи: $a-c$ и $e-g$ – $g_{c,v} = 0.001$; d и h – $g_{c,v} = 0.01$

Fig. 6. Distributions of $Q = (Q_1 + Q_2)/2$ vs. $V_{S,1}$ and $V_{S,2} = V_{S,1} + 0.1$: $a-d$ – Eqs. (4); $e-h$ – the neural network mapping (11). The following combinations of subsystems are represented: a and e – two original systems ($g_{K2} = 0$); b and f – two modified systems ($g_{K2} = 0.12$); c, d, g and h – original and modified systems. Coupling parameter values: $a-c$ and $e-g$ – $g_{c,v} = 0.001$; d and h – $g_{c,v} = 0.01$

длительность пачечных всплесков растёт, пока не превращается в спайковый режим. Переходное поведение иллюстрирует рис. 7, b .

Для пары модифицированных систем сценарий перестроек режимов при движении по параметрам $V_{S,1}$ и $V_{S,2}$ в целом такой же – от пачек к спайкам через протяжённую переходную область. Как и у одиночной модифицированной системы, у пары связанных таких систем имеется область бистабильности, в которой с колебательным решением сосуществует устойчивая неподвижная точка (сравни рис. 3, b и 6, b).

Динамика пары оригинальная–модифицированная системы характеризуется уширением области перехода от пачек к спайкам (см. рис. 6, c), а также к появлению более сложных переходных режимов. В качестве иллюстрации на рис. 8, a показаны колебания, когда возникают пачки

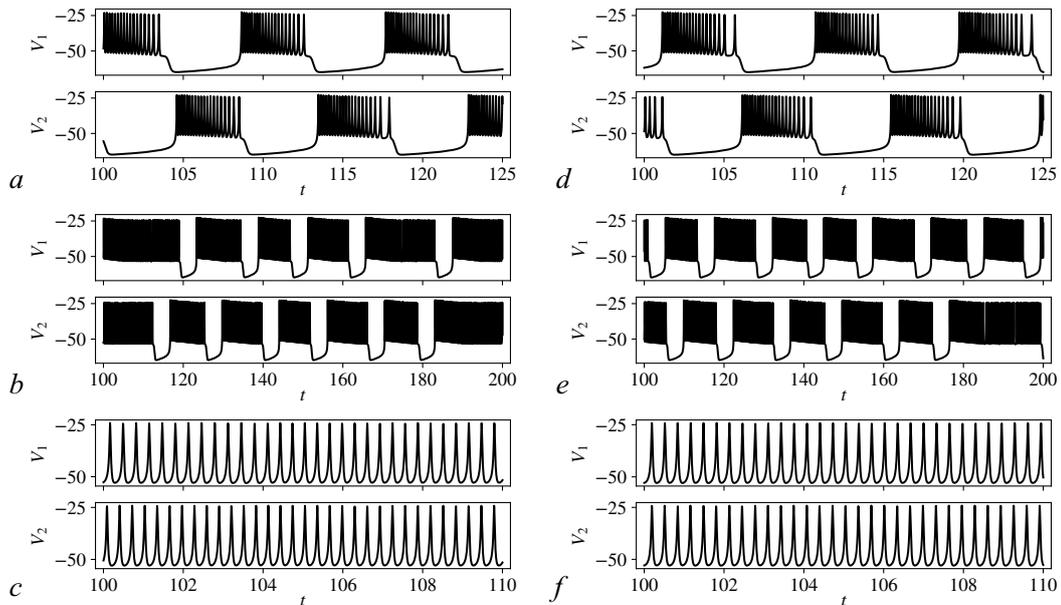


Рис. 7. Решения для связанных оригинальных систем: $a-c$ – уравнения (4); $d-f$ – соответствующие связанные нейросетевые отображения (11). Представлены режимы: a и d – пачечный режим при $V_{S,1} = -36$; b и e – нерегулярные пачки при $V_{S,1} = -33$; c и f – спайки при $V_{S,1} = -31$. Во всех случаях $V_{S,2} = V_{S,1} + 0.1$. Параметр связи $g_{c,V} = 0.001$

Fig. 7. Solutions for the coupled original systems: $a-c$ – Eqs. (4); $d-f$ – corresponding coupled neural network mappings (11). Represented regimes: a and d – bursts with $V_{S,1} = -36$; b and e – irregular bursts with $V_{S,1} = -33$; c and f – spikes at $V_{S,1} = -31$. In all cases $V_{S,2} = V_{S,1} + 0.1$. The coupling parameter is $g_{c,V} = 0.001$

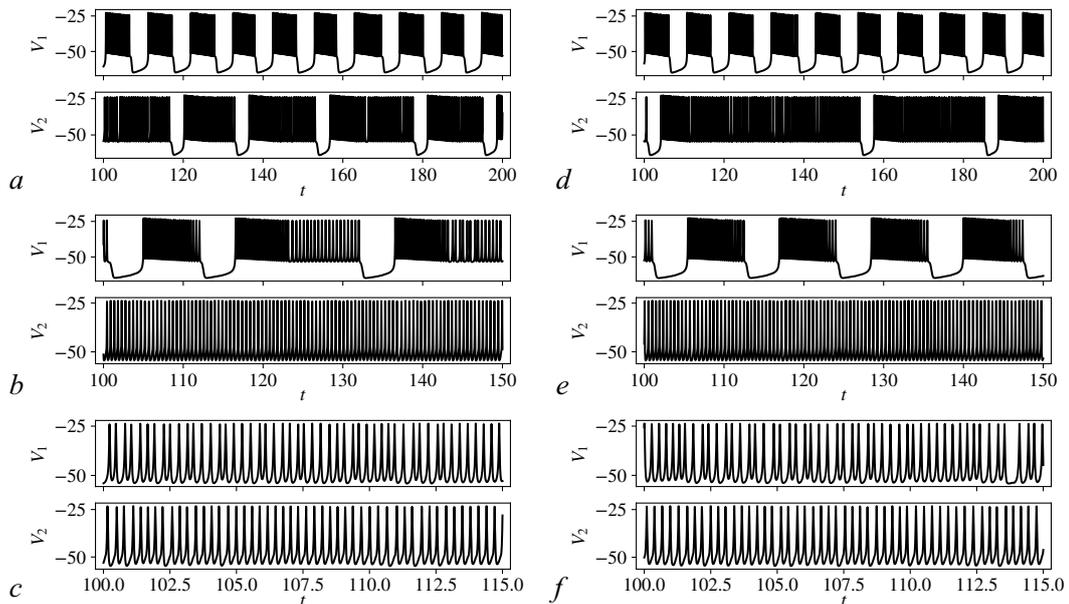


Рис. 8. Решение для пары оригинальная–модифицированная системы: $a-c$ – уравнения (4); $d-f$ – связанные нейросетевые отображения (11). Представлены режимы: a и d – нерегулярные пачки разной длительности при $V_S = -34.4$; b и e – пачки и спайки при $V_S = -33.3$; c и f – нерегулярные спайки при $V_S = -30.5$. Во всех случаях $V_{S,2} = V_{S,1} + 0.1$. Параметр связи: a, b, d и e – $g_{c,V} = 0.001$; c и f – $g_{c,V} = 0.01$

Fig. 8. Solution for the coupled original and modified systems: $a-c$ – Eqs. (4); $d-f$ – coupled neural network mappings (11). Represented regimes: a and d – irregular bursts of different lengths when $V_S = -34.4$; b and e – bursts and spikes at $V_S = -33.3$; c and f – irregular spikes at $V_S = -30.5$. In all cases $V_{S,2} = V_{S,1} + 0.1$. Coupling parameter: a, b, d and e – $g_{c,V} = 0.001$; c and f – $g_{c,V} = 0.01$

нерегулярной длины разные для разных подсистем, а рис. 8, *b* демонстрирует сосуществование пачечного и спайкового режимов для первой и второй подсистем.

