

МЕТОДИ ЕВОЛЮЦІЙНОЇ БАГАТОКРИТЕРІАЛЬНОЇ ОПТИМІЗАЦІЇ

О. В. Тимченко¹, Б. М. Гавриш²

¹Українська академія друкарства,
вул. Під Голоском, 19, Львів, 79020, Україна

²Національний університет «Львівська політехніка»,
вул. С. Бандери, 12, Львів, 79013, Україна

Еволюційна багатокритеріальна оптимізація займається пошуком оптимальних рішень, які задовольняють множини цільових функцій. Основні поняття включають домінуючі і недомінуючі (паретооптимальні) розв'язки. Домінуючим розв'язок x вважається, коли існує інший допустимий розв'язок y , не гірший за x за всіма цільовими функціями.

Сучасні методи у цій галузі діляться на традиційні і еволюційні. До першої групи відносять метод зважених цілей і метод обмежень. Еволюційні алгоритми представлені різними моделями, серед яких VEGA (Генетичний алгоритм з векторною оцінкою), SPEA (Еволюційний алгоритм Парето за силою), а також інші методи та алгоритми. Еволюційні алгоритми відзначаються гнучкістю у застосуванні до складних задач. Вони імітують природний добір, забезпечуючи пошук множини розв'язків одночасно. Це дає змогу ефективно працювати з неперервними і дискретними задачами.

Вирішення реальних багатоцільових проблем оптимізації за допомогою багатоцільових алгоритмів оптимізації стає складним, коли кількість цілей велика, оскільки типи алгоритмів, які зазвичай використовуються для розв'язання цих проблем, базуються на концепції недомінування, яка перестає працювати як кількість цілей зростає. Ця проблема відома як прокляття розмірності. Водночас наявність багатьох цілей, характерних для практичних оптимізаційних задач, дуже ускладнює вибір рішення задачі. У літературі використовуються різні підходи для зменшення кількості цілей, необхідних для оптимізації.

Ключові слова: багатокритеріальна оптимізація, еволюційні алгоритми, паретооптимальні рішення, прокляття розмірності, метод зважених цілей.

Постановка проблеми. Задача багатокритеріальної оптимізації виникає тоді, коли при прийнятті рішення необхідно одночасно враховувати декілька цільових функцій. Можливі розв'язки задачі оптимізації поділяються на домінуючі та недомінуючі (паретооптимальні) розв'язки.

Аналіз останніх досліджень та публікацій. У дослідження різних підходів до задач багатокритеріальної оптимізації обговорюються методи отримання оптимальних за Парето рішень шляхом зведення багатокритеріальних задач до задач

скалярної оптимізації зі спеціальними цільовими функціями. О. С. Ковалишин (2018) аналізує існуючі методи розв'язання задач багатокритеріальної оптимізації в контексті планування, підкреслюючи переваги генетичних алгоритмів для оптимізації графіків реабілітаційної терапії. А. А. Іщук (2019) наголошує на необхідності комп'ютерних рішень для задач багатокритеріальної оптимізації, порівнюючи два методи та відзначаючи ефективність, отриману за допомогою інформаційно-комунікаційних технологій. Куперман (2011) представляє критичний огляд існуючих методів багатокритеріальної оптимізації виробничих планів компанії та пропонує нові підходи, які поєднують різні процедури для використання їх позитивних якостей. Ці статті разом демонструють різноманітні застосування та постійний розвиток методів багатокритеріальної оптимізації в різних сферах, включаючи екологічну безпеку, охорону здоров'я та промислове виробництво.

Мета статті. Основна мета роботи полягає у дослідженні і порівнянні традиційних методів багатокритеріальної оптимізації: алгоритмів VEGA і SPEA (метод зважених цілей, метод обмежень) і еволюційних алгоритмів (VEGA, SPEA та інші), що вирізняються гнучкістю та здатністю одночасно знаходити множину рішень, що важливо для складних задач. Описано переваги й недоліки алгоритмів, зокрема ефективність та труднощі їх реалізації.

