# SYMULOWANE WYŻARZANIE W ZASTOSOWANIU DO WYZNACZANIA EKSTREMUM GLOBALNEGO FUNKCJI O WIELU EKSTREMACH LOKALNYCH DALEKO ODDALONYCH OD SIEBIE LUB BARDZO ZAG....

**Chapter** · January 2014



**1 author:**

Stanisław Kowalik
The University of Dabrowa Gornicza, Poland, Dąbrowa Górnicza

# SYMULOWANE WYŻARZANIE W ZASTOSOWANIU DO WYZNACZANIA EKSTREMUM GLOBALNEGO FUNKCJI O WIELU EKSTERMACH LOKALNYCH DALEKO ODDALONYCH OD SIEBIE LUB BARDZO ZAGĘSZCZONYCH

STANISŁAW KOWALIK*

**Streszczenie**

W tym rozdziale wykonano obliczenia określenia ekstremum globalnego funkcji w przypadku istnienia wielu ekstremów lokalnych. Obliczeń dokonano wykorzystując algorytm symulowanego wyżarzania. Zastosowano zmodyfikowaną wersję tego algorytmu z zapamiętywaniem najlepszego rozwiązania. Do obliczeń użyto programu MATLAB. W rozdziale przedstawiono pięć przykładów funkcji, dla których wyznaczano maksimum globalne. Funkcje tak skonstruowano specjalnie, aby maksima lokalne były daleko oddalone od siebie lub w których funkcje posiadały ekstrema lokalne bardzo gęsto usytuowane obok siebie. Celem obliczeń było sprawdzenie, czy metoda symulowanego wyżarzania poprawnie wyznaczy maksimum globalne dla tych trudnych do obliczeń funkcji.

**Słowa kluczowe:** ekstremum globalne, ekstremum lokalne, wyżarzanie, algorytm, Nicholas Metropolis, iteracja.

## 1. Wprowadzenie

W przypadku funkcji o wielu ekstremach lokalnych, klasyczne metody gradientowe nie są skuteczne w znajdywaniu ekstremum globalnego. Znajdywane jest zazwyczaj ekstremum lokalne zależne od wyboru punktu startowego. W celu znalezienia ekstremum globalnego stosuje się inne metody takie jak szukanie przypadkowe, symulowane wyżarzanie, algorytmy genetyczne, metodę roju cząstek (PSO). W tym rozdziale podano przykłady wykorzystania symulowanego wyżarzania do znajdywania maksimum globalnego funkcji.

W metodzie symulowanego wyżarzania stosuje się tzw. algorytm Metropolisa. Nazwa pochodzi od nazwiska Nicholasa Constantine Metropolisa (1915-1999) amerykańskiego fizyka, informatyka i matematyka, współtwórcy metody symulowanego wyżarzania i metody Monte Carlo. Przodkowie jego pochodzili z Grecji. Był również współtwórca komputerów MANIAC I (1952) i MANIAC II (1957). Podczas II wojny światowej pracował w zespole Enrico Fermiego w Chicago przy budowie pierwszego reaktora jądrowego, a potem od 1943 roku uczestniczył w Projekcie Manhattan w Los Alamos. Tam opracował równania stanu dla materii w ekstremalnych warunkach, które były wykorzystywane w symulacji broni jądrowej [8].

* Wyższa Szkoła Biznesu w Dąbrowie Górniczej

Algorytm symulowanego wyżarzania jest algorytmem wyszukującym przybliżone rozwiązanie dla problemów NP-trudnych. Problemy NP-trudne to takie, gdzie przestrzeń możliwych rozwiązań jest tak duża, że sprawdzenie wszystkich rozwiązań jest niemożliwe. Trwałoby to setki lub tysiące lat. Wykorzystując algorytmy przybliżone jesteśmy w stanie znaleźć rozwiązanie, które nie będzie najlepszym (optymalnym), jednak będzie „wystarczająco dobrym" i zostanie znalezione w czasie w miarę krótkim.

## 2. Symulowane wyżarzanie

Symulowane wyżarzanie to rodzaj algorytmu heurystycznego przeszukującego przestrzeń alternatywnych rozwiązań problemu w celu wyszukania rozwiązań najlepszych. Sposób działania symulowanego wyżarzania przypomina zjawisko wyżarzania w metalurgii (także w hutnictwie szkła) [4].

Wyżarzanie (ang.: annealing) jest to metoda obróbki cieplnej materiału polegająca na ogrzewaniu przedmiotu z metalu lub stopu metali do określonej temperatury, utrzymaniu go w niej przez pewien czas, a następnie powolnym ostudzaniu [3, 5]. Wyżarzanie przeprowadza się, by usunąć naprężenia i niejednorodności, uzyskać pożądaną drobnoziarnistą strukturę materiału, a także np. w celu rekrystalizacji. W zależności od potrzeb stosuje się różne temperatury i czasy wyżarzania oraz różne czasy i sposoby studzenia [5].

