

## MILITARY AFFAIRS AND NATIONAL SECURITY

*DOI 10.51582/interconf.21-22.05.2021.034*

**Голубничий Дмитро Юрійович**

ORCID ID: 0000-0002-6873-7004

кандидат технічних наук, доцент

доцент кафедри Інформаційних систем

Харківський національний економічний університет імені Семена Кузнеця, Україна

**Третяк Вячеслав Федорович**

ORCID ID: 0000-0003-2599-8834

кандидат технічних наук, доцент, старший науковий співробітник

науковий співробітник наукового центру Повітряних Сил

Харківський національний університет Повітряних Сил ім. Івана Кожедуба, Україна

**Костенко Ігор Леонідович**

ORCID ID: 0000-0003-4594-7727

кандидат військових наук, старший науковий співробітник

начальник управління наукового центру Повітряних Сил

Харківський національний університет Повітряних Сил ім. Івана Кожедуба, Україна

**Поляцко Володимир Васильович**

ORCID ID: 0000-0002-4111-7956

Начальник центру зв'язку об'єднаного штабу Командування об'єднаних сил ЗС України

**Апполонов Олександр Опанасович**

ORCID ID: 0000-0001-7522-762X

Заступник начальника штабу в/ч А0307, Україна

**Крук Богдан Миронович**

ORCID ID: 0000-0002-0937-8777

кандидат технічних наук, старший науковий співробітник

провідний науковий співробітник наукового центру Повітряних Сил

Харківський національний університет Повітряних Сил імені Івана Кожедуба, Україна

**Закіров Замір Забірович**

ORCID ID: 0000-0002-8456-9733

кандидат технічних наук, старший науковий співробітник  
старший науковий співробітник науково-дослідного відділу  
наукового центру Повітряних Сил

Харківський національний університет Повітряних Сил імені Івана Кожедуба, Україна

## **ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ АЛГОРИТМІВ РІШЕННЯ ЗАДАЧ ДИСКРЕТНОЇ ОПТИМІЗАЦІЇ НА ОСНОВІ РАНГОВОГО ПІДХОДУ ТА ПРИНЦИПУ ОПТИМІЗАЦІЇ ЗА НАПРЯМКОМ**

*Анотація.* Наведено результати експериментального дослідження алгоритмів рішення задач дискретної оптимізації на основі рангового підходу та принципу оптимізації за напрямком.

*Ключові слова:* оптимізація, цілочисельне лінійне програмування, ранговий підхід.

На основі стратегій  $\{L_w\}$  і правил  $\{K_w\}$  розроблено алгоритми рішення задачі цілочисельного лінійного програмування (ЦЛП) з булевими змінними (БЗ). У результаті були одержані одноетапні і багатоетапні алгоритми, класифікація яких зображені на рис. 1.

Основними кваліфікаційними ознаками є кількість етапів і точність етапів рішення задачі. За першою ознакою алгоритми діляться на одноетапні і багатоетапні. Дане розділення означає, що початкове завдання може розв'язуватися один (одноетапні) або кілька разів (багатоетапні) різними алгоритмами, перелік яких поданий у дужках. Друга ознака визначає погрішність одержуваного допустимого рішення задачі ЦЛП з БП за вагою функціонала від оптимального рішення. Якщо алгоритм принципово не може дати точного рішення за вагою функціонала, то такий алгоритм називатимемо наближеним [1-6].

Метою експериментального дослідження є:

1. Практична перевірка правильності висунутих стратегій відсікань  $\{L_w\}$  і  $\{K_i\}$  заснованих на доведених у розділі 2 теоремах і твердженнях.

2. Оцінка тимчасової складності запропонованих алгоритмів.
3. Отримання числових значень вибраних показників якості алгоритмів в середньому.
4. Порівняння показників ефективності розроблених алгоритмів з відомими.
5. Аналіз причин отримання неточних рішень наближеними алгоритмами.

