НАВЧАЛЬНО-НАУКОВИЙ КОМПЛЕКС
«ІНСТИТУТ ПРИКЛАДНОГО СИСТЕМНОГО АНАЛІЗУ»
НАЦІОНАЛЬНОГО ТЕХНІЧНОГО УНІВЕРСИТЕТУ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»

КАФЕДРА МАТЕМАТИЧНИХ МЕТОДІВ СИСТЕМНОГО АНАЛІЗУ

КУРСОВА РОБОТА

З дисципліни

Теорія прийняття рішень

На тему “Дослідження задачі щодо вибору оптимального обсягу і напрямку експорту та імпорту”

Студент групи ІС-зп92
Єрмоленко Станіслав Олександрович

Керівник: Зайченко Юрій Петрович

Національна оцінка_________________
Кількість балів:___________ Оцінка: ECTS______

Члени комісії ________ __________________________
(підпис) (вчене звання, науковий ступінь, прізвище та ініціали)

____________ _____________________________________
(підпис) (вчене звання, науковий ступінь, прізвище та ініціали)

____________ _____________________________________
(підпис) (вчене звання, науковий ступінь, прізвище та ініціали)

Київ – 2021 рік

Зміст

ВСТУП

Задачі організаційного керування часто складаються з ситуацій, в яких вихідні умови задачі нечітко визначенні, що в свою чергу призводить до недостатньої інформованості особи яка приймає рішення(ОПР). Оскільки використовувана в таких задачах інформація може бути суб'єктивною та містить багато невизначеностей, опис цієї інформації засобами традиційної математики не давав можливості скласти точну математичну модель. Для застосування математичних методів аналізу та та дослідження складних систем був створений новий математичний апарат, який дав змогу формально описувати нечіткі поняття. Одним з напрямків нової теорії є проблема прийняття рішень при нечітких умовах та критеріях, що привела до появи нового напрямку у математичному програмування — нечіткого математичного програмування(НМП).

Постановка проблеми. Провести грунтовне дослідження завдання вибору оптимального обсягу і напрямку імпорту та експорту певної країни, опрацювавши завдання максимізації експорту, досягнувши при цьому мінімізації витрат на виробництво. Створений на основі цієї роботи програмний продукт має дозволити вирішувати подібні завдання при різних вихідних даних. При вирішенні задачі враховувалися обмеження на можливості виробництва продуктів країною на підприємствах для імпорту, потреби кожного суб'єкта, а також виробничий план імпорту та експорту. У роботі розглянута задача, де метою була максимізація експорту за умови збереження торгового балансу.

Мета курсової роботи: Імплементувати знання, отримані при вивченні дисциплін “Теорія прийняття рішень” та “Дослідження операцій”, набувши при цьому навичок вирішення оптимальним шляхом практичних задач, що виникають при взаємодії економічних, фінансових та виробничих систем.

ПОСТАНОВКА ЗАДАЧІ

У задачі ставиться мета знайти оптимальний план розміщення експортних і імпортних поставок за критерієм деякої країни в заданих валютних зонах. Розміщення припадають з урахуванням лінійних обмежень, пов'язаних, з одного боку, з експортно - імпортними потребами, з іншого - до заданих умов обміну, які передбачають певні ціни на товари і певний баланс між розглянутої країною A і даними валютними зонами.

Припустимо, що три валютні зони B_1, B_2, B_3 здійснюють торгівлю з країноюA. Країна A пропонує два товари T_1 і T_2, але їй потрібен товар T_3, пропонований кожної з трьох валютних зон.

Відомо, що країна A запланувала продати на зовнішньому ринку не більше 15 тис.т. товару T_1 і не більше 20 тис.т. товару T_2. Потрібно визначити, при якому плані розміщення зовнішньої торгівлі країна A буде імпортувати найбільшу кількість товару T_3 за умови торгового балансу. Крім того, потрібно визначити такий оптимальний план, при якому витрати на виробництво товарів, що експортуються країною Aтоварів мінімальні. Відомо, що витрати на виробництво однієї тонни T_1обчислюються величиною в три рази меншою, ніж витрати на виробництво однієї тонни товаруT_2. Такий оптимальний план будемо називати шуканим [1].

Всі інші дані представлені в табл.1.

Таблиця 1.

<table>
<tr><td rowspan="3">Варіант</td><td rowspan="3">Країна</td><td colspan="4">Верхні межі кількості товарів, імпортованих в країни тис.т і ціни, грн за одну тонну.</td><td colspan="2">Верхні межі кількості товарів, що експортуються валютними зонами тис.т і ціни, грн. за одну тонну.</td></tr>
<tr><td colspan="2">T1</td><td colspan="2">T2</td><td colspan="2">T3</td></tr>
<tr><td>кількість</td><td>ціна</td><td>кількість</td><td>ціна</td><td>кількість</td><td>ціна</td></tr>
<tr><td rowspan="3">1</td><td>B1</td><td>6000</td><td>50</td><td>8000</td><td>40</td><td>5000</td><td>20</td></tr>
<tr><td>B2</td><td>9000</td><td>40</td><td>6000</td><td>30</td><td>4000</td><td>25</td></tr>
<tr><td>B3</td><td>8000</td><td>38</td><td>7000</td><td>50</td><td>3000</td><td>30</td></tr>
</table>

Припустимо, що ціни товарів T_1, T_2, T_3, що імпортуються і експортуються в валютні зони B_1, B_2, B_3- c_{ij}є нечіткими числами з ФП

$$m_{ij}(c_{ij}) = \frac{1}{1+2\left(\frac{c_{ij}-\bar{c}_{ij}}{\bar{c}_{ij}}\right)^2}, i=\overline{1,3}; j=\overline{1,3},$$ де величини $\bar{c}_{ij}$ наведені в таблиці 1.6

Знайти множину оптимізуючих альтернатив, недомінінуємих зі ступенем $\alpha = 0.8$. Скласти відповідну модель завдання НМП і знайти її рішення для ЛПР, що використовує стратегії песиміста і оптиміста.

МАТЕМАТИЧНА МОДЕЛЬ ЗАДАЧІ

Дане завдання є завданням вибору оптимальних недоминирующих альтернатив, із застосуванням стратегії оптиміста і песиміста.

Позначимо через x_{ij}обсяги імпортованих або експортованих товарів j-го виду, для i-ой валютної зони, c_{ij} –ціна в грн. за тону товару j-го виду, для i-ой валютної зони, a_{ij}верхня межа обсягу товару j-го виду (імпорт або експорт), для i-ой валютної зони, d_j – верхня межа обсягу сумарного імпорту товару j-го виду (j = 1, 2), α ступінь недомінування.

