

Metody numeryczne

Janusz Miller

Katedra Informatyki Stosowanej
AGH University of Science and Technology

16.11.2021

- ① Kincaid D., Cheney W.: Analiza numeryczna, WNT 2006
- ② Nocedal J., Wright S.J.: Numerical Optimization, Springer 1999 (lub późniejsze, np. 2006)

Część VII

Przegląd metod optymalizacji statycznej

- ① Klasyfikacja problemów optymalizacji
- ② Optymalizacja bez ograniczeń
 - Metody bezgradientowe
 - Algorytmy spadku
 - Algorytm aproksymacji kwadratowej – Newtona
 - Metody quasi-newtonowskie
 - Przykład metody obszaru wiarygodności
- ③ Zarys optymalizacji z ograniczeniami

Kryteria podziałów problemów optymalizacji



Klasyfikacja¹:

- Continuous or **Discrete** Optimization
- Unconstrained or Constrained Optimization
- None, One or Many **Objectives**
- Deterministic or **Stochastic** Optimization

W dalszej części skupimy się na podkreślonych.

¹wg <https://neos-guide.org/optimization-tree>

Dana jest funkcja² $f : \mathbb{R}^n \longrightarrow \mathbb{R}$, $x \longmapsto f(x)$.

Należy znaleźć

$$x^* = \arg \min_x F(x)$$

$x \in \mathbb{R}^n$ – gdy zadanie **bez ograniczeń**,

$x \in D \subset \mathbb{R}^n$ – gdy zadanie **z ograniczeniami**.

*Większa część wykładu odnosić się będzie do przypadku **bez ograniczeń**.*

Na zakończenie wrócimy krótko do problemu z ograniczeniami.

²Znany jest algorytm obliczania wartości funkcji – nie zawsze znany „wzór”.

Gradient:

$$\nabla f = \begin{bmatrix} \frac{\partial f}{\partial x_1} \\ \frac{\partial f}{\partial x_2} \\ \vdots \\ \frac{\partial f}{\partial x_n} \end{bmatrix}$$

Hesjan:

$$\nabla^2 f = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 x_2} & \cdots & \frac{\partial^2 f}{\partial x_1 x_n} \\ \frac{\partial^2 f}{\partial x_2 x_1} & \frac{\partial^2 f}{\partial x_2^2} & \cdots & \frac{\partial^2 f}{\partial x_2 x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n x_1} & \frac{\partial^2 f}{\partial x_n x_2} & \cdots & \frac{\partial^2 f}{\partial x_n^2} \end{bmatrix}$$

Warunki wystarczające optymalności



Dla problemu bez ograniczeń i przy założeniu, że funkcja jest dwukrotnie różniczkowalna:

Jeżeli

- $\nabla f(x^*) = 0$
- $\nabla^2 f(x^*) > 0$ – hesjan dodatnio określony,

to w x^* funkcja ma minimum.

3 kategorie problemów

i algorytmy najczęściej stosowane dla ich rozwiązania



Wypukły *convex* – algorytmy spadku: największego (najbardziej stromego) spadku, gradientu sprzężonego, Newtona, quasi-Newtona, obszaru wiarygodności.

Wielomodalny *multimodal* – algorytmy ewolucyjne: genetyczne, strategii ewolucji, rojowe, mrówkowe, symulowanego wyżarzania.

„Zaszumiony” *noisy* – simpleks Nelder-Meada, wielokierunkowego poszukiwania Torczon.

Algorytmy spadku

Zastosowanie w problemach:



Szukanie minimum:

- globalnego w problemie wypukłym,
- lokalnego w problemie wielomodalnym.

Algorytmy spadku – ogólnie

Metody bezpośrednie vs. pośrednie



AGH

www.agh.edu.pl

Metody bezpośrednie – Obliczanie (i porównywanie) **wartości funkcji krok po kroku** w kierunku minimum

Metody pośrednie – Szukanie x takiego, w którym **warunki wystarczające** optymalności są spełnione.

- Wymaga **obliczania pochodnych funkcji**.
- Problem sprowadza się znalezienia rozwiązywania **układu równań** – jeżeli liniowych, to można uniknąć iterowania.

