Київський національний університет імені Тараса Шевченка
Факультет ком'ютерних наук та кібернетики
Кафедра моделювання складних систем

ВИПУСКНА КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА
на здобуття ступеня бакалавра
за спеціальністю 113 «Прикладна математика»
на тему:

Дослідження моделі поведінки фірми в умовах нечітко заданих цін

студента 4 курсу
Коржова Максима Олександровича

____

Науковий керівник:
доцент, кандидат фізико-математичних наук
Коробова М.В.

____

Робота заслухана на засіданні кафедри моделювання складних систем та рекомендована до захисту в ДЕК, протокол №   від   2021 р.

Завідувач кафедри моделювання складних систем

доцент Черній Д.І.

Київ – 2021

ЗМІСТ

ВСТУП

Теорія нечітких множин відіграє досить важливу роль у сучасному житті. Її використовують майже всюди: починаючи від «розумних» пральних машин, і закінчуючи безпілотними автомобілями. Звичайно, нечіткі множини зустрічаються не тільки в побуті, а й у різних галузях науки та економіки.

Наприклад, у часи ринкової економіки ледве не кожен підприємець хоче максимально збільшити прибуток своєї фірми. Тому, ставши директором компанії, що виробляє певний продукт, рано чи пізно кожен стикнеться з потребою у закупівлі сировини, але зразу виникає питання: «Яку сировину потрібно закупити, щоб отримати найбільший прибуток?». У такій ситуації є можливість запропонувати саме задачу оптимізації прибутку однопродуктової фірми при нечітко заданих цінах на ресурси, яка допоможе мінімізувати витрати на сировину і виробництво, а також промоделює поведінку економіки на майбутнє.

Однак на практиці часто застосовують задачу мінімізації виробничих видатків при заданих обсягах випуску продукції, яка дозволяє скоректувати витрати на вже існуючому і налагодженому виробництві.

Метою дипломної роботи є дослідження методів щодо налагодження ефективного виробництва на однопродуктовому підприємстві.

Для досягнення поставленої мети розглядається задача оптимізації прибутку однопродуктової фірми та задача мінімізації виробничих видатків при заданих обсягах випуску продукції для випадку нечітко заданих цін на ресурси із застосуванням теорії нечітких множин та методу послідовних поступок з подальшим пошуком компромісного розв’язку.

РОЗДІЛ 1
ТЕОРІЯ І МЕТОДИ РОЗВ'ЯЗАННЯ ЗАДАЧ

У теорії виробництва мікроекономічної теорії є три основних задачі: неокласична задача поведінки однопродуктової фірми, задача мінімізації витрат і максимізації випуску продукції. Основою з цих задач є модель поведінки однопродуктової фірми за певний проміжок часу, котра полягає у максимізації прибутку (неокласична задача) при заданій виробничій функції, заданих цінах на продукцію та цінах на фактори виробництва.

Під час побудови таких моделей розглядаються різноманітні задачі, які потребують знання деяких означень, теорем та вміння розв'язувати певні типи задач відповідними методами. Під час розв'язання поставленої задачі максимізації прибутку однопродуктової фірми та задачі мінімізації виробничих видатків будемо використовувати апарат теорії нечітких множин та методи розв'язування задач багатокритеріальної оптимізації.

1.1. Нечіткі множини і функції належності

У математиці нечіткі множини дещо схожі на множини, елементи яких мають ступінь належності. Нечіткі відносини, які зараз використовуються в різних областях, таких як лінгвістика, прийняття рішень та кластеризація, є особливими випадками L-відносин, коли L є одиничний інтервал $[0, 1]$.

У класичній теорії множин приналежність елементів до множини оцінюється у бінарних умовах відповідно до двовалентної умови – елемент або належить, або не належить до множини. Навпаки, нечітка теорія множин дозволяє поступово оцінювати належність елементів до множини: це описується за допомогою функції належності, оціненої в реальному одиничному інтервалі $[0, 1]$. Нечіткі множини узагальнюють класичні множини, оскільки функції індикаторів (також характерні функції) класичних множин є особливими випадками функцій належності нечітких множин, якщо останні приймають лише значення 0 або 1. У теорії нечітких множин класичні двовалентні множини зазвичай називають чіткими множинами.

Формально нечітка множина A визначається як множина впорядкованих пар або кортежів виду: $< x, \mu_A(x) >$, де x є елементом деякої універсальної множини або універсуму X, а $\mu_A(x)$ – функція належності, що ставить у відповідність кожному з елементів $x \in X$ деяке дійсне число з інтервалу $[0, 1]$, тобто ця функція визначається у формі відображення:

$$\mu_A(x): X \to [0,1].$$

При цьому значення $\mu_A(x) = 1$ для деякого $x \in X$ означає, що елемент x належить нечіткій множині A, а значення $\mu_A(x) = 0$ означає, що елемент x не належить нечіткій множині A.

Означення 1.1.1. Нечітка множина A, визначена на X, називається порожньою, якщо її функція належності дорівнює 0 на всій множині X, тобто $\mu_A(x) = 0, \forall x \in X$.

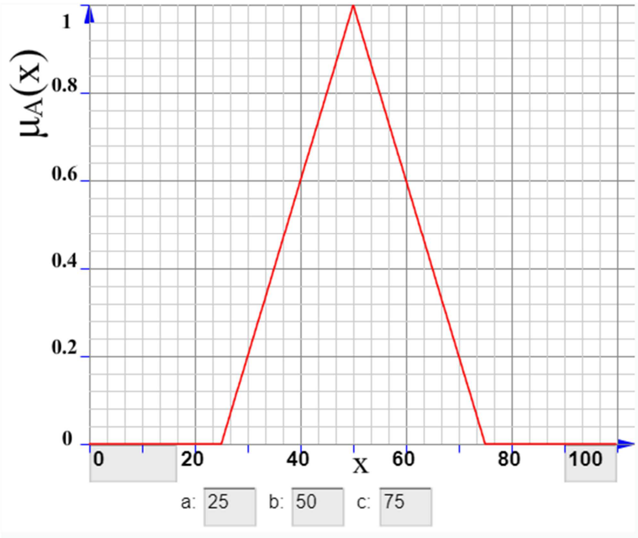
Означення 1.1.2. Універсальна множина X описується функцією належності вигляду $\mu_X(x) = 1, \forall x \in X$. [6]

Означення 1.1.3. Множиною рівня λ (λ-перерізом) нечіткої множини A називається нечітка підмножина універсальної множини X, яка визначається за формулою

$$A_\lambda = \{x \mid \mu_A(x) \geq \lambda\}, \text{ де } \lambda \in [0,1].$$

На практиці нечіткі множини часто описують у вигляді функцій належності. До основних типів функцій належності відносять ***кусково-лінійні*** *функції належності* та ***П-подібні*** *функції належності.*

До кусково-лінійних відносяться такі функції як трикутна та трапецієподібна функції належності (рис. 1).



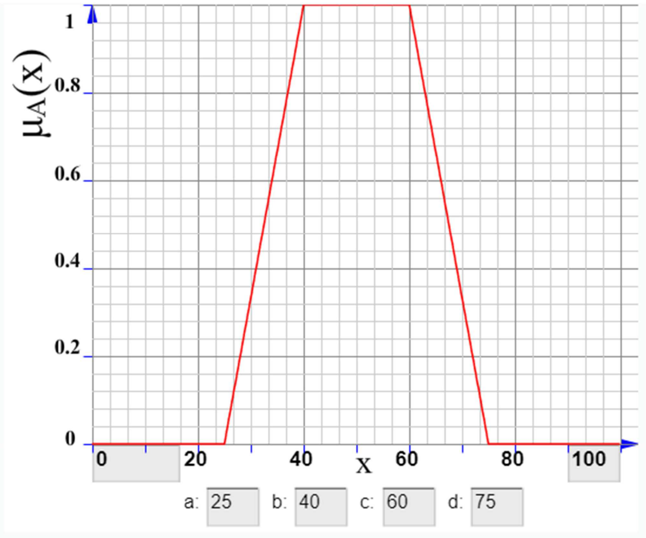


Рис. 1. Трикутна та трапецієподібна функції належності

Трикутна функція належності може бути аналітично задана виразом:

$$f_{\Delta}(x;a,b,c) = \begin{cases} 0, & x < a; \\ \dfrac{x-a}{b-a}, & a \le x < b; \\ \dfrac{c-x}{c-b}, & b \le x \le c; \\ 0, & x > c, \end{cases}$$

де $a,\ b,\ c$ – деякі числові параметри, що приймають довільні значення та упорядковані відношенням: $a \le b \le c$.

Трапецієподібна функція належності ж може бути аналітично задана виразом:

$$f_{\Delta}(x;a,b,c,d) = \begin{cases} 0, & x < a; \\ \dfrac{x-a}{b-a}, & a \le x < b; \\ 1, & b \le x < c; \\ \dfrac{d-x}{d-c}, & c \le x \le d; \\ 0, & x > d, \end{cases}$$

де $a,\ b,\ c,\ d$ – деякі числові параметри, що приймають довільні значення та упорядковані відношенням: $a \le b \le c \le d$.

До П-подібних функцій можна віднести цілий клас дзвоноподібних кривих. Можна розглянути основні види П-подібних функцій належності, але закцентуємо увагу на гаусівській функції (рис. 2).