Побудуємо формальну математичну модель для першої цільової функції

$$\max\left(\sum_{i=1}^{3} x_{i3}\right) \qquad (1)$$

$$\begin{gathered}
\sum_{i=1}^{3} x_{ij} \le d_j,\ j=\overline{1,2} \\
\sum_{j=1}^{2}\sum_{i=1}^{3} x_{ij} c_{ij} = \sum_{i=1}^{3} x_{i3} c_{i3} \quad (2) \\
\mu_{ij}\left(c_{ij}\right) \ge \alpha,\ i=\overline{1,3};\ j=\overline{1,3} \\
x_{ij} \le a_{ij}, \\
x_{ij} \ge 0,\ i=\overline{1,3},\ j=\overline{1,3}
\end{gathered}$$

Побудуємо формальну математичну модель для другої цільової функції

$$\min\left(\sum_{i=1}^{3} x_{i1} + 3\sum_{i=1}^{3} x_{i2}\right)$$

Далі будуємо завдання песиміста і оптиміста для знаходження інтервалу оцінки цільової функції. Для цього треба знайти інтервали значень c_{ij}

$$\mu_{ij}\left(c_{ij}\right) \geq \alpha ,$$

$$\mu_{ij}\left(c_{ij}\right) = \frac{1}{1+2\left(\frac{c_{ij}-\bar{c}_{ij}}{\bar{c}_{ij}}\right)^2} \geq \alpha ,$$

$$\bar{c}_{ij}\left(1-\sqrt{\frac{1-\alpha}{2\alpha}}\right) \leq c_{ij} \leq \bar{c}_{ij}\left(1+\sqrt{\frac{1-\alpha}{2\alpha}}\right)$$

$$i=\overline{1,3}\,;\ j=\overline{1,3}$$

Для першої цільової функції завдання песиміста - мінімальні ціни на експорт і максимальні на імпорт, завдання оптиміста - навпаки. Для другої цільової функції це не важливо, тому що там оптимальним завжди буде нульовий план.

Задача песиміста

$$\max\left(\sum_{i=1}^{3} x_{i3}\right)$$

$$\sum_{i=1}^{3} x_{ij} \le d_j,\ j=\overline{1,2}$$

$$\sum_{j=1}^{2}\sum_{i=1}^{3} x_{ij} c_{ij}\left(1-\sqrt{\frac{1-\alpha}{2\alpha}}\right)=\sum_{i=1}^{3} x_{i3} c_{i3}\left(1+\sqrt{\frac{1-\alpha}{2\alpha}}\right)$$

$$\mu_{ij}\left(c_{ij}\left(1-\sqrt{\frac{1-\alpha}{2\alpha}}\right)\right)\ge\alpha,\ i=\overline{1,3};\ j=\overline{1,3}$$

$$x_{ij}\le a_{ij},$$

$$x_{ij}\ge 0,\ i=\overline{1,3},\ j=\overline{1,3}$$

$$\min\left(\sum_{i=1}^{3} x_{i1}+3\sum_{i=1}^{3} x_{i2}\right)$$

$$\sum_{i=1}^{3} x_{ij} \le d_j,\ j=\overline{1,2}$$

$$\sum_{j=1}^{2}\sum_{i=1}^{3} x_{ij} c_{ij}\left(1-\sqrt{\frac{1-\alpha}{2\alpha}}\right)=\sum_{i=1}^{3} x_{i3} c_{i3}\left(1+\sqrt{\frac{1-\alpha}{2\alpha}}\right)$$

$$\mu_{ij}\left(c_{ij}\left(1+\sqrt{\frac{1-\alpha}{2\alpha}}\right)\right)\ge\alpha,\ i=\overline{1,3};\ j=\overline{1,3}$$

$$x_{ij}\le a_{ij},$$

$$x_{ij}\ge 0,\ i=\overline{1,3},\ j=\overline{1,3}$$

Задача оптиміста

$$\max\left(\sum_{i=1}^{3} x_{i3}\right)$$

$$\sum_{i=1}^{3} x_{ij} \le d_j,\ j=\overline{1,2}$$

$$\sum_{j=1}^{2}\sum_{i=1}^{3} x_{ij} c_{ij}\left(1+\sqrt{\frac{1-\alpha}{2\alpha}}\right)=\sum_{i=1}^{3} x_{i3} c_{i3}\left(1-\sqrt{\frac{1-\alpha}{2\alpha}}\right)$$

$$\mu_{ij}\left(c_{ij}\left(1+\sqrt{\frac{1-\alpha}{2\alpha}}\right)\right)\ge\alpha,\ i=\overline{1,3};\ j=\overline{1,3}$$

$$x_{ij}\le a_{ij},$$

$$x_{ij}\ge 0,\ i=\overline{1,3},\ j=\overline{1,3}$$

$$\min\left(\sum_{i=1}^{3} x_{i1}+3\sum_{i=1}^{3} x_{i2}\right)$$

$$\sum_{i=1}^{3} x_{ij} \le d_j,\ j=\overline{1,2}$$

$$\sum_{j=1}^{2}\sum_{i=1}^{3} x_{ij} c_{ij}\left(1+\sqrt{\frac{1-\alpha}{2\alpha}}\right)=\sum_{i=1}^{3} x_{i3} c_{i3}\left(1-\sqrt{\frac{1-\alpha}{2\alpha}}\right)$$

$$\mu_{ij}\left(c_{ij}\left(1-\sqrt{\frac{1-\alpha}{2\alpha}}\right)\right)\ge\alpha,\ i=\overline{1,3};\ j=\overline{1,3}$$

$$x_{ij}\le a_{ij},$$

$$x_{ij}\ge 0,\ i=\overline{1,3},\ j=\overline{1,3}$$

АЛГОРИТМ ВИРІШЕННЯ ЗАДАЧІ

Як і було сказано вище, для кожної цільової функції і відповідних обмежень вирішимо стандартну задачу лінійної оптимізації для крайніх значень інтервалів цін - отримаємо найгірші і найкращі значення для цільової функції і відповідні плани. Вирішувати завдання лінійної оптимізації будемо модифікованим симплекс-методом, його інша назва - метод оберненої матриці. Перевага методу оберненої матриці перед звичайним симплекс-методом в стислості обчислень і меншому використанні пам'яті.

Загальний алгоритм:

Приводимо задачу до канонічної форми:

1. Вводимо додаткові змінні для обмежень виду ,≤
2. Вводимо штучні змінні для обмежень виду і їх же для цільової функції з великими штрафними коефіцієнтами.

Вирішуємо загальну задачу лінійного програмування в канонічній формі за допомогою модифікованого симплекс-методу.

АЛГОРИТМ МОДИФІКОВАННОГО СИМПЛЕКС-МЕТОДУ

В модифікованному методі матриця не перераховається, зберігається і перераховується тільки матриця B^{-1}. В іншому алгоритм схожий на звичайний симплекс-метод.

$$\begin{bmatrix} 1 & c_B^T B^{-1} A - c^T & c_B^T B^{-1} \\ 0 & B^{-1} A & B^{-1} \end{bmatrix}$$

1. Обчислюємо двоїсті змінні $d = c_B^T B^{-1}$

2. Перевірка оптимальності. $c_B^T B^{-1} A - c^T$ перетвориться в $d^T A - c^T$

Перевірка полягає в обчисленні $d^T A_n - c_n$ для всіх стовпців $n \in N$. Стовпець із значенням <0 можна вводити в базис.

Часто вибирають мінімальне значення, але для цього потрібно перебрати всі стовпці

Найчастіше вибирають значення, менше деякого заданого значення $-\varepsilon$

Якщо такого стовпчика не виявиться, за $-\varepsilon$ приймається максимальне знайдене абсолютне значення і відповідний стовпець A_J вводиться в базис.

3. Визначення виведеного.

Нехай A_J - стовпець який вводиться, відповідний змінній x_J Базисний план - це рішення системи $A_B p = b$ Збільшуємо $A_B p + \vartheta A_J = b$

Помножимо зліва на B^{-1}, $B^{-1} A_B p + \vartheta B^{-1} A_J = B^{-1} b$

Тут $B^{-1} b$- базисний план, $q = B^{-1} A_J$- розкладання вводиться стовпчика по базису.

