

## СУЧАСНІ ШТУЧНІ НЕЙРОННІ МЕРЕЖІ ТА ПІДХОДИ ДО ЇХ МОДЕЛЮВАННЯ

*Розглянуті загальні особливості складних сучасних штучних нейронних мереж. Описана структура штучного нейрона та основні способи навчання нейронних мереж. На основі аналізу методів розпаралелювання штучних нейронних мереж і відповідних програмних пакетів зроблений висновок про невідповідність паралельної структури штучних нейронних мереж і архітектур сучасних комп'ютерних систем, що викликає проблеми з моделюванням складних нейронних мереж. Зроблений висновок про необхідність розробки метода розпаралелювання навчання та функціонування штучних нейронних мереж, який б мав можливість динамічно масштабуватись на слабкоз'язані гетерогенні комп'ютери в мережі Інтернет (мережа GRID).*

*Ключові слова: штучні нейронні мережі, розподілені обчислення, GRID.*

V.O. BOYCHUK, V.Y. NOVAKEVICH  
Khmelnitsk national university

### MODERN ARTIFICIAL NEURAL NETWORKS AND APPROACHES TO THEIR DESIGN

*The paper describes features of the modern artificial neural networks. The structure of the artificial neuron and basic methods of training neural networks were described. Based on the analysis of the parallel artificial neural networks and related software packages conclude were done on discrepancy parallel structure of artificial neural networks and architectures of modern computer systems, that causes problems with the complex neural networks modeling. The conclusion about the need the method of the parallel operation and training of artificial neural networks was done, which would have had the opportunity to dynamically scaled weakly heterogeneous computers on the Internet (network GRID).*

*Keywords: artificial neural networks, distributed computing, GRID.*

**Вступ.** В останнє десятиліття значно зріс інтерес до штучних нейронних мереж (ШНМ). Це пояснюється як новими теоретичними здобутками в цій галузі, так і загальним розвитком комп'ютерної техніки для їх реалізації. З найбільш знакових систем ШНМ можна назвати модель *Spaun*, яка відрізняється від інших штучних нейронних мереж тим, що знаходить вирішення поставлених завдань схожими з людським мозком способами. Інша відома математична модель мозку компанії IBM, *Blue Brain Project*, акцентована для наслідуванні просторової будови мозку і структури його внутрішніх з'єднань. Ці моделі досить складні і потребують великих обчислювальних можливостей для їх реалізації. Зокрема програмна частина моделі *Spaun* складається з 2.5 мільйонів математичних аналогів нейронів, організованих у підсистеми, які відповідають функціональним частинам головного мозку, гангліям, таламусу, передлобовій корі і т.п. Модель мозку має віртуальні органи зору і роботизовану руку, завдяки яким вона може виконувати деякі дії, на які не була здатна жодна модель мозку, створена раніше. Проект *Blue Brain Project* використовує суперкомп'ютер *Blue Gene* для моделювання колонок неокортексу. Наприкінці 2006 року вдалося змоделювати одну колонку неокортексу молодого щура. При цьому використовувався суперкомп'ютер *Blue Gene* і було задіяно 8192 процесора для моделювання 10000 нейронів.

Та незважаючи на ці локальні успіхи ШНМ ще досить далекі від повноцінної імітації людського інтелекту, для чого вони і задумувались.

Розглянемо особливості ШНМ і деякі проблеми в їх реалізації. Штучні нейронні мережі — математичні моделі, а також їхня програмна або апаратна реалізація, побудовані за принципом функціонування біологічних нейронних мереж — мереж нервових клітин живого організму. Основою цих систем виступає штучний нейрон як імітаційна модель нервової клітини мозку — біологічного нейрона.

На вхід штучного нейрона надходить деяка кількість сигналів, кожен з яких є виходом іншого нейрона. Кожен вхід множиться на відповідну вагу, аналогічну синаптичній силі, і всі результати сумуються, визначаючи рівень активації нейрона (рис.1).

ШНМ являють собою систему з'єднаних між собою простих процесорів (штучних нейронів), які взаємодіють (рис.2). Вхідні сигнали, позначені  $x_1, x_2, \dots, x_n$ , надходить на штучний нейрон (сукупності позначаються вектором  $X$ ). Кожен сигнал множиться на відповідну вагу  $w_1, w_2, \dots, w_n$ , і поступає на сумуючий блок, позначений  $\Sigma$ . Кожна вага відповідає «силі» однієї біологічної синаптичного зв'язку. Сумуючий блок, що відповідає тілу біологічного елемента, складає зважені входи алгебраїчно, створюючи вихід, який ми будемо називати NET. Кожен процесор мережі має справу тільки з сигналами, які він періодично отримує, і сигналами, які він періодично посилає іншим процесорам. Проте будучи з'єднаними в досить велику мережу з керованою взаємодією, такі локально прості процесори разом здатні виконувати достатньо складні завдання [1].