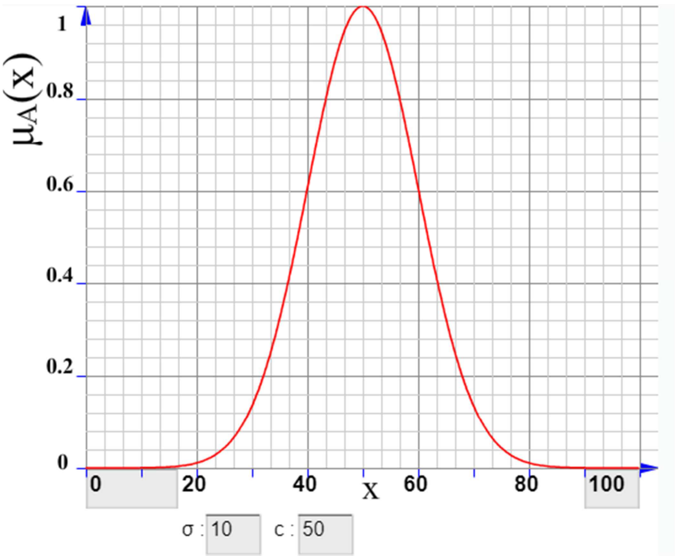


Рис. 2. Гаусівська функція належності

Формується вона з використанням гаусівської кривої:

$$f_{\Pi}(x;\sigma,c) = e^{-\frac{(x-c)^2}{2\sigma^2}},$$

де σ і c – числові параметри, що приймають довільні дійсні значення, причому $\sigma > 0$. [5]

1.2. Задача багатокритеріальної оптимізації

Задачею багатокритеріальної оптимізації називається така задача прийняття рішень, принцип оптимальності якої задається множиною критеріальних функцій. Розглянемо вимірну задачу багатокритеріальної оптимізації:

$$f(x) \to \max, \quad x \in X,$$

де X – це множина альтернатив, яка є множиною з простору E^n,
$f(x) = (f_i(x))_{i \in M}$ – вектор критеріїв, який задається відображенням $f : X \to E^m$,
$M = \{1,2,\dots,m\}$ – множина індексів критеріїв,
m – кількість критеріїв.

У залежності від структури множини X і властивостей функцій f_i виділяють різні класи багатокритеріальних задач. Так, якщо множина X містить скінченну кількість елементів, то задача називається скінченною, а якщо X є зліченною множиною, то – дискретною.

Зокрема, якщо в кожного вектора з X компоненти – цілі числа, то задача називається цілочисельною. А якщо вектори, які утворюють X, є булевими (тобто складаються з нулів та одиниць), то і сама задача називається булевою.

Якщо X опукла множина, а усі f_i – увігнуті функції, то задача називається увігнутою. Зокрема, якщо X – поліедральна множина (тобто «вирізана» з E^n кінцевою системою лінійних нерівностей і рівностей), а усі f_i – лінійні, то багатокритеріальна задача називається лінійною. [1]

Розв’язком задачі багатокритеріальної оптимізації є альтернатива, вибір якої потрібно робити з множини ефективних альтернатив залежно від вимог особи, що приймає рішення (ОПР) і предметної області, у якій приймається рішення. Якщо множина абсолютно-оптимальних альтернатив не є порожньою, то будь-яка з альтернатив може вважатися розв'язком багатокритеріальної задачі. Однак, на практиці зазвичай зустрічаються задачі, де множина абсолютно-оптимальних альтернатив є порожньою. Для розв'язання таких задач використовуються правила вибору ефективних альтернатив.

Так, правила вибору ефективної альтернативи мають певну чисельну реалізацію у вигляді методів багатокритеріальної оптимізації. Методи багатокритеріальної оптимізації за типами інформації, яку дає ОПР для формування правила вибору можна класифікувати так [1]:

- Методи, які не використовують інформацію про перевагу на множині критеріїв.
- Методи, які використовують один тип інформації про перевагу на множині критеріїв.
- Методи, які використовують різні типи інформації про перевагу на множині критеріїв.
- Спеціальні методи.

У даній роботі розглядається і застосовується один із таких методів – метод послідовних поступок. Він належить до класу методів, які використовують різні типи інформації про перевагу на множині критеріїв.

Метод послідовних поступок застосовується у випадку, коли критерії впорядковані за важливістю. Опишемо його для задачі виду:

$$f_i(x) \to \max, \ \ i = \overline{1,m}, \ \ x \in X,$$

$$f_1(x) \geq f_2(x) \geq \cdots \geq f_m(x).$$

Сутність методу послідовних поступок полягає в тому, що вихідна багатокритеріальна задача замінюється послідовністю однокритеріальних задач, область допустимих рішень яких звужується від задачі до задачі за допомогою додаткових обмежень, які враховують вимоги до критеріїв. При формулюванні кожної задачі, по відношенню до важливішого критерію робиться поступка, величина якої залежить від вимог задачі і оптимального рішення за цим критерієм. Таким чином, цей метод використовує інформацію від ОПР на кожному кроці його реалізації.

Опишемо схему метода:

1. Спочатку розв'язується задача оптимізації на всій множині допустимих альтернатив X:

$$f_1(x) \to \max, \quad x \in X.$$

У результаті отримуємо оптимальне значення критерію $f_1(x)$: f_1^{max}.

2. Далі розв'язується задача оптимізації по наступному за важливістю критерію при додатковому обмеженні $f_1(x) \geq f_1^{max} - \Delta_1$, де Δ_1 – допустима поступка по першому критерію, яку вказує ОПР. Тобто розв'язуємо таку задачу:

$$f_2(x) \to \max,$$

$$f_1(x) \geq f_1^{max} - \Delta_1,$$

$$x \in X.$$

У результаті розв'язання задачі отримуємо оптимальне значення критерію $f_2(x)$: f_2^{max}.

3. Нехай у результаті реалізації k кроків процедури, отримані значення мають вигляд: $f_1^{min}, f_2^{min}, \dots, f_k^{min}$. Тоді на $(k+1)$-му кроці розв'язується задача:

$$f_{k+1}(x) \to \max,$$

$$f_1(x) \geq f_1^{max} - \Delta_1,$$

$$f_2(x) \geq f_2^{max} - \Delta_2,$$

$$\dots\dots\dots\dots\dots\dots\dots\dots\dots$$

$$f_k(x) \geq f_k^{max} - \Delta_k,$$

$$x \in X,$$

і буде одержано значення критерію f_k^{max} .

4. Якщо після розглядання усіх критеріїв відповідні результати задовольняють ОПР – задачу розв'язано. Оптимальним рішенням багатокритеріальної задачі буде розв'язок останньої скалярної задачі. Якщо ж деякі значення критеріїв не задовольняють ОПР, то можна повернутися на відповідний крок і скорегувати поступку по відповідному критерію. [3]

Таким чином, початкова багатокритеріальна задача зведена до послідовного вирішення ряду скалярних задач. У найкращому випадку кількість задач буде дорівнювати числу критеріїв.

Цей метод дає можливість врахувати пріоритети критеріїв і не допустити підвищення значень критеріїв більш, ніж деякий допустимий рівень.

Також варто відмітити, що цей метод не обмежує можливості ОПР у виборі ефективних альтернатив, що обґрунтовується теоремою:

Теорема (О. Вентцель). Для будь-якого впорядкування критеріїв і будь-якої ефективної альтернативи x^* існує послідовність невід’ємних поступок $\{\Delta f_i\}_{i=\overline{1,m}}$ таких, що на останньому кроці процедури буде отримана множина ефективних альтернатив, усі елементи якої рівноцінні x^*.

Серед недоліків методу варто відмітити, що лише на першому кроці методу величина поступки відповідає її фактичній величині, оскільки вона визначена на всій множині альтернатив. На наступних кроках величина поступки може бути значно меншою за її фактичну величину, оскільки вона визначається на «уточненій» множині альтернатив.

Недоліком методу є також зростання обчислювальної складності задач оптимізації з кількістю зроблених кроків, оскільки на кожному кроці додається нове обмеження. Метод використовує два типи інформації від ОПР: інформацію про впорядкування критеріїв і про діапазони значень критеріїв.

1.3. Максимізація прибутку

У сучасній економічній теорії часто приймається передумова про те, що метою діяльності фірми є максимізація прибутку, що задається як різниця між загальним доходом ($TR = pq$, де p – ціна готової продукції, q – кількість реалізованої продукції) і загальними витратами TC на виробництво відповідної продукції: $PR = TR - TC$. Зазвичай, аналізуючи підприємство в довгостроковому аспекті, вважають, що прибуток є невід'ємною величиною: $PR \geq 0$, адже в іншому випадку фірма завжди зможе вийти з бізнесу, уникнувши тим самим збитків. Наприклад, завдання максимізації прибутку фірми, що використовує два види ресурсів (факторів), може вирішуватися із застосуванням процедур як однокрокової:

$$\max_{x_1, x_2} PR = \max_{x_1, x_2} \{pq - p_{x_1} x_1 - p_{x_2} x_2\},$$

де x_1, x_2 і $q = F(x_1, x_2)$ – відповідно витрати факторів і обсяг продукції, що випускається як значення виробничої функції (характеризує технологічні процеси на підприємстві), – так і двокрокової оптимізації.

