

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Тернопільський національний економічний університет
Факультет комп'ютерних інформаційних технологій
Кафедра комп'ютерної інженерії

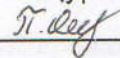
ПИЦУРА Олександр Васильович

Алгоритми нейромережевих багатооперандних операцій / Algorithms of Multioperand operation neural network

спеціальність: 123 - Комп'ютерна інженерія
магістерська програма - Комп'ютерна інженерія

Магістерська робота

Виконав студент групи КІм-21
О. В. Пицура



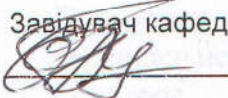
Науковий керівник:
д.т.н., професор, І. Г. Цмоць



Магістерську роботу допущено
до захисту:

"01" 01 2018 р.

Завідувач кафедри

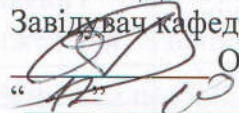
 О. М. Березький

ТЕРНОПІЛЬ - 2018

Тернопільський національний економічний університет
Факультет комп'ютерних інформаційних технологій
Кафедра комп'ютерної інженерії
Освітній ступінь «магістр»
спеціальність: 123 - Комп'ютерна інженерія
магістерська програма - Комп'ютерна інженерія

ЗАТВЕРДЖУЮ

Завідувач кафедри


О.М. Березький
" 11 " 2016 р.

ЗАВДАННЯ НА МАГІСТЕРСЬКУ РОБОТУ СТУДЕНТУ


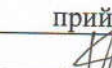
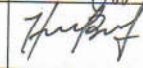
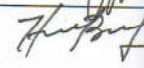
Пицура Олександр Васильович

(прізвище, ім'я, по батькові)

1. Тема магістерської роботи «Алгоритми нейромережових багатооперандних операцій / Algorithms of neural network operations»
керівник роботи д.т.н., професор І.Г. Цмоць.
затверджені наказом по університету від 11 жовтня 2016 р. № 669.
2. Строк подання студентом роботи «15» січня 2018 року
3. Вихідні дані до магістерської роботи
Об'єкт дослідження – алгоритми нейромережових багатооперандних операцій.
Предмет дослідження – пристрої нейромережових багатооперандних операцій.
4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)
 - Засоби обміну для штучних нейронних мереж.
 - Засоби реалізації пристроїв обміну.
 - Розробка пристрою паралельно вертикально-групового перетворювача кодів.
 - Розробка структури пристрою обчислення суми квадратів різниць та скалярного добутку.
 - Програмна реалізація алгоритму роботи пристрою нейроелемента вертикально-групового типу з використанням багатовходового суматора.
5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень):
 - тема, мета, завдання, методи досліджень, наукова новизна, практичне значення;
 - структурна схема пристрою вертикально-групового перетворювача кодів;





- структурна схема компоненти обчислення максимальних мінімальних чисел з паралельним вертикально-груповим методом;
- структурна схема компоненти обчислення суми квадратів різни з паралельним формуванням і послідовним підсумовуванням.

6. Консультанти розділів магістерської роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		Завдання видів	Завдання прийняв
Антиплагіат	Мельник Г.М., доцент		
Нормо-контроль	Гураль І. В., викладач		

7. Дата видачі завдання «12» 10 2016р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів магістерської роботи	Строк виконання етапів магістерської роботи	Примі
1	Аналіз галузей застосування, архітектура та операційний базис нейронних мереж	3.11.2016 – 1.01.2017	
2	Алгоритми реалізації нейромережових багатооперандних операцій	2.01.2017 – 31.05.2017	
3	Апаратна реалізація структури базових компонентів нейромереж	1.06.2017 – 25.01.2018	
4	Нормоконтроль, попередній захист	16.01.2018 – 2.02.2018	
5	Захист	4.02.2018	

Студент


(підпис)

Пицуря О. В.

Керівник магістерської роботи


(підпис)

д.т.н., професор, Цмоць І. Г.

ЗМІСТ

Вступ

1 Аналіз галузей застосування, архітектура та операційний базис нейронних мереж	6
1.1 Галузі застосування нейромережових технологій	6
1.1.1 Застосування нейромереж в системах управління	10
1.1.2 Застосування нейромереж в телекомунікаційних системах.....	13
1.2 Основні типи нейромережових задач.....	16
1.3 Архітектури штучних мереж	19
1.3.1 Ієрархічні структури	20
1.3.2 Конкурентні структури.....	21
1.3.3 Рекурентні структури	23
1.4 Операційний базис нейронних мереж.....	26
2 Алгоритми реалізації нейромережових багатооперандних операцій	31
2.1 Виділення нейромережових багатооперандних операцій.....	31
2.2 Алгоритм обчислення скалярного добутку	34
2.3 Алгоритм групового підсумовування	40
3. Апаратна реалізація структури базових компонентів нейромереж.....	45
3.1 Паралельний вертикально-груповий перетворювач кодів.	45
3.2 Компонента обчислення максимального і мінімального чисел.....	46
3.3 Компонента обчислення суми квадратів різниць.	50
3.4 Компонента обчислення скалярного добутку.....	54
Висновки	58
Список використаних джерел	60

1 АНАЛІЗ ГАЛУЗЕЙ ЗАСТОСУВАННЯ, АРХІТЕКТУРА ТА ОПЕРАЦІЙНИЙ БАЗИС НЕЙРОННИХ МЕРЕЖ

1.1 Галузі застосування нейромережових технологій

Напевно, в кожній предметній області при найближчому розгляді можна знайти постановки нейромережових завдань. Список областей, де рішення такого роду завдань має практичне значення, досить широкий. Нижче представлені такі області з прикладами конкретного застосування.

Економіка та бізнес: передбачення ринків, оцінка ризику неповернення кредитів, передбачення банкрутств, оцінка вартості нерухомості, виявлення пере- і недооцінених компаній, автоматичне рейтингування, оптимізація портфелів, оптимізація товарних і грошових потоків, автоматичне зчитування чеків і форм, безпеку транзакцій за пластиковими картками.

Програмне забезпечення компанії RETEK, дочірньої фірми HNC Software, - лідер з оборотом понад \$ 1 млрд. Її продукт Retek Predictive Enterprise Solution включає розвинені засоби нейромережового аналізу великих потоків даних, характерних для великої роздрібною торгівлі. Він також містить прогностичний блок, щоб можна було заздалегідь прорахувати наслідки тих чи інших рішень.

Медицина: обробка медичних зображень, моніторинг стану пацієнтів, діагностика, факторний аналіз ефективності лікування, очищення показань приладів від шумів.

Група НейроКомп з Красноярська (під керівництвом Олександра Миколайовича Горбаня) спільно з Красноярським міжобласним офтальмологічним центром ім. Макарова розробили систему ранньої діагностики судинної оболонки ока. Цей вид раку складають майже 90% всіх внутрішніх пухлин і легко діагностується лише на пізній стадії. Метод заснований на непрямому вимірюванні вмісту меланіну в віях. Отримані дані спектрофотометрії, а також загальні характеристики обстежуваного (стать, вік і ін.) Подаються на вхідні синапси 43-нейронного класифікатора. Нейромережа вирішує, чи є у

пацієнта пухлина, і якщо так, то визначає її стадію, видаючи, крім цього, процентну ймовірність своєї впевненості.

Зв'язок: стиснення відеоінформації, швидке кодування-декодування, оптимізація стільникових мереж і схем маршрутизації пакетів.

Нейромережі вже продемонстрували коефіцієнт стиснення 120:1 для чорно-білого відео. Кольорове відео допускає приблизно вдвічі більший ступінь стиснення 240:1 за рахунок спеціальної схеми кодування кольорів;

Інтернет: асоціативний пошук інформації, електронні секретарі та агенти користувача в мережі, фільтрація інформації в push-системах, колаборативна фільтрація, рубрикація новинних стрічок, адресна реклама, адресний маркетинг для електронної торгівлі.

Фірма Autonomy відокремилася від батьківської фірми Neurodynamics в червні 1996 року зі статутним капіталом \$ 45 млн і ідеєю просування на ринок Internet електронних нейромережових агентів. Згідно з її прес-релізу, початкові вкладення окупилися вже через рік. Компанія виробляє сімейство продуктів AGENTWARE, що створюють і використовують профілі інтересів користувачів у вигляді персональних автономних нейроагентів. Такі компактні нейроагенти (<1 Кб) можуть представляти користувача в будь-якому з продуктів компанії. Наприклад, агенти можуть служити в якості нейросекретарів, фільтруючи інформацію, що надходить з інформаційних каналів інформації. Вони також можуть постійно перебувати на сервері провайдера, або надсилатися для пошуку у віддалених базах даних, здійснюючи відбір даних на місці. У майбутньому, коли ця технологія отримає достатнього поширення, вона дозволить знизити навантаження на трафік Мережі.

Автоматизація виробництва: оптимізація режимів виробничого процесу, комплексна діагностика якості продукції (ультразвук, оптика, гамма-випромінювання), моніторинг і візуалізація багатовимірної диспетчерської інформації, попередження аварійних ситуацій, робототехніка.

Ford Motors Company впровадила у себе нейросистем для діагностики двигунів після невдалих спроб побудувати експертну систему, тому що хоча

досвідчений механік і може діагностувати несправності він не в змозі описати алгоритм такого розпізнавання. На вхід нейросистеми подаються дані від 31 датчика. Нейромережа навчається різним видам несправностей по 868 прикладів.

Після повного циклу навчання якість діагностування несправностей мережею досягає рівня кращих експертів, і значно перевершує їх у швидкості.

Політичні технології: аналіз і узагальнення соціологічних опитувань, прогноз динаміки рейтингів, виявлення значущих чинників, об'єктивна кластеризація електорату, візуалізація соціальної динаміки населення.

Уже згадувана раніше група НейроКомп з Красноярська досить впевнено пророкує результати президентських виборів в США на підставі анкети з 12 питань. Причому, аналіз навченої нейромережі дозволив виявити п'ять ключових питань, відповіді на яких формують два головні чинники, що визначають успіх президентської кампанії.

Безпека і охоронні системи: системи ідентифікації особистості, розпізнавання голосу, осіб в натовпі, розпізнавання автомобільних номерів, аналіз аерокосмічних знімків, моніторинг інформаційних потоків, виявлення підробок

Багато банків використовують нейромережі для виявлення підробок чеків. Корпорація Nestor (Providence, Rhode Island) встановила подібну систему в Mellon Bank, що за оцінками має заощадити останньому \$ 500,000 в рік. Нейромережа виявляє в 20 разів більше підробок, ніж встановлена до неї експертна система. (Tearing up the Rules, Banking Technology, листопад 1993).

Введення і обробка інформації: Обробка рукописних чеків, розпізнавання підписів, відбитків пальців і голоси. Введення в комп'ютер фінансових і податкових документів.

Розроблені італійською фірмою RES Informatica нейромережеві пакети серії FlexRead, використовуються для розпізнавання і автоматичного введення рукописних платіжних документів і податкових декларацій. У першому випадку вони застосовуються для розпізнавання не тільки кількості товарів та їх вартості, але також і формату документа. У разі податкових декларацій розпізнаються фіскальні коди і суми податків.

Наведені вище застосування нейромереж характеризують їх як ефективний інструмент аналізу даних. І краще за інших ними може скористатися саме фахівець у своїй предметній області. Основні труднощі на шляху ще більш широкого поширення нейротехнологій - в невмінні широкого кола професіоналів формулювати свої проблеми в термінах, що допускають просте нейромережеве рішення.

Основними перевагами нейронних мереж як логічного базису алгоритмів розв'язання складних задач є: інваріантність, адекватність і відмовостійкість.

Навіть такі прості завдання, як додавання, віднімання і множення чисел ефективніше вирішувати за допомогою нейрокомп'ютерів, ніж на відомих булевських елементах.

У США і Фінляндії ведуться роботи по використанню нейрокомп'ютерів для керування хімічними реакторами. У нашій країні цим не займаються через моральне застаріння існуючих реакторів.

Нейронні мережі мають декілька унікальних особливостей: методика програмування методом навчання або самонавчання, можливість вироблення вихідного сигналу (вектора) в умовах неповного набору вхідних даних і працездатність нейронної мережі при її частковому пошкодженні.

Більшість невдач на шляху вдосконалення штучного інтелекту протягом останніх 30 років пов'язані з тим, що для вирішення важливих і складних завдань вибиралися неадекватні за можливостями обчислювальні засоби.

Перспективними завданнями обробки зображень нейрокомп'ютерів є обробка аерокосмічних зображень, пошук, виділення і розпізнавання на зображенні рухомих об'єктів заданої форми.

Нейрокомп'ютери дають стандартний спосіб вирішення багатьох нестандартних завдань. Один нейрокомп'ютер вирішить і цю задачу, і другу, і третю - не треба щораз проектувати спеціалізовану ЕОМ.

Нейрокомп'ютери ефективні там, де потрібен аналог людської інтуїції. Для розпізнавання образів, читання рукописних текстів, підготовки аналітичних прогнозів, перекладу з однієї мови на іншу і т.п.

1.1.1 Застосування нейромереж в системах управління

В історичному плані можна стверджувати, що розробка систем управління (СУ) завжди відбувається поетапно. В якості таких етапів можна виділити:

- етап розробки концепції побудови СУ;
- етап моделювання СУ, відповідно до запропонованої концепції побудови;
- етап аналізу отриманих результатів;
- етап доопрацювання (модернізації) концепції побудови СУ.

Протягом усіх цих етапів не припиняються теоретичні дослідження, які дозволяють вибирати основні напрями вдосконалення спочатку сформульованої концепції побудови СУ і поширювати її основні ідеї на ряд суміжних областей.

За аналогічною схемою відбувається розвиток СУ, що використовують штучні НС. Однак слід зауважити, що велика кількість розроблених апаратних і програмних моделей нейромережевих СУ часто випереджають теоретичне розуміння що відбуваються при цьому процесів і наявних проблем [93].

Необхідним етапом вирішення завдань управління нелінійними динамічними системами є отримання їх адекватних математичних моделей, що базується, як правило, на теоретичному та експериментальному аналізі властивостей цих систем. Теоретичний аналіз процесів, що відбуваються в системі, дозволяє отримати математичний опис у вигляді, наприклад, диференціальних рівнянь. При експериментальному аналізі на основі спостережень вхідних і вихідних сигналів системи отримують або параметричну, або непараметричну модель. Найбільш широкого поширення набули параметричні моделі, які потребують вирішення завдань структурної та параметричної ідентифікації і використовують обмежену кількість параметрів. Незважаючи на величезну кількість робіт, різноманіття видів нелінійностей не дозволяє створити єдину теорію ідентифікації нелінійних систем. Застосовуваний найчастіше класичний підхід заснований на апроксимації нелінійностей, наприклад рядами Вольтера, Гаммерштейна, Вінера, поліномами Колмогорова-Габора та ін. Однак область застосування таких моделей обмежена.

Крім того, додаткові труднощі отримання адекватного математичного опису обумовлено наявністю в реальних сигналах перешкод [43].

Одним з перших використовуваних методів побудови нейромережових СУ був метод, заснований на "копіюванні" існуючого контролера. Застосувавши цей метод в 1964 Уїдроу назвав його методом побудови експертної системи за рахунок отримання знань від вже існуючого експерта. Архітектура такої СУ представлена на рисунку 1.1.

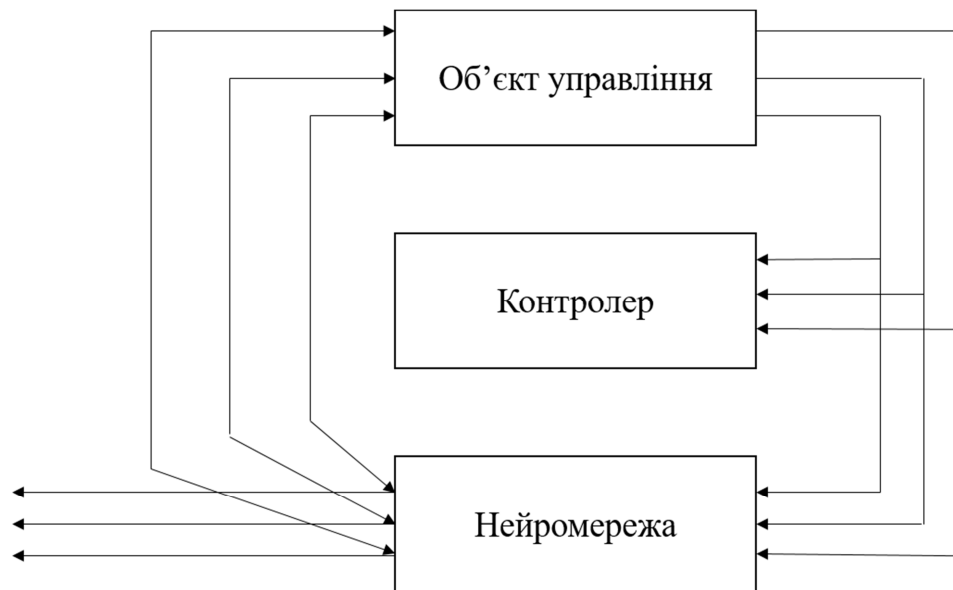


Рисунок 1.1 – Нейромережева СУ, заснована на "копіюванні" існуючого контролера

Дивлячись на цей малюнок можна засумніватися в корисності цього методу. Навіщо потрібно використовувати ще один керуючий контролер (у вигляді НС), якщо один уже існує? Однак, по-перше, існуючий контролер може бути незручний при використанні (наприклад, в ролі такого контролера може виступати людина), а по-друге, для вироблення ефективного управління НС може використовувати відмінну, від існуючого контролера, за способом представлення (легше виміряти, формалізувати і т. д.) інформацію про стан об'єкта управління.