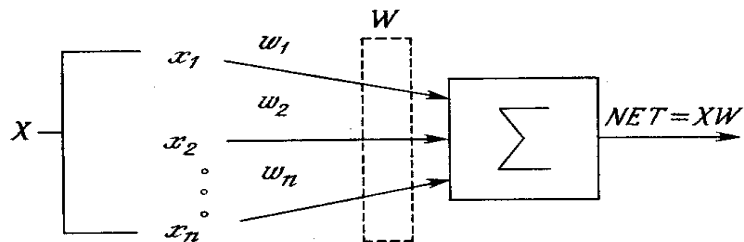


Рис. 1. Штучний нейрон

Сигнал NET далі, як правило, перетворюється активаційною функцією  $F$  і дає вихідний нейронний сигнал OUT.

Розглянута проста модель штучного нейрона ігнорує багато властивостей свого біологічного двійника, але мережі, побудовані з цих нейронів, виявляють властивості, які сильно нагадують біологічну систему.

Нейронні мережі не програмується, а навчаються. Технічно навчання полягає в знаходженні коефіцієнтів зв'язків між нейронами. У процесі навчання нейронна мережа здатна виявляти складні залежності між вхідними даними і вихідними, а також виконувати узагальнення. Це означає, що у разі успішного навчання мережа зможе повернути правильний результат на підставі даних, які були відсутні в навчальній вибірці, а також неповних та частково перекручених даних.

Мережа навчається, щоб для деякої множини входів давати бажану або наближену до неї множину виходів. Кожна така вхідна (або вихідна) множина розглядається як вектор. Навчання здійснюється шляхом послідовного пред'явлення вхідних векторів з одночасним підлаштуванням ваг відповідно до певної процедури. У процесі навчання ваги мережі поступово стають такими, щоб кожен вхідний вектор отримував вихідний вектор. Розрізняють алгоритми навчання з вчителем і без вчителя. Навчання з учителем припускає, що для кожного вхідного вектора існує цільовий вектор, що представляє собою необхідний вихід. Разом вони називаються навчальною парою. Зазвичай мережа навчається на деякому числі таких навчальних пар. Пред'являється вихідний вектор, обчислюється вихід мережі і порівнюється з відповідним цільовим вектором, різниця (помилка) за допомогою зворотного зв'язку подається в мережу і ваги змінюються відповідно до алгоритму, що прагнуть мінімізувати помилку. Вектори навчальної множини пред'являються послідовно, обчислюються помилки і ваги підлаштовуються для кожного вектора доти, поки помилка по всьому навчальному масиву не досягне прийнятно низького рівня.

Розвинене Кохоненом навчання без учителя не потребує цільового вектора для виходів і, отже, не вимагає порівняння з зумовленими ідеальними відповідями. Навчальна множина складається лише з вхідних векторів. Навчальний алгоритм підлаштовує ваги мережі так, щоб виходили узгоджені вихідні вектори, тобто щоб пред'явлення декілька близьких вхідних векторів давало однакові виходи. Процес навчання виділяє статистичні властивості навчальної множини і групує подібні вектори в класи. Пред'явлення на вхід вектора з даного класу дасть певний вихідний вектор, але до навчання неможливо передбачити, який вихід буде вироблятися даним класом вхідних векторів. Більшість сучасних алгоритмів навчання виросло з концепцій Хеба. Ним запропонована модель навчання без учителя, в якій синаптична сила (вага) зростає, якщо активовані обидва нейрона, джерело і приймач. Таким чином, часто використовуваний шляху в мережі посилюються і феномен звички і навчання через повторення отримує пояснення[2].

**Постановка задачі.** Одним з недоліків застосування штучних нейронних мереж у вирішенні сьогоденних завдань є невідповідність явно паралельної архітектури штучних нейронних мереж переважно послідовній обчислювальній архітектурі персональних ЕОМ, що використовуються в переважній більшості випадків для їх реалізації. Наслідком даної обставини є суттєва обчислювальна складність вирішення нейрозадач за допомогою персональних комп'ютерів, у випадках, коли для отримання рішення необхідно використання нейронних мереж великих розмірів. Спеціально розроблені явно паралельні нейрокомп'ютери та інші апаратні рішення дозволяють усунути згадану проблему невідповідності архітектур, однак, на практиці вони використовуються рідко через їх високу вартість. Внаслідок зазначених вище причин для підвищення продуктивності доводиться йти по шляху створення паралельних обчислювальних систем на базі персональних комп'ютерів. На сучасному етапі розвитку обчислювальної техніки такий спосіб є одним з основних способів прискорення обчислень і досягнення необхідної продуктивності.

