

Спайковые нейронные сети

В.А. Евграфов, Е.А. Ильюшин

Аннотация—За последние несколько лет методы глубокого обучения добились значительного прогресса и стали широко распространёнными инструментами для решения различных когнитивных задач. Чтобы использовать возможности глубокого обучения повсеместно, необходимо развернуть глубокое обучение не только на крупномасштабных компьютерах, но и на периферийных устройствах. Однако постоянно растущая сложность глубоких нейронных сетей вкупе с резким увеличением объема обрабатываемых данных предъявляют значительные энергетические требования к современным вычислительным платформам. Модель нейроморфных вычислений предполагает выполнение вычислений биологически правдоподобным образом. Частью нейроморфных вычислений являются спайковые нейронные сети, которые являются одним из ведущих кандидатов для преодоления ограничений нейронных вычислений и эффективного использования алгоритма машинного обучения в реальных приложениях. В данной работе рассматриваются биологические основы спайковых нейронных сетей, методы их обучения и создания, а также программные и аппаратные платформы для их использования.

Ключевые слова—спайковые нейронные сети, нейроморфные вычисления, искусственный интеллект

I. Введение

Человеческий мозг — невероятно сложная система. Он состоит из приблизительно 90 миллиардов нейронов [1], которые структурированы триллионами взаимосвязанных синапсов. Информация между нейронами передается при помощи электрических импульсов, называемых сигналами или спайками (spike). Воздействие, которое производит спайк посылаемый с пресинаптического нейрона на постсинаптический, зависит от силы синаптической связи, характеристики синапса который связывает эти два нейрона. Силы синаптических связей и структура связей между нейронами играют значительную роль в способности нервной системы обрабатывать информацию.

Способность мозга решать сложные задачи вдохновила многих исследователей изучать различные методы обработки информации, а также методы обучения нервной системы.

Как результат этих исследований, появился такой мощный и гибкий инструмент вычислений как искусственные нейронные сети (ИНС).

За последние несколько лет методы глубокого обучения добились значительного прогресса и стали широко распространёнными инструментами для решения различных когнитивных задач, таких как обнаружение объектов, распознавание речи и других. Различные методы

глубокого обучения обеспечивают эффективную оптимизацию глубоких сетей путём построения нескольких уровней иерархии признаков. Эти методы показывают замечательные результаты, которые иногда превосходят человеческую производительность [2, 3, 4]. Чтобы использовать возможности глубокого обучения повсеместно, необходимо развернуть глубокое обучение не только на крупномасштабных компьютерах, но и на периферийных устройствах (например, телефоне, планшете, умных часах, роботе и т. д.).

Однако постоянно растущая сложность глубоких нейронных сетей вкупе с резким увеличением объема обрабатываемых данных предъявляют значительные энергетические требования к современным вычислительным платформам.

Искусственные нейронные сети моделируют биологическую нервную систему. Они успешно используются в различных областях [5, 6]. Однако их высокий уровень абстракции по сравнению с биологическим аналогом [7] вместе с отсутствием в них возможности выразить временную динамику биологических нейронов стало причиной возникновения новых методов построения искусственных нейронных сетей, в которых ставка делается на более биологически достоверные нейронные модели — спайковые нейронные сети.

Модель нейроморфных вычислений предполагает выполнение вычислений биологически правдоподобным образом. Частью нейроморфных вычислений являются спайковые нейронные сети. Спайковая нейронная сеть (СНС) является одним из ведущих кандидатов для преодоления ограничений нейронных вычислений и эффективного использования алгоритма машинного обучения в реальных приложениях. Концепция СНС, которая часто рассматривается как нейронная сеть 3-го поколения [8], вдохновлена биологическими нервными механизмами [9], которые могут эффективно обрабатывать дискретные пространственно-временные сигналы (спайки). Интегрирующий нейрон с утечкой (leaky integrate-and-fire neuron, LIF neuron) - это простая модель спайкового нейрона, которая может быть охарактеризована внутренним состоянием, называемым мембранным потенциалом. Потенциал на мембране интегрирует входные сигналы с течением времени и генерирует выходной спайк всякий раз, когда этот потенциал преодолевает порог срабатывания нейрона. Относительно недавно были разработаны специализированные аппаратные средства для использования основанных на асинхронных сигналах архитектуры. Они имеют перспективы достижения сверхнизкого потребления ресурсов при интеллектуальной обработке потоковых пространственно-временных данных, особенно в глубоких иерархических сетях. Это связано с тем что в моделях СНС было замечено, что число спайков, а следовательно, и объем вычислений, значительно умень-

Статья получена 28 июня 2021.

Владимир Андреевич Евграфов, МГУ им. М.В. Ломоносова, (email: evgrafov.vladimir@gmail.com).

Евгений Альбинович Ильюшин, МГУ им. М.В. Ломоносова, (email: eugene.ilyushin@gmail.com).

шается на более глубоких слоях[10].

Благодаря их способности отразить разнообразную динамику биологических нейронов, а также включить и представить различные измерения информации такие, как время, частота, фаза, спайковые нейронные сети представляют собой многообещающий метод вычислений и потенциально способны моделировать сложные процессы обработки информации происходящие в мозге[11, 12, 13]

Также СНС потенциально способны обрабатывать большие объёмы данных и использовать группы спайков (spike trains) для представления информации[12]. Кроме того, спайковые сети пригодны для реализации на аппаратном обеспечении с низким потреблением энергии.

Модели спайковых нейронных сетей построены на основе биологических методов обработки информации, где разреженные во времени асинхронные сигналы передаются и обрабатываются массово-параллельным образом[14].

СНС работающие на нейроморфном аппаратном обеспечении показывают низкое энергопотребление и быструю обработку информации на основе приходящих событий.

Это делает их интересными кандидатами для эффективных реализаций глубоких нейронных сетей, которые являются предпочтительным методом решения многих задач машинного обучения.

II. Кодирование информации

Как нейроны кодируют информацию с помощью спайков — это один из важных вопросов, обсуждаемых в нейробиологии. Предполагается, что нейронная информация передаётся либо через частоту сигналов, либо через точное время спайков (временное кодирование). Существуют различные методы кодирования частотой спайков такие, как подсчёт количества спайков, плотность спайков или популяционная активность.

Широко распространено мнение что спайки – короткие, резкие всплески в потенциале на мембране нейрона – используются для передачи информации между нейронами[15]. Тем не менее, кодирование информации при помощи спайков является дискутируемой темой в сообществе вычислительной нейробиологии. Ранее, предполагалось что мозг кодирует информацию при помощи частоты спайков[16]. Но исследования в области нейробиологии показали, что высокоскоростная обработка информации в мозге не может быть осуществлена с использованием исключительно схемы кодирования на основе частоты спайков[17].

Хотя кодирование скорости обычно используется в традиционных СНС, такой подход может не передавать всю информацию, связанную с задачей быстрой обработки, такую как обработка цвета, визуальной информации, запаха и качества звука, поскольку информация, инкапсулированная в точное время спайков, игнорируется.

Было показано что человеческая система обработки визуальной информации решает задачу распознавания менее чем за 100 миллисекунд, при этом используя нейроны во множестве слоёв (начиная от сетчатки и заканчивая височной долей)[15]. Каждый нейрон требует приблизительно 10 миллисекунд для обработки сигнала.

На таком коротком промежутке времени частотное кодирование невозможно.

Высокоскоростная обработка сигналов может быть осуществлена с помощью метода точного времени спайка. Кроме того, отправка количества спайков, необходимого для частотного кодирования информации, требует значительного количества энергии и других ресурсов.

Вдобавок метод кодирования через точное время спайка позволяет кодировать большее количество информации используя малое количество спайковых нейронов[18]. Следовательно, становится ясно что кодирование через точное время отдельного спайка, а не только через количество спайков или частоту их появления, скорее всего является способом передачи информации.

Тем не менее, точные механизмы обучения биологических нейронов всё ещё остаются открытым вопросом.

