

1 Preliminaria

Ćwiczenie opracowano na podstawie [1].

1.1 Notacja

Często będziemy pisać $i \in \mathbf{m}$ zamiast $i = 1, \dots, m$.

1.2 Optymalizacja z ograniczeniami nierównościowymi

Rozpatrujemy zadanie optymalizacji wypukłej

$$\begin{aligned} & \underset{\mathbf{x} \in \mathbb{R}^n}{\text{minimize}} && f_0(\mathbf{x}) \\ & \text{subject to} && f_i(\mathbf{x}) \leq 0, \quad i \in \mathbf{m} \end{aligned} \quad (1)$$

Dla zadania tego możemy zdefiniować zbiór dopuszczalny (*feasible set*)

$$\Omega = \{\mathbf{x} \in \mathbb{R}^n \mid f_i(\mathbf{x}) \leq 0, \quad i \in \mathbf{m}\}, \quad (2)$$

który jest zbiorem wypukłym z uwagi to, że funkcje $f_i(\mathbf{x})$, $i \in \mathbf{m}$, są wypukłe. Zadania powyższego typu można rozwiązywać metodami funkcji bariery (*barrier function*). Zastosowanie funkcji bariery umożliwia sprowadzenie zadań optymalizacji z ograniczeniami nierównościowymi do sekwencji zadań bez ograniczeń do rozwiązania których możemy zastosować poznane wcześniej metody np. metodę Newtona.

Będziemy w dalszym ciągu zakładać, że funkcje ograniczeń nierównościowych $f_i(\mathbf{x})$, $i \in \mathbf{m}$ oraz funkcja celu f_0 są dwukrotnie różniczkowalne. Dodatkowo zakładamy, że wartość optymalna p^* funkcji celu jest skończona i osiągana w punkcie optymalnym \mathbf{x}^* , oraz że zadanie jest ściśle dopuszczalne (*strictly feasible*), tzn. że istnieje pewien punkt (wektor) $\bar{\mathbf{x}} \in \mathbb{R}^n$ taki, że

$$f_i(\bar{\mathbf{x}}) < 0, \quad i \in \mathbf{m}. \quad (3)$$

1.3 Logarytmiczna funkcja bariery

Funkcję ciągłą $\varphi : \mathbb{R}^n \rightarrow \mathbb{R}$ nazywamy wypukłą funkcją bariery (*convex barrier function*) dla zbioru Ω , jeśli jest wypukłą na zbiorze Ω oraz $\varphi(\mathbf{x}) \rightarrow \infty$ dla \mathbf{x} dążących do brzegu zbioru Ω . Najczęściej używa się tzw. logarytmicznej funkcji bariery (*logarithmic barrier function*)

$$\varphi(\mathbf{x}) = -\sum_{i=1}^m \log(-f_i(\mathbf{x})). \quad (4)$$

Zauważmy, że dla $i \in \mathbf{m}$, $k \in \mathbf{n}$ mamy

$$\frac{\partial \log(-f_i(\mathbf{x}))}{\partial x_k} = \frac{1}{f_i(\mathbf{x})} \frac{\partial f_i(\mathbf{x})}{\partial x_k}, \quad (5)$$

zatem dla $i \in \mathbf{m}$

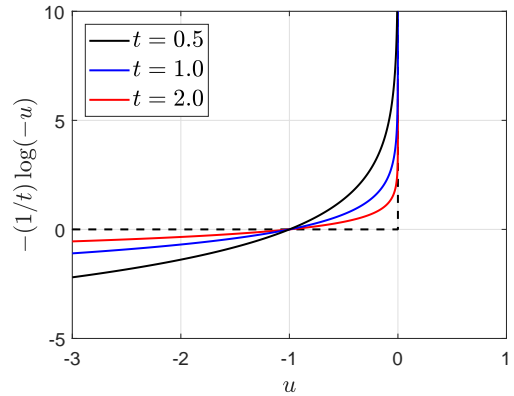
$$\nabla \log(-f_i(\mathbf{x})) = \begin{bmatrix} \frac{1}{f_i(\mathbf{x})} \frac{\partial f_i(\mathbf{x})}{\partial x_1} \\ \frac{1}{f_i(\mathbf{x})} \frac{\partial f_i(\mathbf{x})}{\partial x_2} \\ \vdots \\ \frac{1}{f_i(\mathbf{x})} \frac{\partial f_i(\mathbf{x})}{\partial x_n} \end{bmatrix} = \frac{1}{f_i(\mathbf{x})} \nabla f_i(\mathbf{x}). \quad (6)$$