При увеличении силы связи между оригинальной и модифицированной системами значительно сокращается область устойчивости неподвижной точки, а переходная область становится ещё шире (см. рис. 6, *d*). На рис. 8, *c* показано, что сразу после переходной области в такой системе возникает нерегулярный спайковый режим.

На рис. 9, *a–c* показаны одномерные сечения бассейнов притяжения различных режимов связанных оригинальной и модифицированной систем при увеличении силы связи $g_{c,V}$.

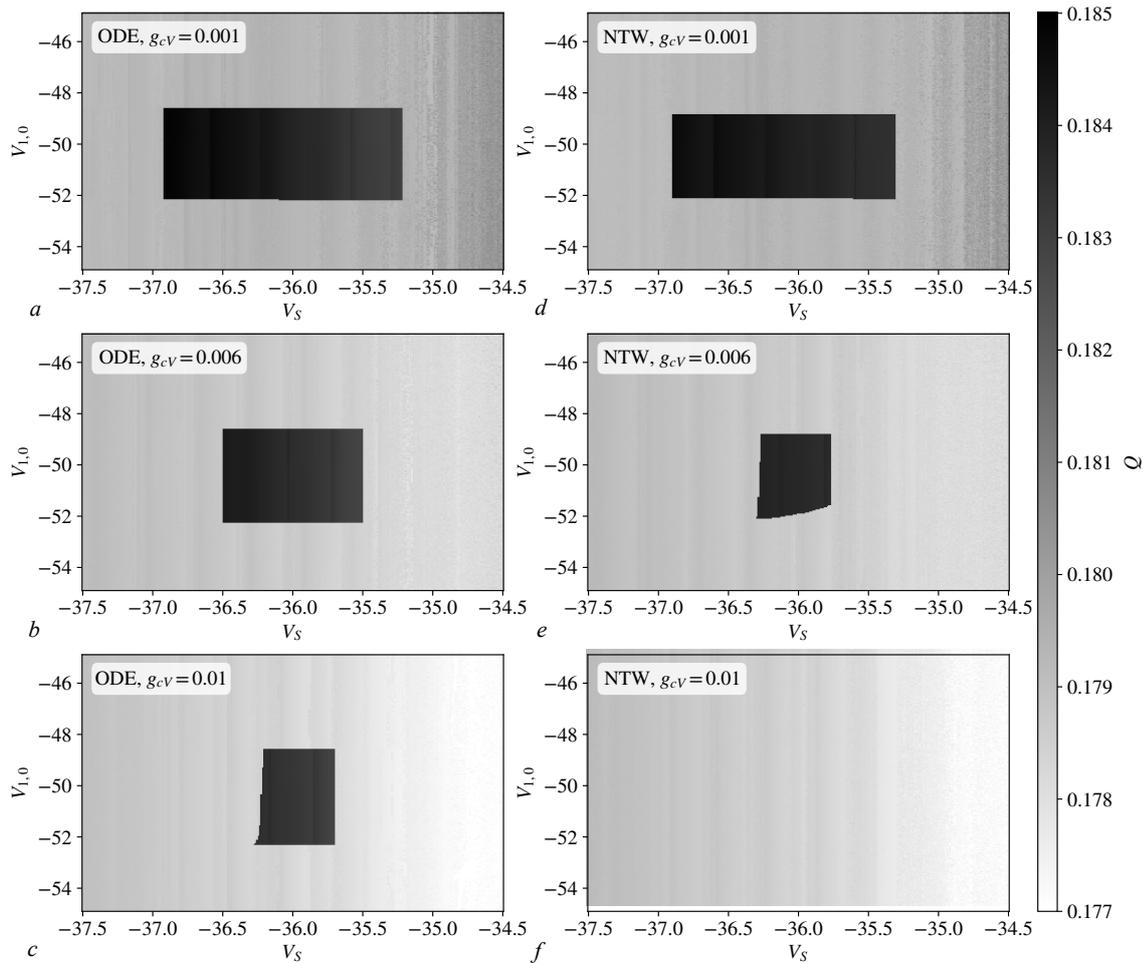


Рис. 9. Одномерные сечения бассейнов притяжения для пары оригинальная–модифицированная системы: *a, b, c* – система уравнений (4); *d, e, f* – связанные нейросетевые отображения (11). Параметр связи $g_{c,V}$: 0.001 (*a, d*); 0.006 (*b, e*); 0.01 (*c, f*). Диаграммы построены в окрестности неподвижной точки $V_{1,f} = -49.8965$, $n_{1,f} = 2.34541 \times 10^{-3}$, $S_{1,f} = 0.199464$, $V_{2,f} = -50.5546$, $n_{2,f} = 2.08592 \times 10^{-3}$, $S_{2,f} = 0.187634$, вычисленной при $V_S = -36$ и $g_{c,V} = 0.001$. Начальное значение $V_{1,0}$ отложенное по вертикали варьируется в пределах $V_{1,f} \pm 0.1V_{1,f}$, начальное значение $V_{2,0}$ также варьируется: $V_{2,0} = V_{1,0} - V_{1,f} + V_{2,f}$, начальные значения остальных переменных выбираются равными приведённым выше значениям

Fig. 9. One-dimensional sections of attraction basins for coupled original and modified system: *a, b, c* – Eqs. (4); *d, e, f* – coupled neural network mappings (11). Coupling parameter $g_{c,V}$: 0.001 (*a, d*); 0.006 (*b, e*); 0.01 (*c, f*). The diagrams are obtained in the neighborhood of a fixed point $V_{1,f} = -49.8965$, $n_{1,f} = 2.34541 \times 10^{-3}$, $S_{1,f} = 0.199464$, $V_{2,f} = -50.5546$, $n_{2,f} = 2.08592 \times 10^{-3}$, $S_{2,f} = 0.187634$ calculated at $V_S = -36$ and $g_{c,V} = 0.001$. Initial values of $V_{1,0}$ plotted vertically vary within the range $V_{1,f} \pm 0.1V_{1,f}$, the initial value of $V_{2,0}$ also varies: $V_{2,0} = V_{1,0} - V_{1,f} + V_{2,f}$, and initial values of other variables are chosen equal to the values given above

Этот рисунок построен аналогично рис. 4 и 5. Только теперь одновременно меняются начальные значения переменной V как для первой, так и для второй подсистем. Чёрная область в центральной части представляет стартовые значения, из которых система выходит на неподвижную точку. Светло-серые области вокруг — точки, из которых система выходит на колебательный режим. Видно, что с увеличением силы связи $g_{c,V}$ область притяжения неподвижной точки уменьшается.

3. Нейросетевое отображение

Нас интересует построение отображения вида $u(t + \Delta t) = F(u(t), p, w)$, способного при правильном подборе числовых значений w воспроизводить поведение различных динамических систем. Здесь $u(t)$ — вектор динамических переменных, p — вектор управляющих параметров, Δt — шаг по времени, который мы будем задавать как фиксированную величину. Математически доказано, см. работы [5–10], что двухслойная полносвязная нейронная сеть может быть использована для аппроксимации произвольных функций многих переменных. Отсюда следует, что она должна быть пригодна для построения упомянутого выше универсального отображения. Этот вопрос исследуется в работе [16]. Было показано достаточно хорошее качество воспроизведения динамики для различных систем: Лоренца, Рёсслера, модели нейрона Хиндмарша–Роуза. Однако было обнаружено, что нейросетевое отображение на основе простой двухслойной архитектуры достаточно трудно обучить для систем жёсткого типа, у которых переменные имеют сильно отличающиеся временные масштабы. Поэтому в работе [17] была предложена более сложная архитектура. Вместо одной сети, получающей на вход вектор $u(t)$ и возвращающей вектор $u(t + \Delta t)$, для каждой динамической переменной теперь создаётся отдельная подсеть — также двухслойная полносвязная. При этом все остальные переменные и значения управляющих параметров вводятся после прохождения ещё одного, дополнительного полносвязного слоя. Такая сеть оказалась в состоянии не только моделировать динамику жёсткой системы, но и правильно воспроизводить поведение, образцы которого не были предъявлены ей в ходе обучения.