Do opisu i wytłumaczenia takiego zachowania się układów fizycznych wykorzystuje się rozkład Boltzmanna: $P(E) = \exp(-E/kT)$, który określa prawdopodobieństwo znalezienia się układu utrzymywanego w temperaturze T w stanie o energii E (k jest tzw. stałą Boltzmanna) [7].

W odniesieniu do obliczeń komputerowych schemat algorytmu symulowanego wyżarzania można przedstawić w następujących punktach (dla określania minimum funkcji) [6]:

1. wyznaczenie rozwiązania początkowego s,
2. wyznaczenie temperatury początkowej T,
3. następnie należy powtarzać cykl obliczeń aż do osiągnięcia wartości prawdy (true) przez warunek zatrzymania

for i = 0 to M

wyznaczenie losowo sąsiedniego rozwiązania $s' \in N(s)$

$\delta = f(s') - f(s)$

if $\delta < 0$ then $s = s'$

else

wylosowanie x z zakresu (0, 1)

if $x < \exp(-\delta/kT)$ then $s = s'$

$T = \alpha(T)$

4. sprawdzenie warunku zatrzymania, czy ma wartość prawdy (true)
5. wydrukowanie (wyświetlenie) rozwiązania s.

W tym schemacie mamy następujące oznaczenia:

*s* - bieżące rozwiązanie,

N(s)- zbiór sąsiednich rozwiązań dla rozwiązania s,

δ - różnica kosztów rozwiązań: nowego i poprzedniego (różnica wartości funkcji dla rozwiązania nowego i poprzedniego), f (s) - funkcja oceny rozwiązania (funkcja kosztu, wartość funkcji f dla rozwiązania s, wartość funkcji f w punkcie s),

T - aktualna temperatura,

$\alpha$(T) - funkcja zmiany temperatury,

M - liczba iteracji obliczeniowych.

Ważną różnicą pomiędzy pierwotnymi metodami iteracyjnymi, a algorytmem symulowanego wyżarzania jest możliwość wyboru przez niego gorszego rozwiązania. Wybór taki jest dokonywany z pewnym prawdopodobieństwem. Dzięki temu algorytm symulowanego wyżarzania może w określonych warunkach wyjść ze znalezionego minimum lokalnego i dalej podążać w kierunku rozwiązania optymalnego. Parametrem algorytmu, który ma wpływ na prawdopodobieństwo wyboru gorszego rozwiązania jest parametr przeniesiony bezpośrednio z podstaw termodynamicznych algorytmu, czyli temperatura. Im wyższa, tym prawdopodobieństwo wyboru gorszego rozwiązania jest większe. Im niższa, tym algorytm jest bardziej zbliżony w działaniu do typowych metod iteracyjnych. To właśnie znajduje odzwierciedlenie w drugim ważnym aspekcie algorytmu symulowanego wyżarzania czyli w powolnym ochładzaniu [6].

Na początku działania algorytmu temperatura jest wysoka, dzięki czemu algorytm może bardzo często zmieniać konfiguracje rozwiązania, niejednokrotnie wybierając rozwiązanie gorsze. Wraz z kolejnymi iteracjami algorytmu temperatura spada i wybierane są częściej rozwiązania lepsze. Pod koniec pracy algorytmu, temperatura jest na tyle niska, ze prawdopodobieństwo wyboru gorszego rozwiązania jest bliskie zeru. Algorytm zachowuje się wówczas, jak typowy algorytm iteracyjny i stara się maksymalnie ulepszyć rozwiązanie [6].

Oprócz przedstawionego powyżej schematu algorytmu, stosuje się często również jego zmodyfikowaną wersję z zapamiętywaniem najlepszego rozwiązania. W cyklu obliczeń stosuje się dodatkowo instrukcję warunkową if $f(s') < f(s_B)$ then $s_B = s'$ zapamiętującą najlepsze rozwiązanie znalezione od początku obliczeń aż do chwili bieżącej ($s_B$ najlepsze rozwiązanie) [6]. W prezentowanej pracy zastosowano tą modyfikację.