Gradient funkcji ϕ jest zatem równy

$$\begin{aligned} \nabla \varphi(\mathbf{x}) &= -\sum_{i=1}^m \nabla \log(-f_i(\mathbf{x})) \\ &= -\sum_{i=1}^m \frac{1}{f_i(\mathbf{x})} \nabla f_i(\mathbf{x}), \end{aligned} \quad (7)$$

zaś jej hesjan jest równy

$$\begin{aligned} \nabla^2 \varphi(\mathbf{x}) &= \begin{bmatrix} \frac{\partial^2 \varphi(\mathbf{x})}{\partial x_1^2} & \frac{\partial^2 \varphi(\mathbf{x})}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 \varphi(\mathbf{x})}{\partial x_1 \partial x_n} \\ \frac{\partial^2 \varphi(\mathbf{x})}{\partial x_2 \partial x_1} & \frac{\partial^2 \varphi(\mathbf{x})}{\partial x_2^2} & \cdots & \frac{\partial^2 \varphi(\mathbf{x})}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 \varphi(\mathbf{x})}{\partial x_n \partial x_1} & \frac{\partial^2 \varphi(\mathbf{x})}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 \varphi(\mathbf{x})}{\partial x_n^2} \end{bmatrix} \\ &= \sum_{i=1}^m \frac{\nabla f_i(\mathbf{x}) \nabla f_i(\mathbf{x})^T}{(f_i(\mathbf{x}))^2} - \sum_{i=1}^m \frac{\nabla^2 f_i(\mathbf{x})}{f_i(\mathbf{x})}. \end{aligned} \quad (8)$$



Rys. 1. Wykres funkcji $-(1/t) \log(-u)$ dla różnych wartości parametru t .

Idea rozwiązania zadania (1) polega na zastąpieniu go przybliżeniem

$$\underset{\mathbf{x} \in \mathbb{R}^n}{\text{minimize}} \quad f_0(\mathbf{x}) + \frac{1}{t} \varphi(\mathbf{x}) \quad (9)$$

gdzie $t > 0$ jest parametrem. Zakładamy, że punkt optymalny zadania (9) istnieje i jest jedyny, oznaczmy go \mathbf{x}_t^* , oraz że istnieje punkt startowy \mathbf{x}_0 , który spełnia

$$f_i(\mathbf{x}_0) < 0, \quad i \in \mathbf{m}. \quad (10)$$

Zadanie (9) jest równoważne zadaniu

$$\underset{\mathbf{x} \in \mathbb{R}^n}{\text{minimize}} \quad t f_0(\mathbf{x}) + \varphi(\mathbf{x}), \quad t > 0. \quad (11)$$

Rolą funkcji $\varphi(\mathbf{x})$ jest zapobieganie opuszczeniu przez rozwiązanie obszaru Ω , innymi słowy $\varphi(\mathbf{x})$ jest swego rodzaju barierą zbioru Ω , ponieważ $\varphi(\mathbf{x}) = +\infty$ dla $\mathbf{x} \notin \Omega$ oraz $\varphi(\mathbf{x})$ dąży do $+\infty$ kiedy \mathbf{x} zbliża się do brzegu zbioru Ω . (Mówiąc bardziej precyzyjnie, $\varphi(\mathbf{x}) = +\infty$ dla $\mathbf{x} \notin \text{relint } \Omega$ oraz $\varphi(\mathbf{x})$ dąży do $+\infty$ kiedy \mathbf{x} zbliża się do względnego brzegu zbioru Ω . Więcej

informacji na temat względnego wnętrza zbioru i względnego brzegu zbioru można znaleźć w [1]).