В настоящей работе мы рассмотрим нейросетевое отображение из работы [17] с незначительной модификацией:

$$u_i(t + \Delta t) = (1 - \chi)u_i(t) + \chi \left(f \left(u_i(t)a_i + \mu_i + g(u_{-i}(t)A_i + pB_i + \beta_i) \right) b_i + \gamma_i \right). \quad (5)$$

Модификация сводится к введению стабилизирующего множителя $\chi = 0.001$. Благодаря этому исключаются ситуации, когда в результате «неудачной» инициализации весовых коэффициентов случайными числами обученное отображение вместо ожидаемого поведения демонстрировало расходимость. Здесь $u(t)$ — вектор динамических переменных системы размера D_u , $u_i(t)$ — i -ая переменная, а $u_{-i}(t)$ — вектор, получаемый из $u(t)$ удалением i -й переменной. В нашем случае вектор $u(t)$ содержит компоненты V , n и S . Как принято для нейронных сетей, мы будем считать, что $u(t)$ и $u_{-i}(t)$ — это векторы-строки. Символом p обозначен вектор-строка управляющих параметров системы размерности D_p . Мы рассматриваем изменение только одного из параметров системы, а именно V_S . Поэтому он одномерный. Описание других элементов формулы (5) приведено в табл. 2. Функции $f(\cdot)$ и $g(\cdot)$ в терминах нейронных сетей называются функциями активации. Это скалярные функции скалярных аргументов. Предполагается, что когда они применяются к векторам, то действуют поэлементно.

Структура формулы (5) соответствует структуре двухслойной полносвязной сети. Первый слой на вход получает скалярное значение i -й динамической переменной u_i , а его выход — это вектор размерности N_h (см. табл. 2), получаемый после вычисления выражения $f(u_i(t)a_i + \mu_i + g(\dots))$. Этот слой называется скрытым. Однако в отличие от простой двухслойной сети, здесь вектор смещений (bias) μ_i корректируется значениями, подаваемыми через дополнительный слой

Таблица 2. Описание элементов формулы (5)

Table 2. Description of the elements of Eq. (5)

$u(t)$	вектор-строка	$D_u = 3$
$u_i(t)$	скаляр	
$u_{-i}(t)$	вектор-строка	$D_u - 1$
p	вектор-строка	$D_p = 1$
a_i, μ_i и β_i	векторы-строки	$N_h = 100$
b_i	вектор-столбец	N_h
γ_i	скаляр	
A_i	матрица	$(D_u - 1) \times N_h$
B_i	матрица	$D_p \times N_h$
χ	константа	0.001
$f(\cdot), g(\cdot)$	скалярные функции	$\tanh(\cdot)$

$g(u_{-i}(t)A_i + pB_i + \beta_i)$, который зависит от других динамических переменных и параметров. Второй, выходной слой сети имеет вид: $f(\dots)b_i + \gamma_i$. Этот слой возвращает скалярное значение, которое далее используется для вычисления значения динамической переменной u_i на новом шаге времени как $(1 - \chi)u_i + \chi(\dots)$. Таким образом, нейросетевое отображение (5) сконструировано как нейронная сеть, которая подстраивается под требуемую систему посредством обучения весовых параметров, представленных в виде следующих матриц и векторов:

$$w = \{a_i, \mu_i, A_i, B_i, \beta_i, b_i, \gamma_i \mid i = 1, 2, 3\}. \quad (6)$$

При фактической реализации нейросетевого отображения (5) требуется принять во внимание, что все современные вычислительные инструменты по работе с нейронными сетями, равно как и соответствующие теоретические разработки, подразумевают стандартизацию набора данных, на которых производится обучение. Это означает, что все величины, подаваемые на вход сети, должны попадать в единичный интервал вблизи нуля. Поэтому сеть будет обучаться и функционировать на данных, перемасштабированных согласно формулам (подразумеваются поэлементные операции)

$$u \rightarrow (u - m_u)/s_u, p \rightarrow (p - m_p)/s_p. \quad (7)$$

Здесь m_u, m_p, s_u и s_p — это, соответственно, векторы средних значений и стандартных отклонений по области определения нейросетевого отображения. Вычисление этих векторов обсуждается в разделе 4, а их значения приведены в табл. 3. Таким образом, исходные данные, подаваемые на вход отображения (5), перемасштабируются согласно (7), затем выполняется требуемое число итераций, а для представления результатов полученные значения перемасштабируются обратно.

Таблица 3. Параметры наборов данных

Table 3. Datasets parameters

$V_S \in [-40, -30], V \in [-70, -18], n \in [0.0, 0.13], S \in [0.14, 0.26],$ $m_u = (-44, 0.065, 0.2), m_p = -35, s_u = (26, 0.065, 0.06), s_p = 5,$ $L_{\text{chunk}} = 10, N_{\text{chunks}} = 10^5, \Delta t = 0.005,$ $L_{\text{train}} = L_{\text{chunk}} \times N_{\text{chunks}} = 10^6, L_{\text{validation}} = 10^5, L_{\text{batch}} = 10^4$

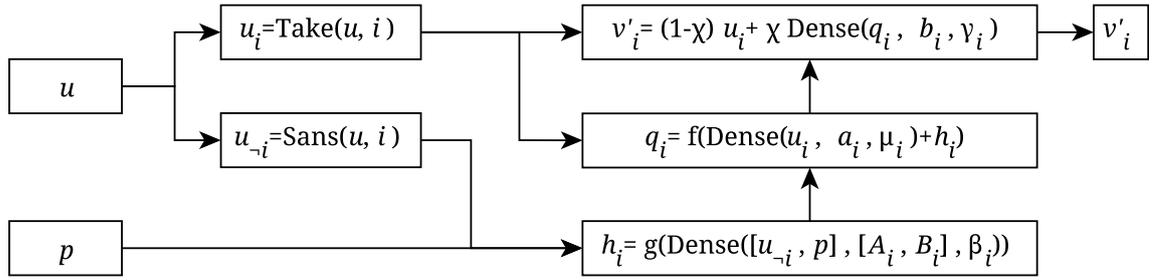


Рис. 10. Структура нейронной сети, соответствующей нейросетевому отображению (5) для i -й переменной моделируемой системы. Здесь $u \equiv u(t)$, $v'_i \equiv u'_i(t + \Delta t)$. Штрих обозначает вычисленную сеть величину, которая при обучении сравнивается с «правильной» величиной $u_i(t + \Delta t)$ из обучающего набора данных

Fig. 10. The structure of the neural network that corresponds to the mapping (5) for i th variable. Here $u \equiv u(t)$, $v'_i \equiv u'_i(t + \Delta t)$. The prime denotes a value computed by the network. This value in the course of training is compared with the “correct” value $u_i(t + \Delta t)$ from the training dataset

При программной реализации отображения в виде нейронной сети используются следующие операторы. Оператор $\text{Take}(u, i)$ возвращает i -й элемент вектора u , а оператор $\text{Sans}(u, i)$ наоборот, удаляет i -й элемент и возвращает вектор u_{-i} без этого элемента. Оператор $\text{Dense}(x, W, b) = xW + b$ представляет полносвязный слой без функции активации: умножение входного вектора x на матрицу W и сдвиг на вектор b . Наконец, квадратные скобки $[\cdot, \cdot]$ обозначают конкатенацию двух векторов или матриц.

