Wydział Matematyki i Nauk Informacyjnych
POLITECHNIKA WARSZAWSKA

Programowanie Matematyczne - Laboratorium 6

Zewnętrzna kwadratowa funkcja kary

Dawid Maksymowski, gr. D

5 stycznia 2024

Spis treści

1 Zagadnienie

1.1 Sformułowanie

Należy znaleźć rzut punktu p na zbiór Ω zdefiniowany następująco:

$$\Omega = \left\{ x \in \mathbb{R}^n : \begin{matrix} Ax = b \\ x \geqslant 0 \end{matrix} \right\} \tag{1}$$
$$A \in \mathbb{R}^{m \times n},\ m < n,\ r(A) = m,\ b \in \mathbb{R}^m$$

1.2 Postać funkcji kwadratowej

Zadanie (1) można sformułować równoważnie jako problem minimalizacji funkcji kwadratowej z ograniczeniami:

$$\min_{x \in \Omega} \frac{1}{2} \|x - p\|^2$$
$$\Omega = \left\{ \begin{matrix} Ax - b = 0 \\ x \geqslant 0 \end{matrix} \right\} \tag{2}$$

Funkcję celu można inaczej zapisać wzorem (3).

$$f_c(x) = \frac{1}{2}\|x - p\|^2 = \frac{1}{2}(x - p)^T(x - p) = \frac{1}{2}\left(x^T x - 2p^T x + p^T p\right) \tag{3}$$

Aby problem ten móc obliczyć przy pomocy funkcji `quadprog` z biblioteki funkcji MATLAB, w dalszej części dokumentu będziemy rozpatrywać problem minimalizacji w następującym, uproszczonym sformułowaniu:

$$
\begin{gathered}
\min_{x\in\Omega} \frac{1}{2}x^T Ix - p^T x \\
\Omega = \begin{Bmatrix} Ax - b = 0 \\ x \geqslant 0 \end{Bmatrix}
\end{gathered}
\tag{4}
$$

Należy pamiętać, że aby uzyskać wartość funkcji celu dla znalezionej wartości x, należy dodać do wartości funkcji celu czynnik $p^T x$. Poza tym, rozwiązanie problemu (2) jest równoważne rozwiązaniu problemu (4).

1.3 Istnienie rozwiązania

Przeprowadzone zostały testy badające statystyczne istnienie rozwiązania dla losowanych zadań. Wartości macierzy A oraz wektorów b, p były losowane z przedziału $[-5, 5]$ dla $n = 10$ oraz różnych wartości m. Warto zwrócić uwagę, że na rozwiązywalność problemu ma jedynie wpływ wartości A i b a wartość p może być dowolna. Jeżeli istnieje jakikolwiek punkt w zbiorze Ω to istnieje również rozwiązanie problemu minimalizacji.

Zostały zbadane wyniki (wartość `exitflag`) zwrócone przez funkcję `quadprog`. Przykładowy wynik testów przedstawia tabela 1. Dla każdej wartości m wykonano $N = 10000$ testów. Wyniki pokazują, że wraz ze zwiększaniem się wartości m (coraz bardziej kwadratowa macierz), zwiększa się również prawdopodobieństwo, że wylosowany układ będzie sprzeczny.

`exitflag`\\m	m=3	m=5	m=7
2	0	0	0
1	9428	6195	1686
0	0	5	2
-2	572	3798	8311
-3	0	0	1
-6	0	0	0
-8	0	2	0

Tabela 1: Rozkład zwróconej wartości flagi `exitflag` dla różnych wartości m.

Znaczenie wartości `exitflag` jest następujące:

- 2 lub 1 – znaleziono RO.
- 0 – liczba iteracji przekroczyła 200 (brak informacji o rozwiązywalności problemu).
- -2 – problem jest sprzeczny LUB algorytm osiągnął wymaganą dokładność ale ograniczenia nie są spełnione.
- -3 – problem jest nieograniczony.
- -6 – problem jest niewypukły.
- -8 – nie udało się obliczyć kierunku działania algorytmu w którymś kroku.

