Sprawozdanie - Laboratorium 10
Poszukiwanie minimum wartości funkcji metodą symulowanego wyżarzania

Marta Dychała

19 maja 2021

1 Wstęp teoretyczny

Minimum funkcji n zmiennych to taki punkt $x_i = [x_1, x_2, ..., x_n]^T$ dla którego zachodzi:

$$\min f(x) = f(x_i) \Leftrightarrow \forall_{x \in R^n} f(x_i) < f(x)$$

z warunkami zawężającymi przestrzeń dopuszczalnych rozwiązań:

$$\begin{aligned} g_j(x) &\leqslant 0, \quad j = 1, 2, ..., m \\ h_j(x) &= 0, \quad j = 1, 2, ..., r \end{aligned}$$

Funkcje $g(x)$ i $h(x)$ to funkcje skalarne określające warunki jakie musi spełniać rozwiązanie.

Jednym ze sposobów znajdowania minimum funkcji jest metoda symulowanego wyżarzania. Jest to metoda stochastyczna (z rodziny metod Monte Carlo); innymi słowy polega ona na losowym przekształcaniu danego rozwiązania, dopóki nie będzie ono najlepszym możliwym. W metodzie symulowanego wyżarzania, gdzie szukane jest minimum funkcji dwóch zmiennych, początkowo na płaszczyźnie wybierane jest N punktów (x_i, y_i) zwanych także „wędrowcami”, gdzie $i = 0, 1, ..., N-1$. Następnie dla każdego z „wędrowców” dokonuje się k przemieszczeń, w których za każdym razem losowane jest nowe położenie $(x_i+\Delta x_i, y_i+\Delta y_i)$. Wędrowiec jest przenoszony do nowego położenia pod warunkiem, że wartość funkcji f tamże jest mniejsza niż w bieżącym położeniu danego wędrowca. Jeżeli tak nie jest, to nowe położenie może być zaakceptowane z pewnym prawdopodobieństwem opisanym wzorem poniżej:

$$P = \exp\left(-\frac{f(x_i + \Delta x) - f(x_i)}{T}\right), \tag{1}$$

pod warunkiem, że jest ono mniejsze od liczby wylosowanej w danej iteracji po k:

$$d_{rand}(0, 1) < P,$$

gdzie $d_{rand}(0, 1)$ to pseudolosowa liczba zmiennoprzecinkowa podwójnej precyzji z przedziału $[0, 1]$. W przypadku kiedy i ten warunek nie będzie spełniony, wędrowiec pozostaje w niezmienionym położeniu.

Parametr T, który pojawia się we wzorze (1) to tak zwana temperatura wyżarzania. Jest to współczynnik, który zmniejsza się wraz ze zwiększaniem się numeru iteracji odpowiedzialnego za obniżanie temperatury. Oznacza to, że dla późniejszych iteracji „gorsze” położenia są coraz rzadziej akceptowane. Przemieszczanie wędrowca do „gorszego” położenia może być opłacalne w początkowej fazie algorytmu, ponieważ z powodu jego losowości trudno przewidzieć czy za pewnym maksimum lokalnym nie może się znajdować minimum globalne.

Po zakończeniu działania algorytmu, szukany jest taki wędrowiec, dla którego wartość funkcji jest najmniejsza, w wyniku czego jego położenie jest szukanym minimum globalnym.

2 Zadanie do wykonania

2.1 Opis problemu

W czasie zajęć laboratoryjnych zadaniem było znalezienie minimum globalnego funkcji f określonej wzorem:

$$f(x,y) = \sin(x)\sin(y) - \exp\left(-\left(x+\frac{\pi}{2}\right)^2 - \left(y-\frac{\pi}{2}\right)^2\right) \tag{2}$$