В даний час досить добре розроблений і широко використовується цілий ряд інших можливих архітектур побудови нейромережових СУ [73, 93]. У всіх з них, призначенням нейромережевого контролера є вироблення адекватного

керуючого сигналу для управління динамікою станів об'єкта управління від початкового стану до бажаного підсумкового стану. Причому зміна станів повинна відбуватися по оптимальній траєкторії. Організація контролю за станом об'єкта управління та реалізація нейромережевого контролера в значній мірі залежать від обраного алгоритму навчання та використовуваної структури управління. Найбільш широко використовуваними є схема прямого (безпосереднього) управління і схема непрямого управління. При цьому найчастіше в якості алгоритму навчання використовується алгоритм зворотного поширення помилки [73].

У схемі непрямого управління параметри об'єкта управління визначаються в кожен момент часу і потім використовуються для оцінки параметрів контролера (рисунок 1.2). Таким чином, має місце явно виражений процес ідентифікації.

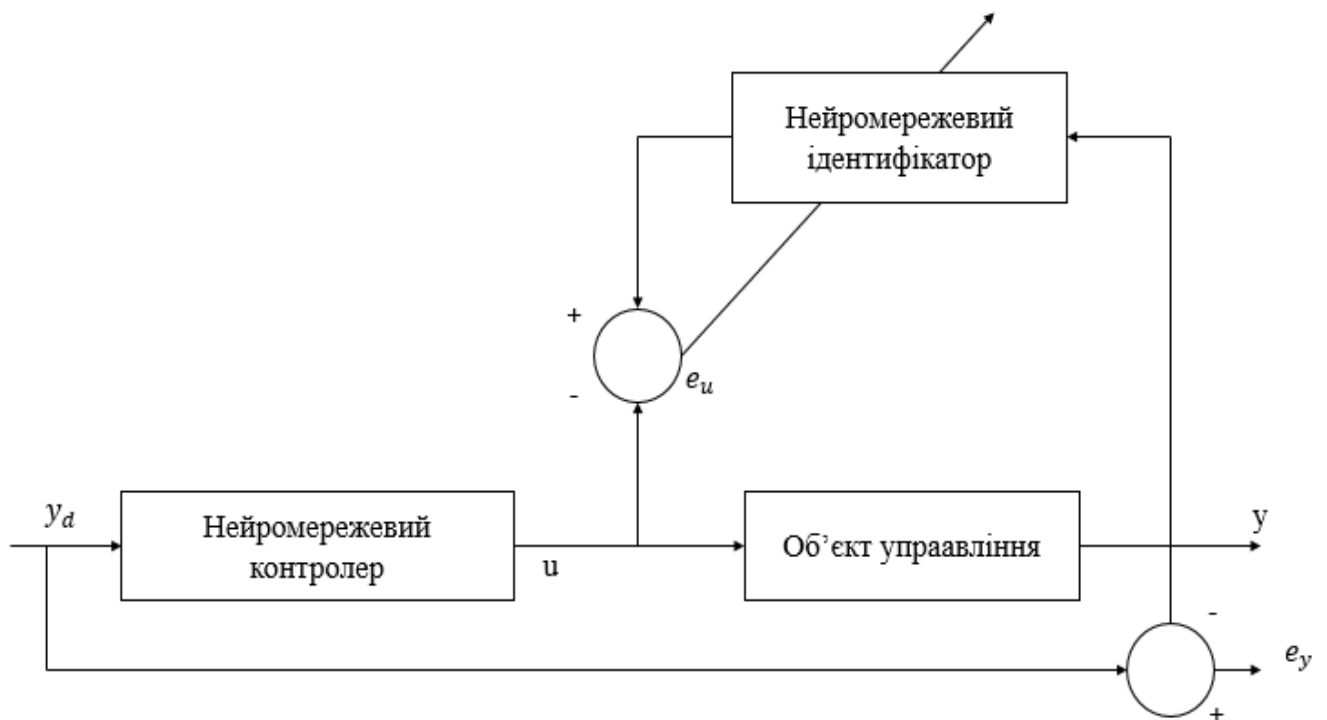


Рисунок 1.2 – Схема не прямого управління

Недоліком такої схеми є те, що ідентифікація та управління ґрунтуються виключно на помилку e_u , отже, мінімізацію помилки на виході системи e_y гарантувати не можна.

У схемі прямого управління параметри нейромережевого контролера регулюються таким чином, щоб зменшити безпосередньо помилку виходу e_y .

Однією з областей теоретичних досліджень, що розглядають проблеми використання нейромереж в системах управління, є порівняння таких методів управління з іншими відомими типами СУ, виявлення притаманних нейромережевим методам особливостей і їх аналіз. Порівнювалися стійкість, швидкість збіжності, робота в умовах шумів, необхідний обсяг пам'яті та ін. Хоча кожен з розглянутих методів має як хороші, так і погані характеристики слід зауважити, що метод нейромережевого управління має такі корисні властивості, які погано реалізуються в двох інших методах.

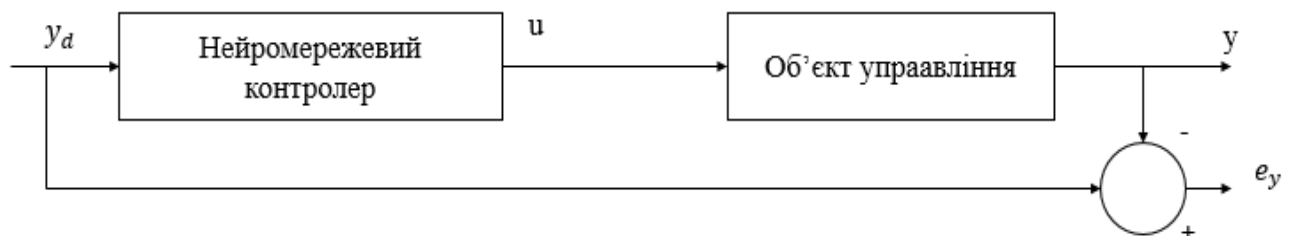


Рисунок 1.3 – Схема прямого управління

Таким чином, напрашується наступний висновок. Майбутнє інтелектуального управління лежить в поєднанні традиційного управління з потенційними можливостями і перспективами використання систем, заснованих на використанні штучних нейромережах.

1.1.2 Застосування нейромереж в телекомунікаційних системах

Тривалий час вважалося, що нейрокомп'ютери ефективні і можуть застосовуватися лише для вирішення так званих неформалізованих і погано формалізованих задач, пов'язаних з необхідністю включення в алгоритм їх вирішення систему адаптації на реальному експериментальному матеріалі. В першу чергу до таких завдань відносяться завдання розпізнавання образів.

Останнім часом сфера застосування нейроінформаційних технологій динамічно розширюється [19]. Вони все ширше і ширше використовуються в задачах з яскраво вираженим природним паралелізмом: обробка сигналів, зображень.

Серед основних виділяють чотири області застосування нейрокомп'ютерів в системах зв'язку [9]:

- управління комутацією;
- маршрутизація;
- управління трафіком;
- розподіл каналів в рухливих системах радіозв'язку.

Рішення практично будь-якої задачі в нейромережевому логічному базисі передбачає наявність наступних етапів [12]:

- формування вхідного і вихідного сигналів НМ;
- формування бажаного вихідного сигналу НМ;
- формування сигналу помилки і функціонала оптимізації;
- формування структури НМ, адекватної обраної завдання;
- розробка алгоритму налаштування НМ, еквівалентного процесу рішення задачі в нейромережевому логічному базисі;
- проведення досліджень процесу рішення задачі.

Нейронна мережа, яка використовується для управління роботою високошвидкісної комутаційної мережі з пакетною передачею в асинхронному режимі використовується для управління комутацією пакетів при передачі мови, зображень і даних. Комутатор представлений як логічний пристрій, на вхід якого надходить N сигналів і на виході відтворює ці сигнали в будь-якому представленому порядку.

В даний час число зв'язків, емульованого в нейрокомп'ютер, може досягати декількох сотень мільйонів. Тому стає можливим побудова комутаторів з нейромережевим управлінням на кілька сот каналів [9].

Також НМ використовують для управління трафіком в складних багатоступеневих системах зв'язку. Труднощі завдання обумовлена тим, що, по-

перше, заздалегідь невідомі параметри, що характеризують потоки інформації, а по-друге, вимоги до якості можуть змінюватися з часом. НМ вирішує завдання оптимізації, пов'язані з перебуванням безконфліктних потоків при заданих вхідних і вихідних значеннях. При цьому НМ легко адаптується до змін умов.

Постановка і вирішення завдання розподілу каналів в рухливих системах радіозв'язку в нейромережевому базисі мало відрізняються від постановки і рішення задачі маршрутизації. Різниця полягає в стільникового структури радіомережі та великій кількості комутованих вузлів [9, 56].

Крім перерахованих вище областей застосування нейронних мереж в телекомунікаційних системах, перспективним є використання нейромережевих алгоритмів в задачах кодування та декодування інформації. Як першочергові можна розглядати завдання пов'язані з обробкою мовної інформації і зображень [12]. Широко відомим є метод стиснення інформації, запропонований в 1987 році. При цьому використовується тришаровий перцептор, у якого число елементів вхідного і вихідного шарів однаково, а число елементів прихованого шару значно менше. Якщо навчання вироблено таким чином, що на виході відтворюється той же самий вектор, який подається на вхідний шар, то такий перцептор автоматично здійснює стиснення інформації (на елементах прихованого шару виникає уявлення кожного вектора, яке значно коротше, ніж довжина вектора, що подається на вхід). Таким чином, передача попередньо стислої інформації по лінії зв'язку дозволяє зменшити число необхідних для цього каналів. Реалізується це таким чином. На одному кінці лінії поміщають вхідний і прихований шари перцептору, а результат роботи елементів прихованого шару (короткі вектори) подають в канал. Розмістивши на іншому кінці лінії копію прихованого шару і вихідний шар, можна на виході останнього відтворити початковий вектор.

Таким чином, область застосування штучних НМ в телекомунікаційних системах постійно розширюється. У цих умовах поява на ринку недорогих як багатофункціональних, так і орієнтованих на рішення конкретних завдань нейрочіпів і нейрокомп'ютерів сприятиме швидкому переходу до чергового етапу

розвитку телекомунікаційних систем - появи інтелектуальних телекомунікаційних систем.

1.2 Основні типи нейромережових задач

Нейронна мережа - це мережа з кінцевим числом шарів з однотипних елементів - аналогів нейронів з різними типами зв'язків між шарами. При цьому число нейронів в шарах вибирається виходячи з необхідності забезпечення заданої якості виконання завдання, а число шарів нейронів - якомога менше для скорочення часу рішення задач.

Штучні нейронні мережі також розвиваються завдяки біології, яка надихала і надихає розробників штучних нейронних систем (НС). Проте, як працює мозок ми знаємо досить мало і розробникам штучних НС доводиться шукати за межами сучасних біологічних знань нові структури, що дозволяють виконувати корисні функції. Через це створюються мережі, які б не могли існувати в живій матерії або можуть бути використані неправдоподібно, тому що у них використано багато допущень про анатомію і функціонуванні мозку.

Нервова система кожної людини дуже складна. Вона побудована з елементів, які називаються нейронами. Приблизно 10-ти нейронів беруть участь в 10¹⁵ передавальних зв'язках. Комунікаційна система мозку утворюється за допомогою, унікальних здібностей нейронів, обробки, передачі і прийому електрохімічних сигналів по нервових шляхах.

З першого погляду штучний нейрон імітує властивості біологічного нейрона. Він володіє групою синапсів - односпрямованих вхідних зв'язків, які з'єднані з виходами інших нейронів, а таке має аксон - вихідний зв'язок даного нейрона, з якого сигнал (збудження або гальмування) надходить на синапси наступних нейронів. Кожен вхід множиться на відповідну вагу, аналогічний синоптичній силі, і всі сигнали підсумовуються, визначаючи рівень активації

нейрона [32]. Загальний вигляд нейрона, що реалізує ці властивості, наведено на рисунку 1.4.

Нейромережі найбільш пристосовані до вирішення широкого кола завдань, так або інакше пов'язаних з обробкою образів. Ось список типово поставлених задач для нейромереж:

Класифікація образів. Завданням класифікації постає визначення приналежності вхідного образу (наприклад, звукового сигналу або написаного символу від руки), який представлений вектором ознак до одного або декількох попередньо визначених класів. До відомих класифікаторів відносять розпізнавання мови, розпізнавання букв, класифікація сигналу електрокардіограми, класифікація кліток крові.

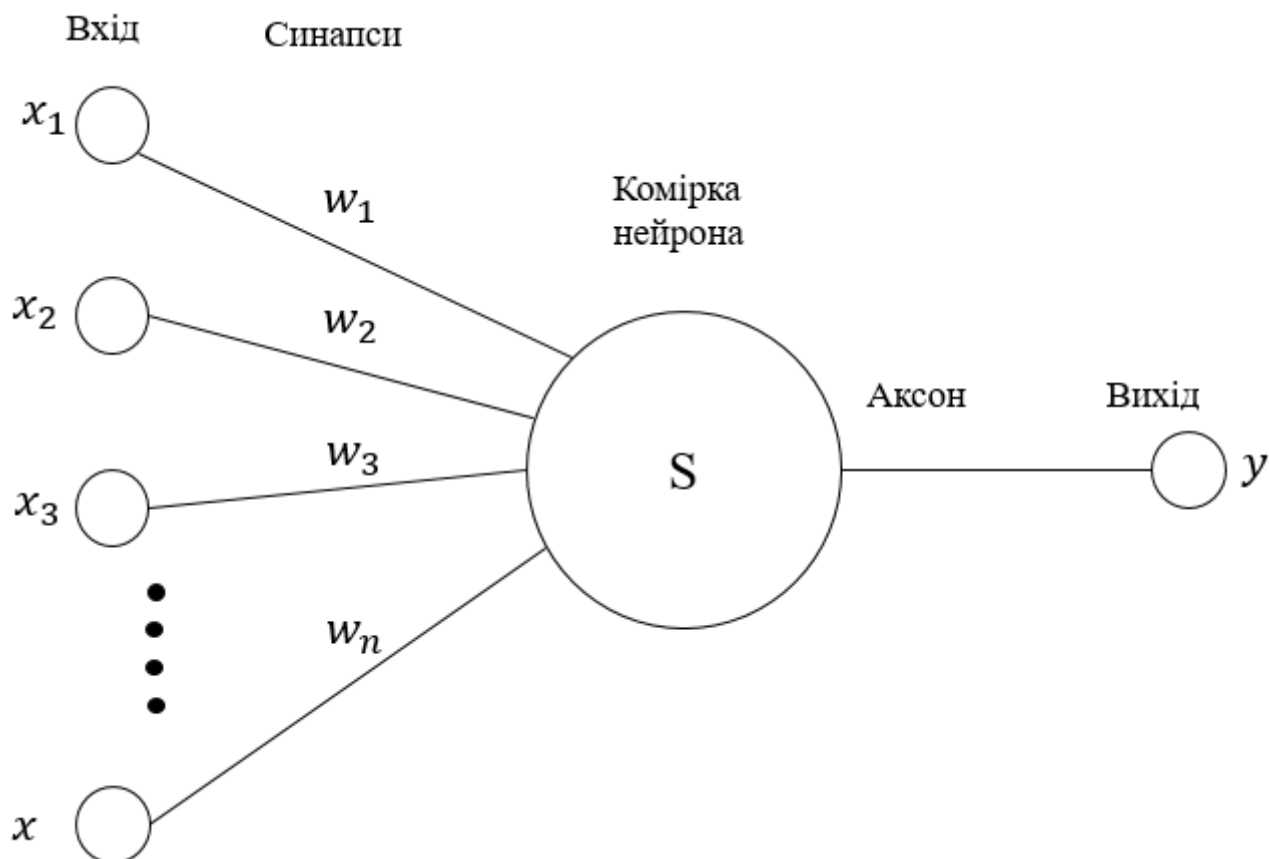


Рисунок 1.4 – Структура штучного нейрона

Кластеризація. При вирішенні задачі кластеризації, навчальна множина не має міток класів. Алгоритм кластеризації підбирає схожі образи і поміщає їх в

один кластер. Кластеризацію застосовують для стиснення і дослідження властивостей даних, а також для видобутку знань.

