

ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ ТА КОМП'ЮТЕРНА ТЕХНІКА

УДК 681.3.055

Т. Б. Мартинюк, к. т. н., доц.;

В. В. Хом'юк, к. т. н., доц.

ОЦІНЮВАННЯ ЕФЕКТИВНОСТІ АЛГОРИТМІВ МУЛЬТИБРОБКИ МАСИВІВ ДАНИХ

Обґрунтовано вибір коефіцієнта узгодження пари «структура-алгоритм» в якості критерію ефективності для алгоритмів мультипідсумовування елементів векторного масиву даних. Проаналізовано особливості паралельних алгоритмів мультиобробки, а саме, алгоритмів логарифмічного, рекурсивного підсумовування та з використанням різницевого зрізів.

Вступ

Відомо [1—5], що теоретичні дослідження продуктивності паралельної системи можна проводити за різними критеріями. Зокрема, в роботі [1] описані критерії оцінювання ефективності якості інформаційних систем, а саме, показано, що ефективність характеризується здатністю системи забезпечити передачу даної кількості інформації з найменшими затратами потужності сигналів, часу та смуги частот. Наведені критерії ефективності [1, 2] розроблені для інформаційних систем і набули статусу класичних у галузі інформаційно-обчислювальних засобів. Але разом з тим, для кожного конкретного випадку визначення ефективності обчислювальних методів та засобів необхідно ретельно обирати і обґрунтовувати вибір критерію ефективності з найуживаніших у цій галузі [4—8].

Постановка задачі

Відомо, що найрозповсюдженішими характеристиками паралельних алгоритмів вважаються ширина паралелізму l_n (паралелізм задачі), глибина паралелізму q_n (алгоритмічна складність) і прискорення S [4, 6, 7]. Причому, ширина паралелізму l_n характеризує в даному випадку розмірність задачі, тобто кількість операндів і їх розрядів, що обробляються одночасно, а глибина паралелізму q_n відображає час виконання алгоритму в тактах, тобто його швидкодію. Таким чином, величина l_n описує алгоритм по горизонталі, а величина q_n — по вертикалі. Для оцінювання паралельних алгоритмів також доцільно скористатись такими відомими характеристиками, як ступінь паралелізму і середній ступінь паралелізму. За означенням *ступенем паралелізму* чисельного алгоритму називається кількість його операцій, які можна виконувати паралельно, а *середнім ступенем паралелізму* чисельного алгоритму називається відношення загальної кількості операцій алгоритму до кількості його етапів (циклів, кроків) [8].

Прискорення паралельного алгоритму визначається з виразу [4]

$$S_p = \frac{T_1}{T_p}, \quad (1)$$

де T_1 — час виконання алгоритму на одному процесорі, T_p — час виконання алгоритму у системі з p процесорів. Також слід відзначити, що на ефективність паралельної ЕОМ (паралельних програм) впливає залежність виконання основних арифметичних операцій від виду операндів, а прискорення такої ПЕОМ можна визначити таким чином [6]:

$$S = \frac{W t_{nc} d}{W_{ck} t_{ck} + W_{pn} t_{pn} / p + \Delta W t_k / u}, \quad (2)$$

де $W = W_{ck} + W_{pn}$ — кількість операцій в задачі, яка ділиться на скалярну W_{ck} та розпаралельнену W_{pn} складові; ΔW — кількість операцій, які використовуються на комутацію; $t_{nc}, t_{ck}, t_{pn}, t_k$ — час такту роботи відповідно послідовної ЕОМ, скалярного процесора, ПЕ та комутатора; p — кількість ПЕ; u — кількість зв'язків, які забезпечуються комутатором одночасно (паралелізм комутатора); d — коефіцієнт поєднання скалярного, розпаралеленого та комутаційного потоків операцій.

