

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЧЕРКАСЬКИЙ ДЕРЖАВНИЙ ТЕХНОЛОГІЧНИЙ УНІВЕРСИТЕТ
ФАКУЛЬТЕТ ЕЛЕКТРОННИХ ТЕХНОЛОГІЙ, АВТОТРАНСПОРТУ
ТА МАШИНОБУДУВАННЯ

ВЕКТОРНА ОПТИМІЗАЦІЯ В MathCAD

Навчально-методичний посібник
для здобувачів галузі знань
G «Інженерія, виробництво та будівництво»
(15 «Автоматизація та приладобудування»
17 «Електроніка, автоматизація та електронні комунікації»)
усіх форм навчання

Черкаси 2025

УДК 004.942 (075.8)
В26

*Затверджено вченою радою ФЕТАМ,
протокол №1 від 18.02.2025 р.,
згідно з рішенням засідання кафедри
приладобудування, мехатроніки та
комп'ютеризованих технологій
протокол № 6 від 23.01.2025 р.*

Упорядники: Гальченко В.Я., *д.т.н., професор*, Трембовецька Р.В., *д.т.н., професор*, Тичков В.В., *к.т.н., доцент*
Рецензент: Федоров Є.Є., *д.т.н., професор*.

ВЕКТОРНА ОПТИМІЗАЦІЯ В MathCAD. Навчально-методичний посібник для здобувачів галузі знань G «Інженерія, виробництво та будівництво» (15 «Автоматизація та приладобудування», 17 «Електроніка, автоматизація та електронні комунікації») усіх форм навчання. [Електронний ресурс] / [Упоряд. В.Я. Гальченко, Р.В. Трембовецька, В.В. Тичков]; М-во освіти и науки України, Черкас. держ. технол. ун-т. - Черкаси: ЧДТУ, 2025. - 45 с.

У навчальному посібнику викладено коротко основні теоретичні положення математичних методів та докладний практичний матеріал щодо розв'язку задач векторної оптимізації у середовищі універсального математичного пакета MathCAD. Істотну увагу приділено комп'ютерній реалізації аналізованих методів, містяться комплекти завдань для самостійної роботи та велика кількість прикладів, що сприяють кращому розумінню та засвоєнню матеріалу.

Навчальне електронне видання
мережного використання

ВЕКТОРНА ОПТИМІЗАЦІЯ В MathCAD.

Навчально-методичний посібник
для здобувачів освітнього ступенів для здобувачів галузі знань
галузі знань G «Інженерія, виробництво та будівництво» (15
«Автоматизація та приладобудування», 17 «Електроніка, автоматизація та
електронні комунікації») усіх форм навчання

упорядники: Гальченко Володимир Якович
Трембовецька Руслана Володимирівна
Тичков Володимир Володимирович
В авторській редакції.

ЗМІСТ

ВСТУП	4
1. РОЗВ'ЯЗАННЯ НЕЛІНІЙНИХ ЗАДАЧ ВЕКТОРНОЇ ОПТИМІЗАЦІЇ	6
1.1 Поняття оптимальності в багатокритеріальних задачах і схеми компромісу	6
1.1.1 Постановка задачі багатокритеріальної оптимізації	6
1.1.2 Проблеми розв'язання задач багатокритеріальної оптимізації	7
1.1.3 Оптимальність за Парето	8
1.2 Скаляризація векторного критерію агрегуванням частинних критеріїв за допомогою згортки	14
ЗАВДАННЯ ДО САМОСТІЙНОЇ РОБОТИ	19
1.3 Апроксимація фронту Парето сітковим методом дослідження простору параметрів	23
ЗАВДАННЯ ДО САМОСТІЙНОЇ РОБОТИ	39
СПИСОК ЛІТЕРАТУРИ	40
ТЕРМІНИ ТА ВИЗНАЧЕННЯ	44

ВСТУП

Векторна оптимізація — це розділ математичної оптимізації, що займається задачами, у яких необхідно одночасно враховувати декілька критеріїв (цільових функцій). Такі задачі часто виникають у різних сферах науки, техніки, економіки, коли потрібно досягти компромісу між суперечливими цілями.

Особливості багатокритеріальної оптимізації

- Непорівнянність розв'язків: у багатокритеріальних задачах часто виникають ситуації, коли жодний розв'язок не є кращим за інший за всіма критеріями, що створює ефект непорівнянності.

- Нормалізація критеріїв: оскільки частинні критерії можуть мати різні одиниці вимірювання, їх приводять до єдиного масштабу для коректного порівняння.

- Вибір принципу оптимальності: необхідно визначити правило, за яким один розв'язок вважається кращим за інший, що є складною задачею через різноманітність цілей і критеріїв.

- Врахування пріоритетів: деякі критерії можуть мати більшу важливість, що враховується під час вибору оптимального розв'язку.

Методи розв'язання

- Скаляризація: зведення багатокритеріальної задачі до однокритеріальної внаслідок агрегування частинних критеріїв (наприклад, адитивна чи мультиплікативна згортка).

- Пошук Парето-оптимальних розв'язків: виділення множини розв'язків, які не домінуються іншими, тобто не можна покращити один критерій без погіршення інших.

- Апроксимація фронту Парето: використання чисельних методів та сіткових процедур для побудови наближення множини Парето-оптимальних розв'язків.

Векторна оптимізація є важливим інструментом для розв'язання реальних задач, де необхідно враховувати декілька суперечливих цілей. Вона широко використовується у:

- економіці (розподіл ресурсів, інвестування)
- інженерії (оптимізація конструкцій)
- логістиці (маршрутизація, розподіл вантажів)
- управлінні проектами (балансування вартості, часу та якості)

Посібник має на меті ознайомити читача з основними поняттями, методами та інструментами векторної оптимізації, а також сформулювати практичні навички розв'язання багатокритеріальних задач у різних сферах діяльності.

В посібнику детально розглядаються окремі теоретичні положення навчальних дисциплін «Оптимізація прийняття рішень у техніці», «Математичне моделювання процесів і систем та методи їх оптимізації» та формуються у здобувачів вміння та навички їх практичного застосування шляхом індивідуального виконання завдань.

Завдання розраховані для здобувачів галузі знань G, які опановують навчальні компоненти.

В системі дистанційної освіти ЧДТУ з дисциплін «Оптимізація прийняття рішень у техніці» <https://moodle.chdtu.edu.ua/course/view.php?id=53>, «Математичне моделювання процесів і систем та методи їх оптимізації» <http://fet.moodle.chdtu.edu.ua/course/view.php?id=18> є електронна версія цього посібника. Також наведені тестові завдання для самоконтролю до кожної теми відповідного змістового модулю. Після вивчення відповідного теоретичного лекційного матеріалу здобувач самостійно проходить тест-самоконтроль, отримує відповідну кількість балів, та аналізує свій рівень знань і може виконувати тестування необмежену кількість разів.

1. РОЗВ'ЯЗАННЯ НЕЛІНІЙНИХ ЗАДАЧ ВЕКТОРНОЇ ОПТИМІЗАЦІЇ

1.1 Поняття оптимальності в багатокритеріальних задачах і схеми компромісу

1.1.1 Постановка задачі багатокритеріальної оптимізації

Задачі оптимізації, у яких є не одна, а кілька цільових функцій (критеріїв), дістали назву *багатокритеріальних задач оптимізації*.

Критерії $F_i(\vec{X})$, $i = 1, 2, \dots, m$, утворюють векторний критерій $\vec{F}(\vec{X}) = (F_1, F_2, \dots, F_m)$. Тому часто також використовується термін "векторна оптимізація".

Нехай $\vec{X}_1 \in D$, тоді

$F_1(\vec{X}_1)$ - частинна оцінка розв'язок \vec{X}_1 за 1-м критерієм або критерієм F_1 ;

$F_2(\vec{X}_1)$ - частинна оцінка розв'язок \vec{X}_1 за 2-м критерієм або критерієм F_2 ;

$F_m(\vec{X}_1)$ - частинна оцінка розв'язок \vec{X}_1 за m -м критерієм або критерієм F_m ;

$\vec{F}(\vec{X}) = (F_1(\vec{X}_1), F_2(\vec{X}_1), \dots, F_m(\vec{X}_1))$ - векторна оцінка для розв'язку \vec{X}_1 .

Для пояснення сутності задач використовують геометричну інтерпретацію, пов'язану з уведенням n - мірного простору E^n - простору змінних і m - мірного простору E^m - критеріїв. Кожній точці простору E^n і E^m відповідають вектори \vec{X} та \vec{Y} .

Отже, допустимій області D , яку назвемо образом, можна поставити у відповідність деяку множину оцінок. Цю множину будемо позначати Y_D і її називатимемо *критеріальним простором* або *областю критеріїв* (областю оцінок), тобто $Y_D = \vec{F}(D)$ - прообраз множини D .

Сформулюємо задачу багатокритеріальної оптимізації. Вона має вигляд:

$$\min \vec{F}(\vec{X}) = \min (F_1(\vec{X}), F_2(\vec{X}), \dots, F_m(\vec{X})).$$

Наприклад, задача багатокритеріальної оптимізації може бути сформульована так: у квадраті $D = \{-1 \leq x_1 \leq 1, -1 \leq x_2 \leq 1\}$ задано два критерії $F_1(x_1, x_2) = 4x_1^2 + x_2^2$, $F_2(x_1, x_2) = (x_1 + 1)^2 + (x_2 - 1)^2$, які необхідно мінімізувати.

Теоретично можна уявити випадок, коли на множині D виявиться одна альтернатива (розв'язок), у якій усі m критеріїв набувають найменших значень; вона і є найкращою. Однак на практиці такі випадки майже не трапляються, і виникає питання розв'язання задачі вибору. Як правило, частинні критерії

суперечливі, тобто зменшення одного критерію веде до збільшення інших критеріїв і спостерігається конфлікт критеріїв.

1.1.2 Проблеми розв'язання задач багатокритеріальної оптимізації

Під час розроблення методів розв'язання задач векторної оптимізації доводиться розв'язувати специфічні проблеми. Розглянемо ці проблеми докладніше.

Непорівнянність розв'язків. Основна складність логічного аналізу багатокритеріальних задач полягає в тому, що в них, на відміну від однокритеріальних задач, з'являється *ефект непорівнянності* розв'язків.

Розглянемо приклад. Множина D складається з чотирьох можливих розв'язків $\vec{X}_1, \vec{X}_2, \vec{X}_3, \vec{X}_4$. Кожному розв'язку відповідають певні значення критеріїв F_1 і F_2 (критерії мінімізуються). Нехай є такі векторні оцінки: $\vec{F}(\vec{X}_1) = (2; 4)$, $\vec{F}(\vec{X}_2) = (3; 5)$, $\vec{F}(\vec{X}_3) = (5; 2)$, $\vec{F}(\vec{X}_4) = (2; 3)$, де значення, наведені в дужках, є оцінками розв'язків за критеріями F_1 і F_2 . Варіант \vec{X}_1 кращий за варіант \vec{X}_2 . Варіанти \vec{X}_1 і \vec{X}_3 непорівнянні між собою, варіант \vec{X}_1 кращий за першим критерієм, але гірший за другим. Варіант \vec{X}_1 гірший за варіант \vec{X}_4 . Варіанти \vec{X}_3 і \vec{X}_4 непорівнянні між собою, варіант \vec{X}_4 кращий за першим критерієм за варіант \vec{X}_3 , але гірший за другим. У результаті розв'язання отримали два недоміновані (такі, що не поліпшуються) розв'язки \vec{X}_3 і \vec{X}_4 . Непорівнянність розв'язків є формою невизначеності. Вибір між непорівнянними розв'язками є складною концептуальною проблемою і становить основний зміст багатокритеріальної оптимізації.

Нормалізація критеріїв. Оскільки частинні критерії мають різний фізичний зміст, тобто вимірюються в різних одиницях; масштаби їхні не співвимірні, тому неможливе порівняння якості отриманих результатів за кожним критерієм.

Операція приведення масштабів частинних критеріїв до єдиного, зазвичай безрозмірного, має назву *нормалізації* критеріїв.

Після нормалізації частинних критеріїв векторні критерії набувають деяких корисних властивостей. Головна з них - будь-яка перестановка частинних критеріїв призводить до векторної оцінки, яка входить до множини векторних оцінок (значень вихідної векторної оцінки).

Вибір принципу оптимальності, тобто потрібно визначити правило, яке дало б змогу сказати, який розв'язок кращий. Вибір принципу оптимальності - основна проблема векторної оптимізації. Формально описати принцип оптимальності (критерії "правильності розв'язку") - виявляється складним.

По-перше, об'єкти, які розглядає теорія оптимізації, настільки різноманітні, що встановити єдині принципи оптимальності для всіх класів задач не є можливим.

По-друге, цілі розв'язуваних задач, як правило, різні і часто протилежні.

По-третє, критерії правильності розв'язання залежать не тільки від характеру задачі, її мети тощо, а й від того, наскільки неупереджено вони обрані.

По-четверте, труднощі вибору розв'язку можуть ховатися і в самій постановці задачі, якщо потрібне досягнення нереальних результатів.

Врахування пріоритету критеріїв. Зазвичай із фізичного змісту задачі випливає, що окремі критерії мають різну важливість під час розв'язання задачі, тобто один окремих критерій має якийсь пріоритет над іншим частинним критерієм. Це слід враховувати під час вибору принципу оптимальності та визначення області можливих розв'язків, віддаючи перевагу важливішим критеріям.

Обчислення оптимуму задачі векторної оптимізації. Розвиток методів розв'язання задач векторної оптимізації йде, в основному, за трьома напрямками:

1. Заміна векторного критерію скалярним критерієм, тобто перехід до однокритеріальної задачі оптимізації;
2. Послідовне розв'язання скінченної множини однокритеріальних задач;
3. Звуження множини D з подальшим безпосереднім вибором оптимального розв'язку.