Виклад основного матеріалу дослідження. Для завдань максимізації набору з k функцій мети

$$f(x) = (f_1(x), f_2(x), \dots, f_k(x))$$

розв'язок x називається домінуючим, якщо існує допустимий розв'язок y , не гірший за x , тобто для будь-якої цільової функції f_i

$$f_i(x) \leq f_i(y) \quad (i = 1, \dots, k)$$

інакше: x – розв'язок недомінуючий (паретооптимальний).

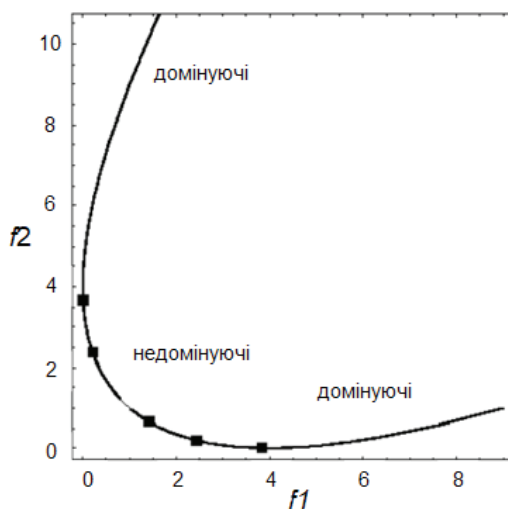


Рис. 1. Домінуючі та недомінуючі розв'язки завдань мінімізації.

Для завдань мінімізації множини k функцій мети

$$f(x) = (f_1(x), f_2(x), \dots, f_k(x))$$

розв'язок x називається домінуючим, якщо існує допустимий розв'язок y , не гірший за x , тобто для будь-якої цільової функції f_i

$$f_i(y) \leq f_i(x) \quad (i = 1, \dots, k)$$

інакше: x – розв'язок недомінуючий (паретооптимальний).

Обрані методи.

Традиційні підходи:

- Метод зважених цілей (Weighting Method)
- Метод обмежень (Constraint Method)

Еволюційні алгоритми:

- VEGA: Генетичний алгоритм з векторною оцінкою (Vector Evaluated Genetic Algorithm (Schaffer 1985)).
- HLGA: Генетичний алгоритм на основі зважування Хаела та Ліна (Hajela's and Lin's Weighting-based Genetic Algorithm (1992)).
- FFGA: Багатоцільовий генетичний алгоритм Фонсеки та Флемінга (Fonseca's and Fleming's Multiobjective Genetic Algorithm (1993)).
- NPGA: Генетичний алгоритм Нічеда Парето (The Niche Pareto Genetic Algorithm (Horn, Nafpliotis, Goldberg 1994)).
- NSGA: Генетичний алгоритм сортування без домінування (The Nondominated Sorting Genetic Algorithm (Srinivas, Deb 1994)).
- SPEA: Еволюційний алгоритм Парето за силою (The Strength Pareto Evolutionary Algorithm (Zitzler, Thiele 1999)).

У цій статті наведено короткий опис методу зважених цілей та алгоритмів VEGA і SPEA. Детальний опис усіх згаданих методів та алгоритмів можна знайти в [1].

1. Метод зважених цілей

Метод зважених цілей полягає у зведенні двовимірної задачі до одновимірної, тобто об'єднанні окремих функцій цілі f в одну функцію цілі F :

$$F(x) = \sum_{i=1}^k w_i f_i(x),$$

де:

k – кількість цільових функцій;

x – вектор розв'язків

w_i – ваги такі, що:

$$w_i \in [0, 1] \quad \text{та} \quad \sum_{i=1}^k w_i = 1$$

(різні вектори ваги дають різні паретооптимальні розв'язки)

Отримана в результаті функція F оптимізується за допомогою стандартних методів оптимізації єдиної функції цілі. Основним недоліком цього методу є проблема вибору відповідних коефіцієнтів ваги для окремих критеріїв (що негативно впливає на якість отриманих рішень).