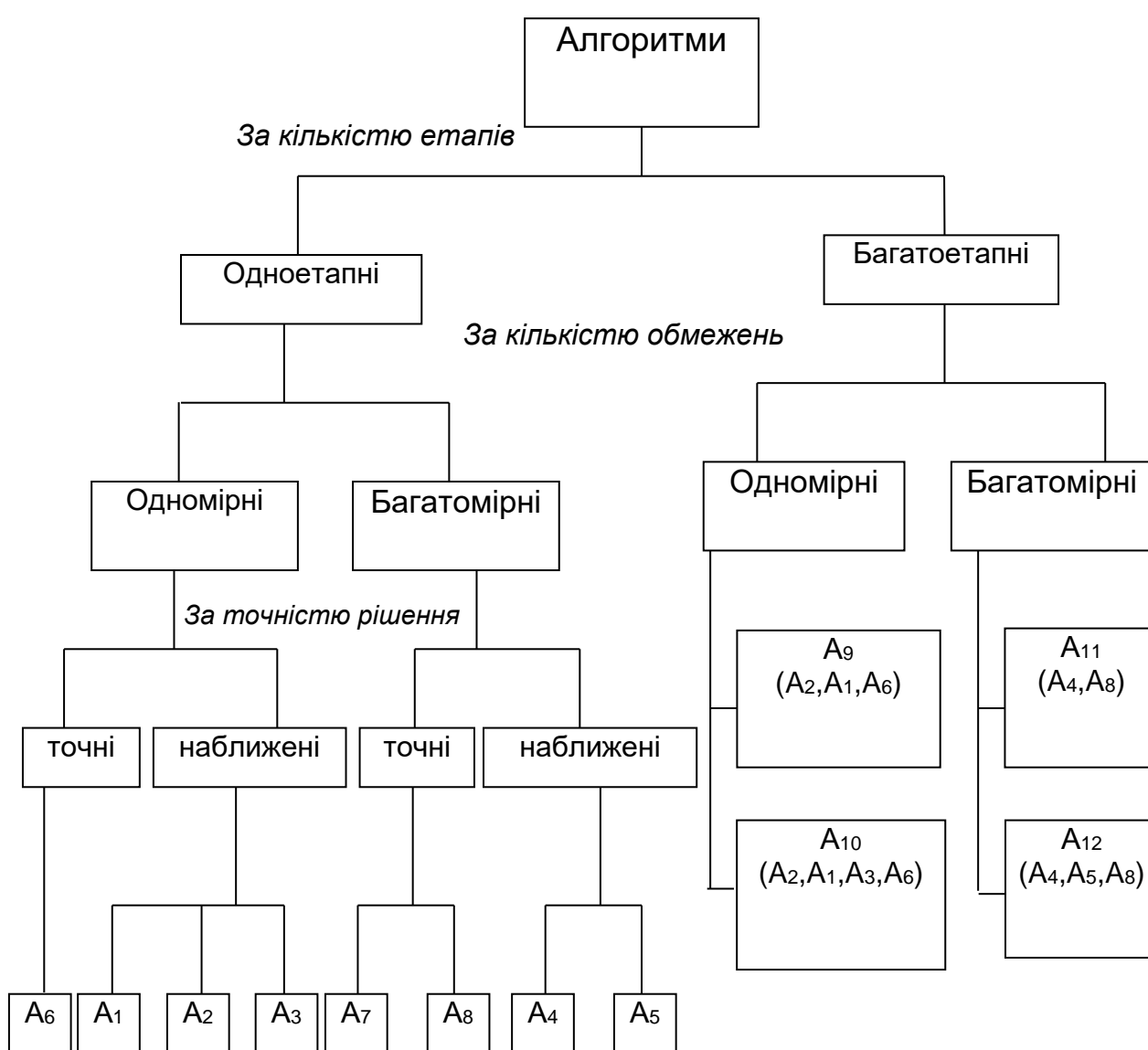


Рис. 1. Класифікація алгоритмів на основі узагальненої процедури  $A_0$  [1]

Для перевірки правильності і точності рішення розробленими алгоритмами  $A_1 - A_{12}$  в якості еталона вибраний модифікований алгоритм

Балаша, що враховує вимоги позитивності коефіцієнтів у функціоналі. Надалі під алгоритмом D розумітимемо саме цей алгоритм.

У ході рішення тестових задач за допомогою датчика випадкових чисел генерувалися коефіцієнти у функціоналі в діапазоні  $[1 \div 100]$  і в обмеженнях умови в діапазоні  $[1 \div 10]$ . Вибір інших діапазонів для функціонала змінив лише його абсолютне значення, але в середньому не вплинув на параметри алгоритмів. Зміна діапазону в обмеженнях впливає тільки на ранг шляху (ранг шляху – кількість вершин графа ДД, які утворюють цей шлях).

Структура дослідження кожного алгоритму полягала в наступному. Розв'язувалося 1 000 тестових задач із заданими вхідними параметрами  $m$  і  $n$  вибраним алгоритмом. У ході рішення обчислювалися показники ефективності алгоритмів. Отримані значення показників зображувались на графіках, апроксимованих за методом найменших квадратів.

Усі алгоритми поділимо на три групи. Першу групу складають одновимірні алгоритми  $A_1, A_3, A_6$ , що вирішують задачі ЦЛП з БП. Другу групу складають  $m$ -мірні алгоритми  $A_4, A_5, A_7, A_8$ , що розв'язують задачі ЦЛП з БП. А третю групу утворюють багато-етапні одновимірні і  $m$ -мірні алгоритми  $A_9, A_{10}, A_{11}, A_{12}$  [1-4].

Алгоритм  $A_2$  не увійшов ні до однієї з груп, оскільки він, з одного боку, є одновимірним, а з іншого боку, використовується як один з етапів в багатоетапних одновимірних алгоритмах. Найбільший інтерес становить не сам алгоритм  $A_2$ , а властивість стратегії  $L_2$ , на основі якої він побудований. Ця властивість використовується в багатоетапних алгоритмах з метою ефективнішого відсіву векторів на першому ранзі. Тому дослідження алгоритму  $A_2$  не проводилося.

**Дослідження алгоритмів першої групи.** Як виявилось, кількісні значення вибраних показників істотно залежать від рангу  $\bar{\Gamma}$  одержуваного рішення, який визначає число одиниць в оптимальному рішенні. Це підтверджують рис. 2 і 3, на яких зображені залежності тимчасового показника  $N_{\text{оп}}$  алгоритмів від  $\bar{\Gamma}$  при фіксованих значеннях вхідних параметрів  $n$  і  $m=1$ .

