Sprawozdanie z laboratorium Teorii Optymalizacji

Imię i nazwisko	Jacek Gołda
Temat ćwiczenia	Optymalizacja wielokryterialna
Data i godzina wykonania ćwiczenia	11 maja 2016, 14:00

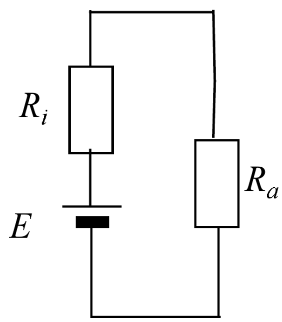
1 Wstęp

Celem laboratorium było rozwiązanie przykładowych problemów optymalizacji wielokryterialnej.

2 Ćwiczenie 1

Treść zadania:

Dla dwójnika elektrycznego:



$A \leqslant R_i \leqslant B$, $x = \frac{R_a}{R_i}$, $x \in [0, +\infty)$, kryteriami jakości są maksymalne wartości współczynnika sprawności i mocy wydzielanej na obciążeniu.

Za R_i przyjęto wartość 4. Parametr E mógł być dowolną wartością, przyjęto więc wartość 4.

2.1 Rozwiązanie

W rozważanym problemie zmienną decyzyjną jest x, przestrzenią decyzyjną jest zbiór liczb rzeczywistych, a zbiorem rozwiązań dopuszczalnych jest przedział $[0, +\infty)$. Należy dobrać rezystancję R_a aby jak najlepiej spełnione były obydwa kryteria. Znaleziony zostanie zbiór kompromisów.

Rozwiązanie rozpoczęto od wyznaczenia kryteriów jakości w zależności od zmiennej decyzyjnej x.

$$P_{Ra} = i^2 R_a = \left(\frac{E}{R_i + R_a}\right)^2 R_a = \left(\frac{E}{R_i + xR_i}\right)^2 xR_i = \frac{E^2}{R_i}\frac{x}{(1+x)^2}$$

$$\eta = \frac{P_{Ra}}{P_{Ra} + P_{Ri}} = \frac{i^2 R_a}{i^2 R_a + i^2 R_i} = \frac{R_a}{R_a + R_i} = \frac{xR_i}{xR_i + R_i} = \frac{x}{1+x}$$

Przyjęto, że:

$$Q_1(x) = P_{Ra} = \frac{E^2}{R_i}\frac{x}{(1+x)^2}$$

$$Q_2(x) = \eta = \frac{x}{1+x}$$

Następnie znaleziono zależność Q_1 od Q_2.

$$Q_2(x) = \frac{x}{1+x} \rightarrow x = \frac{Q_2(x)}{1-Q_2(x)}$$

Tą wartość x wstawiono do Q_1 i uporządkowano

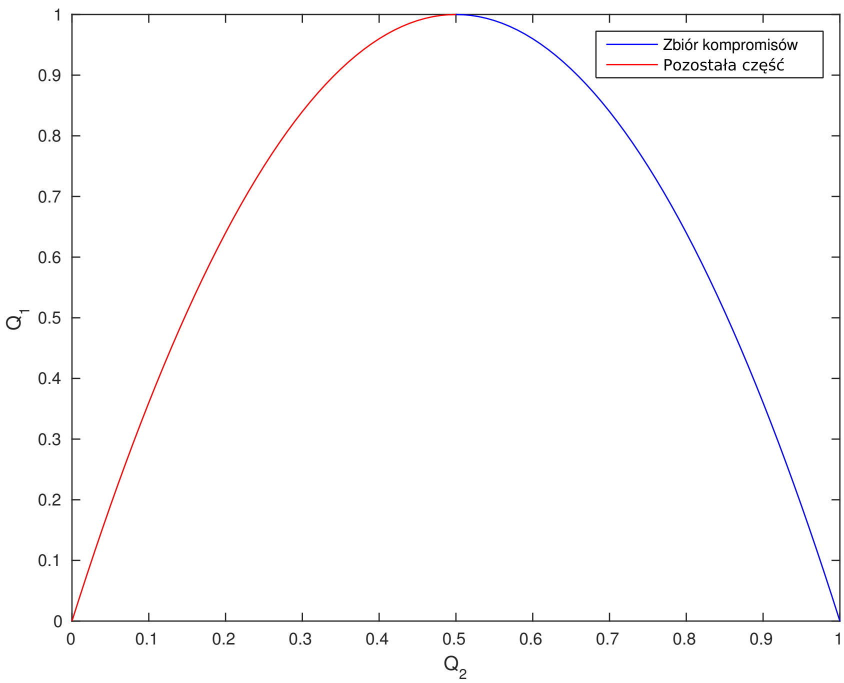
$$Q_1(x) = \frac{E^2}{R_i}\frac{x}{(1+x)^2} = \frac{E^2}{R_i}\frac{\frac{Q_2(x)}{1-Q_2(x)}}{(1+\frac{Q_2(x)}{1-Q_2(x)})^2} = \frac{E^2}{R_i}Q_2(x)(1-Q_2(x)) = -\frac{E^2}{R_i}Q_2(x)(Q_2(x)-1)$$

