Programowanie matematyczne - zadanie 4

Paulina Przybyłek

Grupa laboratoryjna C
25 listopada 2022

Oświadczam, że niniejszy raport wraz z załączonymi m-plikami zostały wykonane przez mnie samodzielnie.

1 Treść zadania

Zadanie dotyczy uzyskania rozwiązania zadania programowania liniowego (ZP) poprzez rozwiązanie zadania do niego dualnego (ZD) i na podstawie rozwiązania optymalnego do niego znalezienie rozwiązania optymalnego do głównego zadania ZP.

Do rozwiązania zadania należy wykorzystać własną implementację algorytmu sympleks oraz wbudowaną metodę z Matlaba - funkcję linprog wykorzystującą ten algorytm. Raport zawiera skrócony opis zadania wraz z wynikami testów i odpowiedzią na kluczowe pytania do omawianego problemu.

1.1 Zadanie programowania liniowego i zadanie do niego dualne

Podane zadanie ZP ma następującą formę:

$$\max_{x\in\Omega} c^T x \tag{1}$$

$$\Omega : \begin{cases} Ax \leqslant b \\ 0 \leqslant x \leqslant g \ \ (g > 0) \end{cases} \tag{2}$$

gdzie $c,\ x \in \mathcal{R}^n$, $b \in \mathcal{R}^m$, $A \in \mathcal{R}^{mxn}$, a parametry m i n są równe 5. W rozwiązywanym zadaniu wartości $c,\ A,\ b$ są losowane z przedziału $[-5, 5]$, natomiast wartości wektora g z przedziału $[1, 30$. Dla tak skonstruowanego zadania zdefiniowano zadanie ZD, gdzie ograniczenie xg traktowano jak ograniczenie do równania, a nie warunki brzegowe, czyli rozważane zadanie (1) ma Ω postaci:

$$\Omega : \begin{cases} Ax \leqslant b \\ x \leqslant g \\ x \geqslant 0 \end{cases}$$

Stąd po przekształceniu tego do zadania dualnego otrzymuje się następujące zadanie do rozwiązania:

$$\min_{y\in\Omega'} [b^T, g^T]y \tag{3}$$

$$\Omega' : \begin{cases} [A^T, I]y \geqslant c \\ y \geqslant 0 \end{cases}$$

To zadanie ZD jest rozwiązywane przez algorytm sympleks w omawianym zadaniu, jednak aby to było możliwe należy doprowadzić je do odpowiedniej formy - **postaci standardowej**. Postać tą przedstawia wzór 4, gdzie mamy maksymalizację zamiast minimalizacji funkcji. Natomiast przekształcona Ω' zawiera teraz ograniczenia równościowe zamiast nierównościowych, co osiągnięto poprzez odjęcie nowych zmiennych tzw. **zmiennych dopełniających**, które są zmiennymi technicznymi dla tego zadania. W ten sposób osiągnięto postać standardową zadania ZD.

$$\max_{y\in\Omega'} (-1)[b^T, g^T]y \tag{4}$$

$$\Omega' : \begin{cases} [A^T, I]y - y_d = c \\ y \geqslant 0 \ (warunki\ brzegowe) \end{cases} \tag{5}$$

Zadanie opisane przez wzory 4 i 5 nie ma jednak **postaci kanonicznej**, gdyż $c \in \mathcal{R}^n$, stąd może przyjmować wartości ujemne. Aby uzyskać postać kanoniczą należy odpowiednie równanie przemnożyć przez wartość -1. Wówczas w bazie A_B początkowej znajdą się zmienne y chyba, że dany wiersz odpowiadający tej zmiennej y_i został przemnożony to wówczas do bazy będzie wzięta zmienna dopełniająca jej odpowiadająca, tak aby zmienne bazowe miały macierz bazy jednostkową (gdybyśmy tego nie zmieniali to A_B nie byłaby jednostkowa, gdyż pojawiłyby się wartości -1 w niej). Stąd wiadomo jak powstaje baza do tabelki sympleks, wektor c również jest nieujemny, a w funkcji 4, którą sympleks maksymalizuje zmienne dopełniające mają wartości zero, czyli rozwiązywane zadanie ma postać: $\max(-1)[b^T, g^T]y + 0y_d$.

