Sprawozdanie – Laboratorium nr 10

Poszukiwanie minimum wartości funkcji metodą największego spadku w 2D

Tomasz Rajchel

2019/05/16

Wstęp teoretyczny

Metoda największego spadku

Metoda największego spadku jest jedną z kierunkowych metod poszukiwania minimum funkcji. Wykorzystujemy w niej gradient szukanej funkcji by “spadać” w kierunku najmniejszej wartości funkcji (lub wnosić się do największej jeżeli zmienimy znak). Metoda ta jest niedeterministyczna, wynik zależy od punktu startowego który zwykle losujemy i powtarzamy algorytm wiele razy dla różnych punktów startowych.

Jako warunek stopu wybieramy zwykle jeden z poniższych:

a) $\|x^{i+1}-x^{i}\|<\epsilon$ - kolejne przybliżenia minimum zmieniają się o bardzo małą wartość ε

b) $\nabla f(x)=0$ - gradient szukanej funkcji w x jest równy 0.

c) $\|x^{i+1}-x^{i}\|$ -norma rośnie w kolejnych iteracjach (brak zbieżności)

Opis zadania

Zadanie polega na wyznaczenie minimum funkcji:

$$f(\vec{r})=f(x,y)=\frac{5}{2}(x^2-y)^2+(1-x)^2 \tag{1}$$

Używając metody największego spadku.
W metodzie tej startujemy od przbliżenia $\vec{r_0}$, które w kolejnych iteracjach “poprawiamy”

$$\vec{r_{i+1}}=\vec{r_i}-h\cdot\nabla f(\vec{r_i}) \tag{2}$$

Gdzie gradient to $\nabla f(\vec{r})$:

$$\nabla f(\vec{r})=\left[\frac{\partial f}{\partial x},\frac{\partial f}{\partial y}\right] \tag{3}$$

Składowe gradientu liczymy następująco:

$$\frac{\partial f(x_i,y_i)}{\partial x}\approx\frac{f(x_i+\Delta x,y_i)-f(x_i-\Delta x,y_i)}{2\Delta x} \tag{4}$$

$$\frac{\partial f(x_i, y_i)}{\partial y} \approx \frac{f(x_i, y_i + \Delta y) - f(x_i, y_i - \Delta y)}{2 \Delta y} \quad (5)$$