Апроксимація функцій. Давайте задамо навчальну вибірку $((x_1, y_1), (x_2, y_2) \dots, (x_n, y_n))$ (пари даних вхід-вихід). Ця вибірка буде генеруватися невідомою функцією F , спотвореною шумом. Завдання апроксимації полягає в тому щоб знайти невідому функцію F . Апроксимація функцій застосовується для вирішенні численних інженерних і наукових задач моделювання.

Передбачення. Нехай задані n дискретних відліків $\{y(t_1), y(t_2), \dots, y(t_n)\}$ в послідовні моменти часу t_1, t_2, \dots, t_n . Завдання полягає в передбаченні значення $y(t_{n+1})$ в наступний момент часу t_{n+1} . Передбачення відіграють велику роль у прийнятті рішень в бізнесі, науці і техніці (передбачення цін на фондовій біржі, прогноз погоди).

Оптимізація. До проблем оптимізації можна віднести проблеми в математиці, техніці, статистиці, науці, медицині та економіці. Завданням алгоритму оптимізації полягає у знаходженні якогось такого рішення, яке б задовольнило систему обмежень і максимізувало або мінімізувало цільову функцію.

Пам'ять, що адресується за змістом. У традиційних комп'ютерах до пам'яті звертаються тільки за допомогою адреси, що не залежить від змісту пам'яті. Навіть якщо допущена якась помилка в обчисленні адреси, то інформація все одно буде знайдена, але в цьому випадку вона буде мати зовсім інший зміст. Асоціативна пам'ять або пам'ять що адресується за змістом є доступна тільки за вказівкою заданого змісту. Вміст пам'яті може бути викликано навіть по частковому входу або пошкодженому змісті. Асоціативна пам'ять може бути використана в мультимедійних інформаційних базах даних.

Управління. Розглянемо динамічну систему, задану сукупністю $\{u(t), y(t)\}$, де $u(t)$ - вхідний керуючий вплив, а $y(t)$ - закриття системи в деякий час t . У системах управління з еталонною моделлю метою управління є розрахунок такого вхідного впливу $u(t)$, при якому система діє по бажаній траєкторії, заданій еталонною моделлю. Прикладом є оптимальне управління двигуном.

Не дивлячись на ті переваги нейронних мереж в окремих областях над традиційними обчисленнями, дані нейромережі не є досконалими для прийняття правильних рішень. Все ж таки вони ще також навчаються і можуть робити "помилки" на даному етапі. Крім того, не можна бути впевненим, що розроблена нейромережа буде оптимальною. Застосування нейромереж вимагає від розробника виконання ряду умов:

- безліч даних, що містять інформацію, що характеризує проблему;
- відповідно встановлене за розмірами безліч даних для навчання і тестування мережі;
- розуміння вирішуваної проблеми;
- вибір функції суматора, передатної функції і методів навчання;
- розуміння інструментальних засобів розробника;
- відповідна потужність обробки.

Нові можливості обчислень вимагають від розробника виходу за рамки традиційних обчислень. На початку, обчислення були лише апаратними, і інженери зробили його працюючими. Потім, були фахівці з програмного забезпечення: програмісти, системні інженери, фахівці по базах даних і проектувальники. Тепер з'явилися нейронні архітектори. Новий працівник повинен мати кваліфікацію вище, ніж у попередників. Наприклад, він повинен знати статистику для вибору і оцінювання навчальних і тестових множин. При створенні ефективних нейромереж важливим для сучасних інженерів програмного забезпечення є логічне мислення, практичні знання та інтуїція.

1.3 Архітектури штучних мереж

Що ж являє собою штучна нейронна мережа(ШНМ)? На даний момент не існує такого визначення яке б задовільнило всіх. Причиною є те, що проблемою

нейронних мереж займаються спеціалісти в різних галузях науки. Кожна галузь визначає це визначення по різному.

Сьогодні є велика кількість нейронних архітектур та їх модифікацій, що орієнтовані на вирішення конкретного типу задач. Зараз розглянемо найбільш поширені архітектури штучних мереж.

1.3.1 Ієрархічні структури

Напрявлені нейронні мережі (ННМ) і перцептони, вони подають інформацію з переду назад (вхід і вихід, відповідно). Нейронні мережі часто описуються як такі, що мають рівні, де кожен рівень складається з паралельно вхідних, прихованих або вихідних комірок.

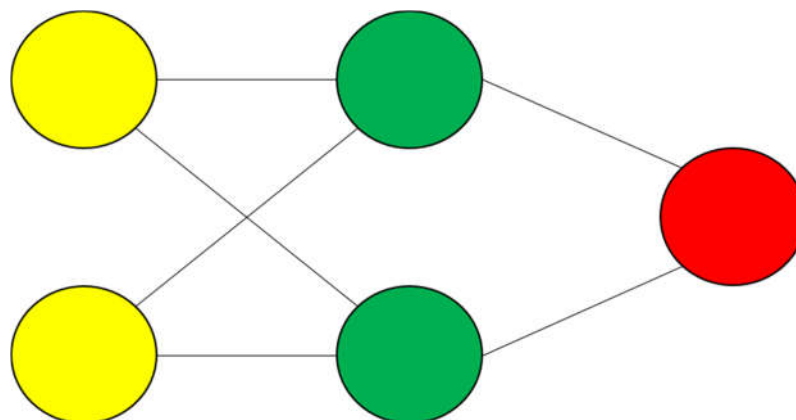


Рисунок 1.5 – Напрявлені нейронні мережі

Рівень сам по собі не має зв'язків і загалом два сусідніх рівня повністю пов'язані (кожен нейрон утворює один рівень для кожного нейрона до іншого рівня).

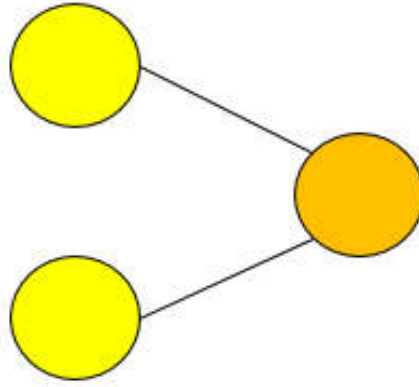


Рисунок 1.6 - Персептон

Найпростіша дещо практична мережа має дві вхідні комірки та одну вихідну комірку, яку можна використовувати для моделювання логічних виходів. Один, як правило, направлені нейронні мережі навчають через зворотне розповсюдження, надаючи мережеві набори даних про "те, що відбувається" та "що ми хочемо вийти".

Радіальна базисна функція (РБФ) – це напрямлена нейронна мережа з радіальними базисними функціями, я функції активації. Це не означає що вони не мають своїх цілей, але більшість ННМ з іншими функціями активації не отримують власного ім'я. В основному це пов'язано з їх викликом у потрібний час[2].

1.3.2 Конкурентні структури

Мережа Кохонена або само організаційна (функціональна) карта. Ця мережа використовує конкурентне навчання для класифікації даних без нагляду. Потрібно ввести дані в мережу, після чого мережа оцінює, які з нейронів найбільш точно відповідають введеним даним. Потім ці нейрони налаштовуються, щоб відповідати введеним даним ще більше, перетягують дані у сусідніх нейронах[3].

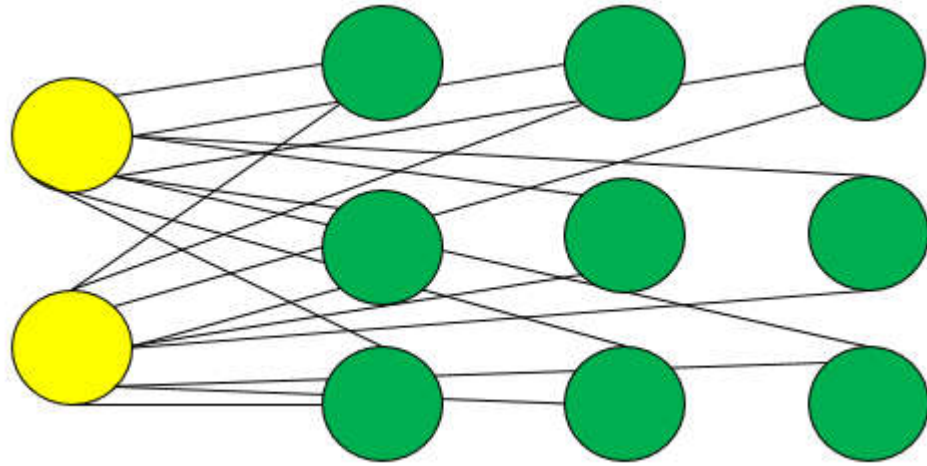


Рисунок 1.7 – Мережа Кохонена

Центральні нейронні системи (ЦНС), як правило, вистрілюють досить регулярно, що залежить від інтегральної пресинаптичної передачі. Це дає можливість припустити, що диференціальні зміни в постсинаптичному потенціалі додано лінійно. Загальна середня тривалість запуску η нейрона потім виражається, як

$$\eta = \sigma[\sum_j \beta_j \xi_j], \quad [1.1]$$

де $\sigma[\]$ визначає характерну функціональну форму, ξ_j – пресинаптичний імпульс частоти всіх синапсів, β_j – синаптичний ефект[3].

Розглянемо масив головних нейронів, як показано на рисунку 1.8., інтернейрони не були чітко показані але проявляються через бічні муфти. Відповідно до формули 1.1 ми будемо писати для кожного виходу,

$$\eta_i(t) = [\varphi_i(t) \sum_{k \in S_i} \gamma_k \eta_k(t - \Delta t)], \quad [1.2]$$

де $\varphi_i(t)$ – інтегрована деполяризація, що викликає всі зовнішні входи і репрезентує входи через бічні муфти. S_i тут набір клітин підключених до осередків. І коефіцієнт γ_k який залежить не тільки від синаптичних ефектів, але і від тих бічних з'єднань і повинен грубо залежати від відстані.

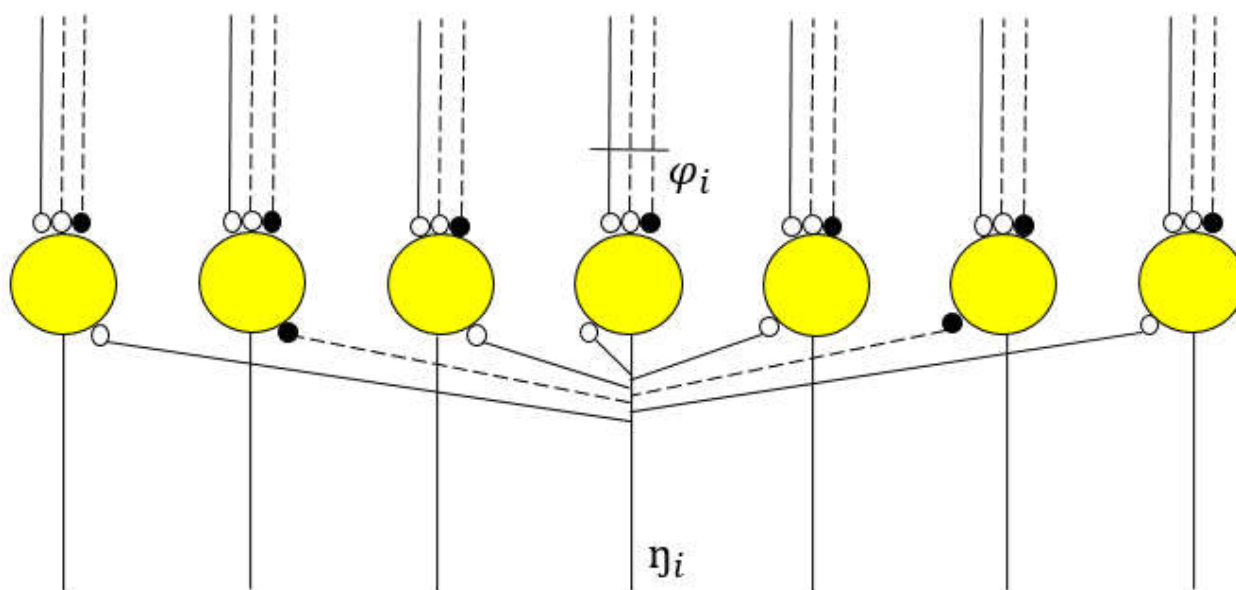


Рисунок 1.8 – Схематичне зображення бічного з'єднання, яке може реалізувати функцію, показану на малому колі.

Синаптична передача та латентність затримуються Δt і бокові люфти вважаються ідентичними, тому що їх варіації тут немає жодного сенсу, це потрібно перш за все для вивчення повторювальних процесів, в яких $\eta_i(t)$ осідають до своїх асимптотичних значень за часом[3].

1.3.3 Рекурентні структури

Рекурентні нейронні мережі(РНМ) – це напрямлені нейронні мережі з часом, вони не є апатридами, вони мають зв'язок між проходами і з'єднаннями через час. Нейрони подають інформацію не тільки з попереднього рівня, але і від себе з попереднього проходу. Це означає, що порядок, в якому ви подаєте на вхід і вчите мережу, має значення: подайте спочатку «молоко», а потім «печиво», можна отримати інші результати порівняно з подачею «печиво», а потім «молоко». [4]

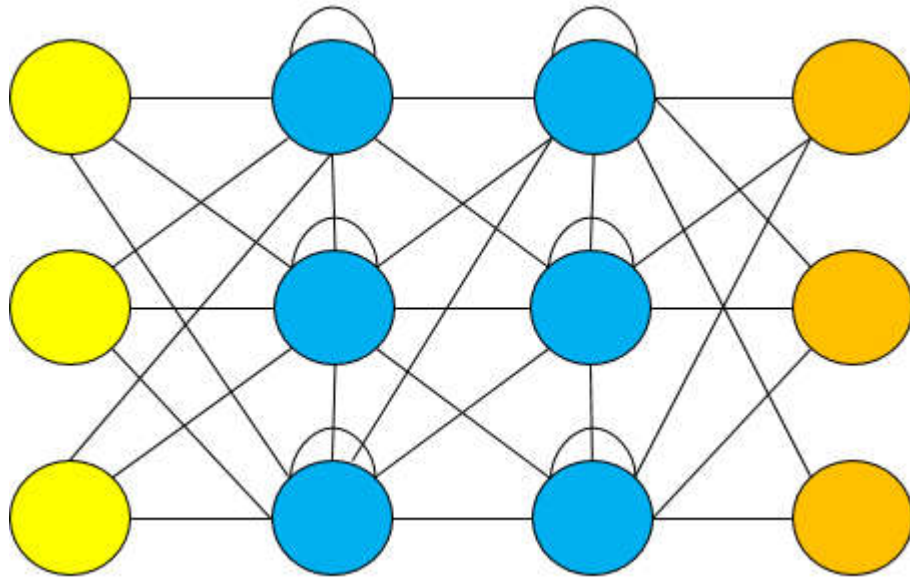


Рисунок 1.9 – Рекурентна нейронна мережа

Одна з великих проблем РНМ – це проблема з градієнтом, що зникає, коли залежно від використовуваних функцій актуальність інформації швидко втрачається з часом. Звичайно це не було б серйозною проблемою, оскільки це значення а не нейронні стани, але це значення з часом є фактично місцем збереження інформації; якщо значення досягне 0 або 1 000 000, то попередній стан вже не буде інформативним.

РНМ можуть використовуватися у багатьох областях, оскільки більшість форм даних, які насправді не мають часової шкали, можуть бути представлені як послідовність. Картинку або рядок тексту можна одночасно подати як один піксель або символ, тому залежність від часу значення використовують для того, щоб знати що було раніше в послідовності, а не те що сталося секундою раніше. Загалом, рекурсивні мережі – це хороший вибір для просування або заповнення інформації, наприклад, автозавершення[4].