Minimalizator \mathbf{x}_t^* (tzn. rozwiązanie) zadania (11) jest ściśle dopuszczalny (*strictly feasible*) tzn.

$$f_i(\mathbf{x}_t^*) < 0, \quad i \in \mathbf{m}. \quad (12)$$

Przyjmijmy oznaczenie

$$\psi_t(\mathbf{x}) = tf_0(\mathbf{x}) + \varphi(\mathbf{x}). \quad (13)$$

Z warunków optymalności pierwszego rzędu mamy

$$\nabla \psi_t(\mathbf{x}_t^*) = \nabla_{\mathbf{x}} [tf_0(\mathbf{x}) + \varphi(\mathbf{x})]|_{\mathbf{x}=\mathbf{x}_t^*} = 0, \quad (14)$$

zatem

$$t\nabla f_0(\mathbf{x}_t^*) - \sum_{i=1}^m \frac{1}{f_i(\mathbf{x}_t^*)} \nabla f_i(\mathbf{x}_t^*) = 0, \quad (15)$$

skąd otrzymujemy

$$\nabla f_0(\mathbf{x}_t^*) + \sum_{i=1}^m \frac{1}{-tf_i(\mathbf{x}_t^*)} \nabla f_i(\mathbf{x}_t^*) = 0. \quad (16)$$

Zdefiniujmy

$$(\lambda_t^*)_i = \frac{1}{-tf_i(\mathbf{x}_t^*)} > 0, \quad i \in \mathbf{m}. \quad (17)$$

Zauważmy, że lagranżjan dla zadania (1) jest postaci

$$L(\mathbf{x}, \boldsymbol{\lambda}) = f_0(\mathbf{x}) + \sum_{i=1}^m \lambda_i f_i(\mathbf{x}). \quad (18)$$

Zauważmy, że dla

$$\lambda_i = (\lambda_t^*)_i = \frac{1}{-tf_i(\mathbf{x}_t^*)} > 0, \quad i \in \mathbf{m} \quad (19)$$

lagranżjan ten

$$L(\mathbf{x}, \boldsymbol{\lambda}_t^*) = f_0(\mathbf{x}) + \sum_{i=1}^m (\lambda_t^*)_i f_i(\mathbf{x}) \quad (20)$$

osiąga minimum w \mathbf{x}_t^* ponieważ

$$\begin{aligned} \nabla_{\mathbf{x}} L(\mathbf{x}, \boldsymbol{\lambda}_t^*)|_{\mathbf{x}=\mathbf{x}_t^*} &= \nabla L(\mathbf{x}_t^*, \boldsymbol{\lambda}_t^*) \\ &= \nabla f_0(\mathbf{x}_t^*) + \sum_{i=1}^m \frac{1}{-tf_i(\mathbf{x}_t^*)} \nabla f_i(\mathbf{x}_t^*) \\ &= \nabla f_0(\mathbf{x}_t^*) + \sum_{i=1}^m (\lambda_t^*)_i \nabla f_i(\mathbf{x}_t^*) \\ &= \mathbf{0}. \end{aligned} \quad (21)$$

Biorąc pod uwagę, że dualna funkcja Lagrange'a

$$g(\boldsymbol{\lambda}) = \min_{\mathbf{x}} L(\mathbf{x}, \boldsymbol{\lambda}) \quad \boldsymbol{\lambda} \geq 0 \quad (22)$$

jest dolnym ograniczeniem dla wartości optymalnej p^* zadania (1), tzn.