При двоетапній оптимізаційній схемі перший крок оптимізаційної процедури полягає в розв'язанні задачі мінімізації витрат підприємства при обмеженні за потужностями і розрахунку для даної технології витрат TC як виробничої функції F. Другий крок полягає власне в безумовній максимізації прибутку як виробничої функції. Отже, за допомогою двоетапної оптимізаційної схеми задачу максимізації прибутку можна представити в наступному вигляді:

$$\max_{q} PR = \max_{q} \{pq - TC(q, p_{x_1}, p_{x_2})\}:$$

$$\begin{cases} TC(q, p_{x_1}, p_{x_2}) = \min\limits_{x_1, x_2} \{p_{x_1} x_1 + p_{x_2} x_2\}: \\ F(x_1, x_2) \geq q;\ x_1 \geq 0, x_2 \geq 0. \end{cases}$$

Задача максимізації прибутку полягає у виборі такої точки на траєкторії розвитку підприємства, в якій різниця між виручкою і видатками буде найбільшою. Необхідна умова максимуму функції прибутку – рівність нулю її похідної:

$$\frac{dPR}{dq} = p - MC(q) = 0.$$

Іншими словами, при диференційованій функції витрат точка максимуму прибутку в умовах досконалої конкуренції повинна задовольняти умові рівності граничних видатків ринковій ціни на продукцію підприємства:

$$p = MC.$$

Достатньою умовою максимізації прибутку за двоетапною схемою буде недодатність її другої похідної, а значить, при досконалій конкуренції – невід'ємність похідної від граничних видатків:

$$\frac{d^2PR}{dq^2} = \frac{d}{dq}\big(p - MC(q)\big) = -\frac{dMC}{dq} \leq 0, \qquad \text{тобто } \frac{dMC}{dq} \geq 0.$$

Іншими словами, оптимальний обсяг виробництва F^* з точки зору максимізації прибутку для абсолютно конкурентної фірми буде відповідати неспадній ділянці граничних видатків, а значить, незростаючій віддачі від масштабу виробництва [8].

1.4. Мінімізація видатків

Припустимо, що у нас є два фактори виробництва з цінами w_1 та w_2, і ми хочемо знайти найдешевший спосіб виробництва заданого обсягу випуску q. Якщо позначити використовувані кількості кожного з двох факторів через x_1 і x_2, а виробничу функцію для фірми – через $f(x_1, x_2)$, то формально таку задачу можна записати у вигляді

$$\min_{x_1, x_2}(w_1 x_1 + w_2 x_2), \quad f(x_1, x_2) = q.$$

Розв’язок цієї задачі мінімізації видатків – величина мінімальних видатків, необхідних для досягнення певного обсягу випуску, – буде залежати від w_1, w_2 і q, тому ми запишемо цей розв’язок як $C(w_1, w_2, q)$, що є нічим іншим, як функцією видатків. Функція видатків $C(w_1, w_2, q)$ показує мінімальні витрати виробництва q одиниць випуску при цінах на фактори, рівних (w_1, w_2).

Щоб зрозуміти розв’язок цієї задачі, зобразимо функцію витрат і технологічні обмеження для фірми на одному графіку. Ізокванти дають нам технологічні обмеження – всі комбінації x_1 і x_2, за допомогою яких можна отримати відповідний обсяг виробництва q.

Припустимо, що ми хочемо нанести на графік всі комбінації факторів, що дають один і той же рівень витрат C. Ми можемо записати це у вигляді виразу

$$w_1 x_1 + w_2 x_2 = C,$$

яке може бути зведено до:

$$x_2 = \frac{C}{w_2} - \frac{w_1}{w_2} x_1.$$

Легко побачити, що це рівняння прямої, що має нахил $(-w_1/w_2)$, і точку перетину з вертикальною віссю C/w_2. Змінюючи число C, ми отримуємо цілу сім'ю ізокост (ліній фіксованих витрат фірми). Кожна точка ізокости встановлює одні і ті ж витрати C, і більш «високі» ізокости пов'язані з більшими витратами.

Таким чином, задача мінімізації витрат може бути перефразована наступним чином: знайти на ізокванті точку, з якою позв'язана «найнижча» ізокоста. Така точка показана на рис 3.

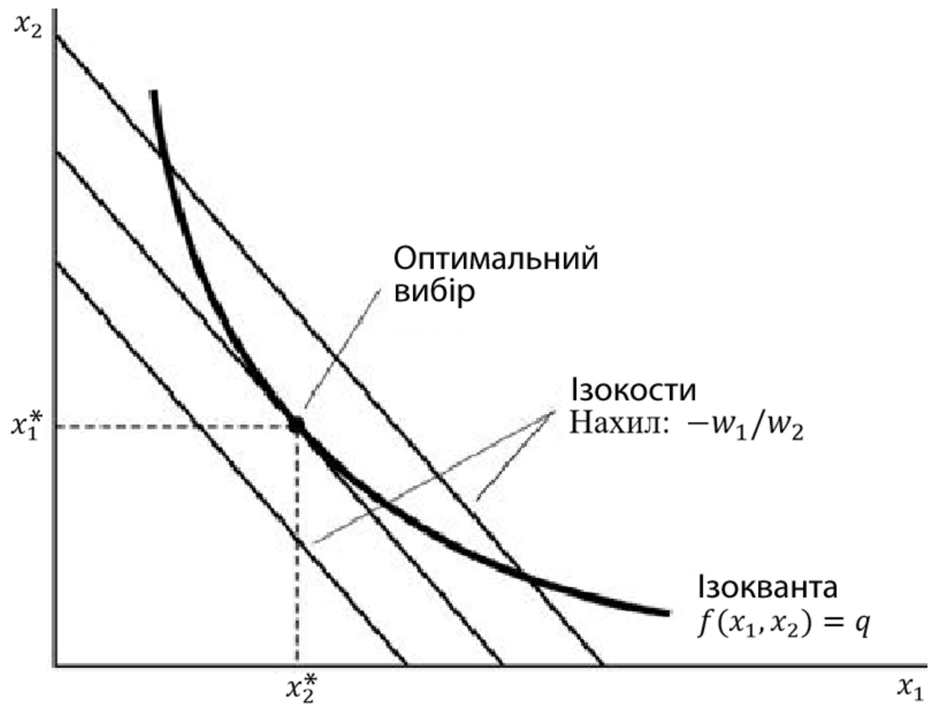


Рис. 3. Мінімізація витрат фірми

Основним моментом є те, що якщо оптимальний розв'язок передбачає використання певної кількості кожного з факторів і якщо ізокванта є гладкою кривою, то точка мінімізації витрат буде характеризуватися умовою дотику: нахил ізокванти повинен дорівнювати нахилу ізокости:

$$-\frac{MP_1(x_1^*,x_2^*)}{MP_2(x_1^*,x_2^*)} = -\frac{w_1}{w_2}, \tag{1.4.1}$$

де $MP_i(x_1^*, x_2^*) = \frac{\partial f(x_1^*, x_2^*)}{\partial x_i}$ – граничний продукт i-того фактора [4]. Розглянемо будь-яку зміну структури виробництва $(\Delta x_1, \Delta x_2)$, при якому випуск продукції залишається постійним. Така зміна повинна задовольняти рівняння:

$$MP_1(x_1^*, x_2^*)\Delta x_1 + MP_2(x_1^*, x_2^*)\Delta x_2 = 0. \quad (1.4.2)$$

Варто відмітити, що Δx_1 і Δx_2 повинні мати протилежні знаки (якщо збільшується кількість першого фактора, то для збереження випуску незмінним доведеться зменшити кількість другого фактора).

Якщо ми знаходимося у точці мінімуму видатків, то подібна зміна не може призвести до зменшення видатків, тому повинна виконуватися така умова:

$$w_1\Delta x_1 + w_2\Delta x_2 \geq 0. \quad (1.4.3)$$

Тепер розглянемо зміну $(-\Delta x_1, -\Delta x_2)$, при якій також виробляється постійний обсяг продукції, і видатки не можуть зменшуватись. Це має на увазі, що виконується:

$$-w_1\Delta x_1 - w_2\Delta x_2 \geq 0. \quad (1.4.4)$$

Об’єднавши вирази (1.4.3) і (1.4.4), отримаємо:

$$w_1\Delta x_1 + w_2\Delta x_2 = 0. \quad (1.4.5)$$

Розв’язок рівнянь (1.4.2) і (1.4.5) для $\Delta x_2/\Delta x_1$ дає

$$\frac{\Delta x_2}{\Delta x_2} = -\frac{w_1}{w_2} = -\frac{MP_1(x_1^*, x_2^*)}{MP_2(x_1^*, x_2^*)},$$

що є умовою мінімізації видатків, виведеної вище. [7]

РОЗДІЛ 2
РОЗВ'ЯЗУВАННЯ ЗАДАЧ ОПТИМАЛЬНОЇ ПОВЕДІНКИ ФІРМИ

2.1. Задача максимізації прибутку

Розглянемо довгострокову задачу щодо можливості придбання ресурсів [2]. Вона має вигляд:

$$\pi(x) = pF(x) - wx \to \max,\ x = (x_i)_{i=\overline{1,n}} \geq 0, \tag{2.1.1}$$

де $F(x)$ – виробнича функція фірми,
$w = (w_i)_{i=\overline{1,n}}$ – вектор-рядок цін на фактори виробництва,
x – вектор-стовпчик попиту на обсяги виробничих факторів (ресурсів),
p – ціна готової продукції.

