

УДК 004.3:004.93

Вопросы реализации нейросетевых алгоритмов на мемристорных кроссбарах*

© 2019 г. А. Ю. Морозов^{1,2}, Д. Л. Ревизников^{1,2}, К. К. Абгарян^{1,2,§}

¹ *Федеральный исследовательский центр «Информатика и управление»
Российской академии наук,
ул. Вавилова, д. 44, корп. 2, Москва, 119333, Россия*

² *Московский авиационный институт (национальный исследовательский университет),
Волоколамское шоссе, д. 4, Москва, 125993, Россия*

Аннотация. Присущее мемристорным кроссбарам свойство естественной параллелизации матрично-векторных операций создает возможности для их эффективного использования в нейросетевых вычислениях. Аналоговые вычисления производятся на порядки быстрее по сравнению с вычислениями на центральном процессоре и на графических ускорителях. Кроме того, значительно ниже энергозатраты на проведение математических операций. При этом существенной особенностью аналоговых вычислений является небольшая точность. В связи с этим актуальным является исследование зависимости качества работы нейронной сети от точности задания ее весов. Рассмотрены две сверточные нейронные сети, обученные на наборах данных MNIST (рукописные цифры) и CIFAR_10 (самолеты, лодки, машины и т. д.). Первая состоит из двух сверточных слоев, одного слоя подвыборки и двух полносвязанных слоев, а вторая — из четырех сверточных слоев, двух слоев подвыборки и двух полносвязанных слоев. Вычисления в сверточных и полносвязанных слоях выполняются через матрично-векторные операции, которые эффективно реализуются на мемристорных кроссбарах. Слои подвыборки подразумевают операцию нахождения максимального значения из нескольких, которая также может быть реализована на аналоговом уровне. Процесс обучения нейронной сети происходит отдельно от анализа данных. Как правило, на этапе обучения используются градиентные методы оптимизации, реализацию которых целесообразно выполнять на центральном процессоре. Показано, что для получения приемлемого качества распознавания в случае с сетью, обученной на MNIST, требуется 3—4 бита точности при задании ее весов, а в случае с сетью, обученной на CIFAR_10, — 6—8 бит.

Ключевые слова: мемристор, кроссбар, точность, нейросеть, свертка, MNIST, CIFAR_10

Введение

В последние десять лет активно идет развитие технологий, в основе которых лежит электрический элемент — мемристор. Мемристор — это сопротивление, проводимость которого меняется в зависимости от суммарного протекшего через него электрического заряда. При этом установившиеся сопротивление в отсутствие тока не изменяется со временем. То есть, мемристор является элементарной ячейкой долговременной энергонезависимой памяти [1, 2].

Мемристорный кроссбар — это объединение мемристоров в матрицу. Благодаря закону Ома и закону Кирхгофа, на основе кроссбара можно выполнять аналоговое произведение матрицы на вектор. Аналоговые вычисления производятся на порядки быстрее по сравнению с вычислениями на центральном процессоре и на графических ускорителях. Кроме того, существенно ниже энергозатраты на проведение математических операций. При этом в связи с ограниченностью количества

уровней проводимости мемристора возникает проблема точности представления чисел. Корпорация Hewlett-Packard уже создала в «железе» мемристорный кроссбар, на котором реализована операция матрично-векторного умножения [3, 4]. На данный момент точность задания матрицы ограничивается 6 битами (для каждого мемристора выделяется 64 уровня проводимости).

Поскольку матрично-векторное умножение — важнейший элемент работы нейронных сетей, появляется возможность эффективной аналого-

Морозов Александр Юрьевич^{1,2} — канд. физ.-мат. наук, научный сотрудник, e-mail: morozov@infway.ru; **Ревизников Дмитрий Леонидович**^{1,2} — доктор физ.-мат. наук, профессор, e-mail: reviznikov@gmail.com; **Абгарян Каринэ Карленовна**^{1,2,§} — доктор физ.-мат. наук, заведующая отделом, e-mail: kristal83@mail.ru

§ Автор для переписки

* Статья подготовлена по материалам доклада, представленного на I-й международной конференции «Математическое моделирование в материаловедении электронных компонентов», Москва, 21—23 октября 2019 г.

вой реализации нейросетевых алгоритмов. Одной из наиболее востребованных задач искусственных нейронных сетей является задача распознавания изображений. За счет естественной параллелизации матрично-векторных операций на кроссбарах имеется возможность существенно ускорить работу нейронных сетей. При этом, в связи с ограниченностью точности, важно понимать, как дискретизация весов нейронной сети будет влиять на качество ее работы. Этому вопросу посвящена настоящая статья.