Для загальної задачі багатокритеріальної оптимізації не існує єдиного розв'язку. Розв'язок залежить від вибору принципу оптимальності, тобто частинні постановки задачі, що мають єдиний розв'язок, які призводять до різних результатів. Тому ОПР (особа, яка приймає рішення) на основі використання оптимізаційних методів має з найбільшою увагою ставитися насамперед до постановки задачі, до того, якою мірою саме така постановка відповідає проблемі, яка стоїть перед ним.

1.1.3 Оптимальність за Парето

Наріжним поняттям у багатокритеріальній оптимізації є *Парето-оптимальна (недомінована) альтернатива*, тому що пошук прийнятної ("оптимальної") альтернативи, яка є розв'язанням багатокритеріальної задачі, слід виконувати на множині недомінованих альтернатив. Отже, у множину Парето входять усі розв'язки, недоміновані іншими розв'язками. Саме тому так актуальними є методи, що дають змогу виділяти підмножини Парето-оптимальних альтернатив із множини можливих альтернатив.

Для задачі пошуку Парето-оптимальних розв'язків на множині можливих значень векторного критерію запровадимо відношення частинної впорядкованості, яке називається домінуванням за Парето, за якого $\vec{F}^{(1)} \prec \vec{F}^{(2)}$ тоді, коли для всіх $i = \overline{1, m}$ виконується умова $F_i^{(1)} \leq F_i^{(2)}$, і хоча б для одного i – умова $F_i^{(1)} < F_i^{(2)}$. Це правило дозволяє ввести відношення домінування на множині D , за якого $\vec{X}_1 \succ \vec{X}_2$ тоді, коли $\mathbf{F}(\mathbf{X}_1) \prec \mathbf{F}(\mathbf{X}_2)$.

Розв'язок $\vec{X}_1 \in D$ називається *Парето-оптимальним*, якщо не існує розв'язку $\vec{X}_2 \in D$ такого, щоб виконувалася умова $\vec{X}_1 \prec \vec{X}_2$. Очевидно, тоді у

складі множини D немає сенсу зберігати розв'язок \vec{X}_2 , воно витісняється (або, як кажуть, "домінується") розв'язком \vec{X}_1 . Якщо розв'язок не домінований жодним іншим розв'язком, то воно називається *недомінованим або оптимальним* у сенсі Парето. Унаслідок такої процедури відкидання задалегідь непридатних, невігідних розв'язків множина D зазвичай сильно зменшується: у ній зберігаються тільки так звані ефективні (інакше "паретівські") розв'язки, характерні тим, що ні для жодного з них не існує домінуючого розв'язку. Множина Парето-оптимальних розв'язків називається множиною Парето, яку позначимо P , $P \subset D$. Слід зазначити, що елементи множини P попарно непорівнянні між собою. Оптимальність за Парето означає, що не можна далі поліпшувати значення одного частинного критерію, не погіршуючи при цьому хоча б одного з інших.

Очевидно, що в узагальненому сенсі визначення оптимальності можна трактувати як опис (виділення) у множині D деякої нової підмножини P , тобто деяке звуження D до $P \subset D$. Залежно від характеру опису, підмножина P може виявитися порожньою, складатися з одного елемента, містити більше одного елемента. Опис P можна проводити або тільки за допомогою критеріїв F_i , або використовувати додаткові умови.

Приклад 1. Дано задачу багатокритеріальної оптимізації:

$$F_1(\vec{X}) = -x_1 + 2 \cdot x_2 \rightarrow \max,$$

$$F_2(\vec{X}) = 2 \cdot x_1 + x_2 \rightarrow \max,$$

$$F_3(\vec{X}) = x_1 - 3 \cdot x_2 \rightarrow \max,$$

з такими обмеженнями:

$$\begin{cases} x_1 + x_2 \leq 6, \\ 1 \leq x_1 \leq 3, \\ 1 \leq x_2 \leq 4 \end{cases}$$

Знайти безліч розв'язків, оптимальних за Парето.

Очевидно, у цій задачі множина Парето збігається з областю допустимих розв'язків (рис. 1.1).

Візьмемо будь-яку точку множини допустимих розв'язків. Під час руху вправо значення критеріїв $F_2(\vec{X})$ і $F_3(\vec{X})$ збільшується, але значення критерію $F_1(\vec{X})$ зменшується. Під час руху вліво значення критеріїв $F_2(\vec{X})$ і $F_3(\vec{X})$ зменшується, але значення критерію $F_1(\vec{X})$ збільшується. Якщо зсуватися вище,

то значення критеріїв $F_1(\vec{X})$ і $F_2(\vec{X})$ збільшаться, але значення критерію $F_3(\vec{X})$ зменшиться. Якщо зсунутися нижче, то значення критеріїв $F_1(\vec{X})$ і $F_2(\vec{X})$ зменшаться, але значення критерію $F_3(\vec{X})$ збільшиться.

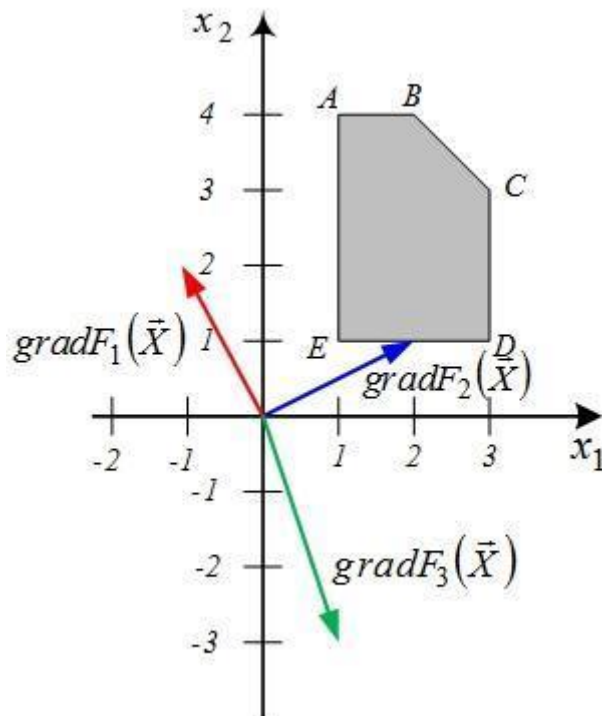


Рис.1.1 - Множина Парето, що збігається з множиною допустимих розв'язків

Отже, жодна з точок множини допустимих розв'язків не домінується іншими, тобто всі допустимі точки оптимальні за Парето.

Рис.1.2 ілюструє типові варіанти того, який вигляд має множина Парето в разі опуклої та неопуклої множини допустимих розв'язків (оптимальні за Парето розв'язки виділено жирним).

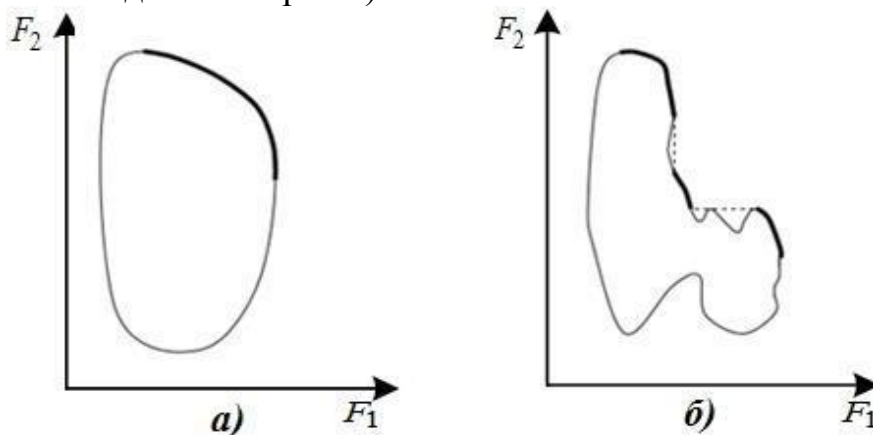


Рис.1.2 - Варіанти конфігурації множини Парето: а) опукла множина допустимих розв'язків; б) неопукла множина допустимих розв'язків

Якщо цільова функція - це векторна функція векторного аргументу і кожному малому значенню змінних \vec{X} відповідає мала зміна кожного критерію, то критеріальний простір - це область із внутрішніми точками і границею (не обов'язково замкнута). Нехай є тільки два критерії $F_1(\vec{X}) \rightarrow \max$ та $F_2(\vec{X}) \rightarrow \max$.

Розглянемо область можливих значень критеріїв (рис. 1.3). Точки, що лежать на одній вертикалі або на одній горизонталі завжди, безумовно, порівняльні (точки 1, 2, 3).

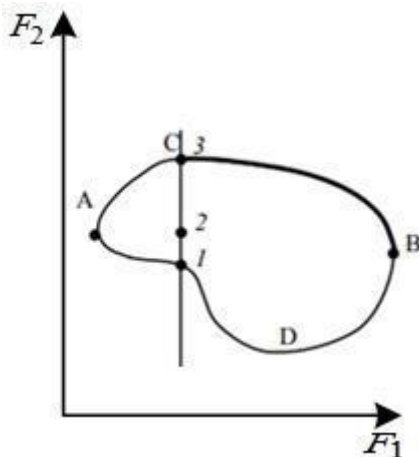


Рис.1.3 - Область можливих значень критеріїв

Виходить, що всі внутрішні точки можна відсіяти. Дуга АВ складається з найкращих точок за критерієм $F_2(\vec{X})$. Але частину точок дуги АВ, а саме точки дуги АС можна відкинути, оскільки вони гірші за точки СВ. Тому точками множини Парето будуть точки, що знаходяться на дузі СВ. Під час аналізу характерними є точки торкання вертикалей і горизонталей до області критеріїв (точки А, В, С, D).

Приклад 2. Проілюструємо прийом виділення паретівських розв'язків на прикладі задачі з двома критеріями $F_1(\vec{X}) \rightarrow \max$ і $F_2(\vec{X}) \rightarrow \max$. Множина D складається з 11-ти можливих розв'язків. Кожному розв'язку відповідають певні значення показників F_1 і F_2 . Нехай є такі векторні оцінки: $\vec{F}(\vec{X}_1) = (2; 4)$, $\vec{F}(\vec{X}_2) = (3; 5)$, $\vec{F}(\vec{X}_3) = (3; 3)$, $\vec{F}(\vec{X}_4) = (5; 2)$, $\vec{F}(\vec{X}_5) = (4; 3)$, $\vec{F}(\vec{X}_6) = (1; 3)$, $\vec{F}(\vec{X}_7) = (2; 3)$, $\vec{F}(\vec{X}_8) = (3; 2)$, $\vec{F}(\vec{X}_9) = (2; 2)$, $\vec{F}(\vec{X}_{10}) = (3; 1)$, $\vec{F}(\vec{X}_{11}) = (2; 1)$.

Векторні оцінки результатів представимо точками координатної площини (по осі абсцис відкладаємо значення критерію F_1 , а по осі ординат - значення критерію F_2). Використовуємо принцип оптимальності за Парето для виділення ефективних розв'язків. Розв'язок \vec{X}_1 витісняється розв'язком \vec{X}_2 , розв'язок \vec{X}_2 кращий за розв'язок \vec{X}_3 , \vec{X}_7 , \vec{X}_8 , \vec{X}_9 , \vec{X}_{10} і \vec{X}_{11} . Розв'язок \vec{X}_4 за першим

критерієм кращий за розв'язок \vec{X}_5 , а за другим навпаки, тобто маємо не покращувальні розв'язки, і т.д. Після проведеного аналізу залишилися три розв'язки \vec{X}_2 , \vec{X}_4 , \vec{X}_5 оптимальних за Парето.

З рисунка видно, що ефективні точки лежать на правій верхній границі області можливих розв'язків.

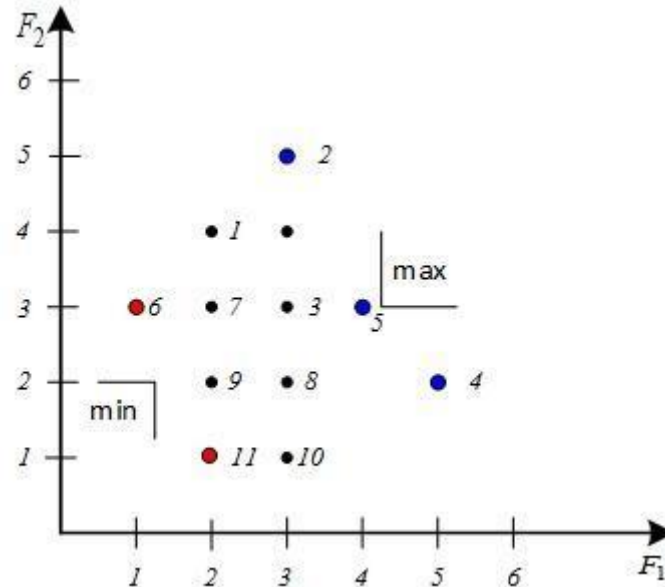


Рис.1.4 - Множина Парето

Коли з множини можливих розв'язків виділено ефективні, вибір може здійснюватися вже в границях цієї "ефективної" множини. На рис.1.4 множину Парето утворюють три розв'язки \vec{X}_2 , \vec{X}_4 , \vec{X}_5 ; з них \vec{X}_4 кращий за критерієм F_1 , а розв'язок \vec{X}_2 за критерієм F_2 . Справа ОПР, вибрати той варіант, який для нього кращий і "прийнятний" за обома критеріями.