2. Генетичний алгоритм з векторною оцінкою

Ідея, використана в алгоритмі VEGA, полягає в тому, щоб розділити популяцію на k підпопуляцій рівної чисельності (k - кількість цілей). Відбір у кожній підпопуляції здійснюється незалежно (кожна підпопуляція відповідає за свій критерій), але схрещування виходить за межі підпопуляцій (охоплюють цілу популяцію).

Основною перевагою алгоритму є простота реалізації. Основним недоліком є тенденція до пропуску проміжних рішень (хороших за кожним критерієм, але не найкращих за жодним з них окремо).

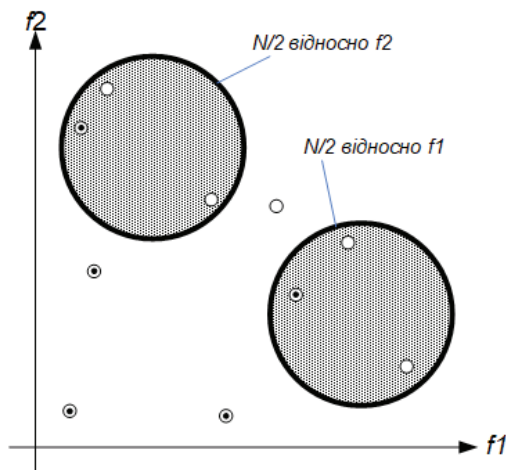


Рис. 2. Поділ популяції на підпопуляції для двох критеріїв.

Використані позначення

t – номер покоління

P_t – чисельність популяції у моєму поколінні

P' – тимчасова популяція (mating pool)

k – кількість критеріїв

Алгоритм VEGA

Вхідні параметри:

N – розмір популяції

T – максимальна кількість поколінь

p_c – ймовірність спарювання

p_m – ймовірність мутації

Результат:

A – множина недомінуючих розв'язків

Крок 1:

Ініціалізація: Створити початкову популяцію P_0 .

Нехай $P_0 = \emptyset$ і $t = 0$. Для $i = 1, \dots, N$ виконати наступні дії:

Вибрати індивіда i .

Додати індивіда i до множини P_0 .

Крок 2:

Визначення відповідності та підбір: $P'_t = \emptyset$. Для $i = 1, \dots, k$ виконати наступні дії:

Для кожного індивіда $i \in P_t$ обчислити його відповідність на основі функції цілі f_i .

Для $j = 1, \dots, N / k$ вибрати індивіда i з P_t і додати його до P' .

Крок 3:

Скорочення: Нехай $P'' = \emptyset$. Для $i = 1, \dots, N / 2$ виконати наступні дії:

Вибрати двох індивідів $i, j \in P'$ і вилучити їх з P' .

Схрестити індивідів i та j ; результат: індивіди k і l .

Додати k, l до P'' з ймовірністю p_c (в іншому випадку до P'' додати індивідів i, j).

Крок 4:

Мутація: Нехай $P''' = \emptyset$. Для кожного індивіда $i \in P''$, виконати наступні дії:

а) Змутувати індивіда i з ймовірністю p_m . Результат: індивід j .

б) Додати індивіда j до множини P''' .

Крок 5:

Висновок: Нехай $P_{t+1} = P'''$ і $t = t + 1$. Якщо $t \geq T$, то закінчити (результат: $A =$ недомінуючі розв'язки з популяції P_t), в іншому випадку повернутися до кроку 2.

3. Еволюційний алгоритм Парето за силою

Характерною особливістю алгоритму SPEA є те, що індивіди, які представляють не домінуючі рішення (серед розглянутих до цього часу рішень), зберігаються в окремій множині (так званій зовнішній множині).