В работе описывается общая проблематика аналоговой реализации нейросетевых алгоритмов и рассматривается архитектура сверточных нейронных сетей. На примере двух нейронных сетей, обученных на наборах данных MNIST и CIFAR_10, исследуется зависимость качества распознавания от точности задания весов. Рассмотрены возможности применения мемристорных кроссбаров для аналоговой реализации нейросетевых алгоритмов.

Цель работы — исследование влияния дискретизации весов нейронной сети на качество ее работы, что позволяет оценивать возможность реализации нейросетевых алгоритмов на мемристорных кроссбарах.

Нейронные сети

Можно выделить два основных подхода к построению аналоговых нейросетей. Первый подход предполагает обучение сети непосредственно на «железном» уровне, что в наибольшей мере соответствует аналоговой парадигме нейроморфных сетей. Наиболее простой вариант может быть представлен двухслойной полносвязной сетью с одним слоем мемристорных элементов (синапсов). В этом случае соответствие синаптического веса (проводимости мемристора) интенсивности входного сигнала, подаваемого на нейрон, обеспечивает достаточно высокую точность решения задачи классификации [5]. В последнее время усиливается интерес к импульсным нейронным сетям, обучение которых организуется с использованием правила Хебба и

синаптической пластичности (метод STDP [6], согласно которому изменение весов синапсов нейрона зависит от разницы во времени между входным и выходным импульсом) [7—11]. Однако такой подход трудно реализуем применительно к сложным многослойным сетям.

Другой подход предполагает разделение этапов обучения и анализа данных. Как правило, на этапе обучения используются градиентные методы оптимизации, реализацию которых целесообразно выполнять на центральном процессоре (или на графических ускорителях). Этап анализа данных состоит в прямом распространении сигнала по сети, что, учитывая возможности параллельного выполнения матрично-векторных умножений, идеально ложится на архитектуру мемристорного кроссбара [12]. Таким образом, этап анализа данных эффективно реализуется в аналоговом режиме.

При таком подходе возникают две основные проблемы. Первая связана с переходом от непрерывного диапазона изменения синаптических весов к дискретному набору значений, обеспечиваемых мемристором. В наиболее радикальном варианте, когда мемристор может находиться в двух состояниях, высокоомном и низкоомном, речь идет о бинаризации синаптических весов. Ясно, что такая дискретизация может вносить погрешности в результаты анализа данных. Поэтому актуальной задачей является выявление зависимости погрешности дискретизации от количества дискретных уровней для различных наборов данных.

Другой проблемой является возможное несовершенство мемристорного элемента [13], ведущее к неконтролируемому изменению уровня проводимости в ходе функционирования системы или инициации. В этой связи актуальным является анализ влияния дефектов мемристорных элементов на точность решения задач обработки данных. Отметим, что модель мемристора можно описать как динамическую систему с неопределенностями и использовать для ее исследования соответствующие методы [14—17].

В настоящей работе рассматривается второй подход к реализации многослойных нейронных

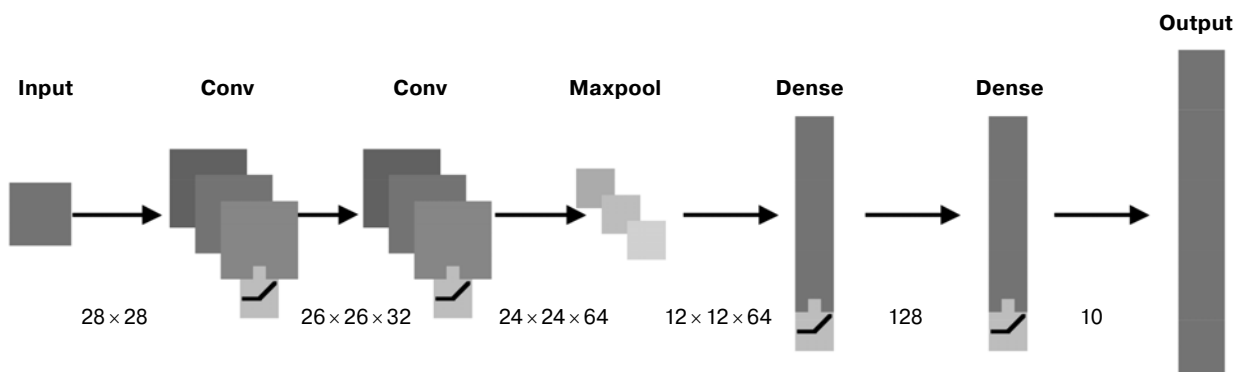


Рис. 1. Архитектура сверточной нейронной сети (MNIST)
Fig. 1. Convolutional Neural Network Architecture (MNIST)

