Sprawozdanie z projektu 3 MOM Dawid Bartosiak

Spis treści

1. Opis problemu

Modelowanie Matematyczne

Projekt 3 (Zestaw 1_JG)

Opis modelowanego problemu:

Przedsiębiorstwo produkuje trzy produkty $P1, P2, P3$ (sztuki). Każdy z tych produktów potrzebuje trzech różnych składników $S1, S2, S3$ (kg/jednostkę). Każdy z produktów ma inną ceną jednostkowę sprzedaży C_{P1}, C_{P2}, C_{P3} (tyś.PLN/jednostkę). Firma zwraca uwagę na ekologię i szacuje jednostkowy poziom zanieczyszczeń emitowanych dla poszczególnych produktów Z_{P1}, Z_{P2}, Z_{P3} (kg/jednostkę). Dostępne są również jednostkowe koszty produkcji K_{P1}, K_{P2}, K_{P3} (tyś.PLN/jednostkę).

Ograniczenia

1. Nie można użyć więcej niż 110 kg składnika $S1$, ale 100 kg jest akceptowalne.
2. Zaleca się użycie 50 kg składnika $S2$, ale zużycie powyżej 55 kg nie jest akceptowalne.
3. Nie jest akceptowalne zużycie składnika $S3$ powyżej 50 kg.
4. Zakłada się się, że produkcja produktu $P1$ powinna być nie mniejsza niż 3 sztuki, a produktu $P3$ nie mniejsza niż 5 sztuk.

Cele postawione przez zarządzających firmą:

1. Maksymalizacja zysków; dążenie do zysku na poziomie 150 tyś. PLN, ale akceptowalny jest zysk na poziomie 130 tyś PLN.
2. Minimalizacja emisji zanieczyszczeń; dążenie do emisji na poziomie 30 kg, ale poziom 35 kg jest akceptowalny.
3. Minimalizacja kosztów produkcji; dążenie do kosztów na poziomie 70 tyś. PLN, ale koszty na poziomie 80 tyś. są akceptowalne.

Rysunek 1. Treść zadania

	S1	S2	S3	C_x	Z_x	K_x
P1	2	8	4	10	1	1
P2	10	1	0	22	1	3
P3	4	4	2	12	3	3

Rysunek 2. Dane

2. Wstęp

Z treści zadania wynikają nam cele (u nas funkcje celu) które będziemy chcieli optymalizować, są to:

- Koszty produkcji poniesione przez firmę dane wzorem:

$$K = P1 \cdot 1 + P2 \cdot 3 + P3 \cdot 3$$

- Dzienny zysk firmy który może zostać zapisany jako:

$$C = P1 \cdot 10 + P2 \cdot 22 + P3 \cdot 12 - K*$$

- Poziom emisji zanieczyszczeń firmy w postaci matematycznej:

$$Z = P1 \cdot 1 + P2 \cdot 1 + P3 \cdot 3$$

*uwaga – rozpatruję jak w poleceniu zysk, nie przychód, stąd odejmuję koszt od przychodu.

Dodatkowo mamy powiązania liczby zużytych składników do produkcji poszczególnych produktów:

$$S1 = 2 \cdot P1 + 10 \cdot P2 + 4 \cdot P3$$

$$S2 = 8 \cdot P1 + 1 \cdot P2 + 4 \cdot P3$$

$$S3 = 4 \cdot P1 + 0 \cdot P2 + 2 \cdot P3$$

Występują także ograniczenia:

$$S1 \leq 110, \qquad S2 \leq 55, \qquad S3 \leq 50, \qquad P1 \geq 3, \qquad P2 \geq 0, \qquad P3 \geq 5$$

Dodatkowo każda z powyższych wartości jest nieujemna. Wartość S1 i S2 jest zalecana do użycia w wysokości odpowiednio 100 i 50 jednostek, zatem zostanie to przyjęte jako dodatkowe kryterium (jednocześnie do optymalizacji jak i ograniczenie).

3. Zadanie 1

Sformułować i opisać wielokryterialny model planowania produkcji z wykorzystaniem metody punktu odniesienia

Do założeń ze wstępu konieczne jest wprowadzenie konkretnej wartości funkcji celu. Z tego powodu należy wprowadzić punkty rezerwacji i aspiracji. Przyjmuję aktualnie wszystkie parametry β o stałej wartości 0,8, γ na poziomie 1,2 oraz ε wynoszący 0,001. Punkty rezerwacji i aspiracji:

$$a_k = 70, \qquad r_k = 80,$$

$$a_c = 150, \qquad r_c = 130,$$

$$a_z = 30, \qquad r_z = 35,$$

$$a_{s1} = 100, \qquad r_{s1} = 110,$$

$$a_{s2} = 50, \qquad r_{s2} = 55,$$

Zakładając klasyczne oznaczenia z metody punktu odniesienia przedstawione na wykładzie, tj.:

$$s_i(a_i, r_i, y_i) = \begin{cases} \beta(y_i - a_i)/(a_i - r_i) + 1, & \text{jeśli} \quad y_i \geqslant a_i \\ (y_i - r_i)/(a_i - r_i), & \text{jeśli} \quad a_i > y_i > r_i \\ \gamma(y_i - r_i)/(a_i - r_i), & \text{jeśli} \quad y_i \leqslant r_i \end{cases}$$

Rysunek 3. Przedstawiona na wykładzie funkcja osiągnięcia w metodzie punktu odniesienia

Celem otrzymania końcowego wzoru przechodzę na format odległości znormalizowanych, który został przedstawiony na wykładzie:

$$
\begin{array}{lllll}
\max & \underline{z} & + & \varepsilon \sum_{i=1}^{m} z_i & \\
\text{p.w.} & \underline{z} & \leqslant & z_i & \text{dla } i = 1, \ldots, m \\
 & z_i & \leqslant & \gamma \dfrac{y_i - r_i}{a_i - r_i} & \text{dla } i = 1, \ldots, m \\
 & z_i & \leqslant & \dfrac{y_i - r_i}{a_i - r_i} & \text{dla } i = 1, \ldots, m \\
 & z_i & \leqslant & \beta \dfrac{y_i - a_i}{a_i - r_i} + 1 & \text{dla } i = 1, \ldots, m
\end{array}
$$

Rysunek 4. Przedstawiona na wykładzie postać znormalizowana metody punktu odniesienia

Powyższa implementacja jest w pełni poprawna przyjmując jako poszczególne elementy (indeksy):

i = 1: zysk,

i = 2: emisja zanieczyszczeń,

i = 3: koszty produkcji,

i = 4: zużycie składnika S1.

i = 5: zużycie składnika S2.

W metodzie punktu odniesienia nie jest ważny kierunek w którym optymalizujemy, tzn. czy dokonujemy minimalizacji, czy maksymalizacji. Uwzględnione jest to z uwagi na punkty aspiracji i rezerwacji, które narzucają nam kierunek optymalizacji poprzez swój udział we wzorach jak zostało pokazane na Rysunku 4.