$$g(\boldsymbol{\lambda}) \leq p^*, \quad (23)$$

otrzymujemy dla $\boldsymbol{\lambda} = \boldsymbol{\lambda}_t^*$

$$\begin{aligned} p^* &\geq g(\boldsymbol{\lambda}_t^*) \\ &= L(\mathbf{x}_t^*, \boldsymbol{\lambda}_t^*) \\ &= f_0(\mathbf{x}_t^*) + \sum_{i=1}^m (\lambda_t^*)_i f_i(\mathbf{x}_t^*) \\ &= f_0(\mathbf{x}_t^*) + \sum_{i=1}^m \frac{1}{-tf_i(\mathbf{x}_t^*)} f_i(\mathbf{x}_t^*) \\ &= f_0(\mathbf{x}_t^*) - \frac{m}{t}, \end{aligned} \quad (24)$$

czyli

$$f_0(\mathbf{x}_t^*) - \frac{m}{t} \leq p^*. \quad (25)$$

Nierówność (25) stanowi podstawowe uzasadnienie metody funkcji bariery, ponieważ oznacza, że rozwiązanie $\mathbf{x}^*(t)$ zadania bez ograniczeń (11) jest ϵ -suboptymalnym rozwiązaniem zadania z ograniczeniami (1), tzn. dla danego $\epsilon > 0$ zachodzi

$$f_0(\mathbf{x}_t^*) - p^* \leq \epsilon \quad \text{jeśli} \quad \frac{m}{t} \leq \epsilon, \quad (26)$$

skąd otrzymujemy

$$f_0(\mathbf{x}_t^*) \xrightarrow{t \rightarrow \infty} p^*. \quad (27)$$

1.4 Sekwencyjna metoda bariery

Mogłoby się wydawać, że najprościej przyjąć

$$t \geq \frac{m}{\epsilon} \quad (28)$$

i rozwiązać (11) używając np. metody Newtona. W praktyce takie podejście może okazać się mało skuteczne, nie tylko dla tego, że punkt startowy \mathbf{x}_0 może być położony daleko od punktu optymalnego \mathbf{x}^* , ale przede wszystkim dlatego, że funkcja (13) jest na ogół źle uwarunkowana dla dużych t (tzn. hesjan funkcji ψ_t zmienia się szybko w pobliżu brzegu zbioru Ω), co powoduje, że metoda Newtona może potrzebować bardzo wielu iteracji aby zbiec do \mathbf{x}_t^* . W związku z tym, w praktyce, rozwiązuje się sekwencję zadań (11) zaczynając od niedużych wartości t i zwiększając je stopniowo, aż do spełnienia warunku

$$\frac{m}{t} \leq \epsilon. \quad (29)$$

Postępowanie takie można przedstawić w postaci następującego algorytmu.

Sekwencyjna metoda bariery (<i>sequential barrier method</i>)	
dane: ściśle dopuszczalny \mathbf{x}_0 , t_{init} , $\gamma > 1$, $\epsilon > 0$	
1. Podstawiamy $k = 0$, $t = t_{\text{init}}$.	
2. Używając metody Newtona z tłumieniem rozwiązujemy zadanie	
minimize $tf_0(\mathbf{x}) + \varphi(\mathbf{x})$	
dla punktu startowego \mathbf{x}_k , oznaczamy rozwiązanie optymalne tego zadania przez \mathbf{x}_k^* .	
3. Aktualizujemy $\mathbf{x}_{k+1} \leftarrow \mathbf{x}_k^*$.	
4. Jeśli $m/t \leq \epsilon$, zwracamy \mathbf{x}_k^* i kończymy algorytm.	
5. Aktualizujemy $t \leftarrow \gamma t$, $k \leftarrow k + 1$ i przechodzimy punktu 2.	