Оскільки ціни на фактори виробництва при прогнозуванні точно не визначені, то в якості їх опису використаємо «нечітко» задані ціни, тобто ціна на i-ий фактор знаходиться в діапазоні $[w_{i1}; w_{i2}]$, $i = \overline{1,n}$, із заданням відповідної функції належності.

Крім цього, можна встановити залежність $p = f(w)$, де $f(w)$ функція, що належить деякому класу, для якої оцінювання невідомих параметрів проводиться методом найменших квадратів.

Для побудови «більш адекватної» моделі задачі використовується математичний апарат теорії нечітких множин. Нечіткі інтервали W^i, $i = \overline{1,n}$, визначаються як множина аргументів , на яких функція належності має рівень не нижчий за λ. Існують різні механізми формування рівня λ. Наприклад, якщо мається множина експертів $R = \{1,2,\dots,r\}$, кожен з яких формує свою неперервну функцію належності $\mu_r^i(w^i)$, $i = \overline{1,n}$, $r = \overline{1,R}$, і встановлює рівень значення $\lambda_r \in [0,1]$, $r = \overline{1,R}$. Тоді при заданих рінвях λ_r $\left(1 \geq \mu_r^i(w^i) \geq \lambda_r\right)$ кожним експертом визначається інтервал зміни значень

W_r^i, $r = \overline{1, R}$, де $W_r^i = \left[w_{r(H)}^i, w_{r(B)}^i\right]$. Значення величини інтервалу W_r^i коректується з урахуванням особливостей експерта: «оптимістичність», «реалістичність», «песимістичність».

Також важливим доповненням буде те, що множиною рівня λ нечіткого інтервалу W^i називається множина $K_\lambda(W^i)$, для якої $\mu^i(w^i) \geq \lambda$, тобто:

$$K_\lambda\left(W^i\right) = \left\{W \in W^i, \mu^i\left(w^i\right) \geq \lambda\,, i = \overline{1, n}\ \right\}.$$

Якщо $W^i = \emptyset$ хоча б для одного i, то процес узгодження продовжується за допомогою методів обробки експертної інформації. [2]

Будемо у подальших дослідженнях використовувати виробничу функцію вигляду $F(x) = x_1^{0.2} \cdot x_2^{0.6}$, ціна готової продукції фіксована і дорівнює $p = 200000$, а для цін w_1, w_2 задано, відповідно, трапецеїдальну (з параметрами $a = 16200,\ b = 17400,\ c = 18200,\ d = 19000$) і гаусівську (з параметрами $\sigma = 50,\ c = 81500$) функції належності. Ці функції можна побачити на рис. 4.

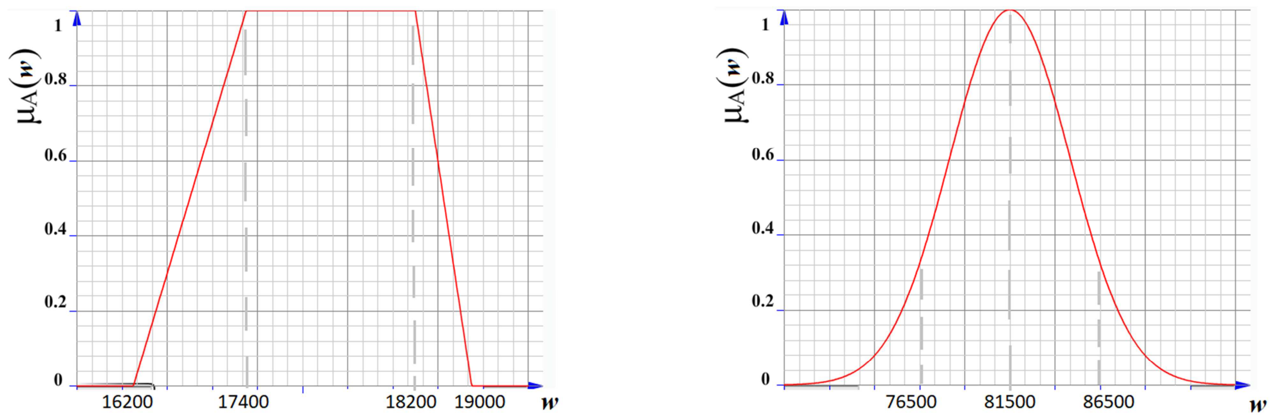


Рис. 4. Функції належності цін на ресурси

Алгоритм розв’язання задачі полягає в наступному:

1. Особа, що приймає рішення, обирає узгоджений експертами рівень значення λ, для якого розв’язується задача

$$\pi(x) = pF(x) - xw \to \max,$$
$$w \in L_\lambda(w), \tag{2.1.2}$$
$$x = (x_i)_{i=\overline{1,n}} \geq 0.$$

2. Аналіз отриманого розв'язку. Для відповідного значення λ обраховується «ймовірність» $P_\lambda = (1 - \lambda) \times 100\%$, з якою $\pi_\lambda(x^*) \in [\pi_{\min}, \pi_{\max}]$, де $\pi_{\min}$ відповідає $w_{\text{В}}$, а $\pi_{\max}$ – $w_{\text{Н}}$. Якщо ОПР задовольняє величина «інтервалу невизначеності» для $\pi_\lambda(x^*)$ та значення P_λ, то кінець, інакше задається новий рівень λ.
3. Пошук компромісного розв'язку. Для пошуку компромісного розв'язку пропонується наступна трьохкритеріальна задача:

$$f_1 = P_\lambda \to \max,$$
$$f_2 = \Delta\pi_\lambda = \pi_{\max}(x, P) - \pi_{\min}(x, P) \to \min \tag{2.1.3}$$
$$f_3 = \pi_\lambda(x, P) \to \max,$$
$$x \geq 0,\ 0 \leq P \leq 1.$$

Перший критерій має зміст достовірності величини інтервалу, другий – точності задання інтервалу, третій – оцінки значення прибутку.

Ця трьохкритеріальна задача розв'язується методом послідовних поступок, для якої критерії впорядковуються за зменшенням їх важливості:

$$f_1 \succ f_2 \succ f_3.$$

Для цільової функції $\pi_\lambda(x, P)$ здійснюється монотонне перетворення до нормованого базрозмірного вигляду:

$$f_k(x, P) = \begin{cases} \frac{\pi_k(x,P) - \overline{\pi_k}}{\pi_k^* - \overline{\pi_k}} & \text{при } \overline{\pi_k} \leq \pi_k(x, P) \leq \pi_k^*, \\ 0, & \text{в іншому випадку,} \end{cases} \tag{2.1.4}$$

де $\overline{\pi_k} = \min \pi_k,\ \pi_k^* = \max \pi_k,\ k = \min,\ \max.$

Позначимо $L(x) = \min\left(f_{min}(x,P), f_{max}(x,P)\right)$. Тоді компромісний розв'язок отримаємо, розв'язавши задачу:

$$\begin{gathered} L \to max, \\ (\pi^*_{min} - \bar{\pi}_{min})L(x) - \bar{\pi}_{min}(x,P) \le -\bar{\pi}_{min}, \\ (\pi^*_{max} - \bar{\pi}_{max})L(x) - \bar{\pi}_{max}(x,P) \le -\bar{\pi}_{max}, \\ \Delta\pi_\lambda = \pi_{max}(x,P) - \pi_{min}(x,P) \le \Delta f_2, \\ P \le \Delta f_1,\ x \ge 0,\ L \ge 0. \end{gathered} \tag{2.1.5}$$

У залежності від психоматичних характеристик ОПР, для нього важливішим буде або точність $\Delta\pi_\lambda$, або достовірність P_λ величини інтервалу. Тому поступкии Δf_1, Δf_2 відповідно коригують зміною їх значення.

Перейдемо від алгоритму до розв'язання задачі.

Спочатку обираємо λ. У даному випадку $\lambda = 0.01$ через неможливість обрати $\lambda = 0$ (одна з цін задана гаусівською функцією належності, яка є аналітичною). Позначимо

$$\begin{gathered} P = (1 - \lambda) \times 100\% \approx 99.99\%, \\ w_1^{min} = 17400, \qquad w_1^{max} = 18200, \\ \Delta w_1^{min} = 1200, \qquad \Delta w_1^{max} = 800, \\ w_2^{min} = w_2^{max} = 81500, \\ \Delta w_2^{min} = \Delta w_2^{max} = 10730, \end{gathered}$$

на що отримаємо:

$$\begin{gathered} \pi_{\min}(x,P) = p * F(x) - (18200 + 800P)x_1 - (81500 + 10730P)x_2, \\ \pi_{\max}(x,P) = p * F(x) - (17400 - 1200P)x_1 - (81500 - 10730P)x_2, \\ \Delta\pi_\lambda = 2000Px_1 + 21460Px_2 + 800x_1 \to \min. \end{gathered}$$

Зробивши аналогічні кроки для різних λ, складаємо таблицю «інтервалів невизначеності» прибутку в залежності від λ (Таблиця 1).