У разі, коли множина Парето є безперервною, її векторні оцінки "заповнюють" деяку область Р на площині й отримуємо зображення, аналогічне зображеному на рис. 1.5. У цьому разі множина Парето-оптимальних оцінок (зображена жирною лінією) являє собою частину границі області, яка має назву фронту Парето, що відповідає мінімізації частинних критеріїв. У разі неопуклої області її Парето-оптимальна границя може мати більш екзотичний вигляд, наприклад, складатися з окремих ліній та/або точок. Для цього прикладу (у разі, якщо критерії максимізуються) - це правий пік.

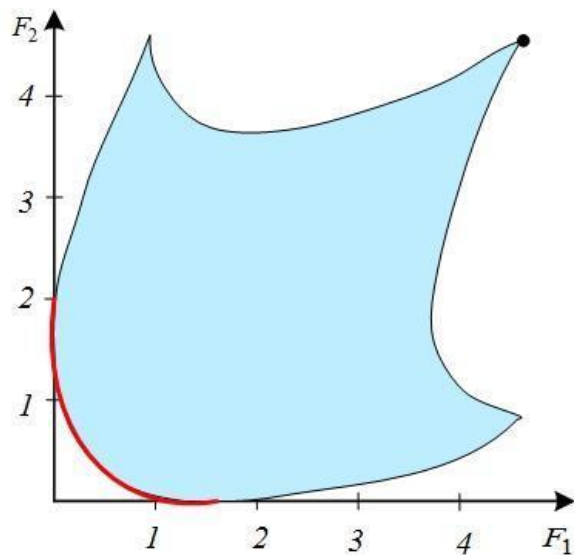


Рис.1.5 - Простір оцінок і фронт Парето

Виділення множини Парето багатокритеріальної задачі оптимізації часто не є остаточним розв'язком. Це пов'язано з тим, що за доволі великої вихідної множини варіантів множина Парето виявляється достатньо великою для того, щоб ОПР була б у змозі здійснити вибір самостійно. Отже, виділення множини Парето можна розглядати лише як попередній етап оптимізації й актуальною є проблема подальшого скорочення цієї множини.

Методи паретової границі (фронту Парето) розв'язання задач векторної оптимізації ґрунтуються на апроксимації границі Парето та інформуванні ОПР щодо неї. При цьому ОПР не ставлять запитань про його переваги. Від ОПР вимагається вказати найкращу точку границі Парето вже після вивчення інформації про границю. Завдяки цьому процес призначення найкращої точки границі Парето може бути розділений у часі з вивченням границі. Часто використовують чисельні методи отримання множин Парето. У множині D вводиться деяка сітка з числом вузлів N , наприклад, координати якої визначаються за допомогою датчика випадкових чисел, розподілених за рівномірним законом. Потім обчислюють значення векторного критерію $\vec{F}(\vec{X})$ у точках цієї сітки, після чого за кінцеву кількість порівнянь, використовуючи функцію вибору за Парето, будують множину Парето на зазначеній сітці, яка за великого N є наближенням множини Парето щодо D .

Особливий інтерес для практики становить випадок задач із двома критеріями. У цьому випадку множина паретівських точок являє собою одновимірне різноманіття на площині й допускає зручне графічне представлення.

Множину паретівських точок у двовимірному просторі критеріїв називають **компромісною кривою (КК)**.

Приклад 3. У квадраті $D = \{-1 \leq x_1 \leq 1, -1 \leq x_2 \leq 1\}$ задано два критерії $F_1(x_1, x_2) = 4x_1^2 + x_2^2$, $F_2(x_1, x_2) = (x_1 + 1)^2 + (x_2 - 1)^2$, які необхідно мінімізувати.

Визначимо мінімуми функцій F_1 і F_2 . Абсолютні мінімуми розташовані в точках $(0,0)$ і $(-1,1)$ і належать D . Побудуємо графіки паретівських кривих у області D і просторі критеріїв (рис. 1.6 і 1.7).

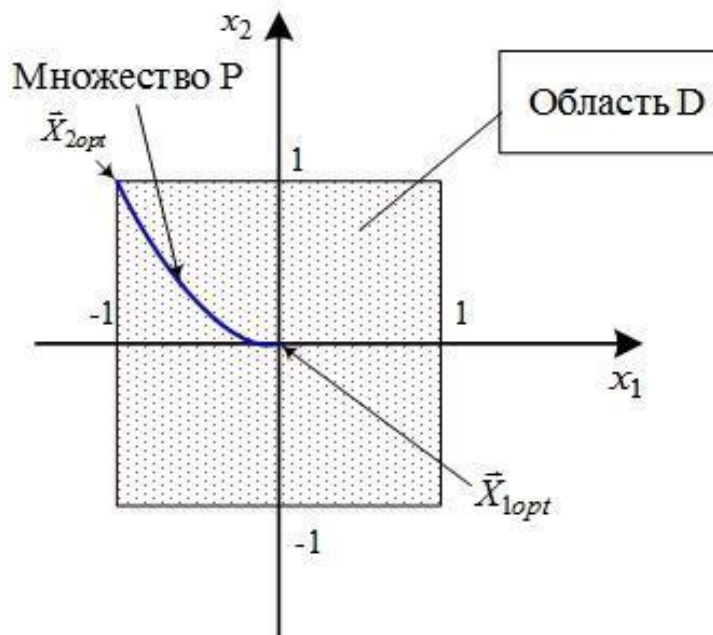


Рис.1.6 - Область D і множина P

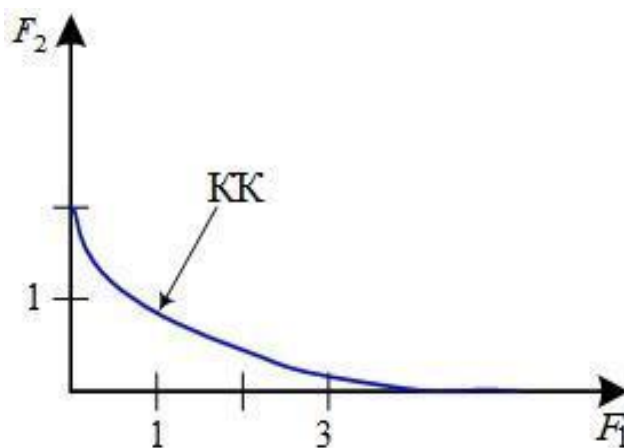


Рис.1.7 - Компромісна крива

1.2 Скаляризація векторного критерію агрегуванням частинних критеріїв за допомогою згортки

Одним із найпростіших способів розв'язання задачі оптимізації в багатокритеріальній постановці є метод скалярної згортки частинних критеріїв оптимальності. За такого підходу розв'язання багатокритеріальної задачі оптимізації вдається звести до розв'язання однокритеріальної задачі шляхом агрегування частинних критеріїв. Для цих цілей на практиці найчастіше використовують лінійну (адитивну):

$$f = \sum_{i=1}^m \alpha_i \cdot F_i \quad (1.1)$$

або степенева (мультиплікативна) згортка:

$$f = \prod_{i=1}^m F_i^{\lambda_i}, \quad (1.2)$$

де m - кількість частинних критеріїв, застосованих у задачі оптимізації.

Під час проведення оптимізації з використанням розглянутого підходу пріоритет частинних критеріїв вдається задавати, змінюючи чисельні значення вагових коефіцієнтів α_i і λ_i . Для визначеності, найчастіше вважають, що сума вагових коефіцієнтів дорівнює одиниці. Однак слід підкреслити складність, що виникає під час вибору величини вагових коефіцієнтів і задавання пріоритету частинних критеріїв. Крім адитивного та мультиплікативного критеріїв, широке застосування також знайшли узагальнений логічний критерій оптимальності, середньостепеневий критерій оптимальності та інші.

Узагальнений логічний критерій задається у формі:

$$f = \max_i \{\alpha_i F_i\}, \quad i = \overline{1, m} \quad (1.3)$$

і дає змогу мінімізувати найбільший зі зважених частинних критеріїв. У середньостепеневому узагальненому критерії в цільовій функції застосовується згортка:

$$f = \left(\frac{1}{m} \sum_{i=1}^m \alpha_i F_i^p \right)^{1/p}, \quad p > 0. \quad (1.4)$$

З огляду на те, що частинні критерії можуть бути пов'язані з різними характеристиками задачі, їхні значення здебільшого потребують нормування, що приводить їх до безрозмірних величин. При цьому можливі різні способи нормування, наприклад:

$$F'_i = \frac{F_i}{\bar{F}_i}, \quad F'_i = \frac{F_i}{F_{i\max}}, \quad F'_i = \frac{F_i - F_{i\min}}{F_{i\max} - F_{i\min}}, \quad (1.5)$$

де $F_{i\min}$, $F_{i\max}$, \bar{F}_i - відповідно найменше, найбільше та середнє значення i -го критерію $i = \overline{1, m}$.

Часто таке нормування дає змогу масштабувати критерій на інтервалі $[0, 1]$, що найзручніше для побудови агрегованого критерію оптимальності.

Приклад 4. Розв'язати задачу пошуку Парето-оптимальних розв'язків на множині $D = \{\bar{X} \mid x_i \in [-1, 1], i = \overline{1, 2}\}$, на якій задано два частинні критерії $F_1(\bar{X}) = x_1^2 + x_2^2$ і $F_2(\bar{X}) = (x_1 - 1)^2 + (x_2 - 1)^2$.

Порядок виконання завдання:

1. Задати цільові функції.
2. Задати значення вагових коефіцієнтів.
3. Скористатися скаляризацією векторного критерію за допомогою адитивної (мультиплікативної) згортки і звести задачу до однокритеріальної.
4. Знайти розв'язок однокритеріальної задачі, використовуючи одну з розглянутих раніше процедур умовної або безумовної оптимізації.

Продемонструємо описану технологію оптимізації на тестовому прикладі з апріорі відомим розв'язком.

Для цієї багатокритеріальної задачі оптимізації множина Парето являє собою відрізок прямої, що з'єднує точки $(0, 0)$ і $(1, 1)$ (рис. 1.8). Нижче наводиться приклад її розв'язання в середовищі MathCAD.

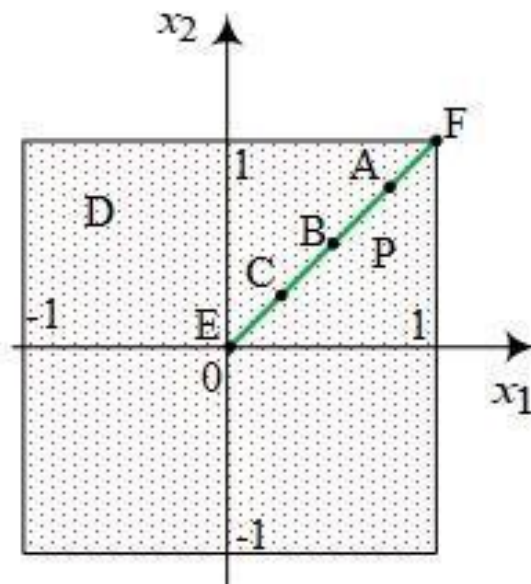


Рис. 1.8 - Множина Парето для розглядуваного прикладу

Приклад виконання завдання в MathCAD

Приклад 4. Розв'язати задачу пошуку Парето-оптимальних розв'язків на множині $D = \{\bar{X} \mid x_i \in [-1, 1], i = \overline{1, 2}\}$, на якій задані два частинних критеріїв $F_1(\bar{X}) = x_1^2 + x_2^2$ та $F_2(\bar{X}) = (x_1 - 1)^2 + (x_2 - 1)^2$.

РОЗВ'ЯЗОК ЗАДАЧІ ВЕКТОРНОЇ ОПТИМІЗАЦІЇ СКАЛЯРИЗАЦІЄЮ З ВИКОРИСТАННЯМ ЛІНІЙНОЇ ЗГОРТКИ

$$f_1(x_1, x_2) := x_1^2 + x_2^2$$

- частинні критерії

$$f_2(x_1, x_2) := (x_1 - 1)^2 + (x_2 - 1)^2$$

ПОШУК РОЗВ'ЯЗКУ В ТОЧЦІ А ТЕСТОВОГО ПРИКЛАДУ

$ALFA_1 := 0.2$ $ALFA_2 := 0.8$ - вагові коефіцієнти

$f_AGR(x1, x2) := ALFA_1 \cdot f1(x1, x2) + ALFA_2 \cdot f2(x1, x2)$

- адитивна (лінійна) згортка частинних критеріїв

$x1 := 1$ $x2 := 3$ - задання початкових значень

Given змінним

$-1 \leq x1 \leq 1$ $-1 \leq x2 \leq 1$ - обмеження на зміну змінних

$PR := \text{Minimize}(f_AGR, x1, x2)$ - виклик функції безумовної оптимізації

$PR = \begin{pmatrix} 0.8 \\ 0.8 \end{pmatrix}$ - отриманий розв'язок в точці **A**

ПОШУК РОЗВ'ЯЗКУ В ТОЧЦІ В ТЕСТОВОГО ПРИКЛАДУ

$ALFA_1 := 0.5$ $ALFA_2 := 0.5$ - вагові коефіцієнти

$f_AGR(x1, x2) := ALFA_1 \cdot f1(x1, x2) + ALFA_2 \cdot f2(x1, x2)$

- адитивна (лінійна) згортка частинних критеріїв

Given

$-1 \leq x1 \leq 1$ $-1 \leq x2 \leq 1$ - обмеження на зміну змінних

$PR := \text{Minimize}(f_AGR, x1, x2)$ - виклик функції безумовної оптимізації

$PR = \begin{pmatrix} 0.5 \\ 0.5 \end{pmatrix}$ - отриманий розв'язок в точці **B**