Każdą iterację k w powyższym algorytmie nazywamy krokiem centrującym (*centering step*) lub iteracją zewnętrzną (*outer iteration*), punkt \mathbf{x}_k^* nazywamy k -tym punktem centralnym (*central point*), zaś krzywą, którą tworzą minima \mathbf{x}_k^* funkcji (13) dla kolejnych wartości k , nazywamy ścieżką centralną (*central path*), zawiera się ona w zbiorze dopuszczalnym Ω . Z tego powodu metoda sekwencyjna metoda bariery należy do tzw. metod punktu wewnętrznego (*interior-point methods*). Każdy krok centrujący wymaga pewnej liczby iteracji wewnętrznych, które są iteracjami algorytmu Newtona, potrzebnymi do wyznaczenia \mathbf{x}_k^* z zadaną dokładnością.

1.4.1 Liczba iteracji metody SBM

Jak już wspomniano, podstawienie $t_{\text{init}} \geq m/\epsilon$ zagwarantowałoby zbieżność omówionej metody w jednym kroku centrującym (iteracji zewnętrznej), jednak mogłoby znacznie zwiększyć liczbę iteracji wewnętrznych, dlatego zwiększa się stopniowo wartość t podstawiając $t_{k+1} = \mu t_k$ gdzie t_k oznacza wartość t w k -tym kroku centrującym. Liczba iteracji wewnętrznych (kroków centrujących) potrzebnych do znalezienia rozwiązania z zadaną dokładnością ϵ wynosi

$$\left\lceil \frac{\log(m\epsilon^{-1}/t_{\text{init}})}{\log \mu} + 1 \right\rceil \quad (30)$$

gdzie $\lceil x \rceil$ oznacza funkcję sufit, tzn. najmniejszą liczbę całkowitą nie mniejszą od x .

1.5 Wyznaczanie ściśle dopuszczalnego punktu początkowego

Metoda bariery wymaga znajomości ściśle dopuszczalnego (*strictly feasible*) punktu startowego \mathbf{x}_0 , który można wyznaczyć rozwiązując pomocnicze zadanie optymalizacji, nazywanym często zadaniem pierwszej fazy (*phase I problem*). Zadanie

$$\begin{aligned} & \underset{\mathbf{x} \in \mathbb{R}^n}{\text{minimize}} && f_0(\mathbf{x}) \\ & \text{subject to} && f_i(\mathbf{x}) \leq 0, \quad i \in \mathbf{m} \end{aligned} \quad (31)$$

można napisać w równoważnej postaci

$$\begin{aligned} & \underset{\mathbf{x} \in \mathbb{R}^n, s \in \mathbb{R}}{\text{minimize}} && s \\ & \text{subject to} && f_i(\mathbf{x}) \leq s, \quad i \in \mathbf{m}, \end{aligned} \quad (32)$$

dla której zawsze możemy łatwo znaleźć ściśle dopuszczalny punkt startowy, wybierając dowolny punkt $\tilde{\mathbf{x}}_0$ (np. $\tilde{\mathbf{x}}_0 = 0$) a następnie dobierając wartość s_0 tak, że

$$s_0 > \max_{i=1, \dots, m} f_i(\tilde{\mathbf{x}}_0), \quad (33)$$

np. kładąc

$$s_0 = 1 + \max_{i=1, \dots, m} f_i(\tilde{\mathbf{x}}_0). \quad (34)$$

Rozwiązując zadanie pomocnicze, otrzymujemy punkt optymalny $(\tilde{\mathbf{x}}^*, s^*)$, wówczas możemy wyróżnić trzy przypadki.

1. Jeśli $s^* < 0$ to znaczy, że $f_i(\tilde{\mathbf{x}}^*) \leq s^* < 0$, zatem $\tilde{\mathbf{x}}^*$ jest punktem ściśle dopuszczalnym dla zadania (1)
2. Jeśli $s^* = 0$ to znaczy, że zadanie wyjściowe nie ma ściśle dopuszczalnego rozwiązania.
3. Jeśli $s^* > 0$ to znaczy, że zadanie wyjściowe nie ma rozwiązania (*infeasible*)

W praktyce jeśli $s^* < -\epsilon$ dla pewnego „rozsądnego” $\epsilon > 0$, to $\tilde{\mathbf{x}}^*$ jest ściśle dopuszczalny (*strictly feasible*).