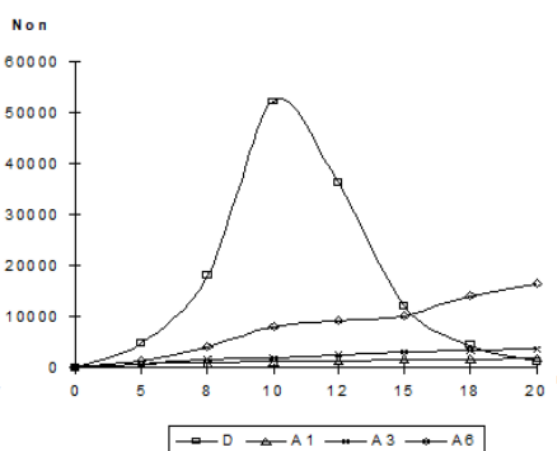
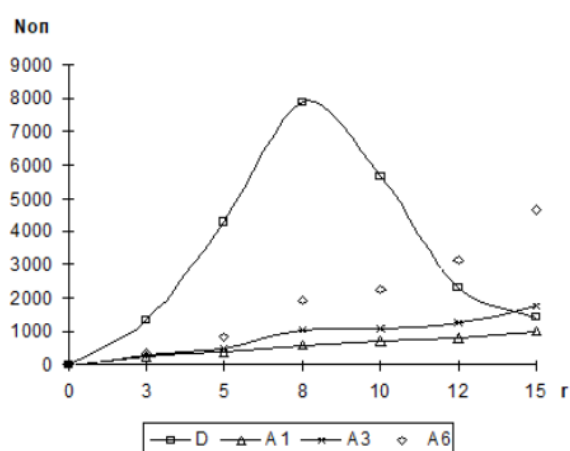
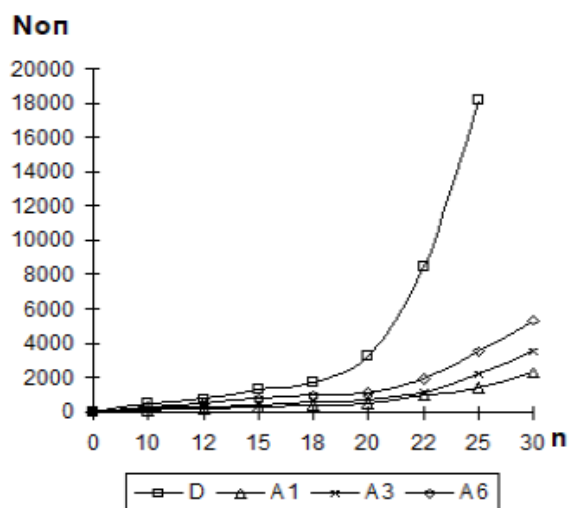
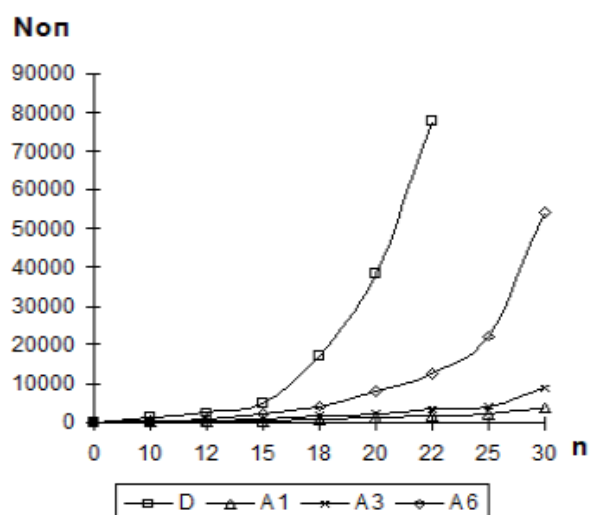
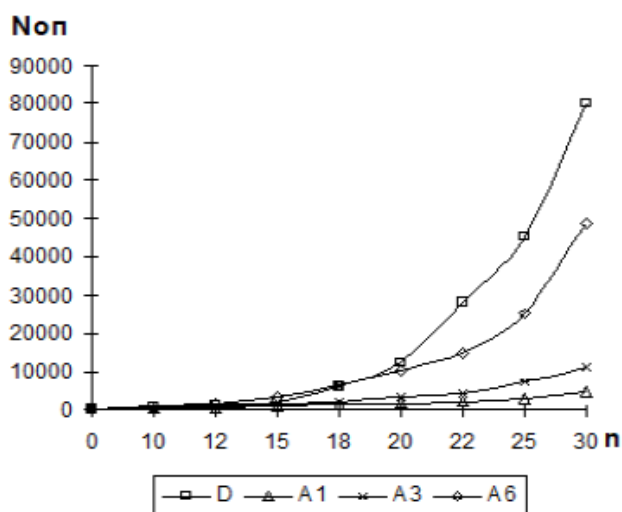


Рис. 2. Залежність  $N_{оп}$  від  $\bar{r}$  при  $n=15$  | Рис. 3. Залежність  $N_{оп}$  від  $\bar{r}$  при  $n=20$

З рис. 2 і 3 видно, що діапазон зміни  $\bar{r}$  можна умовно розбити на три зони: 1 зона –  $\bar{r} = [0 \div n/3]$ ; 2 зона –  $\bar{r} = [n/3 \div 2n/3]$ ; 3 зона –  $\bar{r} = [2n/3 \div n]$ . У першій зоні алгоритм D знаходить рішення швидко, оскільки за рахунок зондування дуже ефективно відсікаються гілки дерева рішень, що відповідають одиничним галузям. Аналогічним чином пояснюється і швидке отримання рішення в 3 зоні, вектори в якій складаються з великого числа одиниць. Прояв всієї експоненціальної складності алгоритму D відбувається саме в другій зоні. Для алгоритмів A1, A3, A6 зростання числа  $N_{оп}$  із збільшенням  $\bar{r}$  визначається числом локальних областей W, які необхідно обробити. Згідно з виразом 
$$\Omega = n + (n-1) + \dots + 1 = \frac{n \cdot (n+1)}{2} \approx \frac{n^2}{2}$$
 кількість таких областей W в графі DΔ не перевищує  $n^2/2$ , тому при  $\bar{r} \rightarrow n$  зростання числа  $N_{оп}$  також не перевищує  $c n^2$ , где  $c = \text{const}$ .

Таким чином, для об'єктивного порівняння алгоритмів необхідно указувати до якої зони вони належать, тобто який відсоток одиниць (або нулів) містить оптимальне рішення. Відповідно до ділення на зони побудуємо залежність числа  $N_{оп}$  від  $n$  для кожної зони. Ці залежності для зон 1, 2, 3 показані на рис. 4, 5, 6. Домовимося, що для отримання величини якого-небудь показника для зони будуть використані його усереднені значення на рангах, що належать цій зоні.

