

## Кодирование и устойчивость обработки сигналов в потоковых рекуррентных нейронных сетях

В. Ю. Осипов<sup>а</sup>, доктор техн. наук, профессор, [orcid.org/0000-0001-5905-4415](https://orcid.org/0000-0001-5905-4415), [osipov\\_vasilyi@mail.ru](mailto:osipov_vasilyi@mail.ru)

В. В. Никифоров<sup>а</sup>, доктор техн. наук, профессор, [orcid.org/0000-0002-6836-6303](https://orcid.org/0000-0002-6836-6303)

<sup>а</sup>Санкт-Петербургский институт информатики и автоматизации РАН, 14-я линия В. О., 39, Санкт-Петербург, 199178, РФ

**Введение:** при обосновании перспективных архитектур потоковых рекуррентных нейронных сетей возникает необходимость оценивания их устойчивости обработки различных входных сигналов. Для этого строятся диаграммы устойчивости, содержащие результаты имитационного моделирования для каждого из узлов этих диаграмм. На такое оценивание могут затрачиваться значительные временные и вычислительные ресурсы, особенно при анализе больших нейронных сетей. **Цель:** поиск методов оперативного построения таких диаграмм и оценивания устойчивости работы потоковых рекуррентных нейронных сетей. **Результаты:** анализ особенностей исследуемых диаграмм устойчивости показал, что узлы диаграмм группируются в сплошные зоны с одинаковыми характеристиками соотношения дефектов обработки входных сигналов. Предложен метод построения этих диаграмм на основе принципа обхода границ их зон. Таким путем можно избежать выполнения имитационного моделирования для внутренних узлов каждой зоны. Моделирование выполняется только для узлов, примыкающих к границам зон. Число узлов, для которых необходимо выполнять сеансы имитационного моделирования, снижается на порядок. Исследовано влияние видов кодирования входных сигналов на устойчивость потоковых рекуррентных нейронных сетей. Показано, что представление входных сигналов в виде последовательностей единичных образов с пересекающимися элементами может обеспечить большую устойчивость, чем без их пересечения.

**Ключевые слова** – рекуррентная нейронная сеть, устойчивость, моделирование, кодирование и обработка сигналов.

Для цитирования: Осипов В. Ю., Никифоров В. В. Кодирование и устойчивость обработки сигналов в потоковых рекуррентных нейронных сетях. *Информационно-управляющие системы*, 2021, № 3, с. 9–18. doi:10.31799/1684-8853-2021-3-9-18

For citation: Osipov V. Yu., Nikiforov V. V. Coding and robustness of signal processing in streaming recurrent neural networks. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2021, no. 3, pp. 9–18 (In Russian). doi:10.31799/1684-8853-2021-3-9-18

### Введение

Потоковые рекуррентные нейронные сети (РНС) ориентированы на непрерывную ассоциативную обработку различных сигналов. В качестве таких сигналов могут выступать предварительно преобразованные к требуемому виду видео- и звуковые потоки, информация от датчиков скорости, температуры, давления, расхода энергии и др. Эти РНС применимы для распознавания и фильтрации динамических образов, прогнозирования и восстановления прошедших событий, управления различными процессами. Такие сети могут наделяться различными пространственно-временными структурами. Они способны самообучаться, формировать модели воспринимаемых сигналов и использовать их для выработки творческих решений. В них реализованы возможности управления ассоциативным запоминанием и вызовом сигналов из памяти, а также переключения от решения одних задач к другим [1–5].

Несмотря на достигнутые результаты в исследовании потоковых РНС, их потенциал во многом не изучен, не снят ряд проблемных вопросов, свя-

занных с кодированием входных и внутренних сигналов, а также с обеспечением устойчивости функционирования этих нейронных сетей.

В общем случае перед подачей на вход потоковых РНС сигналов они должны быть преобразованы к входному языку сети. Используется принцип аналогии с биологическими системами. Примерами таких аналогий выступают преобразования оптических сигналов сетчаткой глаза, звуковых волн ворсинками слухового аппарата и др. В искусственных РНС, как и в естественных нейронных сетях, реализуется также внутрисетевое кодирование сигналов.

От реализуемых видов внешнего и внутрисетевого кодирования сигналов в потоковых РНС во многом зависит их устойчивость, а также возможности пространственно-временного связывания сигналов на элементах сети, запоминания и извлечения из нейросетевой памяти.

Известны некоторые принципы кодирования информации в биологических системах [6–9]. По аналогии с биологическими принципами предложены варианты предварительного кодирования сигналов перед подачей их в искусственные нейронные сети [1, 10–14].

В общем случае «сигналы раскладывают на пространственно-частотные составляющие. Каждую из этих составляющих преобразуют в последовательность единичных образов. Частоту и фазу следования этих импульсов задают как функции от амплитуды и фазы соответствующей пространственно-частотной составляющей. При таком преобразовании сигналы в дискретном времени представляются в виде последовательных совокупностей единичных импульсов (образов)» [4].

Реализуются и другие варианты [15–18]. Во всех случаях возможно задание различных параметров предварительного кодирования сигналов. Остается открытым вопрос: какой из вариантов предварительного кодирования сигналов является наилучшим?

Кроме этого, при обработке сигналов в потоковых РНС большое влияние на ее результаты оказывают также значения порогов возбуждения нейронов и уровень ассоциативного взаимодействия [19]. При несоблюдении ряда условий потоковые, как и другие РНС, могут перевозбуждаться или быть слишком заторможенными. Желательно за счет обоснования параметров потоковых РНС уже на этапе их проектирования обеспечивать последующую устойчивую работу при непрерывной обработке различных сигналов.

Для этого могут применяться известные методы анализа устойчивости РНС [1, 4, 19–21] при варьировании различными параметрами входных сигналов и самой сети. Среди этих методов применительно к потоковым РНС наибольший интерес представляет метод [4], позволяющий находить границы параметров устойчивого функционирования нейронных сетей. Однако в исходном варианте этот метод характеризуется большой вычислительной сложностью, что затрудняет применение его на практике. Необходим поиск новых решений, позволяющих снизить вычислительную сложность анализа устойчивости потоковых РНС и определиться с целесообразными подходами к кодированию обрабатываемых в них сигналов.