Серед методів оцінювання ефективності алгоритмів розпаралелювання послідовних задач перевагу слід віддавати тим методам, які враховують взаємозв'язок між паралелізмом задачі та паралелізмом запропонованої для реалізації цієї задачі структури. Таким можна вважати метод, який використовує коефіцієнт узгодження [9]. Це означає, що багатофункціональність, яка характерна для паралельного алгоритму, дозволяє досягти значного наближення до оптимального значення коефіцієнта узгодження [10], який є важливим фактором урахування взаємозв'язку між часовими витратами та конкретною реалізацією алгоритму A

$$\varphi_A = \frac{1}{2}(\pi_A^* + \rho_A^*), \quad (3)$$

де π_A^* — коефіцієнт структурної надлишковості; ρ_A^* — коефіцієнт корисного інформаційного використання структури, які визначаються за структурною надлишковістю π_A й інформаційним корисним використанням ρ_A

$$\pi_A^* = \frac{\pi_A}{C} = \frac{1}{C} \sum_{i=1}^S \frac{c_i}{v_i}; \quad (4)$$

$$\rho_A^* = \frac{\rho_A}{C} = \frac{1}{C} \sum_{i=1}^S c_i v_i, \quad (5)$$

де C — вартість структури, за формулою

$$C = \sum_{i=1}^S c_i N_i, \quad (6)$$

c_i — вартість i -го блока; N_i — кількість i -х блоків структури; S — кількість блоків у структурі, v_i — коефіцієнт використання i -го блоку, який можна представити таким чином [9]:

$$v_i = \frac{1}{T_A} \sum_{l=1}^L \tau_{il} p_{il}, \quad (7)$$

де T_A — час виконання алгоритму; τ_{il} — час реалізації на i -му блоці операції l -го типу; p_{il} — кількість операцій l -го типу, які реалізуються за час T_A на i -му блоці; L — кількість типів операцій алгоритму. При цьому слід зауважити, що даний коефіцієнт узгодження може бути використаний не тільки при розв'язанні економічних задач [9], але й для визначення ефективності розв'язування технічних задач.

Аналіз ефективності алгоритмів мультипідсумовування числових даних

Означення. Мультипідсумовуванням назвемо процес багатооперандного (багатомісного) підсумовування елементів числової множини Ω у вигляді масиву числових даних, розмірність якої визначає задача, що містить у собі процес підсумовування [11].

За множини Ω будемо розглядати векторний масив числових даних відповідної розмірності. Аналіз ефективності алгоритмів мультипідсумовування числових даних проведемо на прикладі трьох алгоритмів мультипідсумовування: деревоподібного алгоритму або алгоритму логарифмічного підсумовування (рис. 1а); алгоритму рекурсивного подвоєння (рис. 1б); алгоритму підсумовування за методом різницевого зрізів (РЗ) (рис. 1в). Основні характеристики послідовного та даних алгоритмів наведено у таблиці. Як доповнення до даних таблиці необхідно зазначити, що для всіх алгоритмів, крім алгоритму за методом РЗ, кількість тактів i , відповідно, час виконання алго-