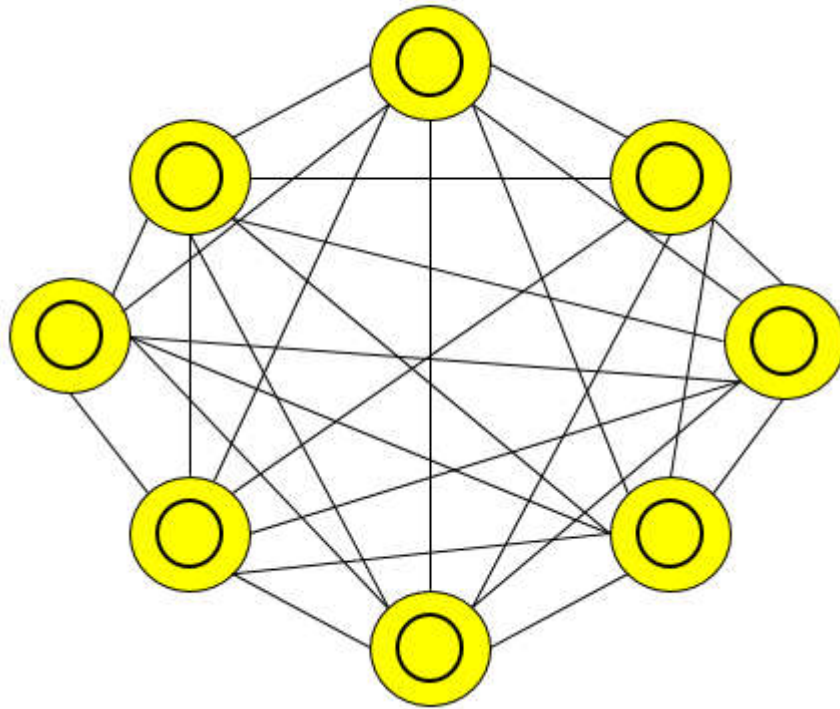


Рисунок 1.10 – Мережа Хопфілда

Мережа Хопфілд (МХ) – це мережа, де кожний нейрон підключається до іншого нейрону. Мережі навчаються, встановлюючи значення нейронів до бажаного рисунку, після чого значення можуть бути обчислені. Після навчання однієї або декількох моделей, мережа завжди збігатиметься з однією із вивчених моделей, оскільки мережа стабільна тільки в таких станах. Зауважте, що це не завжди відповідає бажаному стану. Частково стабілізується через загальну «енергію» або «температуру» мережі, що поступово зменшується під час навчання. Кожний нейрон має поріг активації, який дорівнює цій температурі, якщо вона перевищує сумування вхідного сигналу, то нейрон має прийняти форму одного з двох станів (зазвичай – 1 або 1, інколи – 0 або 1)[5].

Оновлення мережі може виконуватися синхронно або частіше один за одним. Якщо оновлено один за одним, то створюється чесна випадкова послідовність для того, щоб організувати, які клітинки оновлюються в якому порядку (справедливо випадкові всі параметри (n), що відбуваються рівно раз на кожні n елементів). Таким чином, ви можете визначити, коли мережа стабільна (виконується конверсіювання), коли кожна комірка була оновлена, і жодна з них не була змінена, мережа стабільна. Ці мережі часто називають асоціативною

пам'яттю, оскільки вони сходяться до найбільш схожих станів на вході; якщо люди бачать половину столу, то ми можуть зобразити іншу половину, ця мережа буде робити так само[5].

1.4 Операційний базис нейронних мереж

На сучасному етапі розвитку нейромережевих технологій відбувається розширення галузей застосування, в значній частині з яких потрібно розв'язувати задачі у реальному часі на апаратних засобах, що відповідають обмеженням щодо енергоспоживання, габаритів, часу та вартості розроблення. Створення високоефективних нейромережевих засобів реального часу потребує широкого використання сучасної елементної бази (надвеликих інтегральних схем (НВІС)), розроблення нових методів і алгоритмів опрацювання даних у реальному часі, орієнтованих на апаратну реалізацію.

Аналіз операційного базису нейромереж показує, що нейромережеві операції за кількістю операндів, що одночасно опрацьовуються, можна розділити на одно- (корінь квадратний, передатні функції), дво- (додавання, ділення, множення) і багатооперандні (визначення мінімального та максимального чисел, багатооперандне підсумовування, обчислення скалярного добутку, обчислення суми квадратів різниць). Відомі апаратні нейроелементи та нейромережі в основному є одно- і двооперандними, це пов'язано з можливостями елементної бази. Еволюція розвитку архітектури нейроелементів та нейромереж тісно пов'язана з структурною одиницею опрацювання, тобто з розрядністю і кількістю операндів, які одночасно опрацьовує операційний пристрій. З розвитком інтегральної технології з'явилася тенденція зміни структурної одиниці опрацювання з одно- та двооперандної на багатооперандну, яка виконується паралельно. Особливістю багатооперандних нейрооперацій є те, що вони виконуються над множиною операндів і результатом операції є одне число.

Багатооперандні нейрооперації пропонується виконувати на основі багатооперандного підходу, при якому процес обчислення нейрооперації розглядається як виконання єдиної операції, що ґрунтується на елементарних арифметичних операціях.

Паралельна НВІС-реалізація нейроелементів і нейромереж на основі багатооперандного підходу потребує великих затрат обладнання і значної кількості виводів інтерфейсу, які залежать як від кількості операндів, так і від їхньої розрядності. Вартість і швидкодія паралельних НВІС-реалізацій нейроелементів і нейромереж істотно залежить як від рівня технології, так і від кількості виводів інтерфейсу, які визначають розмір кристала. Для забезпечення високої швидкодії, зменшення кількості виводів інтерфейсу та розрядності міжнейронних зв'язків пропонується опрацювання даних у нейромережах здійснювати паралельно розрядними зрізами на основі багатооперандного підходу, тобто паралельно-вертикальними методами. На основі таких методів опрацювання даних розробляються апаратні нейроелементи та нейромережі, які мають архітектуру паралельно-вертикального типу. Тому метою дослідження є розроблення паралельно-вертикальних методів опрацювання даних у реальному часі, орієнтованих на НВІС-реалізацію.

Особливості нейротехнологій та операційний базис спеціалізованих модулів. Аналіз задач і галузей застосування нейротехнологій показав, що вони мають такі особливості:

- високу інтенсивність та постійність вхідних потоків даних;
- постійне ускладнення алгоритмів обробки та підвищення вимог до точності результатів;
- можливість розпаралелення обробки як у часі, так і у просторі;
- здатність до узагальнення та абстрагування;
- здатність до навчання, самонавчання та самоорганізації під впливом зовнішнього середовища.

Для реалізації спеціалізованих модулів виділений операційний базис, який наведений на рисунку 1.9.

Із рисунку 1.9 видно, що нейромережний операційний базис спеціалізованих модулів складається із трьох груп базових операцій:

- попередньої обробки;
- процесорних операцій;
- елементарних функцій.

Група базових операцій попередньої обробки забезпечує перетворення початкових даних для кращого сприйняття нейромережею. В групу базових операцій попередньої обробки входять такі операції: обчислення максимальних і мінімальних чисел, ділення, добування квадратного кореня, обчислення суми квадратів різниць.

Навчання мережі на «сирому» наборі, як правило, не дає якісних результатів. Існує ряд способів покращити «сприйняття» мережі:

- нормалізація виконується тоді, коли на різні входи мережі подаються дані різної розмірності. Наприклад, на перший вхід мережі подаються величини зі значеннями від нуля до одиниці, а на другий – від ста до тисячі. При відсутності нормування значення на другому вході завжди будуть мати набагато більший вплив на вихід мережі, ніж значення на першому вході. При нормалізації розмірності всіх вхідних та вихідних даних зводяться до одного діапазону;
- квантування виконується над неперервними величинами, для яких виділяється скінченний набір дискретних значень. Наприклад, квантування використовуються для задання частот звукових сигналів при розпізнаванні мов;
- фільтрація виконується для «зашумлених» даних і полягає у відкиданні значень, які, швидше за все, є некоректними.
-

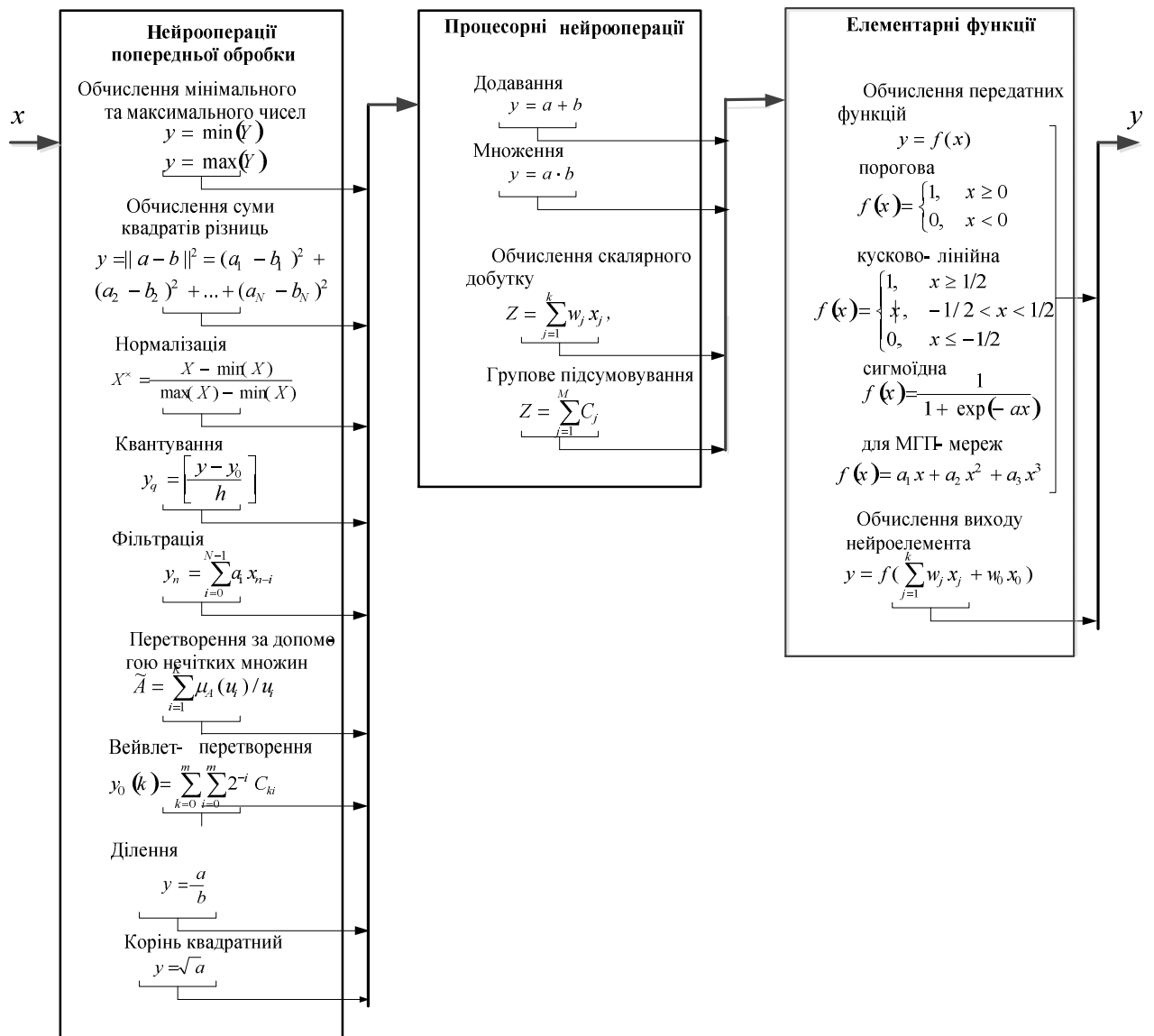


Рисунок 1.11 – Нейромережевий операційний базис

Група процесорних операцій складається із таких операцій:

- обчислення скалярного добутку;
- групового підсумовування.

Серед усієї сукупності операцій, які найчастіше використовуються у нейроалгоритмах, особливої уваги заслуговує операція обчислення скалярного добутку. Традиційно обчислення такої операції здійснюється за наступною формулою:

$$Z = \sum_{j=1}^k w_j x_j, \quad (1.3)$$

де k – кількість входів нейроелемента, W_j – j -й ваговий коефіцієнт, X_j – значення j -го входу.

Існують два підходи до апаратної реалізації обчислення сум парних добутків. Перший з них ґрунтується на операціях множення і додавання, другий – на операціях додавання, інверсії та зсуву. Перший підхід переважно використовують при синтезі пристроїв обчислення сум парних добутків на базі окремих мікросхем (помножувачів, суматорів), а другий – при НВІС-реалізаціях. Використання для НВІС-реалізацій алгоритмів на базі операцій додавання, інверсії та зсуву дозволяє оптимізувати пристрій за швидкістю, апаратними витратами та збільшити регулярність його структури. Основою таких алгоритмів є формування часткових добутків з наступним їх додаванням. У загальному випадку обчислення скалярного добутку в базисі елементарних арифметичних операцій зводиться до макрооперації групового підсумовування часткових добутків:

$$Z = \sum_{j=1}^M C_j, \quad (1.4)$$

де M – кількість доданків; C_j – j -й доданок [3].

Група операцій елементарних функцій складається із операцій: обчислення уніполярної та біполярної порогової функції, гіперболічного тангенса та синусоїдальних функцій з насиченням. В нейроні за допомогою даних операцій реалізується функція активації.

2 АЛГОРИТМИ РЕАЛІЗАЦІЇ НЕЙРОМЕРЕЖЕВИХ БАГАТООПЕРАНДНИХ ОПЕРАЦІЙ

2.1 Виділення нейромережєвих багатооперандних операцій

Комп'ютерна реалізація алгоритмів штучних нейронних мереж пов'язана з вибором засобів реалізації: апаратних, програмних або мікро програмних.

За допомогою алгоритмів програмної реалізації ШНМ більшість обчислювальних процесів розгортаються в часі з великим об'ємом пересилання інформації між операційними пристроями і оперативною пам'яттю. Для програміста є доступними усі програмні засоби. Задачею для нього є мінімізація об'єму програм і часу їх реалізації при заданій точності обчислень. Характерною ознакою вказаних засобів є низька швидкодія і гнучкість з точки зору заміни алгоритмів та можливості їх модифікації.

Мікро програмна реалізація обчислень передбачає їхнє розгортання як в часі, так і в просторі. Під час мікро програмування є доступ до системи мікропрограм процесора. Цей доступ може забезпечити застосування постійної пам'яті, оперативних запам'ятовуючих пристроїв, а також програмних логічних матриць, які використовуються в якості пам'яті мікропрограм. Прикладом мікро програмної реалізації є алгоритми штучних нейронних мереж на базі однорідних обчислювальних середовищ (ООС). Двовимірна регулярна матриця процесорних елементів (ПЕ), кожен з яких фізично зв'язаний входом або виходом з чотирма іншими – знизу, зверху, праворуч та ліворуч називають процесором на базі ООС. Кожний процесорний елемент може виконувати набір бітових операцій для перетворення інформації з вхідних каналів у вихідні. Такий процес на базі однорідних обчислювальних середовищ є універсальною системою, тому що за його допомогою можна реалізувати будь яку обчислювальну функцію. ПЕ мають бітовий рівень та повну систему комутації, що дозволяє реалізувати розпаралелення обчислень на найнижчому бітовому рівні. Саме це є перевагою мікро програмної реалізації алгоритмів штучних нейронних мереж. Реалізація в

повній мірі усіх можливостей мікро програмних засобів може бути здійснена лише при глибокому вивченні задач, які вирішуються, так і внутрішньої мови процесора. Якщо порівнювати мікро програмні засоби реалізації з програмними, то мікро програмні є більш швидкодіючими.

Досягнення в області інтегральної технології дозволяють все більше перекладати реалізацію алгоритмів штучних нейромереж на апаратні засоби, що розгортають обчислення не тільки у часі, а й у просторі. Такі обчислення характеризуються введенням додаткового обладнання і відсутністю проміжного пересилання інформації в процесі обчислення, а також спрощенням функції місцевого управління. В основі структурної організації апаратних засобів лежить принцип зображення графів алгоритмів штучних нейронних мереж та матриця, елементи якої виконують з'єднані між собою графи алгоритмів та функціональні оператори. Синтезовані таким чином структури є алгоритмічними. В таких структурах алгоритм виконується над усіма вхідними даними, але за умови, що вони уже раз пройшли через усі програмовані елементи. За режимом роботи такі структури можна поділити на синхронні та асинхронні. Асинхронні структури ще називають одноктактними. Обробка даних в них здійснюється без проміжного запам'ятовування. Ефективність таких структур отримується в результаті визначення часу виконання найбільш складних операторів алгоритму, які обробляються разом із даними. Всі такі одноктактні структури є послідовними з точки зору реалізації функціональних алгоритмів. Таким чином одноктактність стає причиною обмеження швидкості та неефективність використання пристроїв які обробляють усі потоки даних у реальному часі. В цьому випадку для обробки даних ефективно використовувати синхронні структури, які конвеєрно реалізують графи алгоритмів, що передбачає зміщення у часі під час обробки операторами алгоритму даних. Апаратні засоби дозволяють максимально швидко реалізувати швидкодію алгоритмі штучних нейронних мереж, а також зменшити витрати на експлуатацію та розробку програм і зробити їх доступними тільки для розробників архітектури нейромереж.

