

Різницево-зрізова обробка як одна з альтернатив нейроподібної обробки даних

¹Т.Б. Мартинюк, ²Л.М. Куперштейн

¹Вінницький національний технічний університет, Україна

²Вінницький фінансово-економічний університет, Україна
kuperok@mail.ru

Анотація

У статті розглядається перспективний спосіб порогової паралельної обробки векторних даних за різницево-зрізовими зрізами, що забезпечує ефективну реалізацію нейроподібної обробки інформації у порівнянні з відомими й класичними підходами.

1. Вступ

Нейронні мережі (НМ) використовуються для розв'язання різноманітних задач у багатьох галузях людської діяльності, серед яких: економіка і бізнес, медицина, авіація, зв'язок, безпека й охоронні системи, автоматизація виробництва тощо [1 – 3].

Розв'язання більшості вищезазначених задач базується на класифікації та розпізнаванні образів, прикладом яких може бути задача розпізнавання букв, розпізнавання мови, класифікація сигналу електрокардіограми, визначення кредитоспроможності клієнта банку, медичні задачі, задача керування портфелем цінних паперів, задача визначення життєздатних і схильних до банкрутства фірм тощо [3 – 5].

Тому досить актуальним сьогодні є розробка нових та вдосконалення відомих методів нейроподібної обробки інформації, а також моделей нейромереж та формального нейрона (ФН) [3 – 6].

У класичному виконанні модель ФН являє собою послідовне виконання суматорної та активаційної функції виду [2, 3, 7]:

$$S = \sum_{i=1}^n w_i \cdot x_i, \quad (1)$$

$$Y = f(S) \quad (2)$$

де x_i – i -й вхідний сигнал; w_i – i -й ваговий коефіцієнт; S – суматорна функція; $f(\bullet)$ – активаційна функція, Y – вихідний сигнал нейрона.

Якщо для програмної реалізації така модель є єдиною можливою, то при апаратній реалізації вона вносить значні часові затримки. При всій різноманітності наявних функцій активації [2, 4, 5] для прискорення реалізації функції (1) єдиною використовується багатовхідний суматор у вигляді деревоподібної (пірамідальної) схеми суматорів. Це дозволяє зменшити часові затрати при підсумовуванні n операндів у $n/\log_2 n$ раз [6]. При цьому питання про суміщення виконання обох функцій (1) і (2) залишається відкритим. Не в останню чергу це пов'язане із класичним методом підсумовування чисел. Отже, для підвищення швидкості та суміщення обчислення функцій (1) та (2) необхідно переглянути метод обчислення функції (1).

2. Постановка задачі дослідження

У роботі [7] розглянуті всі можливі способи обчислення оператора групового підсумовування (ГП) виду

$$X_n = \sum_{i=1}^n a_i, \quad (3)$$

який використовується для реалізації функції (1) за умови, що $a_i = w_i \cdot x_i$. Аналіз ефективності відомих способів обчислення цього оператора [8] показав, що оптимальним є спосіб відомий, як алгоритм рекурсивного подвоєння [9], що і реалізується у вигляді згаданої деревоподібної структури. У той же час відомо метод паралельного підсумовування масиву чисел за принципом різницево-зрізових (ПРЗ) [7, 10]. Уперше цей підхід був описаний в [11, 12] як паралельний спосіб додавання групи тривалостей часових інтервалів. Подальші дослідження показали універсальність такого підходу, у результаті чого він був використаний для паралельної обробки векторних масивів числових даних, якими є, по суті, зважені входи нейронів [13].