### Особенности потоковых РНС и постановка задачи

Основу исследуемых потоковых РНС составляет двухслойная нейронная сеть с блоком управления [1]. Принципиальные отличия потоковых РНС от традиционных нейросетевых решений состоят в том, что первые наделяются прозрачными логическими структурами за счет реализации пространственных сдвигов сигналов (в виде последовательных совокупностей единичных образов (СЕО)) при передаче от слоя к слою [1]. За счет

этих сдвигов РНС могут быть наделены линейными, спиральными, петлевыми и другими логическими структурами [1–4]. В этих РНС используются импульсные нейроны с тремя состояниями: ожидание, возбуждение и невосприимчивость. Время невосприимчивости каждого нейрона после возбуждения больше, чем время задержки импульсов в двухслойных контурах сети [5]. Для каждого  $i$ -го нейрона на выходе  $y_i(t + 1)$  формируется единичный образ по правилу

$$y_i(t + 1) = \begin{cases} 1, & \text{если } \sum_{j=1}^J w_{ij}(t) \cdot x_j(t) \geq Thr \text{ и } \Delta t_i > \Delta T; \\ 0 & \text{в других случаях,} \end{cases} \quad (1)$$

где  $J$  — число нейронов, связанных с  $i$ -м нейроном;  $w_{ij}(t)$  — вес  $ij$ -го синапса, связывающего  $i$ -й нейрон с  $j$ -м нейроном;  $x_j(t)$  — значение потенциала на выходе  $j$ -го нейрона на момент времени  $t$ ;  $Thr$  — порог возбуждения нейронов, находящихся в режиме ожидания;  $\Delta t_i$  — время, прошедшее после предыдущего возбуждения  $i$ -го нейрона;  $\Delta T$  — время невосприимчивости нейрона после возбуждения.

Веса  $w_{ij}(t)$  синапсов в такой РНС определяют через произведения весовых коэффициентов  $k_{ij}(t)$ , функций ослабления  $\beta_{ij}(t)$ ,  $\eta_{ij}(t)$  расходящихся и сходящихся единичных образов [4, 5]:

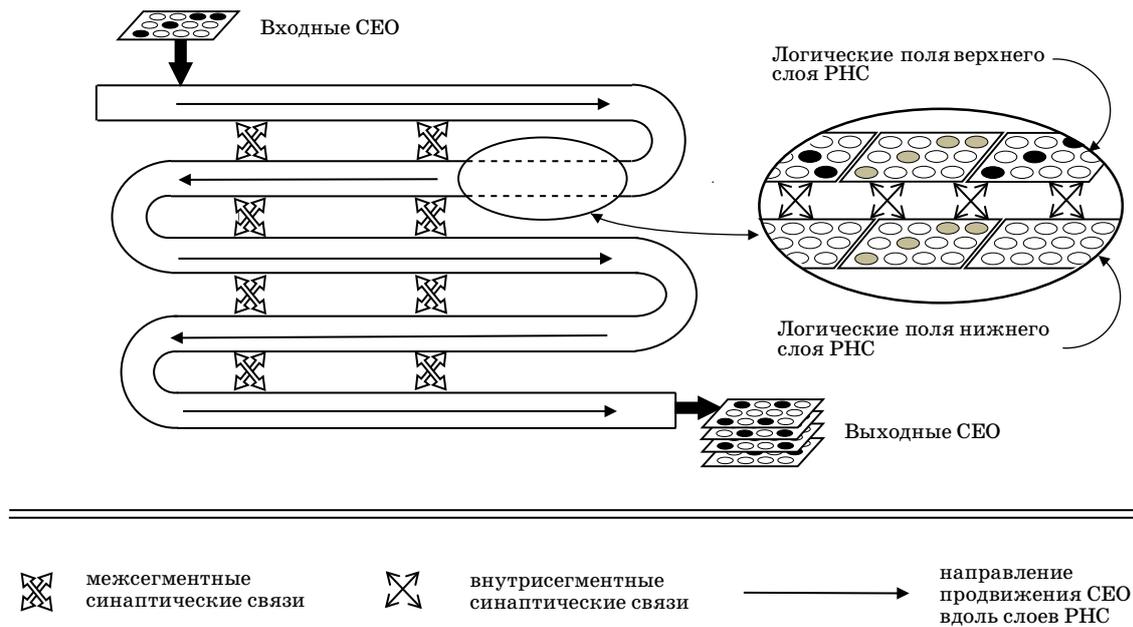
$$w_{ij}(t) = k_{ij}(t) \cdot \beta_{ij}(t) \cdot \eta_{ij}(t). \quad (2)$$

Весовые коэффициенты являются функциями прошедших через синапсы единичных импульсов [1, 18]. Функции ослабления зависят от параметров расходимости и сходимости этих импульсов, а также удаленности взаимодействующих нейронов с учетом принятого масштаба  $\alpha$ . Изменяя  $\beta_{ij}(t)$  или  $\eta_{ij}(t)$  в зависимости от текущих состояний слоев, можно управлять ассоциативным взаимодействием обрабатываемых в РНС сигналов. Управлять также можно порогами возбуждения нейронов.

Заметим, что за счет приоритетности коротких связей между нейронами в потоковых РНС может обеспечиваться однозначное соответствие между входным и выходным пространством сети [1].

Пример спиральной структуры потоковой РНС, ориентированной на устранение искажений в последовательности входных СЕО, показан на рис. 1.

Внутренняя организация каждого из сегментов соответствует схеме в правой части рис. 1. Сегмент содержит два слоя нейронов — верхний и примыкающий к нему нижний слой. Каждый слой строится в виде последовательности прямоугольных полей нейронов. Отдельное поле имеет



■ **Рис. 1.** Пример структуры спиральной РНС  
 ■ **Fig. 1.** An example of the structure of a spiral RNN

формат, идентичный формату кодов входного потока. В примере рис. 1 каждое поле содержит 12 нейронов, изображенных кружками (черный — нейрон возбужден, светлый — нейрон в состоянии ожидания, серый — нейрон в состоянии невосприимчивости). Конкретное поле верхнего слоя с непосредственно примыкающим к нему полем нижнего слоя составляет пару противоположащих полей; каждый сегмент представляет собой последовательность таких пар. Нейроны верхнего и нижнего слоя связываются внутрисегментными синапсами, обеспечивающими потактовое движение введенных СЕО от входного поля к выходу сети. Такое движение достигается должной настройкой параметров внутрисегментных синапсов.

При функционировании РНС могут наблюдаться дефекты типа 0 (пропуск единицы) и типа 1 (ложная единица).

Для конкретных условий применения потоковых РНС необходимо определить набор параметров сети, который обеспечивал бы требуемую безошибочность функционирования сети.

В работе [4] предложен метод, «позволяющий определять число  $Q(l)$  успешных вариантов восстановления искаженной последовательности СЕО при  $l$  циклах обучения:

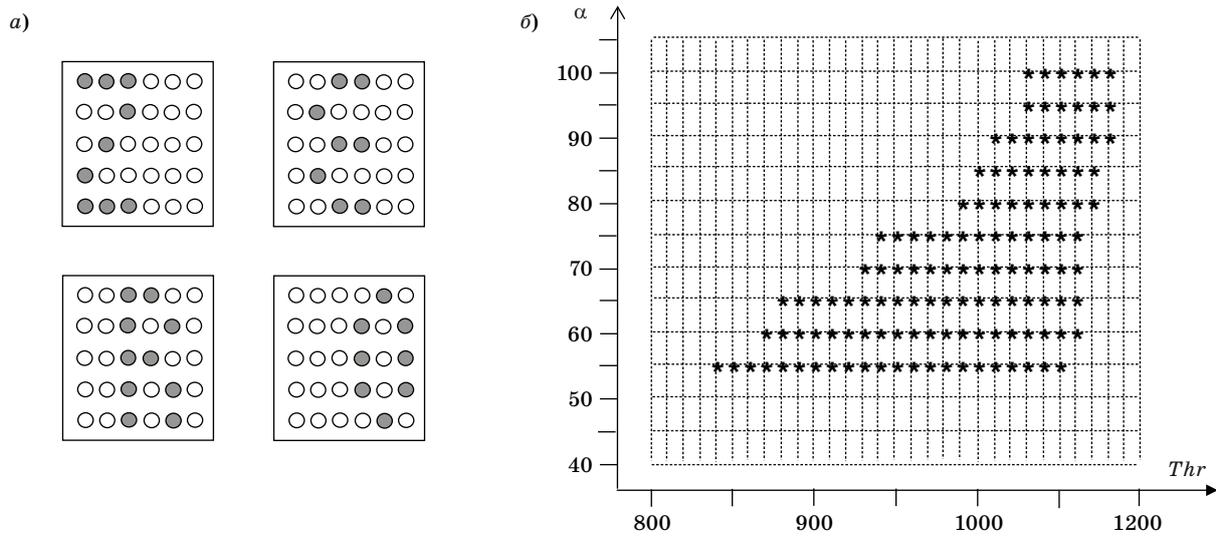
$$Q(l) = \sum_{z=1}^Z \sum_{k=1}^K \delta_{zk}(Thr_z, \alpha_k, l),$$

где  $\delta_{zk}(Thr_z, \alpha_k, l)$  — булева функция успешного восстановления последовательности искажен-

ных СЕО, своеобразных кадров;  $\alpha_k$  — величина масштаба расстояний между нейронами;  $Z$  — число градаций порога возбуждения нейронов;  $K$  — число градаций масштаба расстояний между нейронами. Заметим, что  $\delta_{zk}(Thr_z, \alpha_k, l)$  определяется по результатам испытаний на конкретной модели РНС. В результате таких испытаний для каждого значения  $l$  формируется своя диаграмма  $\delta_{zk}(Thr_z, \alpha_k, l)$  [4]. Эту диаграмму можно называть диаграммой относительной устойчивости РНС. Вариант такой диаграммы с набором СЕО для составления входной последовательности показан на рис. 2, а и б. Успешные восстановления последовательностей СЕО обозначены на диаграмме в виде символов \*. Число таких восстановлений зависит от масштаба диаграммы и заданных дискретностей  $\Delta\alpha, \Delta Thr$  параметров  $\alpha, Thr$ .

В работе [4] для оценки величины  $Q(l)$  реализуются сеансы моделирования для каждого из узлов диаграммы. Для построения этой диаграммы число сеансов моделирования исчисляется в сотнях единиц. Если учесть, что пространство комбинаций значений параметров настройки сети существенно многомерно, то для поиска комбинации, подходящей для конкретного набора входных СЕО, может потребоваться построение десятков и сотен тысяч рассматриваемых диаграмм. Это миллион сеансов моделирования.

Требуется разработать метод ускоренного построения таких диаграмм и анализа устойчивости потоковых РНС.



■ **Рис. 2.** Пример построения диаграммы  $\delta_{zk}(Thr_z, \alpha_k, l)$ : *a* — набор СЕО для составления входной последовательности; *b* — диаграмма

■ **Fig. 2.** An example of constructing a diagram  $\delta_{zk}(Thr_z, \alpha_k, l)$ : *a* — the collection of SSP for compiling an input sequence; *b* — the diagram

**Метод построения диаграмм устойчивости потоковых РНС**

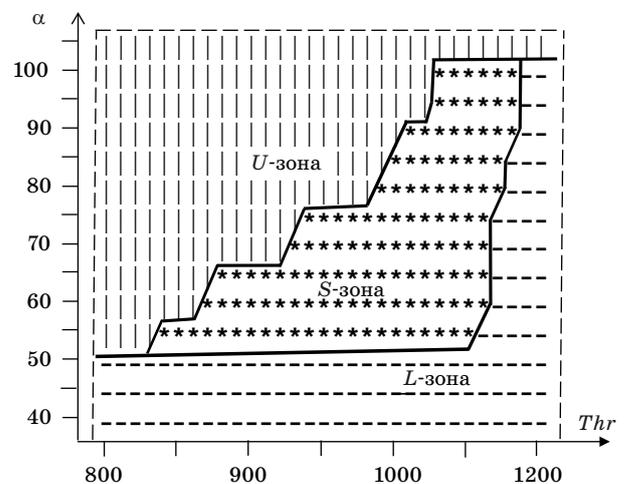
Для пояснения предлагаемого метода уточним особенности диаграммы  $\delta_{zk}(Thr_z, \alpha_k, l)$ , которая при дискретных значениях  $\alpha, Thr$  может рассматриваться как решетка с конкретными узлами. При обработке заданной входной последовательности СЕО рекуррентной нейронной сетью с определенной структурой с различными вариантами значений  $\alpha, Thr$  множество узлов, получаемых на диаграмме, можно разбить на три зоны. К ним относятся: *S*-зона, отражающая отсутствие возможных дефектов; *L*-зона, в которой число дефектов типа «0» больше числа дефектов типа 1; *U*-зона с числом дефектов типа 1, равным числу дефектов типа 0 либо превышающим его. Пример такой диаграммы приведен на рис. 3.

При моделировании потоковых РНС установлено, что если узел  $(\alpha, Thr)$  принадлежит *U*-зоне, то при  $\alpha' > \alpha$  узел  $(\alpha', Thr)$  также входит в эту зону. Если узел  $(\alpha, Thr)$  принадлежит *L*-зоне, то при  $\alpha' < \alpha$  узел  $(\alpha', Thr)$  относится к *L*-зоне. С учетом этого обстоятельства предлагается метод ускоренного оценивания значения относительной устойчивости РНС. Метод основан на построении нижней границы *U*-зоны и верхней границы *L*-зоны.

Построение нижней границы *U*-зоны предлагается выполнять в два этапа.

**Этап 1.** Поиск крайней правой верхней точки  $(\alpha^*, Thr^*)$ , принадлежащей нижней границе *U*-зоны.

Если *U*-зона не пустая, то в силу принятых правил правый верхний узел находится либо на горизонтали от  $(\alpha_{high}, Thr_{left})$  до  $(\alpha_{high}, Thr_{right})$ , либо на вертикали от  $(\alpha_{low}, Thr_{right})$  до  $(\alpha_{high}, Thr_{right})$ . Этот этап требует не более чем  $\max\{\alpha_{high}, Thr_{right}\}$  шагов моделирования. На каждом шаге этапа 1 моделирование выполняется для нового узла  $(\alpha_{cur}, Thr_{cur})$ . Устанавливаются значения  $\alpha_{cur} = \alpha_{high}, Thr_{cur} = Thr_{right}$ . Выполняется сеанс моделирования для узла  $(\alpha_{high}, Thr_{right})$ . Если этот узел принадлежит *U*-зоне, то искомый узел  $(\alpha^*, Thr^*)$  находится на вертикали от  $(\alpha_{low}, Thr_{right})$  до  $(\alpha_{high}, Thr_{right})$ . Иначе  $(\alpha^*, Thr^*)$  находится на горизонтали от  $(\alpha_{high}, Thr_{left})$  до  $(\alpha_{high}, Thr_{right-1})$ .