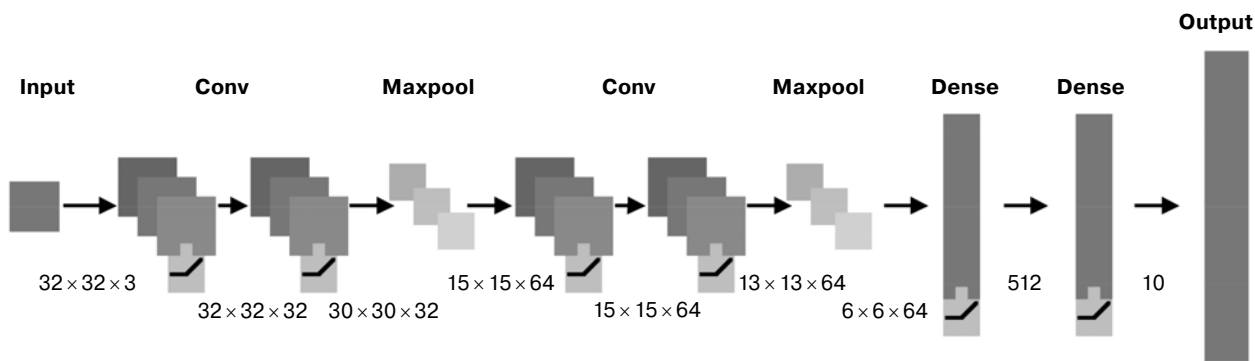


Рис. 2. Архитектура сверточной нейронной сети (CIFAR_10)
 Fig. 2. Convolutional neural network architecture (CIFAR_10)

сетей. Вычисления в каждом слое сети включают перемножение матрицы весов на входной вектор, прибавление вектора смещения и применение нелинейной функции активации. Все эти действия возможно выполнить в аналоговом представлении.

Нейронная сеть заранее обучается с использованием центрального процессора или графических ускорителей и далее ее веса размещаются в соответствующих мемристорных кроссбарах.

Рассматриваются две сверточные нейронные сети. Первая состоит из двух сверточных слоев (*conv*), одного слоя подвыборки (*maxpool*) и двух полносвязанных слоев (*dense*) (рис. 1), а вторая из четырех сверточных слоев, двух слоев подвыборки и двух полносвязанных слоев (рис. 2).

Сверточный слой (*conv*) состоит из набора ядер (рис. 3). Каждое ядро это матрица (в общем случае тензор) небольшого размера (в представленных сетях, размером 3×3 и $3 \times 3 \times 3$) которая «накладывается» с перехлестом на входные данные с последую-

щим перемножением соответствующих элементов и суммированием. Например, такие операции как размытие изображения или выделение контуров являются сверткой. Если представить входные данные в виде вектора, а ядра свертки в виде больших разреженных матриц, то получение результата сверточного слоя сведется к перемножению матрицы на вектор, что в свою очередь эффективно реализуется на мемристорных кроссбарах.

Слой подвыборки (*maxpool*) применяется для уменьшения размера данных (рис. 3). В рассмотренных сетях из каждых четырех промежуточных значений оставляется только одно — максимальное. Здесь подразумевается, что операция выбора максимального элемента из нескольких может быть реализована на аналоговом уровне. Отметим, что если для уменьшения размера данных использовать не выбор максимального элемента, а усреднение, тогда данный слой можно представить как сверточный слой, в котором ядра покрывают входные данные без перехлеста.

