

**Конференція молодих учених «Підстригачівські читання – 2026»,
27–29 травня 2026 р., Львів**

УДК 519.681.5

ПРОГРАМНА РЕАЛІЗАЦІЯ ДЕЯКИХ ПАРАЛЕЛЬНИХ АЛГОРИТМІВ ЦИФРОВОЇ ФІЛЬТРАЦІЇ НА БАГАТОЯДЕРНОМУ КОМП'ЮТЕРІ

Ростислав Вдович

ІППММ ім. Я.С. Підстригача НАН України, rostyslav.r.vdovych@gmail.com

Дослідження об'єктів складних ієрархічно-мережевих систем (СІМС) [1] потребує попереднього опрацювання великих масивів даних. Ці дані зазвичай надходять постійно, тому їх обробка повинна виконуватись у режимі реального часу. Для цього використовується процедура цифрової фільтрації даних та паралельне програмування, яке забезпечує необхідну швидкість обчислень.

У [2] наведено деякі паралельні алгоритми виконання цифрової фільтрації на багатоядерних комп'ютерах, зокрема з використанням методу пірамід для розпаралелювання циклів. Проте, згідно цього методу, в паралельних гілках здійснюються дублювання обчислень. Тому в одному з запропонованих алгоритмів зроблено спробу усунення таких дублювань. У [3] проведено аналіз реалізації деяких паралельних алгоритмів фільтрації. Також, у [4] здійснено порівняння реалізації на комп'ютерах зі спільною та розподіленою пам'яттю [5].

Було покращено програмну реалізацію алгоритмів фільтрації, а сама програма розташована на веб-сервісі GitHub [6]. На відміну від вищезгаданих результатів, надалі для проведення обчислень використовувався багатоядерний комп'ютер з процесором AMD Ryzen 9 7950X (16 обчислювальних ядер, 32 потоки).

Унаслідок проведених чисельних експериментів було одержано реальне прискорення алгоритмів з дублюваннями обчислень та без них. Відповідні результати прискорення з використанням спільної пам'яті складають 12,78 та 14,63 разів, а розподіленої – 16,16 та 16,59 разів. Отримані результати можуть використовуватися для аналізу та оптимізації паралельних алгоритмів розв'язання алгоритмічно складних задач, зокрема дослідження об'єктів СІМС. Реалізовані алгоритми фільтрації можуть використовуватись для попереднього опрацювання великих масивів даних в режимі реального часу.

Конференція молодих учених «Підстригачівські читання – 2026», 27–29 травня 2026 р., Львів

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. *Вдович Р. Р., Яджак М. С.* Про оптимізацію деяких паралельних алгоритмів цифрової фільтрації великих масивів даних // Сучасні комп'ютерні та інформаційні системи і технології: матеріали IV Всеукр. наук.-практ. Інтернет-конф., Запоріжжя, 9–20 грудня 2024 р. – Запоріжжя: ТДАТУ, 2024. – С. 9–12.
3. *Вдович Р. Р., Яджак М. С.* Аналіз реалізації паралельних алгоритмів цифрової фільтрації великих масивів спотворених даних // Фіз.-мат. моделювання та інформаційні технології. – 2025. – Вип. 40 – С. 47–52.
4. *Vdovych R.* On the implementation of some parallel filtering algorithms on computers with shared and distributed memory // XII International Skorobohatko Mathematical Conference (September 23 – 25, 2025, Lviv, Ukraine). – Lviv: Pidstryhach Institute for Applied Problems of Mechanics and Mathematics NAS of Ukraine, 2025. – P. 100.
5. *Pacheco P. S., Malensek M.* An introduction to parallel programming, second edition. – Cambridge, MA: Morgan Kaufmann, 2022. – 496 p.
6. *Vdovych R. R.* CDF_V3. – 2026. – Режим доступу: https://github.com/RostyslavR-Vdovych/CDF_V3.

SOFTWARE IMPLEMENTATION OF SOME PARALLEL DIGITAL FILTERING ALGORITHMS ON A MULTI-CORE COMPUTER

Research on complex hierarchical-network systems requires previous processing of large data sets. Since such data are typically generated continuously, their processing must be performed in real time. For this purpose, digital filtering and parallel programming are used. The analyzed parallel algorithms are based on the pyramid method for loop parallelization and on elimination of redundant computations. The algorithms were implemented and tested on a multicore AMD Ryzen 9 7950X computer. Experiments showed speedups of 12.78 and 14.63 (shared memory) and 16.16 and 16.59 (distributed memory). The results can be used in analysis and optimization of parallel algorithms for solving algorithmically complex problems. The implemented filtering algorithms can be used for preprocessing large data sets in real time.