■ **Рис. 3.** Зоны *U, S* и *L* диаграммы

■ **Fig. 3.** Zones *U, S* and *L* of the diagram

При поиске по горизонтали в начале каждого очередного шага значение  $Thr_{cur}$  уменьшается на величину  $\Delta Thr$  шага решетки по координате  $Thr$ ,  $Thr_{cur} = Thr_{cur} - \Delta Thr$ . Если новое значение  $Thr_{cur}$  меньше, чем  $Thr_{left}$ , то делается вывод, что в диаграмме  $U$ -зона отсутствует. Отсутствует и ее нижняя граница, а также искомый узел ( $\alpha^*$ ,  $Thr^*$ ). В этом случае этап 1 завершен, переход к этапу 2 не требуется.

При  $Thr_{cur} \geq Thr_{left}$  выполняется сеанс моделирования с новым значением  $Thr_{cur}$ . Если  $(\alpha_{high}, Thr_{cur})$  принадлежит  $U$ -зоне, то  $Thr^* = Thr_{cur}$ ,  $\alpha^* = \alpha_{high}$  и полагается, что правый верхний узел ее границы найден. Это означает, что этап 1 завершен, выполняется переход к этапу 2. В противном случае, если узел  $(\alpha_{high}, Thr_{cur})$  не принадлежит  $U$ -зоне, выполняется переход к следующему шагу этапа 1.

Поиск по вертикали выполняется аналогично. На каждом шаге  $Thr_{cur}$  остается равным  $Thr_{right}$ , значение  $\alpha_{cur}$  уменьшается на  $\Delta\alpha$ . При  $\alpha_{cur} < \alpha_{low}$  делается вывод, что  $U$ -зона занимает всю диаграмму. Переход к этапу 2 не требуется. Если  $(\alpha_{cur}, Thr_{right})$  принадлежит  $U$ -зоне, то правый верхний узел ее границы найден,  $Thr^* = Thr_{right}$ ,  $\alpha^* = \alpha_{cur} - \Delta\alpha$ , выполняется переход к этапу 2.

**Этап 2.** Поиск остальных узлов нижней границы  $U$ -зоны.

К первому шагу этапа 2 значения  $\alpha_{cur}, Thr_{cur}$  представляют известные координаты  $\alpha^*, Thr^*$  крайнего правого верхнего узла ( $\alpha^*, Thr^*$ ) нижней границы  $U$ -зоны,  $\alpha_{cur} = \alpha^*, Thr_{cur} = Thr^*$ . К каждому очередному шагу значения  $\alpha_{cur}, Thr_{cur}$  — это найденные координаты следующего узла этой границы. Каждый очередной шаг этапа 2 обеспечивает определение координат следующего узла нижней границы  $U$ -зоны. К началу очередного шага уже известно, что узел  $(\alpha_{cur}, Thr_{cur})$  является узлом границы  $U$ -зоны. Если  $\alpha_{cur} \leq \alpha_{low}$  или  $Thr_{cur} < Thr_{left}$ , то построение границы  $U$ -зоны завершено.

Для определения координат следующего узла границы  $U$ -зоны в общем случае возможен один из трех вариантов:  $(\alpha_{cur}, Thr_{cur} - \Delta Thr)$ ,  $(\alpha_{cur} - \Delta\alpha, Thr_{cur} - \Delta Thr)$ ,  $(\alpha_{cur} - \Delta\alpha, Thr_{cur})$ . Если два предыдущих узла искомой границы лежат на одной горизонтали, то узел  $(\alpha_{cur} - \Delta\alpha, Thr_{cur})$  не может принадлежать границе  $U$ -зоны. В этом случае выполняется сеанс моделирования для узла  $(\alpha_{cur} - \Delta\alpha, Thr_{cur} - \Delta Thr)$ . Если этот узел принадлежит  $U$ -зоне, то это очередной узел границы этой зоны. Координатам  $\alpha_{cur}, Thr_{cur}$  присваиваются значения  $\alpha_{cur} - \Delta\alpha, Thr_{cur} - \Delta Thr$ , выполняется переход к следующему шагу этапа 2.

При условии, что два предыдущих узла искомой границы не лежат на одной горизонтали, выполняется сеанс моделирования для узла  $(\alpha_{cur} - \Delta\alpha, Thr_{cur})$ . Если этот узел принадлежит

$U$ -зоне, то он принимается как очередной узел границы этой зоны. Координате  $\alpha_{cur}$  присваивается значение  $\alpha_{cur} - \Delta\alpha$ , выполняется переход к следующему шагу этапа 2. В противном случае при  $Thr_{cur} = Thr_{left}$  построение границы  $U$ -зоны завершено. При  $Thr_{cur} > Thr_{left}$  выполняется моделирование для узла  $(\alpha_{cur} - \Delta\alpha, Thr_{cur} - \Delta Thr)$ . Если этот узел принадлежит  $U$ -зоне, то он принимается очередным узлом границы этой зоны и координатам  $\alpha_{cur}, Thr_{cur}$  присваиваются, соответственно, значения  $(\alpha_{cur} - \Delta\alpha, Thr_{cur} - \Delta Thr)$ . В противном случае очередным узлом  $U$ -зоны принимается узел  $(\alpha_{cur}, Thr_{cur} - \Delta Thr)$ , координатам  $\alpha_{cur}, Thr_{cur}$  присваиваются, соответственно, значения  $\alpha_{cur}, Thr_{cur} - \Delta Thr$ . В обоих случаях выполняется переход к следующему шагу этапа 2.

Аналогичным образом строится верхняя граница  $L$ -зоны. На этапе 1 поиск крайнего левого нижнего узла ( $\alpha^*, Thr^*$ ) этой зоны начинается с узла  $(\alpha_{low}, Thr_{left})$ . Если  $L$ -зона не пустая, то ее крайний левый нижний узел находится либо на горизонтали от  $(\alpha_{low}, Thr_{left})$  до  $(\alpha_{low}, Thr_{right})$ , либо на вертикали от  $(\alpha_{low}, Thr_{left})$  до  $(\alpha_{high}, Thr_{left})$ . Если  $(\alpha_{high}, Thr_{left})$  принадлежит  $L$ -зоне, делается вывод, что эта зона занимает всю диаграмму и переход к этапу 2 не требуется. На этапе 2 последовательность узлов, принадлежащих границе  $L$ -зоны, строится снизу вверх (в противоположность границе  $U$ -зоны, которая строится сверху вниз).

Когда границы  $U$ -зоны и  $L$ -зоны построены, значение относительной устойчивости РНС определяется как число узлов диаграммы, лежащих между этими границами.

Предложенный метод вместо  $N_\alpha \cdot N_{Thr}$  требует не более чем  $2(N_\alpha + N_{Thr})$  выполнений сеанса моделирования, где  $N_\alpha$  и  $N_{Thr}$  — соответственно число строк и столбцов формируемой диаграммы относительной устойчивости РНС.