Потрібно відзначити, що усі види реалізації алгоритмів в безпосередньому вигляді зустрічаються дуже зрідка. Для реалізації алгоритмів ШНМ використовують комбіновані переваги алгоритмів та засоби перерахованих вище. Переваги того чи іншого засобу визначають за допомогою вимог, до ШНМ за швидкодією, потужністю, габаритами та ціною.

Зараз розглянемо варіанти побудови штучних нейронних мереж реального часу:

- за допомогою спеціальних мікропроцесорів шляхом розробки спеціального програмного забезпечення;
- за допомогою універсального ядра, яке доповнене програмними компонентами, що реалізують складні алгоритми функціонування мереж;
- за допомогою спеціальних алгоритмічних систем у яких архітектура і обчислювальний процес відображає структуру алгоритму розв'язання задачі.

Перший варіант є досить простим і доступним для більшості користувачів. Очевидною його перевагою є те, що він може використовувати програми, які були розроблені ще пізніше. Невисока швидкодія і надлишковість комп'ютерних засобів можна віднести до мінусів цього способу.

Другий варіант має великі перспективи, оскільки він передбачає поєднання універсальних і спеціальних засобів. За допомогою цього поєднання можна забезпечити високу ефективність використання устаткування при створенні систем для опрацювання у реальному часі потоків даних за алгоритмами, що не можуть регулювати високу кількість логічних операцій. Але для того щоб розробити нові апаратні засоби з певними технічними параметрами потрібно доповнювати алгоритми новими апаратно-програмними компонентами.

Третій варіант надає перевагу на обробку у реальному часі складних алгоритмів, які будуть обробляти великий потік даних. За допомогою комбінування обчислювальних можливостей апаратних засобів з надходженням потоків даних досягається висока ефективність.

2.2 Алгоритм обчислення скалярного добутку

Існують два підходи до апаратної реалізації алгоритмів обчислення скалярних добутків. Перший з них ґрунтується на операціях множення, додавання, другий - на операціях додавання, інверсії та зсуву.

Перший підхід переважно використовують при синтезі пристроїв обчислення скалярних добутків на базі окремих мікросхем (помножувачів, суматорів), а другий - при НВІС-реалізаціях. Використання для НВІС-реалізацій алгоритмів на основі операцій додавання, інверсії та зсуву дає змогу оптимізувати пристрій за швидкодією, апаратними витратами та збільшити однорідність і регулярність структури.

Обчислення скалярних добутків у базисі елементарних операцій зводиться до формування і підсумовування часткових добутків. Особливістю існуючих методів обчислення скалярних добутків у базисі елементарних операцій є те, що вони ґрунтуються на формуванні і підсумовуванні часткових добутків для кожної пари операндів. НВІС-реалізація паралельного обчислення скалярних добутків на основі існуючих методів і алгоритмів вимагає великих затрат обладнання і значної кількості виводів інтерфейсу, які залежать як від кількості операндів, так і від їх розрядності. Вартість і швидкодія спеціалізованих НВІС для паралельного обчислення скалярних добутків значною мірою залежить як від кількості, так і від якості виводів інтерфейсу. Для спеціалізованих НВІС число зовнішніх виводів залежить від рівня технології та розміру кристала, а якість виводів інтерфейсу визначається затримкою переключення зовнішніх зв'язків, які навантажені на ці виводи.

Для обчислення скалярних добутків у базисі елементарних операцій будемо використовувати багатооперандний вертикально-паралельний підхід на основі попередніх обчислень, який передбачає паралельно-порозрядне надходження операндів, формування і підсумовування макро- часткових добутків. Особливістю методів і алгоритмів обчислення скалярних добутків за таким підходом є

формування і підсумовування макрочасткових добутоків, кількість яких залежить не від кількості операндів, а від їх розрядності.

Під час вертикально-паралельного обчислення скалярного добутку на основі попередніх обчислень необхідно, щоб множники W_j і X_j були представлені в порозрядному вигляді згідно з формулою:

$$W_j = \sum_{i=1}^n 2^{-i} W_{ji}, X_j = \sum_{i=1}^n 2^{-i} X_{ji}, \quad (2.1)$$

де W_{ji} і X_{ji} - значення i -х розрядів множників W_j і X_j - розрядність множників. Тоді обчислення скалярного добутку можна записати так:

$$S = \sum_{j=1}^k W_j X_j = \sum_{i=1}^n 2^{-i} \sum_{j=1}^k W_j X_{ji} = \sum_{i=1}^n 2^{-i} \sum_{j=1}^k P_{ji} = \sum_{i=1}^n 2^{-i} P_{Mi} \quad (2.2)$$

де P_{ji} - i -й частковий добуток, P_{Mi} - i -й макрочастковий добуток, який формується додаванням k часткових добутоків.

Зменшити в γ разів кількість часткових добутоків P_{ji} , які використовуються для формування макрочасткового добутку P_{Mi} , можна шляхом використання попередніх обчислень.

Обчислюють макрочастковий добуток P_{mi} на основі групових часткових добутоків P_{si} так:

$$Z = \sum_{s=1}^m P_{si}, \quad (2.3)$$

Для обчислення групового часткового добутку P_{si} використовують попередні обчислення, які здійснюються на основі однорозрядних суматорів.

Обчислення макрочасткового добутку P_{mi} на основі групових часткових добутоків P_{si} забезпечує зменшення в γ разів кількості входів багатовходового суматора. Збільшення кількості множників у групі вимагає збільшення кількості попередніх обчислень, але водночас зменшує кількість входів багатовходового

суматора. Вибір кількості множників у групі g залежить насамперед від кількості входів нейроелемента k .

Для найповнішого використання переваг сучасної НВІС-технології та базису елементарних арифметичних операцій в основу побудови вертикально-паралельного пристрою для обчислення скалярного добутку на основі попередніх обчислень доцільно покласти такі принципи:

- узгодженості інтенсивності надходження даних з обчислювальною інтенсивністю пристрою;
- конвеєризації та просторового паралелізму;
- регулярності, модульності та широкого використання стандартних елементів;
- локалізації та зменшення кількості зв'язків між елементами пристрою.

Підвищення ефективності використання обладнання в вертикально-паралельному пристрої для обчислення скалярного добутку на основі попередніх обчислень можна досягти зменшенням:

- часу формування групових часткових добутків;
- кількості групових часткових добутків;
- часу підсумовування групових часткових добутків.

Залежно від алгоритмів формування групових часткових добутків та вимог конкретного застосування можна синтезувати багато структур пристроїв паралельно-вертикального обчислення скалярних добутків, які відрізняються як організацією обчислень, так і технічними характеристиками. Тому доцільно розробити та дослідити базову структуру пристрою паралельно-вертикального обчислення скалярних добутків, яка буде основою для синтезу конкретного пристрою обчислення сум парних добутків із заданими характеристиками.

Базову структуру пристрою для обчислення скалярного добутку на основі попередніх обчислень наведено на рис.2, де k - кількість входів нейроелемента, g - кількість множників у групі, V_x - інформаційні входи, V_xPP - вхід задання режиму роботи, БПО - блок попередніх обчислень, БП - блок пам'яті, ФГЧД - блок формування групового часткового добутку, БСм - багатовходовий суматор, $RgZp$ -

регістр розрядних зрізів множників, $R_{гМЧД}$ - регістр макрочасткових добутоків, $См$ - суматор, $R_{гР}$ - регістр результатів, Вих - вихід.

Обчислення скалярного добутку у цьому пристрої здійснюється в два етапи. На першому етапі за n тактів, починаючи з молодших розрядів W_j , за допомогою блоку БПО виконуються попередні обчислення. Результати попередніх обчислень та вхідні дані запам'ятовуються у блоках пам'яті БП.

На другому етапі в кожному l -му такті ($l=1, \dots, n$) i -ті розряди множників $X_{1l}, X_{2l}, \dots, X_{kl}$ записуються в регістр $R_{гЗр}$. Інформація з цього регістру керує формуванням групових часткових добутоків відповідно до формули (5). Сформовані групові часткові добутки P_{Fsl} надходять на багатовходовий суматор $БСм$, на виході якого формується макрочастковий добуток P_l , який записується у регістр $R_{гМЧД}$. Сформований макрочастковий добуток P_l додається до попередньо накопиченої суми, зсунутої на один розряд вправо.

Варто зауважити, що перший етап обчислення виконується тільки у процесі навчання нейронної мережі, коли оптимізуються значення вагових коефіцієнтів (множники W_l). У процесі функціонування нейронної мережі виконується тільки другий етап обчислення, завдяки чому значно підвищується швидкість роботи мережі.

У випадку, коли $г=k$, всі можливі комбінації множників будуть попередньо обчислені та збережені у пам'яті. Тобто, обчислюють скалярний добуток табличним методом, який доцільно використовувати у певних випадках. При цьому у пристрої відпадає потреба використовувати блоки формування часткового добутку і багатовходовий суматор.

Основними компонентами розробленого пристрою є тригери, регістри, суматори, багато- входові суматори та дешифратори. Оскільки розроблений пристрій для обчислення скалярного добутку на основі попередніх обчислень орієнтований на НВІС-реалізацію, то за одиницю вимірювання витрат обладнання візьмемо логічний вентиль, який є елементом типу інвертор, І, АБО.

Аналітичний вираз розрахунку витрат обладнання залежно від кількості входів k , розрядності n та кількості множників у групі $г$ для реалізації розробленого пристрою має такий вигляд:

$$W_{\Pi} = 9k * 2^{r-1}(n + 2) + \frac{k}{r}(11n - 18) + 14n - 11k \quad (2.4)$$

На основі цього аналітичного виразу для розрядності операндів $n=16$ будуюмо графік витрат обладнання залежно від кількості операндів k та кількості множників у групі $г$.

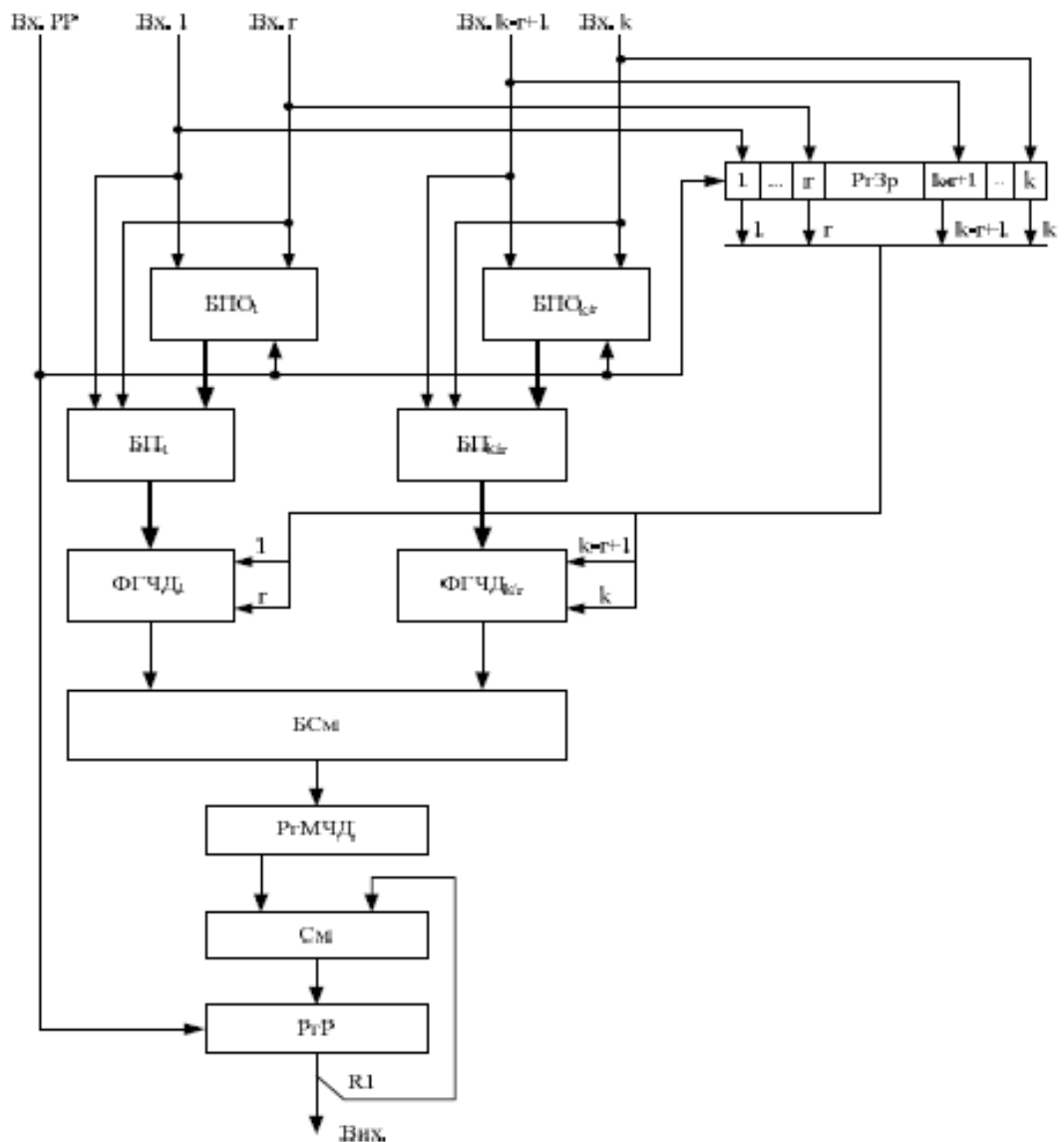


Рисунок 2.1 - Базова структура пристрою для обчислення скалярного добутку на основі попередніх обчислень

Як видно з графіків, незалежно від кількості входів та розрядності найефективнішою є реалізація пристрою для обчислення скалярного добутку на основі попередніх обчислень з групуванням по два множники. Саме у цьому випадку затрати обладнання будуть найменшими.

Тому доцільно розробити структуру пристрою для обчислення скалярного добутку на основі попередніх обчислень з групуванням по два множники. Обчислення скалярного добутку в такому пристрої здійснюється в два етапи. На першому етапі за n тактів (де n – розрядність операндів W_i і X_i), починаючи з молодших розрядів виконується порозрядне накопичення множених W_i та обчислення сум $W_1+W_2, \dots, W_{k-1}+W_k$.

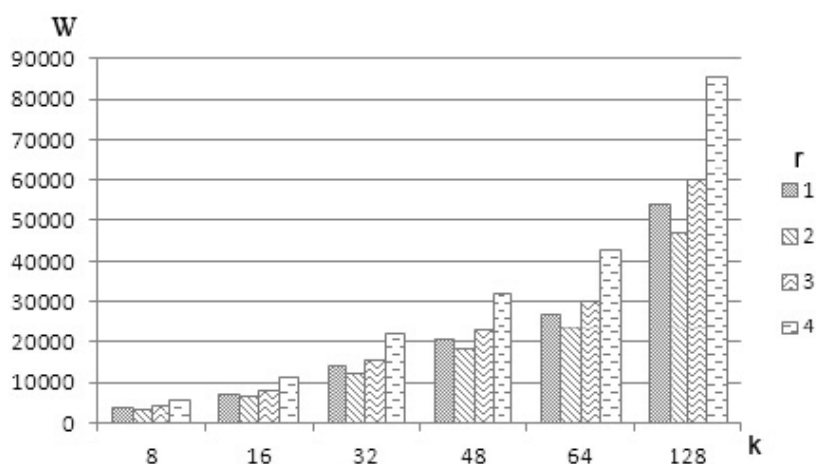


Рисунок 2.2 - Графік витрат обладнання на реалізацію пристрою обчислення скалярного добутку на основі попередніх обчислень залежно від кількості операндів k та кількості множників у групі r

На другому етапі в кожному такті i ($i=1, \dots, n$) для i -х розрядів множників X_1, X_2, \dots, X_k починаючи з молодших розрядів формують групові часткові добутки P_{si} для пар добутків $W_1X_1 + W_2X_2, \dots, W_{k-1}X_{k-1} + W_k X_k$ відповідно до формули, яка у цьому випадку набуде вигляду:

$$P_{si} = \begin{cases} 0, & \text{якщо } X_{(2j-1)i} = X_{2ji} = 0 \\ W_{2j-1}, & \text{якщо } X_{(2j-1)i} = 1, X_{2ji} = 0 \\ W_{2j}, & \text{якщо } X_{(2j-1)i} = 0, X_{2ji} = 1 \\ W_{2j-1} + W_{2j}, & \text{якщо } X_{(2j-1)i} = X_{2ji} = 1 \end{cases} \quad (2.5)$$

2.3 Алгоритм групового підсумовування

До макрооперації групового підсумовування :

$$Y = \sum_{j=1}^m C_j \quad (2.6)$$

де m - кількість додатків; C_j - j -ий доданок. Нехай доданки C_j є додатними числами меншими одиниці, що задані в двійковій системі числення з фіксованою комою n -розрядним кодом :

$$C_j = \sum_{i=1}^n C_{ji} 2^{-i} \quad (2.7)$$

Підставивши значення (2.6) в формулу (2.7) , отримуємо:

$$Y = \sum_{j=1}^m \sum_{i=1}^n C_{ji} 2^{-i}$$

Питання, що пов'язані з часом формування часткових добутоків і їх кількістю, є взаємно пов'язані та детально розглянуті в [12,13]. Тому подальше підвищення ефективності реалізації БО швидких алгоритмів ОТП в основному пов'язується з ефективною реалізацією макрооператора групового підсумовування. Одним із методів ефективної реалізації макрооператора групового підсумовування є реалізація його як єдиного неперервного процесу додавання з врахуванням одиниць переносів тільки на заключних етапах отримання результатів. За цим методом підсумовування здійснюється шляхом перетворення багаторядного коду в дворядний, базовою операцією якого є операція перетворення трирядного коду в дворядний:

$$E = \begin{matrix} \left[\begin{array}{cccc} C_{(j-1)1} & \dots & C_{(j-1)(n-1)} & C_{(j-1)n} \\ + & & & \\ C_{j1} & \dots & C_{j(n-1)} & C_{jn} \\ + & & & \\ C_{(j+1)1} & \dots & C_{(j+1)(n-1)} & C_{(j+1)n} \end{array} \right] = \left[\begin{array}{cccc} 0 & S_1 & \dots & S_{n-1} & S_n \\ + & & & & \\ P_0 & P_1 & \dots & P_{n-1} & 0 \end{array} \right] \end{matrix} .()$$

Перетворення трирядного коду в дворядний здійснюється за допомогою шару однорозрядних суматорів, що не мають зв'язків між собою. Для зменшення

часу додавання шари однорозрядних суматорів необхідно об'єднати за принципом дерева Уоллеса [12,13]. Кількість шарів однорозрядних суматорів для обчислення макрооперації групового підсумовування визначається за формулою:

$$K = \lceil \log_{1,5} 0,5m \rceil \quad (2.8)$$

де $\lceil \rceil$ - операція округлення до більшого числа.