Таблиця 1

λ	$x_{\min}$	$x_{\max}$	$\pi_{\min}(x_{\max})$	$\pi_{\max}(x_{\min})$	π_{λ}
0.01	(10, 20)	(15, 25)	-216380	332920	(-216380, 356150)
0.1	(10.1, 20.2)	(14.9, 24.8)	-114500	253700	(-114500, 260250)
0.2	(10.2, 20.4)	(14.8, 24.6)	-64370	214520	(-64370, 216085)
0.3	(10.3, 20.6)	(14.7, 24.4)	-27900	185520	(-27900, 185670)
0.4	(10.4, 20.8)	(14.6, 24.2)	1355	161730	(1355, 162540)
0.5	(10.5, 21)	(14.5, 24)	25670	141500	(25670, 144430)
0.6	(10.6, 21.2)	(14.4, 23.8)	46100	124080	(46100, 130280)
0.7	(10.7, 21.4)	(14.3, 23.6)	63220	109130	(60300, 119520)
0.8	(10.8, 21.6)	(14.2, 23.4)	77280	96570	(66660, 111880)
0.9	(10.9, 21.8)	(14.1, 23.2)	88280	86530	(70520, 107370)
1	(11, 22)	(14, 23)	95500	79850	(71050, 106700)

Оскільки при $\lambda = \overline{0.01,0.3}$ прибуток може набувати від'ємних значень, то припускаємо, що ОПР вирішила обрати $\lambda = 0.7$ як проміжний рівень між 0.4 та 1.

Тепер записуємо трьохкритеріальну задачу (2.1.3) для обраного рівня λ:

$$f_1 = P_{0.7} \to \max,$$
$$f_2 = \Delta\pi_{0.7} = 40x_1 + 30Px_1 + 299Px_2 \to \min,$$
$$f_3 = \pi_{0.7}(x, P) \to \max,$$
$$x \geq 0, \qquad 0.7 < P \leq 1.$$

Скористаємося методом послідовних поступок для розв'язку цієї задачі. Максимізуємо перший критерій:

$$f_1 = P_{0.7} \to max,$$
$$x \geq 0, \qquad 0.7 < P \leq 1.$$

Після чого мінімізуємо другий критерій:

$$40x_1 + 30Px_1 + 299Px_2 \to \min,$$
$$x \geq 0, \qquad 0.7 < P \leq 1.$$

Розв’язком буде альтернатива $x = (0,0)$, яка має оцінку $y = (1,0,0)$. Отриманий результат не задовольняє ОПР за третім критерієм, тому було вирішено визначити величину поступки як $\Delta f_2 = 1200$.

$$\pi(x, P) \to \max,$$
$$40x_1 + 30Px_1 + 299Px_2 \leq 1200,$$
$$x \geq 0, \qquad 0.7 \leq P < 1.$$

Здійснивши монотонне перетворення до нормованого безрозмірного вигляду, переходимо до задачі (2.1.5):

$$L \to \max,$$
$$276300L - (200000x_1^{0.2}x_2^{0.6} - (17400 - 360P)x_1 -$$
$$-(81500 - 2990P)x_2) \leq -79850,$$
$$311880L - (200000x_1^{0.2}x_2^{0.6} - (18200 + 240P)x_1 -$$
$$-(81500 + 2990P)x_2) \leq 216380,$$
$$40x_1 + 30Px_1 + 299Px_2 \leq 1200,$$
$$x \geq 0, \qquad L \geq 0, \qquad 0.7 \leq P < 1.$$

Розв’язавши задачу за допомогою теореми Куна-Таккера [11], отримаємо оптимальні розв’язки $x_1 = 0.34$, $x_2 = 1.6$.

2.2. Задача мінімізації виробничих видатків

Розглянемо задачу оптимальної поведінки фірми, орієнтовану на мінімізацію виробничих видатків [7]. Вона має вигляд:

$$C(x) = wx \to \min,\ F(x) \geq q,\ x = (x_i)_{i=\overline{1,n}} \geq 0, \qquad (2.2.1)$$

де $F(x)$ – виробнича функція,

w – вектор цін на фактори виробництва,

x – попит на обсяг факторів (ресурсів),

q – попередньо заданий рівень пропозиції фірми.

Для розв'язку такого типу задач використовується метод невизначених множників Лагранжа, який допомагає звести задачу на умовний екстремум функції $C(x)$ до задачі на безумовний екстремум функції Лагранжа $L(x, \alpha)$, що містить додаткову змінну α (множник Лагранжа). Уведена функція Лагранжа матиме вигляд:

$$L(x, \alpha) = wx + \alpha\big(q - F(x)\big). \qquad (2.2.2)$$

Запишемо необхідну умову екстремуму функції:

$$\begin{cases} \dfrac{\partial L}{\partial x_1} = w_1 - \alpha \dfrac{\partial F}{\partial x_1} = 0; \\ \dots \dots \dots \dots \dots \dots \dots \dots \\ \dfrac{\partial L}{\partial x_n} = w_n - \alpha \dfrac{\partial F}{\partial x_n} = 0; \\ \dfrac{\partial L}{\partial \alpha} = q - F(x) = 0. \end{cases} \qquad (2.2.3)$$

Розв'язком системи при заданих значеннях w_i та q буде оптимальний розподіл факторів виробництва, що задовольняють рівню пропозиції фірми при мінімізації повної вартості витрат.

Тепер розв'яжемо задачу мінімізації виробничих видатків для умов минулої задачі при рівні пропозиції $q = 1000$.

Спочатку введемо функцію Лагранжа (2.2.2):

$$L(x, \alpha) = 18440x_1 + 83990x_2 + \alpha(1000 - 200000x_1^{0.2}x_2^{0.6}).$$

Після чого запишемо необхідну умову екстремуму функції (2.2.3):

$$\begin{cases} \dfrac{\partial L}{\partial x_1} = 18440 - \alpha \cdot 200000 \cdot 0{,}2 \cdot x_1^{-0.8} x_2^{0.6} = 0; \\ \dfrac{\partial L}{\partial x_2} = 83990 - \alpha \cdot 200000 \cdot 0{,}6 \cdot x_1^{0.2} x_2^{-0.4} = 0; \\ \dfrac{\partial L}{\partial \alpha} = 1000 - x_1^{0.2} x_2^{0.6} = 0. \end{cases}$$

Розв'язавши систему, отримаємо $x_1 = 18127.4$, $x_2 = 3806.75$, що є оптимальним розподілом факторів виробництва, які задовольняють рівню пропозиції фірми q при мінімізації повної вартості витрат.

2.3. Приклади для макроекономічних даних

Розглянемо наведені вище задачі на практичному прикладі з реальними параметрами.

Так виробнича функція $F(x)$ має вигляд $F(x) = x_1^{0.1} \cdot x_2^{0.3}$, ціна готової продукції фіксована і дорівнює $p = 150$, а для цін w_1, w_2 задано трапецеїдальну (з параметрами $a = 14,\ b = 21,\ c = 23,\ d = 27$) і гаусівську (з параметрами $\sigma = \sqrt{15},\ c = 19$) функції належності. Ці функції, що сформовані на основі даних про ціну на водопостачання [9] й електроенергію [10] відповідно, можна побачити на рис. 5. Також вищезгадані ресурси можна розглядати як макроекономічні, тобто такі, що визначають рівень ВВП країни. В такому випадку макроекономічна виробнича функція буде залежати від обсягів витрат цих ресурсів, і, власне, визначати (формально) рівень ВВП.

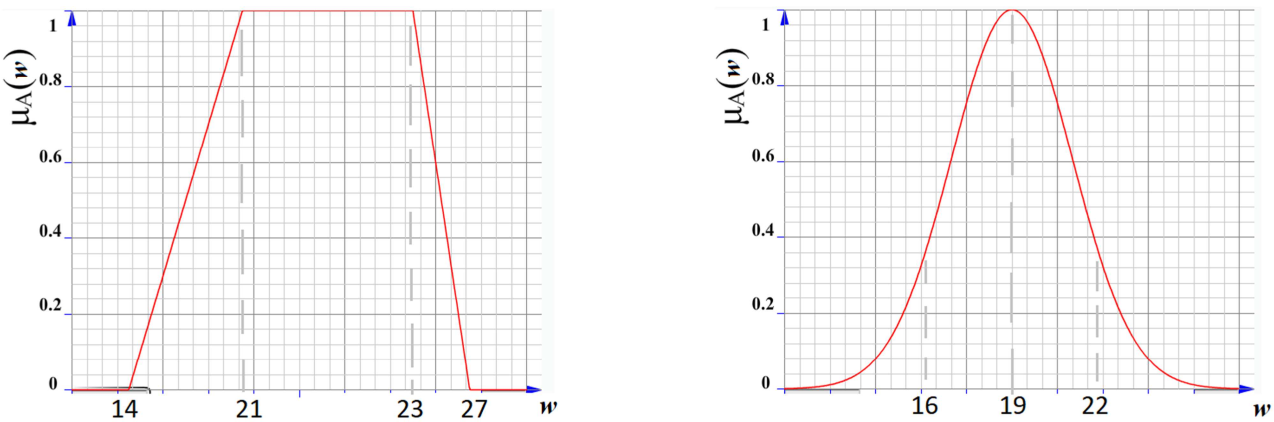


Рис. 5. Функції належності цін на макроекономічні ресурси.