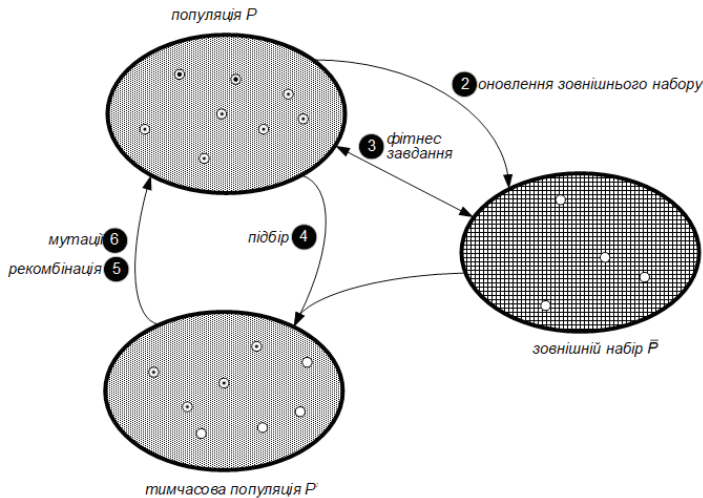


Рис. 3. Схематична модель роботи алгоритму SPEA.

Більше того, адаптаційна цінність індивіда, що належить до популяції, залежить лише від того, наскільки він домінує над індивідами із зовнішньої множини: чи домінують індивіди з популяції один над одним, не має значення. У відборі беруть участь усі індивіди із зовнішньої множини. Розмір зовнішньої множини зменшується до необхідної кількості шляхом кластеризації, без втрати інформації про хід паретооптимального фронту.

Беззаперечною перевагою алгоритму SPEA є те, що, на відміну від алгоритму VEGA, алгоритм не пропускає проміжні рішення, забезпечуючи, таким чином, хороший огляд паретооптимального фронту. Основним недоліком є висока обчислювальна складність алгоритму (зокрема, процедура визначення відповідності індивіда є трудомісткою – необхідна повна перевірка зовнішньої множини).

Використані позначення

t – номер покоління

P_t – чисельність популяції у t -му поколінні

\bar{P}_t – зовнішня множина (external set)

\bar{P}' – тимчасова зовнішня множина

P' – тимчасова популяція (mating pool)

Алгоритм SPEA

Вхідні параметри:

N – розмір популяції

\bar{N} – максимальний розмір зовнішньої множини

T – максимальна кількість поколінь

p_c – ймовірність спарювання

p_m – ймовірність мутації

Результат:

A – множина недомінуючих розв'язків

Крок 1:

Ініціалізація: Створити початкову популяцію P_0 (див. крок 1 алгоритму VEGA), а також пусту зовнішню множину $\bar{P}_0 = \emptyset$. Нехай $t = 0$.

Крок 2:

Комплектація зовнішньої множини: Нехай $\bar{P}' = \bar{P}_t$.

Скопіювати до \bar{P}' індивіди з популяції P_t , недомінуючі над іншими індивідами з популяції P_t .

Видалити з \bar{P}' індивіди домінуючі над іншими індивідами з \bar{P}' .

Зменшити кількість множини \bar{P}' до \bar{N} шляхом кластеризації; результат \bar{P}_{t+1} .

Крок 3:

Визначення відповідності: обчислити значення відповідності F індивідів в P_t і \bar{P}_t за допомогою алгоритму описаного далі.

Крок 4:

Підбір: Нехай $P' = \emptyset$. Для $i = 1, \dots, k$ виконати наступні дії:

Вибрати випадковим чином двох індивідів $i, j \in P_t + \bar{P}_t$.

Якщо $F(i) < F(j)$, то $P' = P' + \{i\}$, в іншому випадку $P' = P' + \{j\}$ (значення пристосування тут мінімізоване).

Крок 5:

Скорочення: див. крок 3 алгоритму VEGA (результат P'').

Крок 6:

Мутація: див. крок 4 алгоритму VEGA (результат P''').

Крок 7:

Висновок: Нехай $P_{t+1} = P'''$ і $t = t + 1$. Якщо $t \geq T$, то закінчити (результат: $A =$ недомінуючі розв'язки з популяції P_t), в іншому випадку повернутися до кроку 2.