III. Топологии

Обычно классификация топологий спайковых сетей рассматривает 3 типа топологий: сети с прямой связью, рекуррентные и гибридные сети. Такие сети как Synfire[19] и отказоустойчивая СНС[20] являются примерами гибридных сетей, в которых некоторые части могут использовать исключительно прямую связь, в то время как другие части имеют рекуррентную топологию.

Хорошо известно, что топология СНС в головном мозге динамически изменяется в процессе обучения. В [21] было показано, что первичные сенсомоторные и зрительные области имеют относительно жесткое ядро, которое мало изменяется с течением времени, но они имеют гибкие периферические области, которые изменяются чаще. Эволюционирующая спайковая нейронная сеть (eSNN)[22], динамически эволюционирующая СНС (deSNN)[12], динамическое формирование кластеров с использованием популяций спайковых нейронов [23] являются примерами СНС с динамической топологией. Эволюционирующая структура СНС повышает их обрабатывающую способность, а также усиливает их биологическую правдоподобность.

IV. Методы обучения

Мы можем разделить СНС на два широких класса: преобразованные СНС и спайковые нейронные сети полученные путем прямого обучения на основе спайков. Первый — это СНС, преобразованные из обученных ИНС для эффективного вывода на основе событий (преобразование ИНС-СНС). Главное их преимущество заключается в том, что они используют современные (state-of-the-art), основанные на оптимизации, обучающие методы ИНС и, следовательно, достигают сравнимой производительности при классификации. Например, специализированные аппаратные средства СНС (такие, как SpiNNaker[24], IBM TrueNorth[25]) продемонстрировали значительно улучшенную энергетическую эффективность относительно ИНС, а также современную производительность для вывода. Сигналы, используемые в таком обучении, имеют реальное значение и, естественно, могут быть рассмотрены как представляющие скорость спайков (частоту). Проблема заключается в том, что для надежной оценки частот требуется нетривиальный проход по временной шкале. С другой стороны,

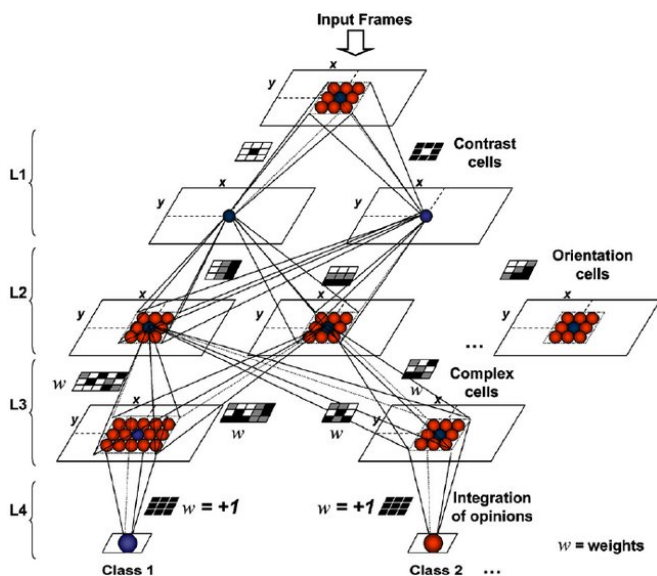


Рис. 1: Четырехслойная архитектура СНС предложенная в [22]. Нейроны в L1 и L2 чувствительны к контрасту изображения и ориентации соответственно. L3 имеет сложные клетки, обученные реагировать на определенные паттерны. L4 накапливает мнения по различным входным возбуждениям во времени.

СНС, полученные путем прямого обучения на основе спайков, также имеют некоторые проблемы. Методы прямого обучения на основе спайков можно разделить на два класса: не основанные на оптимизации, в основном неконтролируемые подходы, включающие только сигналы локальные для синапса (например, время пре- и пост-синаптических спайков, как в случае с пластичностью зависящей от времени спайка) и основанные на оптимизации, в основном контролируемые подходы, включающие глобальную цель, например, минимизация функции потерь.

A. Пластичность зависящая от времени спайка

Для обучения без учителя часто используется *пластичность зависящая от времени спайка* (spike-time-dependent plasticity, STDP)[26, 27]. В данном подходе веса на связанных (пре- и пост-синаптических) нейронах корректируются в зависимости от относительного времени спайка в пределах короткого интервала времени (десятки миллисекунд). Если пре-синаптический нейрон отправляет спайк непосредственно перед тем как пост-синаптический нейрон отправляет свой спайк далее, то вес данной связи увеличивается. Если же наоборот – пре-синаптический нейрон отправляет спайк сразу же после пост-синаптического нейрона, то данная синаптическая связь считается побочной, и её вес уменьшается.

Данное правило учитывает каждую пару нейронов (соединённых синапсом) по отдельности и действует в пределах короткого промежутка времени. Таким образом это обучающее правило является локальным в двух смыслах – как для отдельного синапса, так и для временного промежутка. Использование локального обучающего правила очень привлекательно для практических применений СНС, поскольку оно позволяет более эффективно использовать аппаратные ресурсы[28, 29]. Кроме

того данный подход естественным образом позволяет распознавать пространственно-временные паттерны.

В [30] было показано, что обученная STDP двухслойная сеть (состоящая из 6400 выходных нейронов) достигает 95% точности классификации по набору данных MNIST. Однако главная проблема использования исключительно локальных обучающих правил глубоких сетей состоит в сложности осуществления обратного прохода при обучении на сигнале ошибки. Ошибочный сигнал может быть доступен только на выходном слое сети, при этом поток информации в биологическом синапсе является однонаправленным. В таких условиях остаётся неясным то, каким образом информация об ошибке достигает дальних слоёв. Типичные архитектуры нейронных сетей с прямой связью, которые используются в машинном обучении не способны предоставить синапсам достаточную информацию для обучения при использовании локальных обучающих правил. По этому большинство исследований локального обучения спайковых сетей вводят рекуррентные обратные связи которые регулируют обучение нижних слоёв.

B. Метод обратного распространения

В то время как эффективное извлечение признаков было продемонстрировано с помощью послонного обучения STDP в глубокие сверточные СНС, модели ИНС, обученные с помощью методов стандартного обратного распространения (backpropagation, BP), всё ещё достигают значительно более высокой эффективности классификации. Эти соображения вдохновили исследователей на поиск спайковых версий BP, которые требуют поиска дифференцируемой аппроксимации функции активации спайкового нейрона.

В [31] используется иной метод оценки производной потенциала на мембране. В качестве основы метода взята модель спайкового нейрона с утечкой потенциала (Leaky-integrate-and-fire neuron, LIF neuron). В данной модели нейрон накапливает потенциал мембране за счёт полученных на вход спайков. В случае если накопленный потенциал достигает порогового значения, нейрон выпускает спайк и его потенциал опускается до базового (обычно нулевого) уровня. При этом со временем накопленный потенциал уменьшается («утекает»), пока не достигнет базового уровня. В [31] данный процесс описывается уравнениями:

$$\tau_m \frac{dV_{mem}}{dt} = -V_{mem} + I(t) \quad (1)$$

$$I(t) = \sum_{i=1}^{n^l} \left(w_i * \sum_k (\theta_i(t - t_k)) \right) \quad (2)$$

$$\theta_i(t - t_k) = \begin{cases} 1 & , \text{ если } t = t_k \\ 0 & , \text{ иначе} \end{cases} \quad (3)$$

где V_{mem} - потенциал нейрона, τ_m - показатель утечки потенциала, I_t - взвешенная сумма входных спайков, n^l - число нейронов, t_k - время спайка.

При прямом проходе, группы спайков, представляющие входные паттерны, предоставляются сети для оценки вывода сети. Для генерации спайковых входов значения входных пикселей преобразуются в распределенные

по Пуассону группы спайков и подаются в сеть. Входные спайки умножаются на синаптические веса для получения входного тока, который накапливается в мембранном потенциале постнейронов, как в уравнении 1. Всякий раз, когда его мембранный потенциал превышает порог срабатывания нейрона, постнейрон генерирует выходной спайк и сбрасывает потенциал. В противном случае мембранный потенциал со временем экспоненциально распадается. Нейроны каждого слоя (исключая выходной слой) выполняют этот процесс последовательно, основываясь на взвешенных спайках, полученных от предыдущего слоя.