ПОШУК РОЗВ'ЯЗКУ В ТОЧЦІ С ТЕСТОВОГО ПРИКЛАДУ

$ALFA_1 := 0.8$ $ALFA_2 := 0.2$ - вагові коефіцієнти

$f_AGR(x1, x2) := ALFA_1 \cdot f1(x1, x2) + ALFA_2 \cdot f2(x1, x2)$

- адитивна (лінійна) згортка частинних критеріїв

Given

$-1 \leq x1 \leq 1$ $-1 \leq x2 \leq 1$ - обмеження на зміну змінних

$PR := \text{Minimize}(f_AGR, x1, x2)$ - виклик функції безумовної оптимізації

$PR = \begin{pmatrix} 0.2 \\ 0.2 \end{pmatrix}$ - отриманий розв'язок в точці **C**

ПОШУК РОЗВ'ЯЗКУ В ТОЧЦІ E ТЕСТОВОГО ПРИКЛАДУ

$LAMBDA_1 := 0.5001$ $LAMBDA_2 := 0.4999$ - вагові коефіцієнти

$$f_AGR(x1, x2) := f1(x1, x2)^{LAMBDA_1} \cdot f2(x1, x2)^{LAMBDA_2}$$

- мультиплікативна згортка частинних критеріїв

$x1 := 0$ $x2 := 3$ - задання початкових значень змінним

Given

$-1 \leq x1 \leq 1$ $-1 \leq x2 \leq 1$ - обмеження на зміну змінних

$PR := \text{Minimize}(f_AGR, x1, x2)$ - виклик функції безумовної оптимізації

$$PR = \begin{pmatrix} -8.597 \times 10^{-6} \\ -1.426 \times 10^{-5} \end{pmatrix} \quad \text{- отриманий розв'язок у точці E}$$

ПОШУК РОЗВ'ЯЗКУ В ТОЧЦІ F ТЕСТОВОГО ПРИКЛАДУ

$LAMBDA_1 := 0.499$ $LAMBDA_2 := 0.501$ - вагові коефіцієнти

$$f_AGR(x1, x2) := f1(x1, x2)^{LAMBDA_1} \cdot f2(x1, x2)^{LAMBDA_2}$$

- мультиплікативна згортка частинних критеріїв

$x1 := 3$ $x2 := 3$ - задання початкових значень змінним

Given

$-1 \leq x1 \leq 1$ $-1 \leq x2 \leq 1$ - обмеження на зміну змінних

$PR := \text{Minimize}(f_AGR, x1, x2)$ - виклик функції безумовної оптимізації

$$PR = \begin{pmatrix} 1 \\ 1 \end{pmatrix} \quad \text{- отриманий розв'язок у точці F}$$

Під час розв'язання задачі з використанням в якості цільової функції адитивної згортки (1.1) отриманий розв'язок:

- при $\alpha_1 = 0.2$ і $\alpha_2 = 0.8$ був знайдено розв'язок $X_A = \{0.8, 0.8\}$;
- при $\alpha_1 = \alpha_2 = 0.5$ - розв'язок $X_B = \{0.5, 0.5\}$;
- при $\alpha_1 = 0.8$ і $\alpha_2 = 0.2$ визначено розв'язок $X_C = \{0.2, 0.2\}$.

Усі знайдені розв'язки, як видно з рис. 1.8, належать до множини Парето цієї задачі.

У разі використання мультиплікативної згортки частинних критеріїв (1.2) з ваговими коефіцієнтами $\lambda_1 = \lambda_2 = 0.5$ знаходяться розв'язки $\vec{X}_E = \{0, 0\}$ і $\vec{X}_F = \{1, 1\}$, кожний з яких також є Парето-оптимальним. Кожна з цих точок є точкою мінімуму одного з частинних критеріїв, тому слід зазначити, що використання мультиплікативної згортки частинних критеріїв у загальному випадку виправдане, коли на множині розв'язків жоден із критеріїв не набуває нульового значення. Інакше цільова функція f дорівнюватиме нулю, коли один із

частинних критеріїв дорівнює нулю незалежно від чисельних значень інших критеріїв.



ЗАВДАННЯ ДО САМОСТІЙНОЇ РОБОТИ

Завдання 1. Розв'язати задачу пошуку Парето-оптимальних розв'язків на множині D , на якій задано два частинні критерії $f_1(\vec{X})$ і $f_2(\vec{X})$, використовуючи метод скалярної згортки

№ п/п	Завдання
1	$D = \{ \vec{X} \mid x_i \in [0, 1] \}, X = 30,$ $f_1(x_1) = x_1 \rightarrow \min$ $f_2(x_2, x_3, \dots, x_{ X }) = g(x_2, x_3, \dots, x_{ X }) \cdot h(f_1, g) \rightarrow \min.$ $g(x_2, \dots, x_{ X }) = 1 + 9 \cdot \sum_{i=2}^{ X } \frac{x_i}{ X - 1}$ $h(f_1, g) = 1 - \sqrt{\frac{f_1}{g}}$
2	$D = \{ \vec{X} \mid x_i \in [0, 1] \}, X = 30,$ $f_1(x_1) = x_1 \rightarrow \min$ $f_2(x_2, x_3, \dots, x_{ X }) = g(x_2, x_3, \dots, x_{ X }) \cdot h(f_1, g) \rightarrow \min.$ $g(x_2, \dots, x_{ X }) = 1 + 9 \cdot \sum_{i=2}^{ X } \frac{x_i}{ X - 1}$ $h(f_1, g) = 1 - \left(\frac{f_1}{g} \right)^2$
3	$D = \{ \vec{X} \mid x_i \in [0, 1] \}, X = 30,$ $f_1(x_1) = x_1 \rightarrow \min$ $f_2(x_2, x_3, \dots, x_{ X }) = g(x_2, x_3, \dots, x_{ X }) \cdot h(f_1, g) \rightarrow \min.$ $g(x_2, \dots, x_{ X }) = 1 + 9 \cdot \sum_{i=2}^{ X } \frac{x_i}{ X - 1}$ $h(f_1, g) = 1 - \sqrt{\frac{f_1}{g}} - \frac{f_1}{g} \cdot \sin(10 \cdot \pi \cdot f_1)$
4	$D = \{ \vec{X} \mid x_{1,2} \in [-1, 1] \},$ $f_1(\vec{X}) = 4 \cdot x_1^2 + x_2^2 \rightarrow \min$

№ п/п	Завдання
	$f_2(\vec{X}) = (x_1 + 1)^2 + (x_2 - 1)^2 \rightarrow \min$
5	$D = \{\vec{X} \mid x_i \in [-5, 5], i \in [2: X], x_1 \in [0, 1], X = 10,$ $f_1(x_1) = x_1 \rightarrow \min$ $f_2(x_2, x_3, \dots, x_{ X }) = g(x_2, x_3, \dots, x_{ X }) \cdot h(f_1, g) \rightarrow \min.$ $g(x_2, \dots, x_{ X }) = 1 + 10 \cdot (X - 1) + \sum_{i=2}^{ X } (x_i^2 - 10 \cdot \cos(4 \cdot \pi \cdot x_i))$ $h(f_1, g) = 1 - \sqrt{\left(\frac{f_1}{g}\right)}$
6	$D = \{\vec{X} \mid x_i \in [0, 1], X = 30,$ $f_1(x_1) = 1 - \exp(-4 \cdot x_1) \cdot \sin^6(6 \cdot \pi \cdot x_1) \rightarrow \min$ $f_2(x_2, x_3, \dots, x_{ X }) = g(x_2, x_3, \dots, x_{ X }) \cdot h(f_1, g) \rightarrow \min.$ $g(x_2, \dots, x_{ X }) = 1 + 9 \cdot \left(\frac{1}{ X - 1} \cdot \sum_{i=2}^{ X } x_i\right)^{0.25}$ $h(f_1, g) = 1 - \left(\frac{f_1}{g}\right)^2$ <p>задача має невіпуклий фронт Парето</p>
7	$D = \{\vec{X} \mid x_i \in [0, 1], i = 1, 2\},$ $f_1(\vec{X}) = x_1^2 + x_2^2 \rightarrow \min$ $f_2(\vec{X}) = (x_1 - 1)^2 + (x_2 - 1)^2 \rightarrow \min$ <p>задача має неперервний опуклий фронт Парето</p>
8	$D = \{\vec{X} \mid x_i \in [0, 1], i = 1, 2\},$ $f_1(x_1) = 4 \cdot x_1 \rightarrow \min$ $f_2(\vec{X}) = g(\vec{X}) \cdot h(\vec{X}) \rightarrow \min$ $h(\vec{X}) = \begin{cases} 1 - \left(\frac{f_1(x_1)}{g(\vec{X})}\right)^\alpha, & \text{якщо } f_1(x_1) \leq g(\vec{X}) \\ 0, & \text{інакше} \end{cases}$ $g(\vec{X}) = g(x_2) = 4 - 3 \cdot \exp\left(-\left(\frac{x_2 - 0.2}{0.02}\right)^2\right)$

<i>№ n/n</i>	<i>Завдання</i>
	значенню $\alpha = 0,25$ відповідає опукла множина Парето, $\alpha = 0,4$ -неопукла
9	$D = \{\vec{X} \mid x_i \in [0, 1], i \in [1: X], X = 30,$ $f_1(x_1) = x_1 \rightarrow \min$ $f_2(\vec{X}) = g(\vec{X}) \cdot h(\vec{X}) \rightarrow \min.$ $g(\vec{X}) = g(x_2, \dots, x_{ X }) = 1 + 9 \cdot \sum_{i=2}^{ X } \frac{x_i}{ X - 1}$ $h(\vec{X}) = 1 - \sqrt{\frac{f_1(x_1)}{g(\vec{X})}}$ <p>задача має неперервний опуклий фронт Парето</p>
10	$D = \{\vec{X} \mid x_i \in [-4, 4], 1 \leq i \leq n$ $f_1(\vec{X}) = 1 - \exp\left[-\sum_{i=1}^n \left(x_i - \frac{1}{\sqrt{n}}\right)^2\right] \rightarrow \min$ $f_2(\vec{X}) = 1 - \exp\left[-\sum_{i=1}^n \left(x_i + \frac{1}{\sqrt{n}}\right)^2\right] \rightarrow \min$
11	$D = \{\vec{X} \mid x_i \in [-5, 5], 1 \leq i \leq 3$ $f_1(\vec{X}) = \sum_{i=1}^2 \left[-10 \cdot \exp\left[-0.2 \cdot \sqrt{x_i^2 + x_{i+1}^2}\right]\right] \rightarrow \min$ $f_2(\vec{X}) = \sum_{i=1}^3 \left[x_i ^{0.8} + 5 \cdot \sin(x_i^3)\right] \rightarrow \min$
12	$D = \{\vec{X} \mid x_{1,2} \in [-10, 10]\},$ $f_1(\vec{X}) = x_1^2 + x_1 \cdot x_2 + 3 \cdot x_2^2 \rightarrow \min$ $f_2(\vec{X}) = (x_1 - 4)^2 + (x_2 - 3)^2 \rightarrow \min$
13	$D = \{\vec{X} \mid x_1 \in [-0,5, 0,5], x_2 \in [0, 1]\},$ $f_1(\vec{X}) = x_1^2 + 4 \cdot x_2^2 \rightarrow \min$ $f_2(\vec{X}) = (x_1 + 1)^2 + (x_2 - 1)^2 \rightarrow \min$ $ x_2 - x_1 - 0.375 \geq 0.125$
14	$f_1(\vec{X}) = 2 \cdot x_1 + x_2 \rightarrow \max$ $f_2(\vec{X}) = 2 \cdot x_1 + 3 \cdot x_2 \rightarrow \max$

<i>№ п/п</i>	<i>Завдання</i>
	$\begin{cases} x_1^2 + x_2^2 \leq 100 \\ x_1 \geq 0, x_2 \geq 0 \end{cases}$
15	$D = \{\vec{X} \mid x_1 \in [0, 5], x_2 \in [0, 3]\},$ $f_1(\vec{X}) = 4 \cdot x_1 + 4 \cdot x_2 \rightarrow \min$ $f_2(\vec{X}) = (x_1 - 5)^2 + (x_2 - 5)^2 \rightarrow \min$ $g_1(\vec{X}) = (x_1 - 5)^2 + x_2^2 \leq 25$ $g_2(\vec{X}) = (x_1 - 8)^2 + (x_2 + 3)^2 \geq 7.7$
16	$D = \{\vec{X} \mid x_{1,2} \in [-20, 20]\},$ $f_1(\vec{X}) = 2 + (x_1 - 2)^2 + (x_2 - 1)^2 \rightarrow \min$ $f_2(\vec{X}) = 9 \cdot x_1 - (x_2 - 1)^2 \rightarrow \min$ $g_1(\vec{X}) = x_1^2 + x_2^2 \leq 225$ $g_2(\vec{X}) = x_1 - 3 \cdot x_2 + 10 \leq 0$
17	$D = \{\vec{X} \mid x_{1,2} \in [-7, 4]\},$ $f_1(\vec{X}) = x_1^2 - x_2 \rightarrow \min$ $f_2(\vec{X}) = -0.5 \cdot x_1 - x_2 - 1 \rightarrow \min$ $g_1(\vec{X}) = 6,5 - \frac{x_1}{6} - x_2 \geq 0$ $g_2(\vec{X}) = 7,5 - \frac{x_1}{2} - x_2 \geq 0$ $g_3(\vec{X}) = 30 - 5 \cdot x_1 - x_2 \geq 0$
18	$D = \{\vec{X} \mid x_{1,2,6} \in [0, 10], x_{3,5} \in [1, 5], x_4 \in [0, 6]\},$ $f_1(\vec{X}) = -25 \cdot (x_1 - 2)^2 - (x_2 - 2)^2 - (x_3 - 1)^2 -$ $-(x_4 - 4)^2 - (x_5 - 1)^2 \rightarrow \min$ $f_2(\vec{X}) = \sum_{i=1}^6 x_i^2 \rightarrow \min$ $g_1(\vec{X}) = x_1 + x_2 - 2 \geq 0, \quad g_2(\vec{X}) = 6 - x_1 - x_2 \geq 0$ $g_3(\vec{X}) = 2 - x_2 + x_1 \geq 0, \quad g_4(\vec{X}) = 2 - x_1 + 3 \cdot x_2 \geq 0$ $g_5(\vec{X}) = 4 - (x_3 - 3)^2 - x_4 \geq 0$ $g_6(\vec{X}) = (x_5 - 3)^2 + x_6 - 4 \geq 0$