4. Zadanie 4

Zapisz zadanie/zadania sformułowane w punkcie 1 w postaci do rozwiązania z wykorzystaniem wybranego narzędzia implementacji (np. AMPL, AIMMS) i rozwiąż to zadanie/zadania. W przypadku niedopuszczalności zadania zaproponuj zmianę celów i/lub innych parametrów

Przedstawiony w ramach rozdziału 3 (zadanie 1) model zdecydowałem się rozwiązać z wykorzystaniem narzędzia AMPL. Model przeze mnie otrzymany dał się rozwiązać (był dopuszczalny) więc dalsze kroki nie były konieczne.

```
# Zbiór funkcji celu
set CELE := {1, 2, 3, 4, 5};  # 1-zysk, 2-emisja, 3-koszty, 4-zużycie S1, 5-zużycie S2
# Zmienne decyzyjne
var x1 >= 3 integer; # Liczba sztuk produktu P1
var x2 >= 0 integer; # Liczba sztuk produktu P2
var x3 >= 5 integer; # Liczba sztuk produktu P3
# Funkcje celu
var f {i in CELE} =
  if i = 1 then 9*x1 + 19*x2 + 9*x3
  else if i = 2 then 1*x1 + 1*x2 + 3*x3
  else if i = 3 then 1*x1 + 3*x2 + 3*x3
  else if i = 4 then 2*x1 + 10*x2 + 4*x3
  else 8*x1 + 1*x2 + 4*x3;  # zużycie składnika S2
# Punkty odniesienia
param r {CELE};  # punkty rezerwacji
param a {CELE};  # punkty aspiracji
# Parametry kontrolujące kształt funkcji osiągnięcia - indywidualne dla każdej funkcji celu
param gamma {CELE} default 1.2; # parametr gamma
param beta {CELE} default 0.8;  # parametr beta
# Inicjalizacja punktów odniesienia
data;
param r :=
  1 130  # punkt rezerwacji dla zysku
  2 35   # punkt rezerwacji dla emisji
  3 80   # punkt rezerwacji dla kosztów
  4 110 # punkt rezerwacji dla zużycia S1
  5 55;  # punkt rezerwacji dla zużycia S2
param a :=
  1 150  # punkt aspiracji dla zysku
  2 30   # punkt aspiracji dla emisji
  3 70   # punkt aspiracji dla kosztów
  4 100  # punkt aspiracji dla zużycia S1
  5  50; # punkt aspiracji dla zużycia S2
;
model;
# Zmienne pomocnicze dla funkcji osiągnięcia
var z {i in CELE}; # znormalizowane odległości dla każdej funkcji celu
var min_z;         # minimalna znormalizowana odległość
# Ograniczenia
subject to ogr_S1: 2*x1 + 10*x2 + 4*x3 <= 110;
subject to ogr_S2: 8*x1 + 1*x2 + 4*x3 <= 55;
subject to ogr_S3: 4*x1 + 0*x2 + 2*x3 <= 50;
# Definicje znormalizowanych odległości
subject to def_z_gamma {i in CELE}:
  z[i] <= gamma[i] * (f[i] - r[i])/(a[i] - r[i]);
subject to def_z_normal {i in CELE}:
  z[i] <= (f[i] - r[i])/(a[i] - r[i]);
subject to def_z_beta {i in CELE}:
  z[i] <= beta[i] * (f[i] - a[i])/(a[i] - r[i]) + 1;
# Definicja minimalnej odległości
subject to min_z_def {i in CELE}:
  min_z <= z[i];
# Funkcja celu - maksymalizacja minimalnej odległości + suma ważona
param epsilon := 0.001;
maximize achievement: min_z + epsilon * sum {i in CELE} z[i];
```

Rysunek 5. Implementacja w AMPL zadania 1

```
reset;
model model_punkt_odniesienia.mod;
# Rozwiązanie modelu
solve;

# Wyświetlenie wyników
printf "\n===== WYNIKI OPTYMALIZACJI =====\n\n";

# Zmienne decyzyjne
printf "Zmienne decyzyjne:\n";
printf "x1 = %.0f (liczba sztuk produktu P1)\n", x1;
printf "x2 = %.0f (liczba sztuk produktu P2)\n", x2;
printf "x3 = %.0f (liczba sztuk produktu P3)\n\n", x3;

# Wartości funkcji celu
printf "Wartości funkcji celu:\n";
printf "Zysk = %.0f\n", f[1];
printf "Emisja = %.0f\n", f[2];
printf "Koszty = %.0f\n\n", f[3];

# Znormalizowane odległości
printf "Znormalizowane odległości:\n";
printf "z[1] (zysk) = %.2f\n", z[1];
printf "z[2] (emisja) = %.2f\n", z[2];
printf "z[3] (koszty) = %.2f\n", z[3];
printf "z[4] (zużycie S1) = %.2f\n", z[4];
printf "z[5] (zużycie S2) = %.2f\n", z[5];
printf "min_z = %.2f\n\n", min_z;

# Wykorzystanie zasobów
printf "Wykorzystanie zasobów:\n";
printf "Składnik S1: %.0f / 110 (%.1f%%)\n", 2*x1 + 10*x2 + 4*x3, 100*(2*x1 + 10*x2 + 4*x3)/110;
printf "Składnik S2: %.0f / 55 (%.1f%%)\n", 8*x1 + 1*x2 + 4*x3, 100*(8*x1 + 1*x2 + 4*x3)/55;
printf "Składnik S3: %.0f / 50 (%.1f%%)\n\n", 4*x1 + 0*x2 + 2*x3, 100*(4*x1 + 0*x2 + 2*x3)/50;

# Punkty odniesienia i osiągnięte wartości
printf "Porównanie z punktami odniesienia:\n";
printf "%-10s %-10s %-10s %-10s\n", "Funkcja", "Wartość", "Rezerwacja", "Aspiracja";
printf "%-10s %-10d %-10d %-10d\n", "Zysk", f[1], r[1], a[1];
printf "%-10s %-10d %-10d %-10d\n", "Emisja", f[2], r[2], a[2];
printf "%-10s %-10d %-10d %-10d\n\n", "Koszty", f[3], r[3], a[3];

# Wartość funkcji osiągnięcia
printf "Wartość funkcji osiągnięcia: %.5f\n", min_z + epsilon * sum {i in CELE} z[i];
printf "===================================\n";
```

Rysunek 6. Plik uruchomieniowy do programu AMPL z zadania 1

Udało mi się otrzymać także wynik, który spełnia wszystkie ograniczenia ($x1 \geq 3$, $x3 \geq 5$), oraz jest dopuszczalny z uwagi na wartości zmiennych S1(76/110), S2(49/55) i S3(22/50). Więcej będę mógł o nim powiedzieć, gdy będę miał porównanie z następnymi zadaniami.

```
ampl: include 'E:\AMPL\MOM\P3\solve.run';
CPLEX 22.1.2:              CPLEX 22.1.2: optimal solution; objective 1.17184
9 simplex iterations

===== WYNIKI OPTYMALIZACJI =====

Zmienne decyzyjne:
x1 = 3 (liczba sztuk produktu P1)
x2 = 5 (liczba sztuk produktu P2)
x3 = 5 (liczba sztuk produktu P3)

Wartości funkcji celu:
Zysk = 167
Emisja = 23
Koszty = 33

Znormalizowane odległości:
z[1] (zysk) = 1.68
z[2] (emisja) = 2.12
z[3] (koszty) = 3.96
z[4] (zużycie S1) = 2.92
z[5] (zużycie S2) = 1.16
min_z = 1.16

Wykorzystanie zasobów:
Składnik S1: 76 / 110 (69.1%)
Składnik S2: 49 / 55 (89.1%)
Składnik S3: 22 / 50 (44.0%)

Porównanie z punktami odniesienia:
Funkcja     Wartość     Rezerwacja Aspiracja
Zysk        166         130        150
Emisja      22          35         30
Koszty      32          80         70

Wartość funkcji osiągnięcia: 1.17184
===================================
```

Rysunek 7. Otrzymany wynik w zadaniu 4.