Используя эти обозначения, действие нейросетевого отображения можно проиллюстрировать при помощи диаграммы на рис. 10 и соответствующих ей формул (8), которые показывают, как организованы вычисления для i -й переменной. Нейронная сеть имеет два входных вектора, $p = (V_S)$ и $u = (V(t), n(t), S(t))$, формула (8a). Сначала u разбивается на скаляр u_i и вектор u_{-i} , в котором i -й элемент отсутствует, формула (8b). Слой (8c) принимает u_{-i} и p и вычисляет вектор h_i размерности N_h , который включает в себя информацию о влиянии управляющих параметров и всех переменных за исключением i -й. Этот вектор складывается с выходом слоя, обрабатывающего i -ю компоненту, формула (8d). В результате получается вектор скрытого слоя q_i размерности N_h . Он передаётся на выходной слой (8e), после чего используется непосредственно для вычисления решения на новом шаге времени $v'_i \equiv u(t + \Delta t)$. Повторив эти вычисления для $i = 1, 2, 3$ мы получаем сгенерированный сетью вектор решения $u'(t + \Delta t)$.

$$p = \text{Input}(), \quad u = \text{Input}(), \quad (8a)$$

$$u_i = \text{Take}(u, i), \quad u_{-i} = \text{Sans}(u, i), \quad (8b)$$

$$h_i = g(\text{Dense}([u_{-i}, p], [A_i, B_i], \beta_i)), \quad (8c)$$

$$q_i = f(\text{Dense}(u_i, a_i, \mu_i) + h_i), \quad (8d)$$

$$v'_i = (1 - \chi)u_i + \chi \text{Dense}(q_i, b_i, \gamma_i). \quad (8e)$$

Когда сеть уже обучена и используется для вычисления траектории системы, нужно положить $u'(t + \Delta t) = u(t + \Delta t)$. В процессе же обучения весовые коэффициенты сети (6) настраиваются таким образом, чтобы отличие генерируемого результата $u'(t + \Delta t)$ от требуемого $u(t + \Delta t)$ было минимальным. Поэтому в качестве целевой функции естественно выбрать квадрат нормы разности этих векторов:

$$L = \|v - v'\|^2. \quad (9)$$

Обучение состоит в том, чтобы используя метод градиентного спуска, шаг за шагом вычислять поправки к весовым коэффициентам (6), приводящие к уменьшению L :

$$w \rightarrow w - \kappa \nabla_w L, \quad (10)$$

где κ — обычно малый параметр, регулирующий величину шага. Фактически при обучении мы используем модификацию метода градиентного спуска — метод Adam [27]. Это наиболее часто применяемый в настоящее время алгоритм оптимизации весов сети. Его преимущество в том, что значение κ автоматически подстраивается в процессе итераций, что обеспечивает высокую скорость их сходимости.

4. Тренировочный набор данных и обучение сети

Мы создаём нейронную сеть, которая в некоторой степени похожа на так называемую рекуррентную сеть [11]: на вход подаётся вектор $u(t)$, на выходе формируется $u(t + \Delta t)$, который снова подаётся на вход той же самой сети для вычисления следующего значения. Однако имеются важные отличия. Во-первых, внутренняя структура сети подобрана так, чтобы имелась возможность описать её в форме отображения, явного и достаточно компактного, допускающего дальнейший теоретический анализ, см. формулу (5). Современные рекуррентные ячейки, такие как LSTM и GRU, устроены значительно сложнее [11]. Для них нет большого смысла пытаться выписать соответствующее отображение из-за его громоздкости. Во-вторых, процедура обучения нашей сети будет организована по-другому. Традиционные рекуррентные сети обучаются на последовательностях фиксированной, обычно не очень большой длины. Это предполагает что и функционировать они будут на последовательностях такой же длины. Нейросетевое отображение (5) должно быть способно порождать траектории произвольной длины. Поэтому в процессе обучения мы не задействуем рекурсию — в качестве обучающих данных будут использованы пары $(u(t), u(t + \Delta t))$, где первое значение подаётся на вход, сеть делает только одну итерацию и вычисленный результат $u'(t + \Delta t)$ сравнивается с ожидаемым $u(t + \Delta t)$.

Для формирования набора данных зададим область определения таким образом, чтобы в ней оказались все инвариантные объекты фазового пространства — аттракторы колебательных решений и неподвижная точка. Для этого сначала зададим диапазон изменения параметра V_S , см. табл. 3. Напомним, что мы будем менять значения только одного этого параметра, значения остальных приведены в табл. 1. Меняя V_S , будем строить численные решения системы (1), многократно стартуя из произвольно выбранных начальных точек и определяя минимальные и максимальные значения, которые принимают V , n и S . Дополнительно будем вычислять положение неподвижной точки при различных V_S и корректировать найденные диапазоны изменения переменных, чтобы включить в них и её. Найденные диапазоны затем расширим примерно по 10% с каждого края и округлим до двух значащих цифр. При этом нижняя граница для n немного заходит в отрицательную область — заменяем её нулём. Полученные диапазоны для V , n и S вместе с диапазоном для V_S , см. табл. 3, задают область определения нейросетевого отображения, когда оно моделирует систему (1). Отметим, что в работе [17], где основная цель — моделирование одиночной системы, обучение производилось на фрагментах колебательного решения системы. Так как в настоящей работе исследуется возможность моделирования связанных систем (4), сеть обучается на всей области определения.

Для полученной области определения вычисляются средние значения m_u и m_p , а также соответствующие стандартные отклонения s_u и s_p , см. табл. 3. Данные, подаваемые на вход сети, то есть в отображение (5), предварительно перемасштабируются по формулам (7) так,

что фактически значения всех переменных и параметра лежат в диапазоне от -1 до 1 . После вычисления траектории данные, сгенерированные сетью, подвергаются обратному преобразованию для последующего анализа.

Обучающий набор данных создаётся следующим образом. Случайно выбираются значение параметра V_S и начальное значение динамических переменных $u(0) = (V, n, S)$ из области определения, границы которой приведены в табл. 3. Затем вычисляется решение уравнений (1) за $L_{\text{chunk}} + 1$ шагов с временным интервалом Δt , см. табл. 3. Из полученной последовательности конструируются L_{chunk} записей вида $(V_S, u(0), u(\Delta t))$, $(V_S, u(\Delta t), u(2\Delta t))$, $(V_S, u(2\Delta t), u(3\Delta t))$ и так далее, которые помещаются в массив обучающих данных. В целом для тренировочного набора данных генерируется N_{chunks} отрезков траекторий так, что полная длина тренировочного набора данных равна $L_{\text{train}} = L_{\text{chunk}} \times N_{\text{chunks}}$. Перед началом обучения записи в тренировочном наборе перемешиваются, так что сеть «видит» в каждый момент обучения отдельные точки траекторий, а не фрагменты целиком.

Для контроля качества обучения генерируется валидационный набор данных длиной $L_{\text{validation}}$. Для валидации всегда берутся отрезки траектории длиной один шаг. Это значит, что для сгенерированных случайно V_S и $u(0)$ вычисляется $u(\Delta t)$, запись $(V_S, u(0), u(\Delta t))$ помещается в массив валидационных данных, а затем генерируется новая запись.

Стандартный способ экономии памяти в процессе обучения — подавать на вход сети данные порциями, которые в англоязычной литературе называются «batches», что переводится на русский обычно как «пакеты». Для нашей сети размер пакета равен L_{batch} , см. табл. 3. Сначала сети предъявляется тренировочный набор данных, пакет за пакетом. Для каждого пакета выполняется шаг градиентного спуска (10) и корректируются веса сети w . Когда все пакеты тренировочного набора обработаны, говорят что прошла одна эпоха обучения. После этого сети предъявляются валидационные данные, для которых вычисляется значение целевой функции. Градиентный спуск и коррекция весовых коэффициентов на валидационных данных не производится. Кривые зависимости целевой функции от номера эпохи для тренировочных и валидационных данных называются кривыми обучения.