<i>№ п/п</i>	<i>Завдання</i>
19	$D = \{\vec{X} \mid x_{1,2} \in [0, 1]\},$ $f_1(\vec{X}) = x_1 \rightarrow \min$ $f_2(\vec{X}) = (1 + x_2) \cdot \exp\left(-\frac{x_1}{x_2 + 1}\right) \rightarrow \min$ $g_1(\vec{X}) = \frac{f_2(\vec{X})}{0.858 \cdot \exp(-0.541 \cdot f_1(\vec{X}))} \geq 1$ $g_2(\vec{X}) = \frac{f_2(\vec{X})}{0.728 \cdot \exp(-0.295 \cdot f_1(\vec{X}))} \geq 1$
20	$D = \{\vec{X} \mid x_1 \in [0.1, 1], x_2 \in [0, 5]\},$ $f_1(\vec{X}) = x_1 \rightarrow \min$ $f_2(\vec{X}) = \frac{x_2 + 1}{x_1} \rightarrow \min$ $g_1(\vec{X}) = 9 \cdot x_1 + x_2 \geq 6$ $g_2(\vec{X}) = 9 \cdot x_1 - x_2 \geq 1$

1.3 Апроксимація фронту Парето сітковим методом дослідження простору параметрів

Недоліки, пов'язані зі спробами формалізації в принципі неформалізованої задачі, приводять до думки про пошук розв'язання, що використовує ідею повного перегляду всіх можливих варіантів розв'язків і вибору з них найкращого. Природно, що такий повний перегляд неможливий, оскільки кількість точок перегляду нескінченна. У найпростішому випадку точки перегляду можуть бути розподілені рівномірно в просторі пошуку. Для того, щоб зменшити кількість переглянутих точок, можна якимось способом організувати процедуру перегляду.

На цій ідеї й засновано метод розв'язання проектно-конструкторських задач із суперечливими критеріями, запропонований І.М. Соболев і Р.Б. Статніковим, який базується на твердженні, що максимальна кількість точок, які можна побачити, за мінімуму обчислень досягається, якщо точки обирають із так званої ЛП_τ - послідовності.

Послідовність точок P_0, P_1, \dots, P_i куба K^n називається **ЛП $_{\tau}$ - послідовністю**, якщо будь-яка її двійкова ділянка, яка містить не менше ніж $2^{\tau+1}$ точок, являє собою П-сітку.

Сітка, що складається з точок $N = 2^{\nu}$ куба K^n , називається **П $_{\tau}$ -сіткою**, якщо кожному двійковому паралелепіпеду Π_k с об'ємом $V_{\Pi_k} = \frac{2^{\tau}}{N}$ належать 2^{τ} точок сітки. При цьому завжди передбачається, що $\nu > \tau$.

Отже, переглядаючи варіанти в точках, що відповідають ЛП $_{\tau}$ - послідовності, і, обчислюючи значення критеріїв у цих точках, можна ухвалювати ефективні розв'язки.

Обчислення координат точок Q_i може здійснюватися за допомогою матриці напрямних чисельників табл.1.1.

За таблицею чисельників обчислюються точки з номерами $0 \leq i \leq 10^{20}$ у кубі K^n розмірності $n \leq 51$. Для цього за заданим номером точки i обчислюють:

$$m = 1 + \left[\frac{\ln(i)}{\ln(2)} \right],$$

а потім для $j = 1, 2, \dots, n$, де n - розмірність точки:

$$q_{i,j} = \sum_{k=1}^m 2^{-k+1} \cdot \left\{ \frac{1}{2} \cdot \sum_{l=k}^m \left[2 \{ i 2^{-l} \} \right] \cdot \left[2 \{ r_j^{(l)} \cdot 2^{k-1-l} \} \right] \right\},$$

де $[]$, $\{ \}$ - квадратні та фігурні дужки означають відповідно цілу та дробову частини вмісту цих дужок.

Таблиця 1.1 - Чисельники напрямних чисел

<i>j</i>	<i>i=1</i>	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	1	3	5	15	17	51	85	255	257	771	1285	3855	4369	13107	21845	65535	65537	196611	327685	983055
3	1	1	7	11	13	61	67	79	465	721	823	4091	4125	4141	28723	45311	53505	250113	276231	326411
4	1	3	7	5	7	43	49	147	439	1013	727	987	5889	6915	16647	49925	116487	83243	116529	715667
5	1	1	5	3	15	51	125	141	177	759	267	1839	6929	16241	16565	17139	82207	50979	252717	851901
6	1	3	1	1	9	59	25	89	321	835	833	4033	3913	11643	18777	35225	102401	45059	36865	299009
7	1	1	3	7	31	47	109	173	181	949	471	2515	6211	2147	3169	35873	33841	99889	247315	1032727
8	1	3	3	9	9	57	43	43	225	113	1601	579	1731	11977	7241	63609	81003	15595	144417	685617
9	1	1	7	13	3	35	89	9	235	929	1341	3863	1347	4417	5087	12631	103445	152645	130127	775365
10	1	3	5	11	27	53	69	25	103	615	913	977	6197	14651	2507	27109	5205	91369	302231	172023
11	1	1	5	1	15	19	113	115	411	157	1725	3463	2817	9997	7454	12055	44877	24895	508255	574033
12	1	3	7	3	29	51	47	97	233	39	2021	2909	5459	2615	13329	35887	97323	83101	320901	810643
13	1	1	7	7	21	61	55	19	59	761	1905	3379	8119	13207	8965	9997	75591	226659	187499	628265
14	1	3	1	9	23	37	97	97	353	169	375	1349	5121	13313	19457	1033	62487	250917	234593	308321
15	1	1	3	5	19	33	3	197	329	983	893	3739	7669	2671	18391	31161	12111	259781	36159	232401
16	1	3	3	13	11	7	37	101	463	657	1599	347	2481	5201	3123	32253	78043	63447	508757	974837
17	1	1	7	13	25	5	83	255	385	647	415	387	7101	11469	11699	15865	49173	147489	81991	802875
18	1	3	5	11	7	11	103	29	111	581	605	2381	2677	14855	721	26903	100419	206167	241771	987201
19	1	1	1	3	13	39	27	203	475	505	819	2821	1405	12165	709	41543	57545	77163	357231	378135
20	1	3	1	15	17	63	13	65	451	833	975	1873	7423	5837	20481	12291	86017	12303	299025	774207
21	1	1	5	5	1	27	33	195	263	139	915	1959	725	5387	19285	5165	27985	69809	128325	164575
22	1	3	3	3	25	17	115	177	19	147	1715	1929	2465	12483	13057	28931	54019	21251	62233	248081
23	1	1	3	15	29	15	41	105	249	203	1223	2389	471	12945	32321	29377	127427	103759	472541	1008719
24	1	3	1	7	3	23	79	17	275	81	1367	3251	2887	1279	4865	64771	24321	42247	338691	599831
25	1	3	7	9	31	29	17	47	369	337	663	1149	1715	187	12285	53631	110851	4357	153347	671033

Продовження табл. 1.1

<i>j</i>	<i>i=1</i>	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
26	1	1	5	13	11	3	29	169	393	829	629	243	5595	8133	4929	10817	8261	189901	255947	734787
27	1	3	1	9	5	21	119	109	167	989	525	3609	5689	11819	15889	48083	67537	63993	336469	749285
28	1	1	3	1	23	13	75	149	333	375	469	1131	441	14471	12625	8881	37707	85105	479495	911133
29	1	3	3	11	27	31	73	15	473	365	981	1701	3169	7615	8405	41135	106823	107847	339031	977707
30	1	1	7	7	19	25	105	213	469	131	1667	143	4485	2981	12593	60913	15703	26967	507907	344073
31	1	3	5	5	21	9	7	135	101	215	1587	1339	6311	4081	28637	60935	94129	109273	475921	281389
32	1	1	1	15	5	49	59	253	21	733	1251	3497	3557	7223	13425	58577	69521	217151	424277	789985
33	1	1	1	1	1	33	65	191	451	451	451	2499	483	11843	28285	12029	86021	217093	348165	176165
34	1	3	5	15	17	19	21	155	229	447	481	1571	3781	10799	15893	959	19793	213491	377941	414943
35	1	1	7	11	13	29	3	175	247	177	721	983	3195	9277	15405	19637	87283	186143	343297	1041185
36	1	3	7	5	7	11	113	63	297	57	483	4021	5213	2031	4677	26607	20931	54345	259163	741087
37	1	1	5	3	15	19	61	47	403	471	1209	1625	5085	15371	19493	56445	26369	27399	521499	132383
38	1	3	1	1	9	27	89	7	497	979	1457	3217	185	6603	1129	36087	66817	98051	451841	175361
39	1	1	3	7	31	15	45	23	61	197	415	1163	7323	7563	25321	52563	37745	81777	235347	539895
40	1	3	3	9	9	25	107	39	361	251	1435	2977	1713	11617	14979	5455	68289	209987	346179	521289
41	1	3	7	13	3	3	25	55	215	517	725	3391	4021	4129	4099	12345	102733	21287	128115	20689
42	1	1	5	11	27	21	5	71	393	137	861	675	5875	12061	25469	47423	29505	124097	444613	430923
43	1	3	5	1	15	51	49	87	125	567	41	3093	5363	3471	17589	50131	33137	98739	361365	426737
44	1	1	7	3	29	19	111	103	285	1021	1619	1495	4977	15919	6731	43771	23313	151281	270519	11187
45	1	3	7	7	21	29	119	119	501	167	1579	3443	5441	1097	13483	58779	36561	116819	420599	998391
46	1	1	1	9	23	5	33	135	277	877	1701	557	1779	10369	15325	33331	118321	59665	498897	494137
47	1	3	3	5	19	1	67	153	199	929	869	675	6777	14343	18465	63615	43349	30799	322567	939017
48	1	1	3	13	11	39	101	169	301	269	1151	1489	287	8475	6929	46013	52785	75249	14035	507165
49	1	1	7	13	25	37	19	185	19	327	1897	2303	6919	16139	16677	34579	120981	239693	73299	863545
50	1	3	5	11	7	43	39	201	83	997	1679	3925	1517	305	21765	45827	91157	113679	204881	761911
51	1	1	1	3	13	7	91	217	351	91	1355	3705	1875	7621	4381	9079	94533	37261	431301	176455

Приклад 5. Розглянемо приклад розв'язання багатокритеріальної задачі з двома цільовими функціями.

У найпростішому випадку для визначеності сформуємо матрицю параметрів, використовуючи генератор випадкових чисел *rnd()*. Матриця параметрів *q* у такий спосіб представлена масивом 2×300 , що містить 300 варіантів розв'язків, які складаються зі значень двох критеріїв. Для виділення невідомованих стовпців масиву, а тим самим фронту Парето, застосуємо наведені нижче процедури, які визначають також номери таких стовпців. Для ухвалення остаточного розв'язку зручно скористатися побудованими компромісними кривими, що являють собою залежність значень одного критерію у функції значень другого.

Вибір відповідного варіанта розв'язку робиться на основі графіків. Зазначимо, що під час розв'язання реальних задач оптимізації, матриця параметрів формується на основі наявних цільових функцій.

Приклад виконання завдання в MathCAD

Приклад 5. Розглянемо приклад розв'язання багатокритеріальної задачі з двома цільовими функціями.

АПРОКСИМАЦІЯ ПАРЕТО - ФРОНТУ ЗАДАЧІ ВЕКТОРНОЇ ОПТИМІЗАЦІЇ З ДВОМА КРИТЕРІЯМИ МЕТОДОМ ДОСЛІДЖЕННЯ ПРОСТОРУ ПАРАМЕТРІВ

ORIGIN := 1

Функція *Smp_1(P, Q, i, n, par)*, яка здійснює операцію порівняння точки **P** з *i*-ю точкою множини можливих розв'язків, що зберігається в масиві **Q** розмірності **n**.

par < 0 задається при мінімізації;

par > 0 задається при максимізації.