## 3. Przykłady obliczania maksimum funkcji o ekstremach bardzo oddalonych od siebie

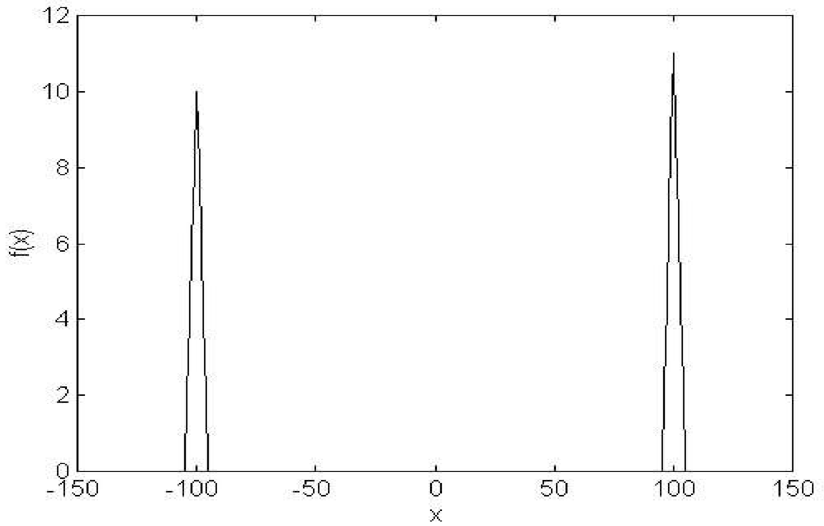
W celu sprawdzenia, jak działa ta metoda w przypadku bardzo nietypowych funkcji, przeprowadzono obliczenia dla funkcji o odległych od siebie ekstremach lokalnych. Wyniki przedstawiono w przykładach 1, 2 i 3.

**Przykład 1**

Dana jest funkcja f(x) w przedziale [-150, 150]. Określona jest ona wzorem

$$f(x)=\begin{cases} -2|x+100|+10 \quad dla \quad x\in(-105,-95) \\ -2.2|x-100|+11 \quad dla \quad x\in(95,105) \\ 0 \quad dla \quad x\notin(-105,-95)\cup(95,105) \end{cases} \tag{1}$$

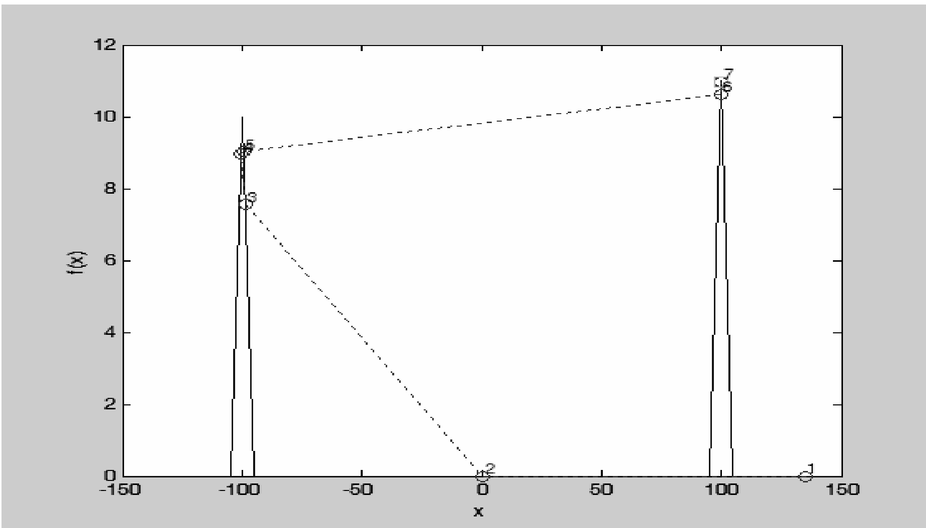
Funkcja ta jest przedstawiona na rysunku 1.




**Rys. 1.** Funkcja utworzona z dwóch modułów

Posiada ona tylko dwa maksima lokalne. Oddalone są one od siebie o 200. Funkcja została tak specjalnie dobrana, aby było wiadomo z góry, w których punktach ma maksima i ile one wynoszą. Dla tej funkcji jest: f(-100)=10, f(100)=11. Funkcja utworzona jest z modułów i posiada wartości różne od zera jedynie w przedziałach (-105, -95) i (95, 105). Dla pozostałych argumentów z przedziału [-150, 150] funkcja ma wartość 0.

Należy określić maksimum globalne tej funkcji. Do obliczeń przyjęto następujące wartości początkowe: T=500, $\alpha$(T)=0.999·T, k=0.1, M=3000. Na rysunku 2 zaznaczono linią przerywaną kolejne etapy znajdywania ekstremum globalnego funkcji. Kółeczka z numerami obok oznaczają kolejne wartości ekstremalne znajdywane w trakcie obliczeń.