ПРЗ дає можливість створити модель формального нейрона, у якій реалізується розпаралелення процесу підсумовування, а головне, підсумовування суміщене з обчисленням порогової функції активації нейрона. Це набагато прискорює процес обробки, оскільки момент перевищення суми над порогом можна зафіксувати, не формуючи кінцеву зважену суму, на противагу класичній моделі [14]. Вихід Y такого нейрона, будучи знаковою функцією виду:

$$Y = f(S) = \text{sign}(\theta - S), \quad (4)$$

де θ – зовнішній поріг обробки, приймає наступні значення:

$$Y = \begin{cases} 1, & \text{якщо } S = \sum_{i=1}^n a_i \\ 0, & \text{в протилежному випадку} \end{cases}, \quad (5)$$

тобто реалізує найпростішу порогову функцію – функцію одиничного стрибка [3, 4]. Це, у свою чергу, не зменшує практичної цінності моделі, оскільки з використанням таких ФН ефективно вирішуються задачі ідентифікації динамічних систем і адаптивного керування ними, класифікація образів, розпізнавання зображень, обробка двійкових сигналів [3-5, 15, 16].

Розглянемо детальніше спосіб порогової паралельної обробки за різницево-зрізовими (РЗ), які реалізують модель нейрона.

3. Спосіб порогової паралельної обробки векторних даних за РЗ

Базовими для обробки даних за РЗ є поняття різницевого зрізу A_j , поточної часткової суми S_j елементів векторного масиву та поточного внутрішнього порога обробки q_j , які формуються в j -му циклі обробки [7, 14].

У випадку, якщо як внутрішній поріг q_j вибирається мінімальним елемент поточного масиву даних A_{j-1} , тобто

$$q_j = \min A_{j-1} = \min \{a_{i,j-1}\}_1^n \quad (6)$$

то РЗ A_j і поточна часткова сума S_j наступного j -го циклу обробки формуються в такий спосіб [9, 16]:

$$A_j = \{a_{i,j}\} = \{a_{i,j-1} - q_j\}_1^n, \quad (7)$$

$$S_j = q_j \cdot d_j, \quad (8)$$

де d_j – кількість додатних елементів $a_{i,j}$ у РЗ A_j .

А варіантність порогової паралельної обробки за РЗ залежить від способу формування поточного зовнішнього порога обробки, тобто від способу порівняння або накопиченої суми S , або поточної часткової суми S_j , або поточного внутрішнього порога q_j із зовнішнім порогом θ обробки з метою визначення знакової функції (4).

В основі запропонованого варіанта порогового паралельного підсумовування елементів векторного масиву даних (цілі невід'ємні числа) лежить спосіб, в якому в кожному циклі обробки виконується формування поточних часткових різниць елементів векторного масиву, а також формування поточного порога для наступного циклу обробки шляхом послідовного віднімання від поточного порога поточного мінімального елемента кількістю, що дорівнює кількості додатних ненульових елементів у поточному векторному масиві, і формування результуючого сигналу, що приводить до прискорення способу за рахунок скорочення часу визначення від'ємної поточної часткової різниці [15].

Математичну модель запропонованого способу порогового паралельного підсумовування можна представити в такий спосіб: якщо суму S елементів первісного масиву можна представити у вигляді

$$\sum_{i=1}^n a_i = S = S_1 + S_2 + \dots + S_N = \sum_{j=1}^N S_j, \quad (9)$$

те величину $\Delta_0 = \theta - S$ можна визначити в такий спосіб:

$$\Delta = \theta - S = \theta - (S_1 + S_2 + \dots + S_N) = \theta - S_1 - S_2 - \dots - S_N =$$

$$= (\dots ((\theta - S_1) - S_2) - \dots - S_N) \leq 0. \quad (10)$$

Уведемо позначення для поточного порога

$$\Delta_j = \Delta_{j-1} - S_j \leq 0, j = \overline{1, N} \quad (11)$$

де $\Delta_0 = \theta$.

Тоді вираз (10) можна записати так:

$$\Delta = [\dots [[\theta - (q_1 + q_1 + \dots + q_1)] - (q_2 + q_2 + \dots + q_2)] -$$

$$- \dots - (q_N + q_N + \dots + q_N)] = \quad (12)$$

або

$$\Delta = [\dots [[\dots ((\theta - q_1^{(1)}) - q_1^{(2)}) - \dots - q_1^{(d_1)}] - q_2^{(1)} -$$

$$- q_2^{(2)} - \dots - q_2^{(d_2)}] - \dots - q_N^{(1)} - q_N^{(2)} - \dots - q_N^{(d_N)}]. \quad (13)$$