Аналогічно до прикладу, наведеного у підрозділі 2.1, складаємо таблицю «інтервалів невизначеності» прибутку для різних λ (Таблиця 2).

ОПР обирає $\lambda = 0.9$, оскільки її задовольняє мінімальний прибуток, наближений до 120 грн., та вона вважає доречним ризикнути у даній ситуації заради 1 грн.

Таблиця 2

λ	$x_{\min}$	$x_{\max}$	$\pi_{\min}(x_{\max})$	$\pi_{\max}(x_{\min})$	π_{λ}
0.01	(0.5, 1.5)	(3, 6)	42.50	134.80	(42.50 179.50)
0.1	(0.6, 1.7)	(2.9, 5.8)	65.75	134.60	(65.75, 158.70)
0.2	(0.7, 1.9)	(2.8, 5.6)	78.80	135.25	(78.80, 148.10)
0.3	(0.8, 2.1)	(2.7, 5.4)	88.70	135.55	(88.70, 140.75)
0.4	(0.9, 2.3)	(2.6, 5.2)	96.70	135.35	(96.70, 135.35)
0.5	(1, 2.5)	(2.5, 5)	103.35	134.70	(103.35, 134.70)
0.6	(1.1, 2.7)	(2.4, 4.8)	108.85	133.80	(108.85, 133.80)
0.7	(1.2, 2.9)	(2.3, 4.6)	113.35	132.70	(113.35, 132.70)
0.8	(1.3, 3.1)	(2.2, 4.4)	117.00	131.35	(117.00, 131.35)
0.9	(1.4, 3.2)	(2.1, 4.2)	119.75	130.35	(119.75, 130.35)
1	(1.5, 3.5)	(2, 4)	121.65	129.70	(121.65, 129.70)

Запишемо задачу (2.1.3) для обраного рівня λ:

$$f_1 = P_{0.9} \to \max,$$
$$f_2 = \Delta\pi_{0.9} = 2x_1 + 1.1Px_1 + 2.5Px_2 \to \min,$$
$$f_3 = \pi_{0.9}(x, P) \to \max,$$
$$x \geq 0, \qquad 0.9 < P \leq 1.$$

Скористаємося методом послідовних поступок для розв'язку цієї задачі. Максимізуємо перший критерій:

$$f_1 = P_{0.9} \to max,$$
$$x \geq 0, \qquad 0.9 < P \leq 1.$$

Після чого мінімізуємо другий критерій:

$$2x_1 + 1.1Px_1 + 2.5Px_2 \to \min,$$
$$x \geq 0, \qquad 0.9 < P \leq 1.$$

Розв'язком буде альтернатива $x = (0,0)$, яка має оцінку $y = (1,0,0)$. Отриманий результат не задовольняє ОПР за третім критерієм, тому було вирішено визначити величину поступки як $\Delta f_2 = 50$.

$$\pi(x,P) \to \max,$$
$$2x_1 + 1.1Px_1 + 2.5Px_2 \le 50,$$
$$x \ge 0, \qquad 0.9 \le P < 1.$$

Здійснивши монотонне перетворення до нормованого безрозмірного вигляду, переходимо до задачі (2.1.5):

$$L \to \max,$$
$$5.85L - (150x_1^{0.1}x_2^{0.3} - (21 - 0.7P)x_1 - (19 - 1.25P)x_2) \le -129.7,$$
$$79.15L - (150x_1^{0.1}x_2^{0.3} - (23 + 0.4P)x_1 - (19 + 1.25P)x_2) \le -42.5,$$
$$2x_1 + 1.1Px_1 + 2.5Px_2 \le 50,$$
$$x \ge 0, \qquad L \ge 0, \qquad 0.9 \le P < 1.$$

Розв'язавши задачу за допомогою теореми Куна-Таккера, отримаємо оптимальні розв'язки $x_1 = 1.12$, $x_2 = 3.3$.

Розглянувши задачу (2.1.3) для різних значень λ, приходимо до висновку, що при всіх λ задача (2.1.5) має розв'язок. В противному випадку це свідчило б про неприйнятність відповідних цін для максимізації прибутку при обраному λ.

Далі розв'яжемо задачу мінімізації виробничих видатків. Виробнича функція та відповідні функції належності будуть тими ж, що й у попередньому прикладі. За рівень пропозиції оберемо $q = 2$.

Спочатку введемо функцію Лагранжа (2.2.2):

$$L(x,\alpha) = 23.4x_1 + 20.25x_2 + \alpha(2 - x_1^{0.1}x_2^{0.3}).$$

Після чого запишемо необхідну умову екстремуму функції (2.2.3):

$$\begin{cases} \dfrac{\partial L}{\partial x_1} = 23.4 - \alpha \cdot 0{,}1 \cdot x_1^{-0.9}x_2^{0.3} = 0; \\ \dfrac{\partial L}{\partial x_2} = 20.25 - \alpha \cdot 0{,}3 \cdot x_1^{0.1}x_2^{-0.7} = 0; \\ \dfrac{\partial L}{\partial \alpha} = 2 - x_1^{0.1}x_2^{0.3} = 0. \end{cases}$$

Розв’язавши систему, отримаємо $x_1 = 2.22$, $x_2 = 7.71$, що є оптимальним розподілом факторів виробництва, які задовольняють рівню пропозиції фірми q при мінімізації повної вартості витрат. Так, функція видатків, що вказує на мінімальні витрати, буде дорівнювати

$$C(x) = 23.4 \cdot 2.22 + 20.25 \cdot 7.71 = 208.07 \text{ грн.}$$

Таким чином, користуючись отриманими даними, можна скласти функцію прибутку

$$\pi(x) = p \cdot 2.2^{0.1} \cdot 7.71^{0.3} - 208.07,$$

де p – ціна готової продукції. Розв’язавши рівняння $\pi(x) = 0$ відносно p, отримаємо граничне значення ціни $p = 104.196$ грн. Так, при ціні вищій за це значення, макроекономічне виробництво буде прибутковим, а при нижчій – збитковим.

ВИСНОВКИ

У 1965 році, коли Лофті Заде вперше застосував термін «Fuzzy Sets» (Нечіткі множини), він навіть не уявляв, наскільки поширеним буде їх застосування всього за 50 років. Теорія нечітких множин є досить важливим і потужним інструментом для вирішення повсякденних, і не тільки, життєвих проблем, і з часом важливість цього інструменту поступово зростає.

У даній дипломній роботі були розглянуті методи для налагодження ефективного виробництва на однопродуктовому підприємстві, такі як: задача оптимізації прибутку фірми та задача мінімізації виробничих видатків при нечітко заданих цінах на ресурси, зручність і дієвість застосування яких у макроекономічному контексті була перевірена практично на реальних даних. Так, у процесі розв’язку задач були використані нечіткі розподіли цін на деякі макроекономічні ресурси України, що дозволяє промоделювати майбутню поведінку економіки стосовно зміни цих цін. Окрім цього, під час розв’язання задачі максимізації прибутку було розроблено програмний продукт, що спрощує використання методу послідовних поступок для розв’язку трьохкритеріальної задачі.

Підбиваючи підсумки, можна з упевненістю стверджувати, що хоча задача оптимізації прибутку однопродуктової фірми при нечіткому ціноутворенні на ресурси та задача мінімізації виробничих видатків при заданих обсягах випуску продукції є вузьконаправленими і не дуже часто видається нагода застосувати їх у реальному житті, однак перша задача є досить простим і потужним інструментом швидкого пошуку ціни для максимізації прибутку при неточній або розпливчастій інформації щодо виробничих ресурсів, а друга дозволяє легко і швидко оптимізувати виробництво з обмеженим (фіксованим) випуском товару та мінімізувати витрати на нього.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Волошин О.Ф., Мащенко С.О. Моделі та методи прийняття рішень: Навч. посібник для студентів вищих навчальних закладів. – К.: ВПЦ «Київський университет», 2009. – 340с.
2. Волошин О.Ф., Пошелюжний О.В. Побудова та дослідження нечіткої моделі поведінки однопродуктової фірми. In: «Bulletin of University of Kiev Series: Physics & Mathematics», 2009 – P. 111-114.
3. Ус С.А., Коряшкіна Л.С. Моделі й методи прийняття рішень: Навч. посібник, 2014. – 300 с.
4. Пономаренко О.І., Перестюк М.О., Бурим В.М. Сучасний економічний аналіз. Навч. посібник: у 2-х част. – К.: Вища школа, 2004. Ч.1. Мікроекономіка. – 262 с.
5. Раскин Л.Г., Серая О.В. Нечеткая математика. Основы теории. Приложения. – Харьков : Парус, 2008. – 231с.
6. Ю.П. Зайченко. Нечеткие модели и методы в интеллектуальных системах. – К.: «Издательский Дом «Слово», 2008 – 333 с.
7. Вэриан Х.Р. Микроэкономика. Промежуточный уровень. Современный подход: Учебник для вузов/Пер. с англ. под ред. Н.Л. Фроловой. – М.: ЮНИТИ, 1997. – 767с.
8. Максимизация прибыли и предложение продукции конкурентной фирмой [Електронний ресурс]. Режим доступу: https://www.econ.msu.ru/ext/lib/Category/x56/x5d/22109/file/%D0%BF%D0%B0%D1%80%D0%B0%D0%B3%D1%80%D0%B0%D1%84%204_3.pdf . Дата звернення: 15.05.2021.
9. ПрАТ АК Водоканал – Тарифи, внески, абонплата [Електронний ресурс]. Архівні копії від 11.02.2018 – 20.11.2020. – Режим доступу: http://web.archive.org/web/20180601000000*/https://vodokanal.kiev.ua/rozraxunki-%D1%96-tarifi . Дата звернення: 12.05.2021.
10. Середні ціни на електроенергію для споживачів. [Електронний ресурс]. – Режим доступу:

http://ukrstat.gov.ua/operativ/operativ2018/energ/ser_cin_el_energ/ser_cin_el_energ_u/arh_sc_elen2018_u.htm . Дата звернення: 12.05.2021.