Рис. 4. Залежність  $N_{оп}$  від  $n$  для зони 1Рис. 5. Залежність  $N_{оп}$  від  $n$  для зони 2Рис. 6. Залежність  $N_{оп}$  від  $n$  для зони 3

Таким чином, для об'єктивного порівняння алгоритмів необхідно указувати до якої зони вони належать, тобто який відсоток одиниць (або нулів) містить оптимальне рішення. Відповідно до ділення на зони побудуємо залежність числа  $N_{оп}$  від  $n$  для кожної зони. Ці залежності для зон 1, 2, 3 показані на рис. 4, 5, 6. Домовимося, що для отримання величини якого-небудь показника для зони будуть використані його усереднені значення на рангах, що належать цій зоні.

З наведених рис. 4 – 6 видно, що для порівняння тимчасових показників алгоритмів необхідно чітко указувати приналежність до умовних зон, оскільки абсолютне значення цих показників в зонах істотно відрізняється. Попадання

рішення задачі в будь-яку зону якраз визначається співвідношенням діапазону зміни коефіцієнтів  $a_{1j}$  у обмеженні і заданого діапазону для  $b_1$  у правій частині

$$\sum_{j=1}^n a_{1j} \times x_j \leq b.$$

нерівності

Показник потрібного обсягу оперативної пам'яті  $V$  залежить від конкретної програмної реалізації. Кількісні значення цього показника, звісно, можуть бути змінені і безпосередньо залежать від кваліфікації програміста. Проте вид кривих не зазнає зміни із-за природи тих процесів, які вони описують. Покажемо, що і величина потрібного обсягу оперативної пам'яті для зберігання множини  $m_{sj}^r$  векторів графа  $D\Delta$  також залежить від середнього рангу оптимального рішення тестової задачі. Залежність  $V=f(\bar{\Gamma})$  для алгоритмів  $D, A_1, A_3, A_6$  зображується на графіках (рис. 7 і 8).

Менше значення потрібного обсягу оперативної пам'яті для алгоритму  $D$  пояснюється тим, що у разі успішного зондування по співвідношенням

$$\exists (i \in \{0, 1, \dots, m\}) \left( \sum_{j \in \Omega} \min(a_{ij}, 0) > b_i \quad \sum_{j \in N/\Omega} a_{ij} \quad x_j \right) \quad (1)$$

де  $N = (\overline{1, n})$  – множина індексів всіх змінних;

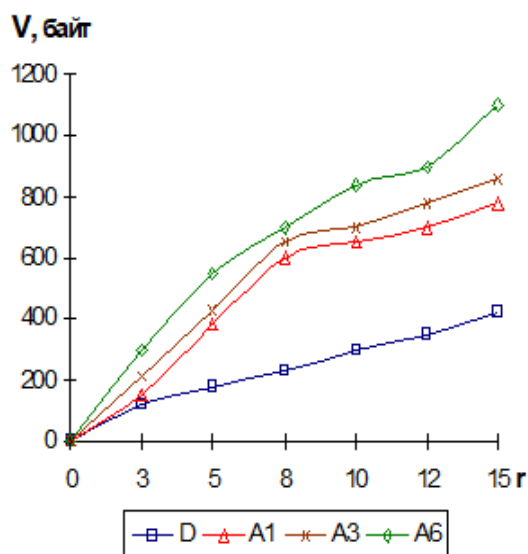
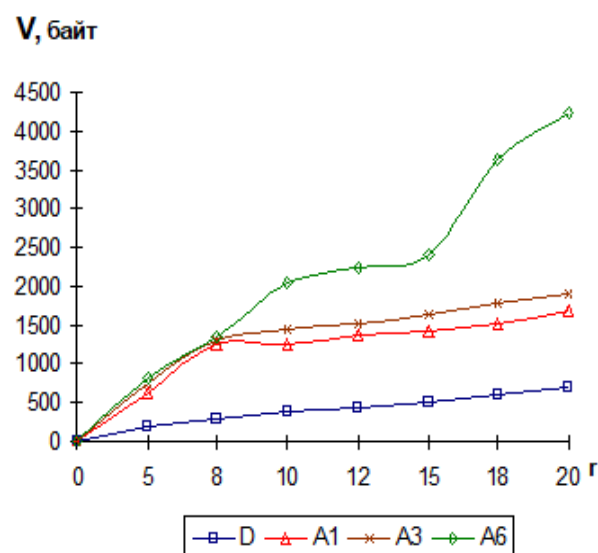
$\Omega$  – множина індексів вільних змінних;

$N/\Omega$  – множина індексів фіксованих змінних.

$$\forall (k \in \Omega) \left\{ \exists [i \in (\overline{0, m})] \left( \sum_{j \in \Omega} \min(a_{ij}, 0) + |a_{ik}| > b_i \quad \sum_{j \in N/\Omega} a_{ij} \quad x_j \right) \right\} \rightarrow \begin{cases} x_k = 0, \text{ при } a_{ik} > 0; \\ x_k = 1, \text{ при } a_{ik} < 0. \end{cases} \quad (2)$$

підмножин альтернатив, нове завдання, а отже новий запис в пам'яті, не утворюється, оскільки коректується список вільних вершин для вибраного завдання. Тільки утворення нового галуження, що відповідає кроку  $b$  алгоритму, вимагає витрат оперативної пам'яті. Для алгоритмів  $A_1, A_3, A_6$  максимальне число оброблюваних векторів знаходиться в другій зоні. Тому для зберігання потрібно більше пам'яті, ніж для першої зони. Максимального

значення витрати досягають в третій зоні, оскільки в цьому випадку відмінність за вагою функціонала усередині множин незначні, що приводить до неефективного відсіву векторів за стратегією  $L_3$

Рис. 7. Вплив  $\bar{r}$  на  $V$  при  $n=15$ Рис. 8. Вплив  $\bar{r}$  на  $V$  при  $n=15$ 

Залежності величини  $V$  від розміру вхідного параметра  $n$  також не перевищує  $c \cdot n^2$ . Результати такого дослідження показані на рис. 9 - 11, відповідних зонам 1 - 3.