## Исходные данные и результаты моделирования

С использованием предложенного метода осуществлялось оценивание относительной устойчивости для двухслойной РНС с числом нейронов в каждом слое 360 единиц. Каждый слой разбивался на  $6 \times 2 = 12$  логических полей размером  $6 \times 5$  нейронов. Продвижение обрабатываемых совокупностей единичных образов вдоль слоев выполнялось по спирали. В отличие от рис. 1 в РНС формировалось не пять, а два сегмента. При определении весов синапсов (2) весовые коэффициенты  $k_{ij}(t)$  определялись как

$$k_{ij}(t) = b \cdot \text{th}(g_{ij}(t)), g_{ij}(t) = g_{ij}(t-1) \pm \Delta g_{ij}(t-1),$$

где  $\text{th}(\cdot)$  — гиперболический тангенс;  $g_{ij}(t)$  — параметр, зависящий от числа прошедших через  $ij$ -й синапс единичных импульсов (образов) на момент времени  $t$ ;  $b$  — положительный коэффициент.

Функция ослабления  $\beta_{ij}(t)$  расходящихся единичных образов в (2) задавалась в виде  $\beta_{ij}(t) = 1 - r_{ij}(t)/(\alpha + r_{ij}(t))$ , где  $r_{ij}(t)$  — расстояние между  $i$ -м и  $j$ -м нейроном;  $\alpha$  — масштаб расстояний между нейронами, являющийся одной из координат рассматриваемых диаграмм. Значение функции  $\eta_{ij}(t)$  в (2) считалось равным единице.

В РНС подавались наборы непересекающихся входных кодов, которые приведены в трех следующих вариантах:

$$\begin{aligned} \mathbf{A}_1 &= \begin{pmatrix} 110000 \\ 110000 \\ 000000 \\ 000000 \\ 000000 \end{pmatrix}, \quad \mathbf{A}_2 = \begin{pmatrix} 000011 \\ 000011 \\ 000000 \\ 000000 \\ 000000 \end{pmatrix}, \\ \mathbf{A}_3 &= \begin{pmatrix} 000000 \\ 000000 \\ 000000 \\ 110000 \\ 110000 \end{pmatrix}, \quad \mathbf{A}_4 = \begin{pmatrix} 000000 \\ 000000 \\ 000000 \\ 000011 \\ 000011 \end{pmatrix}; \end{aligned} \quad (3)$$

$$\begin{aligned} \mathbf{B}_1 &= \begin{pmatrix} 101000 \\ 000000 \\ 101000 \\ 000000 \\ 000000 \end{pmatrix}, \quad \mathbf{B}_2 = \begin{pmatrix} 000000 \\ 010100 \\ 000000 \\ 010100 \\ 000000 \end{pmatrix}, \\ \mathbf{B}_3 &= \begin{pmatrix} 000000 \\ 001010 \\ 000000 \\ 001010 \\ 000000 \end{pmatrix}, \quad \mathbf{B}_4 = \begin{pmatrix} 000000 \\ 000000 \\ 000101 \\ 000000 \\ 000101 \end{pmatrix}; \end{aligned} \quad (4)$$

$$\begin{aligned} \mathbf{C}_1 &= \begin{pmatrix} 100100 \\ 000000 \\ 000000 \\ 100100 \\ 000000 \end{pmatrix}, \quad \mathbf{C}_2 = \begin{pmatrix} 000000 \\ 010010 \\ 000000 \\ 000000 \\ 010010 \end{pmatrix}, \\ \mathbf{C}_3 &= \begin{pmatrix} 000000 \\ 001001 \\ 000000 \\ 000000 \\ 001001 \end{pmatrix}, \quad \mathbf{C}_4 = \begin{pmatrix} 010010 \\ 000000 \\ 000000 \\ 010010 \\ 000000 \end{pmatrix}. \end{aligned} \quad (5)$$

Вариант (3) относится к плотным кодам, (4) — к раздвинутым кодам, (5) — к широким кодам. В (3)–(5) нули соответствуют отсутствию единичных образов. Нежирными единицами отмечены единичные образы, которые перед вводом в сеть искажаются на нули. Жирными единицами обозначены активные единичные образы.

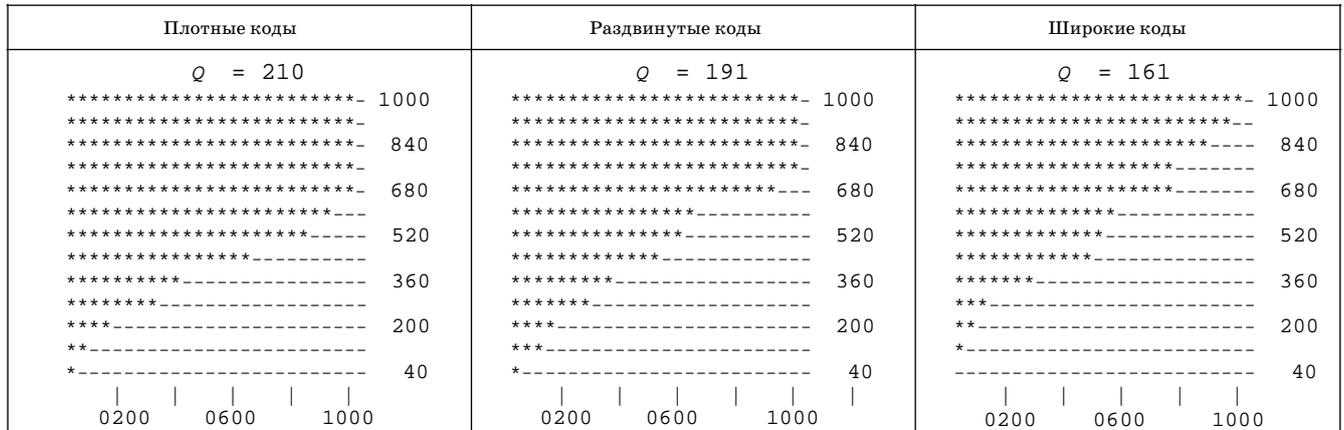
Согласно (3)–(5) выполняются условия:  $\mathbf{A}_i \circ \mathbf{A}_j = \mathbf{Z}$ ,  $\mathbf{B}_i \circ \mathbf{B}_j = \mathbf{Z}$ ,  $\mathbf{C}_i \circ \mathbf{C}_j = \mathbf{Z}$ , где  $\circ$  — поэлементное умножение матриц при  $i, j$ , не равном  $j$ ;  $\mathbf{Z}$  — нулевая матрица. Заметим, что в примерах (3)–(5) в каждом из входных кодов имеется 13,3 % активных элементов, из которых искажается половина.