*W dalszej części będzie mowa o metodach **bezpośrednich***

Poszukiwanie minimum funkcji jednej zmiennej (ogólniej – minimum na zadanym kierunku)

Line search methods

Jeden krok przykładowego algorytmu:

- Obliczanie wartości funkcji w kolejnych punktach, aż do napotkania wzrostu wartości funkcji.
- Interpolacja wielomianem Lagrange'a 3 ostatnich punktów.
- Wyznaczenie przybliżenia minimum w wierzchołku paraboli.

Skrajna wersja: Po dwukrotnym obliczeniu wartości funkcji na kierunku (plus wartość w punkcie startowym) i wyznaczeniu przybliżenia minimum – przejście do szukania „kierunku poprawy”

Zastosowanie *line search methods* do szukania minimum funkcji wielu zmiennych



Problemy:

- 1 Długość „kroku” w poszukiwaniu w kierunku.
- 2 Z jaką dokładnością wyznaczać minimum na jednym kierunku.

Ad 1. W zaawansowanych algorytmach, jako kryterium wyboru długości „kroku” stosuje się warunek Wolfe'a: Redukcja wartości funkcji powinna być proporcjonalna do długości kroku oraz do pochodnej kierunkowej: nierówność Armijo:

$$f(x_k + \alpha_k p_k) \leq f(x_k) + c \alpha_k \nabla f_k^T p_k, \quad c \approx 10^{-4}$$

Ad 2. Należy brać pod uwagę koszt zmiany kierunku (od zerowego, poprzez obliczanie gradientu, aż do hesjanu).

Klasyfikacja metod bez ograniczeń



- Bezgradientowe:
 - spadku względem współrzędnych (Hooke'a-Jeevesa),
 - kierunków sprzężonych (Powella, Rosenbrocka),
 - simpleks Nelder-Mead,
 - minimalizacji wzdłuż współrzędnych (Gausa-Seidela).
- Gradientowe pierwszego rzędu:
 - największego spadku,
 - gradientów sprzężonych.
- Gradientowe drugiego rzędu i „superliniowe”
 - Newtona, BFGS,
 - „Trust region”.

Przykład metody spadku względem współrzędnych – *coordinate descent methods*



Metoda Hooke'a-Jeevesa

Schemat algorytmu:

- Wybór ortogonalnej (niezmiennej) bazy kierunków.
- Start z punktu startowego.
- Każdy krok składa się z 2 etapów:
 - + Próbny – obliczenie wartości funkcji w otoczeniu aktualnego punktu – w każdym kierunku bazy w takiej samej odległości.
 - + Roboczy – przesunięcie bieżącego punktu o wektor będący sumą wektorów poprawy.
- W przypadku braku poprawy – skrócenie kroku albo stop.

Przykłady metod kierunków sprzężonych



AGH

www.agh.edu.pl

Nie należy mylić z metodami gradientów sprzężonych.
Jak Hooke'a- Jeevesa, ale z modyfikacją bazy kierunków.

① Schemat algorytmu **Rosenbrocka**:

- Kolejne kroki pomyślne – jak w metodzie H-J.
- W przypadku braku poprawy — ortogonalny obrót bazy – nowa baza jest tworzona z sumy wektorów przesunięć punktów bieżących.

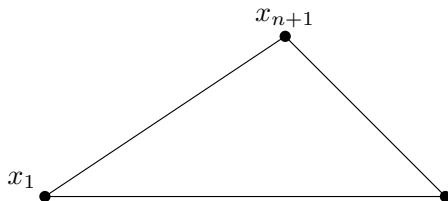
② Schemat algorytmu **Powella**:

- Wybór bazy początkowej – jw.
- Szukanie minimum w każdym kierunku.
- Suma wektorów przesunięć jest nowym kierunkiem – dodawanym do bazy. Równocześnie z bazy jest usuwany kierunek najbardziej zbliżony do dodanego.