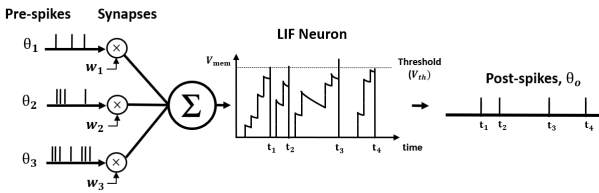


Рис. 2: Схема работы нейрона с утечкой

С течением времени общее взвешенное суммирование входных спайковых групп перед спайком описывается следующим образом:

$$net_j^l(t) = \sum_{i=1}^{n^{l-1}} (w_{ij}^{l-1} x_i^{l-1}(t)) \quad (4)$$

$$x_i^{l-1}(t) = \sum_t \sum_k \theta_i^{(l-1)}(t - t_k) \quad (5)$$

При обратном проходе, утечка в финальном слое рассматривается как шум. Таким образом потенциал на выходном слое оценивается как взвешенная сумма пришедших спайков. Для оценки производной функции активации нейрона в скрытых слоях сначала оценивается функция активации нейрона без утечки как линейная функция. Затем, для отражения эффекта утечки, к производной нейрона без утечки применяется функция, которая масштабирует производную, а также выступает как фильтр низких частот.

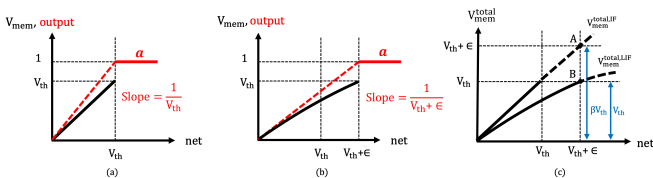


Рис. 3: Оценка функции активации спайкового нейрона

Функция выражена следующим образом:

$$f(t) = \sum_k \exp\left(-\frac{t - t_k}{\tau_m}\right) \quad (6)$$

В результате производная функции активации выражается следующим образом:

$$\frac{\partial a_{LIF}}{\partial net} = \frac{1}{V_{th}} \left(1 + \frac{1}{\gamma} \sum_k -\frac{1}{\tau_m} e^{-\frac{t-t_k}{\tau_m}}\right) \quad (7)$$

Применив данную оценку для вычисления градиента и используя градиентный спуск, исследователи построили сеть с топологией схожей с сетями VGG и ResNet. Получившаяся сеть показывает превосходящую точность

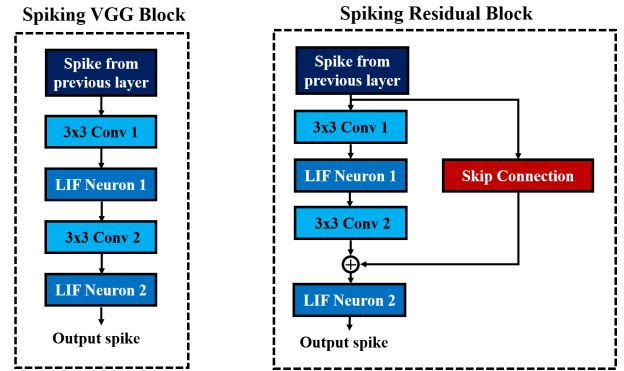


Рис. 4: Спайковые аналоги блоков VGG и ResNet

на таких наборах данных как MNIST, N-MNIST, SVHN, CIFAR-10, по сравнению с различными видами спайковых сетей (другими алгоритмами обратного прохода, сконвертированными классическими нейронными сетями, алгоритмами на основе синаптической пластичности и гибридами всех вышеперечисленных).

Таким образом предложенный в [31] метод существенно уменьшает различия в эффективности между классическим ИНС и спайковыми нейронными сетями. Предложенный метод также существенно сокращает количество требуемых ресурсов как для обучения, так и для работы СНС.

С. Конвертирование ИНС

Существует отдельная группа глубоких спайковых нейронных сетей, которые сконструированы непосредственно из глубоких нейронных сетей (Deep Neural Network, DNN). В этой группе, сначала обучается классическая DNN, состоящая из нейронов с непрерывными значениями активации. Затем эта классическая DNN преобразуется в глубокую спайковую нейронную сеть [34]. С помощью такого подхода state-of-the-art методы обучения DNN могут быть использованы для создания спайковой сети со сравнимыми с оригинальной сетью показателями эффективности. Такое преобразование может влечь за собой потерю точности. Для того чтобы уменьшить потерю точности, используются различные техники такие, как введение дополнительных ограничений на активацию спайкового нейрона или на параметры всей сети, пересчет весов, добавление шума или использование вероятностных весов.

Сгенерированные таким образом спайковые сети требуют большого количества шагов, для того чтобы произвести input-output mapping. Кроме того, такие сети не способны отражать временную динамику данных [35]. Результаты сравнительного исследования множества глубоких спайковых нейронных сетей в [27] исследователи показывают что сконвертированные сети могут иметь большую точность на данных MNIST по сравнению с непосредственно обученными спайковыми сетями. Для

Таблица I: Сравнение [31] с предыдущими результатами

Данные	Модель	ИНС	Сконвертированная СНС[10]	Предыдущая лучшая	Работа Lee et al.
MNIST	LeNet	99.57%	99.59%	99.49%[32]	99.59%
N-MNIST	LeNet	-	-	99.53%[33]	99.59%
SVHN	VGG7	96.36%	96.30%	-	96.06
	ResNet7	96.43%	96.40%	-	96.21%
CIFAR-10	VGG9	91.98%	92.01%	90.53%[33]	90.45%
	ResNet9	91.85%	21.59%		90.35%
	ResNet11	91.87%	90.65%		90.95%

создания аналога вывода классических нейронов, сконвертированные спайковые сети обычно используют частотное кодирование выходных спайков. Однако частотное кодирование может скрывать временную информацию, которая может быть использована спайковой нейронной сетью[36]

V. Аппаратное обеспечение

Успехи, достигнутые в области компьютерного моделирования СНС, продемонстрированных Neurogrid[37], BrainScaleS[38], Intel Loihi[39], IBM TrueNorth[25], Tianjic[40], свидетельствуют о большом потенциале аппаратной реализации спайковых нейронных сетей.

В [41] исследователи перемоделировали масштабные сверточные нейронные сети, с целью того чтобы обойти ограничения оборудования и запустить их на IBM TrueNorth. Была продемонстрирована свёрточная нейронная сеть, которая использовалась для обнаружения и подсчета автомобилей, с сопоставимой точностью по сравнению с обученной при помощи графического процессора классической свёрточной сетью, но с гораздо меньшим потреблением энергии. Bohnstingl et al. разработали спайковую сеть которая «учится учиться» на нейроморфном чипе, который ускоряет процесс обучения, извлекая абстрактные знания из предыдущего опыта. Помимо обычных CMOS-схем, в [42] также были изучены новые устройства такие, как мемристоры (memristors).

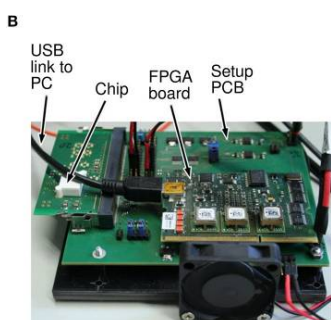


Рис. 5: Измерительная установка и прототипная плата разработанная в [42]. На плате изображен сам нейроморфный чип, интерфейс к главному компьютеру и вспомогательная плата FPGA.

В [43] был предложен основанный на спайковой пластичности алгоритм жадного обучения для спайковых нейронных сетей для снижения уровней веса и повышения устойчивости к неидеальности устройств. Авторы демонстрируют онлайн-обучение на резистивной системе оперативной памяти (RRAM) с неидеальным поведением.