Результат обучения нейронной сети существенным образом зависит от длины фрагментов траекторий L_{chunk} . На рис. 11 показаны кривые обучения на наборе данных модифицированной системы. По вертикали отложено значение целевой функции (9), а по вертикали — номер эпохи

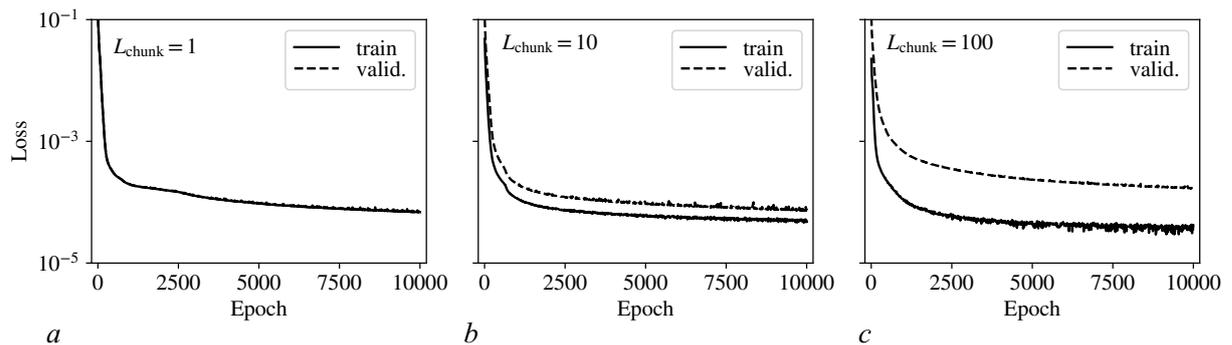


Рис. 11. Кривые обучения нейросетевого отображения (5) для модифицированной системы: a , b и c — длины фрагментов обучающих L_{chunk} траектории равны 1, 10 и 100, соответственно. Показаны тренировочная и валидационная кривые. На диаграмме a эти кривые совпадают

Fig. 11. Learning curves of the neural network mappings (5) for modified system: a , b and c — lengths of the training trajectory chunks L_{chunk} are 1, 10 and 100, respectively. Both training and validation curves are shown. On the diagram a the curves coincide

обучения. Рис. 11, *a, b, c* получены для наборов данных с разными длинами отрезков траекторий, $L_{\text{chunk}} = 1$, $L_{\text{chunk}} = 10$, $L_{\text{chunk}} = 100$, соответственно. Видно, что чем выше L_{chunk} , тем сильнее расходятся тренировочная и валидационная кривые: целевая функция на тренировочных данных остаётся на прежнем уровне, а валидация становится хуже.

Так как в отличие от тренировочных данных на валидационных данных подстройка весовых коэффициентов сети w не производится, то есть не производится адаптация сети непосредственно к этим данным, сравнение значений целевой функции на тренировочных и валидационных данных позволяет судить о том, насколько хорошо сеть научилась находить обобщённые признаки, а следовательно, насколько хорошо она будет функционировать по окончании обучения. Сильное расхождение между валидационной и тренировочной кривыми говорит о переобучении. Это значит, что сеть преимущественно просто запоминает тренировочные данные. Пример такого поведения показан на рис. 11, *c* — когда фрагменты траектории длинные, сеть более «охотно» запоминает именно их, тогда как на произвольных точках фазового пространства её поведение заметно отличается от поведения моделируемой системы.

С другой стороны, уменьшение L_{chunk} также приводит к негативному эффекту. На рис. 4, *b, c* и *d* показаны одномерные сечения бассейнов притяжения различных режимов отображения (5), обученного с $L_{\text{chunk}} = 10$, $L_{\text{chunk}} = 1$, $L_{\text{chunk}} = 100$, соответственно. Сравнивая эти диаграммы с рис. 4, *a*, который построен непосредственно для численных решений системы (1), видим, что сеть, обученная на одношаговых отрезках траекторий, $L_{\text{chunk}} = 1$, вообще не «видит» устойчивую неподвижную точку — на рис. 4, *c* отсутствует характерный тёмный прямоугольник в центре, который есть на всех других диаграммах. Переобученная сеть при $L_{\text{chunk}} = 100$ также плохо воспроизводит особенности динамики системы (1). Видно, что область устойчивости неподвижной точки (ширина тёмного центрального прямоугольника), а также точка перехода от пачечного режима к спайковому (граница между светло-серой и тёмно-серой областями) отличаются от правильных значений на рис. 4, *a*. Таким образом, видно, что оптимальное значение для обучения нейросетевого отображения (5) для воспроизведения модифицированной системы — это $L_{\text{chunk}} = 10$.

Для оригинальной системы не показаны кривые обучения, так как они визуально неотличимы от соответствующих кривых для модифицированной системы на рис. 11. Одномерное сечение бассейнов притяжения в зависимости от параметра V_S для нейросетевого отображения (5), обученного при $L_{\text{chunk}} = 10$, показано на рис. 5, *b*. Видно, что оно хорошо воспроизводит диаграмму для моделируемой системы (1).

Таким образом, для моделирования динамики оригинальной и модифицированной версий системы (1) наилучшим образом подходят наборы обучающих данных, содержащих отрезки траекторий длиной $L_{\text{chunk}} = 10$. В дальнейшем мы рассматриваем нейросетевые отображения, обученные на таких наборах данных.

5. Динамика нейросетевого отображения

Нейросетевое отображение (5), обученное как описано выше, хорошо воспроизводит динамику оригинального и модифицированного вариантов системы (1). На рис. 1, *d, e* и *f* показаны зависимости от времени для модифицированной системы, сгенерированные нейросетевым отображением. Видно очень хорошее соответствие с рис. 1, *a, b* и *c*, полученными как численные решения (1). Здесь не показаны аналогичные пары решений для оригинальной системы, так как они также хорошо соответствуют друг другу и визуально неотличимы от показанных на рис. 1.

Изменения характера динамики в зависимости от параметра V_S демонстрирует рис. 3. Видно, что нейросетевое отображение хорошо воспроизводит бифуркационную перестройку от пачечного к спайковому аттракторам как в оригинальной (сравни рис. 3, *a* и *c*), так

и в модифицированной (сравни рис. 3, *b* и *d*) системах. Кроме того, для модифицированной системы нейросетевое отображение правильно повторяет область бистабильности, в которой пачечный аттрактор сосуществует с неподвижной точкой.

Как уже упоминалось выше, на рис. 4, *a* и *b* показано, что нейросетевое отображение с высокой степенью точности воспроизводит структуру бассейнов притяжения различных режимов системы. Аналогичное соответствие бассейнов притяжения для оригинальной системы показано на рис. 5.

Обсудим теперь связанные системы. Мы собираемся продемонстрировать, что если между нейросетевыми отображениями, обученными для одиночных систем, ввести связь, они будут воспроизводить динамику связанных систем (4).

Так как в уравнениях (4) связь вводится через компоненты V_1 и V_2 , при переходе к отображениям нужно модифицировать соответствующие части нейросетевого отображения (5), добавив к ним отвечающее за связь слагаемое, пропорциональное $g_{c,V}$:

$$V_1(t + \Delta t) = F_0^{(1)}(V_1(t), n_1(t), S_1(t)) + \frac{\Delta t g_{c,V}}{\tau} (V_1(t) - V_2(t)). \quad (11)$$

Здесь $F_0^{(1)}$ обозначает правую часть нейросетевого отображения для нулевой компоненты первой подсистемы, то есть для V_1 . Отображение для V_2 получается заменой индексов $1 \leftrightarrow 2$. Все остальные компоненты вычисляются как и раньше, с использованием соответствующего нейросетевого отображения вида (5).