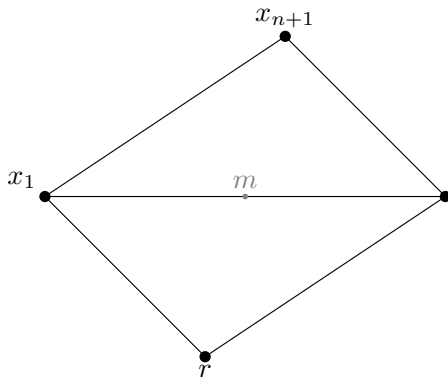
Metoda simpleksów Neldera-Meada

- ① W $\mathbb{R}^n$ tworzymy wokół punktu $\mathbf{x}_0$ simpleks $n+1$ wym.
- ② Porządkujemy punkty simpleksu, aby $f(x_i) < f(x_{i+1})$.
- ③ Generujemy punkt $r = 2m - x_{n+1}$, gdzie
$$m = \frac{1}{n} \sum_{j=1}^n x_j.$$
- ④ Jeżeli $f(x_1) \leq f(r) < f(x_n)$, to akceptujemy r , **reflect**.
- ⑤ Jeżeli $f(r) \leq f(x_1)$, to obliczamy $s = m + 2(m - x_{n+1})$,
 - a jeżeli $f(s) < f(r)$, to akceptujemy s , **expand**.
 - b jeżeli nie, to akceptujemy r , **reflect**.
- ⑥ Jeżeli $f(r) \geq f(x_n)$, to kontrakcja między m a lepszym $\{r, x_{n+1}\}$,
 - a jeżeli $f(r) < f(x_{n+1})$, to obliczamy $c = m + (r - m)/2$,
 - jeżeli $f(c) < f(r)$, to akceptujemy c , **contract outside**,
 - jeżeli nie, to do punktu 7,
 - b jeżeli $f(r) \geq f(x_{n+1})$, to obliczamy $cc = m + (x_{n+1} - m)/2$,
 - jeżeli $f(cc) < f(x_{n+1})$, to akceptujemy cc , **contract inside**.

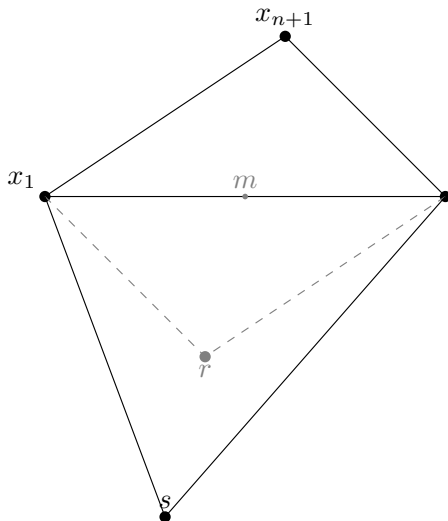
Metoda simpleksów Neldera-Meada



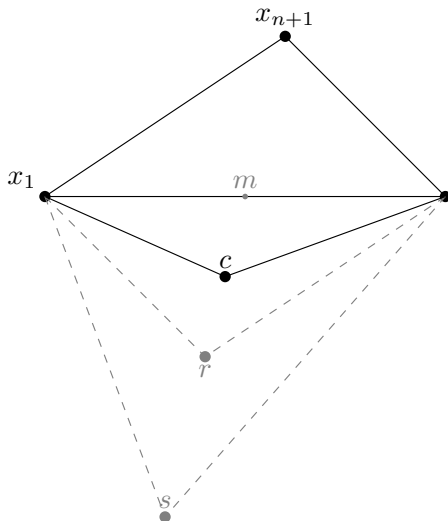
Metoda simpleksów Neldera-Meada



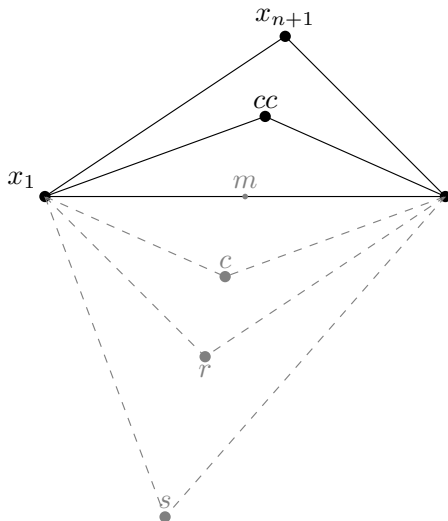
Metoda simpleksów Neldera-Meada



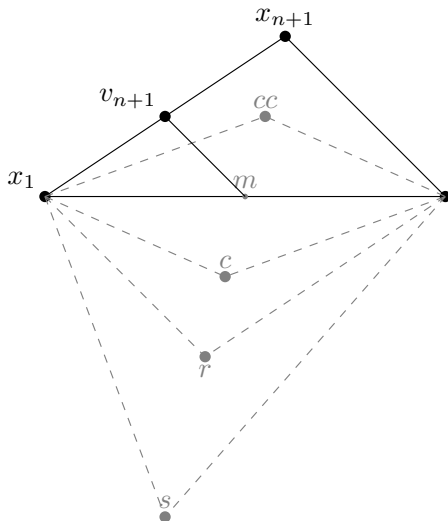
Metoda simpleksów Neldera-Meada



Metoda simpleksów Neldera-Meada



Metoda simpleksów Neldera-Meada



Metoda minimalizacji wzdłuż współrzędnych (Gauss-Seidel, *coordinate descent method*)



Idea:

Szukanie minimum na kierunkach kolejnych współrzędnych.