Uzyskano parabolę o ujemnym współczynniku przy najwyższej potędze. Wyznaczono zbiór wartości funkcji Q_1:

$$x \in [0, +\infty) \rightarrow Q_1(x) \in [0, 1]$$

Zbiorem kompromisów będzie fragment paraboli od $\max_{Q_1}(Q_2)$ do $\max_{Q_2}(Q_1)$, czyli dla $Q_2(x) \in [\frac{1}{2}, 1]$ i $Q_1(x) \in [0, 1]$. Oznacza to wartości $x \in [1, +\infty)$, czyli $R_a \in [4, +\infty)$ Ω

Narysowano wykres zależności Q_1 od Q_2.



Potwierdza on uzyskane wcześniej wyniki. Do rysowania wykorzystano następujący skrypt:

```
clear all;
close all;
clc

E = 4;
R_i = 4;

q2_komp = 0.5:0.01:1;
q1 = -E .^ 2 / R_i .* ( q2_komp .^ 2 - q2_komp );
```

```
plot(q2_komp, q1, 'b');
hold on;

q2_komp = 0:0.01:0.5;
q1 = -E .^ 2 / R_i .* ( q2_komp .^ 2 - q2_komp );

plot(q2_komp, q1, 'r');


xlabel('Q_2');
ylabel('Q_1');
legend('Zbior kompromisow', 'Pozostala czesc');
```

W obrębie zbioru kompromisów można zmieniać wartości oporu R_a, wedle innych preferencji. W tym zbiorze nie da się stwierdzić, że pewna zmiana poprawiła wartość wskaźników jakości.

W tym wypadku można zmieniać wartość rezystancji, w celu zwiększenia sprawności, albo zwiększenia wydzielanej mocy, w zależności od tego, który z tych parametrów jest bardziej w danej chwili istotny.

3 Ćwiczenie 3

Treść zadania:

Rozpatrzyć kryteria jakości związane z pożądanym działaniem i nietolerowanym preparatu medycznego wg poniższej tabeli:

Wariant	Q_1 [%]	Q_2 [%]
1	40	10
2	60	30
3	60	20
4	10	30
5	20	5
6	30	20
7	40	25

Q_1 — doskonałe efekty leczenia
Q_1 — całkowita nieskuteczność

Wartości funkcji celu są przekształcone w następujący sposób:

$$\widetilde{Q}_1(x) = \frac{Q_1(x)^2}{100} \qquad \widetilde{Q}_2(x) = (0.99)^n \frac{Q_2(x)^2}{100}$$

Za wartość parametru n przyjęto 4;

3.1 Rozwiązanie

Rozwiązanie zrealizowano w następujący sposób:

1. Na początku rozwiązywania zbiór Pareto (kompromisów) zawiera pierwszy wariant
2. Przyjęto, że jeden z wariantów jest lepszy od innego, jeżeli jest lepszy w myśl obydwu kryteriów jednocześnie. Oznacza to, że cechuje się większą wartością funkcji Q_1 i jednocześnie mniejszą wartością funkcji Q_2 niż inny wariant.