2 Rozwiązywanie zadań LP metodą SBM

2.1 Logarytmiczna funkcja bariery dla zadania LP

Dla zadania LP

$$\begin{aligned} & \underset{\mathbf{x} \in \mathbb{R}^n}{\text{minimize}} && \mathbf{c}^T \mathbf{x} \\ & \text{subject to} && \mathbf{a}_i^T \mathbf{x} \leq b_i, \quad i \in \mathbf{m} \end{aligned} \quad (35)$$

mamy

$$\varphi(\mathbf{x}) = - \sum_{i=1}^m \log(b_i - \mathbf{a}_i^T \mathbf{x}), \quad (36)$$

$$\nabla \varphi(\mathbf{x}) = \sum_{i=1}^m \frac{\mathbf{a}_i}{b_i - \mathbf{a}_i^T \mathbf{x}}, \quad (37)$$

$$\nabla^2 \varphi(\mathbf{x}) = \sum_{i=1}^m \frac{\mathbf{a}_i \mathbf{a}_i^T}{(b_i - \mathbf{a}_i^T \mathbf{x})^2}. \quad (38)$$

2.2 Nierówności liniowe i wielościany

Zbiór punktów $\mathbf{x} \in \mathbb{R}^n$ spełniających liniową nierówność $\mathbf{a}^T \mathbf{x} \leq b$ dla $b \in \mathbb{R}$ jest domkniętą półprzestrzenią (ang. *closed half-space*). Wektor \mathbf{a} jest normalny do brzegu tej półprzestrzeni i wskazuje na zewnątrz. Układ m liniowych nierówności

$$\mathbf{a}_i^T \mathbf{x} \leq b_i, \quad i \in \mathbf{m}, \quad (39)$$

wyznacza obszar w \mathbb{R}^m , który jest przecięciem m półprzestrzeni i który nazywamy wielościanem (ang. *polyhedron*). Wielościan jest zbiorem wypukłym (jako przecięcie zbiorów wypukłych). W zależności od układu nierówności

$$\mathbf{a}_i^T \mathbf{x} \leq b_i, \quad i \in \mathbf{m}, \quad (40)$$

wielościan może być ograniczony lub nieograniczony. Wielościan ograniczony nazywamy wielokomórką (ang. *polytope*). Przyjmując oznaczenia

$$\mathbf{A} = \begin{bmatrix} \mathbf{a}_1^T \\ \mathbf{a}_2^T \\ \vdots \\ \mathbf{a}_m^T \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix}, \quad (41)$$

możemy napisać układ nierówności

$$\mathbf{a}_i^T \mathbf{x} \leq b_i, \quad i \in \mathbf{m}, \quad (42)$$

w postaci macierzowo-wektorowej

$$\mathbf{A} \mathbf{x} \leq \mathbf{b}, \quad (43)$$

gdzie nierówność jest rozumiana jako nierówność odpowiednich elementów (ang. *component-wise inequality*), tzn. dla

$$\mathbf{v} = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_m \end{bmatrix}, \quad \mathbf{w} = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_m \end{bmatrix}, \quad (44)$$

napis

$$\mathbf{v} \leq \mathbf{w} \quad (45)$$

oznacza, że $v_i \leq w_i, i \in \mathbf{m}$.