Визначення відповідності в алгоритмі SPEA

Кожному індивідууму, тобто $i \in \bar{P}_t$ (що належить до зовнішньої множини), присвоюється порядкове значення $S(i) \in [0,1)$, яке називається силою. Сила індивіда i пропорційна кількості індивідів $j \in P_t$ (що належать до популяції), які представляють розв’язки домінуючі над розв’язком представленим індивідом i :

$$S(i) = \frac{n}{N + 1}$$

де:

$S(i)$ – сила індивіда i

n – кількість індивідів в популяції домінуючих над індивідом i

N – кількість популяції.

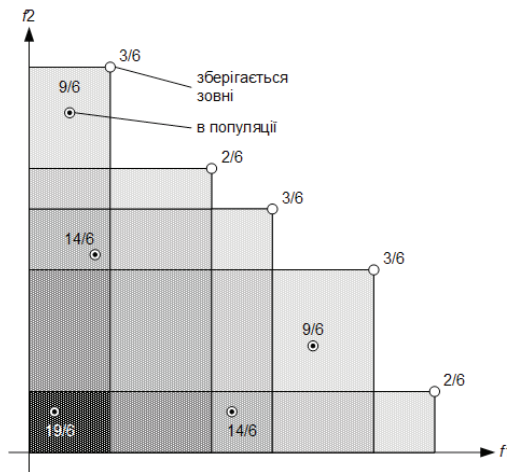


Рис. 4. Приклад значень адаптації для задачі максимізації.

Значення пристосування індивіда дорівнює його силі:

$$F(i) = S(i).$$

Пристосування індивіда $j \in P_t$ (який належить до популяції) розраховується як збільшена на 1 сума сил всіх індивідів $i \in \bar{P}_t$ (які належать до зовнішньої множини), що репрезентують розв’язки домінуючі над розв’язком репрезентованим індивідом j . Додавання 1 призначене для того, щоб індивіди, які належать до зовнішньої множини \bar{P}_t будуть мали кращу адаптацію, ніж особи з внутрішньої популяції P_t .

Скорочення зовнішнього набору шляхом кластеризації

У випадку, коли розмір зовнішньої множини перевищує допустиме значення, необхідно його скоротити. Щоб не втратити інформації про перебіг пареооптимального фронту, вибір елементів до скорочення не може бути випадковим. Перебіг наступних етапів скорочення проілюстровано і описано нижче.

Наступними кроками у зменшенні зовнішньої множини є:

- групування особин у N кластерів (в межах одного кластера розміщуються індивіди, що представляють рішення, близькі один до одного);

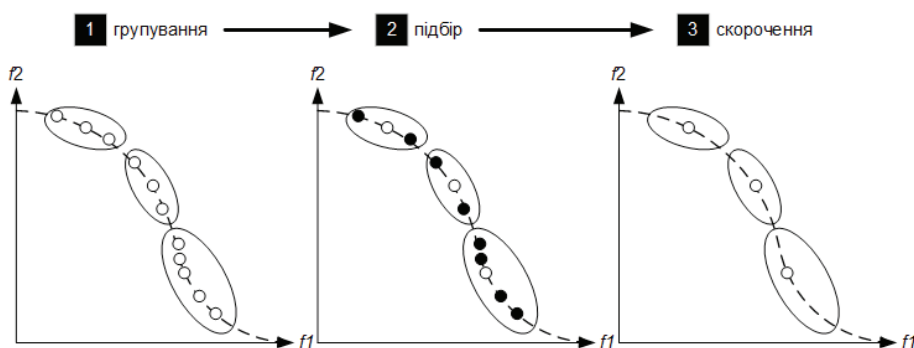


Рис. 5. Скорочення зовнішньої множини шляхом кластеризації

- вибір одного представника для даного кластера (зазвичай це індивід, що знаходиться в центральній точці кластера):
- видалення всіх індивідів, окрім попередньо обраних представників кожного кластера.