11. Бельков В. Н., Ланшаков В.Л. Автоматизированное проектирование технических систем: Учебное пособие, 2009. – 143 с.

ДОДАТОК

```
//MyForm.hpp
#pragma once
namespace Diploma {

	using namespace System;
	using namespace System::ComponentModel;
	using namespace System::Collections;
	using namespace System::Windows::Forms;
	using namespace System::Data;
	using namespace System::Drawing;

	public ref class MyForm : public System::Windows::Forms::Form
	{
	public:
	   MyForm(void){
			InitializeComponent();
	   }

	protected:
	   ~MyForm(){
			if (components)
			{
				delete components;
			}
	   }
	private: System::Windows::Forms::GroupBox^  groupBox1;
	protected:
	private: System::Windows::Forms::Label^  label6;
	private: System::Windows::Forms::Label^  label5;
	private: System::Windows::Forms::Label^  label4;
	private: System::Windows::Forms::Label^  label3;
	private: System::Windows::Forms::Label^  label2;
	private: System::Windows::Forms::Label^  label1;
	private: System::Windows::Forms::Label^  label7;
	private: System::Windows::Forms::NumericUpDown^  numericUpDown1;
	private: System::Windows::Forms::NumericUpDown^  numericUpDown2;
	private: System::Windows::Forms::Button^  button1;
	private: System::Windows::Forms::Button^  button2;
	private: System::Windows::Forms::GroupBox^  groupBox2;
	private: System::Windows::Forms::Label^  label8;
```

```
	private:
		System::ComponentModel::Container ^components;

#pragma region Windows Form Designer generated code
		void InitializeComponent(void){
			this->groupBox1 = (gcnew System::Windows::Forms::GroupBox());
			this->label6 = (gcnew System::Windows::Forms::Label());
			this->label5 = (gcnew System::Windows::Forms::Label());
			this->label4 = (gcnew System::Windows::Forms::Label());
			this->label3 = (gcnew System::Windows::Forms::Label());
			this->label2 = (gcnew System::Windows::Forms::Label());
			this->label1 = (gcnew System::Windows::Forms::Label());
			this->label7 = (gcnew System::Windows::Forms::Label());
			this->numericUpDown1 = (gcnew System::Windows::Forms::NumericUpDown());
			this->numericUpDown2 = (gcnew System::Windows::Forms::NumericUpDown());
			this->button1 = (gcnew System::Windows::Forms::Button());
			this->button2 = (gcnew System::Windows::Forms::Button());
			this->groupBox2 = (gcnew System::Windows::Forms::GroupBox());
			this->label8 = (gcnew System::Windows::Forms::Label());
			this->groupBox1->SuspendLayout();
			(cli::safe_cast<System::ComponentModel::ISupportInitialize^>(this-
>numericUpDown1))->BeginInit();
			(cli::safe_cast<System::ComponentModel::ISupportInitialize^>(this-
>numericUpDown2))->BeginInit();
			this->groupBox2->SuspendLayout();
			this->SuspendLayout();
			//
			// groupBox1
			this->groupBox1->Controls->Add(this->label6);
			this->groupBox1->Controls->Add(this->label5);
			this->groupBox1->Controls->Add(this->label4);
			this->groupBox1->Controls->Add(this->label3);
			this->groupBox1->Controls->Add(this->label2);
			this->groupBox1->Location = System::Drawing::Point(192, 12);
			this->groupBox1->Name = L"groupBox1";
			this->groupBox1->Size = System::Drawing::Size(280, 164);
			this->groupBox1->TabIndex = 0;
			this->groupBox1->TabStop = false;
			this->groupBox1->Text = L"Задача";
			this->groupBox1->Visible = false;
			//
			// label6
```

```
                    this->label6->AutoSize = true;
                    this->label6->Font = (gcnew System::Drawing::Font(L"Times New Roman", 12,
System::Drawing::FontStyle::Italic, System::Drawing::GraphicsUnit::Point,
                              static_cast<System::Byte>(204)));
                    this->label6->Location = System::Drawing::Point(97, 122);
                    this->label6->Name = L"label6";
                    this->label6->Size = System::Drawing::Size(69, 19);
                    this->label6->TabIndex = 4;
                    this->label6->Text = L"0 ≤ P ≤ 1";
                    this->label6->Click += gcnew System::EventHandler(this, &MyForm::label6_Click);
                    //
                    // label5
                    this->label5->AutoSize = true;
                    this->label5->Font = (gcnew System::Drawing::Font(L"Times New Roman", 12,
System::Drawing::FontStyle::Italic, System::Drawing::GraphicsUnit::Point,
                              static_cast<System::Byte>(204)));
                    this->label5->Location = System::Drawing::Point(25, 122);
                    this->label5->Name = L"label5";
                    this->label5->Size = System::Drawing::Size(42, 19);
                    this->label5->TabIndex = 3;
                    this->label5->Text = L"x ≥ 0";
                    //
                    // label4
                    this->label4->AutoSize = true;
                    this->label4->Font = (gcnew System::Drawing::Font(L"Times New Roman", 12,
System::Drawing::FontStyle::Italic, System::Drawing::GraphicsUnit::Point,
                              static_cast<System::Byte>(204)));
                    this->label4->Location = System::Drawing::Point(25, 93);
                    this->label4->Name = L"label4";
                    this->label4->Size = System::Drawing::Size(101, 19);
                    this->label4->TabIndex = 2;
                    this->label4->□□□t = L"π(□,□) → m□□";
                    //
                    // label3
                    this->label3->AutoSize = true;
                    this->label3->Font = (gcnew System::Drawing::Font(L"Times New Roman", 12,
System::Drawing::FontStyle::Italic, System::Drawing::GraphicsUnit::Point,
                              static_cast<System::Byte>(204)));
                    this->label3->Location = System::Drawing::Point(25, 63);
                    this->label3->Name = L"label3";
                    this->label3->Size = System::Drawing::Size(108, 19);
                    this->label3->TabIndex = 1;
```

```
this->label3->Text = L"x1 + x2 → m□n";
this->label3->Click += gcnew System::EventHandler(this, &MyForm::label3_Click);
//
// label2
this->label2->AutoSize = true;
this->label2->Font = (gcnew System::Drawing::Font(L"Times New Roman", 12,
System::Drawing::FontStyle::Italic, System::Drawing::GraphicsUnit::Point,
    static_cast<System::Byte>(204)));
this->label2->Location = System::Drawing::Point(25, 32);
this->label2->Name = L"label2";
this->label2->Size = System::Drawing::Size(71, 19);
this->label2->TabIndex = 0;
this->label2->Text = L"P → m□□";
//
// label1
this->label1->AutoSize = true;
this->label1->Font = (gcnew System::Drawing::Font(L"Times New Roman", 14.25F,
System::Drawing::FontStyle::Italic, System::Drawing::GraphicsUnit::Point,
    static_cast<System::Byte>(204)));
this->label1->Location = System::Drawing::Point(22, 38);
this->label1->Name = L"label1";
this->label1->Size = System::Drawing::Size(31, 21);
this->label1->TabIndex = 1;
this->label1->Text = L"λ=";
//
// label7
this->label7->AutoSize = true;
this->label7->Font = (gcnew System::Drawing::Font(L"Times New Roman", 14.25F,
System::Drawing::FontStyle::Italic));
this->label7->Location = System::Drawing::Point(14, 90);
this->label7->Name = L"label7";
this->label7->Size = System::Drawing::Size(49, 21);
this->label7->TabIndex = 2;
this->label7->Text = L"Δf2=";
this->label7->Visible = false;
//
// numericUpDown1
this->numericUpDown1->DecimalPlaces = 2;
this->numericUpDown1->Font = (gcnew System::Drawing::Font(L"Times New Roman",
14.25F, System::Drawing::FontStyle::Italic));
this->numericUpDown1->Increment = System::Decimal(gcnew cli::array< System::Int32
>(4) { 1, 0, 0, 65536 });
```

```
			this->numericUpDown1->Location = System::Drawing::Point(59, 36);
			this->numericUpDown1->Maximum = System::Decimal(gcnew cli::array< System::Int32
>(4) { 1, 0, 0, 0 });
			this->numericUpDown1->Minimum = System::Decimal(gcnew cli::array< System::Int32
>(4) { 1, 0, 0, 131072 });
			this->numericUpDown1->Name = L"numericUpDown1";
			this->numericUpDown1->Size = System::Drawing::Size(60, 29);