В результаті функція повертає:

-1, якщо точки збігаються;

0, якщо точка **P** гірша;

1, якщо точка **P** не гірша.

```
Smp_1(P, Q, j, n, par) :=
  for k ∈ 1..n
    Lk ← if(Pk = Qk,j, 100, -100)
  Rez_1 ← ∑ L
  return -1 if Rez_1 = n·100
  for k ∈ 1..n
    | Lk ← if(Pk ≥ Qk,j, 0, 1) if par < 0
    | Lk ← if(Pk ≤ Qk,j, 0, 1) otherwise
  Rez_1 ← ∑ L
  return 0 if Rez_1 = 0
  return 1 otherwise
```

Функція $\text{Cmp_Pareto}(P, Q, n, N, \text{par})$, яка визначає належність точки P з масиву Q розмірності $n \times N$ Парето-фронтуну невідомінованих розв'язків
 В результаті функція повертає:
 10, якщо точка P належить до Парето-фронтуну;
 -10, якщо точка P не належить.

```

Cmp_Pareto(P, Q, n, N, par) := | j ← 1
                             | while j ≤ N
                             |-----|
                             | | return -10 if Cmp_1(P, Q, j, n, par) = 0
                             | | j ← j + 1
                             | rez ← 10
    
```

Функція $\text{Basic_Pareto}(Q, N, n, \text{par})$, яка виділяє ефективні точки з N точок масиву критеріїв Q розмірності $n \times N$.
 n - розмірність простору критеріїв якості;
 N - число варіантів параметрів.

```

Basic_Pareto(Q, N, n, par) := | i ← 1
                             | for j ∈ 1..N
                             | | P ← Q<j>
                             | | D<i> ← P if Cmp_Pareto(P, Q, n, N, par) = 10
                             | | i ← i + 1 if Cmp_Pareto(P, Q, n, N, par) = 10
                             | D
    
```

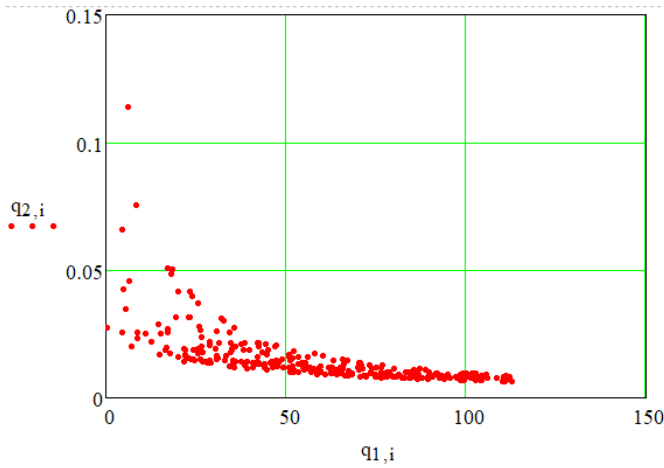
Функція $\text{Num_Pareto}(Q, N, n, \text{par})$, яка повертає вектор номерів паретових точок з N точок масиву критеріїв Q розмірності $n \times N$.
 n - розмірність простору критеріїв якості;
 N - число варіантів параметрів.

```

Num_Pareto(Q, N, n, par) := | i ← 1
                             | for j ∈ 1..N - 1
                             | | P ← Q<j>
                             | | Num<i> ← j if Cmp_Pareto(P, Q, n, N, par) = 10
                             | | i ← i + 1 if Cmp_Pareto(P, Q, n, N, par) = 10
                             | Num
    
```

Побудова масиву варіантів

$i := 1..300$
 $q_{1,i} := \frac{i}{3} + \text{rnd}(20)$ $q_{2,i} := \frac{1}{q_{1,i} + \text{rnd}(50)}$



Матриця параметрів із 300 варіантів за двома критеріями

	1	2	3	4	5	6	7	8	9	10	11	
q =	1	0.359	4.533	12.7	8.339	18.123	5.483	16.543	8.746	4.828	6.28	23.437
	2	0.027	0.025	0.022	0.075	0.049	0.035	0.019	0.023	0.043	0.114	...

Q1 := Basic_Pareto(q, 300, 2, -10) - Мінімізація при $par < 0$

	1	2	3	4	5	6	7	8	9	10	
Q1 =	1	0.359	4.533	7.141	14.991	28.222	20.165	21.732	35.771	44.843	34.276
	2	0.027	0.025	0.02	0.017	0.014	0.016	0.014	0.012	0.011	...

Num1 := Num_Pareto(q, 300, 2, -10) - номери точок Парето - фронту

Num1 ^T =	1	2	3	4	5	6	7	8	9	10	
	1	1	2	18	29	40	52	56	64	84	...

Q1_{1,3} = 7.141 Q1_{2,3} = 0.02

Num1₄ = 29 last(Num1) = 31

- ілюстрація роботи з елементами масивів

Q2 := Basic_Pareto(q, 300, 2, 10) - максимізація при $par > 0$

	1	2	3	4	5	6	
Q2 =	1	8.339	6.28	17.064	18.453	25.578	23.425
	2	0.075	0.114	0.051	0.051	0.037	...

Num2 := Num_Pareto(q, 300, 2, 10) - номери точок Парето - фронту

Num2 ^T =	1	2	3	4	5	6	7	8	9	10	
	1	4	10	20	23	25	35	65	66	67	...

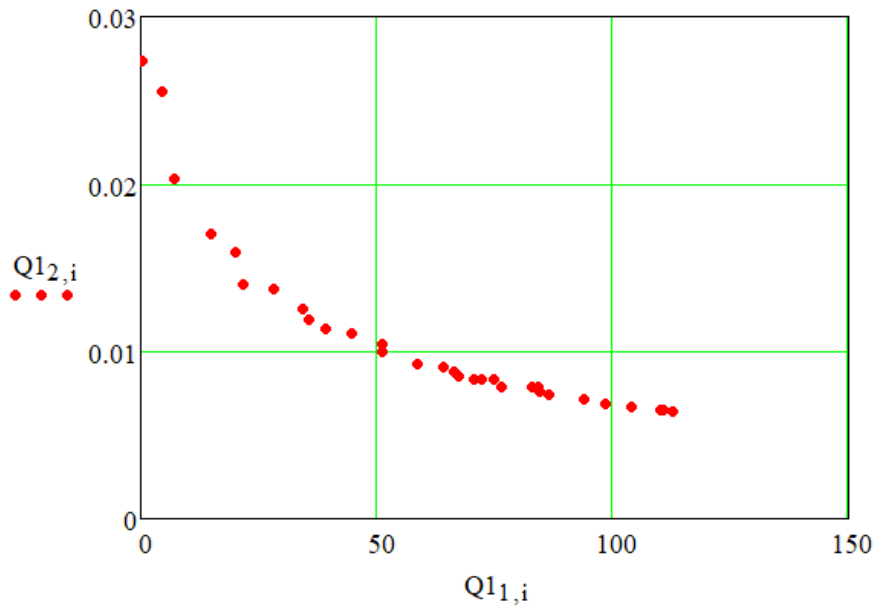
Q2_{1,3} = 17.064 Q2_{2,3} = 0.051

Num2₆ = 35 last(Num2) = 35

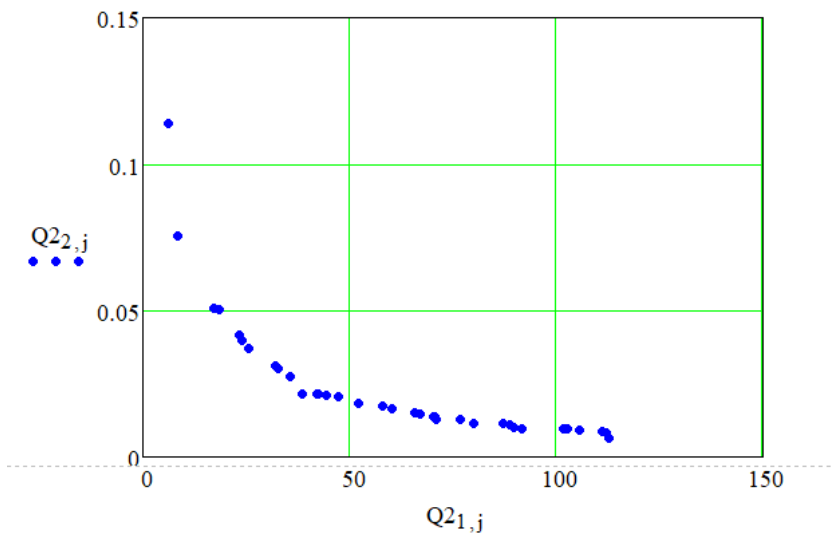
- ілюстрація роботи з елементами масивів

Компромісні криві (для випадку двох критеріїв)

last(Num1) = 31 i := 1 .. last(Num1) - мінімізація



$\text{last}(\text{Num2}) = 35 \quad j := 1 \dots \text{last}(\text{Num2}) \quad - \text{максимізація}$



- відповідний варіант вибирається візуально за графіками Парето - фронту

Розглянемо функцію $P_Mech()$, що дає змогу згенерувати якісну мережу точок, побудовану на основі Π_τ -мережі. Генератор дає змогу отримувати сітки з N точок у n -мірному кубі, причому граничні значення для цих параметрів відповідають $n=8$ та $N=1023$. За необхідності граничні значення можуть бути змінені. Для цих цілей можна скористатися значеннями, наведеними в табл.1.1. Приклад роботи генератора Π_τ -сітки для $n=3$ та $N=1000$ наведено нижче.

Приклад виконання завдання в MathCAD

ГЕНЕРАТОР П τ - МЕРЕЖІ (8 елементів у матриці параметрів;
1023 - кількість варіантів параметрів)

ORIGIN := 1

$$NP := \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 3 & 5 & 15 & 17 & 51 & 85 & 255 & 257 & 771 \\ 1 & 1 & 7 & 11 & 13 & 61 & 67 & 79 & 465 & 721 \\ 1 & 3 & 7 & 5 & 7 & 43 & 49 & 147 & 439 & 1013 \\ 1 & 1 & 5 & 3 & 15 & 51 & 125 & 141 & 177 & 759 \\ 1 & 3 & 1 & 1 & 9 & 59 & 25 & 89 & 321 & 835 \\ 1 & 1 & 3 & 7 & 31 & 47 & 109 & 173 & 181 & 949 \\ 1 & 3 & 3 & 9 & 9 & 57 & 43 & 43 & 225 & 113 \end{pmatrix} \quad \begin{array}{l} \text{- матриця} \\ \text{чисел} \end{array} \quad \begin{array}{l} \text{напрямних} \\ \text{чисел} \end{array}$$

$d(f) := f - \text{floor}(f)$

$m(i) := 1 + \text{floor}\left(\frac{\ln(i)}{\ln(2)}\right)$ - допоміжні функції

$$q(i,j) := \sum_{k=1}^{m(i)} \left[2^{-k+1} \cdot d \left[0.5 \cdot \sum_{s=k}^{m(i)} \left(\text{floor}(2 \cdot d(i \cdot 2^{-s})) \cdot \text{floor}(2 \cdot d(NP_{j,s} \cdot 2^{k-1-s})) \right) \right] \right]$$

Генератор мережі - функція P_Mesh(n,N) дозволяє сформувати мережу з N точок в n - розмірному кубі простору пошуку.

Результат роботи функції:

q - масив розмірністю N x n координат точок мережі, рівномірно розподілених в області пошуку.

```
P_Mesh(n,N) :=
  EN ← if(ORIGIN = 0, N - 1, N)
  En ← if(ORIGIN = 0, n - 1, n)
  for i ∈ ORIGIN..EN
    for j ∈ ORIGIN..En
      mas_pointi,j ← q(i,j)
  mas_point
```

шпрр

$q := P_Mesh(3, 1000)$

- виклик функції генератора мережі з $n=3$ та $N=1000$

	1	2	3
1	0.5	0.5	0.5
2	0.25	0.75	0.25
3	0.75	0.25	0.75
4	0.125	0.625	0.875
5	0.625	0.125	0.375
6	0.375	0.375	0.625
7	0.875	0.875	0.125
8	0.063	0.938	0.688
9	0.563	0.438	0.188
10	0.313	0.188	0.938
11	0.813	0.688	0.438
12	0.188	0.313	0.313
13	0.688	0.813	0.813
14	0.438	0.563	0.063
15	0.938	0.063	0.563
16	0.031	0.531	0.406
17	0.531	0.031	0.906
18	0.281	0.281	0.156
19	0.781	0.781	0.656
20	0.156	0.156	0.531
21	0.656	0.656	0.031
22	0.406	0.906	0.781
23	0.906	0.406	0.281
24	0.094	0.469	0.844
25	0.594	0.969	...

- згенерована мережа точок

Приклад 6. Продемонструємо всі етапи розв'язання оптимізаційної задачі в разі наявності трьох критеріїв. Оптимізована система характеризується критеріями:

$$F_1 = [4 \cdot x + 5 \cdot y - 2 \cdot z - 35]^2; \quad F_2 = [11 \cdot x + 6 \cdot y - 7 \cdot z - 8]^2;$$

$$F_3 = [9 \cdot x + 3 \cdot y + 4 \cdot z - 11]^2$$

Кожен із критеріїв є функцією трьох параметрів x, y, z оптимізованої системи за таких обмежень: $x \in [0; 5], y \in [0; 5], z \in [-10; 0]$.

Необхідно визначити розв'язок (x, y, z) , який мінімізує всі три критерії одночасно.

Порядок виконання завдання:

1. Задати критерії якості (цільової функції).
2. Використовувати раніше задані процедури для генерації P_τ - мережі.
3. Виконати корекцію мережі з урахуванням обмежень на простір параметрів.
4. Обчислити масив критеріїв якості, що відповідає скоригованому масиву параметрів.
5. Виділити точки Парето-фронту, використовуючи процедури.

$Cmp_1(P, Q, i, n, par), Cmp_Pareto(P, Q, n, par), Basic_Pareto(Q, N, n, par),$
 $Num_Pareto(Q, N, n, par)$

6. Визначити номери точок Парето-фронту, використовуючи процедуру $Num_Pareto(Q, N, n, par)$

7. Побудувати графіки точок Парето-фронту для всіх критеріїв як функціональної залежності кожного з критеріїв від номера точок невідомінованих розв'язків.

8. Візуальний вибір раціонального розв'язку виконати, ґрунтуючись на наявному графічному матеріалі.