Якщо вираз в круглих дужках позначити як

$$\delta_j^{(k)} = \delta_j^{(k-1)} - q_j^{(k)}, k = \overline{1, d_j}, \delta_j^0 = p, \quad (14)$$

то поточний поріг (22) буде мати такий вид

$$\Delta_j = \delta_{j-1}^{(d_j)}, \delta_j^0 = \Delta_{j-1}. \quad (15)$$

Отже, у кожному циклі обробки можна перевіряти виконання умови $\delta_j^{(k)} \leq 0$, а не тільки $\Delta_j \leq 0$, що дозволяє прискорити процес порогової обробки.

Таким чином, на кожному кроці обробки, крім першого, виконується порівняння із заданим порогом, що задається на початку процесу підсумовування. При цьому, якщо значення поточної часткової різниці між поточним порогом і поточним мінімальним елементом масиву від'ємне або дорівнює нулю, то процес обробки припиняється. Починаючи із

другого кроку поточне значення порога, буде приймати значення останньої поточної різниці між поточним порогом і поточним мінімальним елементом, отриманими на попередньому кроці.

4. Аналіз порогової обробки векторних даних

Аналізуючи вищенаведений спосіб, можна вказати, що даний варіант порогового паралельного підсумовування є найбільш ефективним і найбільш швидкодіючим у порівнянні з відомими [18, 19]. Це можливо за рахунок відмови від накопичення поточних часткових сум елементів векторного масиву [18] і обчислення поточних часткових величин на кожному кроці обробки [19].

Підхід, що запропонований базується на ПРЗ і забезпечує максимальний паралелізм обробки масиву чисел, оскільки в процесі задіяні одночасно всі операнди, а це, у свою чергу, приводить до прискорення процесу обробки, оскільки у випадку перевищення суми над порогом цей факт можна зафіксувати без формування остаточного значення суми всіх чисел масиву.

Розглянемо залежності тривалості порогової паралельної обробки від розмірності n векторного масиву. Графіки залежностей тривалості порогової паралельної обробки за РЗ від розмірності n масиву зображені на рис. 1. Аналіз цих залежностей доводить теоретичне твердження про те, що на тривалість порогової обробки в загальному впливають розмірність вхідного векторного масиву, характер розподілу елементів масиву, величина порогу та принцип формування його поточного значення. А це, у свою чергу, свідчить про несталий час порогової обробки за РЗ на відміну від класичної моделі, оскільки процес обробки може закінчитись у будь-якому циклі, починаючи з другого, у разі виконання умов (10), (11). На рис. 1: T^{\max} – максимальна тривалість порогової обробки при сталій тривалості циклу обробки; $T^{*\max}$ – максимальна тривалість порогової обробки при змінній тривалості циклу обробки, яка пояснюється поступовим зменшенням у кожному наступному циклі кількості значущих елементів у поточному РЗ, T^{\min} – мінімальна тривалість порогової обробки.

Аналізуючи залежності тривалості порогової паралельної обробки за РЗ видно, що максимальне значення часу обробки на один такт менше у всіх варіантів обробки, ніж у першого. Таким чином, маємо підвищення швидкодії як мінімум на 12% (в 1,14 раза) за умови виконання всіх циклів обробки до отримання нульового залишку. Крім того, мінімальний час обробки складає для аналога та запропонованого способу відповідно $(2n-t-t)$ та $n-t$, де t – час виконання одного такту. Отже, маємо підвищення швидкодії максимум у $(2n-t-t)/n-t = 1,9$ раза (на 47,4 %).

Таким чином, при реалізації нейронних мереж на базі нейронів із запропонованим варіантом обробки даних підвищується швидкість обчислень і розширюється клас розв'язуваних задач. Крім того, особливості методу різницьових зрізів і моделей, побудованих на його основі, дозволяють одержати відсортований вхідний векторний масив даних, а також реалізувати його відновлення [7, 20], що значно розширює функціональні можливості моделей.