Знаходимо максимальне значення ϑ, при якому всі значення не негативні. Якщо ϑ може бути взято як завгодно велике, рішення не обмежене. В іншому випадку один з елементів вийде на нульове значення. Виводимо відповідний стовпець з базису.

4. Перерахунок опорного (базисного) плану.

Обчислюємо новий опорний план по вже наведеною формулою

$x = p + \vartheta B^{-1} A_J$ зі знайденим значенням ϑ.

5. Перераховуємо зворотну до базисної B^{-1}

Нехай $B^{-1} A_F$ це стовбець який виводиться

Матриця В подана в вигляді $[B_G A_F]$

де B_G - базисна матриця без виведеного стовпчика.

Після заміни стовпця базисна матриця буде мати вигляд $[B_G A_J]$

Нам потрібно знайти матрицю B_1, таку що

$[B_G, A_J] B_1^{-1} = E$ => $[B^{-1} B_G B_1^{-1}, B^{-1} A_J] = B^{-1}$ =>

$[B^{-1} B_G, q] B_1^{-1} = B^{-1}$ => $\begin{bmatrix} E & q' \\ 0 & q_f \end{bmatrix} B_1^{-1} = B^{-1}$

Звідки

$$B_1^{-1} = \begin{bmatrix} E & -q'/q_f \\ 0 & 1/q_f \end{bmatrix} B^{-1}$$

СТРАТЕГІЯ ПЕСИМІСТА ТА ОПТИМІСТА ДЛЯ ЗАДАЧІ НМП

Формулювання завдання і її зведення до загальної задачі НМП.

Нехай X - універсальна множина альтернатив. Підмножина допустимих альтернатив описується нерівностями, що випливають з (3.2.4):

$$g_i(\mathbf{x}, a_{i1}, a_{i2}, \ldots a_{in}) \le 0, i = \overline{1,m} \quad (3.\ 2.\ 5)$$

де g_i - задані функції, $X \times R^n \longrightarrow R^1$, a_{ij}, $i = \overline{1,n}, j = \overline{1,m}$, - числові параметри, значення яких описані в формі нечітких множин числової осі.

Нехай $\nu_{ij}(a_{ij})$ - задані функції приналежності цих нечітких множин.

Вибори альтернатив оцінюються значеннями заданої функції

$f : X \times R^n \longrightarrow R^1$ виду $f(\mathbf{x}, c_1, c_2, \ldots, c_n)$, в якій $c_1, c_2, \ldots, c_n$ - числові параметри, значення яких описані також у формі нечітких підмножин числової осі.

Нехай $\mu_j(c_j)$ - задані функції приналежності цих нечітких множин. Для завершення формулювання завдання необхідно відшукати відношення переваги в універсальній множині оцінок альтернатив. В даному випадку ця універсальна множина являє собою числову вісь R^1. Будемо вважати, що вихідне відношення переваги являє нестрогий порядок $\ge$ на R^1. Побудуємо математичну модель задачі у вигляді спільної задачі НМП.

Для цього розглянемо спочатку нечітке відношення (3.2.4) і побудуємо відповідне їм нечітке підмножина допустимих альтернатив, функцію приналежності якого будемо позначати $\mu_C(\mathbf{x})$.

Нехай $\{a_{ij}\}, i = \overline{1,n}, j = \overline{1,m}$ - деякі конкретні числові значення відповідних параметрів в обмеженнях (3.2.4), ступеня їх приналежності заданим нечітким множинам рівні відповідно

$$\nu_{ij}(a_{ij}^0), i = \overline{1,n}, j = \overline{1,m}$$

Позначимо через μ_0 мінімальне з цих чисел, тобто $\mu^0 = \min_i \nu_{ij}(a_{ij}^0)$. Якщо деяка альтернатива $\tilde{\mathbf{x}} \in X$ задовольняє нерівностям $g_i(\tilde{\mathbf{x}}, a_{i1}^0, a_{i2}^0, \dots a_{in}^0) \le 0, \ i = \overline{1,m}$, то природно вважати, що ця альтернатива належить множині допустимих альтернатив зі ступенем, не меншою, μ^0 тобто вважати, що $\mu_C(\tilde{\mathbf{x}}) \ge \mu^0$ Ця нерівність задає безліч допустимих альтернатив. Для зручності запису її функції приналежності введемо такі позначення:

$$\nu(\mathbf{A}) = \min_{(i,j)} \nu_{ij}(a_{ij}), \mathbf{A} = \|a_{ij}\|, i = \overline{1,m}, j = \overline{1,n},$$

$$P(\mathbf{x}) = \{\mathbf{A} = \|a_{ij}\|_{j=\overline{1,n}}^{i=\overline{1,m}} \cdot g_i(a_{i1}, a_{i2}, \dots, a_{in}, \mathbf{x}) \le 0; i = \overline{1,m}\} \qquad (3.2.6)$$

У цих позначеннях отримаємо

$$\mu_C(\mathbf{x}) = \sup_{A \in P(\mathbf{x})} \nu(\mathbf{A})$$

Кожній альтернативі $\tilde{\mathbf{x}}$ функція $\mu_C(\mathbf{x})$ ставить у відповідність ступінь її допустимості з урахуванням нечіткої інформації.

Розглянемо нечітко задану функцію $f(\mathbf{x}, c_1, c_2, \dots, c_n)$ і представимо її у вигляді нечіткої цільової функції аналогічно тому, як це було виконано для обмежень.

Нехай $c_j^0, j=\overline{1,n}$ - деякі конкретні числові значення параметрів функції (3.2.3), ступеня їх приналежності заданим нечітким множинам рівні відповідно $\chi_j(c_j), j=\overline{1,n}$.

Нехай φ - мінімальне з них : $\varphi = \min\limits_j x_j(c_j)$. Нехай, нарешті, $\tilde{\mathbf{x}} \in X$ - деяка альтернатива і число $r_0 = f(\tilde{\mathbf{x}}, c_1^0, c_2^0, \ldots c_n^0)$ являє собою відповідне альтернативі $\tilde{\mathbf{x}}$ і значенням параметрів $\{c_j^0\}, j=\overline{1,n}$ значенням функції (3.2.3). Природно вважати, що це значення r^0 належить нечіткої оцінки альтернативи $\tilde{\mathbf{x}}$ зі ступенем, не меншою φ. Звідси шукана нечітка цільова функція $\varphi(\mathbf{x},r)$ має вид

$$\varphi(\mathbf{x},r) = \sup_{\mathbf{c}\in Q(\mathbf{x},r)} \chi(\mathbf{c}) \qquad (3.2.8)$$

де

$$\chi(\mathbf{c}) = \min_j \chi_j(c_j), \mathbf{c} = [c_1, c_2, ..., c_n]^T; Q(\mathbf{x}, r) = \{\mathbf{c} : \mathbf{c} \in R^n, f(\mathbf{x}, \mathbf{c}) = r\}$$

Остаточно отримуємо, що вихідна задача з нечітко описаними параметрами формулюється у вигляді такої спільної справи НМП [20; 37] - максимізувати нечітко задану цільову функцію

$$\varphi(\mathbf{x},r) = \sup_{\mathbf{c}\in Q(\mathbf{x},r)} \chi(\mathbf{c}) \qquad (3.2.9)$$

на нечіткій множині допустимих альтернатив

$$\mu_C(\mathbf{x}) = \sup_{\mathbf{A}\in P(\mathbf{x})} \nu(\mathbf{A}) \qquad (3.2.10)$$

де $Q(\mathbf{x},r)$ та $P(\mathbf{x})$ задаються відповідно (3.2.8) и (3.2.6).

