

УДК 004.27+621.382

DOI:10.25729/ESI.2024.34.2.005

Решение задачи классификации с использованием сегментной спайковой модели нейрона на ПЛИС

Исаков Тим Тимурович^{1,3}, Бахшиев Александр Валерьевич^{1,2},
Корсаков Антон Михайлович³

¹Санкт-Петербургский политехнический университет Петра Великого,

²ООО Системы Компьютерного зрения, Россия, Санкт-Петербург,

³Центральный научно-исследовательский и опытно-конструкторский институт робототехники и технической кибернетики, Россия, Санкт-Петербург, t.isakov@rtc.ru

Аннотация. Спайковые нейронные сети — это класс нейронных сетей, имеющих в своей основе биоподобные модели нейрона. Спайковая природа таких сетей при наличии специализированных вычислителей позволяет достигать показателей энергоэффективности на порядки выше, чем у классических нейронных сетей, что является особо важным для встраивания нейронных сетей в автономные системы. Однако, на данный момент таких вычислителей нет в открытом доступе, поэтому хорошей альтернативой являются ПЛИС. Одним из классов спайковых моделей нейронов являются сегментные модели. Сегментные модели, в отличие от точечных, позволяют учитывать структуру нейрона, что, в свою очередь, позволяет воссоздавать более сложную динамику нейронных структур. Существующие нейроморфные вычислители позволяют реализовать только ограниченный набор спайковых моделей, что также является поводом для использования ПЛИС. На данный момент нет работ по аппаратной реализации сегментных моделей нейрона, поэтому данная работа является актуальной. В ходе работы были выполнены аппроксимация и аппаратная реализация CSNM модели (Compartmental spiking neuron model) на ПЛИС. Для проверки работоспособности полученной реализации был построен классификатор данных IRIS. На основе результатов был сделан вывод, что полученная модель имеет конкурентные показатели по количеству используемых ресурсов ПЛИС, а скорость расчетов при этом на три порядка выше, чем на компьютере. Точность полученной реализации уступает другим работам по причине использования малого количества нейронов и грубой аппроксимации. Дальнейшие исследования методов аппроксимации и алгоритмов инкрементного обучения позволят увеличить точность. Также планируется использовать оперативную память в целях масштабирования моделей и оптимизации вычислений. Еще одним направлением дальнейших работ является реализация обучения на чипе, как в целях ускорения тестирования моделей, так и для исследований обучения с подкреплением.

Ключевые слова: спайковые нейронные сети, классификация, программируемая логика

Цитирование: Исаков Т.Т. Решение задачи классификации с использованием сегментной спайковой модели нейрона на ПЛИС / Т.Т. Исаков, А.В. Бахшиев, А.М. Корсаков // Информационные и математические технологии в науке и управлении. – 2024. – № 2(34). – С. 50-58. – DOI:10.25729/ESI.2024.34.2.005.

Введение. На сегодняшний день нейронные сети благодаря впечатляющим результатам активно используются в самых разных отраслях жизни. Однако их дальнейшее развитие затруднено из-за различных ограничений. Одним из главных ограничений является низкая энергоэффективность нейронных сетей, что является проблемой при их встраивании в автономные системы. Решением этой и других проблем нейронных сетей занимаются ученые в области спайковых нейронных сетей, в основе которых лежат более биоподобные модели нейронов. Одним из главных отличий таких сетей от традиционных является то, что спайковые нейроны обмениваются не действительными числами, а спайками (единичными импульсами). Принцип работы нейронных сетей, построенных на спайковых моделях нейрона, подразумевает высокую степень асинхронности и параллельности работы сети, что трудновыполнимо на существующих CPU или даже GPU. По этой причине разрабатываются специализированные вычислители для работы со спайковыми нейронными сетями, такие, как TrueNorth [1], SpiNNaker [2], Loihi [3], Tianjic [4]. Однако данные вычислители либо

недоступны, либо способны воспроизводить очень ограниченный набор существующих спайковых моделей нейрона. Отличной альтернативой для исследователей является использование ПЛИС для реализации своих собственных моделей и архитектур. На данный момент существует большое количество реализаций различных моделей нейрона: LIF [5, 6, 7], SRM [8], SRM0 [9], Ижикевич [10], Ходжкин-Хаксли [11]. Существуют также работы по решению посредством ПЛИС задач распознавания образов (pattern recognition) [5, 12], распознавания символов (character recognition) [13], управления (control) [14], а также генерации паттернов (pattern generation) [15, 16].