5. Zadanie 2

Sformułować i opisać wielokryterialny model optymalnego planowania produkcji z wykorzystaniem zbiorów rozmytych

Do założeń ze wstępu dodać należy ponownie nieostro określone aspiracje zarządu przedsiębiorstwa rozpisując funkcje przynależności. Dla każdego kryterium (zysk, emisja zanieczyszczeń, koszty) określono funkcje opisujące stopień ich spełnienia względem wartości aspiracyjnych i dopuszczalnych:

$$\mu_1\ (Zysk) = \begin{cases} 0, dla\ C < 130 \\ \dfrac{C-130}{20}, dla\ 130 \le C < 150 \\ 1, dla\ C \ge 150 \end{cases}$$

$$\mu_2(Emisja) = \begin{cases} 1, dla\ Z < 30 \\ \dfrac{Z-30}{5}, dla\ 30 \le Z < 35 \\ 0, dla\ Z \ge 35 \end{cases}$$

$$\mu_3(Koszt) = \begin{cases} 1, dla\ K < 70 \\ \dfrac{K-70}{10}, dla\ 70 \le K < 80 \\ 0, dla\ K \ge 80 \end{cases}$$

$$\mu_4(Zużycie\ surowca\ S1) = \begin{cases} 1, dla\ S1 < 100 \\ \dfrac{S1-100}{10}, dla\ 100 \le S1 < 110 \\ 0, dla\ S1 \ge 110 \end{cases}$$

$$\mu_5(Zużycie\ surowca\ S2) = \begin{cases} 1, dla\ S2 < 50 \\ \dfrac{S2-50}{5}, dla\ 50 \le S2 < 55 \\ 0, dla\ S2 \ge 55 \end{cases}$$

Ostatecznie funkcja celu to:

$$\max\ (\mu_1 + \mu_2 + \mu_3 + \mu_4 + \mu_5)$$

co pozwala na kompromisowe uwzględnienie wszystkich celów.

6. Zadanie 3

Sformułować równoważne zadanie optymalizacji dla zadania 2 z wykorzystaniem zbiorów rozmytych adaptując podejście Zimmermana dla więcej niż jednego kryterium.

Do sformułowanego w rozdziale 5 (zadaniu 2) modelu należy zmienić funkcję celu zachowując pozostałe czynniki niezmienione. W tym celu stosowany jest operator max-min i chcemy z jego użyciem maksymalizować wszystkie cele jednocześnie, tzn. maksymalizujemy minimalną wartość funkcji przynależności:

$$\max(\alpha)$$

gdzie parametr α opisany jest przez następujące nierówności:

$$\alpha \leq \mu_1$$

$$\alpha \leq \mu_2$$

$$\alpha \leq \mu_3$$

$$\alpha \leq \mu_4$$

$$\alpha \leq \mu_5$$

Oraz jest określony na dziedzinie:

$$\alpha \in [0, 1]$$

7. Zadanie 5

Zapisz zadania sformułowane w punkcie 3 w postaci do rozwiązania z wykorzystaniem wybranego narzędzia implementacji (np. AMPL, AIMMS) i rozwiąż te zadania. W przypadku niedopuszczalności zadania zaproponuj zmianę celów i/lub innych parametrów.

Udało się zaimplementować i otrzymać dopuszczalne wyniki z działania programu AMPL, co jest przedstawione na kolejnych rysunkach.

```
# Zmienne decyzyjne
var x1 >= 3 integer; # Liczba sztuk produktu P1
var x2 >= 0 integer; # Liczba sztuk produktu P2
var x3 >= 5 integer; # Liczba sztuk produktu P3
# Funkcje celu
var zysk = 9*x1 + 19*x2 + 9*x3;       # Zysk (10-1)*x1 + (22-3)*x2 + (12-3)*x3
var emisja = 1*x1 + 1*x2 + 3*x3;       # Emisja zanieczyszczeń
var koszty = 1*x1 + 3*x2 + 3*x3;       # Koszty produkcji
var zuzycie_S1 = 2*x1 + 10*x2 + 4*x3;  # Zużycie surowca S1
var zuzycie_S2 = 8*x1 + 1*x2 + 4*x3;   # Zużycie surowca S2
# Ograniczenia zasobów
subject to ogr_S1: zuzycie_S1 <= 110;
subject to ogr_S2: zuzycie_S2 <= 55;
subject to ogr_S3: 4*x1 + 0*x2 + 2*x3 <= 50;
# Zmienna decyzyjna alfa - poziom satysfakcji
var alfa >= 0, <= 1;
# Funkcje przynależności dla poszczególnych celów
# μ_1 (Zysk) = (zysk - 130)/20 dla 130 ≤ zysk < 150, 0 dla zysk < 130, 1 dla zysk ≥ 150
subject to funkcja_przynaleznosci_zysk:
  alfa <= (zysk - 130)/20;
# μ_2 (Emisja) = (35 - emisja)/5 dla 30 ≤ emisja < 35, 1 dla emisja < 30, 0 dla emisja ≥ 35
subject to funkcja_przynaleznosci_emisja:
  alfa <= (35 - emisja)/5;
# μ_3 (Koszt) = (80 - koszty)/10 dla 70 ≤ koszty < 80, 1 dla koszty < 70, 0 dla koszty ≥ 80
subject to funkcja_przynaleznosci_koszty:
  alfa <= (80 - koszty)/10;
# μ_4 (Zużycie S1) = (110 - zuzycie_S1)/10 dla 100 ≤ zuzycie_S1 < 110, 1 dla zuzycie_S1 < 100,
# 0 dla zuzycie_S1 ≥ 110
subject to funkcja_przynaleznosci_S1:
  alfa <= (110 - zuzycie_S1)/10;
# μ_5 (Zużycie S2) = (55 - zuzycie_S2)/5 dla 50 ≤ zuzycie_S2 < 55, 1 dla zuzycie_S2 < 50, 0 dla zuzycie_S2 ≥ 55
subject to funkcja_przynaleznosci_S2:
  alfa <= (55 - zuzycie_S2)/5;
# Funkcja celu - maksymalizacja poziomu satysfakcji alfa
maximize poziom_satysfakcji: alfa;
```

Rysunek 8. Kod AMPL realizujący podejście Zimmermana

```
# Wczytanie modelu
reset;
model model_zimmerman.mod;
# Rozwiązanie modelu
option solver cplex;
solve;
# Wyświetlenie wyników
display x1, x2, x3;
display zysk, emisja, koszty, zuzycie_S1, zuzycie_S2;
display alfa;
# Obliczenie wartości funkcji przynależności dla poszczególnych celów
printf "Wartość funkcji przynależności dla zysku: %f\n",
  if zysk < 130 then 0
  else if zysk < 150 then (zysk - 130)/20
  else 1;
printf "Wartość funkcji przynależności dla emisji: %f\n",
  if emisja < 30 then 1
  else if emisja < 35 then (35 - emisja)/5
  else 0;
printf "Wartość funkcji przynależności dla kosztów: %f\n",
  if koszty < 70 then 1
  else if koszty < 80 then (80 - koszty)/10
  else 0;
printf "Wartość funkcji przynależności dla zużycia S1: %f\n",
  if zuzycie_S1 < 100 then 1
  else if zuzycie_S1 < 110 then (110 - zuzycie_S1)/10
  else 0;
printf "Wartość funkcji przynależności dla zużycia S2: %f\n",
  if zuzycie_S2 < 50 then 1
  else if zuzycie_S2 < 55 then (55 - zuzycie_S2)/5
  else 0;
```

