

<https://doi.org/10.15407/fmmit2026.42.058>

## Розвиток паралельних методів та алгоритмів розв'язання задач цифрової фільтрації

Михайло Яджак<sup>1</sup>, Марія Тютюнник<sup>2</sup>, Богдан Бекас<sup>3</sup>

<sup>1</sup> Доктор фізико-математичних наук, старший науковий співробітник, Інститут прикладних проблем механіки і математики ім. Я. С. Підстригача НАН України, вул. Наукова, 3-Б, м. Львів, 79060, Україна, Львівський національний університет імені Івана Франка, вул. Університетська, 1, м. Львів, 79000, Україна, e-mail: yadzhak\_ms@ukr.net

<sup>2</sup> Кандидат технічних наук, Інститут прикладних проблем механіки і математики ім. Я. С. Підстригача НАН України, вул. Наукова, 3-Б, м. Львів, 79060, Україна, e-mail: tyutmi@ukr.net

<sup>3</sup> Старший викладач, Національний лісотехнічний університет України, вул. Генерала Чупринки, 103, м. Львів, 79057, Україна, e-mail: bogdan.bekas@gmail.com

*Квазісистоличний метод організації обчислень розвинуто стосовно розв'язання задач фільтрації з використанням адаптивного згладжування та багаторазової каскадної цифрової фільтрації. Цей метод відрізняється від чисто систоличного тим, що дозволяє передавання даних від однієї «інстанції» одразу в декілька точок прийому. На основі згаданого методу побудовано оптимальні за швидкістю паралельно-конверснї алгоритми цифрової фільтрації спотворених даних. Оптимальність доводиться у вказаних класах алгоритмів, які є еквівалентними за інформаційним графом. Ці алгоритми орієнтовані на реалізацію на квазісистоличних обчислювальних структурах та комп'ютерах зі структурно-процедурною організацією обчислень. Схожість структури обчислень згідно з цими алгоритмами фільтрації та обчислень, виконуваних у згорткових нейронних мережах, дає підстави для використання останніх під час розробки більш універсального фільтра для попереднього опрацювання даних різних типів. З цією метою запропоновано гібридну архітектуру нейронної мережі – згортковий автокодувальник. Числові експерименти підтвердили високу ефективність роботи такого фільтра. Одержані наукові результати можуть бути використані під час попереднього опрацювання великих масивів даних у багатьох предметних галузях із застосуванням сучасних обчислювальних засобів.*

**Ключові слова:** цифрова фільтрація, квазісистоличний метод обчислень, паралельно-конверснї алгоритм, оптимальний за швидкістю алгоритм, адаптивне згладжування, багаторазова каскадна фільтрація, згорткова нейронна мережа.

**Вступ.** У монографії [1] зазначено, що дослідження стану та процесу функціонування багатьох реальних складних ієрархічно-мережевих систем потребує використання значних обсягів вхідних даних різних типів про їх об'єкти. Переважно ці дані надходять неперервно та є спотвореними, тому перед подальшим використанням їх потрібно попередньо опрацювати (відфільтрувати) в режимі реального часу з метою усунення завад. Для цього запропоновано квазісистоличний метод (КСМ) [2] організації обчислень під час розв'язання задач цифрової фільтрації (ЗЦФ) різної розмірності. На підставі КСМ побудовано оптимальні за швидкістю та використанням пам'яті паралельно-конверснї алгоритми (ПКА). Оп-

тимальність доводилась у вказаних класах алгоритмів, еквівалентних за інформаційним графом. У цих ПКА згладжування масиву спотворених даних здійснюється на основі зваженого сумування в межах рухомого вікна заданого розміру. Запропонований КСМ можна розвинути для виконання фільтрації на підставі адаптивного згладжування [3] та багаторазової каскадної цифрової фільтрації [4]. Основною операцією в цих задачах фільтрації є згортка, тому важливою виявляється проблема створення більш універсального фільтра для опрацювання спотворених даних з використанням згорткових нейронних мереж (НМ) [5]. Далі, у цій роботі, детально розглянемо згадані напрямки розвитку КСМ обчислень, оцінимо складність відповідних ПКА та можливості застосування НМ для ефективного виконання цифрової фільтрації.