Grupa:

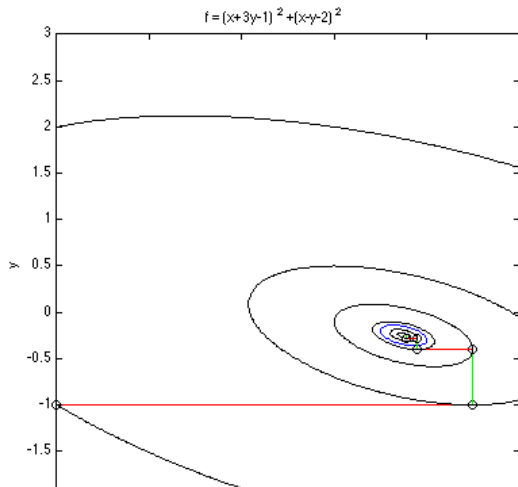
Metody bezgradientowe

Metoda minimalizacji wzdłuż współrzędnych (Gauss-Seidel, *coordinate descent method*)

Przykładowa funkcja:

$$f(x_1, x_2) = (x_1 + 3x_2 - 1)^2 + (x_1 - x_2 - 2)^2.$$

$$x_{\min} = \begin{bmatrix} \frac{7}{4} \\ -\frac{1}{4} \end{bmatrix}$$

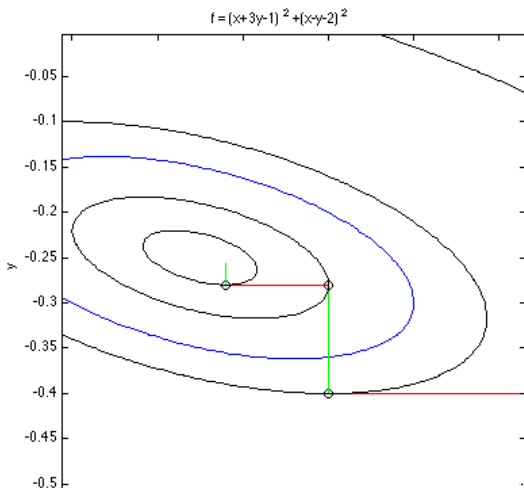


Metoda minimalizacji wzdłuż współrzędnych (Gauss-Seidel, *coordinate descent method*)

Przykładowa funkcja:

$$f(x_1, x_2) = (x_1 + 3x_2 - 1)^2 + (x_1 - x_2 - 2)^2.$$

$$x_{\min} = \begin{bmatrix} \frac{7}{4} \\ -\frac{1}{4} \end{bmatrix}$$



Metoda najszybszego spadku

– *steepest descent*



Idea:

Szukanie minimum na kierunkach antygradientów
wyznaczanych w kolejnych krokach.

$$p_k = -\nabla f(x_k)$$

Grupa:

Metody gradientowe, pierwszego rzędu

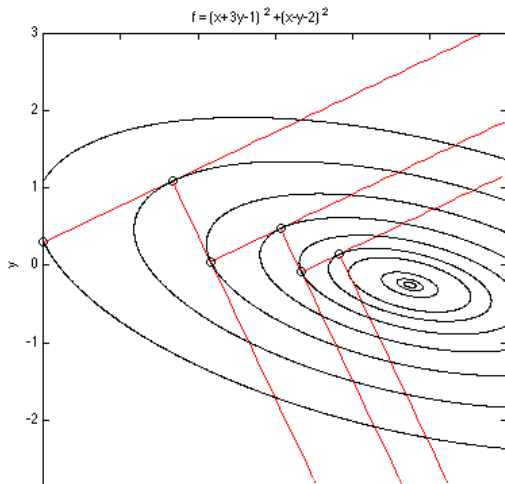
Metoda najszybszego spadku – *steepest descent*