Модели, учитывающие структуру нейрона, называют сегментными моделями (compartmental). На данный момент можно выделить несколько вариантов сегментных моделей: двухсегментную модель [17], состоящую из модулей сомы и дендритов, трехсегментную модель [18], состоящую из модулей сомы, апикальных и базальных дендритов и CSNM модель [19], состоящую из модулей синапсов, ионного механизма, описывающего участки дендритов, и сомы с низкопороговой зоной. В работе [20] демонстрируется, что многосегментные модели нейрона, реализованные на нейроморфных чипах, позволяют имитировать динамические процессы нейронных цепей мозга при низких энергозатратах. Однако сегментные модели нейрона могут также использоваться и для решения практических задач.

В работе [14] предлагается алгоритм обучения спайковой нейронной сети, использующей многосегментные нейроны для обработки данных о состоянии системы. Данная сеть управляет персонажем в компьютерных играх Atari, и, благодаря использованию сегментных нейронов, имеет значительный прирост игровых очков относительно других нейронных сетей, в том числе спайковых. Авторы делают вывод, что сегментные модели играют большую роль в глубоком распределенном обучении с подкреплением, что является важным основанием для дальнейшего развития сегментных моделей.

В работе [21] представлен алгоритм структурного обучения для решения задачи классификации. Данный метод имеет ряд преимуществ, так как для обучения может быть достаточно одного примера. Целью данной работы является реализация на ПЛИС классификатора для данных IRIS [22] на основе CSNM модели нейрона. CSNM модель была выбрана, так как по сравнению с другими сегментными моделями она позволяет строить сложную структуру дендритного дерева, а дальнейшее развитие алгоритма может позволить одновременно использовать структурное и параметрическое обучение. Полученная реализация позволит выполнять моделирование спайковых нейронных сетей на сегментных моделях нейрона в реальном времени или даже быстрее.

1. Сегментная спайковая модель нейрона. Как уже было сказано выше, сегментные модели учитывают структуру дендритного дерева. Модель CSNM описывается некоторой системой дифференциальных уравнений, подробное описание которой представлено в работе [19]. Для решения данной системы используется метод преобразования Эйлера.

Согласно модели, входные сигналы поступают на синапсы, отражающие влияние синаптического тока на сегмент мембраны нейрона. Поведение синапсов моделируется с помощью полученной после преобразований системы (начальные условия $\rho(0) = 0$):

$$\left. \begin{aligned} \rho_i &= \rho_{i-1} + h \left(\frac{x - \rho_{i-1}}{T_s} \right) \\ T_s(t) &= \begin{cases} \tau_s, & x(t) > 0 \\ \tau_d, & x(t) \leq 0 \end{cases} \\ i_s &= \rho \frac{\varepsilon_s}{R_s} \omega \end{aligned} \right\} \quad (1)$$

где ρ – характеризует условную концентрацию медиатора, выделившегося в ответ на импульс,

h – шаг интегрирования,
 T_s – постоянная времени выделения или распада медиатора,
 R_s – сопротивление синапса,
 ε_s – электродвижущая сила синаптической передачи,
 ω – вес связи,
 i_s – выходной синаптический ток.

Значения синаптического тока подаются на сегменты мембраны. Ионные механизмы, описывающие деполяризацию и гиперполяризацию мембраны, моделируются с помощью системы (начальные условия: $U(0) = 0$):

$$\left. \begin{aligned} U_i &= U_{i-1} + h \left(\frac{U_\Sigma - U_{i-1}(1 + g_\Sigma R_m)}{T_I} \right) \\ T_I &= \frac{C_m}{g_\Sigma + R_m^{-1}} \end{aligned} \right\} \quad (2)$$

где U_i – вклад сегмента в величину внутриклеточного потенциала,

h – шаг интегрирования,
 R_m – сопротивление ионного механизма,
 C_m – емкость ионного механизма,

g_Σ – суммарная проводимость моделей активных синапсов,

U_Σ – ожидаемый вклад сегмента в величину внутриклеточного потенциала при отсутствии внешнего возбуждения, которая определяется активностью предыдущих сегментов мембраны.

Выходы сегментов мембраны на теле нейрона (сома) суммируются и подаются на генератор выходного сигнала, имеющий некоторый порог срабатывания. Если нейрон срабатывает, на входы сегментов на теле нейрона подается тормозящий сигнал, что позволяет моделировать рефрактерный период.