W tym celu należało użyć metody symulowanego wyżarzania. Parametr T w i_T-tej iteracji przyjmował wartość:

$$T = \frac{10}{2^{i_T}} \tag{3}$$

gdzie $i_T = 0, 1, 2, ..., 20$. Symulację przeprowadzono dla $N = 200$ wędrowców. Dla każdego z nich w i_T-tej iteracji położenie było zmieniane $k = 100$ razy (tzw. kroki błądzenia). Minimum globalne było wyszukiwane na obszarze $[x_{min}, x_{max}] \times [y_{min}, y_{max}] = [-10, 10] \times [-10, 10]$. Każdy z wędrowców początkowo znajdował się w punkcie $(5, 5)$. W przypadku, gdy nowe położenie danego wędrowca nie mieściłoby się w zadanym przedziale (gdzie Δx oraz Δy to liczby pseudolosowe z przedziału $[-1, 1]$):

$$x + \Delta x \notin [-10, 10] \lor y + \Delta y \notin [-10, 10]$$

to w zależności, która wartość współrzędnej - x czy y - przekraczałaby zakres, tak redukowano parametr Δx lub Δy, aby nowe położenie wędrowca znajdowało się na najbliższym mu z krańców przedziału.

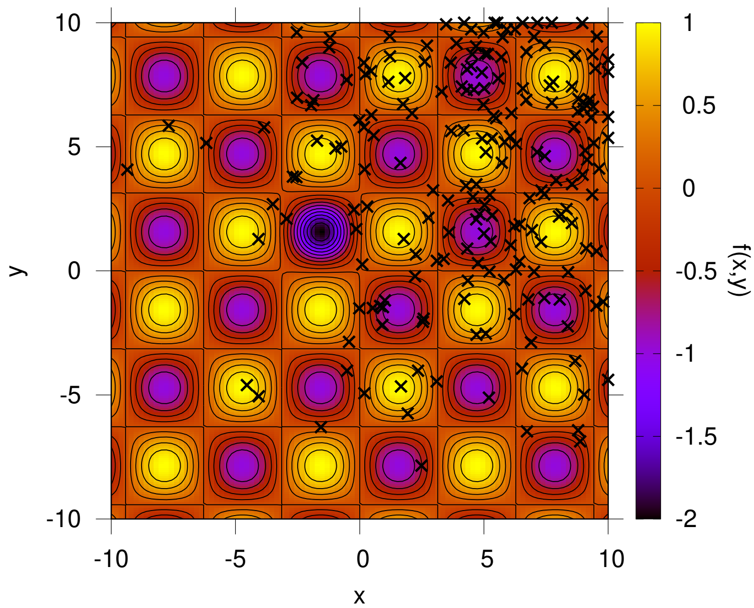
Położenia wszystkich wędrowców dla $i_T \in \{0, 7, 20\}$ zapisano do pliku w celu ich zobrazowania na wykresach. Ponadto dla jednego z wędrowców dokonano pomiaru wartości funkcji w wszystkich jego położeniach w czasie wykonywania się programu.

Wszystkie obliczenia potrzebne do zrealizowania zadania napisano w języku C++. Wszelkie wyniki uzyskano przy użyciu typu double.

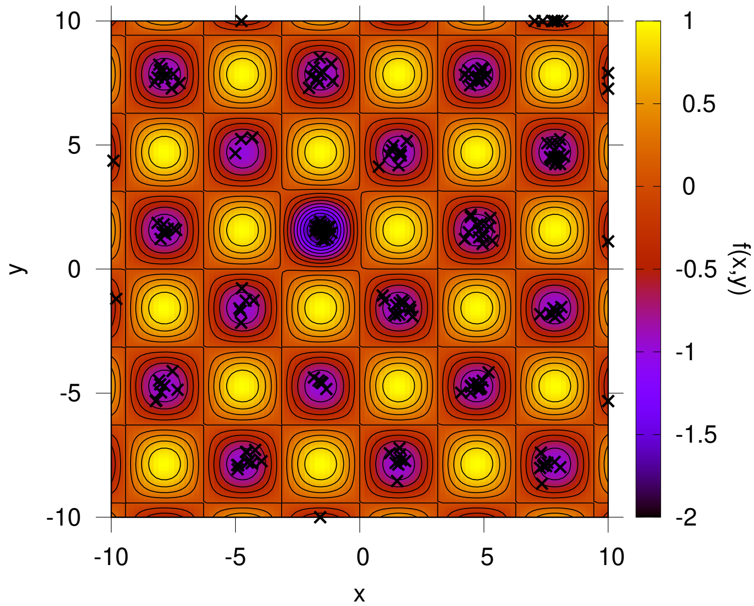
2.2 Wyniki

Do wygenerowania wykresów wykorzystano program Gnuplot.