3. Przyjęto, że jeden z wariantów jest gorszy od innego, jeżeli jest gorszy w myśl obydwu kryteriów jednocześnie. Oznacza to, że cechuje się mniejszą wartością funkcji Q_1 i jednocześnie większą wartością funkcji Q_2 niż inny wariant.
4. Przeglądane są kolejne warianty.
   (a) Jeżeli aktualnie rozpatrywany wariant jest gorszy od któregokolwiek wariantu obecnego już w zbiorze Pareto, to jest on odrzucany i rozważany jest kolejny wariant.
   (b) Jeżeli aktualnie rozpatrywany wariant jest lepszy od jednego z wariantów obecnych w zbiorze Pareto, to wariant ze zbioru Pareto jest usuwany, a aktualnie rozpatrywany wariant jest wstawiany do zbioru Pareto.
   (c) Jeżeli nie jest spełniony żaden z powyższych warunków, to wariant jest dodawany do zbioru Pareto (ponieważ nie da się określić, czy jest gorszy, czy lepszy). Żaden wariant ze zbioru Pareto nie jest w tej sytuacji usuwany

Napisano następujący skrypt, implementujący ten algorytm:

```
clear all;
close all;
clc

% przygotowanie danych w odpowiedniej formie
input1 = [
    40, 60, 60, 10, 20, 30, 40];
input2 = [
    10, 30, 20, 30, 5, 20, 25];

n = 4;

input1 = input1 .^ 2 / 100;
input2 = (0.99).^ n * input2 .^ 2 / 100;
input = [input1;
        input2];

liczba_wariantow = size(input, 2);
input = [(1:liczba_wariantow)', input'];

% przygotowanie zmiennej na zbiÓr pareto
pareto = input(1, :);

% algorytm
for i=2:liczba_wariantow
    current_element = input(i, :);
    better = 0;
    worse = 0;
    for j=1:size(pareto, 1)
        pareto_element = pareto(j, :);
        if ( isBetter(pareto_element, current_element ) )
            better = 1;
            better_pareto_index = j;
            break;
        elseif ( isWorse(pareto_element, current_element ) )
            worse_pareto_index = j;
            worse = 1;
            break;
        end
    end

    if ( better == 1 )
        pareto(better_pareto_index, :) = current_element;
        fprintf('Replaced element %d with element %d\n', better_pareto_index, current_element(1));
    elseif ( worse == 0 )
        pareto = [pareto; current_element];
```

```
        fprintf('Added element %d to pareto\n', current_element(1));
    else
        fprintf('Skipped element %d, element %d is better\n', current_element(1), worse_pareto_index)
    end
end

pareto
input

figure
plot(pareto(:, 2:3), 'ro');
hold on;
plot(input(:, 2:3), 'bx');

print -depsc2 'plaszczyzna.eps'
```

Funkcje `isBetter` i `isWorse` wykorzystywane w skrypcie:

```
function better=isBetter(pareto_element, current_element)
%% sprawdza, czy element current_element
% jest lepszy niz pewny element zbioru pareto

    better = ( pareto_element(2) <= current_element(2) && pareto_element(3) >= current_element(3) );
    return
end
```