2. Аппаратная реализация CSNM модели на ПЛИС. Решение дифференциальных уравнений напрямую требует большого количества вычислительных ресурсов. Ресурсы ПЛИС ограничены, поэтому создание нейронных сетей большого объема в таком случае проблематично. Решения дифференциальных уравнений, описывающих спайковые нейронные сети, включают в себя ядра с экспоненциальной временной зависимостью. Существуют различные варианты аппроксимации таких экспоненциальных функций, направленные на определенную точность и вычислительные затраты. В данной работе используется аппроксимация похожая на методы base-2 и CORDIC, так как операции умножения и деления заменяются на сдвиги и операции сложения. Важным отличием является то, что для полученной модели не нужно использовать память, что позволяет сохранить ресурсы ПЛИС. Также вычисления с плавающей запятой приведены к целочисленным вычислениям, что также экономит ресурсы ПЛИС. Минусом такого подхода является подбор диапазонов значений рассматриваемых величин и констант. Также из-за использования сдвигов теряется гибкость настройки модели, однако на данном этапе это не является проблемой.

Логика ПЛИС позволяет детально аппаратно реализовать сегментную модель нейрона. На рисунке 1 представлена RTL-схема сегментной модели нейрона. Для работы модели используются 3 входа: **clk**, регулирующий синхронную работу элементов; **reset**, обнуляющий значения блоков при запуске; **input spike**, отвечающий за входной сигнал нейрона. Входной сигнал передается в блок **Synapse**, реализующий динамику, описанную в системе уравнений (1). Значения, генерируемые синапсами, передаются в возбуждающий канал **Exc_Channel** (динамика описана в системе уравнений (2)). В данной реализации на блок **Inh_Channel**, моделирующий тормозный ионный механизм, сигнал с синапса не подается. Выходы возбуждающего и тормозного канала передаются в блок **Membrane**, внутри которого эти

значения суммируются. Как только выход блока мембраны **input potential** становится больше порога в блоке **LTZone**, описывающего низкороговую зону, нейрон выдает выходной спайк **Output spike**. Вместе с этим запускается обратная связь, которая записывает значение обратной связи в регистры **Feedback**. Обратная связь уменьшает проходимость сигнала от синапса через ионные механизмы, из-за чего нейрон некоторое время перестает выдавать выходной спайк. Полученная схема полностью описывает цикл работы сегментной спайковой модели нейрона.

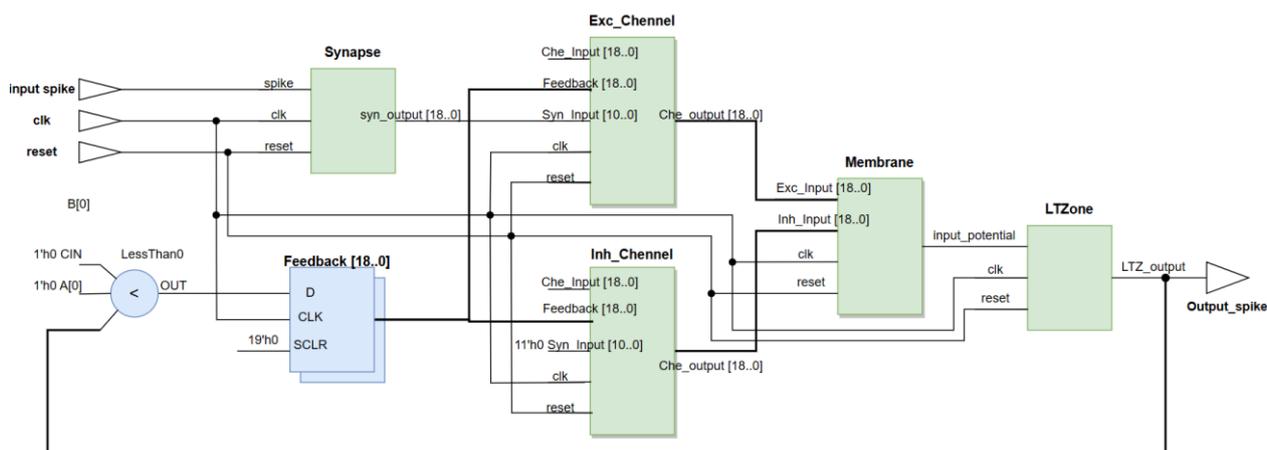


Рис. 1. RTL-схема сегментной модели нейрона

Таким образом, используя описанные модули и добавляя блоки ионных механизмов, можно построить сегментную модель нейрона любой сложности, ограничиваясь только ресурсами ПЛИС. В таблице 1 представлено сравнение полученной реализации CSNM модели с другими реализациями спайковых моделей нейронов в ПЛИС. Реализации сравниваются по количеству используемых ресурсов ПЛИС, количеству реализованных нейронов и среднеквадратичному отклонению реализации.