Тимчасову складність алгоритмів  $A_1$  і  $A_3$  достатньо легко оцінити теоретично. Аналіз кожного вектора  $\vec{x}$  включає одну операцію складання і одну операцію порівняння, тобто дві елементарні операції. Максимальне число векторів у гіршому разі буде на другому ранзі і складає  $n^2/2$ . Число таких рангів в графі  $D\Delta$  дорівнює  $n$ . Отже, тимчасова складність алгоритму  $A_1$  складе  $O(2n \times n^2/2) = O(n^3)$ , зберігання векторів в  $\Omega$ -областях на всіх рангах буде потрібно пам'яті не більше, ніж  $O(n^3)$ .

Оцінка тимчасової складності алгоритму  $A_2$  співпадає з оцінкою для алгоритму  $A_1$ . Оцінимо тимчасову складність алгоритму  $A_3$ . Тимчасова складність алгоритму  $A_1$  складає  $O(n^3)$ . Однак в  $A_3$  кожна множина буде містити, після виділення коридору і фільтрації у гіршому разі, два шляхи, що



збільшить тимчасову складність алгоритму  $A_3$  в два рази і складе  $O(2n^3) \approx O(n^3)$ . Отже, і апаратні витрати на зберігання векторів також збільшаться в два рази і складуть  $O(2n^3) \approx O(n^3)$ .

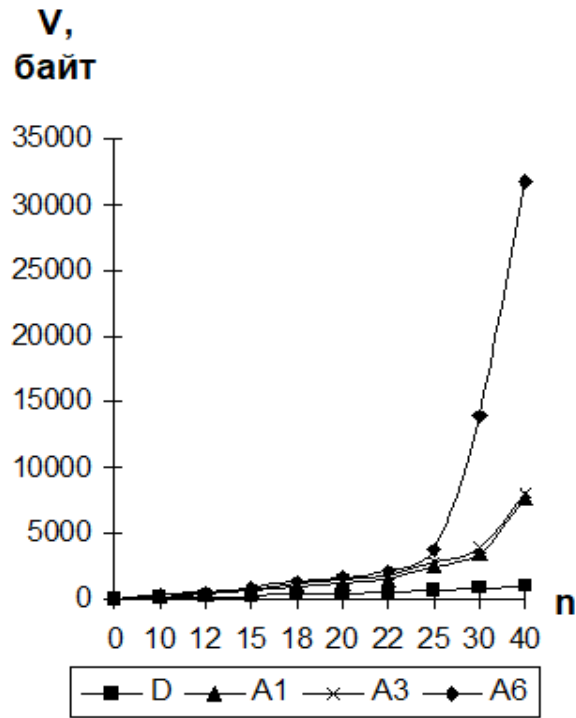
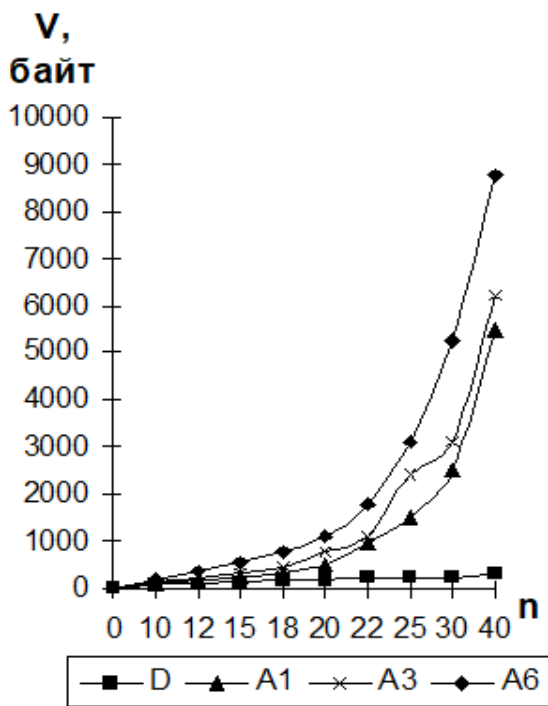


Рис. 9. Залежність  $V=f(n)$  для зони 1'    Рис. 10. Залежність  $V=f(n)$  для зони 2'

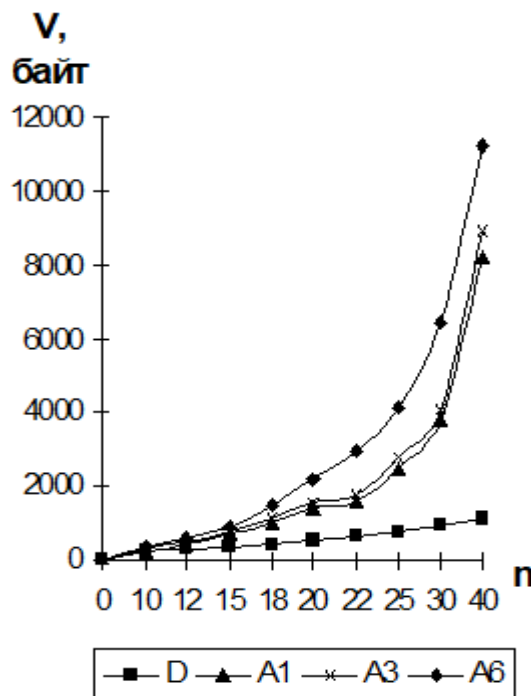


Рис. 11. Залежність  $V = f(n)$  для зони 3

Однією з важливих характеристик наближених алгоритмів служить відносна погрішність  $\Delta f$ , яку він дає у разі рішення задачі. З даних алгоритмів наближеними є  $A_1$  і  $A_3$ . Відносна погрішність залежить від зони, в яку потрапить оптимальне рішення. Проте, отримання неточних рішень алгоритмами  $A_1$  і  $A_3$  для різних зон в середньому виявилось проблематичною справою.