1.2 Metoda sympleks dla rozwiązywanego problemu

Mając bazowe rozwiązanie dopuszczalne (BRD) dla danej bazy opartej na macierzy jednostkowej będącej podmacierzą A' (macierz A' to macierz $[A^T, I, -I$ przekształcona poprzez przemnożenie odpowiednich wierszy przez -1), wyliczane są współczynniki optymalności odpowiadające aktualnemu BRD. Jeśli wszystkie współczynniki są większe bądź równe zero to algorytm się zatrzymuje, w przeciwnym przypadku poszukiwany jest nowy element y, który zastąpi stary w bazie. Dzieje się to na podstawie kryterium wejścia i wyjścia z bazy, a biorą w nim udział te elementy co mają ujemne wartości współczynników optymalności. Po wybraniu nowego elementu zastępuje on stary i wyliczane są nowe wartości macierzy A' i wektora c na podstawie odpowiednich wzorów. Wówczas mamy nowe BRD i wracamy do punktu sprawdzania współczynników. Dzieje się tak aż do zatrzymania algorytmu gdy znajdziemy rozwiązanie bądź okaże się, że zadanie jest sprzeczne.

Po rozwiązaniu zadania ZD przez algorytm, czyli znalezieniu rozwiązania optymalnego (RO), z ostatniej tabeli sympleks można odczytać RO dla zadania ZP. Współczynniki $z_i - c_i$ (wskaźniki optymalności) odpowiadające zmiennym dopełniającym y_{d_i} odpowiadają wartościom x_i rozwiązania optymalnego dla zadania ZP. Jest to możliwe dlatego, że ZP i ZD są to zadania symetrycznie sprzężone i mają skończone rozwiązanie.

2 Rozwiązanie Matlab

Rozwiązanie zadania zawiera się trzerech plikach .m. Plik *symplex_zd.m* zawiera funkcje implementujące metodę sympleks rozwiązującą zadanie ZP poprzez zadanie ZD. Zwraca ona rozwiązania optymalne obu zadań oraz wynik funkcji. Komentarze informują w krótki sposób co się tam dzieje. W pliku *script.m* jest przykładowe wywołanie zadania dla własnej implementacji oraz dla ZP i ZD przez linprog. Natomiast pliki *tests1.m* i *tests2.m* wykorzystane zostały do realizacji części testowej opisanej w kolejnej sekcji.

2.1 Przykład

W pliku *script.m* dla ustalonego ziarna losowego można policzyć przykład zadania. Rysunek 1 przedstawia wylosowane wartości macierzy i wektorów definiujących zadanie. Tak podane (oczywiście pamiętając o zamianie na minimalizację funkcji) podano funkcji linprog i uzyskano rozwiązanie:

$$x = [10, 7, 0, 0, 0]^T$$

Następnie przekształcono dane macierze i wektory pod zadanie ZD i również podano funkcji linprog uzyskując wynik:

$$y = [0, 1.6667, 0, 0, 0, 0.6667, 0, 0, 0, 0]^T$$

Obserwacją, której można dokonać to to, że mnożniki Lagrange'a uzyskane przy obliczaniu RO dla zadania ZP w linprog, które są następujące:

$$L = [0, 1.6667, 0, 0, 0]^T$$

odpowiadają pierwszym wartościom wektora y, tym które dotyczą nierówności $Ax \leqslant b$ (tzn. są takie same jak one).

```
c =
     4    -5    -5    -4     4
A =
    -1    -4    -1     2     3
     2    -3     2    -1     5
    -5    -2    -3     1    -2
    -2    -1     4    -4     2
    -4     0    -5    -3     4
b =
    -4
    -1
     5
     0
     2
g =
    10
    21
    26
     1
    23
```

Rysunek 1: Przyjęte wartości wektorów i macierzy do zadania ZP określonego wzorem 1 i 2.