Rysunek 9. Kod uruchomieniowy do zadania 5

```
ampl: include 'E:\AMPL\MOM\P3\model_zimmerman.run';
CPLEX 22.1.2:              CPLEX 22.1.2: optimal solution; objective 1
1 simplex iterations
x1 = 3
x2 = 5
x3 = 5

zysk = 167
emisja = 23
koszty = 33
zuzycie_S1 = 76
zuzycie_S2 = 49

alfa = 1

Wartość funkcji przynależności dla zysku: 1.000000
Wartość funkcji przynależności dla emisji: 1.000000
Wartość funkcji przynależności dla kosztów: 1.000000
Wartość funkcji przynależności dla zużycia S1: 1.000000
Wartość funkcji przynależności dla zużycia S2: 1.000000
```

Rysunek 10. Wynik działania programu z zadania 5

8. Zadanie 6

Porównaj rozwiązania zadań z poprzednich dwóch punktów

Otrzymane wyniki w zadaniu 4 (Rozdział 4) i zadaniu 5 (Rozdział 7) są dokładnie takie same. Wynika to z brania pod uwagę przeze mnie funkcji które nie są odpowiednio dopasowane do danych, to znaczy da się w przypadku metody punktu odniesienia znacznie lepiej dobrać parametry. Jednym z takich przykładów jest odpowiednie ustawienie parametru β dla elementu 4 i 5 na bardzo niską wartość, lub porzucenie ich optymalizacji przyjmując jedynie twarde ograniczenie. W podejściu Zimmermanna nie mamy z tego żadnych dodatkowych korzyści, gdyż otrzymujemy wtedy w zasadzie losowe pasujące rozwiązanie (spełniające $\alpha = 1$), przez co trudno jest przewidzieć jak algorytm się zachowa. Zmiana podejścia w metodzie punktu odniesienia daje zupełnie inny wynik co zostało pokazane na rysunku 11 i rysunku 12. Rozwiązania są nieporównywalne między sobą (nie zachodzi relacja dominacji) jednak metoda punktu odniesienia daje znacznie większe możliwości.

```
CPLEX 22.1.2:               CPLEX 22.1.2: optimal solution; objective 1.01012014
10 simplex iterations

===== WYNIKI OPTYMALIZACJI =====

Zmienne decyzyjne:
x1 = 3 (liczba sztuk produktu P1)
x2 = 6 (liczba sztuk produktu P2)
x3 = 5 (liczba sztuk produktu P3)

Wartości funkcji celu:
Zysk = 186
Emisja = 24
Koszty = 36

Znormalizowane odległości:
z[1] (zysk) = 2.44
z[2] (emisja) = 1.96
z[3] (koszty) = 3.72
z[4] (zużycie S1) = 1.00
z[5] (zużycie S2) = 1.00
min_z = 1.00

Wykorzystanie zasobów:
Składnik S1: 86 / 110 (78.2%)
Składnik S2: 50 / 55 (90.9%)
Składnik S3: 22 / 50 (44.0%)

Porównanie z punktami odniesienia:
Funkcja    Wartość    Rezerwacja Aspiracja
Zysk       185        130        150
Emisja     23         35         30
Koszty     35         80         70

Wartość funkcji osiągnięcia: 1.01012
===================================
```

Rysunek 11. Zastosowanie mniejszego parametru w metodzie punktu odniesienia dla S1 i S2

```
ampl: include 'E:\AMPL\MOM\P3\solve.run';
CPLEX 22.1.2:               CPLEX 22.1.2: optimal solution; objective 0.00884
7 simplex iterations

===== WYNIKI OPTYMALIZACJI =====

Zmienne decyzyjne:
x1 = 3 (liczba sztuk produktu P1)
x2 = 8 (liczba sztuk produktu P2)
x3 = 5 (liczba sztuk produktu P3)

Wartości funkcji celu:
Zysk = 224
Emisja = 26
Koszty = 42

Znormalizowane odległości:
z[1] (zysk) = 3.96
z[2] (emisja) = 1.64
z[3] (koszty) = 3.24
z[4] (zużycie S1) = 0.00
z[5] (zużycie S2) = 0.00
min_z = 0.00

Wykorzystanie zasobów:
Składnik S1: 106 / 110 (96.4%)
Składnik S2: 52 / 55 (94.5%)
Składnik S3: 22 / 50 (44.0%)

Porównanie z punktami odniesienia:
Funkcja    Wartość    Rezerwacja Aspiracja
Zysk       224        130        150
Emisja     26         35         30
Koszty     42         80         70

Wartość funkcji osiągnięcia: 0.00884
===================================
```

Rysunek 12. Zastosowanie S1 i S2 jedynie jako ograniczeń w metodzie punktu odniesienia

9. Zadanie 7

Rozwiąż zadanie z punktu 2 za pomocą pakietu R – FuzzyLP. Należy w obliczeniach rozpatrywać niezależnie każde z kryteriów.

zadanie5_fuzzyLP.R

```r
library(FuzzyLP)
# Ograniczenia
constraints <- matrix(c(
  2, 10, 4,
  8, 1, 4,
  4, 0, 2,
  1, 0, 0,
  0, 0, 1,
  9, 19, 9,
  1, 1, 3,
  1, 3, 3
), nrow = 8, byrow = TRUE)
directions <- c("<=", "<=", "<=", ">=", ">=", ">=", "<=", "<=")
aspiration <- c(100, 50, 50, 3, 5, 150, 30, 70)
tolerance  <- c(10, 5, 0, 0, 0, 20, 5, 10)
# Funkcja celu
objectives <- list(
  income    = list(vec = c(9, 19, 9), target = 150, tol = 20, max = TRUE),
  emissions = list(vec = c(1, 1, 3), target = 30,  tol = 5,  max = FALSE),
  cost      = list(vec = c(1, 3, 3), target = 70,  tol = 10, max = FALSE),
  s1        = list(vec = c(2, 10, 4),  target = 100, tol = 10, max = FALSE),
  s2        = list(vec = c(8, 1, 4),   target = 50,  tol = 5,  max = FALSE)
)
# Funkcja wykonująca obliczenia
solve_and_show <- function(name, obj) {
  cat("\n===== CEL:", toupper(name), "=====\n")
  res <- FCLP.fuzzyObjective(
    obj$vec, constraints, directions, aspiration, tolerance,
    z0 = obj$target, t0 = obj$tol, maximum = obj$max, verbose = FALSE
  )
  x <- res[, c("x1", "x2", "x3")]
  cat("Rozwiązanie: x =", round(x, 3), "\n")
  cat("Income    =", sum(c(9,19,9) * x), "\n")
  cat("Emissions=", sum(c(1,1,3) * x), "\n")
  cat("Cost      =", sum(c(1,3,3) * x), "\n")
}
# Obliczenia dla każdego celu
for (name in names(objectives)) {
  solve_and_show(name, objectives[[name]])
}
```