Таблица 1. Сравнение полученной реализации CSNM модели со связанными работами

Модель	FF (регистры) + LUT (логические элементы)	DSP	Количество нейронов	Память	Среднеквадратичное отклонение
LIF model (2011) [5]	14098 + 22815	-	648	126Kb (BRAM)	-
LIF model (2020) [6]	135 + 218	-	1	15Kb (-)	0,0015
SRM model (2012) [7]	202 + 378	-	1	-	-
LIF model (2006) [8]	4581	-	1,024	320Kb	LIF model (2006) [8]
SRM0 model (2020) [9]	296 + 314	4	512	1,003x64b (DRAM)	$4 \times 10^{-12} - 1 \times 10^{-10}$
CSNM (Данная работа)	67 + 245 (16607)	-	1(3)	-	0,004 – 0,01

По значениям, представленным в таблице, можно сделать вывод, что модель имеет конкурентные показатели используемых ресурсов. Стоит отметить, что количество ресурсов, занимаемое нейроном, сильно зависит от структуры самого нейрона. Для сравнения, в таблице указаны ресурсы, используемые для одного базового необученного нейрона, а в скобках для 3

нейронов, обученных на данных IRIS [22]. В дальнейшем планируется использование памяти, что позволит сэкономить ресурсы ПЛИС при масштабировании моделей.

3. Классификатор данных IRIS. С помощью среды NeuroModeler [23] на трех примерах (по одному для каждого класса) из данных IRIS [22] была обучена модель классификатора. Полученная схема (рисунок 2) из трех нейронов была реализована в ПЛИС на чипе Cyclone IV EP4CE22F17C6N с 22.320 логическими элементами (рисунок 2). В данных IRIS рассматриваются 4 признака, которые преобразуются во входные паттерны нейронов. Также дополнительно используется калибровочный сигнал, чтобы нейроны оценивали абсолютные значения задержек, а не относительные. Обратная связь при срабатывании нейронов распространяется на все нейроны, чтобы избежать нескольких срабатываний. Таким образом «победителем» будет являться нейрон, испустивший спайк раньше.

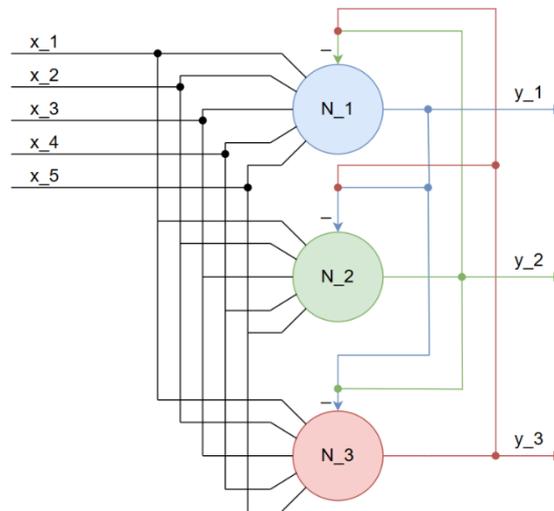


Рис. 2. Схема построенного классификатора

Сравнение представленной реализации с другими решениями задачи классификации данных IRIS с использованием импульсных нейронов представлено в таблице 2. Для описания архитектур используется формат $N_i - N_h - N_o$, указывающий количество нейронов на входном, скрытом и выходном слоях модели. Модели SRESN и DoB имеют по 2 слоя, однако количество нейронов выходного слоя может варьироваться (значения в скобках). Реализация CSNM модели также двухслойная, однако в реализации, представленной в данной работе, используется только один слой из трех нейронов.