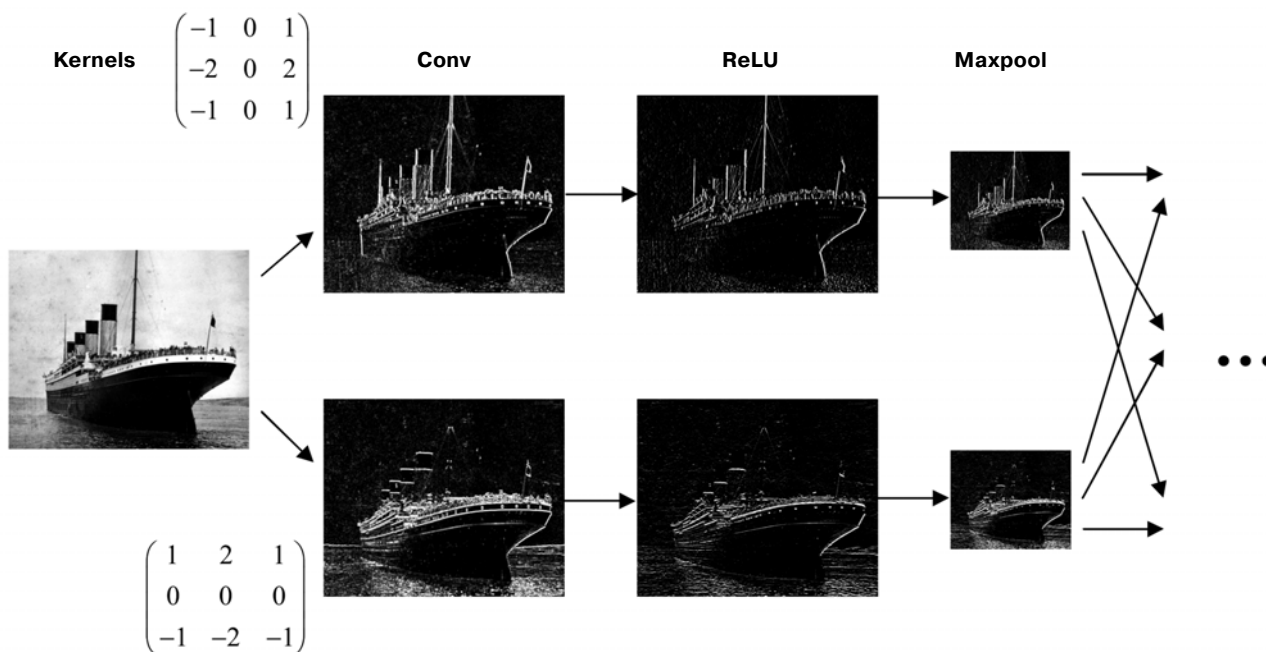
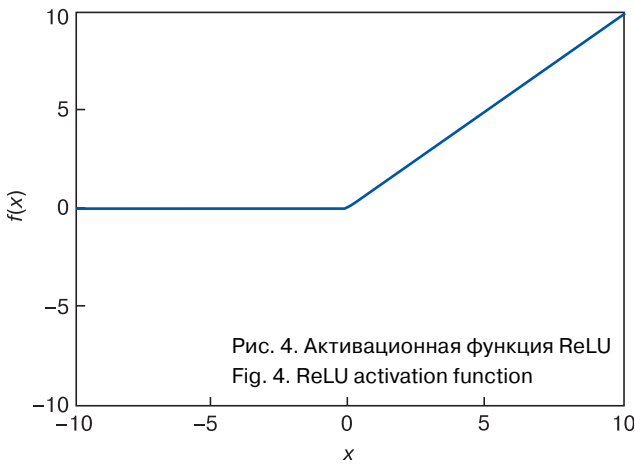


Рис. 3. Часть сверточной нейронной сети
 Fig. 3. Part of a convolutional neural network



К результату каждого слоя (кроме слоя подвыборки) поэлементно применяется активационная функция ReLU (рис. 4):

$$f(x) = \max(0, x).$$

Данная функция представляется на железном уровне в виде полупроводникового диода.

В целом все элементы рассмотренных нейронных сетей возможно реализовать на аналоговых электронных компонентах.



Рис. 5. Набор данных MNIST
Fig. 5. MNIST dataset



Рис. 6. Набор данных CIFAR_10
Fig. 6. Dataset CIFAR_10

Результаты и их обсуждение

Для обучения и работы с сетями использовалась библиотека Keras, написанная на языке Python, которая является надстройкой над фреймворками DeepLearning4j, TensorFlow и Theano [18].

Первая нейронная сеть (см. рис. 1) была обучена на наборе данных MNIST состоящем из изображений рукописных цифр (рис. 5) [19]. Вторая нейронная сеть (см. рис. 2) обучалась на CIFAR_10, состоящем из самолетов, автомобилей, животных и т. д. (рис. 6) [20]. Точность распознавания на тестовых выборках обученной нейронной сети составила: для MNIST — 99,16 % и для CIFAR10 — 74,86 %. Отметим, что для CIFAR_10 возможна реализация с более высокой точностью, однако целью данной работы является не достижение точности искусственной нейронной сети, а оценка погрешности вследствие дискретизации весов.