Rysunek 13. Kod w R wykonujący optymalizację dla każdego kryterium z osobna

```
===== CEL: INCOME =====
Rozwiązanie: x = 3 4.105 5
Income    = 150
Emissions= 22.10526
Cost      = 30.31579

===== CEL: EMISSIONS =====
Rozwiązanie: x = 3 4.105 5
Income    = 150
Emissions= 22.10526
Cost      = 30.31579

===== CEL: COST =====
Rozwiązanie: x = 3 4.105 5
Income    = 150
Emissions= 22.10526
Cost      = 30.31579

===== CEL: S1 =====
Rozwiązanie: x = 3 4.105 5
Income    = 150
Emissions= 22.10526
Cost      = 30.31579

===== CEL: S2 =====
Rozwiązanie: x = 3 4.105 5
Income    = 150
Emissions= 22.10526
Cost      = 30.31579
```

Rysunek 14. Otrzymany wynik dla każdej z optymalizacji

Z uwagi na charakterystykę danych każdy z otrzymywanych wyników był taki sam za każdym odpaleniem programu. Po przybliżeniu produktów do liczb całkowitych, otrzymywany wynik pokrywał się z tym z pozostałych metod.

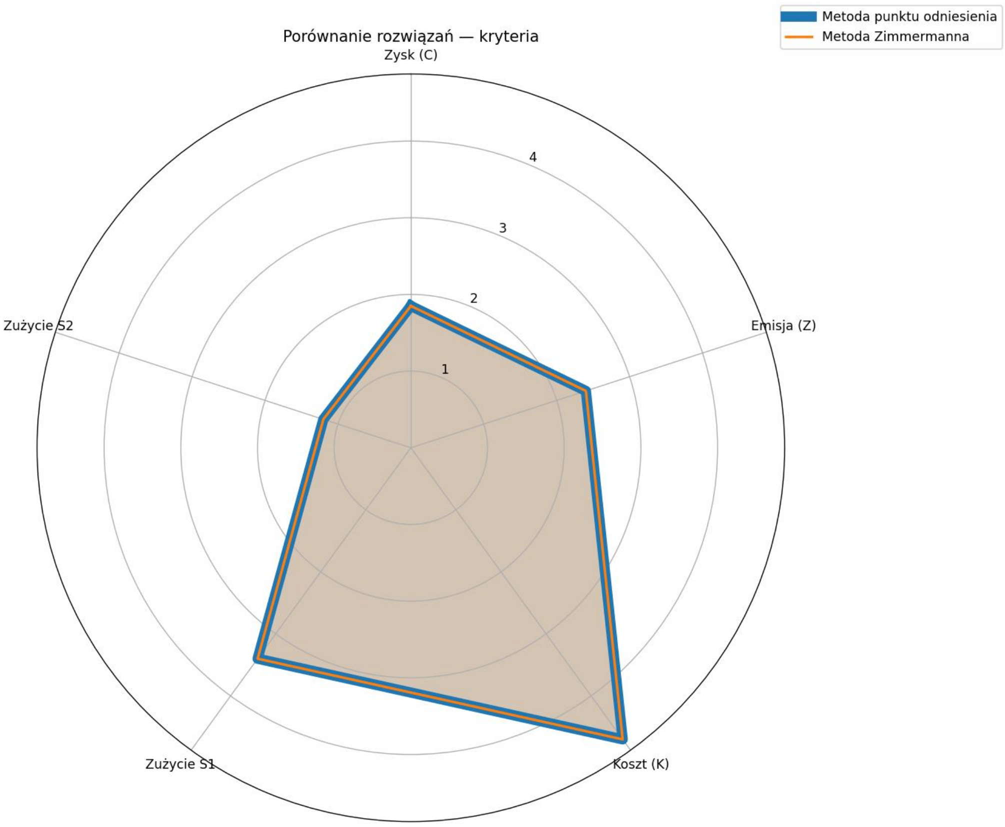
10. Zadanie 8

Zaproponuj i zastosuj graficzną formę analizy rozwiązań.

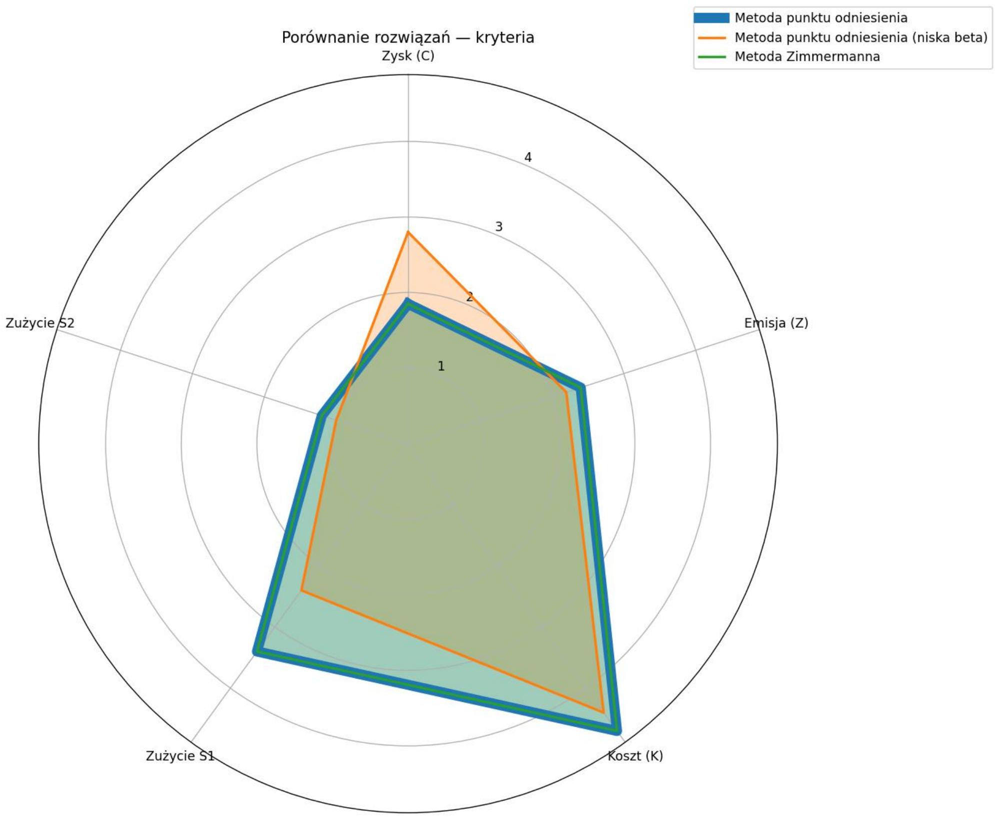
W mojej opinii dość dobrze opisującą formą analizy otrzymanych rozwiązań jest wykres radarowy. Pozwoli on na pierwszy rzut oka określić relację dominacji. Użyć do tego należy znormalizowanych wartości zmiennych, tak jak jest to przyjęte w metodzie punktu odniesienia. Kod wykonujący taki wykres:

```
import numpy as np
import matplotlib.pyplot as plt

# Kryteria
criteria = ['Zysk (C)', 'Emisja (Z)', 'Koszt (K)', 'Zużycie S1', 'Zużycie S2']

method1_values = [167, 23, 33, 76, 49]  # Zadanie 4
method2_values = [167, 23, 33, 76, 49]  # Zadanie 5

# Poziomy aspiracji i rezerwacji zgodnie z danymi:
aspiration = {'Zysk (C)': 150, 'Emisja (Z)': 30, 'Koszt (K)': 70, 'Zużycie S1': 100, 'Zużycie S2': 50}
reservation = {'Zysk (C)': 130, 'Emisja (Z)': 35, 'Koszt (K)': 80, 'Zużycie S1': 110, 'Zużycie S2': 55}


def normalize(value, crit):
    a = aspiration[crit]
    r = reservation[crit]
    return max(0, (value - r) / (a - r))


norm1 = [normalize(val, crit) for val, crit in zip(method1_values, criteria)]
norm2 = [normalize(val, crit) for val, crit in zip(method2_values, criteria)]

angles = np.linspace(0, 2 * np.pi, len(criteria), endpoint=False).tolist()
angles += angles[:1]

fig, ax = plt.subplots(figsize=(6, 6), subplot_kw=dict(polar=True))
ax.set_theta_offset(np.pi / 2)
ax.set_theta_direction(-1)
plt.xticks(angles[:-1], criteria)

# Rysowanie krzywych
norm1 += norm1[:1]
norm2 += norm2[:1]
ax.plot(angles, norm1, linewidth=2, label='Metoda punktu odniesienia')
ax.fill(angles, norm1, alpha=0.25)

ax.plot(angles, norm2, linewidth=2, label="Metoda Zimmermanna")
ax.fill(angles, norm2, alpha=0.25)

plt.legend(loc='upper right', bbox_to_anchor=(1.3, 1.1))
plt.title('Porównanie rozwiązań – kryteria')
plt.show()
```