**Rys. 2.** Etapy znajdywania ekstremum globalnego funkcji dla funkcji o 2 maksimach lokalnych

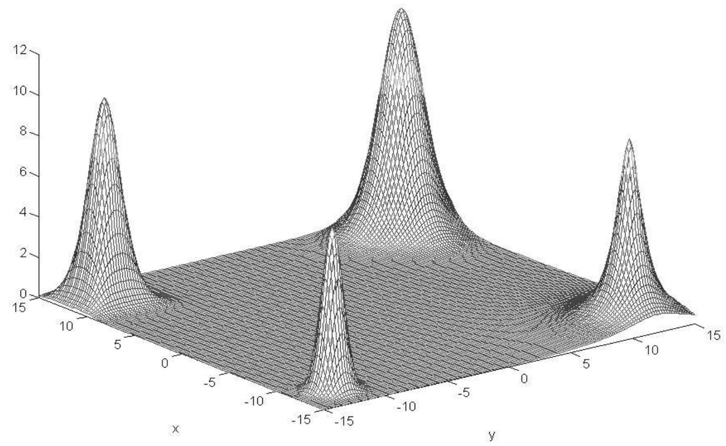
Na 3000 iteracji tylko 7 razy było poprawiane ekstremum na większą wartość. W pozostałych przypadkach wartość funkcji była 0 lub była mniejsza od poprzednio znalezionej maksymalnej na danym etapie obliczeń. Po tych 3000 iteracji i 7 korekcjach rozwiązania otrzymano: wartość funkcji wyniosła f(x)=10.980462 dla x=100.008881.

**Przykład 2**

Dana jest funkcja f(x, y) w obszarze [-15, 15]×[-15, 15]. Określona ona jest wzorem:

$$\begin{gathered} f(x,y) = 8\cdot\exp(-(x+12)^2-(y+12)^2)+ \\ +9/(1+(x+12)^2+(y-12)^2)+ \\ +20/(\cosh(x-12)^2+\cosh(y+12)^2)+ \\ +176/((\exp(x-12)+2+\exp(-x+12))\cdot(\exp(y-12)+2+\exp(-y+12))) \end{gathered} \tag{2}$$

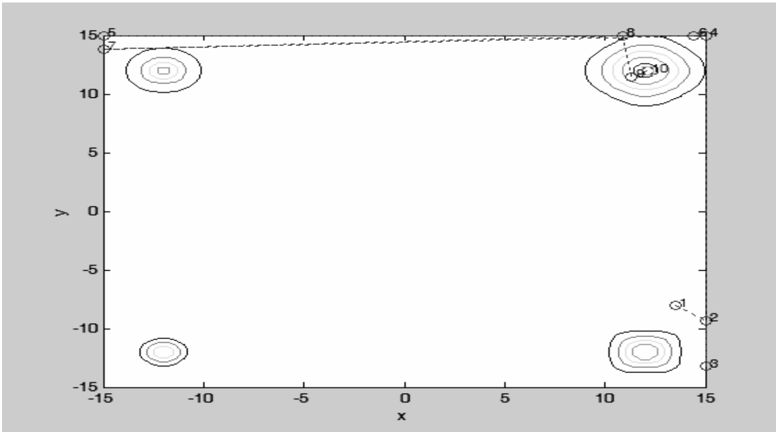
Funkcja ta jest przedstawiona na rysunku 3.




**Rys. 3.** Funkcja o czterech maksimach lokalnych

Funkcja ta posiada cztery maksima lokalne. Oddalone są one od siebie o 24. Funkcja została tak specjalnie dobrana, aby było wiadomo z góry, w których punktach ma maksima i ile one wynoszą. Maksima występują w punktach (-12, -12), (-12, 12), (12, -12), (12, 12). Kolejne składniki tej funkcji brane z osobna posiadają maksima wynoszące kolejno: 8, 9, 10, 11. Funkcja rozpatrywana w całości posiada następujące maksima: f(-12, -12)=8.015598, f(-12, 12)=9.000000, f(12, -12)=10.007806, f(12, 12)=11.015598.

Należy określić maksimum globalne tej funkcji. Do obliczeń przyjęto następujące wartości początkowe: T=90, $\alpha$(T)=0.999·T, k=0.5, M=200. Na rysunku 4 zaznaczono linią przerywaną kolejne etapy znajdywania ekstremum globalnego funkcji. Kółeczka z numerami obok oznaczają kolejne wartości ekstremalne znajdywane w trakcie obliczeń.