Таблица 2. Сравнение существующих спайковых классификаторов данных IRIS

Модель	Архитектура	Кол-во настраиваемых параметров	Точность (%)
SpikeProp (Bohte et al. 2002) [24]	25-10-3	4480	96,1
SWAT (Wade et al. 2010) [25]	24-312-3	936	95,3
SRESN (2016) [26]	24-(5-11)	120-264	97,0
DoB-SNN (2021) [27]	5-(5-8)	25-40	97,7
CSNM (2021) [21]	12-3	120	83,3
CSNM (Данная работа)	3	30	76,6

Из таблицы видно, что полученная модель имеет сравнительно низкие показатели точности. Снижение точности относительно оригинального эксперимента [24] вызвано упрощением модели с 15 до 3 нейронов для экономии ресурсов ПЛИС, а также аппроксимацией вычислений. Положительной стороной модели является относительно малое количество настраиваемых параметров. Другим важным достижением является скорость работы алгоритма. Реакция одного нейрона на компьютере посредством CPU измеряется в

секундах или долях секунд (в зависимости от мощности CPU), в то время как реакция такого же нейрона на ПЛИС измеряется в долях миллисекунд. Таким образом, в данной реализации мы получаем прирост в тысячу раз, что особо ощутимо при оценке относительно больших моделей на больших данных, что на CPU может занимать несколько суток. Стоит отметить, что на данном этапе обучение нейрона выполняется с помощью одного примера, что также отражается в низкой точности. Инкрементное обучение, позволяющее дообучать нейроны на нескольких примерах, является одним из направлений будущих исследований.

Заключение. В ходе работы была выполнена аппроксимация и аппаратная реализация CSNM модели нейрона. На основе полученной модели для оценки работы реализации был построен классификатор, ресурсные характеристики которого не уступают схожим работам, а вычислительные характеристики превосходят таковые на CPU. Точность полученной модели ниже, чем в аналогичных работах, однако дальнейшие исследования в области инкрементного обучения и методов аппроксимации позволят увеличить точность. Также планируется использовать оперативную память в целях масштабирования моделей и оптимизации вычислений. Еще одним направлением дальнейших работ является реализация обучения на чипе как в целях ускорения тестирования моделей, так и для исследований обучения с подкреплением.

Благодарности. Результаты получены в рамках выполнения государственного задания Минобрнауки России №075-00697-24-00 от 27.12.2023 «Исследование и разработка биоподобной системы управления поведением мобильных роботов на базе энергоэффективных программно-аппаратных нейроморфных средств» (FNRG-2022-0016 1021060307690-3-1.2.1;2.2.2).

Список источников

1. Merolla P.A. et al. A million spiking-neuron integrated circuit with a scalable communication network and interface. *Science*, 2014, vol.345, pp. 668-673.
2. Van Albada S.J. et al. Performance comparison of the digital neuromorphic hardware SpiNNaker and the neural network simulation software NEST for a full-scale cortical microcircuit model. *Frontiers in neuroscience*, 2018, vol.12, p. 291.
3. Davies M. et al. Loihi: A neuromorphic manycore processor with on-chip learning. *Ieee Micro*, 2018, vol. 38, pp. 82-99.
4. Pei J. et al. Towards artificial general intelligence with hybrid Tianjic chip architecture. *Nature*, 2019, vol. 572, pp. 106-111.
5. Caron L. C., Mailhot F., Rouat J. FPGA implementation of a spiking neural network for pattern matching. *IEEE International Symposium of circuits and systems (ISCAS)*, 2011, pp. 649-652.
6. Perez-Peña F., Cifredo-Chacon M.A., Quiros-Olozabal A. Digital neuromorphic real-time platform. *Neurocomputing*, 2020, vol.371, pp. 91-99.
7. Ros E. et al. Real-time computing platform for spiking neurons (RT-spike). *IEEE Trans. Neural Networks*, 2006, vol.17, pp. 1050-1063.
8. Rosado-Muñoz A., Bataller-Mompeán M., Guerrero-Martínez J. FPGA implementation of spiking neural networks. *IFAC Proceedings Volumes*, 2012, vol.45, pp. 139-144.
9. Kim J. et al. Hardware-efficient emulation of leaky integrate-and-fire model using template-scaling-based exponential function approximation. *IEEE Transactions on circuits and systems I: regular papers*, 2020, v.68, pp. 350-362.
10. Ambroise M. et al. Biorealistic spiking neural network on FPGA, 2013, 47th Annual Conference on information sciences and systems (CISS). *IEEE*, 2013, pp. 1-6.
11. Yaghini Bonabi S. et al. FPGA implementation of a biological neural network based on the Hodgkin-Huxley neuron model. *Frontiers in neuroscience*, 2014, v.8, pp. 379.
12. Nevarez Y. et al. Accelerator framework of spike-by-spike neural networks for inference and incremental learning in embedded systems. 9th International Conference on modern circuits and systems technologies (MOCASST), *IEEE*, 2020, pp. 1-5.
13. Rice K.L. et al. FPGA implementation of Izhikevich spiking neural networks for character recognition. *International Conference on reconfigurable computing and FPGAs*, *IEEE*, 2009, pp. 451-456.