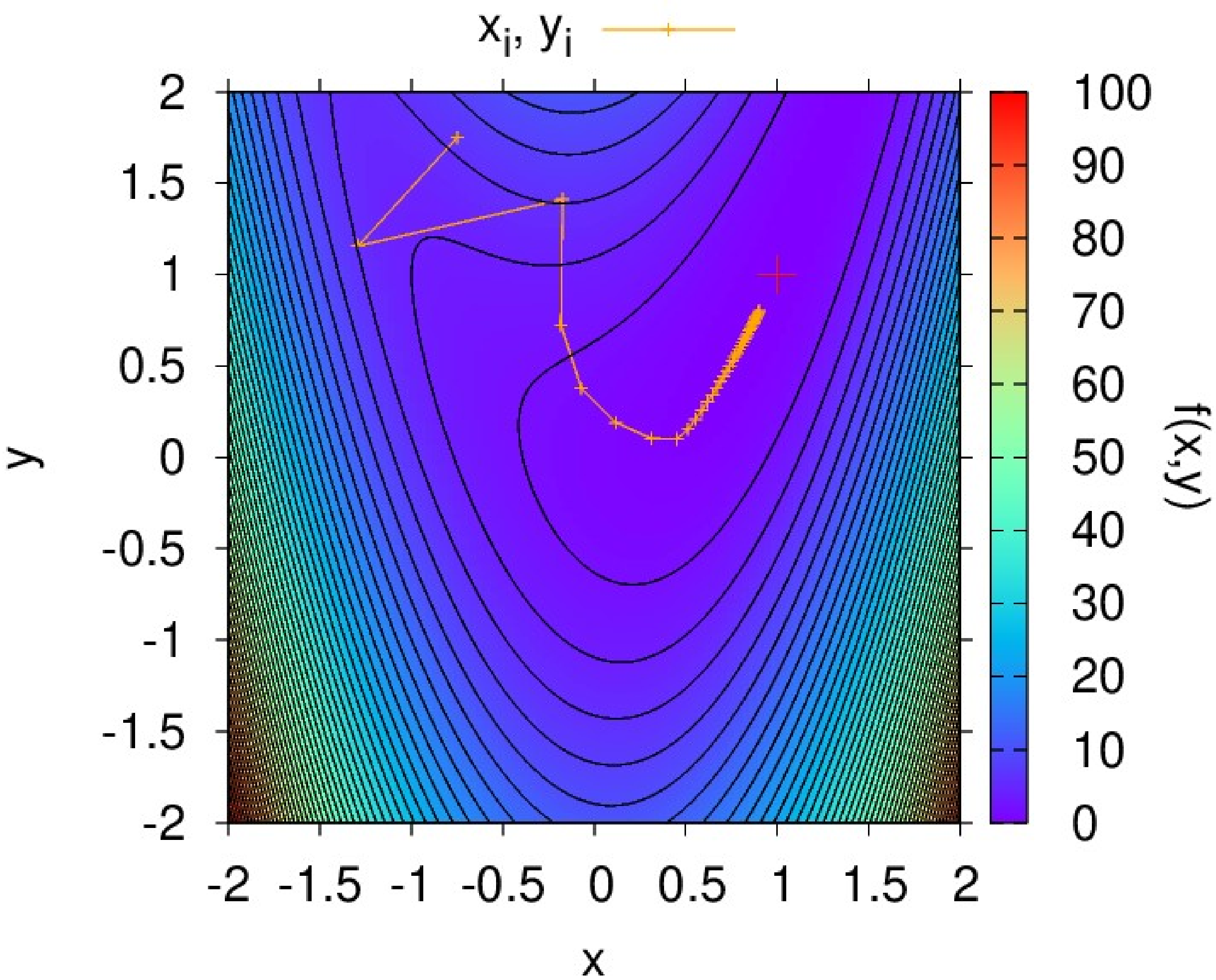
Przy kroku przestrzennym $\Delta x = \Delta y = 10^{-4}$

Startujemy z punktu $\vec{r_0} = [-0.75, 1.75]$, przy stałej h = 0.1. Jako warunek stopu przyjmujemy $\| \vec{r_{i+1}} - \vec{r} \|_2 < \epsilon$ (norma euklidesowa), maksymalną liczbę iteracji ustawiamy na 1000.
Wykonamy obliczenia dla