Існує два основних варіанти організації систем паралельних обчислень: із загальною пам'яттю й з поділюваною пам'яттю [32].

Перший підхід забезпечує однорідний доступ до пам'яті і є основою для побудови симетричних мультипроцесорів - SMP (symmetric multiprocessor). Цей підхід може зокрема бути реалізованим на базі персональних комп'ютерів з багатоядерними процесорами.

Представниками систем другого класу є мультикомп'ютери - багатопроцесорні системи з розподіленою пам'яттю NORMA (no-remote memory access). Принципова особливість мультикомп'ютерів - кожен процесор системи може використати тільки свою локальну пам'ять. При цьому для доступу до даних, розташованих на інших процесорах, необхідно явно виконати операції передачі повідомлень. Даний підхід використовується при побудові двох важливих типів багатопроцесорних обчислювальних систем

- масивних-паралельних систем MPP (massively parallel processor) і кластерів. Це системи з розподіленою пам'яттю й з довільною комунікаційною системою. При цьому, як правило, кожний із процесорних елементів MPP системи є універсальним процесором, що діє по своїй власній програмі. Даний підхід може бути реалізованим при обчисленнях на базі кластерів персональних комп'ютерів.

Розглянемо підходи по розпаралелюванню роботи штучних нейронних мереж. Розпаралелювання зводиться до структуризації, тобто процесу поділу(декомпозиції) задачі на незалежні процеси, які не вимагають послідовного виконання і можуть, відповідно, бути виконані на різних процесорах незалежно один від одного

При функціонуванні нейронних мереж найчастіше використовуються такі методи розпаралелювання:

1. Розпаралелювання за алгоритмами (за навчальними алгоритмами або за робочими алгоритмами).
2. Розпаралелювання по навчальній вибірці.
3. Розпаралелювання по вузлах.
  - а) розпаралелювання по нейронам;
  - б) розпаралелювання по зв'язкам.

В основному методи розпаралелювання ШНМ виходять із того, що найбільш ресурсномістким є не стільки процес функціонування мережі, скільки процес її навчання.

Основним і найбільш популярним є розпаралелювання на рівні навчальної вибірки. При використанні даного підходу обчислювальні вузли одержують пари навчальної вибірки й виконують операції відповідно до алгоритму навчання тільки для локальної групи даних.

Відповідно до іншого підходу елементарні обчислювальні елементи мережі (нейрони) організуються в групи. Кроки алгоритму навчання виконуються для кожної групи нейронів на окремому обчислювальному вузлі. Вважається, що такий підхід має переваги перед розподілом по обчислювальних вузлах окремих нейронів, оскільки обчислювальне навантаження елементарного нейрона мале в порівнянні з витратами на обмін повідомленнями при синхронізації обчислень.

Наступний підхід зводиться до ефективного розпаралелювання ресурсномістких математичних операцій у процесі навчання мережі, так як практично всі алгоритми навчання штучних нейронних мереж припускають наявність великих обсягів матричних обчислень. При побудові математичної моделі процесу навчання застосовуються такі алгоритми, як алгоритм Хеба й алгоритм Кеннона.

Існує варіант розпаралелювання штучних нейронних мереж заснований на понятті колективів нейронних мереж. Даний підхід припускає підвищення ефективності процесу безпосереднього функціонування нейронної мережі. При цьому розв'язуване завдання одночасно запускається на декількох екземплярах нейронних мереж. Кожний екземпляр запускається на своєму обчислювальному вузлі. Після завершення роботи всіх екземплярів вибирається найкращий отриманий результат[3].

**Основна частина.** Розглянемо розповсюджене програмне забезпечення і програмні бібліотеки для моделювання процесів навчання й функціонування ШНМ і його можливості щодо розпаралелювання роботи нейронних мереж.

Emergent Neural Network Simulation System - це комплексне середовище моделювання для створення складних моделей мозку й пізнавальних процесів за допомогою апарату нейронних мереж. Emergent містить у собі графічне оточення для дослідження штучних нейронних мереж, а також різні інструменти, що допомагають розуміти, як протікає процес роботи мережі під час моделювання. Emergent Neural Network Simulation System має підтримку роботи симулятора на кластерах на базі стандарту MPI, а також обчислень на багатоядерних системах із загальною пам'яттю. При цьому MPI використовується в основному для розподілених обчислень математичних величин, наприклад, часток похідних функції помилки в межах шару.

Neural Network Toolbox - пакет, що входить до складу комплексної системи MATLAB і дозволяє проектувати, навчати, і моделювати різні види штучних нейронних мереж. Даний пакет може бути застосований для проорокування часових рядів, розпізнавання образів, моделювання й керування динамічними системами.