VI. Программное обеспечение

Инструменты программирования были одним из ключевых компонентов, стимулирующих развитие в исследованиях ИНС, примерами которых являются TensorFlow[44], PyTorch[45], Keras. Эти удобные инструменты программирования позволяют исследователям создавать и обучать крупномасштабные нейронные сети, используя только базовые программные методы. Для сравнения, инструменты программирования для СНС довольно ограничены. Такие инструменты как SpiNNaker[24], BindsNET[46], и PyNN[47] обеспечивают базовый программный интерфейс для поддержки простых и небольших имитаций СНС. Как правило, исследователям приходится создавать СНС с нуля, что может занять много времени и потребовать значительно большего количества программных навыков. Таким образом, разработка удобных для пользователя программных средств для эффективного развертывания крупномасштабных СНС является обязательным условием для развития данной области. В [48] была предложена высокоскоростная платформа моделирования СНС с открытым исходным кодом, основанная на PyTorch. SpykeTorch имитирует сверточные СНС с не более чем одним спайком на нейрон (схема кодирования рангового порядка) и основанные на STDP правила обучения.

Nengo[49] – это основанный на Python пакет нейронного моделирования, который использует небольшой набор простых объектов для создания функциональных спайковых нейронных сетей. Эти объекты и их основные модели поведения основаны на Neural Engineering Framework[50], но также способны реализовывать обычные спайковые нейронные сети. Объектами являются Ансамбли (группа нейронов), Узлы (не нейронные компоненты, используемые для управления абстрактными компонентами, такими как роботы-манипуляторы), Связи (связи между любыми комбинациями Ансамблей и Узлов) и Сети(группы всех вышеупомянутых объектов). Для помощи в проектировании и тестировании сетей Nengo был создан графический пользовательский интерфейс (GUI) с сетевым визуализатором и средой программирования.

Nengo позволяет не только производить симуляции спайковых нейронных сетей, но и развертывать данные сети на различных нейроморфных вычислителях. Nengo имеет поддержку таких целевых платформ как FPGA, Intel Loihi, SpiNNaker, OpenCL, MPI.

VII. Заключение

Считается что СНС имеют более высокую производительность при обработке разреженной в пространстве и времени, подверженной шуму информации с высокой

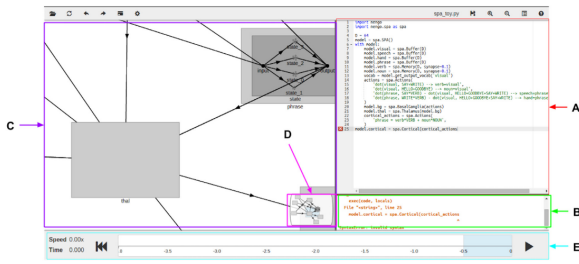


Рис. 6: Nengo графического интерфейса. А) редактор сценариев модели. (В) консоль, показывающая ошибки компиляции. (С) исследуемый сетевой визуализатор. (D) миникарта, показывающая всю сеть. Е) управление тренажером.

энергоэффективностью. В данной работе были рассмотрены типы спайковых нейронных сетей, их топологии, а также программные и аппаратные инструменты для создания и применения данного вида сетей. Кроме того были описаны основные методы обучения СНС и метод конвертирования из классических искусственных нейронных сетей.

В заключение следует отметить, что спайковые нейронные сети достигают превосходных показателей в обработке сложной, разреженной и зашумленной пространственно-временной информации с высокой энергетической эффективностью, используя нейронную динамику в событийном режиме. Событийная коммуникация особенно привлекательна для создания энергоэффективных систем искусственного интеллекта с вычислениями в памяти, которые будут играть важную роль в повсеместном применении технологии данного класса. Исследования СНС продолжаются, и можно ожидать гораздо большего прогресса в его алгоритмах обучения, структуре бенчмаркинга, инструментах программирования и эффективном оборудовании.

Список литературы

- [1] Equal numbers of neuronal and nonneuronal cells make the human brain an isometrically scaled-up primate brain / Frederico A. C. Azevedo, Ludmila R. B. Carvalho, Lea T. Grinberg et al. // *Journal of Comparative Neurology*. — 2009. — Vol. 513, no. 5. — P. 532–541. — eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1002/cne.21974>. online; accessed: <https://onlinelibrary.wiley.com/doi/abs/10.1002/cne.21974> (online; accessed: 2020-05-09).
- [2] Deep Residual Learning for Image Recognition / Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun // *arXiv:1512.03385 [cs]*. — 2015. — Dec. — *arXiv: 1512.03385*. Access mode: <http://arxiv.org/abs/1512.03385> (online; accessed: 2020-05-10).
- [3] Mastering the game of Go with deep neural networks and tree search / David Silver, Aja Huang, Chris J. Maddison et al. // *Nature*. — 2016. — Jan. — Vol. 529, no. 7587. — P. 484–489. — Number: 7587 Publisher: Nature Publishing Group. online; accessed: <https://www.nature.com/articles/nature16961> (online; accessed: 2020-05-10).
- [4] Krizhevsky A., Sutskever I., Hinton G. E. ImageNet Classification with Deep Convolutional Neural Networks // *Advances in Neural Information Processing Systems 25* / Ed. by F. Pereira, C. J. C. Burges, L. Bottou, K. Q. Weinberger. — Curran Associates, Inc., 2012. — P. 1097–1105. — Access mode: <http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf> (online; accessed: 2020-05-10).
- [5] Hinton G. E., Salakhutdinov R. R. Reducing the Dimensionality of Data with Neural Networks // *Science*. — 2006. — Jul. — Vol. 313, no. 5786. — P. 504–507. — Publisher: American Association for the Advancement of Science Section: Report. online; accessed: <https://science.sciencemag.org/content/313/5786/504> (online; accessed: 2020-05-09).
- [6] Deep Neural Networks for Acoustic Modeling in Speech Recognition: The Shared Views of Four Research Groups / Geoffrey Hinton, Li Deng, Dong Yu et al. // *IEEE Signal Processing Magazine*. — 2012. — Nov. — Vol. 29, no. 6. — P. 82–97. — Conference Name: IEEE Signal Processing Magazine.
- [7] Pham D. T., Packianather M. S., Charles E. Y. A. Control chart pattern clustering using a new self-organizing spiking neural network: // *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*. — 2008. — Oct. — Publisher: SAGE PublicationsSage UK: London, England. online; accessed: <https://journals.sagepub.com/doi/10.1243/09544054JEM1054> (online; accessed: 2020-05-09).
- [8] Maass W. Networks of spiking neurons: The third generation of neural network models // *Neural Networks*. — 1997. — Dec. — Vol. 10, no. 9. — P. 1659–1671. — online; accessed: <http://www.sciencedirect.com/science/article/pii/S0893608097000117> (online; accessed: 2019-12-17).
- [9] Izhikevich E. Simple model of spiking neurons // *IEEE Transactions on Neural Networks*. — 2003. — Nov. — Vol. 14, no. 6. — P. 1569–1572. — online; accessed: <http://ieeexplore.ieee.org/document/1257420/> (online; accessed: 2020-05-10).
- [10] Going Deeper in Spiking Neural Networks: VGG and Residual Architectures / Abhronil Sengupta, Yuting Ye, Robert Wang et al. // *arXiv:1802.02627 [cs]*. — 2019. — Feb. — *arXiv: 1802.02627*. Access mode: <http://arxiv.org/abs/1802.02627> (online; accessed: 2020-04-06).
- [11] Simulation of networks of spiking neurons: A review of tools and strategies / Romain Brette, Michelle Rudolph, Ted Carnevale et al. // *Journal of Computational Neuroscience*. — 2007. — Dec. — Vol. 23, no. 3. — P. 349–398. — online; accessed: <https://doi.org/10.1007/s10827-007-0038-6> (online; accessed: 2020-05-09).
- [12] Dynamic evolving spiking neural networks for on-line spatio- and spectro-temporal pattern recognition / Nikola Kasabov, Kshitij Dhoble, Nuttapol Nuntalid, Giacomo Indiveri // *Neural Networks*. — 2013. — May. — Vol. 41. — P. 188–201. — online; accessed: <http://www.sciencedirect.com/science/article/pii/S0893608012003139> (online; accessed: 2020-05-09).
- [13] Izhikevich E. M. Polychronization: Computation with Spikes // *Neural Computation*. — 2006. — Feb. — Vol. 18, no. 2. — P. 245–282. — Publisher: MIT Press. Access mode: <https://www.mitpressjournals.org/doi/10.1162/089976606775093882> (online; accessed: 2020-05-09).