Недомінуючі альтернативи в загальній задачі нечіткого математичного програмування.

1. Розглянемо спочатку більш просту задачу з нечіткою цільовою функцією (3.2.3) і звичайною (чіткою) множиною допустимих альтернатив, заданих нерівностями $g_i(\mathbf{x}, a_{i1}, a_{i2}, \dots a_{in}) \le 0, i = \overline{1,m}$ з точно відомими значеннями параметрів a_{ij}.

Функція $\varphi(\mathbf{x}, r)$ і природний порядок $(\ge)$ на числової осі генерують на множині альтернатив X узагальнене н.о.п. виду η

$$\eta(x_1, x_2) = \sup_{\substack{z,y \in R \\ z \ge y}} \min\{\varphi(x_1, z), \varphi(x_2, y)\} = \\ = \sup_{\substack{z,y \in R \\ z \ge y}} \min\{\sup_{c \in Q(x_1,z)} \chi(c), \sup_{c \in Q(x_2,y)} \chi(c)\} \qquad (3.2.11)$$

Нехай $\eta^{нд}(\mathbf{x})$ - відповідна множина недомініруемих альтернатив в множині X з н.о.п. η. Виберемо деякий число α з інтервалу $0 \le \alpha \le 1$ і розглянемо задачу знаходження альтернативи, ступінь недомінуємості якої не менше α, тобто $\eta^{нд}(\mathbf{x}) \ge \alpha$. Будемо припускати, що всі вихідні нечіткі множини $\chi_j(c_j)$ такі, що $\sup_j \chi_j(c_j) \ge \alpha, j = \overline{1,n}$ і покажемо, що при цьому функція $\varphi(\mathbf{x}, r)$ має властивість:

$$\sup_{r \in R^1} \varphi(\mathbf{x}, r) \ge \alpha, \forall \mathbf{x} \in X \qquad (3.2.12)$$

Припустимо: знайдеться такий $\tilde{x} \in X$, для якого $\sup_r \varphi(\tilde{x}, r) = d < \alpha$

Це означає (див. (9.5.15)), що при будь-якому $r \in R^1$ і будь-якому $\mathbf{c} \in Q(\tilde{\mathbf{x}}, r)$ виконується нерівність $\chi(\mathbf{c}) \le d < \alpha$ (3.2.14)

Виберемо довільне число ε з інтервалу $0 \le \varepsilon \le \alpha - d$ і нехай $\tilde{c}_j, j = \overline{1,n}$ такі, що $\chi_j(c_j) > \alpha - \varepsilon$.

Отримаємо

$\chi(\tilde{\mathbf{c}}) = \min \chi_j(\tilde{c}_j) > \alpha - \varepsilon$

Разом з тим для $\tilde{r} = f(\tilde{\mathbf{x}}, \tilde{c}_1, \tilde{c}_2, ..., \tilde{c}_n)$ виконується умова $\tilde{\mathbf{c}} \in Q(\tilde{\mathbf{x}}, \tilde{r})$. Однак в силу вибору ε нерівність (3.2.14) не виконується для $\tilde{r} \in R^1$ і, отже, не виконується нерівність (3.2.13).

Таким чином, якщо всі задані нечіткі множини c_j такі, що мають $\sup_{c_j} x_j(c_j) \ge \alpha$, то виконуються всі умови теореми 9.4. і тому для знаходження альтернатив, ступінь недомініруемих яких не менше α, досить вирішити наступне завдання:

максимізувати r (3.2.15)

за виконання наступних умов:

$\varphi(\tilde{\mathbf{x}}, r) \ge \alpha$

$g_i(\tilde{\mathbf{x}}, a_{i1}, ..., a_{in}) \le 0, i = \overline{1,m}, r \in R^1$ (3.2.16)

Припустимо, що множина X компактна, безперервна на R^1, функція $f(\mathbf{x}, c_1, c_2, ..., c_n)$ також неперервна на творі $X \times R^n$. Неважко показати, що при цих умовах завдання (3.2.15), (3.2.16) еквівалентна наступній завданню [20]:

максимізувати $f(\mathbf{x}, c_1, c_2, ..., c_n)$ (3.2.17)

при обмеженнях

$$\chi_j(c_j) \ge \alpha;$$
$$g_i(\mathbf{x}, a_{i1}, ... a_{in}) \le 0, i = \overline{1,m}. \quad (3.2.18)$$

Дійсно, нехай пара $(\mathbf{x}^0, r^0)$ - рішення задачі (3.2.15), (3.2.16). У прийнятих вище припущеннях функція $\chi(\mathbf{c})$ неперервна на R^n, а множину $Q(\mathbf{x}, r)$ замкнуто в R^1. Нехай вектор $\mathbf{c}^0 \in Q(\mathbf{x}, r)$ такий, що $\chi(\mathbf{c}^0) \geq \alpha$. Тоді за визначенням множини $Q(\mathbf{x}, r)$ отримуємо, що $f(\mathbf{x}^0, c_1^0, c_2^0, \ldots c_n^0) = r^0$. Припустимо, що $(\mathbf{x}^0, \mathbf{c}^0)$ не є вирішенням завдання (3.2.17), тобто знайдуться $(\mathbf{x}', \mathbf{c}')$, що задовольняють (3.2.18), такі, що $f(\mathbf{x}', c_1', c_2', \ldots, c_n') = r' > r^0$. Це означає, що $\mathbf{c}' \in Q(\mathbf{x}', r')$ і $\varphi(\mathbf{x}', r') \geq \alpha$, тобто пара $(\mathbf{x}', r')$ задовольняє обмеженням (3.2.16). Але тоді не може бути $r' > r^0$, так як $(\mathbf{x}^0, r^0)$ - рішення задачі (3.2.15).

Аналогічно можна показати, що будь-яке рішення задачі (3.2.17), (3.2.18) є також рішенням завдання (3.2.15).

2. Розглянемо загальну задачу НМП, в якій нечітко задані параметри c_j функції і параметри a_{ij} обмежень (3.2.5). Як показано вище, це завдання можна записати у вигляді (3.2.15), (3.2.16).

Насамперед зазначимо, що в даній задачі вибір альтернатив може здійснюватися з урахуванням двох відносин: нечіткого, індукованого функцією $\varphi(\mathbf{x}, r)$ (3.2.8), і чіткого, індукованого функцією $\mu_C(\mathbf{x})$ і природним порядком на R^1.

Як було показано в розд. 9.5, функція $\varphi(\mathbf{x}, r)$ і природний порядок $(\geq)$ на R^1 індукують на X нечітке відношення переваги виду

$$\eta_1(\mathbf{x}_1, \mathbf{x}_2) = \sup_{\substack{(y,z) \in R^1 \\ y \geq z}} \min\{\varphi(\mathbf{x}_1, y), \varphi(\mathbf{x}_2, z)\} \qquad (3.2.19)$$

Друге відношення на X визначається тим, що кращими є альтернативи, які мають велику ступінь допустимості, тобто ті, яким відповідають великі значення функції.

Таким чином,

$$\eta_2(\mathbf{x_1},\mathbf{x_2}) = \begin{cases} 1, npu & \mu_C(\mathbf{x_1}) \ge \mu_C(\mathbf{x_2}); \\ 0, npu & \mu_C(\mathbf{x_1}) < \mu_C(\mathbf{x_2}). \end{cases}$$

Завдання такого типу розглядалися в розд. 9.4, де запропонована процедура побудови підмножини недомініруемих альтернатив при наявності декількох критеріїв (відносин переваги). Відповідно до неї потрібно побудувати дві згортки вихідних н.о.п. η_1 і η_2 - їх перетин і зважену суму.