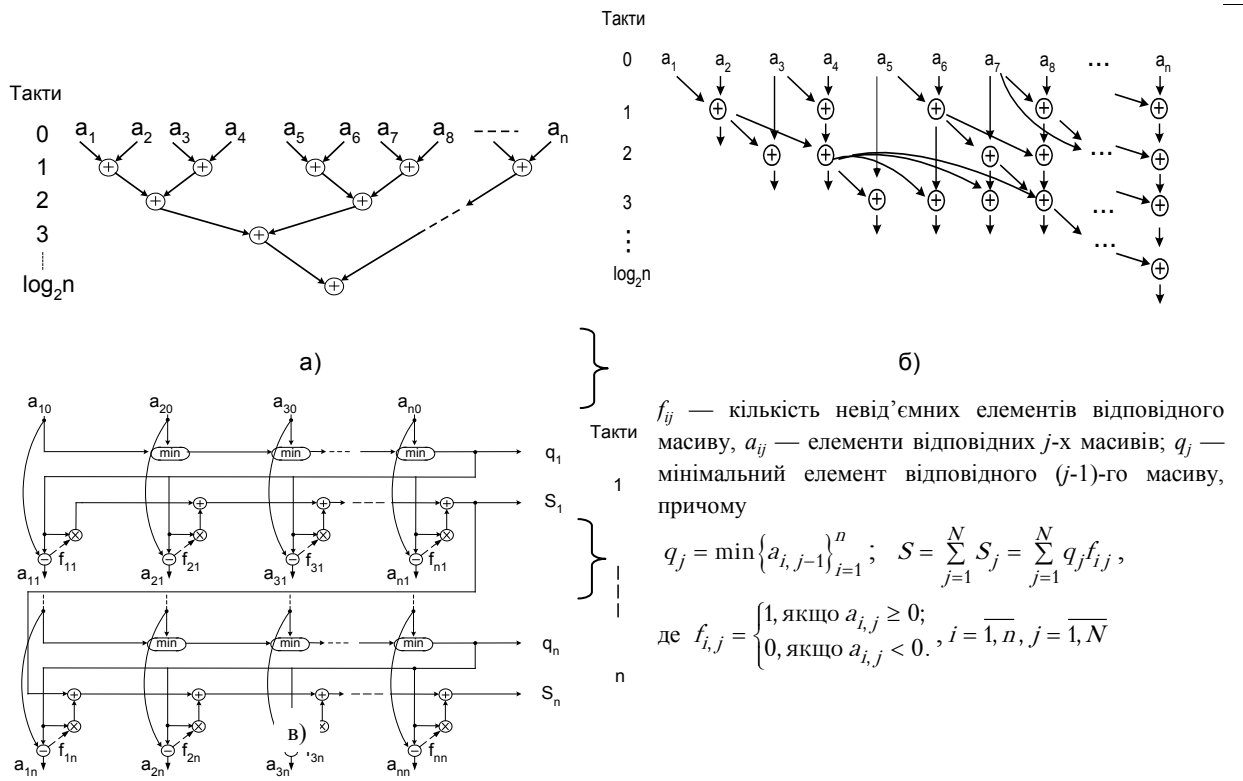


Рис. 1. Орієнтовані графи процесу мультипідсумовування
 а) алгоритм логарифмічного підсумовування б) алгоритм рекурсивного подвоєння
 в) алгоритм підсумовування за методом різницевого зрізу

ритму є величиною фіксованою, а для останнього у таблиці наведено максимальне значення цих параметрів.

Основні характеристики алгоритмів мультипідсумовування

Параметри	Алгоритми реалізації оператора мультипідсумовування			
	Послідовний алгоритм	Алгоритм логарифмічного підсумовування	Алгоритм рекурсивного подвоєння	Алгоритм з використанням різницевого зрізу
Кількість тактів, t	$n - 1$	$\log_2 n$	$\log_2 n$	n
Кількість процесорів, p	1	$n/2 \leq p \leq n - 1$	$n/2 \leq p \leq n/2 \lceil \log_2 n \rceil$	n
Загальна кількість операцій алгоритму, ω	$L(n - 1)$	$L(n - 1)$	$L(n/2 \lceil \log_2 n \rceil)$	$L(4n^2 - 2n)$
Час виконання алгоритму, T_p^n	$\tau(n - 1)$	$\tau \log_2 n$	$\tau \log_2 n$	$\tau n(n - 1)$
Кількість та тип операцій, L	1 (додавання)	1 (додавання)	1 (додавання)	4

Примітки: τ — час виконання одного такту алгоритму; n — розмірність початкового векторного масиву

