



Programowanie matematyczne - laboratorium 3

Algorytm Simplex

Dawid Maksymowski, gr. D

17 listopada 2023

Spis treści

1	Zagadnienie	1
1.1	Treść zadania	1
1.2	Kroki algorytmu	1
2	Analiza algorytmu i przykłady obliczeniowe	2
3	Oświadczenie o samodzielności	2

1 Zagadnienie

1.1 Treść zadania

Celem zadania jest zaimplementowanie algorytmu sympleks rozwiązującego następujące zagadnienie:

$$\begin{aligned} \max_{x \in \Omega} c^T x \\ \Omega : \begin{cases} Ax \leq b, b > 0 \\ 0 \leq x \leq u, u > 0 \end{cases} \\ c, u, x \in \mathbb{R}^n, b \in \mathbb{R}^m, A \in \mathbb{R}^{m \times n} \\ n = 10, m = 5 \end{aligned}$$

1.2 Kroki algorytmu

Danymi wejściowymi algorytmu są wektory c, b, u oraz macierz A .

1. Ponieważ zadanie w sformułowaniu (1.1) nie jest zadane w postaci standardowej ZPL, przed rozpoczęciem właściwej pracy algorytmu, należy je do takiej postaci przekształcić. Cały algorytm pracuje na rozszerzonych macierzy oraz wektorach tak, że startowa macierz rozszerzona A_{ext} oraz wektor b_{ext} wyglądają następująco:

$$\begin{bmatrix} A & I & 0 \\ I & 0 & I \end{bmatrix} \begin{bmatrix} b \\ u \end{bmatrix}$$

gdzie I - macierz jednostkowa.

Następnie algorytm biegnie zgodnie ze swoją standardową procedurą:

2. Wyznaczana jest początkowa baza: $\{x_{n+1}, \dots, x_{2n+m}\}$ (czyli początkowe x_{BRD} jest wektorem zerowym).
3. W pętli:
 - (a) Wyznaczenie wskaźników optymalności odpowiadających aktualnemu BRD

- (b) Znalezienie indeksu nie-bazowej zmiennej wprowadzanej do części bazowej nowego BRD zgodnie ze standardowym kryterium wejścia.
- (c) Wyznaczenie indeksu bazowej zmiennej wyprowadzanej z części bazowej aktualnego BRD zgodnie ze standardowym kryterium wyjścia.
- (d) Połączona macierz $[A_{ext} \ b_{ext}]$ jest gaussowana tak, żeby wektor odpowiadający zmiennej wprowadzanej do bazy był wektorem jednostkowym, z jedynką w kolumnie, w której znajduje się indeks zmiennej wyprowadzanej z bazy.

Pętla wykonuje się tak długo, jak istnieje jakiś element w wektorze $z - c$, który jest mniejszy od 0. Co każdy krok wykonania pętli, do pliku out.txt są wypisywane tabelki oraz rezultat obliczeń.

2 Analiza algorytmu i przykłady obliczeniowe

W pliku main.m można wywołać 2 funkcje:

- `simple_test()`, która liczy algorytm na podstawie jednego wylosowanego przykładu
- `positivity_test()`, która uruchamia algorytm dla N (np. N=10000) wylosowanych przykładów i oblicza poprawność algorytmu, wypisując do konsoli procent przypadków, w których wynik wyszedł równy z algorytmem MATLAB linprog z dokładnością do EPS (przekazane jako parametr).

Testy wskazują, że zaimplementowany algorytm posiada ok. 99% skuteczności dla wartości EPS=0.00001.

Algorytm, poza wskazaniem rozwiązania, sprawdza również, czy jest to jedyne rozwiązanie. A jeśli nie jest - wskazuje inne rozwiązanie optymalne.

3 Oświadczenie o samodzielności

Niniejszym oświadczam, że powyższa praca, wraz załączonym kodem w formie plików '.m', została wykonana przeze mnie w pełni samodzielnie.

Dawid Maksymowski