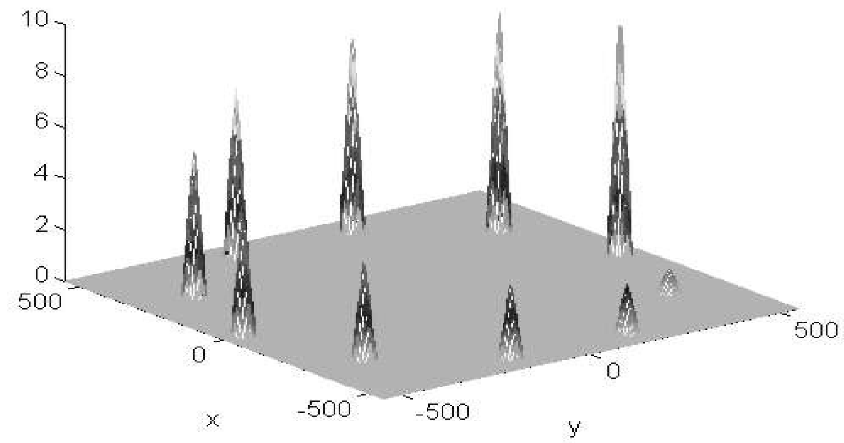

**Rys. 4.** Etapy znajdywania ekstremum globalnego funkcji dla funkcji o 4 maksimach lokalnych

Na 200 iteracji tylko 10 razy było poprawiane ekstremum na większą wartość. W pozostałych przypadkach wartość funkcji była bliska 0 lub była mniejsza od poprzednio znalezionej maksymalnej na danym etapie obliczeń. Po tych 200 iteracjach i 10 korekcjach rozwiązania otrzymano: wartość funkcji wyniosła f(x,y)= 10.972868 dla x=12.107300 i y=11.936668.

**Przykład 3**

Dana jest funkcja f(x, y) w przedziale [-550, 550]. Funkcja ta utworzona jest z dziesięciu stożków rozmieszczonych na okręgu o promieniu 500 i środku w początku układu współrzędnych. Podstawy stożków mają promienie równe 25, a wysokości wynoszą kolejno 10, 9, ..., 1. Stożki te są rozmieszczone na promieniach okręgu oddalonych od siebie co $\pi/5$.

Funkcja ta jest przedstawiona na rysunku 5.




**Rys. 5.** Funkcja utworzona z dziesięciu stożków

Współrzędne środków stożków (xs, ys) oraz wysokości stożków h zawarte są w tabeli 1.

**Tabela 1.** Współrzędne środków stożków

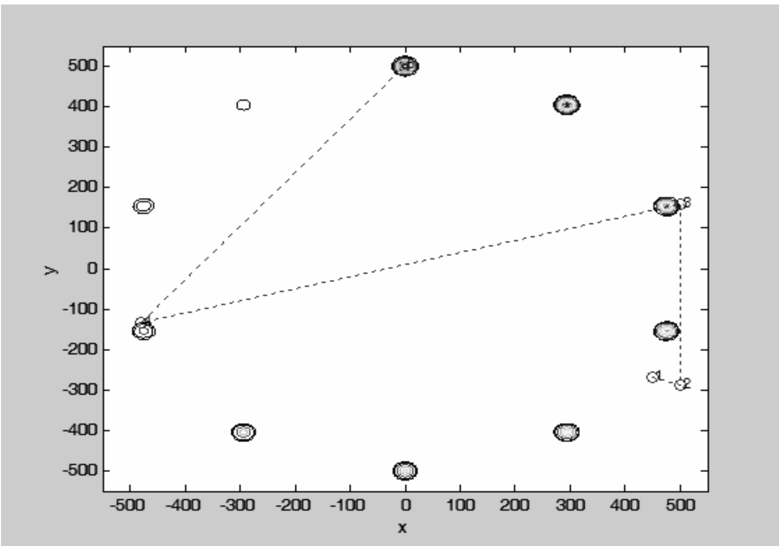
| nr (k) | xs | ys | h |
|---|---|---|---|
| 1 | 0.0000 | 500.0000 | 10 |
| 2 | 293.8926 | 404.5085 | 9 |
| 3 | 475.5283 | 154.5085 | 8 |
| 4 | 475.5283 | -154.5085 | 7 |
| 5 | 293.8926 | -404.5085 | 6 |
| 6 | 0.0000 | -500.0000 | 5 |
| 7 | -293.8926 | -404.5085 | 4 |
| 8 | -475.5283 | -154.5085 | 3 |
| 9 | -475.5283 | 154.5085 | 2 |
| 10 | -293.8926 | 404.5085 | 1 |

Wartości poszczególnych stożków (dla k=1, ..., 10) były wyznaczane według wzoru

$$f(x, y) = -\frac{h(k)}{r}\sqrt{(x - xs(k))^2 + (y - ys(k))^2} + h(k) \tag{3}$$

gdzie:
r = 25 jest promieniem podstawy stożków,
h(k) = 10, 9, ..., 1 są wysokościami kolejnych stożków.

Dla punktów (x, y) dalej odległych od środków stożków (xs(k), ys(k)) niż promień r=25, przyjęto wartość funkcji f(x, y)=0.