Враховуючи залежності (3)—(7) коефіцієнт узгодження представимо таким чином:

$$\varphi_A = \frac{0,5}{C} \sum_{i=1}^S c_i \left(T_A / \sum_{l=1}^L \tau_{il} p_{il} + \sum_{l=1}^L \tau_{il} p_{il} / T_A \right). \quad (8)$$

Послідовний алгоритм

Нехай вартість c_i кожного i -го блоку однакова, тобто

$$c_i = \text{const} = c'. \quad (9)$$

Це означає, що $\sum_{i=1}^S c_i = c'$, оскільки $S = 1$. Крім того, через те, що час τ_{il} реалізації на кожному i -му блоці операції підсумовування однаковий

$$\tau_{il} = \text{const} = \tau_1, \quad (10)$$

то $\sum_{l=1}^L \tau_{il} p_{il} = \tau_1 p_i = \tau_1 (n-1)$, оскільки $L = 1$ і $p_i = \omega = n-1$.

$$\text{Таким чином, для послідовного алгоритму } \varphi_A^1 = \frac{0,5}{C} c' \left(\frac{\tau(n-1)}{\tau_1(n-1)} + \frac{\tau_1(n-1)}{\tau(n-1)} \right).$$

Оскільки τ — час виконання одного такту алгоритму підсумовування, τ_1 — час виконання операції підсумовування і $\tau = \tau_1$, а $C = c'$, то

$$\varphi_A^1 = 1. \quad (11)$$

Далі розглянемо паралельні алгоритми мультипідсумовування.

Алгоритм логарифмічного підсумовування

Враховуючи дані таблиці, припущення (9) і (10), а також те, що $p_i = \frac{\omega}{t} = \frac{n-1}{\log n}$,

тобто $v_i(A) = \frac{1}{\tau \log n} \tau \frac{n-1}{\log n} = \frac{n-1}{\log^2 n}$, маємо

$$\varphi_A^2 = \frac{0,5}{C} \sum_{i=1}^S c_i \left(\frac{\tau \log^2 n}{\tau(n-1)} + \frac{\tau(n-1)}{\tau \log^2 n} \right) = \frac{0,5}{C} \sum_{i=1}^S c_i \left(\frac{\log^2 n}{n-1} + \frac{n-1}{\log^2 n} \right).$$

Оскільки $C = \sum_{i=1}^S c_i N_i$, де N_i — кількість i -х

блоків, то, враховуючи припущення (9), отримаємо $C = c' \sum_{i=1}^S N_i = c' \frac{n}{2}$.

Таким чином, коефіцієнт узгодження для алгоритму логарифмічного підсумовування має вигляд

$$\varphi_A^2 = 0,5 \left(\frac{\log^2 n}{n-1} + \frac{n-1}{\log^2 n} \right). \quad (12)$$

На рис. 2 показано графік залежності для $\varphi_A^2 = \varphi(n)$, де $n = 2, \dots, 64$.

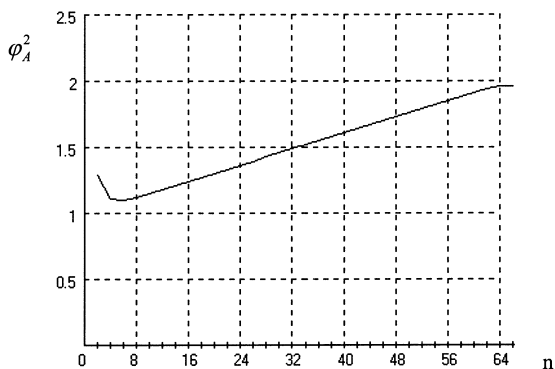


Рис. 2. Графік залежності для $\varphi_A^2 = \varphi(n)$ (алгоритм логарифмічного підсумовування)

Алгоритм рекурсивного підсумовування

Враховуючи дані таблиці, припущення (9) і (10), а також те, що $p_i = n/2$ в даному випадку отримаємо