Przykładowa funkcja:

$$f(x_1, x_2) = (x_1 + 3x_2 - 1)^2 + (x_1 - x_2 - 2)^2.$$

$$x_{\min} = \begin{bmatrix} \frac{7}{4} \\ -\frac{1}{4} \end{bmatrix}$$



Metoda gradientów sprzężonych

– *Conjugate gradient (CG) method*



Przesłanki:

- W przypadku funkcji kwadratowej $f(x) = \frac{1}{2}x^T A x + b^T x + c$ warstwy mają kształt elips. Metoda najszybszego spadku daje rozwiązanie w n krokach jeżeli osie elips (warstw funkcji) są równoległe do osi współrzędnych.
- Osie elips są równoległe do osi współrzędnych gdy macierz A jest diagonalna.
- Dla przekształcenia problemu do zadania z macierzą diagonalną potrzebna jest znajomość macierzy kierunków sprzężonych

$$S = [p_0, p_1, \dots, p_{n-1}] : p_i^T A p_j = 0, \quad i \neq j.$$

Wniosek: Jeżeli poszukiwania minimów będą prowadzone wzdłuż kierunków sprzężonych p_i , to liczba kroków $= n$.

Metoda gradientów sprzężonych

– *Conjugate gradient (CG) method*



Idea:

Kolejny kierunek minimalizacji jest sprzężony do wszystkich poprzednich i jest generowany tylko na podstawie bieżącego punktu i poprzedniego kierunku

$$p_k = -r_k + \beta_k p_{k-1}, \quad \beta_k = \frac{r_k^T A p_{k-1}}{p_{k-1}^T A p_{k-1}}, \quad r_k = \nabla f(x_k) = Ax_k - b.$$

Grupa:

Metody gradientowe, pierwszego rzędu.

Grupa: Metoda gradientowa **drugiego rzędu**.

Cechy:

- Szybka — jeżeli miarą jest tylko liczba kroków (z pominięciem kosztu pojedynczego kroku).
- Duży koszt pojedynczego kroku — wymaga częstego obliczania hesjanu.
- Kiedy skuteczna:
 - w pobliżu minimum (ogólnie — gdy model 2. rzędu dobrze przybliża funkcję),
 - gdy hesjan jest macierzą dodatnio określoną.

Metoda Newtona



Przykładowa funkcja:

$$f(x_1, x_2) = (x_1 + 3x_2 - 1)^2 + (x_1 - x_2 - 2)^2.$$

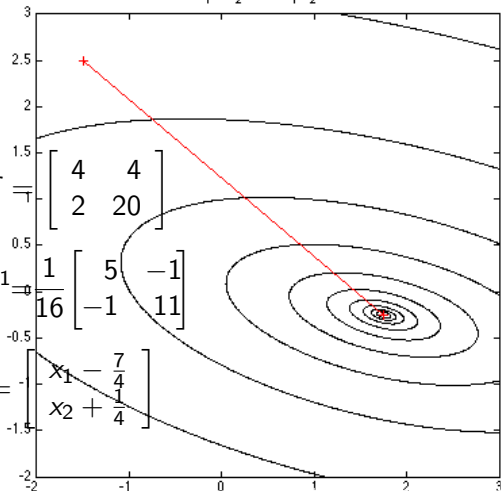
$$f = [x_1 + 3x_2 - 1]^2 + [x_1 - x_2 - 2]^2$$

$$\nabla f = \begin{bmatrix} 4x_1 + 4x_2 - 6 \\ 4x_1 + 20x_2 - 2 \end{bmatrix}$$

$$\nabla^2 f = \begin{bmatrix} 4 & 4 \\ 2 & 20 \end{bmatrix}$$

$$\nabla^2 f^{-1} = \frac{1}{16} \begin{bmatrix} 5 & -1 \\ -1 & 11 \end{bmatrix}$$

$$p = \begin{bmatrix} x_1 - \frac{7}{4} \\ x_2 + \frac{1}{4} \end{bmatrix}$$



Metody quasi-newtonowskie



Cechy:

- Zbieżność superliniowa (rzęd ≥ 1).
- Mniejszy koszt - nie wymagają obliczania hesjanu.
- Hesjan zastąpiony macierzą B_k - uaktualnianą informacją o funkcji uzyskaną w kolejnych krokach. Zmiana gradientu wzdłuż kierunku poszukiwań dostarcza przybliżoną informację o drugiej pochodnej.

Najbardziej popularne:

- BFGS (Broyden, Fletcher, Goldfarb, Shanno),
- DFP (Davidon, Fletcher, Powell).