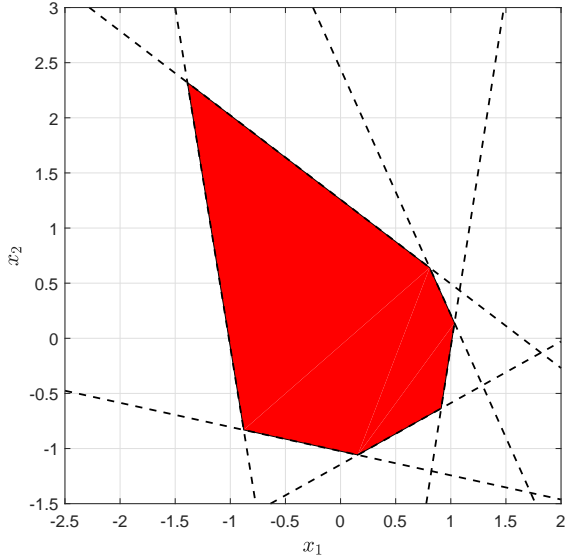
Przykład 1. Weźmy pod uwagę układ nierówności

$$\mathbf{A}\mathbf{x} \leq \mathbf{b}, \quad (46)$$

gdzie

$$\mathbf{A} = \begin{bmatrix} 0.4873 & -0.8732 \\ 0.6072 & 0.7946 \\ 0.9880 & -0.1546 \\ -0.2142 & -0.9768 \\ -0.9871 & -0.1601 \\ 0.9124 & 0.4093 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}. \quad (47)$$

Mamy więc 6 nierówności liniowych, z których każda definiuje pewną półprzestrzeń, natomiast ich układ definiuje wielościan (Rys. 2.). Wielokomórkę reprezentowaną przez przecięcie półprzestrzeni nazywamy \mathcal{H} -wielokomórką (ang. \mathcal{H} -polytope).



Rys. 2 Wielokomórka opisana zależnościami (46) i (47).

Dowolna wielokomórka może być przedstawiona jako powłoka wypukła swoich wierzchołków (ang. *vertices*), wówczas nazywamy ją \mathcal{V} -wielokomórką (ang. \mathcal{V} -polytope). Rozpatrywaną wielokomórkę opisaną zależnościami (46) i (47) możemy równoważnie przedstawić jako powłokę wypukłą kolumny macierzy \mathbf{V}

$$\mathbf{V} = \begin{bmatrix} 0.1562 & 0.9127 & 0.8086 & 1.0338 & -1.3895 & -0.8782 \\ -1.0580 & -0.6358 & 0.6406 & 0.1386 & 2.3203 & -0.8311 \end{bmatrix}.$$

3 Zadania

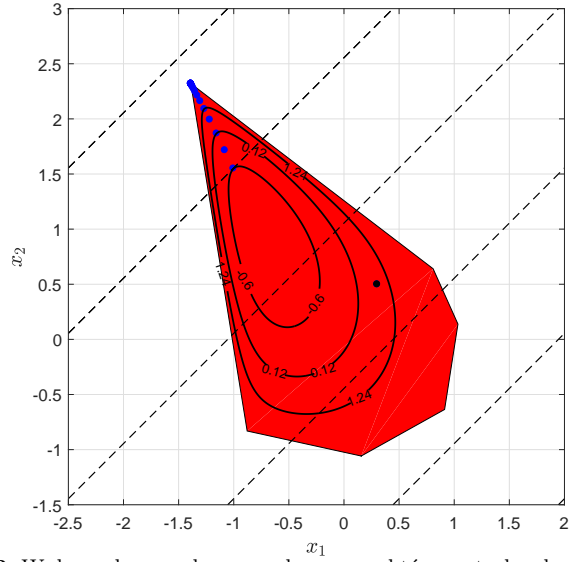
Zadanie 1 Napisać w środowisku Matlab skrypt do rozwiązywania zadania LP

$$\begin{aligned} & \underset{\mathbf{x} \in \mathbb{R}^n}{\text{minimize}} && \mathbf{c}^T \mathbf{x} \\ & \text{subject to} && \mathbf{A}\mathbf{x} \leq \mathbf{b} \end{aligned}$$

metodą SBM przy założeniu, że znany jest ściśle dopuszczalny punkt startowy (przyjąć $\mathbf{x}_0 = 0$), gdzie

$$\mathbf{c} = \begin{bmatrix} 1 \\ -1 \end{bmatrix},$$

zaś \mathbf{A}, \mathbf{b} są dane przez (47). Wykonać odpowiednie wykresy (jak na Rys. 2 i 3, można skorzystać z polecenia `fill`). Porównać otrzymany wynik z rozwiązaniem znalezionym za pomocą funkcji `linprog`.