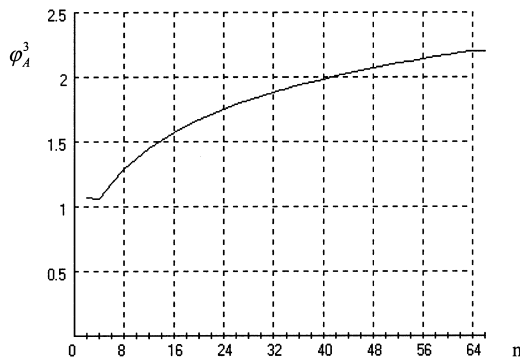


Рис. 3. Графік залежності для $\varphi_A^3 = \varphi(n)$ (алгоритм рекурсивного підсумовування)

$$\varphi_A^3 = 0,5 \left(\log n + \frac{1}{\log n} \right), \quad (13)$$

На рис. 3 показано графік залежності для $\varphi_A^3 = \varphi(n)$, де $n = 2, \dots, 64$.

Алгоритм мультипідсумовування за методом РЗ

У нашому випадку для алгоритму на базі різнице-вих зрізів коефіцієнт v_i використання i -го блока можна подати, з урахуванням кількості алгоритмів, які одночасно реалізуються на даній структурі, таким чином [12]:

$$v_i = \frac{1}{T_A F} \sum_{l=1}^L \tau_{il} p_{il}, \quad (14)$$

де F — кількість алгоритмів, які одночасно реалізуються на даній структурі. Зокрема, в нашому випадку $T_A = N_{\max} = n$, $L = 4$, а $F = 3$ за рахунок того, що на одній структурі одночасно реалізуються: алгоритм мультипідсумовування; алгоритм сортування, наприклад, у порядку збільшення елементів; алгоритм формування поточного порога як середнього значення суми елементів масиву при пороговому обробленні інформації [13]. В результаті будемо мати:

$$v_i^{\min} = \frac{1}{3n} \sum_{l=1}^4 \tau_{il} p_{il}. \quad (15)$$

Відповідно для $T_{\min} = 1$, $L = 4$, $F = 3$

$$v_i^{\max} = \frac{1}{3} \sum_{l=1}^4 \tau_{il} p_{il}. \quad (16)$$

Таким чином, зі зменшенням часу виконання алгоритму, що характерно для масивів чисел із групами однакових елементів [13], відбувається зростання коефіцієнта використання i -го блока v_i .

Розглянемо далі специфіку отримання коефіцієнта узгодження φ_A^4 .

Враховуючи співвідношення (3)—(7) під час визначення коефіцієнта узгодженості φ_A , припущення (9) і (10) та формулу (14), а також те, що в даному випадку $T_A^{\max} = (n-1)n\tau \approx n^2\tau$, отримаємо

$$\varphi_A^4 = \frac{0,5}{n} \sum_{i=1}^S \left(\frac{4\tau n^2}{\sum_{l=1}^4 \tau_{il} p_{il}} + \frac{\sum_{l=1}^4 \tau_{il} p_{il}}{4\tau n^2} \right).$$

Оскільки $\sum_{l=1}^4 \tau_{il} p_{il} = 4n\tau$ і $p_i = L(n+1) - 1$, то для алгоритму мультипідсумовування за методом РЗ отримаємо

$$\varphi_A^4 = \frac{0,5}{n} \left(\frac{4n+3}{3n} + \frac{3n}{4n+3} \right) n,$$

або

$$\varphi_A^4 = 0,5 \left(\frac{4n+3}{3n} + \frac{3n}{4n+3} \right). \quad (17)$$

На рис. 4 показано графік залежності для $\varphi_A^4 = \varphi(n)$, де $n = 2, \dots, 64$.