Приклад виконання завдання в MathCAD

Приклад 6. Продемонструємо всі етапи розв'язання оптимізаційної задачі в разі наявності трьох критеріїв. Оптимізована система характеризується критеріями:

$$F_1 = [4 \cdot x + 5 \cdot y - 2 \cdot z - 35]^2; \quad F_2 = [11 \cdot x + 6 \cdot y - 7 \cdot z - 8]^2;$$

$$F_3 = [9 \cdot x + 3 \cdot y + 4 \cdot z - 11]^2$$

Кожен з критеріїв є функцією трьох параметрів x, y, z оптимізованої системи при наступних обмеженнях: $x \in [0; 5], y \in [0; 5], z \in [-10; 0]$.

Необхідно визначити розв'язок (x, y, z) , який мінімізує всі три критерії одночасно.

АПРОКСИМАЦІЯ ПАРЕТО - ФРОНТУ ЗАДАЧІ ВЕКТОРНОЇ ОПТИМІЗАЦІЇ З ТРЬОМА КРИТЕРІЯМИ З ЗАСТОСУВАННЯМ Пт - МЕРЕЖІ РІВНОМІРНО РОЗПОДІЛЕНИХ ТОЧОК

ORIGIN := 1

Функція $Str_1(P, Q, i, n, par)$, яка здійснює операцію порівняння точки P з i -ю точкою множини можливих розв'язків, що зберігається в масиві Q розмірності n .

$par < 0$ задається при мінімізації;

$par > 0$ задається при максимізації.

В результаті функція повертає:

-1, якщо точки збігаються;

0, якщо точка P гірша;

1, якщо точка P не гірша.

```

Cmp_1(P, Q, j, n, par) :=
  for k ∈ 1..n
    Lk ← if(Pk = Qk,j, 100, -100)
  Rez_1 ← ∑L
  return -1 if Rez_1 = n·100
  for k ∈ 1..n
    | Lk ← if(Pk ≥ Qk,j, 0, 1) if par < 0
    | Lk ← if(Pk ≤ Qk,j, 0, 1) otherwise
  Rez_1 ← ∑L
  return 0 if Rez_1 = 0
  return -1 otherwise

```

Функція Cmp_Pareto(P, Q, n, N, par), яка визначає належність точки **P** з масиву **Q** розмірності **n** x **N** Парето - фронту недомінованих розв'язків. В результаті функція повертає:
 10, якщо точка **P** належить до Парето-фронту;
 -10, якщо точка **P** не належить.

```

Cmp_Pareto(P, Q, n, N, par) :=
  j ← 1
  while j ≤ N
    | return -10 if Cmp_1(P, Q, j, n, par) = 0
    | j ← j + 1
  rez ← 10

```

Функція Basic_Pareto(Q, N, n, par), яка виділяє ефективні точки з **N** точок масиву критеріїв **Q** розмірності **n** x **N**, тобто недомінуючі розв'язки.
n - розмірність простору критеріїв якості;
N - кількість варіантів параметрів.

```

Basic_Pareto(Q, N, n, par) :=
  i ← 1
  for j ∈ 1..N
    | P ← Q(j)
    | D(i) ← P if Cmp_Pareto(P, Q, n, N, par) = 10
    | i ← i + 1 if Cmp_Pareto(P, Q, n, N, par) = 10
  D

```

Функція Num_Pareto(Q, N, n, par), яка повертає вектор номерів паретових точок з **N** точок масиву критеріїв **Q** розмірності **n** x **N**.
n - розмірність простору критеріїв якості;
N - кількість варіантів параметрів.

```

Num_Pareto(Q, N, n, par) := | i ← 1
                           | for j ∈ 1..N - 1
                           |   | P ← Q<sup>j</sup>
                           |   | Numi ← j if Cmp_Pareto(P, Q, n, N, par) = 10
                           |   | i ← i + 1 if Cmp_Pareto(P, Q, n, N, par) = 10
                           | Num

```

ГЕНЕРАТОР П τ - МЕРЕЖІ (8 елементів у матриці параметрів; 1023 - кількість варіантів параметрів)

NP :=
$$\begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 3 & 5 & 15 & 17 & 51 & 85 & 255 & 257 & 771 \\ 1 & 1 & 7 & 11 & 13 & 61 & 67 & 79 & 465 & 721 \\ 1 & 3 & 7 & 5 & 7 & 43 & 49 & 147 & 439 & 1013 \\ 1 & 1 & 5 & 3 & 15 & 51 & 125 & 141 & 177 & 759 \\ 1 & 3 & 1 & 1 & 9 & 59 & 25 & 89 & 321 & 835 \\ 1 & 1 & 3 & 7 & 31 & 47 & 109 & 173 & 181 & 949 \\ 1 & 3 & 3 & 9 & 9 & 57 & 43 & 43 & 225 & 113 \end{pmatrix}$$
 - матриця чисельників напрямних чисел

$d(f) := f - \text{floor}(f)$ $m(i) := 1 + \text{floor}\left(\frac{\ln(i)}{\ln(2)}\right)$ - допоміжні функції

$$q(i, j) := \sum_{k=1}^{m(i)} \left[2^{-k+1} \cdot d \left[0.5 \cdot \sum_{s=k}^{m(i)} \left(\text{floor}(2 \cdot d(i \cdot 2^{-s})) \cdot \text{floor}(2 \cdot d(NP_{j,s} \cdot 2^{k-1-s})) \right) \right] \right]$$

Генератор мережі - функція P_Mesh(n,N) дозволяє сформувати мережу з N точок в n - розмірному кубі простору пошуку.

Результат роботи функції:

q - масив розмірністю N x n координат точок мережі, рівномірно розподілених в області пошуку.

```

P_Mesh(n, N) := | EN ← if(ORIGIN = 0, N - 1, N)
                 | En ← if(ORIGIN = 0, n - 1, n)
                 | for i ∈ ORIGIN..EN
                 |   for j ∈ ORIGIN..En
                 |     mas_pointi,j ← q(i, j)
                 | mas_point

```

$q := P_Mesh(3, 1000)$

- виклик функції генератора мережі

	1	2	3
1	0.5	0.5	0.5
2	0.25	0.75	0.25
3	0.75	0.25	0.75
4	0.125	0.625	0.875
5	0.625	0.125	0.375
6	0.375	0.375	0.625
7	0.875	0.875	0.125
8	0.063	0.938	0.688
9	0.563	0.438	0.188
10	0.313	0.188	0.938
11	0.813	0.688	0.438
12	0.188	0.313	0.313
13	0.688	0.813	0.813
14	0.438	0.563	0.063
15	0.938	0.063	0.563
16	0.031	0.531	...

$q =$

- згенерована мережа точок

$$Fkr_1(x, y, z) := (4 \cdot x + 5 \cdot y - 2 \cdot z - 35)^2$$

$$Fkr_2(x, y, z) := (11 \cdot x + 6 \cdot y + 7 \cdot z - 8)^2$$

- задані критерії якості (цільової функції)

$$Fkr_3(x, y, z) := (9 \cdot x + 3 \cdot y + 4 \cdot z - 11)^2$$

$i := 1..1000$

$$q_kr_{i,1} := q_{i,1} \cdot (5 - 0) + 0$$

$$q_kr_{i,2} := q_{i,2} \cdot (5 - 0) + 0$$

$$q_kr_{i,3} := q_{i,3} \cdot (0 + 10) - 10$$

- корекція мережі з урахуванням обмежень на простір параметрів:

$x(0; 5)$

$y(0; 5)$

$z(-10; 0)$

$q_kr^{(1)} =$

	1
1	2.5
2	1.25
3	3.75
4	0.625
5	3.125
6	1.875
7	4.375
8	0.313
9	2.813
10	1.563
11	4.063
12	0.938
13	3.438
14	2.188
15	4.688
16	...

$q_kr^{(2)} =$

	1
1	2.5
2	3.75
3	1.25
4	3.125
5	0.625
6	1.875
7	4.375
8	4.688
9	2.188
10	0.938
11	3.438
12	1.563
13	4.063
14	2.813
15	0.313
16	...

$q_kr^{(3)} =$

	1
1	-5
2	-7.5
3	-2.5
4	-1.25
5	-6.25
6	-3.75
7	-8.75
8	-3.125
9	-8.125
10	-0.625
11	-5.625
12	-6.875
13	-1.875
14	-9.375
15	-4.375
16	...

$M_{kr1,i} := Fkr_1(q_{kr1,i}, q_{kr1,2}, q_{kr1,3})$ - обчислення масиву критеріїв
 $M_{kr2,i} := Fkr_2(q_{kr1,i}, q_{kr1,2}, q_{kr1,3})$ якості відповідного
 $M_{kr3,i} := Fkr_3(q_{kr1,i}, q_{kr1,2}, q_{kr1,3})$ скоригованому масиву параметрів
 $Front := Basic_Pareto(M_kr, 1000, 3, -10)$ - виділення точок Парето - фронту,
 (недомінуючих роз'язків)

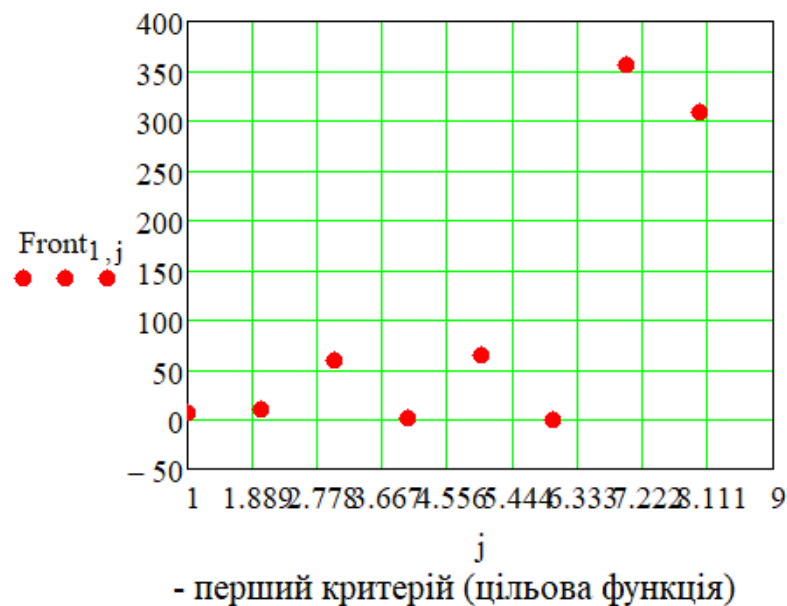
$$Front^T = \begin{pmatrix} 6.25 & 0.25 & 1 \\ 9.047 & 0.505 & 3.906 \times 10^{-3} \\ 58.918 & 0.017 & 0.699 \\ 0.145 & 3.208 \times 10^{-3} & 6.973 \\ 64.674 & 5.045 \times 10^{-4} & 0.762 \\ 2.384 \times 10^{-5} & 0.7 & 8.427 \times 10^{-3} \\ 356.524 & 8.583 \times 10^{-6} & 0.927 \\ 308.133 & 0.83 & 2.018 \times 10^{-3} \end{pmatrix}$$

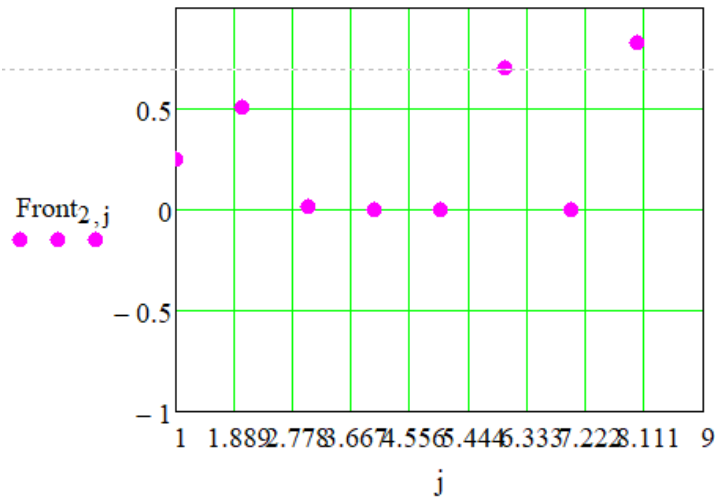
$NUM := Num_Pareto(M_kr, 1000, 3, -10)$ - визначення номерів точок
 Парето - фронту

$$NUM^T = (1 \ 81 \ 169 \ 454 \ 529 \ 553 \ 653 \ 814)$$

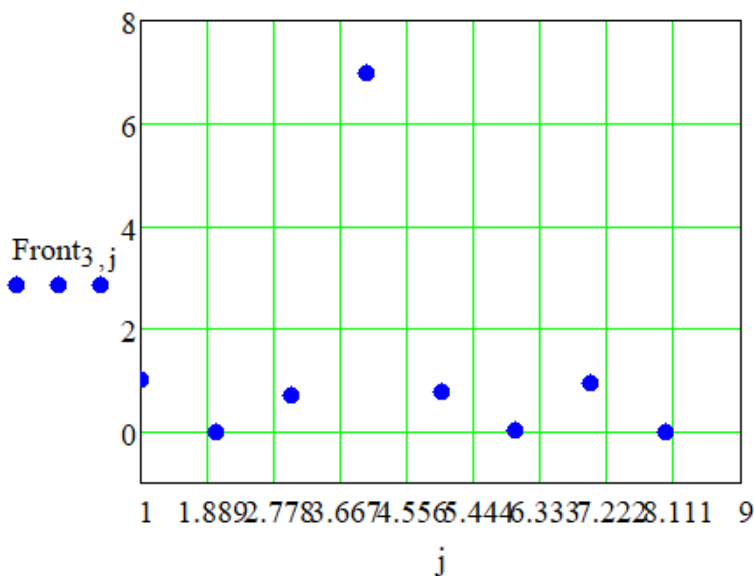
$K_point := last(NUM)$ - визначення кількості точок

$$K_point = 8 \quad j := 1 .. K_point$$





- другий критерій (цільова функція)



- третій критерій (цільова функція)

У разі кількості критеріїв більше трьох має сенс побудова сімейства графіків, що відображають функціональну залежність кожного з критеріїв від номерів точок невідоміючих розв'язків.