Таким образом, для моделирования динамики связанных систем (4) мы будем использовать нейросетевые отображения, обученные для одиночных систем, как описано в разделах 3 и 4, вводя между ними связь вида (11).

На рис. 6 сравниваются трансформации режимов динамики связанных осцилляторов (4) с соответствующей нейросетевой моделью (11). Левая колонка получена на основе численных решений уравнений (4), а колонка справа — для итераций связанных нейросетевых отображений. Сопоставлены пары оригинальная—оригинальная (см. рис. 6, *a* и *e*), модифицированная—модифицированная (см. рис. 6, *b* и *f*), и оригинальная—модифицированная подсистемы (см. рис. 6, *c*, *d*, *g* и *h*). Для последней показаны случаи двух разных значений силы связи $g_{c,V}$. Видно очень хорошее соответствие рисунков. Характер перестроек режимов, демонстрируемый нейросетевыми отображениями, соответствует перестройкам в системе, заданной уравнениями (4). Это также иллюстрируют рис. 7 и 8, на которых показаны примеры решений при разных значениях V_S . На этих рисунках правые колонки получены как численные решения (4), а справа — итерации связанных нейросетевых отображений. Мы наблюдаем очень хорошее соответствие во всех случаях.

Обсудим теперь ситуацию, когда поведение связанных нейросетевых отображений соответствует поведению системы (4) недостаточно точно. На рис. 6, *c* и *d*, которые построены для пары оригинальная—модифицированная подсистемы при $g_{c,V} = 0.001$ и $g_{c,V} = 0.01$, в районе $V_S = -36$ наряду с колебательным решением имеется диапазон устойчивости неподвижной точки. Отметим низкую насыщенность изображения этого решения, что свидетельствует о низкой вероятности выхода на него системы при старте со случайных начальных точек. Также отметим, что ширина по V_S диапазона существования устойчивой неподвижной точки уменьшается с ростом $g_{c,V}$. Соответствующее нейросетевое отображение очень хорошо воспроизводит область устойчивости неподвижной точки при $g_{c,V} = 0.001$ (сравни рис. 6, *g* и *c*). Однако с увеличением $g_{c,V}$ до 0.01 у нейросетевого отображения она исчезает (сравни рис. 6, *h* и *d*). Эта ситуация проанализирована на рис. 9. Здесь показаны одномерные сечения бассейнов притяжения различных режимов в зависимости от V_S для пары оригинальная—модифицированная системы для разных значений силы связи. Рис. 9, *a–c* построены для решений связанных уравнений (4),

а рис. 9, *в–f* — для связанных нейросетевых отображений. Чёрная центральная область на этих рисунках представляет значения переменных, стартовав из которой система выходит на неподвижную точку. При слабой связи эти области практически идентичны для уравнений и для нейросетевых отображений. Однако с ростом $g_{c,V}$ видоизменение области происходит по-разному: в обоих случаях она сужается по V_S , но для отображений это происходит быстрее. Отличие хорошо видно при $g_{c,V} = 0.006$ на рис. 9, *б* и *е*: чёрная центральная область на рисунке справа значительно уже, чем слева. Далее, при $g_{c,V} = 0.01$ область выхода на неподвижную точку для уравнений ещё сохраняется, а для нейросетевых отображений уже отсутствует. Это говорит о том, что с ростом силы связи в системе нейросетевых отображений неподвижная точка полностью теряет устойчивость раньше чем в моделируемых уравнениях.

Таким образом, в целом можно заключить, что связанные нейросетевые отображения, обученные по отдельности, в целом хорошо воспроизводят поведение связанных систем. Для неподвижной точки наблюдаются незначительные расхождения в границах области её устойчивости, наблюдаемые при изменении силы связи.

Заключение

Мы проанализировали процедуру построения и обучения нейросетевого отображения, способного моделировать динамику нелинейных систем. Детально рассмотрена архитектура такого отображения, метод формирования набора данных и непосредственно процесс обучения. В качестве моделируемой системы рассматривался нейрон, задаваемый уравнениями на основе формализма Ходжкина—Хаксли [18]. Отличительная особенность динамики нейрона — её жёсткость, то есть наличие сильно различающихся временных масштабов изменения переменных. Это выступает дополнительным фактором, затрудняющим построение соответствующего нейросетевого отображения и требует использования специальной архитектуры нейронной сети — каждая переменная должна моделироваться отдельной подсетью [17].

Основной целью настоящей работы было показать, что предложенное нейросетевое отображение можно использовать в составе связанных систем — связанные нейросетевые отображения, обученные для одиночных систем, могут без дополнительного обучения воспроизводить динамику соответствующих связанных нейронов. На примере связанных нейронов показано хорошее соответствие динамики исходных уравнений и связанных нейросетевых отображений. Хорошо воспроизводятся перестройки характера поведения при изменении параметра, в частности — появление и исчезновение бистабильности.

Развиваемый в работе подход сформулирован в максимально общем виде. Рассматриваемое нейросетевое отображение можно попробовать построить для любых динамических систем и ввести произвольные связи между ними. При этом вопрос о границах его применимости пока не проработан. В частности, не ясно, можно ли указать практически интересные, не «экзотические» примеры динамических систем, для которых обучение нейросетевого отображения невозможно. Также представляет интерес вопрос о том, способны ли нейросетевые отображения без дообучения хорошо воспроизводить динамику более двух связанных систем с разными, в том числе нелинейными, связями. Все эти вопросы требуют дальнейшего изучения.

Список литературы

1. *Levin E., Gewirtzman R., Inbar G. F.* Neural network architecture for adaptive system modeling and control // *Neural Networks*. 1991. Vol. 4, no. 2. P. 185–191. DOI: 10.1016/0893-6080(91)90003-N.
2. *Grieger B., Latif M.* Reconstruction of the El Niño attractor with neural networks // *Climate Dynamics*. 1994. Vol. 10, no. 6–7. P. 267–276. DOI: 10.1007/BF00228027.
3. *Zimmermann H. G., Neuneier R.* Combining state space reconstruction and forecasting by neural