Результаты построения искомых диаграмм для вариантов входных кодов (3)–(5) приведены в табл. 1, из которой следует, что для более плотных кодов значения относительной устойчивости  $Q$  работы потоковой РНС наибольшие. Эта тенденция может объясняться тем, что в режиме обучения при обработке более плотных кодов формируются более насыщенные синаптические связи. Отметим, что в диаграммах (см. табл. 1) нет  $U$ -зон. Подобные диаграммы могут возникать в случае отсутствия входных пересечений входных кодов. Для построения подобных диаграмм, в которых заведомо отсутствуют  $U$ -зоны, допустимо использовать упрощенный алгоритм ускоренной оценки значений относительной устойчивости. Достаточно ограничиться построением верхней границы  $L$ -зоны. В общем случае для выполнения ускоренного алгоритма построения диаграммы требуется число сеансов моделирования равно числу узлов, лежащих на ее периметре. При упрощенном варианте этого алгоритма число сеансов моделирования уменьшается вдвое.

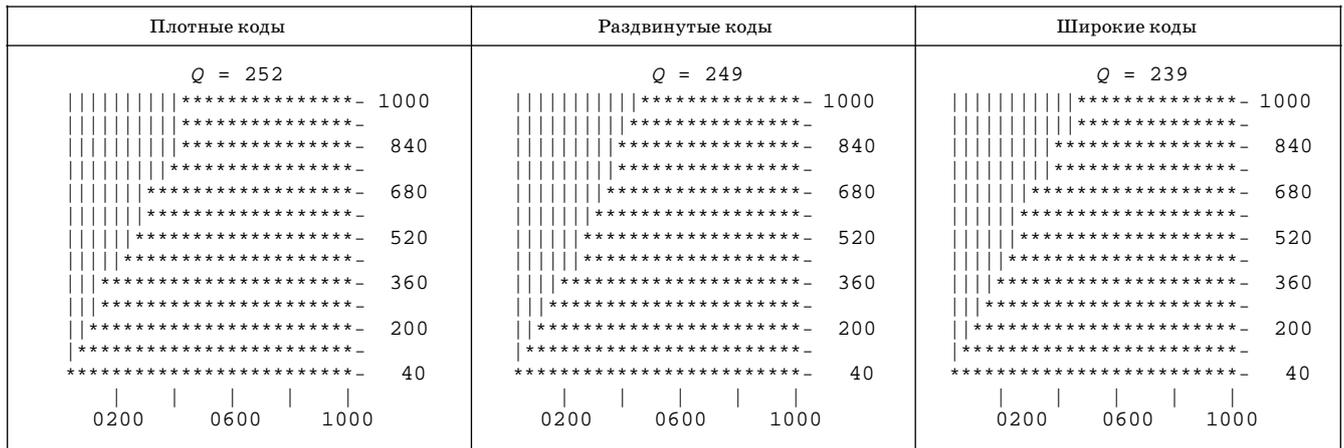
Выражения

$$\begin{aligned} \mathbf{A}_1^* &= \begin{pmatrix} 110000 \\ 110000 \\ 000000 \\ 000000 \\ 000000 \end{pmatrix}, \quad \mathbf{A}_2^* = \begin{pmatrix} 000000 \\ 011000 \\ 011000 \\ 000000 \\ 000000 \end{pmatrix}, \\ \mathbf{A}_3^* &= \begin{pmatrix} 000000 \\ 000000 \\ 001100 \\ 001100 \\ 000000 \end{pmatrix}, \quad \mathbf{A}_4^* = \begin{pmatrix} 000000 \\ 000000 \\ 000000 \\ 000110 \\ 000110 \end{pmatrix}; \quad (6) \\ \mathbf{B}_1^* &= \begin{pmatrix} 001000 \\ 100000 \\ 001000 \\ 100000 \\ 000000 \end{pmatrix}, \quad \mathbf{B}_2^* = \begin{pmatrix} 000000 \\ 000010 \\ 001000 \\ 000010 \\ 001000 \end{pmatrix}, \end{aligned}$$

■ Таблица 1. Диаграммы относительной устойчивости РНС для вариантов (3)–(5) кодирования входных сигналов  
 ■ Table 1. Diagrams of the relative RNN stability for (3)–(5) coding variants of input signals



■ Таблица 2. Диаграммы относительной устойчивости РНС для вариантов (6)–(8)  
 ■ Table 2. Diagrams of the relative RNN stability for (6)–(8) coding variants



$$B_3^* = \begin{pmatrix} 000000 \\ 000000 \\ 000101 \\ 000000 \\ 001010 \end{pmatrix}, B_4^* = \begin{pmatrix} 000000 \\ 000100 \\ 000001 \\ 000100 \\ 000001 \end{pmatrix}; \quad (7)$$

$$C_1^* = \begin{pmatrix} 000100 \\ 100000 \\ 000000 \\ 000100 \\ 100000 \end{pmatrix}, C_2^* = \begin{pmatrix} 000000 \\ 100100 \\ 000000 \\ 000000 \\ 010010 \end{pmatrix},$$

$$C_3^* = \begin{pmatrix} 010000 \\ 000010 \\ 000000 \\ 010000 \\ 000010 \end{pmatrix}, C_4^* = \begin{pmatrix} 000000 \\ 010010 \\ 000000 \\ 000000 \\ 001001 \end{pmatrix} \quad (8)$$

соответствуют другим трем вариантам кодов, но уже с пересекающимися единичными образами:  $A_1^* \circ A_2^* \neq Z$ ,  $A_2^* \circ A_3^* \neq Z$ ,  $A_3^* \circ A_4^* \neq Z$ ,  $B_1^* \circ B_2^* \neq Z$ ,  $B_2^* \circ B_3^* \neq Z$ ,  $B_3^* \circ B_4^* \neq Z$ ,  $C_1^* \circ C_2^* \neq Z$ ,  $C_2^* \circ C_3^* \neq Z$ ,  $C_3^* \circ C_4^* \neq Z$ .

В диаграммах табл. 2, в отличие от табл. 1, присутствуют  $U$ -зоны. Однако  $L$ -зона практически отсутствует. Такое обстоятельство может объясняться тем, что в режиме обучения в ходе обработки пересекающихся кодов для небольших значений  $\alpha$  формируются сравнительно более насыщенные синаптические связи. В результате в левом верхнем углу диаграммы появляется  $U$ -зона, а  $L$ -зона в правом нижнем углу исчезает. При этом значения относительной устойчивости РНС для (6)–(8) выше, чем для (3)–(5).

Для данных, приведенных в табл. 1 и 2, нет необходимости выполнять полный алгоритм ускоренного построения диаграмм относительной устойчивости. Достаточно реализовать

упрощенный алгоритм. В случае табл. 1 достаточно найти границу  $U$ -зоны, для табл. 2 — границу  $L$ -зоны. Для столбцов «Плотные коды», «Раздвинутые коды» и «Широкие коды» табл. 1 потребовалось выполнить соответственно 34, 35 и 38 сеансов моделирования. Для аналогичных столбцов табл. 2 потребовалось 23, 24 и 24 сеанса моделирования. Полный алгоритм ускоренного построения диаграмм на этих же данных предусматривает выполнение 76 сеансов моделирования вместо 325 сеансов для известного метода [4].

### Заключение

Полученные новые результаты по свойствам потоковых РНС состоят в следующем.