Серед всієї сукупності операцій, які найчастіше використовуються в алгоритмах цифрової обробки сигналів і нейроалгоритмах, особливої уваги заслуговує операція обчислення сум парних добутоків [13]. Традиційно обчислення такої операції здійснюється за наступною формулою:

$$Z = \sum_{j=1}^m A_j B_j = A_1 B_1 + A_2 B_2 + \dots + A_m B_m \quad (2.9)$$

При обчисленні сум парних добутоків за формулою (2.9) потрібно виконати m множень і $(m-1)$ додавань. У значній частині застосувань обчислення сум парних добутоків необхідно виконувати над інтенсивними потоками даних в реальному масштабі часу на апаратних засобах з високою ефективністю використання обладнання. Забезпечити такі вимоги можливо шляхом використання конвеєрних паралельно-потоківих НВІС-пристроїв, структура яких відображає алгоритм обчислення сум парних добутоків та враховує інтенсивність надходження потоків даних:

Для забезпечення обробки потоків даних у реальному часі необхідно узгодити інтенсивність надходження даних P_d з пропускною здатністю конвеєра НВІС-пристрою $D_k = F_k m h$, де F_k – тактова частота роботи конвеєра, m - кількість трактів обробки, h – кількість розрядів для яких формуються часткові добутки. У випадку коли m величина постійна, то таке узгодження досягається шляхом зміни частоти роботи конвеєра F_k або вибором необхідної кількості розрядів h для одночасного формування часткових добутоків. Задача проектування НВІС-

структур обчислення сум парних добутків з високою ефективністю використання зводиться до мінімізації апаратних затрат при забезпеченні режиму реального часу.

Існують два підходи до апаратної реалізації обчислення сум парних добутків[13]. Перший з них ґрунтується на операціях множення, додавання, другий – на операціях додавання, інверсії та зсуву. Перший підхід, переважно, використовують при синтезі пристроїв обчислення сум парних добутків на базі окремих мікросхем (помножувачів, суматорів), а другий – при НВІС-реалізаціях. Використання для НВІС-реалізацій алгоритмів на базі операцій додавання, інверсії та зсуву дозволяє оптимізувати пристрій за швидкістю, апаратними витратами та збільшити регулярність його структури. Основою таких алгоритмів є формування часткових добутків з наступним їх додаванням. Існуючі алгоритми та структури обчислення сум парних добутків на базі елементарних операцій не враховують інтенсивність надходження і величину зміни вхідних даних.

У своїй простій формі (на рисунку 2.3) суматор має 4 сигнальних лінії: пара входів для сигналів, що представляють однорозрядні двійкові числа "А" і "В", вихід "S" (сума "А" і "В") і сигнал перенесення "М". Ця схема, яка забезпечує складання одnobітових чисел "А" і "В", при цьому "S" найменш значущий біт, "М" найбільш значущий біт, називається "півсуматор".

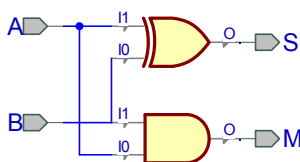


Рисунок 2.3 - Схема однорозрядного напівсуматора

Задіявши додаткову схему "АБО", можна підключити дві ці "половинки" разом (на рисунку 2.4 і створити нову схему з додатковим входом, "m", який приймає сигнал перенесення старшого біта з попередньої схеми. Таке складання двох "півсуматорів" дає повний суматор.

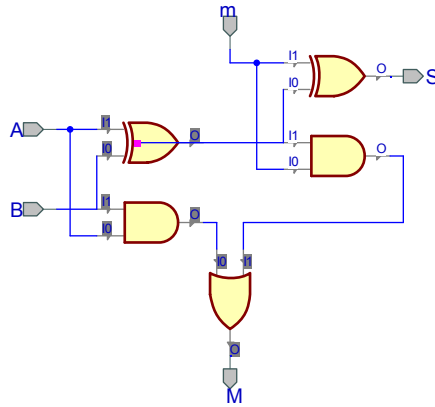


Рисунок 2.4 - Схема повного однорозрядного суматора

Схема повного суматора може бути використана як "будівельні блоки" для побудови схем, шляхом додавання двійкових чисел з необмеженим числом біт, як показано на рис. 5.8, а також для побудови багатооперандних суматорів

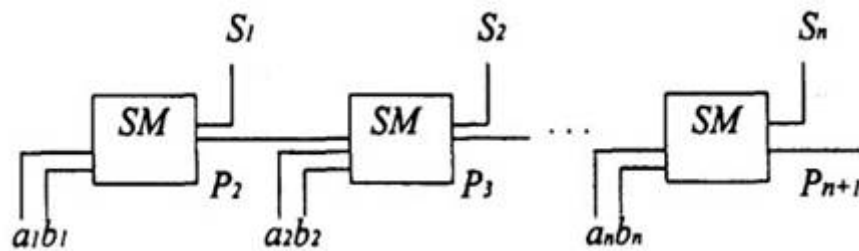


Рисунок 2.5 - Структурна схема багаторозрядного суматора

Розглянемо два методи проектування швидкодіючих багатооперандних суматорів, які визначаються їх структурними схемами і способами організації перенесення.

Широко поширена структура суматора (його каскаду) представлена на рис. 5.9.а. Тут БСФ - блок симетричних функцій; БС - блок суми; БП - блок перенесення; $V_1..V_m$ - доданки; S' - часткова сума (без урахування перенесення); P - перенесення в $(2,.., k)$ - і розряди; $S_{вх}$ - вхідні перенесення в $(1, 2,.., l)$ - й розряди; $S_{вих}$ - вихідні перенесення з $(k+1, k+2, .., k+1)$ - го розрядів; S - остаточна сума, рівна (для 1-го розряду маємо $S_1=S'_1$)

$$S=S' \oplus P.$$

При цьому вхідне перенесення розглядається як додатковий доданок з l розрядами. У цій структурі внутрішнє перенесення P є паралельним, а зовнішній (міжкаскадний) C - послідовним; перенесення є позиційними.

У класичній структурі (двооперандний суматор з перенесенням в $(k+1)$ -й розряд) $M=2$ і $l=1$; як СФ використовуються відомі функції генерації і прозорості. Ця структура узагальнена на випадок довільних M і k за рахунок вступу $m+k$ СФ. Як і в класичній структурі, в БС здійснюється порозрядне додавання по модулю два, а в БП формується вихідне перенесення.

3. АПАРАТНА РЕАЛІЗАЦІЯ СТРУКТУРИ БАЗОВИХ КОМПОНЕНТІВ НЕЙРОМЕРЕЖ

3.1 Паралельний вертикально-груповий перетворювач кодів.

Перетворювач кодів повинен забезпечувати паралельне послідовно-групове перетворення, тобто формування групових розрядних зрізів. Паралельне послідовно-групове перетворення здійснюється шляхом послідовного приймання N чисел з подальшим формуванням в кожному такті розрядних зрізів із k розрядів для всіх N чисел. Таке перетворення використовується для завантаження компонент апаратних нейромереж з паралельним вертикально-груповим опрацюванням вхідними даними. При опрацюванні потоків даних в кожному g -у такті роботи на вході компоненти необхідно формувати такі розрядні зрізи з k розрядів для всіх N чисел. Для забезпечення опрацювання неперервних потоків даних розроблено паралельний вертикально-груповий перетворювач кодів, структура якого наведена на **рисунок 3.1**, де БлРг - блок реєстрів, Рг –реєстр, V – режим роботи перетворювача, D_j – вхідні дані.

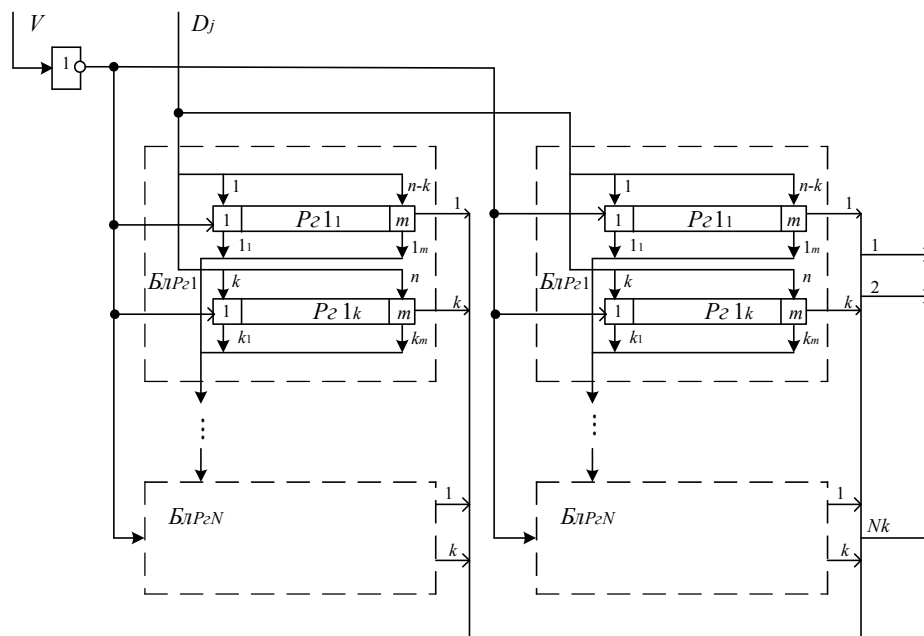


Рисунок 3 . 1. Розроблена структура паралельного вертикально-групового перетворювача кодів

Паралельний вертикально-груповий перетворювач кодів складається з двох груп блоків реєстрів, одна із яких працює на паралельний запис вхідних даних, а друга на порозрядну видачу раніше накопичених даних. Групи реєстрів працюють у двох режимах роботи: $V=0$ - паралельний запис; $V=1$ - порозрядна видача даних. При паралельному записі послідовні виходи реєстрів знаходяться в третьому стані.

3.2 Компонента обчислення максимального і мінімального чисел.

Паралельний вертикально-груповий метод обчислення максимального D_{\max} і мінімального D_{\min} чисел у одновимірному $\{D_k\}_{k=1}^N$ та двовимірному $\{D_{kj}\}_{k=1; j=1}^{N;M}$ масивах передбачає у кожному g -у такті ($g=1, \dots, m$ де $m = \left\lceil \frac{n}{k} \right\rceil$, k - кількість розрядів у групі, n - розрядність чисел) паралельне надходження N чисел старшими розрядами уперед зрізом із k розрядів [12]. Обчислення максимального D_{\max} і мінімального D_{\min} чисел у одновимірному масиві $\{D_k\}_{k=1}^N$ за даним методом ґрунтується на виконанні для кожного r -о розрядного зрізу ($r=1, \dots, k$) однотипних базових макрооперацій, які виконуються на базі трьох простих операцій. При обчисленні максимального числа D_{\max} у одновимірному масиві $\{D_k\}_{k=1}^N$ використовуються такі операції:

- 1) формування значення r -о розрядного зрізу P_r за формулою:

$$P_r = \bigvee_{h=1}^N D_{rh} \wedge y_{rh}, \quad (10)$$

де D_{rh} - значення r -о розряду h -о числа масиву, y_{rh} - значення h -о розряду r -о слова управління, значення 1 -о слова управління дорівнює $y_{11}=y_{12}=\dots=y_{1N}=1$;

- 2) визначення r -о розряду максимального числа $D_{\max r}$ за виразом:

$$D_{\max r} = \begin{cases} 0, & \text{коли } P_r = 0 \\ 1, & \text{коли } P_r = 1 \end{cases}, \quad (11)$$

3) формування h розрядів $(r+1)$ -о слова управління за формулою:

$$y_{(r+1)h} = \begin{cases} 0, & \text{коли } P_r = 1, D_{hr} \neq y_{hr} \\ 1, & \text{коли } P_r = D_{hr} = y_{hr} = 1 \\ y_{hr}, & \text{коли } P_r = 0 \end{cases}. \quad (12)$$

При обчисленні мінімального числа D_{\min} у одновимірному масиві $\{D_k\}_{k=1}^N$ базова макрооперація, реалізується на таких операціях:

1) формування значення r -о розрядного зрізу P_r , яке виконується за формулою:

$$P_r = \bigvee_{h=1}^N \bar{D}_{rh} \wedge y_{rh}, \quad (13)$$

де \bar{D}_{rh} – інверсне значення r -о розряду h -о числа масиву, y_{rh} – значення h -о розряду r -о слова управління, значення 1-о слова управління дорівнює $y_1 = y_2 = \dots = y_N = 1$;

2) визначення r -о розряду мінімального числа $D_{\min r}$ за виразом:

$$D_{\min r} = \begin{cases} 0, & \text{коли } P_r = 1 \\ 1, & \text{коли } P_r = 0 \end{cases}, \quad (14)$$

3) формування h розрядів $(r+1)$ -о слова управління, яке здійснюється за виразом:

$$y_{(r+1)h} = \begin{cases} 0, & \text{коли } P_r = 1, \bar{D}_{hr} \neq y_{hr} \\ 1, & \text{коли } P_r = \bar{D}_{hr} = y_{hr} = 1 \\ y_{hr}, & \text{коли } P_r = 0 \end{cases}. \quad (15)$$

Особливість розглянутого паралельного вертикально-групового методу обчислення максимального (мінімального) числа є те, що у кожному g -у такті роботи визначаються k розрядів максимального D_{\max} (мінімального D_{\min}) числа.

Визначення максимального D_{\max} та мінімального D_{\min} чисел в двовимірному масиві $\{D_{hj}\}_{h=1; j=1}^{N;M}$ ґрунтується на базових макроопераціях, які виконуються за формулами відповідно (10)-(12) та (13)-(15). Різниця визначення максимального D_{\max} (мінімального D_{\min}) числа в двовимірному масиві $\{D_{kj}\}_{k=1; j=1}^{N;M}$ полягає у тому, що

після виконання кожних m тактів над одновимірним масивом із $(N+1)$ чисел до участі в обчисленні долучається максимальне (мінімальне) число нового j -о одновимірного масиву та здійснюється запис одиниць у всі розряди регістра правління. Обчислення максимального (мінімального) числа в двовимірному масиві $\{D_{kj}\}_{k=1; j=1}^{N; M}$ вимагає виконання $M \times m$ базових операцій.

Особливістю паралельного вертикально-групового методу обчислення максимальних і мінімальних чисел у масивах чисел є:

- використання однієї базової макрооперації;
- можливість використання розпаралелення та конвеєризації обчислень;
- можливість одночасного опрацювання N розрядних зрізів;
- час обчислення в основному визначається як кількістю розрядів у групі k , так і розрядністю чисел n , а не їх кількістю N .

Структура компоненти обчислення максимальних і мінімальних чисел. Компонента обчислення максимальних і мінімальних чисел реалізується на базі однотипних ПЕ. Кожний ПЕ апаратно реалізує k базових макрооперацій обчислення максимального і мінімального чисел.