14. Sun Y. et al. Multi-compartment neuron and population encoding improved spiking neural network for deep distributional reinforcement learning. arXiv preprint arXiv:2301.07275, 2023.
15. Donati E. et al. A spiking implementation of the lamprey's Central Pattern Generator in neuromorphic VLSI. 2014. IEEE Biomedical circuits and systems conference (BioCAS). Proceedings, IEEE, 2014, pp. 512-515.
16. Rostro-Gonzalez H. et al. A CPG system based on spiking neurons for hexapod robot locomotion. Neurocomputing, 2015, v.170, pp. 47-54.
17. Urbanczik R., Senn W. Learning by the dendritic prediction of somatic spiking. Neuron, 2014, vol.81, pp. 521-528.
18. Sacramento J. et al. Dendritic cortical microcircuits approximate the backpropagation algorithm. Advances in neural information processing systems, 2018, vol.31.
19. Бахшиев А.В. Сегментная спайковая модель нейрона CSNM / А.В. Бахшиев, А.А. Демчева // Известия высших учебных заведений. Прикладная нелинейная динамика, 2022. – Т. 30. – №. 3. – С. 299-310.
20. Kopsick J.D. et al. Robust resting-state dynamics in a large-scale spiking neural network model of area CA3 in the mouse hippocampus. Cognitive computation, 2022, pp. 1-21.
21. Astarova L.A. et al. Compartmental spiking neuron model for pattern classification. Journal of physics: conference series, IOP Publishing, 2021, vol. 2094, p. 032032.
22. UCI Machine Learning Repository, available at: <https://archive.ics.uci.edu/ml/datasets/iris> (accessed: 08/24/2023).
23. Neuro modeler. Available at: <https://neuromodeler.ru/> (accessed: 08/24/2023).
24. Bohte S.M., Kok J.N., La Poutré J.A. SpikeProp: backpropagation for networks of spiking neurons. ESANN, 2000, vol.48, pp. 419-424.
25. Wade J.J. et al. SWAT: A spiking neural network training algorithm for classification problems. IEEE Transactions on neural networks, 2010, vol.21, pp. 1817-1830.
26. Dora S. et al. Development of a self-regulating evolving spiking neural network for classification problem. Neurocomputing, 2016, vol.171, pp. 1216-1229.
27. Saranirad V. et al. DoB-SNN: a new neuron assembly-inspired spiking neural network for pattern classification. 2021 International Joint Conference on Neural Networks (IJCNN), IEEE, 2021, pp. 1-6.

Исаков Тим Тимурович. Аспирант Санкт-Петербургского политехнического университета Петра Великого, математик в Центральном научно-исследовательском и опытно-конструкторском институте робототехники и технической кибернетики (ЦНИИ РТК). Область научных интересов: алгоритм обучения спайковых нейронных сетей и их аппаратной реализацией на ПЛИС. AuthorID: 1117866, SPIN: 8742-1353, ORCID: 0000-0003-4437-5018, 194064, Санкт-Петербург, Тихорецкий пр., 21.

Бахшиев Александр Валерьевич. К.т.н., СПбПУ, доцент. Направления исследований: искусственные нейронные сети, нейроморфные системы, компьютерное зрение. AuthorID: 172135, SPIN: 7055-8676, ORCID: 0000-0002-1284-0088, palexab@gmail.com, 195251, Санкт-Петербург, ул. Политехническая, д. 29.

Корсаков Антон Михайлович. Старший научный сотрудник ЦНИИ РТК. Направления исследований: искусственные нейронные сети, нейроморфные системы, компьютерное зрение. AuthorID: 172137, SPIN: 3736-3401, anton_korsakov@mail.ru, 194064, Санкт-Петербург, Тихорецкий пр., д. 21.