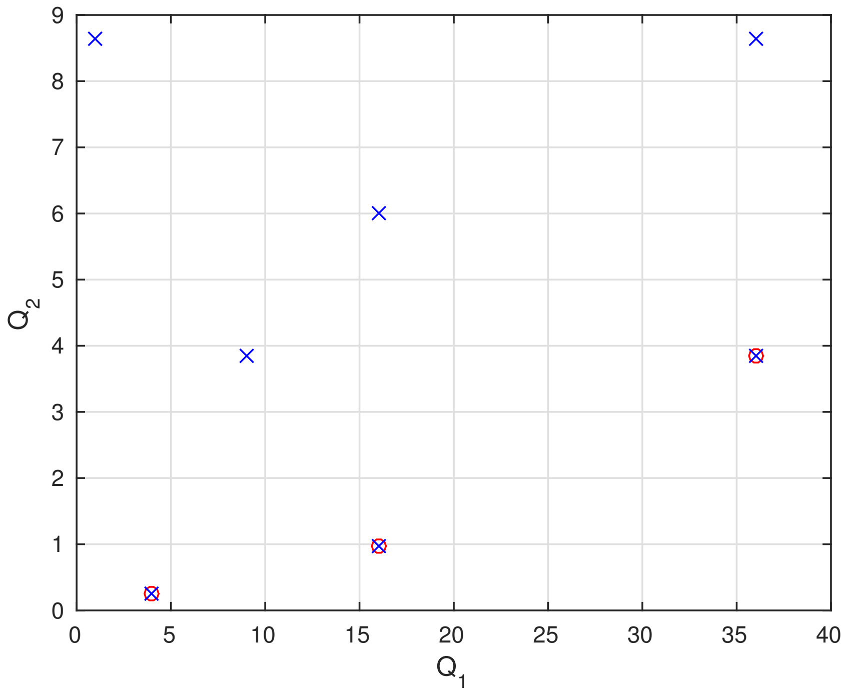
```
function worse=isWorse(pareto_element, current_element)
%% Funkcja sprawdza, czy element current_element
% jest gorszy od pewnego elementu zbioru pareto

    worse = ( pareto_element(2) >= current_element(2) && pareto_element(3) <= current_element(3) );
    return;
end
```

Uzyskano zbiór Pareto składający się z wariantów o numerach 1, 3, 5. Pełny wydruk uzyskiwany z programu (uwzględniający umotywowanie decyzji podejmowanych przez algorytm) przedstawiony jest poniżej.

```
Added element 2 to pareto
Replaced element 2 with element 3
Skipped element 4, element 1 is better
Added element 5 to pareto
Skipped element 6, element 1 is better
Skipped element 7, element 1 is better
pareto =
    1.0000   16.0000    0.9606
    3.0000   36.0000    3.8424
    5.0000    4.0000    0.2401
input =
    1.0000   16.0000    0.9606
    2.0000   36.0000    8.6454
    3.0000   36.0000    3.8424
    4.0000    1.0000    8.6454
    5.0000    4.0000    0.2401
    6.0000    9.0000    3.8424
    7.0000   16.0000    6.0037
```

Zbiór Pareto jest określony w powyższym wydruku jako „pareto", dane wejściowe są określone jako „input". Powyżej znajduje się spis kolejnych decyzji algorytmu podejmowanych dla kolejnych wariantów.

Następnie zaznaczono zbiór Pareto na płaszczyźnie, gdzie współrzędnymi są wartości funkcji Q_1 i Q_2.



Niebieskim krzyżykiem oznaczono dane wejściowe, a wśród nich czerwonym kółkiem zaznaczono punkty ze zbioru Pareto.

Uzyskany zbiór jest zbiorem kompromisów — w jego obrębie wybór jednej z możliwości może być podyktowany innymi preferencjami, gdyż nie można stwierdzić, który jest optymalny w sensie obydwu wskaźników jakości.

4 Wnioski

W trakcie laboratorium rozwiązano dwa problemy optymalizacji wielokryterialnej. Jeden z nich był problemem ciągłym, a drugi problemem dyskretnym. Wyznaczono zbiory kompromisów dla obydwu problemów. W obrębie zbioru kompromisów można wybrać jedną z decyzji, wybór może być podyktowany innymi względami. Dla decyzji ze zbioru kompromisów nie można stwierdzić, która jest optymalna w myśl wskaźników jakości wykorzystanych przy optymalizacji (ze względu na pojawiające się pomiędzy wskaźnikami sprzeczności — zwiększenie pewnej wartości spowoduje poprawę jednego ze wskaźników, ale jednocześnie spowoduje pogorszenie innego).