Побудована структура компоненти обчисленням максимальних і мінімальних чисел у одновимірному масиві $\{D_k\}_{k=1}^N$ за паралельним вертикально-груповим методом наведено на рисунку 3.2, де ТІ – тактові імпульси, ПУ – початкова установка, Тг – тригер, Рг – регістр, $D_{h1} - D_{hk}$ - h -й вхід групи із k розрядів, $D_{h1min} - D_{hkmin}$ та $D_{h1max} - D_{hkmax}$ D_{imin} – вихід груп із k розрядів відповідно максимального та мінімального чисел.

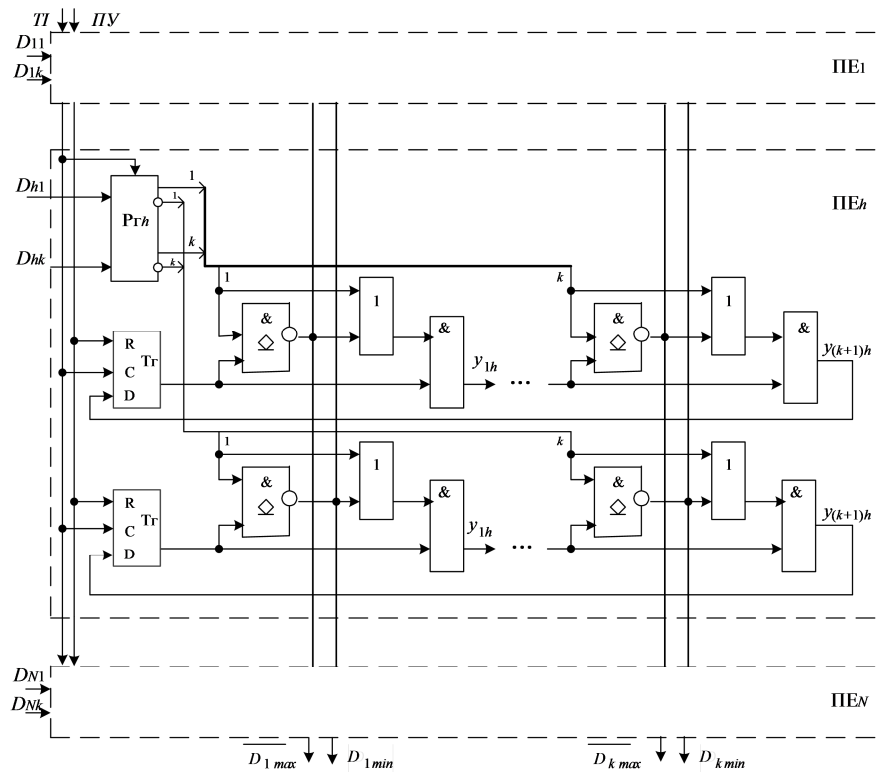


Рисунок 3.2. Структура компоненти обчисленням максимальних і мінімальних чисел за паралельним вертикально-груповим методом

Кількість ПЕ, що підключаються до спільної шини результатів, при одночасному обчисленні максимального та мінімального чисел для одновимірного масиву $\{D_k\}_{k=1}^N$ визначається його розміром. Використання спільних шин результатів забезпечує розпаралелення процесу опрацювання розрядного зрізу, час опрацювання якого визначає такт роботи пристрою. Обчисленням максимальних і мінімальних чисел за паралельним вертикально-груповим методом у такому пристрої здійснюється за час, який визначається за наступним виразом:

$$t_{\text{Обч}} = m(t_{\text{Tr}} + 3kt_1), \quad (16)$$

де t_{Tr} та t_1 час спрацювання відповідно тригера та логічних елементів типу АБО, І, І-НЕ, k – кількість розрядів у групі.

3.3 Компонента обчислення суми квадратів різниць.

Паралельний вертикально-груповий метод обчислення суми квадратів різниць вимагає, щоб кожний операнд був представлений у вигляді груп із k розрядів. При такому представленні операнди записуються так:

$$X_j = \sum_{i=1}^n 2^{-(i-1)} x_{ji} = \sum_{g=1}^m 2^{-(g-1)k} (x_{j[(g-1)k+1]} + 2^{-1} x_{j[(g-1)k+2]} + \dots + 2^{-(k-1)} x_{j[(g-1)k+k]}), \quad (17)$$

де x_{ji} – значення i -о розряду j -о операнда; n – розрядність операнда, m – кількість груп, на які розбивається операнд.

Піднесення до квадрату є основною операцією обчислення суми квадратів різниць. Для виконання такої операції використаємо вертикальний алгоритм:

$$X^2 = (0.01) \wedge x_1 + 2^{-1} (0.x_1 01) \wedge x_2 + 2^{-2} (0.x_1 x_2 01) \wedge x_3 + \dots + 2^{-(n-1)} (0.x_1 x_2 \dots x_{n-1} 01) \wedge x_n = \sum_{i=1}^n 2^{-(i-1)} R_i, \quad (18)$$

де R_i – частковий результат піднесення до квадрату, який визначається так:

$$R_i = (0.x_1 x_2 \dots x_{i-1} 01) \wedge x_i. \quad (19)$$

Розвитком розглянутого алгоритму є формування для групи із k розрядів R_{Mg} макрочасткового результату піднесення до квадрату:

$$R_{Mg} = R_{g1} + 2^{-1} R_{g2} + \dots + 2^{-(k-1)} R_{gk} = \sum_{r=1}^k 2^{-(r-1)} R_{gr}, \quad (20)$$

де R_{gr} - частковий результат піднесення до квадрату.

Алгоритм піднесення до квадрату з використанням формування макрочасткових результатів R_{Mg} записується так:

$$X^2 = \sum_{g=1}^m 2^{-(g-1)k} R_{Mg}. \quad (21)$$

Обчислення суми квадратів різниць будемо здійснювати на основі багатооперандного підходу, який полягає у одночасному опрацюванні всіх операндів і формуванні макрочасткового результату суми квадратів різниць. Обчислення суми квадратів різниць будемо здійснювати за паралельним вертикально-груповим методом, який записується так:

$$\begin{aligned}
y &= (X_1^e - X_1^b)^2 + (X_2^e - X_2^b)^2 + \dots + (X_N^e - X_N^b)^2 = \Delta X_1^2 + \Delta X_2^2 + \dots + \Delta X_N^2 = \\
&= \sum_{g=1}^m 2^{-(g-1)k} R_{1Mg} + \dots + \sum_{g=1}^m 2^{-(g-1)k} R_{NMg} = \sum_{j=1}^N \sum_{g=1}^m 2^{(g-1)k} R_{jMg} = \sum_{g=1}^m 2^{(g-1)k} \sum_{j=1}^N R_{jMg} = \sum_{g=1}^m 2^{(g-1)k} P_{Mg}
\end{aligned}
\tag{22}$$

де P_{Mg} – g -й макрочастковий результат суми квадратів різниць.

Основними етапами паралельного вертикально-групового методу обчислення суми квадратів різниць є:

- одночасне послідовно-групове надходження операндів X_i^e, X_i^b і обчислення модуля ΔX_i ;
- формування макрочасткових результатів піднесення до квадрату R_{Mg} ;
- формування макрочасткового результату суми квадратів різниць P_{Mg} підсумовуванням макрочасткових результатів піднесення до квадрату R_{Mg} ;
- отримання результату суми квадратів різниць шляхом підсумовування макрочасткових результатів обчислення P_{Mg} із зсувом вправо на k розрядів.

Структура компонента для паралельного вертикально-групового обчислення суми квадратів різниць. В залежності від способу формування та підсумовування макрочасткових результатів суми квадратів різниць P_{Mg} можливі такі варіанти реалізації компоненти обчислення суми квадратів різниць:

- з послідовним формуванням і підсумовуванням P_{Mg} ;
- з паралельним формуванням і послідовним підсумовуванням P_{Mg} ;
- з паралельним формуванням і підсумовуванням P_{Mg} .

Розроблена структура компоненти обчислення суми квадратів різниць з паралельним формуванням і послідовним підсумовуванням P_{Mg} наведена на рисунок 3.3, де Pg – регістр, $БСм$ – багатовходовий суматор, $См$ – суматор, $ПЕ$ – процесорний елемент.

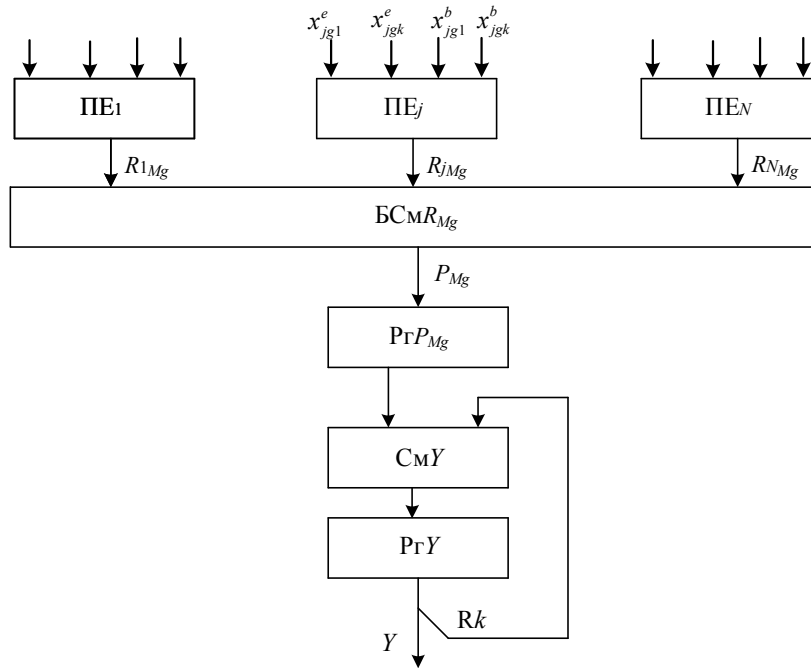


Рисунок 3.3 - Структура компоненти обчислення суми квадратів різниць з паралельним формуванням і послідовним підсумовуванням

Основними елементами даної структури є: PE_j , які призначені для формування макрочасткових результатів піднесення до квадрату R_{jMg} ; $БСМR_{jMg}$, який шляхом паралельного підсумовування R_{jMg} формує макрочастковий результату суми квадратів різниць P_{Mg} ; $Pг P_{Mg}$, $СМУ$ та $PгУ$, які забезпечують послідовне обчислення суми квадратів різниць за формулою:

$$Y_g = 2^{-k} Z_{g-1} + P_{Mg}, \quad (23)$$

де $Y_0=0$.

Побудована структура PE_j наведена на рисунок 3.4, де Від – віднімач, Тг – тригер, ПК – перетворювач паралельного коду у вертикально-груповий, $|\Delta x_j|$ - формувач модуля різниці, $R_{гг}$ – формувач часткового результату піднесення до квадрату.

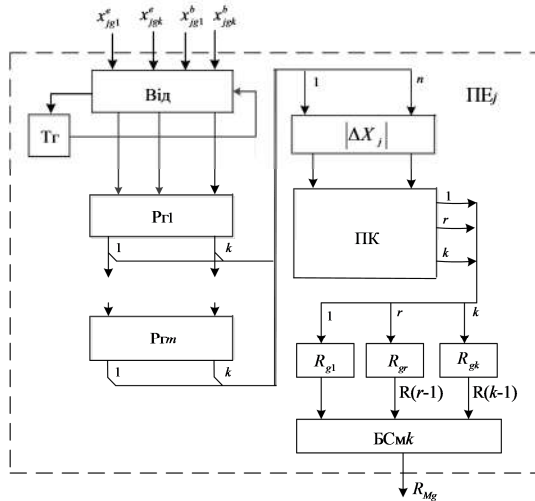


Рисунок 3.4 - Розроблена структура ПЕ

Находження операндів x_j^e і x_j^b на вхід ПЕ_j здійснюється послідовно групами з k розрядів, починаючи з молодших розрядів. В кожному ПЕ_j за допомогою віднімача Від за m тактів обчислюється різниця ΔX_j , яка записується в регістри R_{g1}, \dots, R_{gm} . Обчислена різниця ΔX_j надходить на входи формувача $|\Delta X_j|$, на виході якого отримуємо її модуль $|\Delta X_j|$. У наступних тактах роботи на виходах формувачів R_{rg} отримується часткові результати піднесення до квадрату. Формування часткових результатів піднесення до квадрату R_{rg} здійснюється починаючи з старших розрядів модуля $|\Delta X_j|$ у відповідності до формули (19). Сформовані k часткових результатів піднесення до квадрату R_{rg} надходять із зсувом вправо на $(r-1)$ розряди входи багатовходового суматора БСмк де вони додаються. Отримана на виході багатовходового суматора БСмк сума є макрочастковим результатом піднесення до квадрату R_{jMg} . Макрочасткові результати піднесення до квадрату R_{1Mg}, \dots, R_{NMg} додаються з допомогою багатовходового суматора БСм R_{Mg} . Отримана сума, яка є макрочастковим результатом суми квадратів різниць R_{Mg} , записується у регістр $R_{gR_{Mg}}$. На суматорі СМУ у кожному такті виконується додавання даних з виходу регістр $R_{gR_{Mg}}$ до раніше накопиченої суми з регістра R_{gY} зсунутої на k розрядів вправо.

Особливістю роботи даної компоненти є суміщення у часі процесів введення даних одного масиву та обчислення другого. Використовуючи таке

суміщення обчислення суми квадратів різниць у даній компоненті буде виконуватися за m тактів.

3.4 Компонента обчислення скалярного добутку.

Паралельний вертикально-груповий метод обчислення скалярного добутку ґрунтуються на елементарних арифметичних операцій, орієнтований на НВІС-реалізацію та забезпечує зменшення кількості тактів роботи, а відповідно часу обчислення. Обчислення скалярного добутку за даним методом зводиться до формування та підсумовування часткових добутків у відповідності з наступною формулою:

$$Z = \sum_{j=1}^N W_j X_j = \sum_{j=1}^N \sum_{g=1}^m 2^{-(g-1)k} (W_j X_{j[(g-1)k+1]} + 2^{-1} W_j X_{j[(g-1)k+2]} + \dots + 2^{-(k-1)} W_j X_{j[(g-1)k+k]}) = \sum_{j=1}^N \sum_{g=1}^m 2^{-(g-1)k} P_{jg}, \quad (24)$$

де P_{jg} - груповий частковий добуток,

$$P_{jg} = W_j X_{j[(g-1)k+1]} + 2^{-1} W_j X_{j[(g-1)k+2]} + \dots + 2^{-(k-1)} W_j X_{j[(g-1)k+k]}.$$

Зробивши необхідні зміни у формулі (24) обчислення скалярного добутку можна записати так:

$$Z = \sum_{g=1}^m 2^{-(g-1)k} \sum_{j=1}^N P_{jg} = \sum_{g=1}^m 2^{-(g-1)k} R_g, \quad (25)$$

де R_g - частковий результат скалярного добутку, $R_g = \sum_{j=1}^N P_{jg}$, який можна обчислювати як паралельно, так і послідовно. Послідовне обчислення результату скалярного добутку R_g виконується за формулою:

$$R_g = \sum_{j=1}^m 2^{-(g-1)k} \sum_{j=1}^N (w_{j[(g-1)k+1]} x_{j[(g-1)k+1]} + 2^{-1} w_{j[(g-1)k+2]} x_{j[(g-1)k+2]} + \dots + 2^{k-1} w_{j[(g-1)k+k]} x_{j[(g-1)k+k]}). \quad (26)$$

З формули (25) випливає, що для обчислення скалярного добутку необхідно виконати m тактів, у кожному з яких виконуються такі операції:

- формування для кожної j -ї пари операндів k часткових добутків у відповідності з формулою $P_{j[(g-1)k+r]} = W_j X_{j[(g-1)k+r]}$, де $r=1, \dots, k$;

- обчислення для j -ї пари операндів групового часткового добутку P_{jg} у відповідності з формулою $P_{jg} = \sum_{r=1}^k 2^{-(r-1)} W_j X_{j[(g-1)k+r]}$;

- обчислення часткового результату скалярного добутку R_g шляхом підсумовування групових часткових добутків P_{jg} у відповідності з формулою

$$R_g = \sum_{j=1}^N P_{jg};$$

- підсумовування часткових результатів скалярного добутку R_g у відповідності з виразом $Z_g = 2^{-k} Z_{g-1} + R_g$, де $Z_0=0$.