Пересічення відношень η_1, η_2 має вигляд

$$Q_1(\mathbf{x_1},\mathbf{x_2}) = \min\{\eta_1(\mathbf{x_1},\mathbf{x_2}), \eta_2(\mathbf{x_1},\mathbf{x_2})\} \qquad (3.2.20)$$

Зважена сума критеріїв за умови рівності вагових коефіцієнтів відносин η_1, η_2

$$Q_2(\mathbf{x_1},\mathbf{x_2}) = \frac{1}{2}[\eta_1(\mathbf{x_1},\mathbf{x_2}) + \eta_2(\mathbf{x_1},\mathbf{x_2})] \qquad (3.2.21)$$

Нехай $Q_1^{нд}(\mathbf{x})$, $Q_2^{нд}(\mathbf{x})$ - нечіткі підмножини недомініруемих альтернатив множин $(\mathbf{x},Q_1)$, $(\mathbf{x},Q_2)$ відповідно. Тоді результуюча підмножина недомінуємих альтернатив має вигляд

$$Q^{нд}(\mathbf{x}) = \min\{Q_1^{нд}(\mathbf{x}); Q_2^{нд}(\mathbf{x})\} \qquad (3.2.22)$$

Функція приналежності $Q^{нд}(\mathbf{x})$ служить основою для вибору конкретних альтернатив в загальній завдачі НМП.

У цьому завданню, як і раніше, становить практичний інтерес питання про знаходження альтернатив $\mathbf{x} \in X$, для яких $Q^{нд}(\mathbf{x}) \ge \alpha$, де $\alpha \in [0;1]$. Однак для спрощення спочатку розглянемо задачу знаходження ч.нд. альтернатив (тобто таких, що $Q^{нд}(\mathbf{x}) = 1$ $\alpha = 1$).

Отже, розглянемо задачу визначення ч.нд. альтернативи, для якої $Q^{нд}(\mathbf{x}) = 1$

З (3.2.22) отримуємо, що для цього необхідно і достатньо виконання умов $Q_1^{нд}(\mathbf{x}) = 1$, $Q_2^{нд}(\mathbf{x}) = 1$.

З'ясуємо спочатку умови, при яких $\mathbf{x}$ є ч.нд. альтернатива в множині $(\mathbf{x}, Q_1)$, тобто коли $Q_1^{нд}(\mathbf{x}) = 1$.

Якщо вихідні нечіткі множини $\chi_j(c_j),\ j = \overline{1,n}$ нормальні, тобто коли $\sup\limits_{c_j} \chi_j(c_j) = 1$, то, як показано вище, при $\alpha = 1$ функція $\varphi(\mathbf{x}, r)$ в (3.2.8) має властивість $\sup\limits_{r \in R^1} \varphi(\mathbf{x}, r) = 1$ при будь-якому $\mathbf{x} \in X$.

Звідси з теореми 9.2 випливає, що н.о.п. η_1-сильно лінійне відношення. Неважко бачити, що η_2 - також сильно лінійне відношення. Однак їх перетин Q_1 цієї властивості в загальному випадку не має.

Нехай $X_2^{чнд}$ -подмножіни всіх ч.нд. альтернатив множини $(\mathbf{x}, \eta_2)$, тобто множина всіх $\mathbf{x}^0 \in X$, для яких рівність $\eta_2(\mathbf{x}^0, \mathbf{x}) = 1$ виконується при будь-якому $\mathbf{x} \in X$. Можна показати, що ч.нд. альтернативи множини $(\mathbf{x}, Q_1)$ досить шукати серед альтернатив множини $X_2^{чнд}$. Нехай Q_1-нечеткое відношення строгої переваги, відповідне н.о.п. Q_1 і $\mathbf{x_0} \in X_2^{чнд}$. Покажемо, що $Q_1(\mathbf{x}, \mathbf{x_0}) = 0$ при будь-якому $\mathbf{x_0} \in X_2^{чнд}$. Припустимо, що знайдеться така альтернатива $\mathbf{x_1} \notin X_2^{чнд}$, для якої $Q_1(\mathbf{x_1}, \mathbf{x_0}) > 0$ тобто $\min\{\eta_1(\mathbf{x_1}, \mathbf{x_0}), \eta_2(\mathbf{x_1}, \mathbf{x_0})\} - \min\{\eta_1(\mathbf{x_0}, \mathbf{x_1}), \eta_2(\mathbf{x_0}, \mathbf{x_1})\} > 0$ (3.2.23)

Але, оскільки $\mathbf{x_1} \notin X_2^{чнд}$, а $\mathbf{x_0} \in X_2^{чнд}$ то $\eta_2(\mathbf{x_1}, \mathbf{x_0}) = 0$ і нерівність (3.2.23) неможлива, тобто виконується $Q_1^S(\mathbf{x}, \mathbf{x}_0) = 0$.

Таким чином, жодна альтернатива $\mathbf{x} \notin X_2^{чнд}$ не домінує зі строго позитивним ступенем альтернативи з множини $X_2^{чнд}$.

Покажемо тепер, что если $\mathbf{x_0}$ - ч.нд. альтернатива в множині (тобто на множині $X_2^{чнд}$ по відношенню η_1), то вона є такою и в множині $(\mathbf{x}, Q_1)$.

Дійсно, як показано вище, при цьому $Q_1^S(\mathbf{x}, \mathbf{x}_0) = 0$ для будь-якого $\mathbf{x} \notin X_2^{чнд} \subseteq X$.

Нехай $\mathbf{x} \in X_2^{чнд}$, тоді з того, що $\mathbf{x_1}$ - ч.нд. альтернатива в множині $(X_2^{чнд}, \eta_1)$ і η_1 - сильно лінійне н.о.п., отримуємо відповідно до теореми 9.3, що $\eta_1(\mathbf{x_1}, \tilde{\mathbf{x}}) = 1$.

Крім того, $\eta_2(\mathbf{x_1}, \tilde{\mathbf{x}}) = 1$, оскільки $\mathbf{x_1} \in X_2^{чнд}$. Звідси, робимо висновок, що нерівність $Q_1^S(\tilde{\mathbf{x}}, \mathbf{x_1}) = \min\{\eta_1(\tilde{\mathbf{x}}, \mathbf{x_1}), \eta_2(\tilde{\mathbf{x}}, \mathbf{x_1})\} - \min\{\eta_1(\mathbf{x_1}, \tilde{\mathbf{x}}), \ \eta_2(\mathbf{x_1}, \tilde{\mathbf{x}})\} > 0$ неможлива.

Отже, якщо $\mathbf{x_1}$ вибрано описаним вище способом, то рівність $Q_1^S(\mathbf{x}, \mathbf{x_1}) = 0$ виконується при будь-якому $\mathbf{x} \in X$, значить - ч.нд. альтернатива в множині $(\mathbf{X}, Q_1)$.