### 1. Цифрова фільтрація з використанням адаптивного згладжування

Розглядається задача фільтрації, згідно з якою переобчислення елемента масиву даних здійснюється через вікно, розмір якого визначений лише для цього елемента. Тобто переобчислення згладжування виконуємо за формулою [3]:

$$x_i = \sum_{s=-m_i}^{m_i} x_{i+s} f_s^i; \quad i = \overline{1, n}. \quad (1)$$

В (1)  $M_i = 2m_i + 1$  – розмір рухомого вікна для  $x_i$ , а вагові коефіцієнти  $f_s^i$  ( $s = \overline{-m_i, m_i}; i = \overline{1, n}$ ) і значення  $x_{m_0}, x_{m_0+1}, \dots, x_0; x_{n+1}, x_{n+2}, \dots, x_{m^0}$  – задані константи, до того ж  $m_0 = \min_{1 \leq i \leq n} \{i - m_i\}$ ,  $m^0 = \max_{1 \leq i \leq n} \{i + m_i\}$ .

Для розв'язання такої ЗЦФ на підставі КСМ організації обчислень побудовано ПКА:

$$\begin{aligned} & \text{FOR } i = 1, n \text{ DO SYNCH} \\ & \text{DELAY } (2(m_{i^0} - m_i + i - i^0)) \\ & \text{FOR } j = 1, C \text{ DO} \\ & \{ \text{FOR } s = 1, M_i \text{ DO} \\ & \{ x_i = \text{IF } (s = 1) \text{ THEN } f_0^i * x_i \text{ ELSE } x_i + z_i, \\ & z_i = f_{h(s, m_i)}^i * x_{i+h(s, m_i)} \} \\ & \text{DELAY } (2(m_{\max} - m_i + 1)) \}. \end{aligned} \quad (2)$$

Тут *SYNCH* – це тип паралельності [6], що означає синхронізацію зсунутих між собою  $n$  пар паралельних гілок алгоритму після виконання в них кожної операції; *DELAY* ( $p$ ) – оператор затримки на  $p$  тактів;  $i^0$  – номер змінної, для якої справджується рівність  $2m_{i^0} + 1 + 2n - 2i^0 = \max_{1 \leq i \leq n} \{2m_i + 1 + 2n - 2i\}$ ;  $C$  – кількість виконуваних переобчислень згладжування елементів масиву даних, а

$m_{\max} = \max_{1 \leq i \leq n} \{m_i\}$ . Передбачається, що оператори  $x_i = \dots$  і  $z_i = \dots$ , розділені ко-  
мою, виконуються синхронно. Цілочисельна функція  $h(s, m_i)$  – це такий умов-  
ний оператор:

$$IF ((s/2) = [s/2]) THEN s/2 - m_i - 1 ELSE (s+1)/2,$$

а  $[ \cdot ]$  тут і далі означає взяття цілої частини.

Якщо вважати, що час виконання операцій додавання та множення є одна-  
ковим і дорівнює одному такту, то отримуємо, що ПКА (2) потребує

$$2(n + (1 + m_{\max})(C - 1)) + 2m_{i_0} + C - 2i^0$$

тактів. Доведено [3], що цей алгоритм є оптимальним за швидкодією у вказаному  
класі алгоритмів, які є еквівалентними за інформаційним графом.

Запропонований в [2] КСМ організації обчислень можна розвинути для по-  
будови оптимальних за швидкодією ПКА розв'язання задач фільтрації більшої  
розмірності з використанням адаптивного згладжування.