Так, для зони 1 експоненціальне зростання векторів в графі  $D\Delta$  натикається на обмеження і тому в множинах оптимальні шляхи будуть домінуючими по вагах функціонала. У зоні 3 оптимальний шлях проходитиме по вищих вершинах графа  $D\Delta$ , що не дозволить його відсікти. Тому на рис.12 показано зміну погрішності  $f$  і коефіцієнта  $K_n$  залежно від розмірності  $n$  тільки для другої зони.

Як видно з рис. 12 для обох алгоритмів  $A_1$  і  $A_3$  середня похибка  $f$  стабілізується близько 5% – для першого алгоритму і близько 2% – для другого при  $n > 25$ .

**Дослідження алгоритмів другої групи.** Оцінимо властивості алгоритмів другої групи  $A_4, A_5, A_7, A_8$  за вибраними показниками ефективності. Алгоритми  $A_4$  і  $A_5$  є наближеними, а  $A_7$  і  $A_8$  – точними. Так, як і для одновимірних завдань, значення вказаних показників залежить від  $\bar{\Gamma}$ , що підтверджує графік на рис. 14. Криві на цьому рисунку ідентичні за своєю формою відповідним показникам одновимірних алгоритмів. Тому, проведене ділення по зонах правомірно і для багатовимірних алгоритмів; залежності вибраних показників від  $n$  для кожної зони також співпадають з відповідними залежностями для одновимірних алгоритмів.

У зв'язку з необхідністю виконання  $m$  разів перевірки умови  $\sum_{j=1}^n a_{ij} \times x_j \leq b_i$ , часова складність алгоритму  $A_4$  складе  $O(2mn^3) \approx O(mn^3)$ . Часова складність алгоритму  $A_5$  із-за вибраної стратегії  $L_{11}$  в  $m$  раз більше, ніж часова складність алгоритму  $A_4$  і складає  $O(m \times n^3 m) = O(m^2 n^3)$ , оскільки з кожної множини,

згідно зі співвідношенням 
$$\mu_{sp}^{r=r+1} = \min_{a_{ij}} \{ \mu_{sj}^r \cup (j, p) \}; \quad p = \overline{(r+1, n)}; \quad j = \overline{(r, n)}, \quad i = 1$$

необхідно вибирати, у гіршому разі, на  $m$  векторів більше. Отже, і величина апаратних витрат пам'яті, що використовується для зберігання шляхів, також в  $m$  раз збільшиться і дорівнюватиме  $O(m \times n^3)$ .

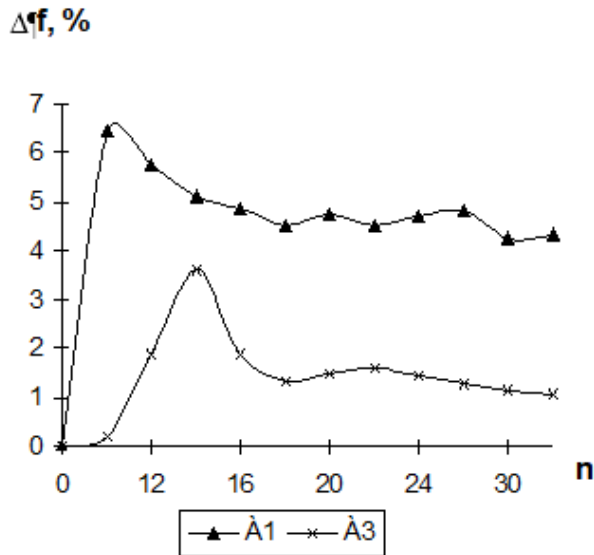


Рис. 12. Залежність  $\Delta f$  від  $n$  для зони 2

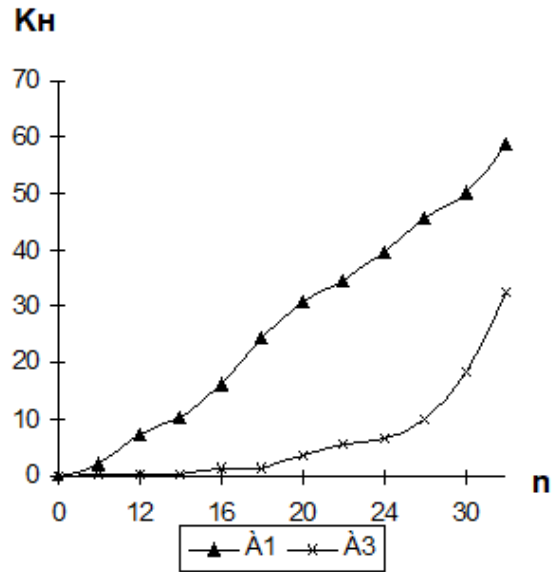


Рис. 13. Залежність  $K_n$  від  $n$  для зони 2

При  $m = 5$  проведено дослідження погрішності алгоритмів  $A_4$  і  $A_5$ . Залежність  $\Delta f$  і  $K_n$  від  $n$  і  $m$  для 2-ї зони подані у вигляді графіків на рис. 14.