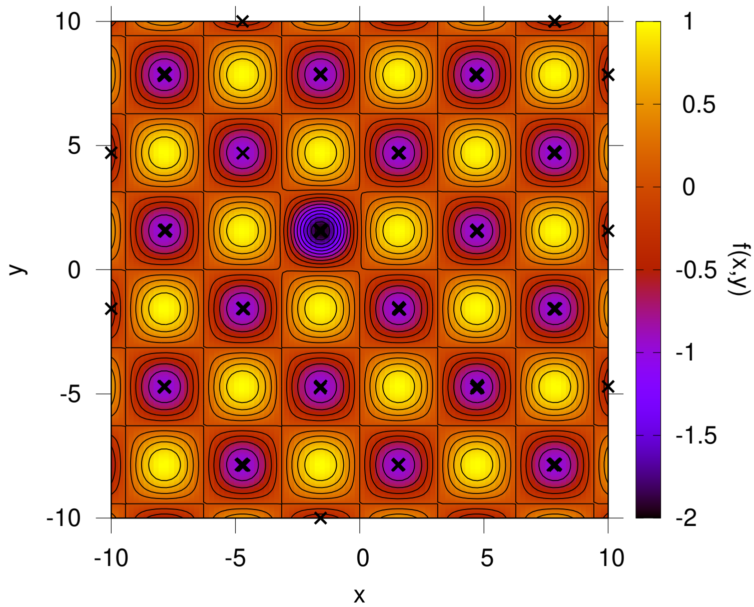
Poniżej znajdują się położenia każdego z 200 wędrowców dla $i_T \in \{0, 7, 20\}$:



Rysunek 1: Położenia wędrowców po zakończonym błądzeniu dla $i_T = 0$



Rysunek 2: Położenia wędrowców po zakończonym błądzeniu dla $i_T = 7$

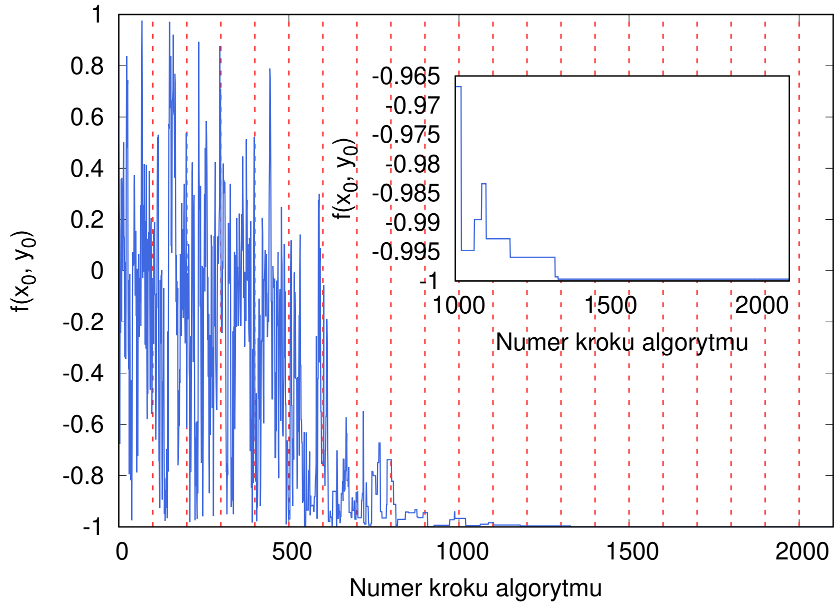


Rysunek 3: Położenia wędrowców po zakończonym błądzeniu dla $i_T = 20$

Na wykresie dla przypadku $i_T = 0$ można zauważyć wyraźne rozproszenie danych, które spowodowane jest wysoką temperaturą początkową $T = 10$, w wyniku czego gorsze położenia są często akceptowane. Jednak już po 8 iteracjach wędrowcy „gromadzą się” w okolicy minimów lokalnych, ponieważ prawdopodobieństwo przyjęcia gorszego położenia jest dużo mniejsze, niemal bliskie zeru. Dlatego też dla $i_T = 20$ wszyscy wędrowcy znajdują się albo w otoczeniach minimów lokalnych, albo na krańcach przedziałów w okolicach, gdzie funkcja maleje.