Rysunek 15. Kod do analizy otrzymanych rozwiązań.

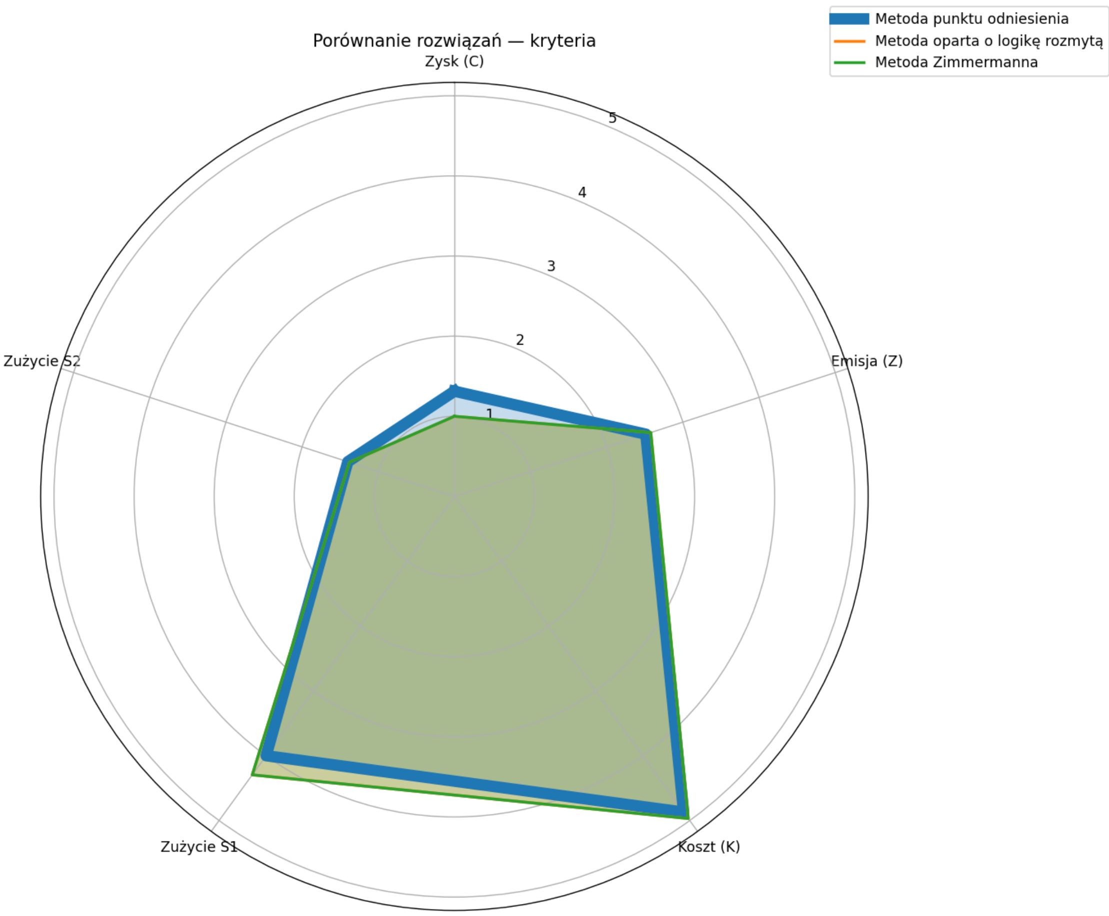


Rysunek 16. Otrzymane wyniki - podejście klasyczne



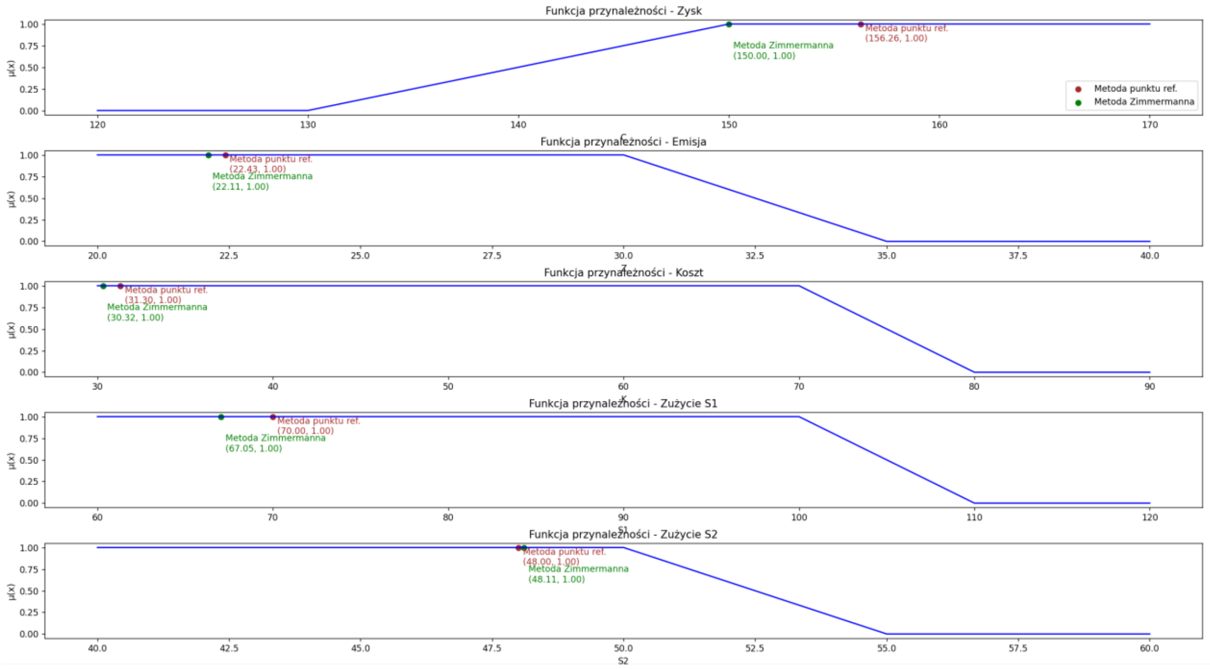
Rysunek 17. Otrzymane wyniki na tle metody punktu odniesienia z niską betą dla parametrów spoza celu zarządu

Różnica widoczna jest jednak także przechodząc z dziedziny liczb całkowitych na liczby rzeczywiste:



Rysunek 18. Otrzymane wyniki przyjmując liczby rzeczywiste.