Для прискорення розрахунків, сполучених з моделюванням нейронних мереж, може бути застосований пакет Parallel Computing Toolbox. Даний пакет дозволяє розподіляти більші обсяги даних всередині кластера або здійснювати паралельну обробку в межах однієї багатоядерної системи. Зазначені можливості реалізовані за допомогою введення паралельних аналогів ряду конструкцій мови, що здійснюють розрахунки в паралельному середовищі, а також має додаткову реалізацію паралельних версій ряду математичних функцій. Розподілені обчислення в основному здійснюються за допомогою поділу зовнішніх даних і не торкають алгоритми навчання ШНМ.

STATISTICA Automated Neural Networks – це потужне середовище аналізу нейромережевих моделей. Воно підтримує сучасні, оптимізовані й потужні алгоритми навчання мережі (включаючи методи сполучених градієнтів, алгоритм Кохонена). Дозволяє контролювати параметри, що впливають на якість мережі, такими як: функції активації й помилок; складність мережі; вибір найбільш популярних мережних архітектур, включаючи багатозарові перцептрони; можливість генерації вихідного коду на мовах C, C++, C#, Java, PMML. Однак дане середовище моделювання не орієнтоване на виконання обчислень у розподіленому середовищі.

NeuroSolutions - це простий у використанні інструмент для розробки нейромережових моделей для операційних систем Microsoft Windows. Дана система включає можливості модульного й візуального проектування ШНМ, а також великі можливості їхнього навчання з використанням розповсюджених алгоритмів, удосконалених за допомогою введення генетичної оптимізації. Завдання, розв'язувані пакетом NeuroSolutions: кластерний аналіз, прогнозування ринкових продажів, прогнозування спортивних змагань, рішення завдання класифікації в медицині й інші. Паралельна обробка в рамках NeuroSolutions здійснюється за допомогою застосування технології NVIDIA CUDA - організації паралельних обчислень із використанням відеокарти персонального комп'ютера.

FANN (Fast Artificial Neural Network Library) - це відкрита крос-платформова нейромережева бібліотека, що реалізує ШНМ мовою програмування С. У можливості FANN входить: навчання по методу зворотного поширення помилки, велика кількість параметрів, що набудовуються, збереження й завантаження параметрів ШНМ, керування навчальними вибірками, графічний інтерфейс і інших. Паралельні й розподілені обчислення реалізовані на рівні математичних операцій з більшими обсягами даних у процесі навчання нейронної мережі, а також на рівні ручного розподілу даних між декількома мережами й подальшим їхнім запуском на різних обчислювальних вузлах.

**Висновки.** Таким чином, стосовно сучасних програмних реалізацій ШНМ можна вказати наступні недоліки:

1. Висока складність завдання обробки сучасних ШНМ, що не дозволяє вирішувати достатньо складні завдання як на персональних комп'ютерах, так і на кластерах персональних комп'ютерів на основі локальних обчислювальних мереж.

2. В існуючих системах використовується тільки низькорівневий, статичний підход до організації розподілених обчислень на основі технології MPI з використанням кластерної архітектури, або ж відсутні можливості організації розподілених паралельних обчислень.

3. Організація паралельної обробки ШНМ із використанням підходів на базі розподілу навчальної вибірки або на базі матричних операцій, що не дозволяє повною мірою використати можливості паралельної обробки ШНМ.

Таким чином, представляється актуальною розробка розподіленої обчислювальної системи, яка б динамічно масштабувалась не тільки на багатоядерні процесори в персональних комп'ютерах або на кластерну архітектуру на основі локальних обчислювальних мереж, але й на слабкоз'язані гетерогенні комп'ютери (мережа GRID). Таке рішення не може бути реалізоване без розробки відповідного спеціалізованого методу розпаралелювання нейрообчислень, який б базувався на виділенні слабкоз'язаних обчислювальних процесів з мінімізацією потоків передачі даних, що протікають між ними.

## Література

1. Штучні нейронні мережі: концепції та теорії, IEEE Computer Society, 1992
2. Річард П. Липпман, Введення в програмування з нейронними мережами, IEEE Акустика, мови і обробки сигналів (журнал), квітень 1987.
3. Плотникова Н. П. Модели, алгоритмы и реализация нейронных сетей в многопроцессорных и распределенных вычислительных средах, Диссертация на соискание ученой степени кандидата технических наук, Саранск 2014.

## References

1. Artificial Neural Networks: Concepts and Theory, IEEE Computer Society Press, 1992
2. Richard P. Lippmann, An Introduction to Computing with Neural Nets, IEEE Acoustics, Speech, and Signal Processing Magazine, April 1987.
3. Plotnikova N. P. Models, algorithms and implementation of neural networks in multiprocessor and distributed computing environments, Thesis for the degree of technical sciences candidate, Saransk 2014.

Рецензія/Peer review : 22.12.2014 р.

Надрукована/Printed : 2.1.2015 р.