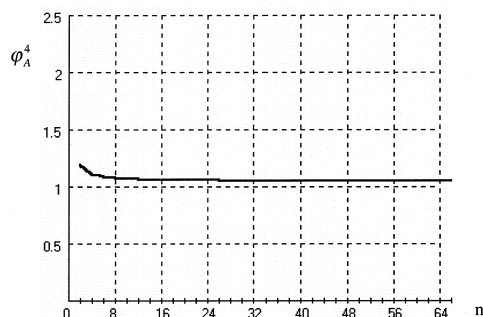


Рис. 4. Графік залежності для $\varphi_A^4 = \varphi(n)$
(алгоритм за методом РЗ)

Типідсумовування за методом різницевого зрізу навпаки — після $n > 8$ спостерігається асимптотичне наближення зверху до прямої $\varphi_0 = 1$.

Таким чином, у результаті дослідження коефіцієнта узгодження φ_A отримано графік (рис. 5), що ілюструє залежність для $\varphi_A^k = \varphi(n)$ відповідно для розглянутих паралельних алгоритмів мультипідсумовування ($k = 2, 3, 4$).

З аналізу формул (12), (13) і (17) та отриманих графіків (рис. 5) випливає, що для алгоритмів логарифмічного і рекурсивного підсумовування оптимальною з огляду на найближче до $\varphi_0 = 1$ значення коефіцієнта узгодження є розмірність вхідного масиву $n = 6$. Із зростанням n коефіцієнти узгодження φ_A^2 і φ_A^3 зростають, причому φ_A^2 повільніше, ніж φ_A^3 . Для алгоритму мульти-

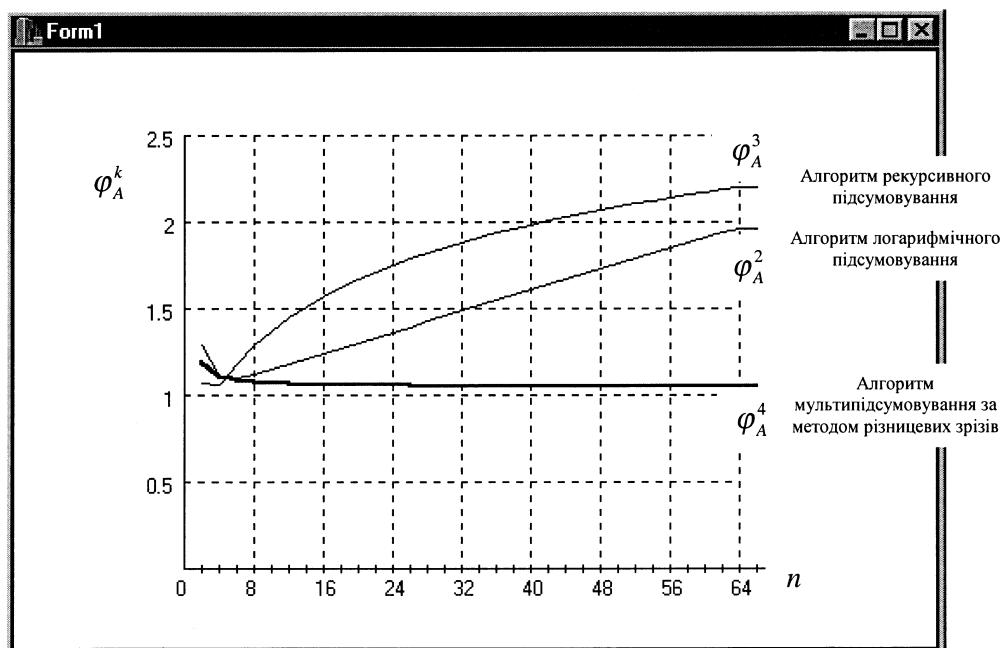


Рис. 5 Графік залежності $\varphi_A^k = \varphi(n)$ для наведених паралельних алгоритмів мультипідсумовування

Отже, доцільність використання для оброблення значних векторних масивів інформації алгоритму мультипідсумовування за методом РЗ, оскільки тут існує можливість виконання на одній структурі не одного, а декількох алгоритмів, що враховує коефіцієнт узгодження. Графіки на рис. 5 підтверджують, що структура та алгоритм, який на ній реалізується, краще узгоджені в тому випадку (значення коефіцієнта φ_A ближче до оптимального значення $\varphi_0 = 1$), коли, як показали дослідження [12] $L = F$, тобто коли кількість операцій дорівнює кількості алгоритмів, що одночасно реалізуються на структурі.