Rys. 3. Wykres obszaru dopuszczalnego, punktów centralnych oraz poziomicy funkcji bariery dla Zadania 1.

Wskazówka 1: Dla zadania LP metoda SBM ma postać

Sekwencyjna metoda bariery (dla zadania LP)

dane: ściśle dopuszczalny \mathbf{x}_0 , t_{init} , $\gamma > 1$, $\epsilon > 0$

1. Podstawiamy $k = 0$, $t = t_{\text{init}}$.
2. Używając metody Newtona z tłumieniem rozwiązujemy zadanie

$$\underset{\mathbf{x}}{\text{minimize}} \quad \psi_t(\mathbf{x}) \quad (48)$$

dla punktu startowego \mathbf{x}_k , gdzie

$$\begin{aligned} \psi_t(\mathbf{x}) &= t f_0(\mathbf{x}) + \varphi(\mathbf{x}) \\ &= t \mathbf{c}^T \mathbf{x} - \sum_{i=1}^m \log(b_i - \mathbf{a}_i^T \mathbf{x}), \end{aligned}$$

$$\begin{aligned} \nabla \psi_t(\mathbf{x}) &= t \nabla f_0(\mathbf{x}) + \nabla \varphi(\mathbf{x}) \\ &= t \mathbf{c} + \sum_{i=1}^m \frac{\mathbf{a}_i}{b_i - \mathbf{a}_i^T \mathbf{x}}, \end{aligned}$$

$$\begin{aligned} \nabla^2 \psi_t(\mathbf{x}) &= t \nabla^2 f_0(\mathbf{x}) + \nabla^2 \varphi(\mathbf{x}) \\ &= \sum_{i=1}^m \frac{\mathbf{a}_i \mathbf{a}_i^T}{(b_i - \mathbf{a}_i^T \mathbf{x})^2}. \end{aligned}$$

Rozwiązanie optymalne zadania (48) oznaczamy przez \mathbf{x}_k^* .

3. Aktualizujemy $\mathbf{x}_{k+1} \leftarrow \mathbf{x}_k^*$.
4. Jeśli $m/t \leq \epsilon$, zwracamy \mathbf{x}_k^* i kończymy algorytm.
5. Aktualizujemy $t \leftarrow \gamma t$, $k \leftarrow k + 1$ i przechodzimy punktu 2.

Wskazówka 2: Przyjmując oznaczenia

$$\mathbf{A} = \begin{bmatrix} \mathbf{a}_1^T \\ \mathbf{a}_2^T \\ \vdots \\ \mathbf{a}_m^T \end{bmatrix} \in \mathbb{R}^{m \times n}, \quad \mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix}, \quad \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}, \quad (49)$$

możemy wyrażenia

$$\sum_{i=1}^m \log(b_i - \mathbf{a}_i^T \mathbf{x}), \quad \sum_{i=1}^m \frac{\mathbf{a}_i}{b_i - \mathbf{a}_i^T \mathbf{x}}, \quad \sum_{i=1}^m \frac{\mathbf{a}_i \mathbf{a}_i^T}{(b_i - \mathbf{a}_i^T \mathbf{x})^2} \quad (50)$$

zapisać w środowisku Matlab następująco

```
sum(log(b-A*x))
```

```
(A')*(1./(b-A*x))
```

```
((diag(1./(b-A*x))*A)')*(diag(1./(b-A*x))*A)
```

Zadanie 2 Rozbudować skrypt z Zadania 1 w taki sposób, aby ściśle dopuszczalny punkt startowy był wyznaczany automatycznie (metodą opisaną w podrozdziale 1.5).

Literatura

- [1] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, New York, NY, USA, 2004.