- [14] Pfeiffer M., Pfeil T. Deep Learning With Spiking Neurons: Opportunities and Challenges // *Frontiers in Neuroscience*. — 2018. — Oct. — Vol. 12. — P. 774. — Access mode: <https://www.frontiersin.org/article/10.3389/fnins.2018.00774/full> (online; accessed: 2020-03-16).
- [15] A review of learning in biologically plausible spiking neural networks / Aboozar Taherkhani, Ammar Belatreche, Yuhua Li et al. // *Neural Networks*. — 2020. — Feb. — Vol. 122. — P. 253–272. — online; accessed: <http://www.sciencedirect.com/science/article/pii/S0893608019303181> (online; accessed: 2020-01-17).
- [16] Masquelier T., Deco G. Learning and Coding in Neural Networks // *Principles of Neural Coding*. — CRC Press, 2013. — May. — P. 513–526. — online; accessed: <http://www.crcnetbase.com/doi/10.1201/b14756-30> (online; accessed: 2020-05-09).
- [17] Brette R. Philosophy of the Spike: Rate-Based vs. Spike-Based Theories of the Brain // *Frontiers in Systems Neuroscience*. — 2015. — Vol. 9. — Publisher: Frontiers. online; accessed: <https://www.frontiersin.org/articles/10.3389/fnsys.2015.00151/full> (online; accessed: 2020-05-09).
- [18] Paugam-Moisy H., Bohte S. Computing with Spiking Neuron Networks // *Handbook of Natural Computing* / Ed. by Grzegorz Rozenberg, Thomas Bäck, Joost N. Kok. — Berlin, Heidelberg : Springer, 2012. — P. 335–376. — online; accessed: https://doi.org/10.1007/978-3-540-92910-9_10 (online; accessed: 2020-05-10).
- [19] Abeles M. Synfire chains // *Scholarpedia*. — 2009. — Jul. — Vol. 4, no. 7. — P. 1441. — online; accessed: http://www.scholarpedia.org/article/Synfire_chains (online; accessed: 2020-05-09).
- [20] Srinivasa N., Cho Y. Self-Organizing Spiking Neural Model for Learning Fault-Tolerant Spatio-Motor Transformations // *IEEE Transactions on Neural Networks and Learning Systems*. — 2012. — Oct. — Vol. 23, no. 10. — P. 1526–1538. — Conference Name: IEEE Transactions on Neural Networks and Learning Systems.
- [21] Task-Based Core-Periphery Organisation of Human Brain Dynamics / Danielle S. Bassett, Nicholas F. Wymbs, M. Puck Rombach et al. // *arXiv:1210.3555 [cond-mat, physics:nlin, q-bio, stat]*. — 2013. — Oct. — *arXiv:1210.3555*. Access mode: <http://arxiv.org/abs/1210.3555> (online; accessed: 2020-05-10).
- [22] Wysocki S. G., Benuskova L., Kasabov N. Fast and adaptive network of spiking neurons for multi-view visual pattern recognition // *Neurocomputing*. — 2008. — Aug. — Vol. 71, no. 13. — P. 2563–2575. — online; accessed: <http://www.sciencedirect.com/science/article/pii/S0925231208002191> (online; accessed: 2020-05-10).
- [23] Belatreche A., Paul R. Dynamic cluster formation using populations of spiking neurons // *Unknown Host Publication*. — 2012. — Jul. — P. 1–6. — online; accessed: <https://pure.ulster.ac.uk/en/publications/dynamic-cluster-formation-using-populations-of-spiking-neurons-3> (online; accessed: 2020-05-10).
- [24] The SpiNNaker Project / Steve B. Furber, Francesco Galluppi, Steve Temple, Luis A. Plana // *Proceedings of the IEEE*. — 2014. — May. — Vol. 102, no. 5. — P. 652–665. — Conference Name: Proceedings of the IEEE.
- [25] TrueNorth: Design and Tool Flow of a 65 mW 1 Million Neuron Programmable Neurosynaptic Chip / Filipp Akopyan, Jun Sawada, Andrew Cassidy et al. // *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*. — 2015. — Oct. — Vol. 34, no. 10. — P. 1537–1557. — Conference Name: IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems.
- [26] Caporale N., Dan Y. Spike Timing–Dependent Plasticity: A Hebbian Learning Rule // *Annual Review of Neuroscience*. — 2008. — Jul. — Vol. 31, no. 1. — P. 25–46. — online; accessed: <http://www.annualreviews.org/doi/10.1146/annurev.neuro.31.060407.125639> (online; accessed: 2020-03-15).
- [27] Deep learning in spiking neural networks / Amirhossein Tavanaei, Masoud Ghodrati, Saeed Reza Kheradpisheh et al. // *Neural Networks*. — 2019. — Mar. — Vol. 111. — P. 47–63. — online; accessed: <http://www.sciencedirect.com/science/article/pii/S0893608018303332> (online; accessed: 2020-04-28).
- [28] Kheradpisheh S. R., Ganjtabesh M., Masquelier T. Bio-inspired unsupervised learning of visual features leads to robust invariant object recognition // *Neurocomputing*. — 2016. — Sep. — Vol. 205. — P. 382–392. — online; accessed: <http://www.sciencedirect.com/science/article/pii/S0925231216302880> (online; accessed: 2020-04-30).
- [29] Tavanaei A., Maida A. S. A spiking network that learns to extract spike signatures from speech signals // *Neurocomputing*. — 2017. — May. — Vol. 240. — P. 191–199. — online; accessed: <http://www.sciencedirect.com/science/article/pii/S0925231217303119> (online; accessed: 2020-04-30).
- [30] Diehl P. U., Cook M. Unsupervised learning of digit recognition using spike-timing-dependent plasticity // *Frontiers in Computational Neuroscience*. — 2015. — Vol. 9. — Publisher: Frontiers. online; accessed: <https://www.frontiersin.org/articles/10.3389/fncom.2015.00099/full> (online; accessed: 2020-05-10).
- [31] Lee C., Sarwar S. S., Roy K. Enabling Spike-based Backpropagation in State-of-the-art Deep Neural Network Architectures // *arXiv:1903.06379 [cs]*. — 2019. — Aug. — *arXiv:1903.06379*. Access mode: <http://arxiv.org/abs/1903.06379> (online; accessed: 2019-12-22).
- [32] Jin Y., Zhang W., Li P. Hybrid Macro/Micro Level Backpropagation for Training Deep Spiking Neural Networks // *Advances in Neural Information Processing Systems 31* / Ed. by S. Bengio, H. Wallach, H. Larochelle et al. — Curran Associates, Inc., 2018. — P. 7005–7015. — Access mode: <http://papers.nips.cc/paper/7932-hybrid-macro-micro-level-backpropagation-for-training-deep-spiking-neural-networks.pdf> (online; accessed: 2020-05-10).
- [33] Direct Training for Spiking Neural Networks: Faster, Larger, Better / Yujie Wu, Lei Deng, Guoqi Li et al. // *arXiv:1809.05793 [cs]*. — 2018. — Nov. — *arXiv:1809.05793*. Access mode: <http://arxiv.org/abs/1809.05793> (online; accessed: 2020-05-10).
- [34] Rueckauer B., Liu S.-C. Conversion of analog to spiking neural networks using sparse temporal coding // *2018 IEEE International Symposium on Circuits and Systems (ISCAS)*. — 2018. — May. — P. 1–5. — ISSN: 2379-447X.