Висновки

1. Вибираючи оптимальний метод оцінювання ефективності алгоритмів розпаралелювання послідовних задач, найприйнятнішим через коректність і наочність можна вважати коефіцієнт узгодження Φ_A .

Багатофункціональність, яка характерна для алгоритму мультипідсумовування з використанням різницевих зрізів, забезпечує значне наближення до оптимального значення коефіцієнта узгодження незалежно від розмірності вхідного векторного масиву, що дозволяє віднести даний алгоритм до перспективних, особливо в разі його застосування при побудові нейроподібних мереж (нейрочіпів і нейроприскорювачів), для елементів яких (формальних нейронів) однією з визначальних є процедура формування суми зважених вхідних сигналів, кількість яких в реальних нейроподібних мережах є значною.

СПИСОК ЛІТЕРАТУРИ

1. Кузьмин И. В., Кедров В. А. Основы теории информации и кодирования — 2-е изд., перераб. и доп. — К.: Вища шк. Головное изд-во, 1986. — 238 с.
2. Кузьмин И. В. Оценка эффективности и оптимизации автоматизированных систем контроля и управления. — М.: Советское радио, 1971. — 296 с.
3. Барский А. Б. Параллельные процессы в вычислительных системах: Планирование и организация. — М.: Радио и связь, 1990. — 256 с.
4. Системы параллельной обработки: Пер. с англ. / Под ред. Д. Ивенса. — М.: Мир, 1985. — 416 с.
5. Хохлюк В. И. Параллельные алгоритмы целочисленной оптимизации. — М.: Радио и связь. — 1987. — 224 с.
6. Шпаковский Г. И. Архитектура параллельных ЭВМ. — Минск: Университетское, 1989. — 192 с.
7. Воробьев В. А. Об эффективности параллельных вычислений // Автометрия. — 2001. — № 1 — С. 50—58.
8. Ортега Дж. Введение в параллельные и векторные методы решения линейных систем: Пер. с англ. — М.: Мир, 1991. — 367 с.
9. Трухний В. Д. Структурно-информационные оценки качества цифровых вычислительных машин // Экономика и математические методы. — 1969. — Вып. 1. — С. 79—89.
10. Мартинюк Т. Б., Заболотна Н. І., Шолота В. В. Оцінювання структурно — інформаційної складності алгоритму паралельного додавання чисел // ВісникВІП. — 1996. — № 4. — С. 21—26.
11. Хом'юк В. В. Дослідження процесу прискорення мультипідсумовування методом різницевих зрізів // Тези доповідей ІХ-ої всеукраїнської наукової конференції «Сучасні проблеми прикладної математики та інформатики». — Львів, 2002. — С. 129—130.
12. Мартинюк Т. Б. Рекурсивні алгоритми багатооперандної обробки інформації. Монографія. — Вінниця: «Універсум-Вінниця», 2000. — 216 с.
13. Хом'юк В. В. Методи та засоби паралельних перетворень векторних масивів даних: Автореф. дис...канд. техн. наук: 05.13.13 / Вінниц. нац. техн. ун-т. — Вінниця, 2003. — 20с.

Рекомендована кафедрою прикладної математики

Надійшла до редакції 12.10.04
Рекомендована до друку 5.07.05

Мартинюк Тетяна Борисівна — доцент кафедри лазерної та оптоелектронної техніки; **Хом'юк Віктор Вікторович** — доцент кафедри прикладної математики.

Вінницький національний технічний університет