Poza rozwiązaniem tego zadania przy użyciu linprog, zastosowano własną implementację sympleks. Uzyskane wyniki x oraz y pokrywały się z wynikami wbudowanej metody Matlaba. Własna implementacja oblicza sympleksem jedynie zadanie dualne, a wynik RO dla ZP odczytuje z ostatniej tabeli z odpowiednich wartości wektora $z - c$ (rozdział 1.2, ostatni akapit). Rysunek 2 przedstawia ostatnią tabelę metody sympleks - wartości wektora nazwanego tam b (jest to wektor prawych stron ograniczeń) są zaokrąglone. Można z niej odczytać RO dla zadania ZD oraz ZP. W bazie mamy tylko y_2 i y_6 oraz zmienne dopełniające, więc wektor y ma same zera poza 2 i 6 elementem. Natomiast wektor x to wartości $z - c$ odpowiadające zmiennym dopełniającym, czyli od y_{11} do y_{15}.

		4	1	-5	0	-2	-10	-21	-26	-1	-23	0	0	0	0	0	
c_B	x_B	y1	y2	y3	y4	y5	y6	y7	y8	y9	y10	y11	y12	y13	y14	y15	b
-10	y6	-3.67	0	-6.33	-2.67	-4	1	0.67	0	0	0	-1	-0.67	0	0	0	0.67
0	y15	3.67	0	5.33	-0.33	-4	-0	-1.67	-0	-0	-1	0	1.67	0	0	1	4.33
0	y13	3.67	0	4.33	-3.33	5	0	-0.67	-1	0	0	0	0.67	1	0	0	8.33
0	y14	-3.33	0	-1.67	3.67	3	0	0.33	0	-1	0	0	-0.33	0	1	0	2.33
1	y2	1.33	1	0.67	0.33	0	0	-0.33	0	0	0	0	0.33	0	0	0	1.67
	z	38	1	64	27	40	-10	-7	0	0	0	10	7	0	0	0	-5
	z-c	34	0	69	27	42	0	14	26	1	23	10	7	0	0	0	>=0

Rysunek 2: Ostatnia tabela algorytmu sympleks dla zadania dualnego wybranego przykładu zadania ZP.

3 Testy

Zrealizowano dwa testy:

- przeprowadzono 100 razy rozwiązanie zadania ZP dla funkcji linprog oraz własnej implementacji w celu porównania skuteczności implementacji;
- przeprowadzono 100 razy rozwiązanie zadań ZP, które mają RO, algorytmowi własnej implementacji w celu zbadania skuteczności.

3.1 Test 1

Dla 100 wywołań obliczano RO dla różnych losowych danych. Rysunek 3 przedstawia skuteczność algorytmu własnego i z funkcji linprog w znajdowaniu rozwiązań. Skuteczność jest w nich taka sama, więc można powiedzieć, że algorytm sympleks we własnej implementacji ma identyczną, tj. 100% skuteczność w znalezieniu rozwiązania, jeśli ono jest skończone.

```
MATLAB

Liczba skończonych rozwiązań: 57
Liczba braku rozwiązań: 43

WŁASNA IMPLEMENTACJA SYMPLEKS

Liczba skończonych rozwiązań: 57
Liczba braku rozwiązań: 43
```

Rysunek 3: Liczba RO i ich braku dla Matlaba i własnej implementacji algorytmu sympleks rozwiązującego ZPL poprzez zadanie dualne.

3.2 Test 2

Test ten różni się od poprzedniego tym, że sprawdzano tylko te zadania, które mają skończone rozwiązanie znalezione przez linprog. Rysunek 4 przedstawia liczbę znalezionych i nieznalezionych rozwiązań dla własnej implementacji sympleks. Znaleziono 100% rozwiązań, więc to daje identyczną skuteczność.

```
WŁASNA IMPLEMENTACJA SYMPLEKS

Liczba skończonych rozwiązań: 100
Liczba braku rozwiązań: 0
```

Rysunek 4: Liczba RO i ich braku dla własnej implementacji algorytmu sympleks rozwiązującego ZPL poprzez zadanie dualne.