Висновки. Основна тема статті — еволюційна багатокритеріальна оптимізація, яка займається пошуком оптимальних рішень для множини цільових функцій. Представлено традиційні методи (метод зважених цілей, метод обмежень) і еволюційні алгоритми (VEGA, SPEA та інші). Еволюційні алгоритми вирізняються гнучкістю та здатністю одночасно знаходити множину рішень, що важливо для складних задач. Визначено проблему зростання обчислювальної складності при збільшенні кількості цілей, відому як «прокляття розмірності». Описано переваги й недоліки алгоритмів, зокрема ефективність та труднощі їх реалізації.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Ковалишин О.С. (2018). Аналіз методів оптимізації розкладів у контексті відновлювальної терапії. *Scientific Bulletin of UNFU*.
2. Іщук, А.А. (2019). Розв'язування багатокритеріальних задач оптимізації за допомогою комп'ютера. *Науковий часопис НПУ імені М.П. Драгоманова. Серія 2. Комп'ютерно-орієнтовані системи навчання*.
3. Куперман, В.В. (2011). Методи багатокритеріальної оптимізації виробничої програми підприємства.
4. Y. Liang, H. Huang, Z. Cai and Z. Hao, «Multiobjective Evolutionary Optimization Based on Fuzzy Multicriteria Evaluation and Decomposition for Image Matting,» in *IEEE Transactions on Fuzzy Systems*, vol. 27, no. 5, pp. 1100-1111, May 2019, doi: 10.1109/TFUZZ.2019.2896533.
5. T. -Y. Huang and W. -Y. Chiu, «Systematic framework for solving real-world problems with multiple objectives,» 2016 IEEE 5th Global Conference on Consumer Electronics, Kyoto, Japan, 2016, pp. 1-2, doi: 10.1109/GCCE.2016.7800365.
6. K. Zhang, G. G. Yen and Z. He, «Evolutionary Algorithm for Knee-Based Multiple Criteria Decision Making,» in *IEEE Transactions on Cybernetics*, vol. 51, no. 2, pp. 722-735, Feb. 2021, doi: 10.1109/TCYB.2019.2955573.

7. C. L. d. V. Lopes, F. V. C. Martins, E. F. Wanner and K. Deb, «Analyzing Dominance Move (MIP-DoM) Indicator for Multiobjective and Many-Objective Optimization,» in IEEE Transactions on Evolutionary Computation, vol. 26, no. 3, pp. 476-489, June 2022, doi: 10.1109/TEVC.2021.3096669.
8. M. Aggarwal, «Learning of a Decision-Maker's Preference Zone With an Evolutionary Approach,» in IEEE Transactions on Neural Networks and Learning Systems, vol. 30, no. 3, pp. 670-682, March 2019, doi: 10.1109/TNNLS.2018.2847412.
9. Z. He, G. G. Yen and J. Ding, «Knee-Based Decision Making and Visualization in Many-Objective Optimization,» in IEEE Transactions on Evolutionary Computation, vol. 25, no. 2, pp. 292-306, April 2021, doi: 10.1109/TEVC.2020.3027620.
10. L. Zhao, W. Tan, L. Xu, N. Xie and L. Huang, «Crowd-Based Cooperative Task Allocation via Multicriteria Optimization and Decision-Making,» in IEEE Systems Journal, vol. 14, no. 3, pp. 3904-3915, Sept. 2020, doi: 10.1109/JSYST.2020.2966646.