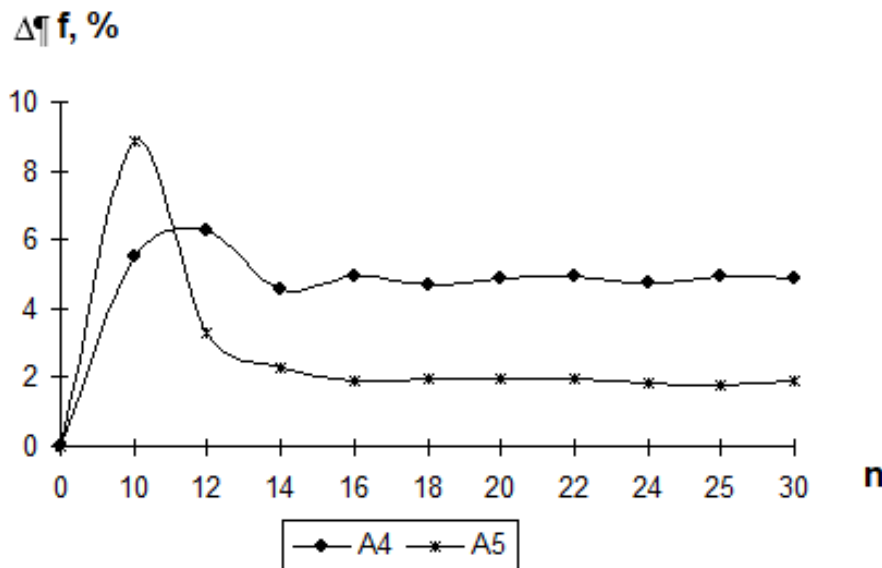


Рис. 14. Залежність  $\Delta f$  від  $n$  для зони 2

Якщо порівняти рис. 11 і рис. 14, то видно, що відносна погрішність  $\Delta f$  не залежить від  $m$ , а залежить тільки від вибраного алгоритму. Таким чином, можна стверджувати, що алгоритм  $A_4$  володіє погрішністю – 5%, а алгоритм  $A_5$  – 2%.

Для точних алгоритмів  $A_7$  і  $A_8$  використовується підхід з виділенням  $m$ -мірного коридору. При цьому одержані алгоритми мають експоненціальну складність, яка полягає в експоненціальному числі векторів, що можуть потрапити в цей коридор при побудові шляхів, починаючи з рангу  $r = 1$ . Тому використання цих алгоритмів є найбільш ефективним при вже знайденому допустимому рішенні яким-небудь з наближених алгоритмів, а оцінка значень показників ефективності буде проведена при побудові багатоетапних точних алгоритмів.

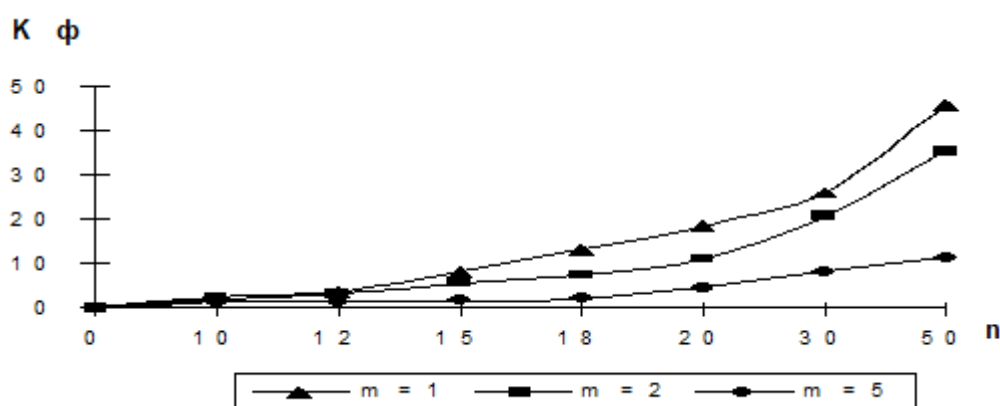


Рис. 15. Залежність  $K_\phi$  від  $n$  для зони 2

Порівняємо алгоритми  $A_7$  і  $A_8$  між собою. Як основний критерій такого порівняння вибраний коефіцієнт фільтрації  $K_\phi$ . Для якнайгіршого розподілу (зона 2) залежність  $K_\phi$  від  $n$  зображені на рис. 15. Як видно з рис. 15 із збільшенням  $m$  при  $n = \text{const}$   $K_\phi$  прагне до 1. Це пояснюється тим, що зменшується вірогідність існування шляху у вищих вершинах графа ДЛ на поточному ранзі  $r$ , що має менше значення за вагою всіх обмежень і більше значення за вагою функціонала.

**Дослідження алгоритмів третьої групи.** Проведемо експериментальне дослідження алгоритмів третьої групи:  $A_9$ ,  $A_{10}$ ,  $A_{11}$ ,  $A_{12}$ . Перші два відносяться до одновимірних багато-етапних алгоритмів. Другі - до багатовимірних.

Експеримент полягав в рішенні 1000 тестових завдань алгоритмом із заданою розмірністю вхідних параметрів  $n$  і  $m$ . Діапазон змін величин в обмеженнях підбирався так, щоб ранг оптимального рішення, що одержується на останньому етапі алгоритму (він є точним), належав другій зоні, тобто  $\bar{r} = [n/3 \div 2n/3]$ .

Результати дослідження показали (рис. 16, 17), істотний вигравш алгоритмів  $A_9$  і  $A_{10}$  в порівнянні з алгоритмом  $A_7$ . Тому, використання значень одержаного допустимого рішення на попередніх етапах дозволяє зменшити число оброблюваних векторів алгоритмом. Аналогічні результати одержані і для алгоритмів  $A_{11}$  і  $A_{12}$ .

Такий самий експеримент був проведений для алгоритмів  $A_{11}$  і  $A_{12}$ . Тільки тепер змінювався вхідний параметр  $m$ , який задає кількість обмежень. Результати цього експерименту для якнайгіршого випадку (зони 2) наведені на рис. 18 і 19 відповідно.