- networks // In: Bol G., Nakhaeizadeh G., Vollmer K. H. (eds) *Datamining und Computational Finance*. Vol. 174 of *Wirtschaftswissenschaftliche Beiträge*. Heidelberg: Physica, 2000. P. 259–267. DOI: 10.1007/978-3-642-57656-0_13.
4. *Gilpin W., Huang Y., Forger D. B.* Learning dynamics from large biological data sets: Machine learning meets systems biology // *Current Opinion in Systems Biology*. 2020. Vol. 22. P. 1–7. DOI: 10.1016/j.coisb.2020.07.009.
 5. *Колмогоров А. Н.* О представлении непрерывных функций нескольких переменных суперпозициями непрерывных функций меньшего числа переменных // *ДАН СССР*. 1956. Т. 108, № 2. С. 179–182.
 6. *Арнольд В. И.* О функциях трех переменных // *ДАН СССР*. 1957. Т. 114, № 4. С. 679–681.
 7. *Колмогоров А. Н.* О представлении непрерывных функций нескольких переменных в виде суперпозиций непрерывных функций одного переменного и сложения // *ДАН СССР*. 1957. Т. 114, № 5. С. 953–956.
 8. *Sybenko G.* Approximation by superpositions of a sigmoidal function // *Mathematics of Control, Signals and Systems*. 1989. Vol. 2, no. 4. P. 303–314. DOI: 10.1007/BF02551274.
 9. *Горбань А. Н.* Обобщенная аппроксимационная теорема и точное представление многочленов от нескольких переменных суперпозициями многочленов от одного переменного // *Известия вузов. Математика*. 1998. № 5 (432). С. 6–9.
 10. *Хайкин С.* *Нейронные сети: полный курс*. 2-е издание. М.: Издательский дом «Вильямс», 2006. 1104 с.
 11. *Николенко С., Кадурич А., Архангельская Е.* *Глубокое обучение*. СПб.: Питер, 2018. 480 с.
 12. *Cook S.* *CUDA Programming: A Developer's Guide to Parallel Computing with GPUs*. Morgan Kaufmann, 2012. 592 p.
 13. *Jouppi N. P., Young C., Patil N., Patterson D., Agrawal G., Bajwa R., Bates S., Bhatia S., Boden N., Borchers A., Boyle R., Cantin P.-L., Chao C., Clark C., Coriell J., Daley M., Dau M., Dean J., Gelb B., Ghaemmaghami T. V., Gottipati R., Gulland W., Hagmann R., Ho C. R., Hogberg D., Hu J., Hundt R., Hurt D., Ibarz J., Jaffey A., Jaworski A., Kaplan A., Khaitan H., Killebrew D., Koch A., Kumar N., Lacy S., Laudon J., Law J., Le D., Leary C., Liu Z., Lucke K., Lundin A., MacKean G., Maggiore A., Mahony M., Miller K., Nagarajan R., Narayanaswami R., Ni R., Nix K., Norrie T., Omernick M., Penukonda N., Phelps A., Ross J., Ross M., Salek A., Samadiani E., Severn C., Sizikov G., Snelham M., Souter J., Steinberg D., Swing A., Tan M., Thorson G., Tian B., Toma H., Tuttle E., Vasudevan V., Walter R., Wang W., Wilcox E., Yoon D. H.* In-datacenter performance analysis of a Tensor Processing Unit // *ACM SIGARCH Computer Architecture News*. 2017. Vol. 45, no. 2. P. 1–12. DOI: 10.1145/3140659.3080246.
 14. *Welsler J., Pitera J. W., Goldberg C.* Future computing hardware for AI // In: 2018 IEEE International Electron Devices Meeting (IEDM). 1-5 December 2018, San Francisco, CA, USA. New York: IEEE, 2018. P. 131–136. DOI: 10.1109/IEDM.2018.8614482.
 15. *Karras K., Pallis E., Mastorakis G., Nikoloudakis Y., Batalla J. M., Mavromoustakis C. X., Markakis E.* A hardware acceleration platform for AI-based inference at the edge // *Circuits, Systems, and Signal Processing*. 2020. Vol. 39, no. 2. P. 1059–1070. DOI: 10.1007/s00034-019-01226-7.
 16. *Kuptsov P. V., Kuptsova A. V., Stankevich N. V.* Artificial neural network as a universal model of nonlinear dynamical systems // *Russian Journal of Nonlinear Dynamics*. 2021. Vol. 17, no. 1. P. 5–21. DOI: 10.20537/nd210102.
 17. *Kuptsov P. V., Stankevich N. V., Bagautdinova E. R.* Discovering dynamical features of Hodgkin–Huxley-type model of physiological neuron using artificial neural network // *Chaos, Solitons & Fractals*. 2023. Vol. 167. P. 113027. DOI: 10.1016/j.chaos.2022.113027.

18. *Sherman A., Rinzel J., Keizer J.* Emergence of organized bursting in clusters of pancreatic beta-cells by channel sharing // *Biophysical Journal*. 1988. Vol. 54, no. 3. P. 411–425. DOI: 10.1016/S0006-3495(88)82975-8.
19. *Stankevich N., Mosekilde E.* Coexistence between silent and bursting states in a biophysical Hodgkin-Huxley-type of model // *Chaos*. 2017. Vol. 27, no. 12. P. 123101. DOI: 10.1063/1.4986401.
20. *Malashchenko T., Shilnikov A., Cymbalyuk G.* Six types of multistability in a neuronal model based on slow calcium current // *PLoS ONE*. 2011. Vol. 6, no. 7. P. e21782. DOI: 10.1371/journal.pone.0021782.
21. *Rozhnova M. A., Pankratova E. V., Stasenko S. V., Kazantsev V. B.* Bifurcation analysis of multistability and oscillation emergence in a model of brain extracellular matrix // *Chaos, Solitons & Fractals*. 2021. Vol. 151. P. 111253. DOI: 10.1016/j.chaos.2021.111253.
22. *Pankratova E. V., Sinitsina M. S., Gordleeva S., Kazantsev V. B.* Bistability and chaos emergence in spontaneous dynamics of astrocytic calcium concentration // *Mathematics*. 2022. Vol. 10, no. 8. P. 1337. DOI: 10.3390/math10081337.
23. *Press W. H., Teukolsky S. A., Vetterling W. T., Flannery B. P.* *Numerical Recipes: The Art of Scientific Computing*. 3rd Edition. New York: Cambridge University Press, 2007. 1256 p.
24. *Shilnikov A., Cymbalyuk G.* Transition between tonic spiking and bursting in a neuron model via the blue-sky catastrophe // *Phys. Rev. Lett.* 2005. Vol. 94, no. 4. P. 048101. DOI: 10.1103/PhysRevLett.94.048101.
25. *Marković D., Mizrahi A., Querlioz D., Grollier J.* Physics for neuromorphic computing // *Nature Reviews Physics*. 2020. Vol. 2. P. 499–510. DOI: 10.1038/s42254-020-0208-2.
26. *Stankevich N., Koseska A.* Cooperative maintenance of cellular identity in systems with intercellular communication defects // *Chaos*. 2020. Vol. 30, no. 1. P. 013144. DOI: 10.1063/1.5127107.
27. *Kingma D. P., Ba J.* Adam: A method for stochastic optimization // arXiv:1412.6980. arXiv Preprint, 2014. DOI: 10.48550/arXiv.1412.6980.

References

1. Levin E, Gewirtzman R, Inbar GF. Neural network architecture for adaptive system modeling and control. *Neural Networks*. 1991;4(2):185–191. DOI: 10.1016/0893-6080(91)90003-N.
2. Grieger B, Latif M. Reconstruction of the El Niño attractor with neural networks. *Climate Dynamics*. 1994;10(6–7):267–276. DOI: 10.1007/BF00228027.
3. Zimmermann HG, Neuneier R. Combining state space reconstruction and forecasting by neural networks. In: Bol G, Nakhaeizadeh G, Vollmer KH, editors. *Datamining und Computational Finance*. Vol. 174 of *Wirtschaftswissenschaftliche Beiträge*. Heidelberg: Physica; 2000. P. 259–267. DOI: 10.1007/978-3-642-57656-0_13.
4. Gilpin W, Huang Y, Forger DB. Learning dynamics from large biological data sets: Machine learning meets systems biology. *Current Opinion in Systems Biology*. 2020;22:1–7. DOI: 10.1016/j.coisb.2020.07.009.
5. Kolmogorov AN. On the representation of continuous functions of several variables by superpositions of continuous functions of a smaller number of variables. *Amer. Math. Soc. Transl.* 1961;17:369–373.
6. Arnold VI. On functions of three variables. *Amer. Math. Soc. Transl.* 1963;28:51–54.
7. Kolmogorov AN. On the representation of continuous functions of many variables by superposition of continuous functions of one variable and addition. *Amer. Math. Soc. Transl.* 1963;28:55–59.
8. Cybenko G. Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems*. 1989;2(4):303–314. DOI: 10.1007/BF02551274.
9. Gorban AN. Generalized approximation theorem and the exact representation of polynomials in