			this->numericUpDown1->TabIndex = 3;
			this->numericUpDown1->Value = System::Decimal(gcnew cli::array< System::Int32
>(4) { 1, 0, 0, 131072 });
			this->numericUpDown1->ValueChanged += gcnew System::EventHandler(this,
&MyForm::numericUpDown1_ValueChanged);
			//
			// numericUpDown2
			this->numericUpDown2->DecimalPlaces = 2;
			this->numericUpDown2->Font = (gcnew System::Drawing::Font(L"Times New Roman",
14.25F, System::Drawing::FontStyle::Italic, System::Drawing::GraphicsUnit::Point,
				static_cast<System::Byte>(204)));
			this->numericUpDown2->Location = System::Drawing::Point(59, 88);
			this->numericUpDown2->Maximum = System::Decimal(gcnew cli::array< System::Int32
>(4) { 10000, 0, 0, 0 });
			this->numericUpDown2->Name = L"numericUpDown2";
			this->numericUpDown2->Size = System::Drawing::Size(120, 29);
			this->numericUpDown2->TabIndex = 4;
			this->numericUpDown2->Visible = false;
			this->numericUpDown2->ValueChanged += gcnew System::EventHandler(this,
&MyForm::numericUpDown2_ValueChanged);
			//
			// button1
			this->button1->Enabled = false;
			this->button1->Location = System::Drawing::Point(18, 134);
			this->button1->Name = L"button1";
			this->button1->Size = System::Drawing::Size(75, 42);
			this->button1->TabIndex = 5;
			this->button1->Text = L"Попередній крок";
			this->button1->UseVisualStyleBackColor = true;
			this->button1->Click += gcnew System::EventHandler(this, &MyForm::button1_Click);
			//
			// button2
			this->button2->Location = System::Drawing::Point(104, 134);
			this->button2->Name = L"button2";
			this->button2->Size = System::Drawing::Size(75, 42);
```

```
this->button2->TabIndex = 6;
this->button2->Text = L"Наступний крок";
this->button2->UseVisualStyleBackColor = true;
this->button2->Click += gcnew System::EventHandler(this, &MyForm::button2_Click);
//
// groupBox2
this->groupBox2->Controls->Add(this->label8);
this->groupBox2->Location = System::Drawing::Point(10, 203);
this->groupBox2->Name = L"groupBox2";
this->groupBox2->Size = System::Drawing::Size(462, 146);
this->groupBox2->TabIndex = 7;
this->groupBox2->TabStop = false;
this->groupBox2->Text = L"Поточна дія";
//
// label8
this->label8->AutoSize = true;
this->label8->Font = (gcnew System::Drawing::Font(L"Times New Roman", 12,
System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,
    static_cast<System::Byte>(204)));
this->label8->Location = System::Drawing::Point(12, 26);
this->label8->Name = L"label8";
this->label8->Size = System::Drawing::Size(140, 19);
this->label8->TabIndex = 0;
this->label8->Text = L"Введіть значення λ.";
this->label8->Click += gcnew System::EventHandler(this, &MyForm::label8_Click);
//
// MyForm
this->AutoScaleDimensions = System::Drawing::SizeF(6, 13);
this->AutoScaleMode = System::Windows::Forms::AutoScaleMode::Font;
this->ClientSize = System::Drawing::Size(484, 371);
this->Controls->Add(this->groupBox2);
this->Controls->Add(this->button2);
this->Controls->Add(this->button1);
this->Controls->Add(this->numericUpDown2);
this->Controls->Add(this->numericUpDown1);
this->Controls->Add(this->label7);
this->Controls->Add(this->label1);
this->Controls->Add(this->groupBox1);
this->Name = L"MyForm";
this->Text = L"Метод послідовних поступок";
this->groupBox1->ResumeLayout(false);
this->groupBox1->PerformLayout();
```

```
			(cli::safe_cast<System::ComponentModel::ISupportInitialize^>(this->numericUpDown1))->EndInit();
			(cli::safe_cast<System::ComponentModel::ISupportInitialize^>(this->numericUpDown2))->EndInit();
			this->groupBox2->ResumeLayout(false);
			this->groupBox2->PerformLayout();
			this->ResumeLayout(false);
			this->PerformLayout();
		}
#pragma endregion

private: System::Void numericUpDown1_ValueChanged(System::Object^  sender, System::EventArgs^  e);
private: System::Void numericUpDown2_ValueChanged(System::Object^  sender, System::EventArgs^  e);
private: System::Void button1_Click(System::Object^  sender, System::EventArgs^  e);
private: System::Void button2_Click(System::Object^  sender, System::EventArgs^  e);
private: System::Void label3_Click(System::Object^  sender, System::EventArgs^  e){};
private: System::Void label6_Click(System::Object^  sender, System::EventArgs^  e){};
private: System::Void label8_Click(System::Object^  sender, System::EventArgs^  e){};
};
}

//MyForm.cpp

#include "MyForm.h"
#include <cmath>
#include <type_traits>

using namespace System;
using namespace System::Windows::Forms;

enum State {
	FIRST,
	SECOND,
	THIRD,
	FOURTH
};

float price= 150, trap_a=14, trap_b=21, trap_c=23, trap_d=27, gauss_c=19, gauss_sig=3.87;
float lambda, d_f, dw1min, dw1max, dw2;
State state = FIRST;
```

```
void SetDw() {
	dw1min = (trap_b - trap_a)*(1 - lambda);
	dw1max = (trap_d - trap_c)*(1 - lambda);
	dw2 = (gauss_sig*pow(log(pow(lambda, -1)), 0.5));
}

[STAThread]
void main(array<String^>^ arg) {
	Application::EnableVisualStyles();
	Application::SetCompatibleTextRenderingDefault(false);

	Diploma::MyForm form;
	Application::Run(%form);
}

System::Void Diploma::MyForm::numericUpDown1_ValueChanged(System::Object ^ sender,
System::EventArgs ^ e){
	return System::Void();
}

System::Void Diploma::MyForm::numericUpDown2_ValueChanged(System::Object ^ sender,
System::EventArgs ^ e){
	return System::Void();
}

System::Void Diploma::MyForm::button1_Click(System::Object ^ sender, System::EventArgs ^ e){
	switch (state) {
	case FOURTH:
	   state = THIRD;
	   button2->Enabled = true; numericUpDown2->ReadOnly = false;
	   label3->Text = (trap_c - trap_b) + "x1 + " + (dw1max + dw1min) + "Px1 + " + (roundf(dw2 *
200) / 100) + L"Px2 \u2192 min";
	   label8->Text = L"Мінімізуйте II критерій:\n" + label3->Text + L"\n x ≥ 0,    " + lambda + L" ≤ P
≤ 1\n Якщо отримана оцінка не задовольняє-\nоберіть величину поступки \u0394 f2";
	   break;
	case THIRD:
	   state = SECOND;
	   label7->Visible = false; numericUpDown2->Visible = false;
	   label8->Text = L"Знайдіть максимум I критерію на множині альтернатив:\n P \u2192 max\n x
≥ 0,    " + lambda + L" ≤ P ≤ 1";
	   break;
```

```
        case SECOND:
            state = FIRST;
            button1->Enabled = false; numericUpDown1->ReadOnly = false; groupBox1->Visible = false;
            label6->Text = L"0 ≤ P ≤ 1";
            label8->Text = L"Введіть значення λ.";
            break;}
        return System::Void();
}

System::Void Diploma::MyForm::button2_Click(System::Object ^ sender, System::EventArgs ^ e){
        switch (state) {
        case FIRST:
            state = SECOND;
            if (Convert::ToDouble(numericUpDown1->Value) > 1.) lambda = 1.;
            else lambda = Convert::ToDouble(numericUpDown1->Value);
            SetDw();
            button1->Enabled = true; numericUpDown1->ReadOnly = true; groupBox1->Visible = true;
            label3->Text = (trap_c-trap_b)+"x1 + "+(dw1max+dw1min)+"Px1 + "+ (roundf(dw2*200)/100)
+L"Px2 \u2192 min";
            label6->Text = lambda + L" ≤ P ≤ 1";
            label8->Text = L"Знайдіть максимум I критерію на множині альтернатив:\n P \u2192 max\n x
≥ 0,    " + lambda + L" ≤ P ≤ 1";
            break;
        case SECOND:
            state = THIRD;
            label7->Visible = true; numericUpDown2->Visible = true;
            label8->Text = L"Мінімізуйте II критерій:\n"+ label3->Text + L"\n x ≥ 0,    " + lambda + L" ≤ P
≤ 1\n Якщо отримана оцінка не задовольняє-\nоберіть величину поступки \u0394 f2";
            break;
        case THIRD:
            state = FOURTH;
            d_f = Convert::ToDouble(numericUpDown2->Value);
            button2->Enabled = false; numericUpDown2->ReadOnly = true;
            label3->Text = (trap_c - trap_b) + "x1 + " + (dw1max + dw1min) + "Px1 + " + (roundf(dw2 *
200) / 100) + L"Px2 ≤ " + d_f;
            label8->Text = L"Мінімізуйте III критерій на «уточненій» множині альтернатив:\n" + label4-
>Text + L"\n x ≥ 0,    " + lambda + L" ≤ P ≤ 1\n";
            break;}
        return System::Void();
}
```