- [35] A Comprehensive Analysis on Adversarial Robustness of Spiking Neural Networks / Saima Sharmin, Priyadarshini Panda, Syed Shakib Sarwar et al. // 2019 International Joint Conference on Neural Networks (IJCNN). — 2019. — Jul. — P. 1–8. — ISSN: 2161-4407.
- [36] Lee J. H., Delbruck T., Pfeiffer M. Training Deep Spiking Neural Networks Using Backpropagation // *Frontiers in Neuroscience*. — 2016. — Vol. 10. — online; accessed: <https://www.frontiersin.org/articles/10.3389/fnins.2016.00508/full> (online; accessed: 2020-04-06).
- [37] Neurogrid: A Mixed-Analog-Digital Multichip System for Large-Scale Neural Simulations / Ben Varkey Benjamin, Peiran Gao, Emmett McQuinn et al. // *Proceedings of the IEEE*. — 2014. — May. — Vol. 102, no. 5. — P. 699–716. — Conference Name: Proceedings of the IEEE.
- [38] Live demonstration: A scaled-down version of the BrainScaleS wafer-scale neuromorphic system / Johannes Schemmel, Andreas Grübl, Stephan Hartmann et al. // 2012 IEEE International Symposium on Circuits and Systems (ISCAS). — 2012. — May. — P. 702–702. — ISSN: 2158-1525.
- [39] Loihi: A Neuromorphic Manycore Processor with On-Chip Learning / Mike Davies, Narayan Srinivasa, Tsung-Han Lin et al. // *IEEE Micro*. — 2018. — Jan. — Vol. 38, no. 1. — P. 82–99. — Conference Name: IEEE Micro.
- [40] Towards artificial general intelligence with hybrid Tianjic chip architecture / Jing Pei, Lei Deng, Sen Song et al. // *Nature*. — 2019. — Aug. — Vol. 572, no. 7767. — P. 106–111. — Number: 7767 Publisher: Nature Publishing Group. online; accessed: <https://www.nature.com/articles/s41586-019-1424-8> (online; accessed: 2020-05-06).
- [41] REMODEL: Rethinking Deep CNN Models to Detect and Count on a NeuroSynaptic System / Rohit Shukla, Mikko Lipasti, Brian Van Essen et al. // *Frontiers in Neuroscience*. — 2019. — Feb. — Vol. 13. — P. 4. — Access mode: <https://www.frontiersin.org/article/10.3389/fnins.2019.00004/full> (online; accessed: 2020-04-06).
- [42] Neuromorphic Hardware Learns to Learn / Thomas Bohnstingl, Franz Scherr, Christian Pehle et al. // *Frontiers in Neuroscience*. — 2019. — May. — Vol. 13. — Access mode: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6536858/> (online; accessed: 2020-05-10).
- [43] Unsupervised Learning on Resistive Memory Array Based Spiking Neural Networks / Yilong Guo, Huaqiang Wu, Bin Gao, He Qian // *Frontiers in Neuroscience*. — 2019. — Vol. 13. — Publisher: Frontiers. online; accessed: <https://www.frontiersin.org/articles/10.3389/fnins.2019.00812/full> (online; accessed: 2020-05-10).
- [44] TensorFlow: A system for large-scale machine learning / Martin Abadi, Paul Barham, Jianmin Chen et al. // 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16). — 2016. — P. 265–283. — Access mode: <https://www.usenix.org/system/files/conference/osdi16/osdi16-abadi.pdf> (online; accessed: 2020-05-10).
- [45] PyTorch: An Imperative Style, High-Performance Deep Learning Library / Adam Paszke, Sam Gross, Francisco Massa et al. // *Advances in Neural Information Processing Systems 32* / Ed. by H. Wallach, H. Larochelle, A. Beygelzimer et al. — Curran Associates, Inc., 2019. — P. 8026–8037. — Access mode: <http://papers.nips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf> (online; accessed: 2020-05-10).
- [46] BindsNET: A Machine Learning-Oriented Spiking Neural Networks Library in Python / Hananel Hazan, Daniel J. Saunders, Hassaan Khan et al. // *Frontiers in Neuroinformatics*. — 2018. — Vol. 12. — Publisher: Frontiers. online; accessed: <https://www.frontiersin.org/articles/10.3389/fninf.2018.00089/full> (online; accessed: 2020-05-10).
- [47] PyNN: a common interface for neuronal network simulators / Andrew P. Davison, Daniel Brüderle, Jochen M. Eppler et al. // *Frontiers in Neuroinformatics*. — 2009. — Vol. 2. — Publisher: Frontiers. online; accessed: <https://www.frontiersin.org/articles/10.3389/neuro.11.011.2008/full> (online; accessed: 2020-05-10).
- [48] SpykeTorch: Efficient Simulation of Convolutional Spiking Neural Networks With at Most One Spike per Neuron / Milad Mozafari, Mohammad Ganjtabesh, Abbas Nowzari-Dalini, Timothée Masquelier // *Frontiers in Neuroscience*. — 2019. — Jul. — Vol. 13. — Access mode: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6640212/> (online; accessed: 2020-05-10).
- [49] Nengo: a Python tool for building large-scale functional brain models / Trevor Bekolay, James Bergstra, Eric Hunsberger et al. // *Frontiers in Neuroinformatics*. — 2014. — Vol. 7. — online; accessed: <https://www.frontiersin.org/articles/10.3389/fninf.2013.00048/full> (online; accessed: 2020-03-16).
- [50] A Large-Scale Model of the Functioning Brain / Chris Eliasmith, Terrence C. Stewart, Xuan Choo et al. // *Science*. — 2012. — Nov. — Vol. 338, no. 6111. — P. 1202–1205. — Publisher: American Association for the Advancement of Science Section: Report. online; accessed: <https://science.sciencemag.org/content/338/6111/1202> (online; accessed: 2020-05-10).

On Spiking Neural Networks

Vladimir Evgrafov, Eugene Ilyushin

Abstract—Over the past few years, deep learning methods have made significant progress and have become widespread tools for solving various cognitive tasks. To leverage the power of deep learning everywhere, you need to deploy deep learning not only to large-scale computers but also to peripherals. However, the ever-growing complexity of deep neural networks, coupled with a dramatic increase in the amount of data processed, place significant energy demands on modern computing platforms. The neuromorphic computing model assumes that computations are performed in a biologically plausible way. Part of neuromorphic computing is spike neural networks, which are one of the leading candidates for overcoming the limitations of neural computing and effectively using machine learning algorithms in real-world applications. This paper discusses the biological foundations of spike neural networks, methods for training and creating them, as well as software and hardware platforms for their use.