**Rys. 6.** Etapy znajdywania ekstremum globalnego funkcji dla funkcjio 10 maksimach lokalnych

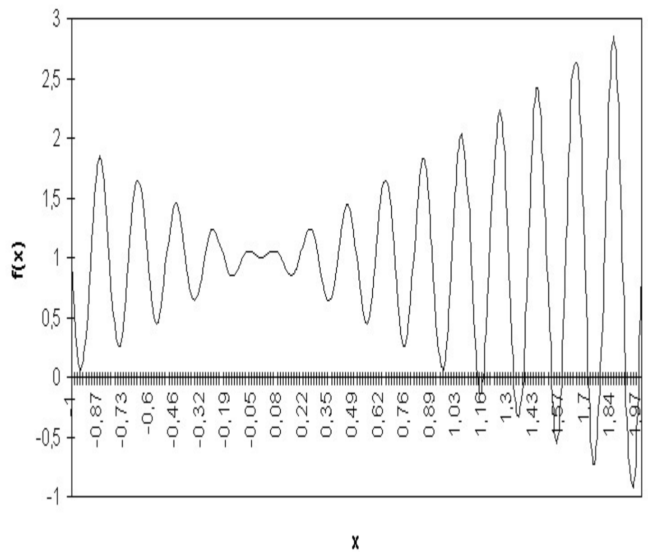
Należy określić maksimum globalne tej funkcji. Do obliczeń przyjęto następujące wartości początkowe: T=1400, α(T)=0.999·T, k=0.1, M=100. Na rysunku 6 zaznaczono linią przerywaną kolejne etapy znajdywania ekstremum globalnego funkcji. Kółeczka z numerami obok oznaczają kolejne wartości ekstremalne znajdywane w trakcie obliczeń.

Na 100 iteracji tylko 5 razy było poprawiane ekstremum na większą wartość. W pozostałych przypadkach wartość funkcji była 0 lub była mniejsza od poprzednio znalezionej maksymalnej na danym etapie obliczeń. Po tych 100 iteracjach i 5 korekcjach rozwiązania otrzymano: wartość funkcji wyniosła f(x,y)=9.989801 dla x=-0.025497 i y=500.000000.

## 4. Przykłady obliczania maksimum funkcji o ekstremach bliskich siebie

**Przykład 4**

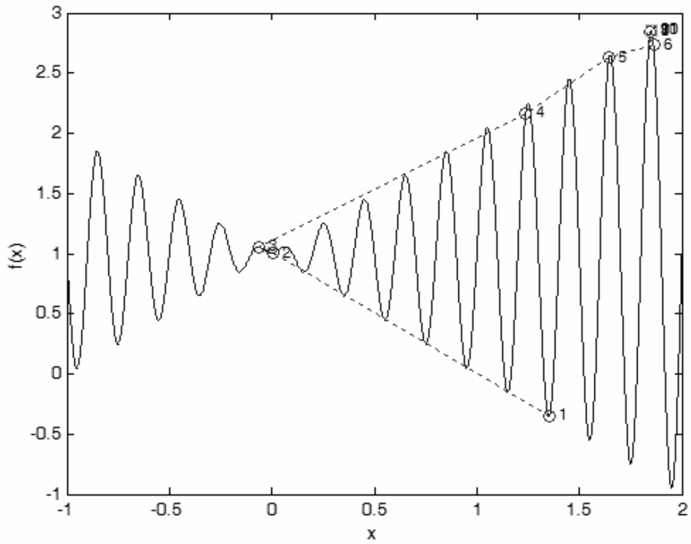
Należy określić maksimum funkcji f(x)=x·sin(10πx)+1 w przedziale [-1, 2]. Funkcja ta jest przedstawiona na rysunku 7 [1, 2].




**Rys. 7.** Wykres funkcji f(x)=x·sin(10πx)+1

Funkcja ta posiada 17 maksimów lokalnych w przedziale [-1, 2] (wliczając w to także wartości na końcach przedziału). Maksimum globalne w tym przedziale występuje dla x=1.850547 i wynosi f(x)=2.85027376656965.

Należy określić maksimum globalne tej funkcji. Do obliczeń przyjęto następujące wartości początkowe: T=5, α(T)=0.997·T, k=0.1, M=1200. Na rysunku 8 zaznaczono linią przerywaną kolejne etapy znajdywania ekstremum globalnego funkcji. Kółeczka z numerami obok oznaczają kolejne wartości ekstremalne znajdywane w trakcie obliczeń.