UDC 004.27+621.382

DOI:10.25729/ESI.2024.34.2.005

Solution of the classification task using a compartmental spiking neuron model on an FPGA

Tim T. Isakov^{1,3}, Alexandr V. Bakhshiev^{1,2}, Anton M. Korsakov³

¹Peter the Great St. Petersburg Polytechnic University,

²LLC Computer Vision Systems, Russia, Saint Petersburg,

³Russian state scientific center for robotics and technical cybernetics (RTC),

Russia, Saint Petersburg, t.isakov@rtc.ru

Abstract. Spike neural networks are a class of neural networks based on plausible neuron models. The spiking nature of such networks, in the presence of specialized accelerators, makes it possible to achieve energy efficiency values that are orders of magnitude higher than those of classical neural networks, which is especially important for integrating neural networks into autonomous systems. However, now such computers are not publicly available, so FPGAs are a good alternative. One of the classes of spike neuron models are segmental models. Segmental models, unlike point ones, allow one to consider the structure of a neuron, which in turn allows one to reproduce more complex dynamics of neural structures. Existing neuromorphic computers allow the implementation of only a limited set of spike models, which is also a reason for using FPGAs. Today, there is no work on hardware implementation of segmented neuron models, so this work is relevant. During the work, the approximation and hardware implementation of CSNM model (Compartmental spiking neuron model) on an FPGA was performed. To evaluate the performance of the resulting implementation, an IRIS data classifier was built. Based on the results, it was concluded that the resulting model has competitive indicators in terms of the amount of FPGA resources used, and the calculation speed is three orders of magnitude higher than on a computer. The accuracy of the resulting implementation is inferior to other works due to the use of a small number of neurons and rough approximation. Further research into approximation methods and incremental learning algorithms will improve accuracy. It is also planned to use RAM for scaling models and optimizing calculations. Another area of future work is implementing on-chip learning, both to speed up model testing and for reinforcement learning research.

Keywords: spiking neural networks, classification, FPGA

Acknowledgements: The work was carried out as the part of the state task of the Russian Ministry of Education and Science for 2024 №075-00697-24-00 "Research and development of a biosimilar system for controlling the behavior of mobile robots based on energy-efficient software and hardware neuromorphic tools" (FNRG-2022-0016 1021060307690-3-1.2.1;2.2.2).

References

1. Merolla P. A. et al. A million spiking-neuron integrated circuit with a scalable communication network and interface, *Science*, 2014, v.345, pp. 668-673.
2. Van Albada S. J. et al. Performance comparison of the digital neuromorphic hardware SpiNNaker and the neural network simulation software NEST for a full-scale cortical microcircuit model, *Frontiers in neuroscience*, 2018, v.12, p. 291.
3. Davies M. et al. Loihi: A neuromorphic manycore processor with on-chip learning, *Ieee Micro*, 2018, v.38, pp. 82-99.
4. Pei J. et al. Towards artificial general intelligence with hybrid Tianjic chip architecture, *Nature*, 2019, v.572, pp. 106-111.
5. Caron L. C., Mailhot F., Rouat J. FPGA implementation of a spiking neural network for pattern matching, 2011 IEEE International Symposium of Circuits and Systems (ISCAS), 2011, pp. 649-652.
6. Perez-Peña F., Cifredo-Chacon M. A., Quiros-Olozabal A. Digital neuromorphic real-time platform, *Neurocomputing*, 2020, v.371, pp. 91-99.
7. Ros E. et al. Real-time computing platform for spiking neurons (RT-spike), *IEEE Trans. Neural Networks*, 2006, v.17, pp. 1050-1063.
8. Rosado-Muñoz A., Bataller-Mompeán M., Guerrero-Martínez J. FPGA implementation of spiking neural networks, *IFAC Proceedings Volumes*, 2012, v.45, pp. 139-144.
9. Kim J. et al. Hardware-efficient emulation of leaky integrate-and-fire model using template-scaling-based exponential function approximation, *IEEE Transactions on Circuits and Systems I: Regular Papers*, 2020, v.68, pp. 350-362.
10. Ambroise M. et al. Biorealistic spiking neural network on FPGA, 2013, 47th Annual Conference on information sciences and systems (CISS). IEEE, 2013, pp. 1-6.
11. Yaghini Bonabi S. et al. FPGA implementation of a biological neural network based on the Hodgkin-Huxley neuron model, *Frontiers in neuroscience*, 2014, v.8, pp. 379.
12. Nevarez Y. et al. Accelerator framework of spike-by-spike neural networks for inference and incremental learning in embedded systems, 9th International Conference on Modern Circuits and Systems Technologies (MOCASST), IEEE, 2020, pp. 1-5.
13. Rice K.L. et al. FPGA implementation of Izhikevich spiking neural networks for character recognition. *International Conference on reconfigurable computing and FPGAs*, IEEE, 2009, pp. 451-456.
14. Sun Y. et al. Multi-compartment neuron and population encoding improved spiking neural network for deep distributional reinforcement learning. *arXiv preprint arXiv:2301.07275*, 2023.