Далее была выполнена дискретизация весов: начиная с 2-х значений (1 бит точности) и до

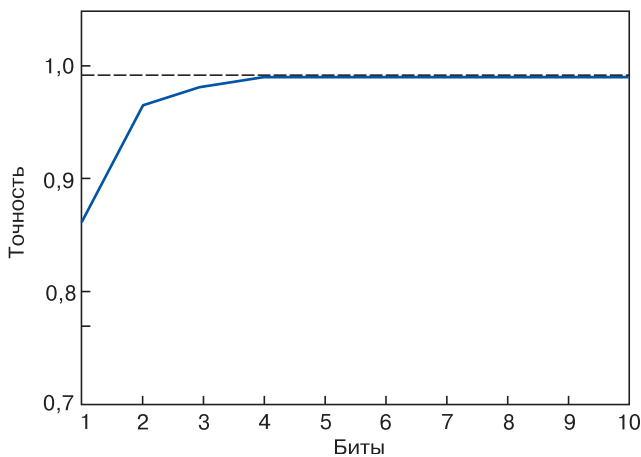


Рис. 7. Зависимость качества нейронной сети от точности задания весов (MNIST)

Fig. 7. Dependence of the quality of the neural network on the accuracy of setting the weights (MNIST)

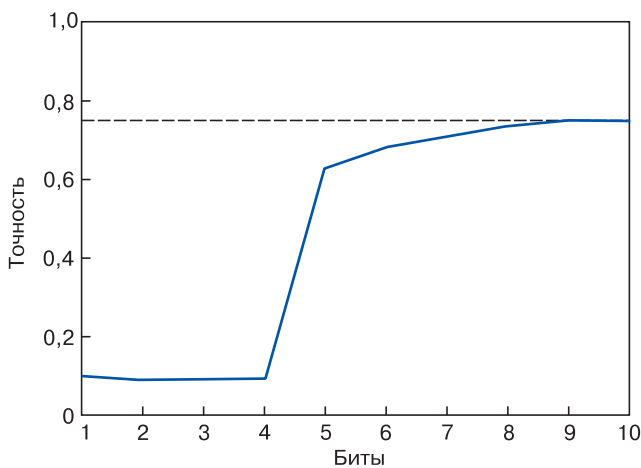


Рис. 8. Зависимость качества нейронной сети от точности задания весов (CIFAR_10)

Fig. 8. Dependence of the quality of the neural network on the accuracy of setting the weights (CIFAR_10)

1024 значений (10 бит точности). Дискретизация выполнялась в рамках каждого слоя отдельно. На рис. 7 и 8 показаны зависимости качества распознавания от количества бит точности задания весов сети. Пунктирной линией показана исходная точность распознавания.

Для сети, обученной на MNIST'e, для достижения приемлемой точности (98—99 %) требуется 3—4 бита точности. А для нейронной сети, обученной на CIFAR_10, нужно 6—8 бит.

Заключение

В работе рассмотрены некоторые вопросы реализации нейросетевых алгоритмов на мемристорных кроссбарах. В связи с ограниченностью точности аналоговых вычислений исследовано влияние дискретизации весов нейронных сетей на качество распознавания. На примере двух сверточных сетей, обученных на наборах данных MNIST и CIFAR_10 показано, что возможно применять мемристорные кроссбары для аналоговой реализации нейросетевых алгоритмов. Для сети, обученной на MNIST'e, для достижения приемлемой точности (98—99 %) требует 3—4 бита точности. А для нейронной сети, обученной на CIFAR_10, нужно 6—8 бит.