**Rys. 8.** Etapy znajdywania ekstremum globalnego funkcji f(x)=x·sin(10·pi·x)+1

Na 1200 iteracji tylko 11 razy było poprawiane ekstremum na większą wartość. W pozostałych przypadkach wartość funkcji była mniejsza od poprzednio znalezionej maksymalnej na danym etapie obliczeń. Po tych 1200 iteracji i 11 korekcjach rozwiązania otrzymano: wartość funkcji wyniosła f(x)= 2.85027325340030 dla x=1.85052376133795. Z dokładnością 6 miejsc po przecinku otrzymano dokładną wartość funkcji a argument x różni się od dokładnego dopiero na 5 miejscu po przecinku.

W tabeli 2 przedstawiono kolejne etapy poprawiania wartości maksymalnej.

**Tabela 2.** Kolejne etapy poprawiania wartości maksymalnej

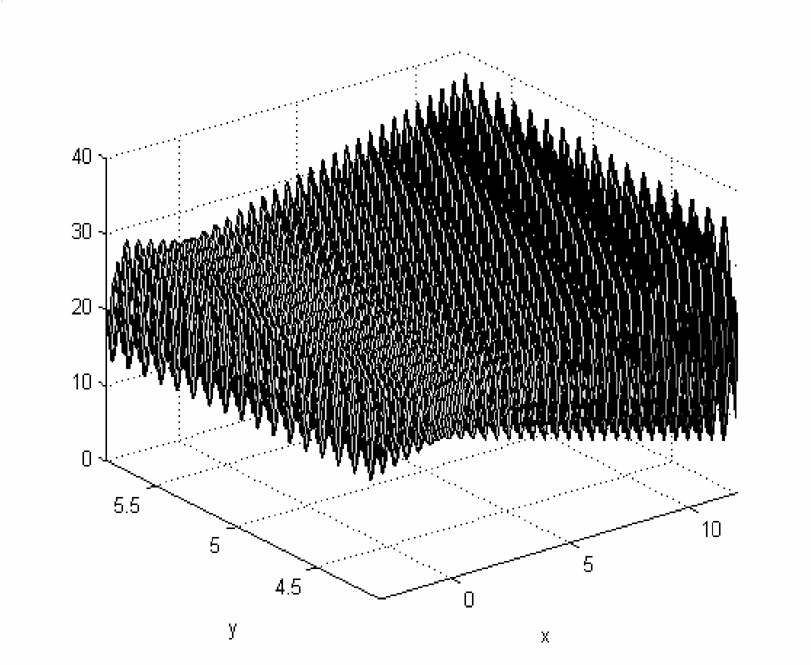
| nr | X | f(x) |
|---|---|---|
| 1 | 1.35038785544153 | -0.35028761045134 |
| 2 | 0.00608042331297 | 1.00115444457923 |
| 3 | -0.06379697017118 | 1.05789730747448 |
| 4 | 1.23883608433246 | 2.16342063622794 |
| 5 | 1.64527265054883 | 2.62716155812742 |
| 6 | 1.86199258827883 | 2.73139648177747 |
| 7 | 1.84760291290835 | 2.84236641709338 |
| 8 | 1.84934742008058 | 2.84895878638185 |
| 9 | 1.85082753691331 | 2.85020209642664 |
| 10 | 1.85052327560565 | 2.85027323214605 |
| 11 | 1.85052376133795 | 2.85027325340030 |

**Przykład 5**

Dana jest funkcja f(x, y) w obszarze [-3, 12]×[4.1, 5.8]. Określona ona jest wzorem [2]:

$$f(x,y) = 21.5 + x \cdot \sin(4 \cdot \pi \cdot x) + y \cdot \sin(20 \cdot \pi \cdot y) \qquad (4)$$

Funkcja ta jest przedstawiona na rysunku 9 [1, 2].