Biorąc pod uwagę powyższy wykres stwierdzić można że nie zostały wygenerowane żadne rozwiązania zdominowane, jednak z uwagi na uwzględnianie kryteriów w różny sposób – zostały wyznaczone różne rozwiązania, które z punktu widzenia przyjętych kryteriów są tak samo dobre (pareto-optymalne). Dodatkowo jednak możemy postarać się przedstawić nasze rozwiązania opierając się o funkcje przynależności i na nich także przeprowadzać analizy. Przykładowa analiza zawierać może poszczególne funkcje przynależności:

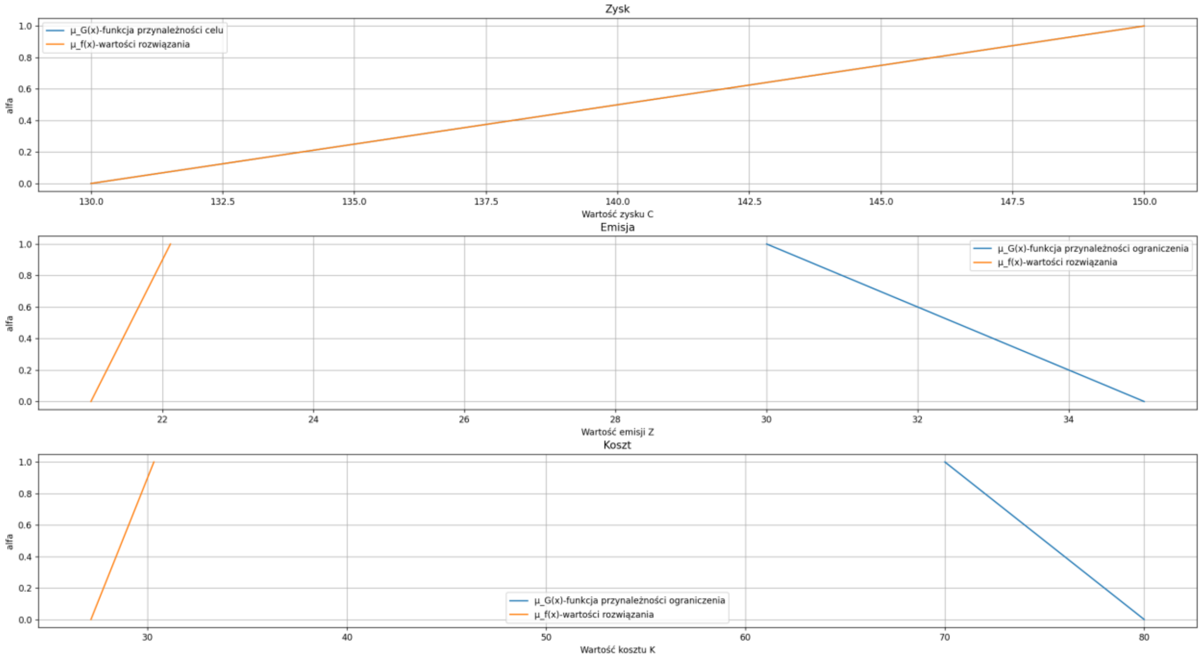


Rysunek 19. Analiza graficzna w oparciu o poszczególne funkcje przynależności

```python
import matplotlib.pyplot as plt
zysk = [(120, 0), (130, 0), (150, 1), (170, 1)]
emisja = [(20, 1), (30, 1), (35, 0), (40, 0)]
koszt = [(30, 1), (70, 1), (80, 0), (90, 0)]
s1 = [(60, 1), (100, 1), (110, 0), (120, 0)]
s2 = [(40, 1), (50, 1), (55, 0), (60, 0)]
case1 = {
    "Zysk": (156.26, 1.0),
    "Emisja": (22.43, 1.0),
    "Koszt": (31.30, 1.0),
    "S1": (70,     1.0),
    "S2": (48,     1.0)
}
case2 = {
    "Zysk": (150, 1.0),
    "Emisja": (22.10526, 1.0),
    "Koszt": (30.31579, 1.0),
    "S1": (67.0526,    1.0),
    "S2": (48.1053,    1.0)
}
Tabnine | Edit | Test | Fix | Explain | Document
def plot_fn(ax, data, point, label, xlabel, color='blue'):
    xs, ys = zip(*data)
    ax.plot(xs, ys, color=color)
    ax.set_ylim(-0.05, 1.05)
    ax.set_title(label)
    ax.set_xlabel(xlabel)
    ax.set_ylabel("µ(x)")
fig, axs = plt.subplots(5, 1, figsize=(8, 15))
plot_fn(axs[0], zysk, None, "Funkcja przynależności - Zysk", "C")
plot_fn(axs[1], emisja, None, "Funkcja przynależności - Emisja", "Z")
plot_fn(axs[2], koszt, None, "Funkcja przynależności - Koszt", "K")
plot_fn(axs[3], s1, None, "Funkcja przynależności - Zużycie S1", "S1")
plot_fn(axs[4], s2, None, "Funkcja przynależności - Zużycie S2", "S2")

colors = {'case1': 'brown', 'case2': 'green'}
labels = {'case1': 'Metoda punktu ref.', 'case2': 'Metoda Zimmermanna'}
i = 2
for case, props in zip([case1, case2], ['case1', 'case2']):
    for ax, crit, data_fn in zip(axs, ["Zysk", "Emisja", "Koszt", "S1", "S2"], [zysk, emisja, koszt, s1, s2]):
        x, mu = case[crit]
        ax.scatter(x, mu, color=colors[props], label=labels[props])
        ax.annotate(f"{labels[props]}\n({x:.2f}, {mu:.2f})", (x, mu),
                    textcoords="offset points", xytext=(5, -10*i), ha='left', color=colors[props])
    i *= 2
axs[0].legend(loc='lower right')
plt.tight_layout()
plt.show()
```

Rysunek 20. Kod do analizy z użyciem funkcji przynależności

Lub można analizować poprzez sprawdzanie wartości dla różnych poziomów alfy:

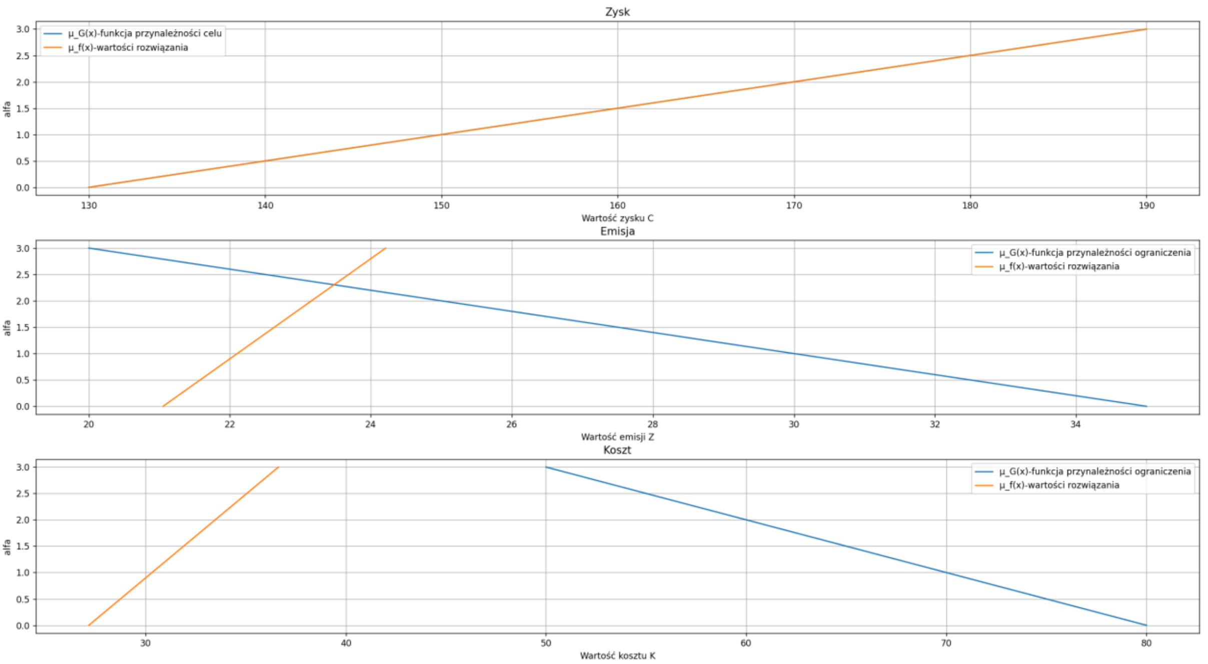


Rysunek 21. Analiza z użyciem poziomów alfy w metodzie Zimmermanna

```python
import numpy as np
import matplotlib.pyplot as plt
alphas = np.append(np.arange(0, 1.1, 0.1), [2.0, 3.0])
income_ys = []
emissions_ys = []
cost_ys = []
for alpha in alphas:
    f1 = 150 - 20 * (1-alpha)
    f2 = 30 + 5 * (1-alpha)
    f3 = 70 + 10 * (1-alpha)
    income_ys.append(f1)
    emissions_ys.append(f2)
    cost_ys.append(f3)
income_xs = list(reversed([190, 170, 150, 148, 146, 144, 142, 140, 138, 136, 134, 132, 130]))
emissions_xs = list(reversed([24.2105, 23.1579, 22.1053, 22, 21.8947, 21.7895, 21.6842, 21.5789, 21.4737, 21.3684, 21.2632, 21.1579, 21.0526]))
cost_xs = list(reversed([36.6316, 33.4737, 30.3158, 30, 29.6842, 29.3684, 29.0526, 28.7368, 28.4211, 28.1053, 27.7895, 27.4737, 27.1579]))
plt.figure(figsize=(8, 12))
plt.subplot(3, 1, 1)
plt.title("Zysk")
plt.plot(income_ys, alphas, label="μ_G(x)-funkcja przynależności celu")
plt.plot(income_xs, alphas, label="μ_f(x)-wartości rozwiązania")
plt.xlabel('Wartość zysku C')
plt.ylabel('alfa')
plt.grid(True)
plt.legend()
plt.subplot(3, 1, 2)
plt.title("Emisja")
plt.plot(emissions_ys, alphas, label="μ_G(x)-funkcja przynależności ograniczenia")
plt.plot(emissions_xs, alphas, label="μ_f(x)-wartości rozwiązania")
plt.xlabel('Wartość emisji Z')
plt.ylabel('alfa')
plt.grid(True)
plt.legend()
plt.subplot(3, 1, 3)
plt.title("Koszt")
plt.plot(cost_ys, alphas, label="μ_G(x)-funkcja przynależności ograniczenia")
plt.plot(cost_xs, alphas, label="μ_f(x)-wartości rozwiązania")
plt.xlabel('Wartość kosztu K')
plt.ylabel('alfa')
plt.grid(True)
plt.legend()
plt.tight_layout()
plt.show()
```

Rysunek 22. Kod do analizy z użyciem poziomu alfy w metodzie Zimmermanna

Zalety takiego rozwiązania to możliwość analizowania który parametr ma wpływ przy zmianie alfa. Tutaj widać przede wszystkim pokrywanie alfa z zyskiem, co wskazuje nam, że celem zwiększania poziomu przynależności będziemy zwiększać przede wszystkim poziom zysku. Ostatecznie jest to rozwiązanie które pozwala na lepszą analizę działania algorytmu, podczas gdy wykres radarowy pozwala na porównywanie różnych algorytmów i rozwiązań oraz relacji ich wzajemnej dominacji. Rozpatrując przypadek dowolnie wysokiej alfy dojdziemy do innych ograniczeń:



Rysunek 23. Graficzna analiza poziomu alfa z rozpatrzeniem alfy większej niż 1

W przedstawionym tutaj przypadku pojawia się nam przecięcie dla wartości emisji 23,478 oraz wartości alfy 2,304. Celowo w analizie pominąłem tutaj ograniczenia, gdyż przy aktualnych nie jesteśmy w stanie przekroczyć lambdy na poziomie 1,31. Pokazałem zatem sposób uniwersalny do znajdowania rozwiązań optymalnych posługując się analizą graficzną połączoną z metodą Zimmermanna.

11. Zadanie 9

Opisz zalety i wady modelowania opisanego problemu z wykorzystaniem zbiorów rozmytych.

Zalety:

- W teorii łatwa interpretacja wyników
- Łatwo o rozwiązanie kompromisowe (w tym konkretnym przypadku nie wystąpiło)
- Płynne przechodzenie między różnymi określeniami biznesowymi
- Większe możliwości do porozumienia w sprawie ustalenia zadowalających kryteriów (dość intuicyjne wskazywanie celu)

Wady:

- Może wymagać większej ilości obliczeń
- Mało intuicyjne rezultaty (często ogromna przestrzeń która nie jest sprawdzona, a zwrócony jest nam dowolny pasujący wynik)
- Problem z określeniem konkretnej metody agregacji (zaś wybór metody może mieć znaczący wpływ)
- Utrudnienia w otrzymywaniu nowych rozwiązań (rozdział 8 porównanie z metodą punktu odniesienia)
- Problematyczny stosunek do kryteriów spełnianych ze znacznym zapasem (problemy z dalszą optymalizacją)

12. Podsumowanie

Ostatecznie udało wykonać się wszystkie zadania, jednak otrzymane w moim przypadku dane niestety nie przedstawiały odpowiedniego rozróżnienia między poszczególnymi metodami. Utrudniło to końcowo analizy uzyskiwanych wyników, lecz każde polecenie udało się w pełni wykonać. Ostatecznie brak danych uniemożliwił postawienie oraz zweryfikowanie innych tez, które mogłyby się pojawić w szczególności podczas analizy porównawczej różnych rozwiązań.