Библиографический список

1. Wong H.-S.P., Lee H.-Y., Yu S., Chen Y.-S., Wu Y., Chen P.-S., Lee B., Chen F. T., Tsai M.-J. Metal-oxide RRAM // Proceedings of the IEEE. 2012. V. 100, Iss. 6. P. 1951—1970. DOI: 10.1109/JPROC.2012.2190369
2. Yang J. J., Strukov D. B., Stewart D. R., Memristive devices for computing // Nature Nanotechnology. 2013. V. 8, N 1. P. 13—24. DOI: 10.1038/nnano.2012.240
3. Li C., Hu M., Li Y., Jiang H., Ge N., Montgomery E., Zhang J., Song W., Dávila N., Graves C. E., Li Z., Strachan J. P., Lin P., Wang Z., Barnell M., Wu Q., Williams R. S., Yang J. J., Xia Q. Analogue signal and image processing with large memristor crossbars // Nature Electronics. 2018. V. 1, N 1. P. 52—59. DOI: 10.1038/s41928-017-0002-z
4. Hu M., Graves C. E., Li C., Li Y., Ge N., Montgomery E., Dávila N., Jiang H., Williams R. S., Yang J. J., Xia O., Strachan J. P. Memristor-based analog computation and neural network classification with a dot product engine // Advanced Materials. 2018. V. 30, Iss. 9. P. 1705914. DOI: 10.1002/adma.201705914
5. Тарков М. С. Реализация нейронной WTA-сети на мемристорном кроссбаре // ПДМ. Приложение, 2015, Вып. 8. С. 151—154. DOI: 10.17223/2226308X/8/59
6. Diehl P. U., Cook M. Unsupervised learning of digit recognition using spike-timing-dependent plasticity // Front. Comput. Neurosci. 2015. V. 9, Art. No. 99. P. 9. DOI: 10.3389/fncom.2015.00099
7. Ambrogio S., Balatti S., Milo V., Carboni R., Wang Z.-Q., Calderoni A., Ramaswamy N., Ielmini D. Neuromorphic learning and recognition with one-transistor-one-resistor synapses and bistable metal oxide RRAM // IEEE Transactions on Electron Devices. 2016. V. 63, N 4. P. 1508—1515. DOI: 10.1109/TED.2016.2526647
8. Guo Y., Wu H., Gao B., Qian H. Unsupervised learning on resistive memory array based spiking neural networks // Front. Neurosci. 2019. V. 13, Art. No. 812. DOI: 10.3389/fnins.2019.00812
9. Milo V., Pedretti G., Laudato M., Bricalli A., Ambrosi E., Bianchi S., Chicca E., Ielmini D. Resistive switching synapses for unsupervised learning in feed-forward and recurrent neural networks // IEEE International Symposium on Circuits and Systems (ISCAS). Florence (Italy): IEEE, 2018. P. 1—5. DOI: 10.1109/ISCAS.2018.8351824
10. Pedretti G., Bianchi S., Milo V., Calderoni A., Ramaswamy N., Ielmini D. Modeling-based design of brain-inspired

spiking neural networks with RRAM learning synapses // IEEE International Electron Devices Meeting (IEDM). San Francisco (CA, USA): IEEE, 2017. P. 28.1.1—28.1.4. DOI: 10.1109/IEDM.2017.8268467

11. Milo V., Ielmini D., Chicca E. Attractor networks and associative memories with STDP learning in RRAM synapses // IEEE International Electron Devices Meeting (IEDM). San Francisco (CA, USA): IEEE, 2017. P. 11.2.1—11.2.4. DOI: 10.1109/IEDM.2017.8268369

12. Li B., Shan Y., Hu M., Wang Y., Chen Y., Yang H., Memristor-based approximated computation // International Symposium on Low Power Electronics and Design (ISLPED). Beijing (China): IEEE, 2013. P. 242—247. DOI: 10.1109/ISLPED.2013.6629302

13. Теплов Г. С., Горнев Е. С. Модель на языке Verilog—А многоуровневого биполярного мемристора с учетом девиаций параметров переключения // Микроэлектроника. 2019. Т. 48, № 3. С. 163—175. DOI: 10.1134/S0544126919030104

14. Morozov A. Yu., Reviznikov D. L. Adaptive interpolation algorithm based on a kd-tree for numerical integration of systems of ordinary differential equations with interval initial conditions // Differential Equations. 2018. V. 54, N 7. P. 945—956. DOI: 10.1134/S0012266118070121

15. Morozov A. Yu., Reviznikov D. L., Gidaspov V. Yu. Adaptive interpolation algorithm based on a kd-tree for the problems of chemical kinetics with interval parameters // Mathematical Models and Computer Simulations. 2019. V. 11, N 4. P. 622—633. DOI: 10.1134/S2070048219040100

16. Morozov A. Yu., Reviznikov D. L. Modelling of dynamic systems with interval parameters on graphic processors // Программная инженерия. 2019. Т. 10, № 2. С. 69—76. DOI: 10.17587/prin.10.69-76

17. Морозов А. Ю., Ревизников Д. Л. Методы компьютерного моделирования динамических систем с интервальными параметрами. М.: Изд-во МАИ, 2019. 160 с.

18. Gulli A., Pal S. Deep learning with Keras: implement neural networks with Keras on Theano and TensorFlow. Birmingham; Mumbai: Packt Publishing Ltd., 2017. 490 p.