a) $\epsilon = 10^{-2}$
b) $\epsilon = 10^{-3}$

Wyniki

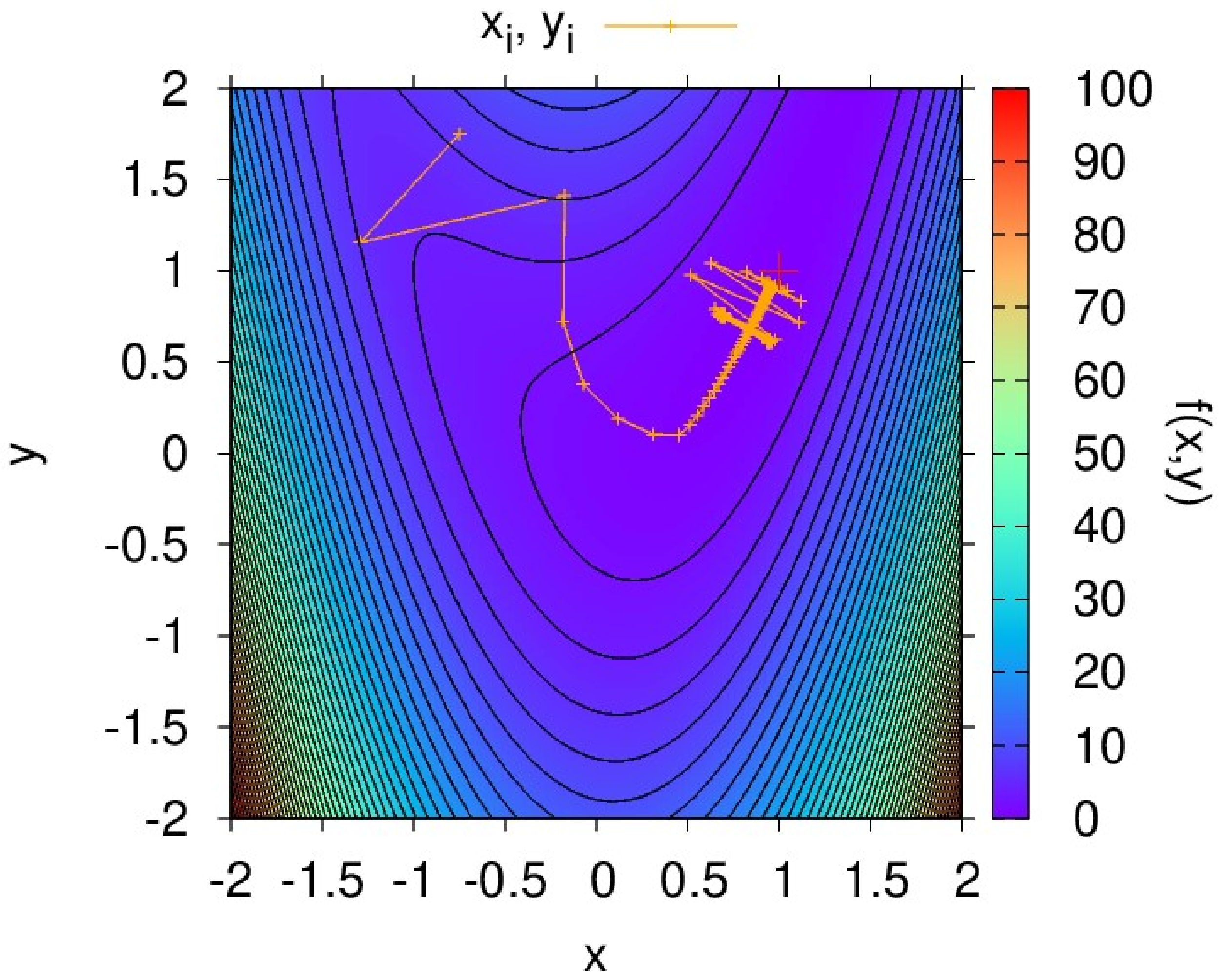
a)



$\epsilon = 10^{-2}$, program wykonał 37 iteracji

Algorytm szybko znajdywał kolejne przybliżenia minimum funkcji „staczając” się po wykresie funkcji jednak z powodu bardzo małych zmian wartości funkcji miał problem z dokładniejszym określeniem minimum i ostatecznie podał (x = 0.904391, y = 0.801376). Minimum znajduje się w punkcie (1, 1).

b)



$\epsilon = 10^{-3}$, program wykonał 1000 iteracji

Po zaostrzeniu warunku końcowego, algorytm dotarł do coraz dokładniejszych położeń minimum jednak bardzo blisko minimum funkcji kolejne przybliżenia zaczeły wokół niego oscylować. Ostatecznie algorytm podał przbliżenie (0.693964, 0.761186)

Wnioski

a) czy warunek stopu jest właściwy?

Warunek stopu jest prawidłowy. Algorytm kończy działanie gdy kolejne przybliżenia zmieniają się o bardzo małą wartosć.

b) dlaczego uzyskane przybliżenie jest dalekie od dokładnego

Ponieważ obszar funkcji wokół minimum przypomina dolinę. Algorytm zamiast podążać w dół doliny to przypadkowo „wspina” się po jej ścianach które są bardzo strome i oscyluje wokół osi doliny.

c) jaki wpływ na rozwiązanie ma utrzymywanie stałej wartości h?

Jeżeli nie korzystamy z optymalizacji parametru h, to „skoki” kolejnych przybliżeń nie są regulowane. Pogarsza to dokładność i zbieżność naszej metody.

Problem metody gradientowej w dolinie może zostać złagodzony przez dodawanie „pędu” poprzednich przybliżeń do wyliczania nowego przybliżenia.

Metoda gradientowa działa lepiej przy większej ilości wymiarów, ponieważ prawdopodobieństwo, że utknie w miejsu gdzie gradient jest bliski zeru jest mniejsze.