## 2. Багаторазова каскадна цифрова фільтрація

Одно- або багаторазова каскадна цифрова фільтрація виконується послідо-  
вно набором простих фільтрів, що діють один за одним. Для будь-якого  $v$ -го разу  
фільтрації обчислення у довільному  $w$ -му каскаді виконуємо за формулою:

$$x_i = \sum_{s=-m(v,w)}^{m(v,w)} x_{i+s} f_s(v, w); i = \overline{1, n}, \quad (3)$$

тобто переобчислюємо значення масиву  $x_i (i = \overline{1, n})$  через рухоме вікно розміром  
 $2m(v, w) + 1$ . У наведеній формулі (3) значення  $x_{1-m(v,w)}, x_{2-m(v,w)}, \dots, x_0$ ;  
 $x_{n+1}, x_{n+2}, \dots, x_{n+m(v,w)}$  та вагові коефіцієнти  $f_{-m(v,w)}, \dots, f_0, \dots, f_{m(v,w)}$  – за-  
дані константи.

Послідовний алгоритм реалізації  $r$ -разової каскадної фільтрації в цьому ви-  
падку має вигляд:

$$\begin{aligned} &FOR u = 1, r DO \\ &FOR p = 1, c(u) DO \\ &\{ FOR i = 1, n DO \\ &\{ t = 0 \\ &FOR s = -m(u, p), m(u, p) DO \\ &t = t + f_s(u, p) * x_{i+s} \} \\ &x_i = t \}. \end{aligned} \quad (4)$$

Тут  $c(u)$  – це кількість каскадів під час  $u$ -го разу фільтрації. Наведений алгоритм міс-  
тить рекурсію: для переобчислення значень  $x_i (i = \overline{1, n})$  у  $p$ -му каскаді використовується  
частина вже переобчислених у цьому ж каскаді значень.

Використовуючи КСМ обчислень, одержуємо оптимальний за швидкодією ПКА виконання  $r$ -разової каскадної цифрової фільтрації [7]:

```

FOR i = 1, n DO SYNCH
  DELAY(2i - 2)
  FOR u = 1, r DO
    FOR p = 1, c(u) DO
      { FOR s = 1, 2m(u, p) + 1 DO
        {  $x_i = IF(s = 1) THEN f_0(u, p) * x_i ELSE x_i + z_i,$ 
           $z_i = f_{g(s, m(u, p))}(u, p) * x_{i+g(s, m(u, p))}$  }
        DELAY(2) }
    }
  }

```

(5)

У наведеному ПКА фільтрації оператори  $x_i = \dots$  та  $z_i = \dots$ , розділені комою, виконуються у синхронному режимі. Функція  $g(s, m(u, p))$  – це умовний оператор:

$IF(s/2 = [s/2]) THEN s/2 - m(u, p) - 1 ELSE (s + 1)/2.$

Для свого виконання ПКА (5) потребує

$$2 \left( \sum_{u=1}^r \sum_{p=1}^{c(u)} m(u, p) + \sum_{u=1}^r c(u) + n - 2 \right)$$

такти. Цей алгоритм передбачає зміну кількості фільтрів та їхніх характеристик у кожному разі реалізації каскадної фільтрації, що з практичного погляду є доволі важливим. Одержаний результат узагальнюється стосовно розв'язання такого типу задач фільтрації більшої розмірності [4].

Зазначимо, що побудовані ПКА (2) та (5) орієнтовані на реалізацію на відповідних квазісistolічних обчислювальних структурах [6, 8], які можна розглядати як високопродуктивні блоки кластерних систем [9, 10] для виконання спеціалізованих обчислень, або комп'ютерах зі структурно-процедурною організацією обчислень [11] з урахуванням концепції обмеженого паралелізму [2].

### 3. Цифрова фільтрація за допомогою НМ

Описані вище підходи до цифрової фільтрації масивів даних потребують формування наборів оптимальних значень необхідних параметрів. Використання НМ [12, 13] дозволяє цей процес автоматизувати і створити більш універсальний фільтр для опрацювання вхідних даних з різними типами завад. З цією метою запропоновано гібридну архітектуру мережі – згортковий автокодувальник [5]. Згорткові НМ добре зарекомендували себе під час опрацювання даних завдяки здатності виявляти в них локальні закономірності. Запропонована архітектура мережі складається з кодувальника, який стискає вхідний сигнал, та декодувальника, який відновлює його з отриманого подання. Кожен прихований згортковий шар мережі використовує функцію активації ReLU, а вихідний шар – лінійну функцію активації. Важливою особливістю цієї архітектури є відсутність повно-

зв'язних шарів. Обчислювальну модель мережі реалізовано на мові Python з використанням бібліотек TensorFlow і Keras.