Предложен метод ускоренного оценивания значения относительной устойчивости РНС, который позволяет в  $N_\alpha \cdot N_{Thr} / 2(N_\alpha + N_{Thr})$  раз уменьшить число реализуемых сеансов модели-

рования. Метод основан на построении границ интересующей зоны, содержащей все положительные решения. Полученные результаты моделирования подтверждают преимущества предложенного метода по сравнению с известными решениями.

Показано, что на относительную устойчивость РНС существенное влияние оказывает принятый вариант кодирования входных сигналов. Установлено, что эта устойчивость повышается, если входные последовательные сигналы будут иметь частичные пересечения элементов. Она также растет с повышением плотности используемых кодов. С учетом этого при поиске целесообразных архитектур РНС необходимо их согласовывать с возможными вариантами кодирования входных сигналов.

Предложенный метод рекомендуется применять при исследовании перспективных моделей РНС, которые в дальнейшем могут реализовываться аппаратным путем.

### Литература

- Osipov V., Osipova M. Space-time signal binding in recurrent neural networks with controlled elements. *Neurocomputing*, 2018, no. 308, pp. 194–204. <https://doi.org/10.1016/j.neucom.2018.05.009>
- Osipov V., Nikiforov V., Zukova N., Miloserdov D. Urban traffic flows forecasting by recurrent neural networks with spiral structures of layers. *Neural Computing and Applications*, 2020, no. 32, pp. 14885–14897. <https://doi.org/10.1007/s00521-020-04843-5>
- Osipov V., Kuleshov S., Zaytseva A., Levonevskiy D., Miloserdov D. Neural network forecasting of news feeds. *Expert Systems with Applications*, 2021, no. 169, Article 114521. <https://doi.org/10.1016/j.eswa.2020.114521>
- Осипов В. Ю., Никифоров В. В. Возможности рекуррентных нейронных сетей с управляемыми элементами по восстановлению потоков кадров. *Информационно-управляющие системы*, 2019, № 5, с. 10–17. doi:10.31799/1684-8853-2019-5-10-17
- Осипов В. Ю., Милосердов Д. И. Нейросетевое прогнозирование событий для роботов с непрерывным обучением. *Информационно-управляющие системы*, 2020, № 5, с. 33–42. doi:10.31799/1684-8853-2020-5-33-42
- Rolls E., Treves A. The neural encoding of information in the brain. *Progress in Neurobiology*, 2011, no. 95, pp. 448–490. doi:10.1016/j.pneurobio.2011.08.002
- Rozzi S., Fogassi L. Neural coding for action execution and action observation in the prefrontal cortex and its role in the organization of socially driven behavior. *Frontiers in Neuroscience*, 2017, no. 11, Article 492. <https://doi.org/10.3389/fnins.2017.00492>
- Kong Q., Han J., Zeng Yi., Xu B. Efficient coding matters in the organization of the early visual system. *Neural Networks*, 2018, no. 108, pp. 218–226. <https://doi.org/10.1016/j.neunet.2018.04.019>
- Kim T., Bair W., and Pasupathy A. Neural coding for shape and texture in macaque area V4. *The Journal of Neuroscience*, 2019, no. 39, pp. 4760–4774. <https://doi.org/10.1523/JNEUROSCI.3073-18.2019>
- Gerstner W., Kistler W. M. *Spiking neuron models*. Cambridge University Press, 2002. <https://doi.org/10.1017/CBO9780511815706>
- Kreiman G. Neural coding: computational and biophysical perspectives. *Physics of Life Reviews*, 2004, no. 1, pp. 71–102. <https://doi.org/10.1016/j.plrev.2004.06.001>
- Cessac B., Paugam-Moisy H., Vieville T. Overview of facts and issues about neural coding by spikes. *Journal of Physiology — Paris*, 2010, no. 104, pp. 5–18. <https://doi.org/10.1016/j.jphysparis.2009.11.002>
- Palm G. Neural associative memories and sparse coding. *Neural Networks*, 2013, no. 37, pp. 165–171. <https://doi.org/10.1016/j.neunet.2012.08.013>
- Ruys de Perez A., Matusевич L. F., Shiu A. Neural codes and the factor complex. *Advances in Applied Mathematics*, 2020, no. 114, Article 101977. <https://doi.org/10.1016/j.aam.2019.101977>
- Zhengzhong L., Schwartz D., Ditzler G., Koyluoglu O. The impact of encoding-decoding schemes and weight normalization in spiking neural networks. *Neural Networks*, 2018, no. 108, pp. 365–378. <https://doi.org/10.1016/j.neunet.2018.08.024>
- Azarfar A., Calcini N., Huang C., Zeldenrust F., Celikel T. Neural coding: A single neuron’s perspective. *Neuroscience & Biobehavioral Reviews*, 2018, no. 94,

- pp. 238–247. <https://doi.org/10.1016/j.neubiorev.2018.09.007>
17. Guo W., Fouda M. E., Eltawil A. M., and Salama K. N. Neural coding in spiking neural networks: A comparative study for robust neuromorphic systems. *Frontiers in Neuroscience*, 2021, no. 15, Article 638472. <https://doi.org/10.3389/fnins.2021.638472>
18. Zeldenrust F., Wadman W., Englitz B. Neural coding with bursts—current state and future perspectives. *Frontiers in Computational Neuroscience*, 2018, vol. 12, Article 48. <https://doi.org/10.3389/fncom.2018.00048>
19. Haykin S. *Neural networks and learning machines*. Third Ed. New-York, Prentice Hall, 2008. 938 p.
20. Aouiti C., M’hamdi M. S., Chérif F., Alimi A. M. Impulsive generalized high-order recurrent neural networks with mixed delays: Stability and periodicity. *Neurocomputing*, 2018, no. 321, pp. 296–307. <https://doi.org/10.1016/j.neucom.2017.11.037>
21. Ozcan N. New conditions for global stability of neutral-type delayed Cohen – Grossberg neural networks. *Neural Networks*, 2018, no. 106, pp. 1–7. <https://doi.org/10.1016/j.neunet.2018.06.009>

UDC 004.8

doi:10.31799/1684-8853-2021-3-9-18

### Coding and robustness of signal processing in streaming recurrent neural networks

V. Yu. Osipov<sup>a</sup>, Dr. Sc., Tech., Professor, [orcid.org/0000-0001-5905-4415](https://orcid.org/0000-0001-5905-4415), [osipov\\_vasily@mail.ru](mailto:osipov_vasily@mail.ru)

V. V. Nikiforov<sup>a</sup>, Dr. Sc., Tech., Professor, [orcid.org/0000-0002-6836-6303](https://orcid.org/0000-0002-6836-6303)

<sup>a</sup>Saint-Petersburg Institute for Informatics and Automation of the RAS, 39, 14 Line V. O., 199178, Saint-Petersburg, Russian Federation

**Introduction:** When substantiating promising architectures of streaming recurrent neural networks, it becomes necessary to assess their stability in processing various input signals. For this, stability diagrams are constructed containing the results of simulation for each of the nodes of these diagrams. Such an estimation can be time-consuming and computationally intensive, especially when analyzing large neural networks. **Purpose:** Search for methods of quick construction of such diagrams and assessing the stability of streaming recurrent neural networks. **Results:** Analysis of the features of the stability diagrams under study showed that the nodes of the diagrams are grouped into continuous zones with the same ratio characteristics of the input signal processing defects. With this in mind, the article proposes a method for constructing these diagrams based on bypassing the boundaries of their zones. With this approach, you do not have to perform simulation for the interior nodes of each zone. The simulation should be performed only for the nodes adjacent to zone boundaries. Due to this, the number of nodes for which you need to perform simulation sessions is reduced by an order of magnitude. The influence of the input signal coding types on the streaming recurrent neural network stability has been investigated. It is shown that the representation of input signals in the form of sequences of single pulses with intersecting elements can provide greater stability as compared to pulses without any intersection.