Metody obszaru wiarygodności

Trust-region methods



Idea::

- Budujemy model otoczenia punktu x_k

$$m_k(p) = \frac{1}{2} p^T B_k p + \nabla f_k^T p + f_k$$

B_k – macierz – jak w metodzie Newtona, CG, BFGS itp..

- Szukamy minimum modelu.

$$\min_{p \in \mathbb{R}^n} m_k(p).$$

- Długość i kierunek kroku zależy od promienia obszaru wiarygodności Δ_k .
- Promień Δ_k ustalany na podstawie zgodności modelu m i funkcji celu f w poprzednim kroku.

Przykład metody obszaru wiarygodności

Dogleg method

1. Wyznaczenie promienia obszaru wiarygodności (zaufania, ufności)



$$\rho_k = \frac{f(x_k) - f(x_k + p_k)}{m_k(0) - m_k(p_k)} = \frac{\text{redukcja faktyczna}}{\text{redukcja przewidywana}}$$

- Jeżeli $\rho_k < 0.25$, to $\Delta_{k+1} = 0.25\|p_k\|$.
- Jeżeli $0.25 \leq \rho_k \leq 0.75$, to $\Delta_{k+1} = \Delta_k$.
- Jeżeli $\rho_k > 0.75$, to $\Delta_{k+1} = \min(2\Delta_k, \Delta_{\max})$.

Przykład metody obszaru wiarygodności

Dogleg method

2. Wyznaczamy punkt Cauchy'ego



- Każdy punkt spadku p^S , (w tym także punkt Cauchy'ego) leży na kierunku antygradientu **modelu**,
$$p_k^S = \tau \frac{\nabla f_k}{\|\nabla f_k\|}$$
- Odległość τ^{min} do minimum w tym kierunku oblicza się z równania kwadratowego na τ

$$m(\tau p_k^S) = \frac{1}{2}(\tau p_k^S)^T B_k (\tau p_k^S) + \nabla f_k^T (\tau p_k^S) + f_k$$

- Punkt Cauchy'ego leży w **obszarze wiarygodności**,
a więc $\tau \leq \Delta_k$,
- Dodatkową informację o wiarygodności niesie wypukłość modelu (gdy B_k dodatnio określona),
- Ostatecznie ustalamy punkt Cauchy'ego

$$p_k^C = -\tau_k \frac{\Delta_k}{\|\nabla f_k\|} \nabla f_k, \quad \tau_k = \begin{cases} 1, & (\nabla f_k)^T B_k (\nabla f_k) \leq 0 \\ \dots \end{cases}$$

Przykład metody obszaru wiarygodności

Dogleg method

3. Wyznaczamy kierunek do minimum modelu m (2. rzędu)



AGH

Przy wyznaczaniu „lepszego” kierunku korzystamy z hesjanu albo – w przypadku metody quasi-Newtona – z macierzy B_k

$$p_k^B = -B_k^{-1} \nabla f_k.$$

Przykład metody obszaru wiarygodności

Dogleg method

4. Wyznaczenie ostatecznego kierunku i długości kroku



- Jeżeli minimum modelu leży wewnątrz obszaru wiarygodności, to wykonujemy krok do tego minimum.
- Jeżeli na zewnątrz obszaru, to krok wg ***dogleg path***:
 - jeżeli Δ_k jest małe w stosunku do odległości do minimum modelu τ^{min} , to krok wg gradientu (bo mała wiarygodność modelu każe nie wierzyć w prawdziwość jego minimum)

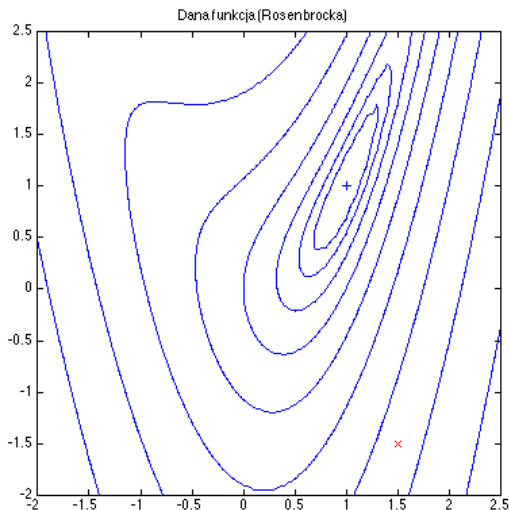
$$p_k = \tau^{min} p_k^C,$$

- w przeciwnym przypadku – konstruujemy ścieżkę zbliżoną do dwóch wektorów p^C i p^B :

$$p_k = p_k^C + (\tau^{min} - 1)(p_k^B - p_k^C), \quad 1 \leq \tau^{min} \leq 2.$$