15. Donati E. et al. A spiking implementation of the lamprey's Central Pattern Generator in neuromorphic VLSI, 2014 IEEE Biomedical Circuits and Systems Conference (BioCAS) Proceedings, IEEE, 2014, pp. 512-515.
16. Rostro-Gonzalez H. et al. A CPG system based on spiking neurons for hexapod robot locomotion, Neurocomputing, 2015, v.170, pp. 47-54.
17. Urbanczik R., Senn W. Learning by the dendritic prediction of somatic spiking, Neuron, 2014, v.81, pp. 521-528.
18. Sacramento J. et al. Dendritic cortical microcircuits approximate the backpropagation algorithm, Advances in neural information processing systems, 2018, v.31.
19. Bakhshiev A.V., Demcheva A.A. Segmentnaja spaikovaja model neirona CSNM [Compartmental spiking neuron model CSNM]. Izvestiya vysshikh uchebnykh zavedeniy. Prikladnaya nelineynaya dinamika [News of higher educational institutions. Applied nonlinear dynamics], 2022, v.30, no. 3, pp. 299-310.
20. Kopsick J. D. et al. Robust resting-state dynamics in a large-scale spiking neural network model of area CA3 in the mouse hippocampus, Cognitive computation, 2022, pp. 1-21.
21. Astapova L. A. et al. Compartmental spiking neuron model for pattern classification, Journal of Physics: Conference Series, IOP Publishing, 2021, v.2094, p. 032032.
22. UCI Machine Learning Repository, available at: <https://archive.ics.uci.edu/ml/datasets/iris> (accessed: 08/24/2023).
23. Neuro modeler. Available at: <https://neuromodeler.ru/> (accessed: 08/24/2023).
24. Bohte S. M., Kok J. N., La Poutré J. A. SpikeProp: backpropagation for networks of spiking neurons, ESANN, 2000, v.48, pp. 419-424.
25. Wade J. J. et al. SWAT: A spiking neural network training algorithm for classification problems, IEEE Transactions on neural networks, 2010, v.21, pp. 1817-1830.
26. Dora S. et al. Development of a self-regulating evolving spiking neural network for classification problem. Neurocomputing, 2016, v.171, pp. 1216-1229.
27. Saranirad V. et al. DoB-SNN: A New Neuron Assembly-Inspired Spiking Neural Network for Pattern Classification, 2021 International Joint Conference on Neural Networks (IJCNN), IEEE, 2021, pp. 1-6.

Isakov Tim Timurovich. Graduate student at Peter the Great St. Petersburg polytechnic university, mathematician at the Russian state scientific center for robotics and technical cybernetics (RTC). Research direction: research into learning algorithms for spiking neural networks and their hardware implementation on FPGAs. AuthorID: 1117866, SPIN 8742-1353, ORCID: 0000-0003-4437-5018, t.isakov@rtc.ru, 194064, Saint Petersburg, Tikhoretsky Avenue 21.

Bakhshiev Alexander Valeryevich. Ph.D., SPbPU, assistant professor. Research direction: artificial neural networks, neuromorphic systems, computer vision. AuthorID 172135, SPIN 7055-8676, ORCID 0000-0002-1284-0088, palexab@gmail.com, 195251, St. Petersburg, Polytechnicheskaya, 29.

Korsakov Anton Mikhailovich. RTC, senior researcher, Research direction: artificial neural networks, neuromorphic systems, computer vision. AuthorID: 172137, SPIN: 3736-3401, anton_korsakov@mail.ru, 194064, St. Petersburg, Tikhoretsky pr., 21.

Статья поступила в редакцию 08.12.2023; одобрена после рецензирования 30.05.2024; принята к публикации 03.06.2024.

The article was submitted 12/08/2023; approved after reviewing 05/30/2024; accepted for publication 06/03/2024.