REFERENCES

1. Kovalyshyn O.S. (2018). Analiz metodiv optymizatsiyi rozkladiv u konteksti vidnovlyuvalnoyi terapiyi. Scientific Bulletin of UNFU.
2. Ishchuk, A.A. (2019). Rozvyazuvannya bahatokryterialnykh zadach optymizatsiyi za dopomohoyu kompyutera. Naukovyy chasopys NPU imeni M.P. Drahomanova. Seriya 2. Kompyuterno-oriyentovani systemy navchannya.
3. Kuperman, V.V. (2011). Metody bahatokryterialnoyi optymizatsiyi vyrobnychoyi prohramy pidpryemstva.
4. Y. Liang, H. Huang, Z. Cai and Z. Hao, «Multiobjective Evolutionary Optimization Based on Fuzzy Multicriteria Evaluation and Decomposition for Image Matting,» in IEEE Transactions on Fuzzy Systems, vol. 27, no. 5, pp. 1100-1111, May 2019, doi: 10.1109/TFUZZ.2019.2896533.
5. T. -Y. Huang and W. -Y. Chiu, «Systematic framework for solving real-world problems with multiple objectives,» 2016 IEEE 5th Global Conference on Consumer Electronics, Kyoto, Japan, 2016, pp. 1-2, doi: 10.1109/GCCE.2016.7800365.
6. K. Zhang, G. G. Yen and Z. He, «Evolutionary Algorithm for Knee-Based Multiple Criteria Decision Making,» in IEEE Transactions on Cybernetics, vol. 51, no. 2, pp. 722-735, Feb. 2021, doi: 10.1109/TCYB.2019.2955573.
7. C. L. d. V. Lopes, F. V. C. Martins, E. F. Wanner and K. Deb, «Analyzing Dominance Move (MIP-DoM) Indicator for Multiobjective and Many-Objective Optimization,» in IEEE Transactions on Evolutionary Computation, vol. 26, no. 3, pp. 476-489, June 2022, doi: 10.1109/TEVC.2021.3096669.
8. M. Aggarwal, «Learning of a Decision-Maker's Preference Zone With an Evolutionary Approach,» in IEEE Transactions on Neural Networks and Learning Systems, vol. 30, no. 3, pp. 670-682, March 2019, doi: 10.1109/TNNLS.2018.2847412.
9. Z. He, G. G. Yen and J. Ding, «Knee-Based Decision Making and Visualization in Many-Objective Optimization,» in IEEE Transactions on Evolutionary Computation, vol. 25, no. 2, pp. 292-306, April 2021, doi: 10.1109/TEVC.2020.3027620.

10. L. Zhao, W. Tan, L. Xu, N. Xie and L. Huang, «Crowd-Based Cooperative Task Allocation via Multicriteria Optimization and Decision-Making,» in IEEE Systems Journal, vol. 14, no. 3, pp. 3904-3915, Sept. 2020, doi: 10.1109/JSYST.2020.2966646.

doi: 10.32403/1998-6912-2024-2-69-11-20

METHODS OF EVOLUTIONARY MULTI-CRITERIA OPTIMIZATION

O. V. Tymchenko¹, B. M. Havrysh²

¹Ukrainian Academy of Printing,
19, Pid Holoskom, St., Lviv, 79020, Ukraine

²Lviv Politechnic National University,
12, Bandera St., Lviv, 79013, Ukraine
o_tymch@ukr.net, dana.havrysh@gmail.com

Evolutionary multicriteria optimization is concerned with finding optimal solutions that satisfy a set of objective functions. The main concepts include dominant and non-dominant (pareto-optimal) solutions. A solution x is considered dominant when there is another admissible solution y that is not worse than x for all objective functions.

Modern methods in this area are divided into traditional and evolutionary. The first group includes the weighted objective method and the constraint method. Evolutionary algorithms are represented by various models, including VEGA (Genetic Algorithm with Vector Evaluation), SPEA (Strength Pareto Evolutionary Algorithm), and other methods and algorithms. Evolutionary algorithms are characterized by flexibility in applying to complex problems. They imitate natural selection, providing a search for a set of solutions simultaneously. This makes it possible to work effectively with continuous and discrete problems.

Solving real-world multi-objective optimization problems using multi-objective optimization algorithms becomes difficult when the number of objectives is large, because the types of algorithms commonly used to solve these problems are based on the concept of non-dominance, which stops working as the number of objectives increases. This problem is known as the curse of dimensionality. At the same time, the presence of many objectives, which is typical for practical optimization problems, makes it very difficult to choose a solution. Various approaches are used in the literature to reduce the number of objectives required for optimization.

Keywords. multicriteria optimization, evolutionary algorithms, pareto-optimal solutions, curse of dimensionality, weighted objective method.

Стаття надійшла до редакції 02.07.2024.

Received 02.07.2024.