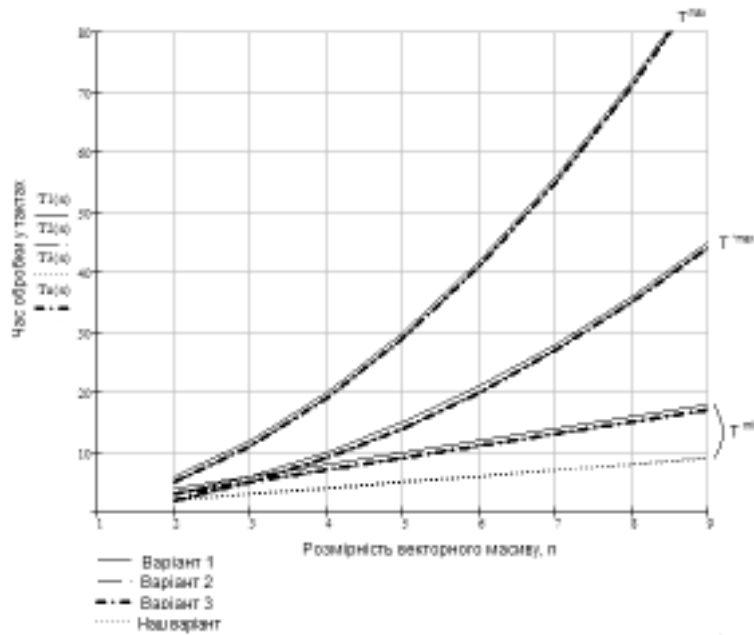


Рис. 1

Ще один з варіантів граничної обробки зображений на рис. 2, що не базується на принципах обробки за різницевиими зрізами та має послідовний характер обробки. У цьому випадку із заданого порога віднімається перший елемент векторного масиву й перевіряється умова невід'ємності залишку. У випадку її виконання процес обробки триває з формуванням нульового сигналу, і сформований залишок є порогом на наступному такті обробки, а в іншому випадку процес обробки припиняється з формуванням одиничного сигналу, тобто

$$\Delta_j = \Delta_{j-1} - a_j \leq 0, j = \overline{1, N}, \quad (16)$$

де $\Delta_0 = \theta$.

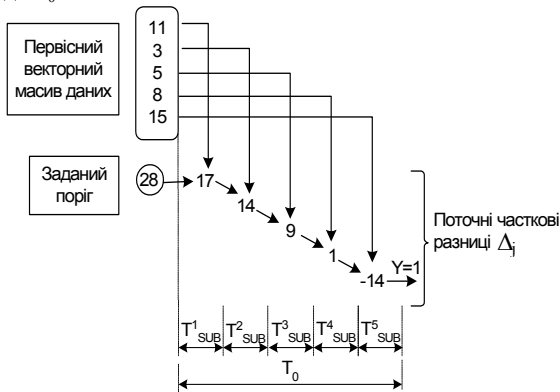


Рис. 2.

Загальний, максимальний і мінімальний час порогової обробки таким методом становить:

$$T = \sum_{j=1}^N T_{SUB}^j = 5 \cdot T_{SUB}, \quad (17)$$

$$T^{\max} = n \cdot T_{SUB}, \quad (18)$$

$$T^{\min} = T_{SUB}, \quad (19)$$

де $T_{SUB} = t_{sub} = t$. Тобто, мінімальний час обробки дорівнює часу одного такту, коли перший елемент перевищує значення порога, а максимальний час пропорційний кількості елементів у масиві. У табл. 1 наведено порівняльний аналіз алгоритмів граничної обробки елементів векторного масиву. Як видно, час

обробки з використанням підходу послідовної порогової обробки значно менше за способи на базі РЗ. Проте, він неадекватно відтворює основні принципи нейронної обробки, у якій невід'ємною частиною є формування суми або часткової суми зважених вхідних сигналів [4, 7]. Крім того, цей підхід, як і підхід обробки на базі рекурсивного подвоєння, обмежений своїми функціональними можливостями, оскільки у нього відсутнє суміщення операції підсумовування й порівняння із заданим порогом на протипагу запропонованому варіанту. Саме таке суміщення властиве нейробіологічним структурам, а відсутність його зменшує час нейрообробки.