Таким чином, для знаходження ч.нд. альтернативи в множині X з н.о.п. Q_1 досить знайти ч.нд. альтернативу в множині $X_2^{чнд}$ з н.о.п. η_1

Нехай вихідні нечіткі множини $v_{ij}(a_{ij}), i = \overline{1,m}, j = \overline{1,n}$ такі, що існують $\{a_{ij}^0\}$, для яких $v_{ij}(a^0{}_{ij}) = 1$. При цьому множина складає альтернативи, які задовольняють умовам

$$\begin{aligned} &g_i(\mathbf{x}, a_{i1}, a_{i2}, \dots, a_{in}) \le 0, i = \overline{1,m}, \\ &v_{ij}(a_{ij}) = 1, i = \overline{1,m}, j = \overline{1,n}. \end{aligned} \qquad (3.2.24)$$

Завдання знаходження ч.нд. альтернативи аналогічне завданню, розглянутої в п.1. даного розділу.

Якщо виконуються введені в ній припущення (компактність множини X, неперервність функції f), то приходимо до наступної задачі для знаходження ч.нд. альтернативи в безлічі [20; 37]

максимізувати $f(\mathbf{x}, c_1, c_2, \dots, c_n)$ (3.2.25) при обмеженнях

$$g_i(\mathbf{x}, a_{i1}, a_{i2}, ..., a_{in}) \le 0, a_{ij} \in R^1,$$
$$\chi_j(c_j) = 1, j = \overline{1, n},$$
$$\nu_{ij}(a_{ij}) = 1, i = \overline{1, m}, j = \overline{1, n}. \qquad (3.2.26)$$

Будь-яке рішення $\mathbf{x}^0 \in X$ задачі (3.2.25), (3.2.26) є ч.нд. альтернатива в множині (X, Q_1), і в той же час ч.нд. альтернатива в множинах (X, η_1) і (X, η_2). Покажемо, що будь-яке рішення цього завдання є ч.нд. альтернатива і в множині $(\mathbf{x}, Q_2)$.

Дійсно, нечітке підмножина недомінуємих альтернатив множини X з н.о.п. Q_2 має вигляд

$$Q_2^{нд}(\mathbf{x}) = 1 - \frac{1}{2}\sup_{\mathbf{y}}[\eta_1(\mathbf{y}, \mathbf{x}) - \eta_1(\mathbf{x}, \mathbf{y})] + [\eta_2(\mathbf{y}, \mathbf{x}) - \eta_2(\mathbf{x}, \mathbf{y})] \qquad (3.2.27)$$

Нехай $\mathbf{x}^0$ рішення задачі (3.2.25), (3.2.26). Тоді, оскільки $\mathbf{x}^0$ - ч.нд. альтернатива в $(\mathbf{x}, Q_1)$, то при будь-якому $\mathbf{y} \in X$ виконується умова $\eta_1^S(\mathbf{y}, \mathbf{x}^0) = 0$, або $\eta_1(\mathbf{y}, \mathbf{x}^0) - \eta_1(\mathbf{x}^0, \mathbf{y}) \le 0$. Крім того $\mathbf{x}^0$ -ч.нд. альтернатива в множині $(\mathbf{x}, \eta_2)$, і, отже, при будь-якому $\mathbf{y} \in X$ виконується умова $\eta_2^S(\mathbf{y}, \mathbf{x}^0) = 0$ або $\eta_2(\mathbf{y}, \mathbf{x}^0) - \eta_2(\mathbf{x}^0, \mathbf{y}) \le 0$

З двох останніх варіантів виходить, що в (3.2.27)

$$\sup[\eta_1(\mathbf{y}, \mathbf{x}^0) - \eta_1(\mathbf{x}^0, \mathbf{y}) + \eta_2(\mathbf{y}, \mathbf{x}^0) - \eta_2(\mathbf{x}^0, \mathbf{y})] = 0 \qquad (3.2.28)$$

Отже, $Q_2^{нд}(\mathbf{x}^0) = 1$

Таким чином, будь-яке рішення задачі (3.2.25), (3.2.26) відповідає умовам, $Q_1^{нд}(\mathbf{x}^0) = 1$, $Q_2^{нд}(\mathbf{x}^0) = 1$ тобто $\mathbf{x}^0$ -ч.нд. альтернатива для вихідної загальної задачі НМП.

Наявність в отриманої задачі НМП (3.2.25), (3.2.26) обмежень виду $\chi_j(c_j) = 1, \nu_{ij}(a_{ij}) = 1$ свідчить про те, що для знаходження ч.нд. альтернатив у вихідній задачі досить враховувати лише ті значення параметрів, які завідомо, тобто зі ступенем 1, належать відповідним нечітким множинам. Тобто, якщо шукати лише ч.нд. альтернативи, то при формулюванні початкової задачі можна не вимагати повного опису нечітких множин значень параметрів, а обмежитися лише вказівкою інтервалів їх значень зі ступенем 1, що належать цим множинам.

Аналогічними міркуваннями можна прийти до висновку про те, що для знаходження альтернативи, має ступінь недомінуємості не меншу ніж α (тобто $\eta^{нд}(\mathbf{x}^0) \geq \alpha$) досить вирішити наступне завдання математичного програмування [20]:

максимізувати $f(\mathbf{x}, c_1, c_2, ..., c_n)$ (3.2.29)

при обмеженнях (3.2.30)

$$g_i(\mathbf{x}, a_{i1}, a_{i2}, ..., a_{in}) \leq 0, a_{ij} \in R^1;$$
$$\chi_j(c_j) \geq \alpha, j = \overline{1, n};$$
$$\nu_{ij}(a_{ij}) \geq \alpha, i = \overline{1, m}, j = \overline{1, n}. \quad (3.2.31)$$

Рішення завдання типу (3.2.29) - (3.2.31) дозволяє визначити лише деякі з недомініруемих альтернатив з відповідним ступенем α для вихідної загальної задачі НМП.

Можна показати, що ще один спосіб знаходження недомініруемих альтернатив зі ступенем α полягає у вирішенні наступного завдання:

знайти $\max\limits_{\mathbf{x}} \min\limits_{c_1, ..., c_n} f(\mathbf{x}, c_1, ..., c_n)$ (3.2.32)

при обмеженнях (3.2.30), (3.2.31) Отримані в цьому завданні значення функції $f(\mathbf{x}, c)$ є найбільш обережною оцінкою альтернатив, що не домінованих зі ступенем, не меншим, ніж α, а відповідне значення в задачі (3.2. 29) - (3.2.31) - найменш обережну оцінку. Інтервал між цими оцінками дозволяє судити про можливі значеннях функції $f(\mathbf{x})$ при виборі альтернатив, недомінуємих зі ступенем α.