Na podstawie wyznaczonych położeń wędrowców po zakończeniu symulacji, wyznaczono minimum funkcji (2) znajdując wędrowca dla którego wartość funkcji była najmniejsza. Otrzymane minimum globalne dla konkretnego przebiegu algorytmu przedstawionego na rysunkach 1,2,3 wyniosło $f(x, y) = -1.99999$ dla $(x, y) = (-1.57104, 1.57311)$. Wynik więc jest zbliżony do teoretycznego, który wynosi $f(x, y) = -2$ dla $(x, y) = (-\frac{\pi}{2}, \frac{\pi}{2})$. Ponieważ algorytm użyty do zrealizowania ćwiczenia jest probabilistyczny, ponownie uruchomienie programu może zwrócić inne minimum lokalne, ale także zbliżone do wartości teoretycznej.

Dla wędrowca o indeksie 0 dokonano pomiarów wartości funkcji w kolejnych położeniach:



Rysunek 4: Wartość funkcji $f(x, y)$ we wszystkich położeniach pierwszego wędrowca w czasie trwania algorytmu.

Po lewej stronie wykresu można zaobserwować liczne oscylacje wartości funkcji, które są związane z początkowo wysoką temperaturą wyżarzania. Wraz ze wzrostem temperatury wartości oscylują jednak coraz mniej, by w końcu ustabilizować się w okolicy minimum. W tym przypadku znalezione minimum jest lokalne, ponieważ - tak jak to było wcześniej wspomniane - minimum globalne tej funkcji to $f(x, y) = -2$. Gdyby ten wędrowiec był jedynym wędrowcem w całej symulacji, to w celu znalezienia minimum globalnego należało by uruchomić program nieokreśloną liczbę razy, co świadczyłoby o nieskuteczności algorytmu.

3 Wnioski

Pomimo pewnej losowości w działaniu, metoda symulowanego wyżarzania okazuje się być bardzo skuteczną metodą znajdowania ekstremów lokalnych, a także globalnych. Losowość doboru położeń wędrowców można zniwelować poprzez dobranie ich odpowiedniej liczby oraz stopniowe zmniejszanie temperatury.

Odpowiednie dobranie liczby wędrowców pozwala na zwiększenie prawdopodobieństwa znalezienia ekstremum globalnego. Ponieważ położenia wędrowców zmieniają się z pewną losowością, możliwe jest że część z nich „dotrze” do minimum globalnego. Za pomocą jednego wędrowca także jest możliwe odnalezienie minimum lokalnego, jednak dużo trudniej jest odnaleźć minimum globalne, przynajmniej dla funkcji rozważanej w czasie zajęć.

Temperatura jest wyznacznikiem, który wskazuje jak często gorsze położenie może być akceptowane. Stosując obniżanie temperatury można kontrolować stopień zmiany rozkładu położeń wędrowców. W przypadku braku obniżania temperatury istotne staje się dobranie jej wartości. Przy dużych wartościach temperatury, gorsze położenia wędrowców są akceptowane częściej, co oznacza większą chaotyczność otrzymanych danych i w rezultacie potrzebna jest o wiele większa liczba iteracji do znalezienia minimów. Dla bardzo małych wartości temperatur bardzo szybko można znaleźć minima. Z drugiej strony niemalże nigdy nie są rozważanie gorsze położenia, poza którymi może być minimum globalne, co może być problematyczne przy chociażby znajdowaniu ekstremów globalnych funkcji takich jak będąca przedmiotem rozważań funkcja (2). Z tego nasuwa się wniosek, że obniżanie temperatury jest konieczne, ponieważ pozwala na lepsze oszacowanie położeń i wartości minimów funkcji - początkowo przyjmuje się z większym prawdopodobieństwem gorsze położenia, by wraz z postępowaniem algorytmu, kiedy to wędrowcy znajdują się w okolicy minimów, zmniejszać tolerancję akceptacji gorszych położeń.