Структура компоненти паралельного вертикально-групового обчислення скалярного добутку. В залежності від способів надходження операндів, використання кількості шин для введення операндів, формування та підсумовування часткового результату скалярного добутку R_g , протіканням процесів надходження та обчислення скалярного добутку у часі, можливі такі варіанти реалізації компоненти обчислення скалярного добутку:

- з паралельним вертикально-груповим надходженням даних молодшими (старшими) розрядами вперед;
- з окремими або мультиплексованими шини для вхідних даних X_j та вагових коефіцієнтів W_j ;
- з паралельним або послідовним формуванням часткового результату скалярного добутку;
- з паралельним або послідовним підсумовуванням часткових результатів скалярного добутку;
- з розділенням або суміщенням процесів надходження та обчислення скалярного добутку.

Для НВІС-реалізації найефективнішою є така структура: з надходженням даних молодшими розрядами вперед; використанням для введення операндів мультиплексованих шин; паралельним формуванням часткового результату скалярного добутку R_g ; послідовним підсумовуванням часткових результатів скалярного добутку R_g ; суміщенням процесів надходження та обчислення

скалярного добутку. Така структура наведена на рисунок.4, де ФЧР – формувач часткових результатів скалярного добутку.

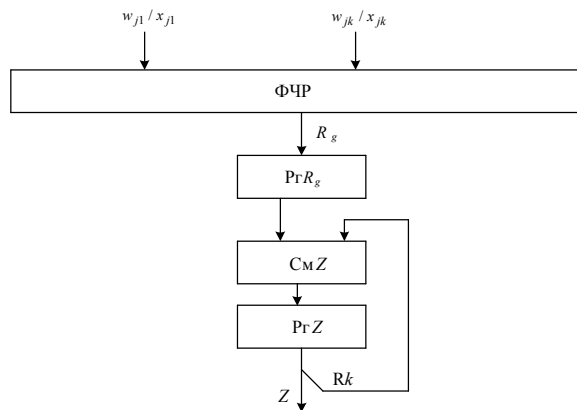


Рис.4. Структура компоненти з паралельним вертикально-груповим обчисленням скалярного добутку

Основним елементом даної структури є ФЧР, який може здійснювати як паралельне, так послідовне формування часткового результату скалярного добутку R_g . Структура ФЧР з паралельним формування часткового результату скалярного добутку R_g наведена на рис.5, де $w_{j1}/x_{j1} \dots w_{jk}/x_{jk}$ мультиплексовані однорозрядні інформаційні входи; Рг – регістр, ПК – перетворювач послідовних груп розрядів у паралельний код; kC_m , NC_m – N і k входів суматори.

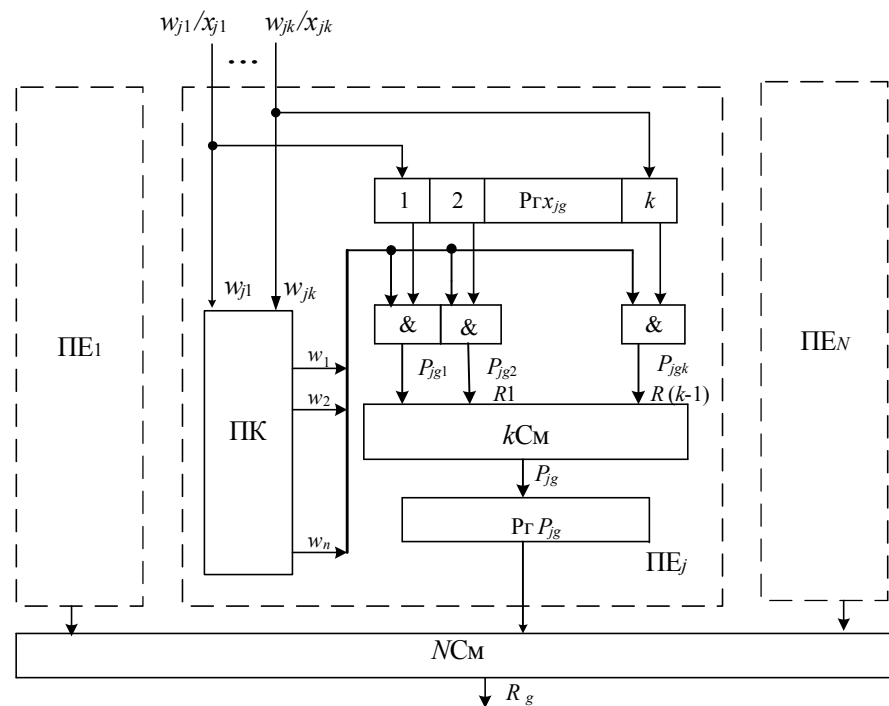


Рис.5. Структура формувача часткових результатів скалярного добутку

Паралельне вертикально-групове обчислення скалярного добутку у даній компоненті можна розбити на два етапи, кожний з яких виконується за m тактів.

На першому етапі в кожному g -у такті, починаючи з молодших розрядів, на j -й інформаційний вхід надходять k розрядів множника W_j . В ПК виконується накопичення послідових груп розрядів множеного W_j і їх перетворення в паралельний код.

На другому етапі у кожному g у такті роботи в ПЕ $_j$ для групи розрядів множника $X_{j_{g1}}X_{j_{g2}}\dots X_{j_{gk}}$ формується k часткових добутків у відповідності з формулою $P_{jgr} = W_j X_{jgr}$. Сформовані часткові добутки надходять на вхід k -входовий суматор, причому r -й ($r=1, \dots, k$) частковий добуток $W_j X_{jgr}$ зсунутий відносно $(r-1)$ -о часткового добутку $W_j X_{jg(r-1)}$ на один розряд вправо. Шляхом додавання часткових добутків на виході k -входовому суматорі отримуємо груповий частковий добуток P_{jg} , який записується у регістр PgP_{jg} . Групові часткові добутки P_{jg} з виходів ПЕ надходять на входи суматора NCM , де вони додаються. На виході суматора NCM отримуємо g -й частковий результат скалярного добутку R_g , який записується в регістр PgR_g . На суматорі CMZ здійснюється підсумовування часткових результатів скалярного добутку R_g , у відповідності з виразом $Z_g = 2^{-k} Z_{g-1} + R_g$, де $Z_0=0$.

Розроблена компонента для обчислення скалярного добутку працює за конвеєрним принципом і орієнтована на опрацювання неперервних потоків даних. Конвеєрний такт роботи такого пристрою визначається так:

$$T_k = t_{Pg} + t_{NCM},$$

де t_{Pg} – час спрацювання регістра, t_{NCM} – час додавання N чисел. Обчислення скалярного добутку здійснюється за m конвеєрних тактів.

ВИСНОВКИ

1. У роботі показано, що розробку компонентів нейромережевих технологій з вертикально-паралельною обробкою даних доцільно здійснювати на основі інтегрального підходу, який охоплює: сучасні методи та алгоритми навчання та функціонування нейромереж; сучасну елементну базу; нові, орієнтовані на НВІС-реалізацію, алгоритмічні, архітектурні та схемотехнічні рішення.

2. Запропоновано нейромережеві засоби будувати за такими принципами: змінного складу обладнання; модульності, який передбачає реалізацію компонентів у вигляді функціонально-завершених пристроїв, що мають вихід на стандартний інтерфейс; узгодженості інтенсивності надходження даних з обчислювальною здатністю апаратних засобів; конвеєризації та просторового паралелізму обробки даних; локалізації та спрощення зв'язків між елементами компонентів; спеціалізації та адаптації апаратно-програмних засобів до структури алгоритмів та інтенсивності надходження даних.

3. Показано, що основними шляхами підвищення ефективності використання обладнання компонентів штучних нейронних мереж є: вибір ефективних методів та алгоритмів реалізації компонентів штучних нейронних мереж; зменшення розрядності операційних пристроїв, ємності пам'яті, кількості і розрядності каналів передачі даних; узгодження інтенсивності надходження даних з обчислювальною здатністю компонентів штучних нейронних мереж.

4. Визначено, що узгодження інтенсивності надходження даних з обчислювальною здатністю компонентів штучних нейронних мереж реального часу здійснюється шляхом зміни тривалості конвеєрного такту, кількості і розрядності каналів надходження даних.

5. Визначено основні етапи синтезу компонентів штучних нейронних мереж: вибір та розробка методів алгоритмів; визначення основних параметрів апаратних засобів; перехід від алгоритму до узгоджено-паралельної структури.

6. Вдосконалено метод переходу від алгоритму до структури апаратних засобів систем реального часу, основними етапами якого є: декомпозиція алгоритмів розв'язання задачі; проектування комунікацій між функціональними операторами; укрупнення функціональних операторів; планування обчислень.

7. Вдосконалено вертикальний багатооперандний метод обчислення оператора суми парних добутків у реальному часі, який ґрунтується на формуванні та підсумовуванні макрочасткових добутків, кількість яких залежить від їх розрядності та алгоритму формування часткових добутків для пар операндів. При використанні даного методу час обчислення визначається розрядністю, а не кількістю чисел.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Хайкин С. Нейронные сети / Пер. с английского – М.: Вильямс, 2006. – 1104 с.
2. Рассел С., Норвиг П. Искусственный интеллект: современный подход / Пер. с английского – М.: Вильямс, 2007. – 1408 с.
3. Каллан Р. Основные концепции нейронных сетей / Пер. с английского – М.: Вильямс, 2001. – 288 с.
4. Рутковская Д., Пилиньский Л., Рутковский Л. Нейронные сети, генетические алгоритмы и нечеткие системы / Пер. с польского – М.: Горячая линия-Телеком, 2007. – 452 с.
5. Головкин В.А. Нейронные сети: обучение, организация и применение – М.: ИПРЖР, 2002. – 256 с.
6. Круглов В.В., Борисов В.В. Искусственные нейронные сети. Теория и практика – М.: Горячая Линия-Телеком, 2002 – 382 с.
7. Круглов В.В., Борисов В.В. Искусственные нейронные сети. Теория и практика. – 2-е изд., стереотип. – М.: Горячая линия-Телеком, 2002. – 382 с.
8. Галушкин А.И. Нейрокомпьютеры. Кн.3.-М; ИПРЖР,2000.-528с.
9. Осовский С. Нейронные сети для обработки информации / Пер. с польского. – М.: Финансы и статистика, 2002. – 344 с.
10. Параллельная обработка информации: в 5т./ АН УССР. Физ-мех. ин-т. – Киев: Наукова думка, 1984. – т5: Проблемно-ориентированные и специализированные средства обработки информации / А.И. Аксенов, В.В. Аристов, Е.Ю. Барзилович и др.; Под ред. Б.Н. Малиновского и Грицика В. В. – 1990. – 504с.
11. Цмоць І.Г. Інформаційні технології та спеціалізовані засоби обробки сигналів і зображень у реальному часі. Львів: УАД, 2005. – 227с.
12. Цмоць І.Г. Паралельні алгоритми та матричні НВІС-структури пристроїв множення для комп'ютерних систем реального часу. Науково-технічний журнал “Інформаційні технології і системи”. Том 7, №1, Львів 2004, С. 5-16.
13. Стрямець С.П., Цмоць І.Г. Паралельні алгоритми та НВІС-структури обчислення суми парних добутоків. Вісник НУ “Львівська політехніка”

“Комп’ютерна інженерія та інформаційні технології” № 496, Львів 2003. С.255-263.

14. Цмоць І.Г., Іванців В.Р. Паралельно-вертикальний метод і базова структура пристрою обчислення сум парних добутків.

15. Березький О.М., Цмоць І.Г. Методи, алгоритми та НВІС-структури для множення матриці на вектор у реальному часі. Вісник Хмельницького національного університету, №3, 2007/ Т1. – С.134-140/

16. Каневский Ю.С. Системные процессоры. — К: Техніка, 1991. — 173с.

17. Шалыто А.А. Методы аппаратной и программной реализации алгоритмов. – СПб.: Наука, 2000. – 780 с.

18. Савельев А.Я. Арифметические и логические основы цифровых автоматов. М.: Высш. школа, 1980. – 225 с.

19. Соловьев Г.Н. Арифметические устройства ЭВМ. М.: Энергия, 1978. – 176 с.

20. Карцев М.А. Арифметика цифровых машин - М., Наука, 1969.

21. Карцев М.А., Брик В.А. Вычислительные системы и синхронная арифметика. - М.: Радио и связь, 1981. - 360с.

22. Вышинский В.А. Об одном способе умножения чисел // Управляющие системы и машины. – 2001. - №1. –с.48-51.

23. Цмоць І.Г. Алгоритми і структури для ВІС перемножувача комплексних чисел. Вісник ДУ "Львівська політехніка" "Комп’ютерні системи проектування. Теорія і практика" № 327. Львів, 1998.- с.231-240.

24. А.с. 1705820 (СССР). Вычислительное устройство. Е.Я. Ваврук, И.Г. Цмоць. – Оpubл. в Б.И., 1992, №2.

25. Солонина А.И., Улахович Д.А., Яковлев А.А. Алгоритмы и процессоры цифровой обработки сигналов. СПб: БХВ-СПб 2001, 464с.

26. С.Кун. Матричные процессоры на СБИС:-М.:Мир,1991.-672 с.

27. Паралельная обработка информации: Т.4. Высокопроизводительные системы параллельной обработки информации /Под ред. В.В. Грицыка — Киев: Наук. думка, 1988. — 272с.

28. Цмоць І. Г. Алгоритмічні операційні пристрої для обчислення базових операцій алгоритмів швидкого перетворення Фур’є комплексної послідовності.

Збірник наукових праць Інституту проблем моделювання в енергетиці НАН України, 1999. Випуск 2, с. 159-173.

29. О. Березький, І. Цмоць. Методи та НВІС-структури для множення матриці на матрицю у реальному часі. Вісник НУ “Львівська політехніка” “Комп’ютерні системи проектування. Теорія і практика” № 591. Львів 2007. С.63-76.

30. Самофалов К.Г. и др. Прикладная теория цифровых автоматов. К: Вища шк., 1987. – 375с.

31. Грушицкий Р.И., Мурсаев А.Х., Угрюмов Е.П. Проектирование систем на микросхемах программируемой логики. – СПб.: БХВ-Петербург, 2002. – 608с.

32. А.В. Палагин, В.Н. Опанасенко. Реконфигурируемые вычислительные системы. – К.: Просвіта, 2006.- 280с.

33. Грицик В.В., Опотяк Ю.В., Цмоць І.Г. Інтегрований підхід до розробки високоефективних апаратних засобів інформаційних технологій обробки сигналів реального часу. Системні технології. Регіональний міжвузівський збірник наукових праць. - Випуск 6 (47).- Дніпропетровськ, 2006.-С.84-95.

34. Цмоць І.Г. Паралельно-потоківі системи цифрової обробки сигналів з обміном через багатопортову пам’ять // Вісник ДУ “Львівська політехніка”: "Комп’ютерні системи проектування. Теорія і практика ", №373. - Львів, 1999. - С.106-111.

35. Цмоць І.Г. Принципи побудови і базові структури паралельної пам’яті // Збірник наук. праць ІПМЕ НАН України: "Моделювання та інформаційні технології", Вип. 1. –Львів, 1999. – С.27-33.

36. Цмоць І.Г., Демида Б.А. Особливості та основні принципи побудови пам’яті програмованих процесорів цифрової обробки сигналів // Науково-технічний журнал "Технічні вісті", №1(6),2(7). - Львів, 1998 – С.34-39.

37. Цмоць І.Г., Рашкевич Ю.М., Демида Б.А., Ревич М.Р., Кашем А.М. Паралельна пам’ять систем управління та цифрової обробки і оцінка її основних характеристик // Вестник Харьковского государственного политехнического университета ”Системный анализ, управление и информационные технологии”, Вип.97. Харьков, 2000. – С.79-84.

38. Цмоць І.Г., Демида Б.А. Структури спеціалізованої паралельної пам’яті високопродуктивних процесорів управління та цифрової обробки сигналів //

Вісник ДУ “Львівська політехніка”; “Комп’ютерна інженерія та інформаційні технології”, №380. - Львів, 1999. – С.18-29.

39. Цмоць І.Г., Ваврук Є.А., Демида Б.А. Аналіз методів управління доступом до паралельної пам’яті // Вісник ДУ “Львівська політехніка”: “Комп’ютерна інженерія та інформаційні технології”, №392. - Львів, 2000. - С.27-31.

40. Цмоць І.Г., Демида Б.А. Синтез паралельної пам’яті для систем керування та цифрової обробки сигналів // Вісник ДУ “Львівська політехніка”: “Комп’ютерна інженерія та інформаційні технології”, №370. - Львів, 1999. – С.9-18.

41. Батюк А.Є., Цмоць І.Г. Методи синтезу спеціалізованих обчислювальних систем для розв’язання задач у реальному часі // Інформаційні технології і системи. Т2, №1, Львів 1999 – с.155-161.

42. Цмоць І.Г. Принципи розробки і оцінка основних характеристик високопродуктивних процесорів на надвеликих інтегральних схемах/ Вісник ДУ “Львівська політехніка”, №349, Львів, 1998 - с.5-11.