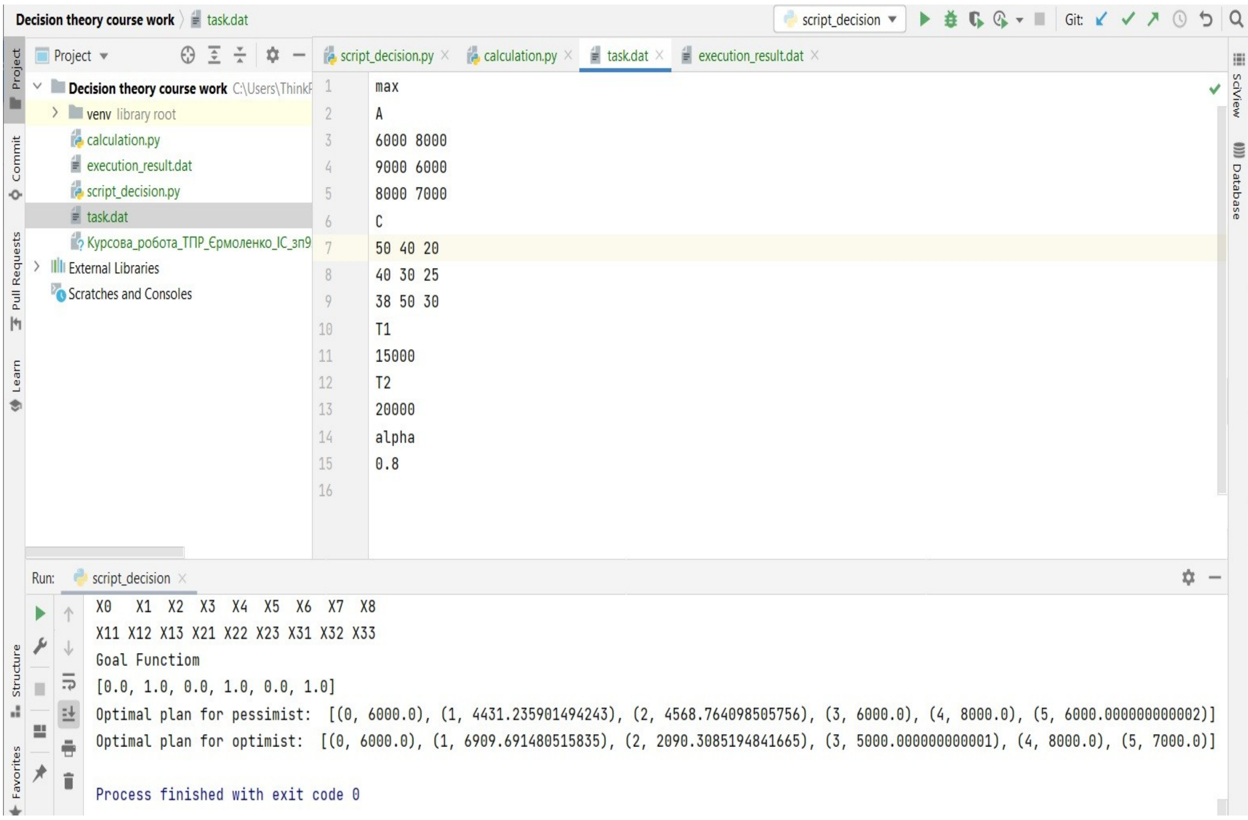
ОПИС ПРОГРАМИ

Програма була розроблена на мові програмування Python версії 3.7 в середовищі PyCharm. Програма складається з файлу script_decision.py, додаткового файлу calculations.py та двох файлів: task.dat містить умову задачі, execution_result.dat виводить результат роботи програми в окремий файл. Повний лістинг програми наведений окремо в додатку 1.

КЕРІВНИЦТВО КОРИСТУВАЧА

Для запуску програми на комп’ютері має бути встановлений Python версії 3.7, рекомендується встановити IDE PyCharm Community Edition Запуск програми здійснюється за допомогою запуску файлу script_decision.py, вихідні дані знаходяться у файлі task.dat, інтерфейс представлено на рисунку 1

Рисунок 1



АНАЛІЗ ОТРИМАНИХ РЕЗУЛЬТАТІВ

Результати для першої цільової функції

F_{pes} = 16431; X0 = 6000; X1 = 4431; X2 = 4568; X3 = 6000; X4 = 8000; X5 = 6000

F_{opt} = 18909; X0 = 6000; X1 = 6909; X2 = 2090; X3 = 5000, X4 = 8000; X5 = 7000

Значення цільової функції не співпадають, і хоча для обох задач використовується однакова структура імпорту, в задачі песиміста взяті, мінімальні ціни для експорту і максимальні для імпорту, в задачі оптиміста, навпаки, взяті мінімальні ціни для імпорту і максимальні для експорту і якщо ми взагалі прибираемо обмеження по кількості на імпорт то оптиміст буде експортувати максимально доступну кількість товару.

ВИСНОВКИ

В процесі виконання курсової роботи був детально опрацьований матеріал з нечіткого математичного програмування та вирішена задача, яка може мати практичне значення в контексті дій особи приймаючою рішення(ОПР) щодо оптимального варіанту в рамках вибору економічної стратегії.

Також в курсовій роботі було проведено дослідження задач про вибір торгового плану для міжнародної торгівлі між певною країною і валютними зонами, кожен суб'єкт має попит або пропозицію набору товарів, обсяги обмежені а ціни є нечіткими, для них задані функції приналежності. Отримані результати свідчать що недомінуючі альтернативи в більшій мірі залежать від обмежень і цільової функції ніж від степені альфа та показують що модель не чуттєва до функцій приналежності. В процесі програмної реалізації були також удосконалені навички програмування на Python.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Ю.П.Зайченко. Дослідження операцій. Підручник. Сьоме видання, перероблене та доповнене. = К.: Видавничий Дім «Слово», 2006. – 584с.
2. О.Ю.Зайченко. Дослідження операцій. Збірник задач. /О.Ю.Зайченко, Ю.П.Зайченко. – К.: Видавничий Дім «Слово», 2006. – 472с.
3. Ю.П.Зайченко. Теорія прийняття рішень: підручник. / Ю. П. Зайченко. - К.:НТУУ "КПІ", 2014. – 412 с. – Бібліогр.: с. 537
4. Nelder J. A simplex method for function minimazation / J. Nelder, R. Mead // The Computer Journal – 1965 – №7 с. 308 — 313
5. Кононюк А.Е. Основы теории оптимизации. Безусловная оптимизация К.2.ч.1. Киев: "Освіта України", 2011. - 544 с.
6. М. Ю. Черняк, М. С. Эльберг Методы оптимизации в технике. Практикум. Красноярск, 2017.
7. Фадеев М.А., Марков К.А. Численные методы (учебное пособие) ННГУ, 2010. 158с.