При чітко вертикальному розташуванні графіків один під одним вдається на одній вертикалі візуалізувати вектори невідоміючих рішень. Вибір найбільш прийняттого з них здійснюється ОПР (особою, яка приймає рішення) на основі суб'єктивних суджень, керуючись раціональними міркуваннями.

nom := NUM₅ nom = 529

- вибір варіанту вектора невідоміючого розв'язку при вирішенні задачі мінімізації, тобто на основі обраної точки з номером 5

$$q_{kr_{nom},1} = 2.661$$

$$q_{kr_{nom},2} = 3.921$$

$$q_{kr_{nom},3} = -6.396$$

$$M_{kr_1,nom} = 64.674$$

$$M_{kr_2,nom} = 5.045 \times 10^{-4}$$

$$M_{kr_3,nom} = 0.762$$

- знайдені значення координат точки екстремуму $\mathbf{P}(x, y, z)$ за Парето

- значення, що приймаються критеріями якості в точці компромісу \mathbf{P}



ЗАВДАННЯ ДО САМОСТІЙНОЇ РОБОТИ

Завдання 2. Розв'язати задачі у виділення фронту Парето, використовуючи варіанти завдань 1-12 з попереднього пункту.

СПИСОК ЛІТЕРАТУРИ

1. Anh, L. Q., Thuy, V. T. M., & Zhao, X. (2025). Qualitative Properties of Robust Benson Efficient Solutions of Uncertain Vector Optimization Problems. *Journal of Optimization Theory and Applications*, 205(1), 1-37.
2. Rahebi, J. (2022). Vector quantization using whale optimization algorithm for digital image compression. *Multimedia Tools and Applications*, 81(14), 20077-20103.
3. Гіріна, О., & Івченко, В. (2023). Багатокритеріальна оптимізація структури портфеля реальних інвестицій. *Економіка та суспільство*, (50).
4. Малік, Т. В., Ковальов, Ю. М., Калашнікова, В. В., & Нерушева, В. М. (2023). Багатокритеріальна оптимізація дизайну цивільних укриттів в Україні з урахуванням ізраїльського досвіду. *Art and Design*.
5. Ivanchenko, H., Koshevyi, O., & Zatyliuk, G. (2024). Багатокритеріальна параметрична оптимізація переміщення і ваги оболонки мінімальної поверхні на круглому контурі, що складається із двох похилих еліпсів, при термосиловому навантаженні з урахуванням геометричної нелінійності. *Strength of Materials and Theory of Structures*, (112), 209-221.
6. Тупало, Я. О. (2021). Багатокритеріальна оптимізація при проектуванні систем з високою завантаженістю. *Кібернетика та комп'ютерні технології*, (4), 43-50.
7. Дмитренко, С., & Барандич, К. (2025). Багатокритеріальна оптимізація розміщення деталей в умовах автоматизованого складу. *Вісник Київського політехнічного інституту. Серія Приладобудування*, (69 (1)), 45-51.
8. Koshevyi, O. O., & Kosheva, I. S. (2022). Багатокритеріальна параметрична оптимізація в парі цільових функцій: вага і переміщення оболонки мінімальної поверхні на прямокутному контурі при термосиловому навантаженні. *Ways to Improve Construction Efficiency*, 1(49), 66-78.
9. Гадецька, С. В., Дубницький, В. Ю., Кушнерук, Ю. І., Ходирєв, О. І., & Шкодїна, І. В. (2021). Багатокритеріальна (векторна) оптимізація портфеля валют при нестохастично невизначеному зовнішньому економічному середовищі. *Системи обробки інформації*, (3 (166)), 6-21.
10. Lárraga, G., & Miettinen, K. (2025). Survey of interactive evolutionary decomposition-based multiobjective optimization methods. *Evolutionary Computation*, 1-39.
11. Ji, J., Zhang, X., Yang, C., Li, X., & Sui, G. (2025). A similar environment transfer strategy for dynamic multiobjective optimization. *Information Sciences*, 707, 122018.
12. Yu, K., Yang, Z., Liang, J., Qiao, K., Qu, B., & Suganthan, P. N. (2025). An individual adaptive evolution and regional collaboration based

evolutionary algorithm for large-scale constrained multiobjective optimization problems. *Swarm and Evolutionary Computation*, 95, 101925.

13. Liang, Z., Li, H., Yu, N., Sun, K., & Cheng, R. (2025). Bridging evolutionary multiobjective optimization and GPU acceleration via tensorization. *IEEE Transactions on Evolutionary Computation*.

14. Shao, S., Tian, Y., Zhang, Y., & Zhang, X. (2025). Knowledge learning-based dimensionality reduction for solving large-scale sparse multiobjective optimization problems. *IEEE Transactions on Cybernetics*.

15. Deng, Y., Zheng, Y., Meng, Z., Gao, H., Hua, Y., Jin, Q., & Cao, L. (2025). Gaussian process regression for evolutionary dynamic multiobjective optimization in complex environments. *Swarm and Evolutionary Computation*, 94, 101883.

16. Sankar, A., Gupta, K. K., Bhalla, V., & Pandey, D. S. (2025). Multi-criteria optimization of nanofluid-based solar collector for enhanced performance: An explainable machine learning-driven approach. *Energy*, 320, 135212.

17. Yan, L., Guo, S., Liang, J., Qu, B., Li, C., & Yu, K. (2025). A subspace strategy based coevolutionary framework for constrained multimodal multiobjective optimization problems. *Swarm and Evolutionary Computation*, 95, 101941.

18. Li, N., Xue, B., Ma, L., & Zhang, M. (2025). Automatic fuzzy architecture design for defect detection via classifier-assisted multiobjective optimization approach. *IEEE Transactions on Evolutionary Computation*.

19. Bade, S. O., Tomomewo, O. S., Meenakshisundaram, A., Dey, M., Alamooti, M., & Halwany, N. (2025). Multi-Criteria Optimization of a Hybrid Renewable Energy System Using Particle Swarm Optimization for Optimal Sizing and Performance Evaluation. *Clean Technologies*, 7(1).

20. Qiu, W., Yang, Y., Song, J., Que, W., Liu, Z., Weng, H., ... & Wu, J. (2025). A deep-learning-based multiobjective optimization for the design of in-situ uranium leaching system under multiple uncertainties. *Journal of Hydrology*, 651, 132576.

21. Zhang, P., Zhang, R., Tian, Y., Tan, K. C., & Zhang, X. (2025). A dual model-based evolutionary framework for dynamic large-scale sparse multiobjective optimization. *Swarm and Evolutionary Computation*, 97, 102011.

22. Ekstrom, L., Wang, H., & Schmitt, S. (2025). Variational quantum multiobjective optimization. *Physical Review Research*, 7(2), 023141.

23. Ekel, P. I., Libório, M. P., & Pedrycz, W. (2025). Book Review: Multicriteria Decision-Making Under Conditions Of Uncertainty: A Fuzzy Set Perspective. John Wiley & Sons. ISBN: 978-1-119-53,492-1. *International Journal of Computational Intelligence Systems*, 18(1), 1-5.

24. Huang, X., Wang, J., Zhang, K., Yuan, B., Dai, C., & Ablameyko, S. V. (2025). An Enhanced Sparse Multiobjective Evolutionary Algorithm in Large-Scale Multiobjective Optimization. *Information Sciences*, 122476.

25. Phan Duc, H., Bui Trong, D., Nguyen Thi, T., & Huynh Thi Thanh, B. (2025, July). Pareto Front Grid Guided Multiobjective Optimization In Dynamic Pickup And Delivery Problem Considering Two-Sided Fairness. In *Proceedings of the Genetic and Evolutionary Computation Conference* (pp. 277-285).
26. Deng, Y., Gao, H., Zheng, Y., Meng, Z., Hua, Y., Jin, Q., & Cao, L. (2025). Nearest Neighbor Regression for Evolutionary Dynamic Multiobjective Optimization. *Information Sciences*, 122513.
27. Deng, Q., Zou, J., Yang, S., Liu, Y., Yu, F., Xie, T., & Zheng, J. (2025). A niching-based nondominated sorting for multimodal multiobjective optimization with local pareto fronts. *Applied Soft Computing*, 113223.
28. Hu, Y., Qu, Y., Li, W., & Huang, Y. (2025). A Pareto Front searching algorithm based on reinforcement learning for constrained multiobjective optimization. *Information Sciences*, 705, 121985.
29. Yan, L., Guo, S., Liang, J., Qu, B., Li, C., & Yu, K. (2025). A subspace strategy based coevolutionary framework for constrained multimodal multiobjective optimization problems. *Swarm and Evolutionary Computation*, 95, 101941.
30. Ji, J., Zhang, X., Yang, C., Li, X., & Sui, G. (2025). A similar environment transfer strategy for dynamic multiobjective optimization. *Information Sciences*, 707, 122018.
31. Martins, P. H., & Santos, A. A. (2025). Metamodeling for robust design of energy harvesting devices using multiobjective optimizations. *International Journal of Mechanics and Materials in Design*, 1-21.
32. Wang, X., Kang, Q., Zhou, M., Deng, Q., Fan, Z., & Liu, H. (2025). Knowledge classification-assisted evolutionary multitasking for two-task multiobjective optimization problems. *IEEE/CAA Journal of Automatica Sinica*, 12(6), 1176-1193.
33. Laha, V., Dwivedi, A., & Jaiswal, P. (2025). On quasidifferentiable interval-valued multiobjective optimization. *Annals of Operations Research*, 1-31.
34. de Castro, P. J. (2025). Decomposition and Coordination for Multiobjective Optimization: A Framework and Methodology.
35. Shams Taleghani, A., & Izadi, M. (2025). Multiobjective Optimization of a Single Slotted Flap Using Artificial Neural Network and Metaheuristic Algorithms. *Journal of Engineering Mechanics*, 151(8), 05025001.
36. Song, W., Liu, Z., Yu, J., Sun, X., Jin, Y., & Lai, K. W. (2025). Multisource and Hidden Source-Based Knowledge Transfer for Solving Dynamic Multiobjective Optimization Problems. *IEEE Transactions on Evolutionary Computation*.
37. Ma, R., Lu, Y., Yu, X., & Yang, B. (2025). Thermodynamic, Economic, and Environmental Multi-Criteria Optimization of a Multi-Stage Rankine System for LNG Cold Energy Utilization. *Modelling*, 6(2), 45.
38. Zhang, Y., Huang, J., Giacomini, A., Xie, J., & Lu, J. (2025). Robust calibration of shaft and base resistance factors for piles based on multiobjective

optimization. *Journal of Geotechnical and Geoenvironmental Engineering*, 151(3), 04024169.

ТЕРМІНИ ТА ВИЗНАЧЕННЯ

Багатокритеріальні задачі: оптимізаційні задачі, де замість однієї цільової функції розглядається вектор критеріїв, кожен з яких відображає окремий аспект оцінки розв'язок.

Векторний критерій: сукупність частинних критеріїв, що формують вектор оцінки для кожного розв'язок.

Допустима область (D): множина всіх можливих розв'язків, які задовольняють заданим обмеженням.

Критеріальний простір (Y_D): область, що містить всі можливі векторні оцінки розв'язків із допустимої області.

Парето-оптимальність: розв'язок вважається оптимальним за Парето, якщо не існує іншого розв'язку, який покращує хоча б один критерій без погіршення інших. Векторна (багатокритеріальна) оптимізація — це процес одночасної оптимізації двох або більше конфліктуючих цільових функцій $f_1(x), f_2(x), \dots, f_k(x)$ на множині допустимих розв'язків S , де $x \in S$ — вектор змінних.

Цільова функція — функція, значення якої потрібно мінімізувати або максимізувати. У векторній оптимізації таких функцій кілька, і вони формують вектор цільових функцій $f(x) = (f_1(x), \dots, f_k(x))$.

Область допустимих розв'язків (область визначення) — множина S , у границях якої шукаються розв'язки задачі оптимізації.

Оптимальний розв'язок за Парето (Парето-оптимальний розв'язок) — вектор розв'язку $x^* \in S$, для якого не існує іншого $x \in S$, що покращує всі цільові функції одночасно, тобто неможливо знайти x , для якого $f_i(x) \leq f_i(x^*)$ для всіх i і при цьому $f_j(x) < f_j(x^*)$ хоча б для одного j .

Парето-фронт (множина Парето-оптимальних розв'язків) — множина всіх Парето-оптимальних розв'язків, яка відображає компроміс між різними критеріями.

Слабкий оптимум за Парето — розв'язок, для якого не існує іншого, що покращує всі цільові функції строго, але може існувати розв'язок, який покращує деякі з них без погіршення інших.

Упорядкування за Парето — частковий порядок у просторі цільових функцій, що визначає, коли один вектор цілей є кращим або не гіршим за інший.

Критеріальний конус — множина напрямків, що визначають допустимі напрямки покращення цільових функцій у векторній оптимізації.

Лексикографічний порядок — спосіб впорядкування векторів цільових функцій, коли критерії ранжуються за пріоритетністю, і порівняння відбувається послідовно за кожним критерієм.

Ефективна точка — точка, яка є оптимальною за Парето, тобто не може бути покращена за жодним критерієм без погіршення іншого.