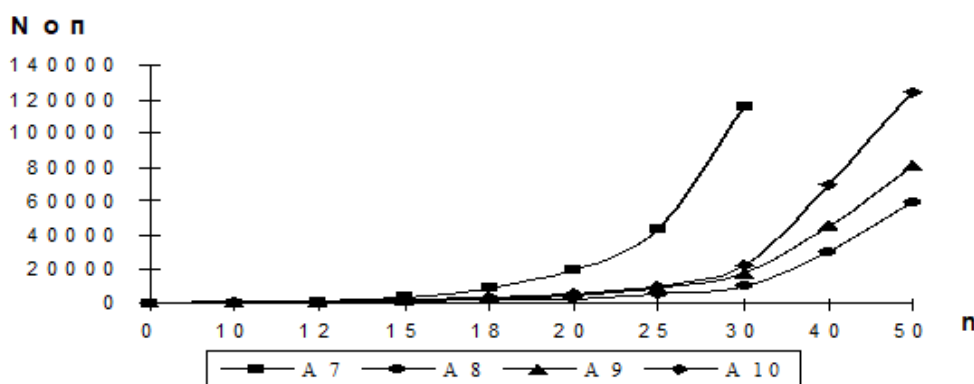


Рис. 16. Залежність  $N_{op}$  від  $n$  для зони 2

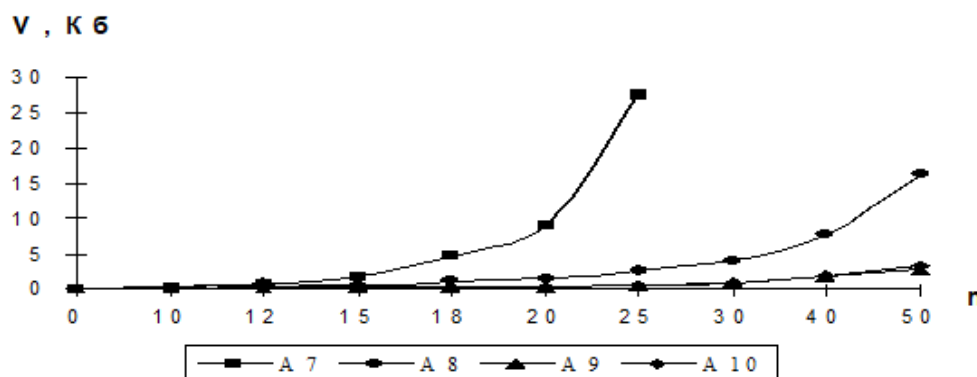


Рис. 17. Залежність  $V, K_b$  від  $n$  для зони 2

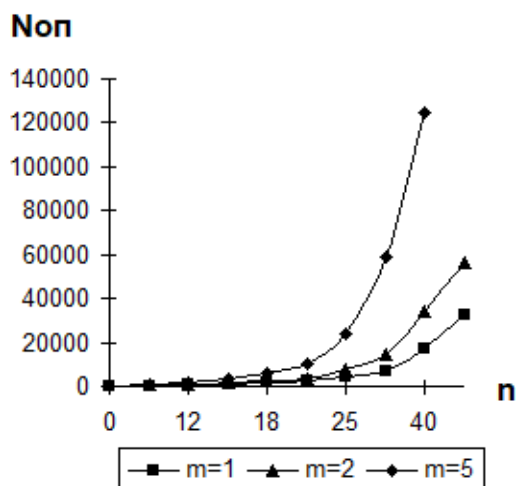


Рис. 18. Залежність  $N_{opt}$  від  $n$  та  $m$  для алгоритма  $A_{11}$

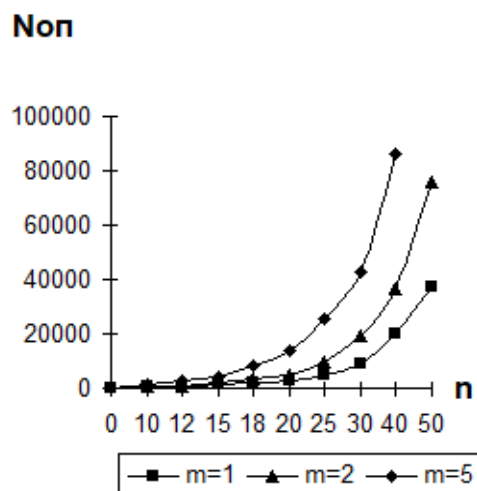


Рис. 19. Залежність  $N_{opt}$  від  $n$  та  $m$  для алгоритма  $A_{12}$

Таким чином, в роботі експериментально досліджені властивості розроблених алгоритмів і дана оцінка показників ефективності для них.

#### Список джерел:

1. Пономаренко В.С., Голубничий Д.Ю., Третяк В.Ф. Цілочисельне програмування в економіці. – Харків: Вид. ХНУ, 2005. – 204 с.
2. Третяк В.Ф., Кужель І.Є, Приходько В.М., Використання технології реплікації у системі управління розподіленими базами даних // Збірник наукових праць Харківського університету Повітряних Сил. – 2010. – № 2(24). – С. 109-114.
3. Третяк В.Ф., Місюра О.М., Більчук В.М. Метод оптимізації структури розподіленої бази даних у вузлах інфокомунікаційної мережі хмарного середовища // Наука і техніка Повітряних Сил Збройних Сил України. – 2017. – № 1. – С. 92-96.
4. Третяк, В., Голубничий, Д., Коломійцев, О., Мегельбей, Г., Возний, О., & Філіпенков, О. (2020). Математична модель рангового підходу. Збірник наукових праць ЛОГОС, 116-122. <https://doi.org/10.36074/25.12.2020.v1.40>
5. Коломійцев, О., Голубничий, Д., Коц, Г., Третяк, В., Євстрат, Д., & Лисиця, А. (2020). Задачі дискретної оптимізації та їх постановка для розміщення засобів захисту в розподіленій системі. Збірник наукових праць ЛОГОС, 36-41. <https://doi.org/10.36074/20.11.2020.v5.1>
6. Голубничий, Д., Третяк, В., Запара, Д., Деменко, М., Новіченко, С., Доска, О., & Савельєв, А. (2021). Функціональна модель управління системою інформаційної безпеки. ГРААЛЬ НАУКИ, (2-3), 175-186. <https://doi.org/10.36074/grail-of-science.02.04.2021.035>