ДОДАТОК 1

Лістинг програми

script_decision.py

```
from calculations import *
from math import sqrt
from matplotlib.pyplot import plot, show, xlabel, ylabel

input_file = "task.dat"
output_file = "execution_result.dat"

first_GF = True

ifile = open(input_file, 'r')

s = ifile.readline()
if(s[:3]=="max"):
    f_max = True
else:
    f_max = False

A = []
ifile.readline()
s = ifile.readline()
while s.find("C") == -1:
    try:
        A.append([float(x) for x in s.split()])
        s = ifile.readline()
    except ValueError:
        print ("A is unacceptable value")
        exit()

C = []
s = ifile.readline()
while s.find("T") == -1:
    try:
        C.append([float(x) for x in s.split()])
        s = ifile.readline()
```

```
    except ValueError:
        print ("C is unacceptable value")
        exit()

T = []
s = ifile.readline()
while s.find("alpha") == -1:
    try:
        t = s.split()
        T.append(float(t[0]))
        s = ifile.readline()
        if s.find("alpha") != -1: break
        s = ifile.readline()
    except ValueError:
        print ("T is unacceptable value")
        exit()

try:
    s = ifile.readline()
    t = s.split()
    alpha = float(t[0])
except ValueError:
    print ("Alpha is unacceptable value")
    exit()

ifile.close()



N = len(A)
M = len(A[0])
G = len(T) + 1 + N * M

_I = get_matrix(G, 1)
for i in range(len(T)):
    _I[i] = "<="

_I[len(T)] = "="
for i in range(N*M):
    _I[len(T)+1+i] = "<="
```

```
_B = get_matrix(G,1)
for i in range(len(T)):
    _B[i] = T[i]
_B[len(T)] = 0.0
for i in range(N*M):
    _B[len(T)+1+i] = A[i//M][i%M]


_Z = get_matrix(N*M,1)
if(first_GF):

    for i in range(N*M):
        if(i%M==M-1):
            _Z[i] = 1.0
        else:
            _Z[i] = 0.0
else:

    for i in range(N):
        _Z[i] = 1.0
        _Z[i+N] = 3.0
        _Z[i+2*N] = 0.0


is_optimist = True


def zero_func(x):
    if x<0:
        return 0.0
    else:
        return x

def _C(C_copy,index,alph,optimist,_first_GF):
    _M = len(C_copy[0])
    if _first_GF:

        if optimist:
            sign_ = 1.0
        else:
```

```
            sign_ = -1.0
        c = C[index//_M][index%_M]
        if(index%_M==_M-1):
            return zero_func(c*(1 - sign_ * sqrt((1-alph)/(2*alph))))
        else:
            return zero_func(c*(1 + sign_ * sqrt((1-alph)/(2*alph))))
    else:

        pass

_A = get_matrix(G, N*M)
for j in range(len(T)):
    for i in range(N):
        _A[j][i+N*j] = 1.0
for i in range(N*M):
    sign = 1.0
    if(i%M==M-1):
        sign = -1.0
    _A[len(T)+1+i][i] = 1.0




step = 0.01
ar_alpha = []
opt_f_alph = []
for k in range(1,100):
    alpha = step*k
    ar_alpha.append(alpha)

    print("Optimist:")
    for i in range(N*M):
        sign = 1.0
        if(i%M==M-1):
            sign = -1.0
        _A[len(T)][i] = sign * _C(C,i,0.8,is_optimist,first_GF)
    x = multy_simplex(_A, _I, _B, _Z, look_max=f_max)
    if x is None:
        opt_f_alph.append(0)
    else:
        x = sorted(x.items())
```

```
        print("Optimal plan:",x)
        optim_plan = x

        opt_f = 0.0
        for i in x:
            opt_f+=_Z[i[0]]*i[1]
        print("opt_f=",opt_f)
        opt_f_alph.append(opt_f)


step = 0.01
ar_alpha = []
pess_f_alph = []
for k in range(1,100):
    alpha = step*k
    ar_alpha.append(alpha)

    print("Pessimist: ", i)
    is_optimist = False
    for i in range(N*M):
        sign = 1.0
        if(i%M==M-1):
            sign = -1.0
        _A[len(T)][i] = sign * _C(C,i,0.8,is_optimist,first_GF)
    x = multy_simplex(_A, _I, _B, _Z, look_max=f_max)
    print("Pessimist:")
    if x is None:
        pess_f_alph.append(0)
    else:
        x = sorted(x.items())
        print("Optimal plan:",x)
        pess_plan = x

        pess_f = 0.0
        for i in x:
            pess_f+=_Z[i[0]]*i[1]
        print("pess_f=",pess_f)
        pess_f_alph.append(pess_f)

print("Goal function is between: "+str(pess_f)+" and "+str(opt_f))
```

```
print("X0   X1  X2  X3  X4  X5  X6  X7  X8 ")
print("X11 X12 X13 X21 X22 X23 X31 X32 X33")
print("Goal Functiom")
print(_Z)
print("Optimal plan for pessimist: ",pess_plan)
print("Optimal plan for optimist: ",optim_plan)


ofile = open(output_file, 'w')
ofile.write("Goal function is between: "+str(pess_f)+" and "+str(opt_f)+'\n')
ofile.write("Optimal plan for pessimist: "+str(pess_plan)+'\n')
ofile.write("Optimal plan for optimist: "+str(optim_plan)+'\n')
```

calculations.py

```
MAX_ITER = 100
INF = -1E+15
EPS = 1E-5

def multy_simplex(A, I, B, Z, look_max=True):
    n = len(A[0])
    m = len(I)

    if (not look_max):
        Z = [-x for x in Z]

    if m < n:
        return 'not realized yet'

    dx = m
    dy = 0
    for i in range(m):
        if (I[i] == ">="):
            dy += 1

    M = []
    for i in range(m):
        temp = [0.0 for i in range(n + dx + dy)]
```

```
    M.append(temp)
i = j = 0
for i in range(m):
    for j in range(n):
        M[i][j] = A[i][j]

for i in range(m):
    M[i][i + n] = 1.0

Z = Z + [0.0] * (dx + dy)

temp = i = 0
for i in range(m):
    if (I[i] == ">="):
        M[i][n + dx + temp] = -1.0
        temp += 1
        Z[i + n] = INF
    if (I[i] == "="):
        Z[i + n] = INF

print("Z", Z)



dopM = []
for i in range(m):
    temp = [0.0 for i in range(1 + dx)]
    dopM.append(temp)
for i in range(m):
    dopM[i][0] = B[i]
for i in range(m):
    dopM[i][i + 1] = 1.0

Bx = []
for i in range(m):
    Bx.append(n + i)

numb_iter = 0
x = dict()
while 1:
```

```
print("=========", numb_iter)


L = []
for i in range(m):
    sum = 0
    for j in range(m):
        sum += dopM[j][i + 1] * Z[Bx[j]]
    L.append(sum)


D = []
for i in range(n + dx + dy):
    sum = 0
    for j in range(m):
        sum += M[j][i] * L[j]
    D.append(sum - Z[i])


if (min(D) >= -EPS):

    for i in range(n):
        for j in range(m):
            if (Bx[j] == i):
                x[i] = dopM[j][0]
    return x
else:
    mind = min(D)
    for i in range(n + dx + dy):
        if abs(D[i] - mind) < 1e-5:
            in_i = i
            break

Ak = []
for i in range(m):
    Ak.append(M[i][in_i])


Ak_t = []
for i in range(m):
```

```
            sum = 0
            for j in range(m):
                sum += dopM[i][j + 1] * Ak[j]
            Ak_t.append(sum)
        Ak = Ak_t



        mind = 0
        out_i = 'nosolve'
        first = True
        for i in range(m):
            if (Ak[i] > 0):
                if (first):
                    mind = dopM[i][0] / Ak[i]
                    out_i = i
                    first = False
                if (mind > dopM[i][0] / Ak[i]):
                    mind = dopM[i][0] / Ak[i]
                    out_i = i

        if (out_i == 'nosolve'):
            return None

        Bx[out_i] = in_i
        znam = Ak[out_i]
        for i in range(m):
            if Ak[i] != 0.0 and i != out_i:
                for j in range(m + 1):
                    dopM[i][j] -= dopM[out_i][j] * Ak[i] * 1.0 / znam
        for i in range(m + 1):
            dopM[out_i][i] /= znam
        numb_iter += 1
        if (MAX_ITER < numb_iter):
            return None

def get_matrix(n, m):
    res = []
    for i in range(n):
        temp = []
        for j in range(m):
```

```python
            temp.append(0.0)
        res.append(temp)
    return res

def printm(label, A):
    print(label + ":")
    for i in A:
        print(i)
```

task.dat

```
A
6000 8000
9000 6000
8000 7000
C
50 40 20
40 30 25
38 50 30
T1
15000
T2
20000
alpha
0.8
```

execution_result.dat

```
Goal function is between: 16431 and 18910
Optimal plan for pessimist: [(0, 6000), (1, 4431), (2, 4568), (3, 6000),  (4, 8000), (5, 6000.0)]
Optimal plan for optimist: [(0, 6000), (1, 6909.0), (2, 2090), (3, 5000), (3, 8000.0), (5, 7000.0)
```