1.4 Warunki Kuhna-Tuckera

Badane jest zadanie z ograniczeniami równościowymi przy nieujemności zmiennych decyzyjnych (jak w sformułowaniu (4)). Funkcja Lagrange'a jest następująca:

$$
L(x, \lambda_E) = \frac{1}{2}x^T Ix - p^T x + \sum_{i=1}^{m} \lambda_{Ei}(A_i x - b_i) \tag{5}
$$

$$
= \frac{1}{2}x^T Ix - p^T x + \lambda_E^T(Ax - b) \tag{6}
$$

gdzie A_i jest i-tym wierszem macierzy A, a b_i jest i-tym elementem wektora b.
Gradienty funkcji $f(x) = \frac{1}{2}x^T Ix - p^T x$ oraz $g_i(x) = A_i x - b_i$ są równe:

$$
\begin{aligned}
\nabla f(x) &= x - p \\
\nabla g_i(x) &= A_i{}^T
\end{aligned}
\tag{7}
$$

Warunki Kuhna-Tuckera dla tego zadania są następujące ($x^\star$ – RO zadania):

1. $\nabla_x L(x^\star, \lambda_E) \geqslant 0$
2. $\nabla_{\lambda_E} L(x^\star, \lambda_E) = 0$
3. $x^\star \geqslant 0$
4. $x^{\star T} \cdot \nabla_x L(x^\star, \lambda_E) = 0$
5. $\lambda_E \in \mathbb{R}$

1. $x^\star - p + A^T \lambda_E \geqslant 0$
2. $Ax^\star - b = 0$
3. $x^\star \geqslant 0$
4. $x^{\star T}(x^\star - p + A^T \lambda_E) = 0$
5. $\lambda_E \in \mathbb{R}$

Warunki te są sprawdzane w funkcji `callQuadprog`. Oczywiście wszystkie są spełnione dla każdego przypadku, dla którego rozwiązanie istnieje.

2 Algorytm zewnętrznej funkcji kary

2.1 Funkcja kary

Problem (4) można zapisać równoważnie, w postaci standardowej:

$$\begin{gathered}\min_{x\in\Omega} \frac{1}{2}x^T I x - p^T x \\ \Omega = \begin{Bmatrix} Ax - b = 0 \\ -x \leqslant 0 \end{Bmatrix}\end{gathered} \tag{8}$$

co umożliwia nam bezpośrednie zastosowanie wzoru na zewnętrzną funkcję kary funkcji kwadratowej jak w równaniu (9),

$$P_k(x, r_k) = r_k \left(\sum_{j\in E} h_j^2(x) + \sum_{i\in I} [\max(0, g_i(x))]^2 \right) \tag{9}$$

gdzie E - zbiór ograniczeń równościowych, I - zbiór ograniczeń nierównościowych zbioru Ω.

$$P_k(x, r_k) = r_k \left(\sum_{j=1}^{m} (A_j x - b_j)^2 + \sum_{i=1}^{n} [\max(0, -x_i)]^2 \right) \tag{10}$$

Wówczas minimalizowana w każdym kroku algorytmu funkcja F_k oraz jej gradient (po kilku przekształceniach) ∇F_k wyglądają następująco:

$$F_k(x, r_k) = \frac{1}{2}x^T x - p^T x + r_k \left(\sum_{j=1}^{m} (A_j x - b_j)^2 + \sum_{i=1}^{n} [\max(0, -x_i)]^2 \right) \tag{11}$$

$$\nabla F_k(x, r_k) = x - p + r_k \left(2A^T(Ax - b) + \min(0, 2x) \right) \tag{12}$$

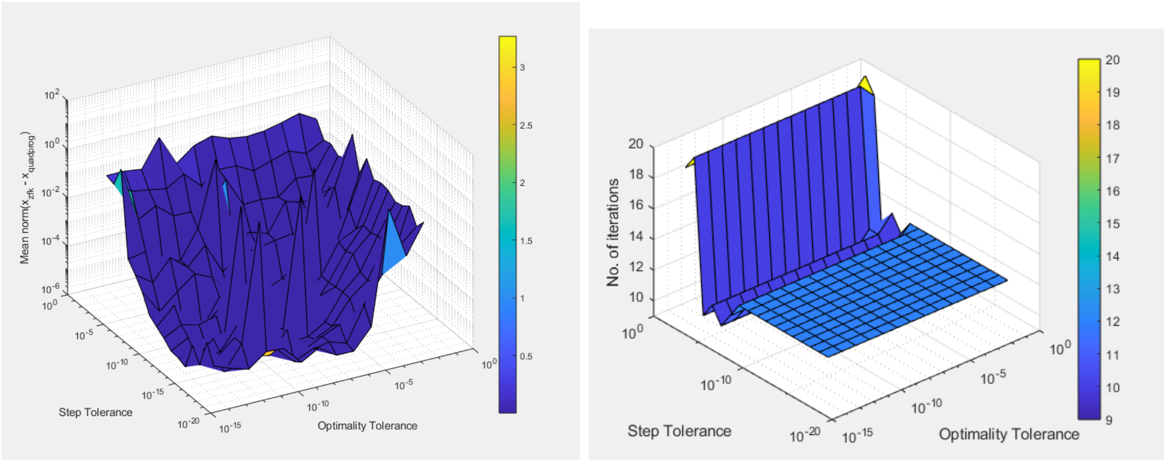
gdzie $\min(0, v)$ (v - wektor) oznacza minimum po każdej współrzędnej wektora v z zerem.