19. MNIST CNN. URL: https://keras.io/examples/mnist_cnn/ (дата обращения: 01.10.2019).

20. Train a simple deep CNN on the CIFAR10 small images dataset. URL: https://keras.io/examples/cifar10_cnn/ (дата обращения: 01.10.2019).

Работа выполнена при поддержке гранта РФФИ № 19-29-03051 мк.

Статья поступила в редакцию 26 декабря 2019 г.

Izvestiya vuzov. Materialy elektronnoy tekhniki = Materials of Electronics Engineering. 2019, vol. 22, no. 4, pp. 272—278. DOI: 10.17073/1609-3577-2019-4-272-278

Issues of implementing neural network algorithms on memristor crossbars

A. Yu. Morozov^{1,2}, D. L. Reviznikov^{1,2}, K. K. Abgaryan^{1,2,§}

¹ Federal Research Centre “Information and Control” of the Russian Academy of Sciences, 44 Vavilov Str., Moscow 119333, Russia

² Moscow Aviation Institute (National Research University), 4 Volokolamskoe shosse, 4, Moscow 125993, Russia

Abstract. The property of natural parallelization of matrix–vector operations inherent in memristor crossbars creates opportunities for their effective use in neural network computing. Analog calculations are orders of magnitude faster in comparison to calculations on the central processor and on graphics accelerators. Besides, mathematical operations energy costs are significantly lower. The essential feature of analog computing is its low accuracy. In this regard, studying the dependence of neural network quality on the accuracy of setting its weights is relevant. The paper considers two convolutional neural networks trained on the MNIST (handwritten digits) and CIFAR_10 (airplanes, boats, cars, etc.) data sets. The first convolutional neural network consists of two convolutional layers, one subsample layer and two fully connected layers. The second one consists of four convolutional layers, two subsample layers and two fully connected layers. Calculations in convolutional and fully connected layers are performed through matrix–vector operations that are implemented on memristor crossbars. Sub–sampling layers imply the operation of finding the maximum value from several values. This operation can be implemented at the analog level. The process of training a neural network runs separately from data analysis. As a rule, gradient optimization methods are used at the training stage. It is advisable to perform calculations using these methods on CPU. When setting the weights, 3—4 precision bits are required to obtain an acceptable recognition quality in the case the network is trained on MNIST. 6—10 precision bits are required if the network is trained on CIFAR_10.

Keywords: memristor, crossbar, accuracy, neural network, convolution, MNIST, CIFAR_10

References

1. Wong H.–S. P., Lee H.–Y., Yu S., Chen Y.–S., Wu Y., Chen P.–S., Lee B., Chen F. T., Tsai M.–J. Metal–oxide RRAM. *Proceedings of the IEEE*, 2012, vol. 100, no. 6, pp. 1951—1970. DOI: 10.1109/JPROC.2012.2190369

2. Yang J. J., Strukov D. B., Stewart D. R., Memristive devices for computing. *Nature Nanotechnology*, 2013, vol. 8, no. 1, pp. 13—24. DOI: 10.1038/nnano.2012.240

3. Li C., Hu M., Li Y., Jiang H., Ge N., Montgomery E., Zhang J., Song W., Dávila N., Graves C. E., Li Z., Strachan J. P., Lin P., Wang Z.,

Barnell M., Wu Q., Williams R. S., Yang J. J., Xia Q. Analogue signal and image processing with large memristor crossbars. *Nature Electronics*, 2018, vol. 1, no. 1, pp. 52—59. DOI: 10.1038/s41928-017-0002-z

Information about authors:

Alexander Yu. Morozov^{1,2}: Cand. Sci. (Phys.–Math.), Researcher (morozov@infway.ru); **Dmitry L. Reviznikov**^{1,2}: Dr. Sci. (Phys.–Math.), Professor (reviznikov@gmail.com); **Karine K. Abgaryan**^{1,2,§}: Dr. Sci. (Phys.–Math.), Head of the Department (kristal83@mail.ru)