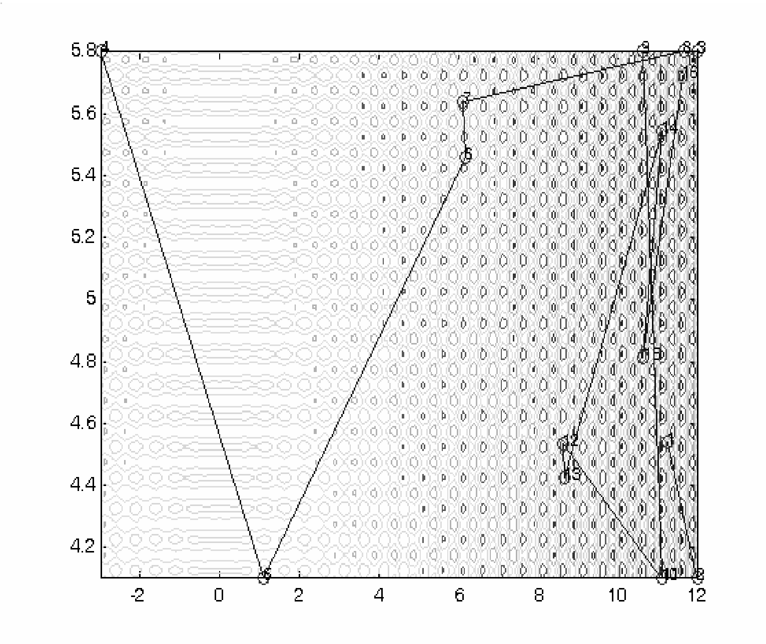



**Rys. 9.** Funkcja f(x,y) = 21.5 + x·sin(4·π·x) + y·sin(20·π·y)

Funkcja ta posiada bardzo dużo maksimów lokalnych w danym obszarze. Występują one bardzo blisko siebie. Maksimum globalne w tym obszarze wynosi 38.850294 i występuje dla x=11.625545 i y=5.7250444.

Należy określić maksimum globalne tej funkcji. Do obliczeń przyjęto następujące wartości początkowe: T=100, α(T)=0.999·T, k=0.2, M=7000. Na rysunku 10 zaznaczono linią przerywaną kolejne etapy znajdywania ekstremum globalnego funkcji. Kółeczka z numerami obok oznaczają kolejne wartości ekstremalne znajdywane w trakcie obliczeń.

Na 7000 iteracji 19 razy było poprawiane ekstremum na większą wartość. W pozostałych przypadkach wartość funkcji była mniejsza od poprzednio znalezionej maksymalnej na danym etapie obliczeń. Po 7000 iteracjach i 19 korekcjach rozwiązania otrzymano: wartość funkcji wyniosła f(x,y)=38.849009 dla x=11.625310 i y=5.724714.




**Rys. 10.** Etapy znajdywania ekstremum globalnego funkcji
$f(x,y)=21.5+x\cdot\sin(4\cdot\pi\cdot x)+y\cdot\sin(20\cdot\pi\cdot y)$

## 5. Zakończenie

Jak pokazują przykłady metoda symulowanego pozwoliła na znalezienie przybliżonego rozwiązania i to w miarę bardzo bliskiego optymalnemu. Funkcje były celowo dobierane tak, aby maksimum globalne było trudno znaleźć. Należy zwrócić uwagę na to, różne warunki startowe metody dawały różne rozwiązania (nieraz te same). Uzyskane wyniki prezentowane w przykładach nie były otrzymane w trakcie pierwszego przypadkowego uruchamiania programu komputerowego. Otrzymano je po kilkukrotnych próbach zmieniając wartości startowe T, $\alpha$(T), k, M. W niektórych przypadkach należało znacznie zwiększyć liczbę iteracji M, aby uzyskać zadowalające rozwiązanie. Wartość temperatury T na starcie przyjmowano tak, aby obejmowała cały przedział zmienności każdej współrzędnej. Do każdego przykładu ustalano wartości startowe indywidualnie.

## 6. Bibliografia

[1] KOWALIK S: *Wyznaczanie ekstremum globalnego funkcji wielu zmiennych. Metoda strefowo-równoległa*. Wydawnictwo Politechniki Śląskiej, Gliwice 2005.

[2] MICHALEWICZ Z.: *Algorytmy genetyczne + struktury danych = programy ewolucyjne.* WNT, Warszawa 1996.

[3] http://pl.wikipedia.org/wiki/Wyżarzanie.
[4] http://pl.wikipedia.org/wiki/Symulowane_wyżarzanie.
[5] http://portalwiedzy.onet.pl/36330,,,,wyzarzanie,haslo.html.
[6] http://zsi.ii.us.edu.pl/~mboryczka/pliki/hsw.pps.
[7] http://pl.wikipedia.org/wiki/Rozkład_Boltzmanna.
[8] http://de.wikipedia.org/wiki/Nicholas_Metropolis.

## SIMULATED ANNEALING APPLIED TO DETERMINE THE GLOBAL EXTREME FUNCTIONALITY OF MANY LOCAL EXTREMES OF VERY DISTANT FROM EACH OTHER, OR VERY DENSE

**Abstract**

In the chapter, calculations were made to determine the extreme global functions where there are several local extremes. The calculations were made using the simulated annealing algorithm. The modified version of the algorithm with memory the best solution employ. Uses a MATLAB software to calculate. This chapter presents five examples of functions for which the global maximum was determined. Functions designed specifically so that the local maximum were far from each other or were very close to each other. The purpose of the calculation was to check whether the simulated annealing method correctly determines global maximum for those difficult to calculate the function.

**Key words:** global extreme, local extreme, annealing, algorithm, Nicholas Metropolis, iteration.