5. Висновки

1. Запропонований спiсiб порогового паралельного підсумовування елементів векторного масиву даних за різницевиими зрізами відрізняється формуванням поточного порога для наступного кроку обробки шляхом послідовного віднімання від поточного порога поточного мінімального елемента кількiстю, що дорiвнює кiлькiстю додатних ненульових елементiв у поточному векторному масивi. Це дозволяє завершити процес обробки при наявності одиничного підсумкового сигналу без формування поточного порога на кожному кроці обробки. Така обробка прискорює порогове паралельне підсумовування за рахунок скорочення часу визначення від'ємної поточної часткової різниці. Одержжане прискорення обробки даним способом становить від 12% до 47%.

2. Проведений аналіз способів порогової обробки векторних даних із застосуванням залежностей часу порогової паралельної обробки від кількості елементів масиву показав швидкодiю запропонованого в порiвнянні з вiдомими. Такий пiдхiд дозволяє створити математичну модель формального нейрона на базi РЗ. Модель відрізняється від вiдомої реалiзацiєю розпаралелення операцiї підсумовування i сумiщенням її iз пороговою обробкою. Такий пiдхiд дозволяє значно пiдвищити швидкiсть обчислень у нейромережах, побудованих на основi такої моделi, наприклад, багат шарових перцептронiх.

Таблиця 1 Порівняльний аналіз алгоритмів порогової обробки елементів векторного масиву

Характеристики алгоритмів	Алгоритм порогового підсумовування на базі різницевого зрізів	Алгоритм послідовної порогової обробки	Алгоритм порогового підсумовування на базі рекурсивного подвоєння
Часові залежності: а) якісна оцінка	Нефіксована кількість циклів обробки, яка залежить від розмірності n масиву, характеру розподілу елементів в масиві та величини порогу	Нефіксована кількість циклів обробки, яка залежить від розмірності n масиву, характеру розподілу елементів в масиві та величини порогу	Строго фіксована кількість циклів обробки, яка залежить тільки від розмірності n масиву
б) кількісна оцінка	$T^{\min} = T_{\min} + t_{sm} = t_{com}(n-1) + t_{sub}$ $T^{\max} = t_{com} \sum_{j=1}^N (n-j) + N \cdot t_{sub}$	$T^{\min} = t_{sub}$ $T^{\max} = n \cdot t_{sub}$	$T = \log_2 n \cdot t_{sm} + t_{sub}$
Особливості обробки: а) можливість суміщення операції підсумовування з іншими операціями	Можливість використання сформованих на кожному кроці часткових сум (різниць) як загальної частини всіх елементів поточного різницевого зрізу для порівняння із зовнішнім порогом обробки	Відсутня можливість суміщення операцій підсумовування та порівняння з порогом	Використання попарних часткових сум, які формуються в кожному циклі, тільки для підсумовування
б) рекурсивний характер процесу обробки	Наявність рекурсії в рівняннях для основних величин: $S_{i,j} = S_{i-1,j} + q_j \cdot f_{i,j}, \quad a_{i,j} = a_{i,j-1} - q_j,$ $q_j = \min_{i=1}^n \{a_{i,j}, a_{i+1,j}\}$ $\Delta_j = \Delta_{j-1} - S_j \leq 0, \quad j = \overline{1, N}$	Наявність рекурсії в рівнянні $\Delta_j = \Delta_{j-1} - a_j \leq 0, \quad j = \overline{1, N}$	Відсутність рекурсії в основному рівнянні виду: $S_{i,j} = a_{i,j} + S_{i+2, \log_2^{n-i} j}$