- several variables via the superpositions of polynomials in one variable. *Russian Mathematics*. 1998;42(5):4–7.
10. Haykin S. *Neural Networks: A Comprehensive Foundation*. Upper Saddle River, NJ: Prentice Hall; 1999. 842 p.
 11. Nikolenko S, Kadurin A, Arkhangelskaya E. *Deep Learning*. Saint Petersburg: Piter; 2018. 480 p. (in Russian).
 12. Cook S. *CUDA Programming: A Developer's Guide to Parallel Computing with GPUs*. Morgan Kaufmann; 2012. 592 p.
 13. Jouppi NP, Young C, Patil N, Patterson D, Agrawal G, Bajwa R, Bates S, Bhatia S, Boden N, Borchers A, Boyle R, Cantin PL, Chao C, Clark C, Coriell J, Daley M, Dau M, Dean J, Gelb B, Ghaemmaghami TV, Gottipati R, Gulland W, Hagmann R, Ho CR, Hogberg D, Hu J, Hundt R, Hurt D, Ibarz J, Jaffey A, Jaworski A, Kaplan A, Khaitan H, Killebrew D, Koch A, Kumar N, Lacy S, Laudon J, Law J, Le D, Leary C, Liu Z, Lucke K, Lundin A, MacKean G, Maggiore A, Mahony M, Miller K, Nagarajan R, Narayanaswami R, Ni R, Nix K, Norrie T, Omernick M, Penukonda N, Phelps A, Ross J, Ross M, Salek A, Samadiani E, Severn C, Sizikov G, Snellman M, Souter J, Steinberg D, Swing A, Tan M, Thorson G, Tian B, Toma H, Tuttle E, Vasudevan V, Walter R, Wang W, Wilcox E, Yoon DH. In-datacenter performance analysis of a Tensor Processing Unit. *ACM SIGARCH Computer Architecture News*. 2017;45(2):1–12. DOI: 10.1145/3140659.3080246.
 14. Welser J, Pitera JW, Goldberg C. Future computing hardware for AI. In: 2018 IEEE International Electron Devices Meeting (IEDM). 1-5 December 2018, San Francisco, CA, USA. New York: IEEE; 2018. P. 131–136. DOI: 10.1109/IEDM.2018.8614482.
 15. Karras K, Pallis E, Mastorakis G, Nikoloudakis Y, Batalla JM, Mavromoustakis CX, Markakis E. A hardware acceleration platform for AI-based inference at the edge. *Circuits, Systems, and Signal Processing*. 2020;39(2):1059–1070. DOI: 10.1007/s00034-019-01226-7.
 16. Kuptsov PV, Kuptsova AV, Stankevich NV. Artificial neural network as a universal model of nonlinear dynamical systems. *Russian Journal of Nonlinear Dynamics*. 2021;17(1):5–21. DOI: 10.20537/nd210102.
 17. Kuptsov PV, Stankevich NV, Bagautdinova ER. Discovering dynamical features of Hodgkin–Huxley-type model of physiological neuron using artificial neural network. *Chaos, Solitons & Fractals*. 2023;167:113027. DOI: 10.1016/j.chaos.2022.113027.
 18. Sherman A, Rinzel J, Keizer J. Emergence of organized bursting in clusters of pancreatic beta-cells by channel sharing. *Biophysical Journal*. 1988;54(3):411–425. DOI: 10.1016/S0006-3495(88)82975-8.
 19. Stankevich N, Mosekilde E. Coexistence between silent and bursting states in a biophysical Hodgkin-Huxley-type of model. *Chaos*. 2017;27(12):123101. DOI: 10.1063/1.4986401.
 20. Malashchenko T, Shilnikov A, Cymbalyuk G. Six types of multistability in a neuronal model based on slow calcium current. *PLoS ONE*. 2011;6(7):e21782. DOI: 10.1371/journal.pone.0021782.
 21. Rozhnova MA, Pankratova EV, Stasenko SV, Kazantsev VB. Bifurcation analysis of multistability and oscillation emergence in a model of brain extracellular matrix. *Chaos, Solitons & Fractals*. 2021;151:111253. DOI: 10.1016/j.chaos.2021.111253.
 22. Pankratova EV, Sinitsina MS, Gordleeva S, Kazantsev VB. Bistability and chaos emergence in spontaneous dynamics of astrocytic calcium concentration. *Mathematics*. 2022;10(8):1337. DOI: 10.3390/math10081337.
 23. Press WH, Teukolsky SA, Vetterling WT, Flannery BP. *Numerical Recipes: The Art of Scientific Computing*. 3rd Edition. New York: Cambridge University Press; 2007. 1256 p.
 24. Shilnikov A, Cymbalyuk G. Transition between tonic spiking and bursting in a neuron model via the blue-sky catastrophe. *Phys. Rev. Lett.* 2005;94(4):048101. DOI:10.1103/PhysRevLett.94.048101.

25. Marković D, Mizrahi A, Querlioz D, Grollier J. Physics for neuromorphic computing. *Nature Reviews Physics*. 2020;2:499–510. DOI: 10.1038/s42254-020-0208-2.
26. Stankevich N, Koseska A. Cooperative maintenance of cellular identity in systems with intercellular communication defects. *Chaos*. 2020;30(1):013144. DOI: 10.1063/1.5127107.
27. Kingma DP, Ba J. Adam: A method for stochastic optimization. arXiv:1412.6980. arXiv Preprint; 2014. DOI: 10.48550/arXiv.1412.6980.

Купцов Павел Владимирович — родился в Саратове (1972). Окончил Саратовский государственный университет (1994). Защитил диссертацию на соискание учёной степени кандидата физико-математических наук по специальности «Радиофизика» (СГУ, 1998, научный руководитель С. П. Кузнецов) и доктора физико-математических наук по специальности «Математическое моделирование, численные методы и комплексы программ» (СГТУ имени Гагарина Ю. А., 2013, научный консультант С. П. Кузнецов). В настоящее время работает главным научным сотрудником в Саратовском филиале Института радиотехники и электроники им. В. А. Котельникова РАН, руководит в этом институте лабораторией теоретической нелинейной динамики. По совместительству работает профессором в СГТУ имени Гагарина Ю. А., в СГУ, в Нижегородском кампусе НИУ ВШЭ. Область научных интересов — нелинейная динамика, численные методы, машинное обучение и наука о данных. Опубликовал свыше 100 научных статей. Член редакционной коллегии журнала «Russian Journal of Nonlinear Dynamics».



Россия, 410019 Саратов, ул. Зеленая, 38
 Саратовский филиал Института радиотехники
 и электроники им. В. А. Котельникова РАН
 E-mail: kupav@mail.ru
 ORCID: 0000-0003-2685-9828
 AuthorID (eLibrary.Ru): 29426

Станкевич Наталья Владимировна — родилась в Саратове (1985), окончила Саратовский государственный университет имени Н. Г. Чернышевского (2007). Защитила диссертацию на соискание ученой степени кандидата физико-математических наук по специальности «Радиофизика» (СГУ, 2011, научный руководитель А. П. Кузнецов). С 2011 по 2019 год работала доцентом и профессором Саратовского государственного технического университета имени Гагарина Ю. А. на кафедре «Радиоэлектроника и телекоммуникации». В настоящее время работает старшим научным сотрудником в лаборатории топологических методов в динамике НИУ «Высшая школа экономики» — Нижний Новгород. Опубликовала 50 научных статей в области теории колебаний и волн, радиофизика и электроника, применение нелинейной динамики в различных областях науки. Член редакционной коллегии журналов «Universal Journal of Electrical and Electronic Engineering» и «Дифференциальные уравнения и процессы управления».



Россия, 603055 Нижний Новгород, ул. Большая Печерская, 25/12
 Национальный исследовательский университет Высшая школа экономики
 Россия, 410019 Саратов, Зелёная, 38
 Саратовский филиал Института радиотехники
 и электроники им. В. А. Котельникова РАН
 E-mail: stankevichnv@mail.ru
 ORCID: 0000-0002-4781-0567
 AuthorID (eLibrary.Ru): 600847