**Keywords** — recurrent neural network, stability, simulation, coding and signal processing.

**For citation:** Osipov V. Yu., Nikiforov V. V. Coding and robustness of signal processing in streaming recurrent neural networks. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2021, no. 3, pp. 9–18 (In Russian). doi:10.31799/1684-8853-2021-3-9-18

### References

- Osipov V., Osipova M. Space-time signal binding in recurrent neural networks with controlled elements. *Neurocomputing*, 2018, no. 308, pp. 194–204. <https://doi.org/10.1016/j.neucom.2018.05.009>
- Osipov V., Nikiforov V., Zukova N., Miloserdov D. Urban traffic flows forecasting by recurrent neural networks with spiral structures of layers. *Neural Computing and Applications*, 2020, no. 32, pp. 14885–14897. <https://doi.org/10.1007/s00521-020-04843-5>
- Osipov V., Kuleshov S., Zaytseva A., Levonevskiy D., Miloserdov D. Neural network forecasting of news feeds. *Expert Systems with Applications*, 2021, no. 169, Article 114521. <https://doi.org/10.1016/j.eswa.2020.114521>
- Osipov V. Yu., Nikiforov V. V. Recurrent neural networks with controlled elements in restoring frame flows. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2019, no. 5, pp. 10–17 (In Russian). doi:10.31799/1684-8853-2019-5-10-17
- Osipov V. Yu., Miloserdov D. I. Neural network event forecasting for robots with continuous training. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2020, no. 5, pp. 33–42 (In Russian). doi:10.31799/1684-8853-2020-5-33-42
- Rolls E., Treves A. The neural encoding of information in the brain. *Progress in Neurobiology*, 2011, no. 95, pp. 448–490. doi:10.1016/j.pneurobio.2011.08.002
- Rozzi S., Fogassi L. Neural coding for action execution and action observation in the prefrontal cortex and its role in the organization of socially driven behavior. *Frontiers in Neuroscience*, 2017, no. 11, Article 492. <https://doi.org/10.3389/fnins.2017.00492>
- Kong Q., Han J., Zeng Yi., Xu B. Efficient coding matters in the organization of the early visual system. *Neural Networks*, 2018, no. 108, pp. 218–226. <https://doi.org/10.1016/j.neunet.2018.04.019>
- Kim T., Bair W., and Pasupathy A. Neural coding for shape and texture in macaque area V4. *The Journal of Neuroscience*, 2019, no. 39, pp. 4760–4774. <https://doi.org/10.1523/JNEUROSCI.3073-18.2019>
- Gerstner W., Kistler W. M. *Spiking neuron models*. Cambridge University Press, 2002. <https://doi.org/10.1017/CBO9780511815706>
- Kreiman G. Neural coding: computational and biophysical perspectives. *Physics of Life Reviews*, 2004, no. 1, pp. 71–102. <https://doi.org/10.1016/j.plrev.2004.06.001>
- Cessac B., Paugam-Moisy H., Vieville T. Overview of facts and issues about neural coding by spikes. *Journal of Physiology — Paris*, 2010, no. 104, pp. 5–18. <https://doi.org/10.1016/j.jphysparis.2009.11.002>
- Palm G. Neural associative memories and sparse coding. *Neural Networks*, 2013, no. 37, pp. 165–171. <https://doi.org/10.1016/j.neunet.2012.08.013>

14. Ruys de Perez A., Matusevich L. F., Shiu A. Neural codes and the factor complex. *Advances in Applied Mathematics*, 2020, no. 114, Article 101977. <https://doi.org/10.1016/j.aam.2019.101977>
15. Zhengzhong L., Schwartz D., Ditzler G., Koyluoglu O. The impact of encoding-decoding schemes and weight normalization in spiking neural networks. *Neural Networks*, 2018, no. 108, pp. 365–378. <https://doi.org/10.1016/j.neunet.2018.08.024>
16. Azarfar A., Calcini N., Huang C., Zeldenrust F., Celikel T. Neural coding: A single neuron's perspective. *Neuroscience & Biobehavioral Reviews*, 2018, no. 94, pp. 238–247. <https://doi.org/10.1016/j.neubiorev.2018.09.007>
17. Guo W., Fouda M. E., Eltawil A. M., and Salama K. N. Neural coding in spiking neural networks: A comparative study for robust neuromorphic systems. *Frontiers in Neuroscience*, 2021, no. 15, Article 638472. <https://doi.org/10.3389/fnins.2021.638474>
18. Zeldenrust F., Wadman W., Englitz B. Neural coding with bursts—current state and future perspectives. *Frontiers in Computational Neuroscience*, 2018, vol. 12, Article 48. <https://doi.org/10.3389/fncom.2018.00048>
19. Haykin S. *Neural networks and learning machines*. Third Ed. New-York, Prentice Hall, 2008. 938 p.
20. Aouiti C., M'hamdi M. S., Chérif F., Alimi A. M. Impulsive generalized high-order recurrent neural networks with mixed delays: Stability and periodicity. *Neurocomputing*, 2018, no. 321, pp. 296–307. <https://doi.org/10.1016/j.neucom.2017.11.037>
21. Ozcan N. New conditions for global stability of neutral-type delayed Cohen – Grossberg neural networks. *Neural Networks*, 2018, no. 106, pp. 1–7. <https://doi.org/10.1016/j.neunet.2018.06.009>

---

### УВАЖАЕМЫЕ АВТОРЫ!

Научные базы данных, включая SCOPUS и Web of Science, обрабатывают данные автоматически. С одной стороны, это ускоряет процесс обработки данных, с другой — различия в транслитерации ФИО, неточные данные о месте работы, области научного знания и т. д. приводят к тому, что в базах оказывается несколько авторских страниц для одного и того же человека. В результате для всех по отдельности считаются индексы цитирования, что снижает рейтинг ученого.

Для идентификации авторов в сетях Thomson Reuters проводит регистрацию с присвоением уникального индекса (ID) для каждого из авторов научных публикаций.

Процедура получения ID бесплатна и очень проста, есть возможность провести регистрацию на 12-ти языках, включая русский (чтобы выбрать язык, кликните на зеленое поле сверху справа на стартовой странице): <https://orcid.org>

---