6. Література

- Галушкин А.И., Иванов В.В. Некоторые концептуальные вопросы развития нейрокомпьютеров// Зарубежная радиоэлектроника.–1997, №2, С.3-9.
- Комашинский В.И., Смирнов Д.А. Нейронные сети в системах управления и связи. – М.: Горячая линия – Телеком, 2003. – 94 с.
- Галушкин А.И. Нейрокомпьютеры. Кн. 3: Учеб. пособие для вузов / Общ. ред. А.И. Галушкина. – М.: ИПРЖР, 2000. – 528 с.
- Осовский С. Нейронные сети для обработки информации. – М.: Финансы и статистика. – 2002. – 289 с.
- Уоссермен Ф. Нейрокомпьютерная техника: Теория и практика. Пер. с англ.– М.: Мир, 1992. – 240 с.
- Хохлюк В.И. Параллельные алгоритмы целочисленной оптимизации. – М.: Радио и связь, 1987. – 224 с.
- Мартинюк Т.Б. Рекурсивні алгоритми багатооперандної обробки інформації. Монографія. – Вінниця: “Універсум – Вінниця”, 2000. – 216 с.
- Справочник по цифровой вычислительной технике/ Под ред. В.Н. Малиновского. – К.: Техника, 1980. – 320 с.
- Системы параллельной обработки: Пер с англ./ Под ред Д. Ивенса. – М.: Мир, 1985. – 416 с.
- Мартинюк Т.Б., Кожем’яко А.В. Реалізація концепції різницевого зрізу при обробленні зображень та розпізнаванні образів // Оптико-електронні інформаційно-енергетичні технології. – 2001. - №1. – С. 79-85.
- Свечников С.В., Кожем’яко В.П., Тимченко Л.И. Квазимульсно-потенциальные оптоэлектронные элементы и устройства логико-временного типа. – К.: Наукова думка, 1987.- 256 с.
- А.с. 1119035 СССР, МКИ G 06G 7/14 Способ параллельного сложения длительностей группы временных интервалов / В.П. Кожем’яко, Л.И. Тимченко, Т.В. Головань, Н.Е. Фурдияк, Т.Б. Мартинюк (СССР). - №3528309/18; Заявлено 24.12.82; Опубл. 15.10.84, Бюл. №38 –5 с.
- Мартинюк Т.Б., Хом’юк В.В. Методи та засоби паралельних перетворень векторних масивів даних. Монографія. – Вінниця: УНІВЕРСУМ – Вінниця, 2005. – 203 с.
- Паралельно-ієрархічне перетворення як системна модель оптико-електронних засобів штучного інтелекту. Монографія / Кол. авторів під заг. ред. В.П. Кожем’яко. – Вінниця: УНІВЕРСУМ-Вінниця, 2003. – 324 с.
- Мартинюк Т.Б. Модель порогового нейрона на основе параллельной обработки по разностным срезам // Кибернетика и системный анализ. – 2005. - №4. – С. 78-89.
- Архангельский В.И., Богаенко И.Н., Грабовский Г.Г., Рюмшин Н.А. Нейронные сети в системах автоматизации. - Киев: Техника, 1999. - 364 с.
- Патент №70886А, МПК 7 G06G7/14. Спосіб порогового паралельного додавання тривалостей групи часових інтервалів / Т.Б.Мартинюк, Л.М. Куперштейн. - №2003121328; Заявлено 31.12.2003; Опубл. 15.10.2004. - Бюл. №10. - 10с.
- Патент №70884А, МПК 7 G06G7/14. Спосіб порогового паралельного додавання тривалостей групи часових інтервалів / Т.Б.Мартинюк, А.С. Васюра, Л.М. Куперштейн, А.В. Кожем’яко. - №20031213278; Заявлено 31.12.2003; Опубл. 15.10.2004.- Бюл. №10.- 10 с.
- Патент №40228, 7G06G7/14. Спосіб порогового паралельного додавання тривалостей групи часових інтервалів/ В.П. Кожем’яко, Т.Б.Мартинюк, І.А. Вербицький, О.Б.Мартинюк. - №2000105973; Заявлено 23.07.2000; Опубл. 16.07.2001.- Бюл. №6.- 4с.
- Мартинюк Т.Б., Буда А.Г., Куперштейн Л.М., Балашов К.А. Особливості багатооперандної обробки векторного масиву чисел за методом різницевого зрізу// Матеріали ІІІ МНПК “Динаміка наукових досліджень (2004)”. Сучасні інформаційні технології. – Дніпропетровськ: Наука і освіта.–2004. – С. 8-12.