Keywords—spike neural networks, neuromorphic computing, artificial intelligence

References

- [1] Equal numbers of neuronal and nonneuronal cells make the human brain an isometrically scaled-up primate brain / Frederico A. C. Azevedo, Ludmila R. B. Carvalho, Lea T. Grinberg et al. // *Journal of Comparative Neurology*. — 2009. — Vol. 513, no. 5. — P. 532–541. — eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1002/cne.21974>. online; accessed: <https://onlinelibrary.wiley.com/doi/abs/10.1002/cne.21974> (online; accessed: 2020-05-09).
- [2] Deep Residual Learning for Image Recognition / Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun // *arXiv:1512.03385 [cs]*. — 2015. — Dec. — *arXiv: 1512.03385*. Access mode: <http://arxiv.org/abs/1512.03385> (online; accessed: 2020-05-10).
- [3] Mastering the game of Go with deep neural networks and tree search / David Silver, Aja Huang, Chris J. Maddison et al. // *Nature*. — 2016. — Jan. — Vol. 529, no. 7587. — P. 484–489. — Number: 7587 Publisher: Nature Publishing Group. online; accessed: <https://www.nature.com/articles/nature16961> (online; accessed: 2020-05-10).
- [4] Krizhevsky A., Sutskever I., Hinton G. E. ImageNet Classification with Deep Convolutional Neural Networks // *Advances in Neural Information Processing Systems 25* / Ed. by F. Pereira, C. J. C. Burges, L. Bottou, K. Q. Weinberger. — Curran Associates, Inc., 2012. — P. 1097–1105. — Access mode: <http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf> (online; accessed: 2020-05-10).
- [5] Hinton G. E., Salakhutdinov R. R. Reducing the Dimensionality of Data with Neural Networks // *Science*. — 2006. — Jul. — Vol. 313, no. 5786. — P. 504–507. — Publisher: American Association for the Advancement of Science Section: Report. online; accessed: <https://science.sciencemag.org/content/313/5786/504> (online; accessed: 2020-05-09).
- [6] Deep Neural Networks for Acoustic Modeling in Speech Recognition: The Shared Views of Four Research Groups / Geoffrey Hinton, Li Deng, Dong Yu et al. // *IEEE Signal Processing Magazine*. — 2012. — Nov. — Vol. 29, no. 6. — P. 82–97. — Conference Name: IEEE Signal Processing Magazine.
- [7] Pham D. T., Packianather M. S., Charles E. Y. A. Control chart pattern clustering using a new self-organizing spiking neural network: // *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*. — 2008. — Oct. — Publisher: SAGE PublicationsSage UK: London, England. online; accessed: <https://journals.sagepub.com/doi/10.1243/09544054JEM1054> (online; accessed: 2020-05-09).
- [8] Maass W. Networks of spiking neurons: The third generation of neural network models // *Neural Networks*. — 1997. — Dec. — Vol. 10, no. 9. — P. 1659–1671. — online; accessed: <http://www.sciencedirect.com/science/article/pii/S0893608097000117> (online; accessed: 2019-12-17).
- [9] Izhikevich E. Simple model of spiking neurons // *IEEE Transactions on Neural Networks*. — 2003. — Nov. — Vol. 14, no. 6. — P. 1569–1572. — online; accessed: <http://ieeexplore.ieee.org/document/1257420/> (online; accessed: 2020-05-10).
- [10] Going Deeper in Spiking Neural Networks: VGG and Residual Architectures / Abhronil Sengupta, Yuting Ye, Robert Wang et al. // *arXiv:1802.02627 [cs]*. — 2019. — Feb. — *arXiv: 1802.02627*. Access mode: <http://arxiv.org/abs/1802.02627> (online; accessed: 2020-04-06).
- [11] Simulation of networks of spiking neurons: A review of tools and strategies / Romain Brette, Michelle Rudolph, Ted Carnevale et al. // *Journal of Computational Neuroscience*. — 2007. — Dec. — Vol. 23, no. 3. — P. 349–398. — online; accessed: <https://doi.org/10.1007/s10827-007-0038-6> (online; accessed: 2020-05-09).
- [12] Dynamic evolving spiking neural networks for on-line spatio- and spectro-temporal pattern recognition / Nikola Kasabov, Kshitij Dhoble, Nuttapod Nuntalid, Giacomo Indiveri // *Neural Networks*. — 2013. — May. — Vol. 41. — P. 188–201. — online; accessed: <http://www.sciencedirect.com/science/article/pii/S0893608012003139> (online; accessed: 2020-05-09).
- [13] Izhikevich E. M. Polychronization: Computation with Spikes // *Neural Computation*. — 2006. — Feb. — Vol. 18, no. 2. — P. 245–282. — Publisher: MIT Press. Access mode: <https://www.mitpressjournals.org/doi/10.1162/089976606775093882> (online; accessed: 2020-05-09).
- [14] Pfeiffer M., Pfeil T. Deep Learning With Spiking Neurons: Opportunities and Challenges // *Frontiers in*

- Neuroscience. — 2018. — Oct. — Vol. 12. — P. 774. — Access mode: <https://www.frontiersin.org/article/10.3389/fnins.2018.00774/full> (online; accessed: 2020-03-16).
- [15] A review of learning in biologically plausible spiking neural networks / Aboozar Taherkhani, Ammar Belatreche, Yuhua Li et al. // *Neural Networks*. — 2020. — Feb. — Vol. 122. — P. 253–272. — online; accessed: <http://www.sciencedirect.com/science/article/pii/S0893608019303181> (online; accessed: 2020-01-17).
- [16] Masquelier T., Deco G. Learning and Coding in Neural Networks // *Principles of Neural Coding*. — CRC Press, 2013. — May. — P. 513–526. — online; accessed: <http://www.crcnetbase.com/doi/10.1201/b14756-30> (online; accessed: 2020-05-09).
- [17] Brette R. Philosophy of the Spike: Rate-Based vs. Spike-Based Theories of the Brain // *Frontiers in Systems Neuroscience*. — 2015. — Vol. 9. — Publisher: Frontiers. online; accessed: <https://www.frontiersin.org/articles/10.3389/fnsys.2015.00151/full> (online; accessed: 2020-05-09).
- [18] Paugam-Moisy H., Bohte S. Computing with Spiking Neuron Networks // *Handbook of Natural Computing* / Ed. by Grzegorz Rozenberg, Thomas Bäck, Joost N. Kok. — Berlin, Heidelberg : Springer, 2012. — P. 335–376. — online; accessed: https://doi.org/10.1007/978-3-540-92910-9_10 (online; accessed: 2020-05-10).
- [19] Abeles M. Synfire chains // *Scholarpedia*. — 2009. — Jul. — Vol. 4, no. 7. — P. 1441. — online; accessed: http://www.scholarpedia.org/article/Synfire_chains (online; accessed: 2020-05-09).
- [20] Srinivasa N., Cho Y. Self-Organizing Spiking Neural Model for Learning Fault-Tolerant Spatio-Motor Transformations // *IEEE Transactions on Neural Networks and Learning Systems*. — 2012. — Oct. — Vol. 23, no. 10. — P. 1526–1538. — Conference Name: IEEE Transactions on Neural Networks and Learning Systems.
- [21] Task-Based Core-Periphery Organisation of Human Brain Dynamics / Danielle S. Bassett, Nicholas F. Wymbs, M. Puck Rombach et al. // *arXiv:1210.3555 [cond-mat, physics:nlin, q-bio, stat]*. — 2013. — Oct. — *arXiv:1210.3555*. Access mode: <http://arxiv.org/abs/1210.3555> (online; accessed: 2020-05-10).
- [22] Wysocki S. G., Benuskova L., Kasabov N. Fast and adaptive network of spiking neurons for multi-view visual pattern recognition // *Neurocomputing*. — 2008. — Aug. — Vol. 71, no. 13. — P. 2563–2575. — online; accessed: <http://www.sciencedirect.com/science/article/pii/S0925231208002191> (online; accessed: 2020-05-10).
- [23] Belatreche A., Paul R. Dynamic cluster formation using populations of spiking neurons // *Unknown Host Publication*. — 2012. — Jul. — P. 1–6. — online; accessed: <https://pure.ulster.ac.uk/en/publications/dynamic-cluster-formation-using-populations-of-spiking-neurons-3> (online; accessed: 2020-05-10).
- [24] The SpiNNaker Project / Steve B. Furber, Francesco Galluppi, Steve Temple, Luis A. Plana // *Proceedings of the IEEE*. — 2014. — May. — Vol. 102, no. 5. — P. 652–665. — Conference Name: Proceedings of the IEEE.
- [25] TrueNorth: Design and Tool Flow of a 65 mW 1 Million Neuron Programmable Neurosynaptic Chip / Filipp Akopyan, Jun Sawada, Andrew Cassidy et al. // *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*. — 2015. — Oct. — Vol. 34, no. 10. — P. 1537–1557. — Conference Name: IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems.
- [26] Caporale N., Dan Y. Spike Timing-Dependent Plasticity: A Hebbian Learning Rule // *Annual Review of Neuroscience*. — 2008. — Jul. — Vol. 31, no. 1. — P. 25–46. — online; accessed: <http://www.annualreviews.org/doi/10.1146/annurev.neuro.31.060407.125639> (online; accessed: 2020-03-15).
- [27] Deep learning in spiking neural networks / Amirhossein Tavanaei, Masoud Ghodrati, Saeed Reza Kheradpisheh et al. // *Neural Networks*. — 2019. — Mar. — Vol. 111. — P. 47–63. — online; accessed: <http://www.sciencedirect.com/science/article/pii/S0893608018303332> (online; accessed: 2020-04-28).
- [28] Kheradpisheh S. R., Ganjtabesh M., Masquelier T. Bio-inspired unsupervised learning of visual features leads to robust invariant object recognition // *Neurocomputing*. — 2016. — Sep. — Vol. 205. — P. 382–392. — online; accessed: <http://www.sciencedirect.com/science/article/pii/S0925231216302880> (online; accessed: 2020-04-30).
- [29] Tavanaei A., Maida A. S. A spiking network that learns to extract spike signatures from speech signals // *Neurocomputing*. — 2017. — May. — Vol. 240. — P. 191–199. — online; accessed: <http://www.sciencedirect.com/science/article/pii/S0925231217303119> (online; accessed: 2020-04-30).
- [30] Diehl P. U., Cook M. Unsupervised learning of digit recognition using spike-timing-dependent plasticity // *Frontiers in Computational Neuroscience*. — 2015. — Vol. 9. — Publisher: Frontiers. online; accessed: <https://www.frontiersin.org/articles/10.3389/fncom.2015.00099/full> (online; accessed: 2020-05-10).
- [31] Lee C., Sarwar S. S., Roy K. Enabling Spike-based Backpropagation in State-of-the-art Deep Neural Network Architectures // *arXiv:1903.06379 [cs]*. — 2019. — Aug. — *arXiv:1903.06379*. Access mode: <http://arxiv.org/abs/1903.06379> (online; accessed: 2019-12-22).
- [32] Jin Y., Zhang W., Li P. Hybrid Macro/Micro Level Backpropagation for Training Deep Spiking Neural Networks // *Advances in Neural Information Processing Systems 31* / Ed. by S. Bengio, H. Wallach, H. Larochelle et al. — Curran Associates, Inc., 2018. — P. 7005–7015. — Access mode: <http://papers.nips.cc/paper/7932-hybrid-macro-micro-level-backpropagation-for-training-deep-spiking-neural-networks.pdf> (online; accessed: 2020-05-10).
- [33] Direct Training for Spiking Neural Networks: Faster, Larger, Better / Yujie Wu, Lei Deng, Guoqi Li et al. // *arXiv:1809.05793 [cs]*. — 2018. — Nov. — *arXiv:1809.05793*. Access mode: <http://arxiv.org/abs/1809.05793> (online; accessed: 2020-05-10).
- [34] Rueckauer B., Liu S.-C. Conversion of analog to spiking neural networks using sparse temporal coding // *2018 IEEE International Symposium on Circuits and Systems (ISCAS)*. — 2018. — May. — P. 1–5. — ISSN: 2379-447X.
- [35] A Comprehensive Analysis on Adversarial Robustness of Spiking Neural Networks / Saima Sharmin,