Для навчання та тестування НМ було сформовано синтетичний набір даних, який складався з 30000 чистих та відповідних їм спотворених сигналів. Спершу були згенеровані сигнали різної форми з випадковими значеннями параметрів у заданих межах. До таких сигналів належать синусоїдальні, прямокутні, полігармонійні з явно заданими складовими, аперіодичні (у вигляді затухаючих гармонійних коливань), а також пило- та трикутноподібні сигнали [14]. Далі до кожного сигналу було додано адитивні завади різних типів: білий та рожевий шум, імпульсні та гармонійні завади, а також їх поєднання. Мережа навчалась за допомогою алгоритму зворотного поширення похибки [15] з використанням міні-пакетного градієнтного спуску. Як функцію втрат, яка визначає різницю між відновленим та еталонним (чистим) сигналом, було обрано середньоквадратичну похибку (Mean Squared Error (MSE)), яка обчислюється за формулою:

$$MSE = \left( \sum_{i=1}^n (d_i - y_i)^2 \right) / n,$$

де  $d_i$ ,  $y_i$  ( $i = \overline{1, n}$ ) – відповідно значення еталонного та відновленого сигналу.

Навчання тривало протягом 30 епох. Чисельні експерименти показали, що похибка обчислень спочатку стабільно зменшувалась, а потім вийшла на плато (plateau), тобто згаданій кількості епох виявилось цілком достатньо для досягнення збіжності обчислювального процесу.

Результати тестування [16] навченої НМ підтвердили високу ефективність її роботи стосовно фільтрації спотворених різними завадами масивів даних.

**Висновки.** Розвинуто КСМ організації обчислень стосовно виконання цифрової фільтрації з використанням адаптивного згладжування та багаторазової каскадної фільтрації. Унаслідок цього одержано оптимальні за швидкодією ПКА. Запропоновано гібридну архітектуру згорткової НМ для побудови більш універсального фільтра з метою опрацювання спотворених даних. Ці наукові результати можна використати для опрацювання в режимі реального часу великих масивів даних у різних галузях [17–20] під час дослідження об'єктів складних систем.

#### Література

1. Поліщук О. Д., Яджак М. С. Моделі та методи комплексного дослідження складних мережевих систем та міжсистемних взаємодій. Львів: Інститут прикладних проблем механіки і математики ім. Я. С. Підстригача НАН України, 2023. 385 с.
2. Anisimov A. V., Yadzhak M. S. Construction of optimal algorithms for mass computations in digital filtering problems. *Cybernetics and Systems Analysis*. 2008. Vol. 44, N 4. P. 465–476.
3. Yadzhak M. S., Tyutyunyk M. I. An optimal algorithm to solve digital filtering problem with the use of adaptive smoothing. *Cybernetics and Systems Analysis*. 2013. Vol. 49, N 3. P. 449–456.
4. Jadzhak M. S. On a numerical algorithm of solving the cascade digital filtration problem. *Journal of automation and information sciences*. 2004. Vol. 36, N 6. P. 23–34.
5. Грубий Н., Яджак М. Застосування нейронних мереж для фільтрації сигналів. *Матеріали конф. мол. учених «Підстригачівські читання–2025», 27–29 травня 2025 р.* Львів: ІППММ ім. Я. С. Підстригача НАН України, 2025. Реж. дост.: [www.iapmm.lviv.ua/chyt2025/abstracts/Grubyi.pdf](http://www.iapmm.lviv.ua/chyt2025/abstracts/Grubyi.pdf).