§ Corresponding author

4. Hu M, Graves C. E., Li C., Li Y., Ge N., Montgomery E., Dávila N., Jiang H., Williams R. S., Yang J. J., Xia O., Strachan J. P. Memristor-based analog computation and neural network classification with a dot product engine. *Advanced Materials*, 2018, vol. 30, no. 9, p. 1705914. DOI: 10.1002/adma.201705914
5. Tarkov M. S. Implementation of a neural WTA-network on the memristor crossbar. *Prikl. Diskr. Mat. Suppl.*, 2015, no. 8, pp. 151—154. (In Russ.). DOI: 10.17223/2226308X/8/59
6. Diehl P. U., Cook M. Unsupervised learning of digit recognition using spike-timing-dependent plasticity. *Front. Comput. Neurosci.*, 2015, vol. 9, art. 99, p. 9. DOI: 10.3389/fncom.2015.00099
7. Ambrogio S., Balatti S., Milo V., Carboni R., Wang Z.-Q., Calderoni A., Ramaswamy N., Ielmini D. Neuromorphic learning and recognition with one-transistor-one-resistor synapses and bistable metal oxide RRAM. *IEEE Transactions on Electron Devices*, 2016, vol. 63, no. 4, pp. 1508—1515. DOI: 10.1109/TED.2016.2526647
8. Guo Y., Wu H., Gao B., Qian H. Unsupervised learning on resistive memory array based spiking neural networks. *Front. Neurosci.*, 2019, vol. 13, art. 812. DOI: 10.3389/fnins.2019.00812
9. Milo V., Pedretti G., Laudato M., Bricalli A., Ambrosi E., Bianchi S., Chicca E., Ielmini D. Resistive switching synapses for unsupervised learning in feed-forward and recurrent neural networks. In: *IEEE International Symposium on Circuits and Systems (ISCAS)*. Florence (Italy): IEEE, 2018, pp. 1—5. DOI: 10.1109/ISCAS.2018.8351824
10. Pedretti G., Bianchi S., Milo V., Calderoni A., Ramaswamy N., Ielmini D. Modeling-based design of brain-inspired spiking neural networks with RRAM learning synapses. In: *IEEE International Electron Devices Meeting (IEDM)*. San Francisco (CA, USA): IEEE, 2017, pp. 28.1.1—28.1.4. DOI: 10.1109/IEDM.2017.8268467
11. Milo V., Ielmini D., Chicca E. Attractor networks and associative memories with STDP learning in RRAM synapses. In: *IEEE International Electron Devices Meeting (IEDM)*. San Francisco (CA, USA): IEEE, 2017, pp. 11.2.1—11.2.4. DOI: 10.1109/IEDM.2017.8268369
12. Li B., Shan Y., Hu M., Wang Y., Chen Y., Yang H., Memristor-based approximated computation. In: *International Symposium on Low Power Electronics and Design (ISLPED)*. Beijing (China): IEEE, 2013, pp. 242—247. DOI: 10.1109/ISLPED.2013.6629302
13. Teplov G. S., Gornev E. S. Multilevel bipolar memristor model considering deviations of switching parameters in the Verilog-A language. *Russ. Microelectron.*, 2019, vol. 48, no. 3, pp. 131—142. DOI: 10.1134/S1063739719030107
14. Morozov A. Yu., Reviznikov D. L. Adaptive interpolation algorithm based on a kd-tree for numerical integration of systems of ordinary differential equations with interval initial conditions. *Differential Equations.*, 2018, vol. 54, no. 7, pp. 945—956. DOI: 10.1134/S00122666118070121
15. Morozov A. Yu., Reviznikov D. L., Gidasov V. Yu. Adaptive interpolation algorithm based on a kd-tree for the problems of chemical kinetics with interval parameters. *Mathematical Models and Computer Simulations*, 2019, vol. 11, no. 4, pp. 622—633. DOI: 10.1134/S2070048219040100
16. Morozov A. Yu., Reviznikov D. L. Modelling of dynamic systems with interval parameters on graphic processors. *Programmnaya Ingeneria = Software Engineering*, 2019, vol. 10, no. 2, pp. 69—76. DOI: 10.17587/prin.10.69-76
17. Morozov A. Yu., Reviznikov D. L. *Metody komp'yuternogo modelirovaniya dinamicheskikh sistem s interval'nymi parametrami* [Methods for computer modeling of dynamic systems with interval parameters]. Moscow: Izd-vo Moskovskogo aviatsionnogo instituta, 2019, 160 p. (In Russ.)
18. Gulli A., Pal S. Deep learning with Keras: implement neural networks with Keras on Theano and TensorFlow. Birmingham; Mumbai: Packt Publishing Ltd., 2017, 490 p.
19. MNIST CNN. URL: https://keras.io/examples/mnist_cnn/ (accessed: 01.10.2019).
20. Train a simple deep CNN on the CIFAR10 small images dataset. URL: https://keras.io/examples/cifar10_cnn/ (accessed: 01.10.2019).

Acknowledgments. *This work was supported by the RFBR grant No. 19-29-03051 MK.*

Received December 26, 2019