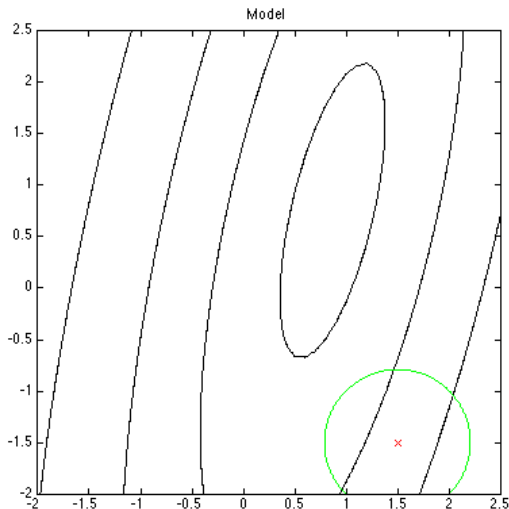
Przykład metody obszaru wiarygodności

Dogleg method



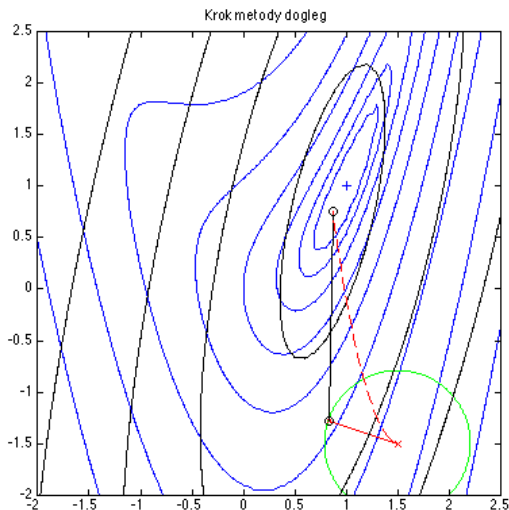
Przykład metody obszaru wiarygodności

Dogleg method



Przykład metody obszaru wiarygodności

Dogleg method



Inne metody typu TR



Są metody bardziej zaawansowane

- inne warianty dogleg – <http://www.math.unl.edu/~s-bbockel1/833-notes/node19.html>,
- https://optimization.mccormick.northwestern.edu/index.php/Trust-region_methods
- i wiele innych.

Zagadnienie pokrewne

Rozwiązywanie układu równań nieliniowych



- Metody:
 - Z zastosowaniem macierzy Jacobiego.
 - Sprowadzenie do problemu optymalizacji – minimalizacja sumy residuów.
- Zastosowanie w analizie i symulacji
 - Aproksymacja nieliniowa
 - Rozwiązywanie równań różniczkowych metodami niejawnymi, strzałów wielokrotnych, kolokacyjnymi itp.

Tu są pola zastosowania różniczkowania automatycznego, arytmetyki przedziałowej itp.

Optymalizacja z ograniczeniami

Sformułowanie problemu, oznaczenia



Szukamy minimum funkcji f

$$\min_{x \in \mathbb{R}^n} f(x),$$

które spełnia warunki równościowe

$$c_i(x) = 0, \quad i \in \mathcal{E}$$

oraz nierównościowe

$$c_i(x) \geq 0, \quad i \in \mathcal{I}.$$

Numeryczne metody optymalizacji z ograniczeniami



- eliminacja zmiennych,
- metody z funkcją kary,
- metoda Lagrange'a,
- Quadratic Programming,
- SQP - Sequential Quadratic Programming

Warunek konieczny pierwszego rzędu

Lagrangian:

$$L(x, \lambda) = f(x) - \sum_{i \in \mathcal{E} \cup \mathcal{I}} \lambda_i c_i(x)$$

Warunek konieczny minimum (1. rzędu):