2.2 ZFK

Ogólną ideą stojącą za metodami funkcji kary jest przeniesienie ograniczeń na zbiór Ω do funkcji celu. Innymi słowy, przekształcamy zadanie programowania z ograniczeniami do zadania bez ograniczeń (być może rozwiązywanego wielokrotnie, iteracyjnie) ale taki sposób, aby nieopłacalnym był wybór wartości spoza zbioru Ω. W przypadku zewnętrznej funkcji kary skupiamy się na tym, aby „karać" funkcję poza zbiorem Ω – coraz agresywniej z każdym kolejnym przybliżeniem.

Funkcja ZFK implementuje algorytm zewnętrznej funkcji kary, wykorzystując do minimalizacji w każdym kroku wbudowaną funkcję `fminunc`. `fminunc` przyjmuje parametry obliczeń *OptimalityTolerance* i *StepTolerance*. Rysunek 1a przedstawia średnią z normy $x_{quadprog} - x_{zfk}$ (gdzie $x_{quadprog}$ to wartość „dokładna" obliczona przez funckję MATLABową) dla różnych wartości tych parametrów. Dla każdej pary wartości ($OptimalityTolerance$, $StepTolerance$) wykonano 10 testów i średnią wyników wprowadzono na wykres. Brano pod uwagę losowe zadania, dla których istniało rozwiązanie. Widać, że im mniejsze wartości, tym wynik jest dokładniejszy.

Rysunek 1b przedstawia natomiast wykres liczby iteracji dla różnych wartości parametrów. Można z niego wywnioskować, że minimum występuje mniej więcej dla wartości $StepTolerance = 1e-4$ i zbieżność dla tego przypadku następuje w 10 iteracjach. Biorąc jednak pod uwagę dodatkowo dokładność (rys. 1a) warto wziąć



(a) Średnia norma wyników dla różnych wartości *OptimalityTolerance*, *StepTolerance*. Dla przejrzystości wszystkie osie są w skali logarytmicznej.

(b) Liczba wykonanych iteracji algorytmu dla pewnego zadania, dla różnych wartości *OptimalityTolerance*, *StepTolerance*. $e = 1e - 6$.

Rysunek 1: Testy parametrów algorytmu ZFK.

mniejsze wartości tego parametru, kosztem zaledwie dodatkowych dwóch (w tym przypadku) iteracji. Optimum następuje dla ok. $1e - 8$ i dalej wykres się już wypłaszcza. Wartość *OptimalityTolerance* ma znikomy wpływ na liczbę iteracji.

Dla innych wylosowanych zadań wykresy wyglądają bardzo podobnie.

2.3 Istnienie rozwiązania

Rozwiązanie znalezione przez algorytm iteracyjny należy na końcu „przepuścić" przez warunki Kuhna-Tuckera, aby sprawdzić, czy jest to rzeczywiście rozwiązanie problemu. Nie każde bowiem zadanie (2) posiada rozwiązanie. W tym celu wykorzystany jest warunek 4., tj.: ${x^{\star}}^T(x^{\star} - p + A^T\lambda_E) = 0$. Z niego powstaje układ równań, który, jeśli jest niesprzeczny, to zwraca wartości λ_E.

Skuteczność algorytmu ZFK w porównaniu do `quadprog` wynosi około 62% (wykonano 1000 testów i jako zgodność traktowano jednoczesny brak rozwiązania albo jednoczesne istnienie rozwiązania, gdzie norma różnicy $(x_{zfk} - x_{quadprog})$ jest mniejsza niż 5.

3 Oświadczenie o samodzielności

Niniejszym oświadczam, że powyższa praca, wraz z załączonym kodem MATLAB, została wykonana przeze mnie w pełni samodzielnie.

Dawid Maksymowski