6. Valkovskii V. A. An optimal algorithm for solving the problem of digital filtering. *Pattern Recognition and Image Analysis*. 1994. Vol. 4, N 3. P. 241–247.
7. Яджак М. Про чисельну реалізацію каскадної цифрової фільтрації. *Вісник Львів. ун-ту. Серія прикл. матем. та інформат.* 2000. Вип. 3. С. 75–79.
8. Cheryala N. K. Systolic Arrays and the TPU: [Electronic resource]. 2020. Available: <https://www.linkedin.com/pulse/Systolic-arrays-tpu-neeraj-cheryala>.
9. The list Top500: [Electronic resource]. Available: [www.top500.org](http://www.top500.org).
10. Луцків А. М., Лупенко С. А., Пасічник В. В. Паралельні та розподілені обчислення: навч. посіб. Львів: ПП «Магнолія 2006», 2024. 565 с.
11. Kalyaev I. A., Levin I. I., Semernikov E. A., Shmoilov V. I. Reconfigurable Multipipeline Computing Structures (Computer Science, Technology and Applications). Nova Science Pub Inc, 2013. 315 p.
12. Терейковський І. А., Бушуєв Д. А., Терейковська Л. О. Штучні нейронні мережі: базові положення: навч. посіб. Київ: КІП ім. Ігоря Сікорського, 2022. 123 с.
13. Ткаліченко С. В. Штучні нейронні мережі: навч. посіб. Кривий Ріг: Державний ун-т економіки і технологій, 2023. 150 с.
14. Oppenheim A. V., Schaffer R. W., Buck J. R. Discrete-Time Signal Processing'. Upper Saddle River. New Jersey, 1999. 893 p.
15. Новотарський М. А., Нестеренко Б. Б. Штучні нейронні мережі: обчислення. Київ: Ін-т математики НАН України, 2004. 408 с.
16. Грубий Н. В. Застосування нейронних мереж для фільтрації одновимірних сигналів: кваліфікаційна (магістерська) робота, спеціальність 122 – «Комп'ютерні науки». Львів: ЛНУ імені Івана Франка, 2025. 59 с.
17. Valdez L. D., Braunstein L. A., Havlin S. Epidemic spreading on modular networks: the fear to declare a pandemic. *arXiv: 1909.09695v2 [physics.soc-ph]*. 23 March 2020. 38 p.
18. Jackson M. O. Social and economic networks. Princeton: Princeton University Press, 2010. 520 p.
19. Demydyuk M. V., Lytwyn B. A. Optimization of the parameters of feet and the laws of motion of bipedal walking robots. *Journal of Mathematical Sciences*. 2023. Vol. 270, N 1. P. 214–236.
20. Demydyuk M. V., Hoshovs'ka N. V. Parametric optimization of the transport operations of a two-link manipulator. *Journal of Mathematical Sciences*. 2019. Vol. 238, N 2. P. 174–188.

## **Progress of parallel methods and algorithms for solving digital filtering problems**

Yadzhak Mykhailo, Tyutyunnyk Maria, Bekas Bogdan

*The quasisystolic method of organizing computations has been progressed for solving filtering problems using adaptive smoothing and multiple cascade digital filtering. This method differs from the purely systolic one in that it allows data transmission from one port to several reception points at once. Based on the mentioned method, optimal by speed parallel-pipeline algorithms for digital filtering of distorted data have been constructed. Optimality is proven in the specified classes of algorithms, which are equivalent in terms of the information graph. These algorithms are oriented towards implementation on quasisystolic computational structures and computers with a structural-procedural organization of calculations. The similarity of the structure of computations according to these filtering algorithms and computations performed in convolutional neural networks gives grounds for using the latter when developing a more universal filter for preprocessing data of various types. For this purpose, a hybrid neural network architecture is proposed – a convolutional autoencoder. Numerical experiments have confirmed the high efficiency of such a filter. The obtained scientific results can be used in the pre-processing of large data arrays in many subject areas using modern computing means.*

Отримано 01 05 2026