- Priyadarshini Panda, Syed Shakib Sarwar et al. // 2019 International Joint Conference on Neural Networks (IJCNN). — 2019. — Jul. — P. 1–8. — ISSN: 2161-4407.
- [36] Lee J. H., Delbruck T., Pfeiffer M. Training Deep Spiking Neural Networks Using Backpropagation // *Frontiers in Neuroscience*. — 2016. — Vol. 10. — online; accessed: <https://www.frontiersin.org/articles/10.3389/fnins.2016.00508/full> (online; accessed: 2020-04-06).
- [37] Neurogrid: A Mixed-Analog-Digital Multichip System for Large-Scale Neural Simulations / Ben Varkey Benjamin, Peiran Gao, Emmett McQuinn et al. // *Proceedings of the IEEE*. — 2014. — May. — Vol. 102, no. 5. — P. 699–716. — Conference Name: Proceedings of the IEEE.
- [38] Live demonstration: A scaled-down version of the BrainScaleS wafer-scale neuromorphic system / Johannes Schemmel, Andreas Grübl, Stephan Hartmann et al. // 2012 IEEE International Symposium on Circuits and Systems (ISCAS). — 2012. — May. — P. 702–702. — ISSN: 2158-1525.
- [39] Loihi: A Neuromorphic Manycore Processor with On-Chip Learning / Mike Davies, Narayan Srinivasa, Tsung-Han Lin et al. // *IEEE Micro*. — 2018. — Jan. — Vol. 38, no. 1. — P. 82–99. — Conference Name: IEEE Micro.
- [40] Towards artificial general intelligence with hybrid Tianjic chip architecture / Jing Pei, Lei Deng, Sen Song et al. // *Nature*. — 2019. — Aug. — Vol. 572, no. 7767. — P. 106–111. — Number: 7767 Publisher: Nature Publishing Group. online; accessed: <https://www.nature.com/articles/s41586-019-1424-8> (online; accessed: 2020-05-06).
- [41] REMODEL: Rethinking Deep CNN Models to Detect and Count on a NeuroSynaptic System / Rohit Shukla, Mikko Lipasti, Brian Van Essen et al. // *Frontiers in Neuroscience*. — 2019. — Feb. — Vol. 13. — P. 4. — Access mode: <https://www.frontiersin.org/article/10.3389/fnins.2019.00004/full> (online; accessed: 2020-04-06).
- [42] Neuromorphic Hardware Learns to Learn / Thomas Bohnstingl, Franz Scherr, Christian Pehle et al. // *Frontiers in Neuroscience*. — 2019. — May. — Vol. 13. — Access mode: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6536858/> (online; accessed: 2020-05-10).
- [43] Unsupervised Learning on Resistive Memory Array Based Spiking Neural Networks / Yilong Guo, Huaqiang Wu, Bin Gao, He Qian // *Frontiers in Neuroscience*. — 2019. — Vol. 13. — Publisher: Frontiers. online; accessed: <https://www.frontiersin.org/articles/10.3389/fnins.2019.00812/full> (online; accessed: 2020-05-10).
- [44] TensorFlow: A system for large-scale machine learning / Martin Abadi, Paul Barham, Jianmin Chen et al. // 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16). — 2016. — P. 265–283. — Access mode: <https://www.usenix.org/system/files/conference/osdi16/osdi16-abadi.pdf> (online; accessed: 2020-05-10).
- [45] PyTorch: An Imperative Style, High-Performance Deep Learning Library / Adam Paszke, Sam Gross, Francisco Massa et al. // *Advances in Neural Information Processing Systems 32* / Ed. by H. Wallach, H. Larochelle, A. Beygelzimer et al. — Curran Associates, Inc., 2019. — P. 8026–8037. — Access mode: <http://papers.nips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf> (online; accessed: 2020-05-10).
- [46] BindsNET: A Machine Learning-Oriented Spiking Neural Networks Library in Python / Hananel Hazan, Daniel J. Saunders, Hassaan Khan et al. // *Frontiers in Neuroinformatics*. — 2018. — Vol. 12. — Publisher: Frontiers. online; accessed: <https://www.frontiersin.org/articles/10.3389/fninf.2018.00089/full> (online; accessed: 2020-05-10).
- [47] PyNN: a common interface for neuronal network simulators / Andrew P. Davison, Daniel Brüderle, Jochen M. Eppler et al. // *Frontiers in Neuroinformatics*. — 2009. — Vol. 2. — Publisher: Frontiers. online; accessed: <https://www.frontiersin.org/articles/10.3389/neuro.11.011.2008/full> (online; accessed: 2020-05-10).
- [48] SpykeTorch: Efficient Simulation of Convolutional Spiking Neural Networks With at Most One Spike per Neuron / Milad Mozafari, Mohammad Ganjtabesh, Abbas Nowzari-Dalini, Timothée Masquelier // *Frontiers in Neuroscience*. — 2019. — Jul. — Vol. 13. — Access mode: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6640212/> (online; accessed: 2020-05-10).
- [49] Nengo: a Python tool for building large-scale functional brain models / Trevor Bekolay, James Bergstra, Eric Hunsberger et al. // *Frontiers in Neuroinformatics*. — 2014. — Vol. 7. — online; accessed: <https://www.frontiersin.org/articles/10.3389/fninf.2013.00048/full> (online; accessed: 2020-03-16).
- [50] A Large-Scale Model of the Functioning Brain / Chris Eliasmith, Terrence C. Stewart, Xuan Choo et al. // *Science*. — 2012. — Nov. — Vol. 338, no. 6111. — P. 1202–1205. — Publisher: American Association for the Advancement of Science Section: Report. online; accessed: <https://science.sciencemag.org/content/338/6111/1202> (online; accessed: 2020-05-10).