Jeżeli funkcja f ma lokalne minimum w punkcie x^* , to istnieje wektor λ^* taki, że:

$$\nabla_x L(x^*, \lambda^*) = 0,$$

$$c_i(x^*) = 0, \quad \text{dla każdego } i \in \mathcal{E},$$

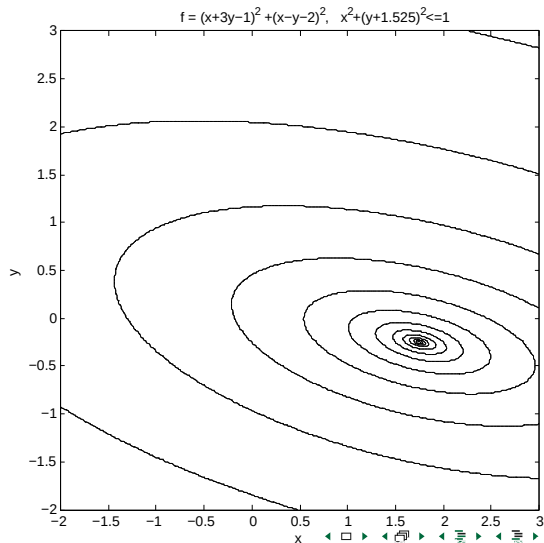
$$c_i(x^*) \geq 0, \quad \text{dla każdego } i \in \mathcal{I},$$

$$\lambda_i^* \geq 0, \quad \text{dla każdego } i \in \mathcal{I},$$

$$\lambda_i^* c_i(x^*) = 0, \quad \text{dla każdego } i \in \mathcal{E} \cup \mathcal{I}.$$

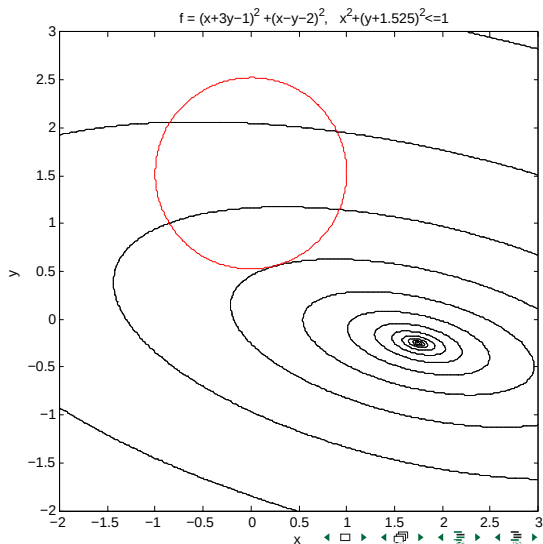
Optymalizacja z ograniczeniem nierównościowym

Ilustracja warunku $\nabla_x [f(x^*) - \lambda^* c(x^*)] = 0$



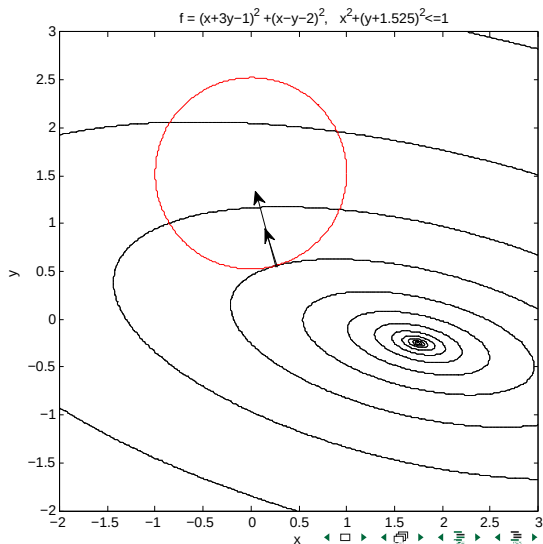
Optymalizacja z ograniczeniem nierównościowym

Ilustracja warunku $\nabla_x [f(x^*) - \lambda^* c(x^*)] = 0$



Optymalizacja z ograniczeniem nierównościowym

Ilustracja warunku $\nabla_x [f(x^*) - \lambda^* c(x^*)] = 0$



Kryterium stopu



- Zagadnienie istotne, nie tylko w problemach „zazsumionych”.
- „Numeryczna niejednoznaczność” zera albo wartości minimalnej wynikająca z błędu numerycznego obliczania wartości funkcji.
- Uwaga praktyczna: Nie należy wymagać dokładności względnej lepszej niż rzędu 10^{-8} (w przypadku arytmetyki na liczbach typu podwójnej precyzji – o dokładności maszynowej $\varepsilon = 2.22 \cdot 10^{-16}$).