Sprawozdanie z listy 3

Michał Gancarczyk 244923

Streszczenie

Celem doświadczeń przeprowadzonych w ramach listy zadań jest przeanalizowanie algorytmów służących do obliczenia miejsc zerowych funkcji na różne sposoby.

Sprawozdanie poświęca jedną sekcję każdemu z sześciu zadań. W skład każdej sekcji wchodzą kolejno: krótki opis przedstawionego w zadaniu problemu, wyniki uzyskane przy użyciu tego algorytmu, opcjonalne wykresy obrazujące przedstawione wcześniej dane, oraz wypływające z doświadczenia wnioski.

Do rozwiązania zadań użyty został język Julia, wersja 1.2.0.

1 Metoda Bisekcji

Metoda opiera się na twierdzeniu Bolzana-Cauchy'ego:

"Jeżeli funkcja ciągła $f(x)$ ma na końcach przedziału domkniętego wartości różnych znaków, to wewnątrz tego przedziału, istnieje co najmniej jeden pierwiastek równania $f(x) = 0$".

Jako dane wejściowe zadana jest funkcja, oraz przedział przeszukiwania [a, b]. Metoda polega ona sprawdzaniu czy pierwiastkiem analizowanego równania jest punkt $x_0 = \frac{a+b}{2}$. Jeśli x_0 nie przekracza maksymalnego błędu, to stanowi ono wynik algorytmu, a w przypadku niepowodzenia początkowo zadany przedział ograniczany jest o połowę jego długości. Powyższe kroki powtarzane są w pętli, a algorytm kończy swoją pracę, gdy odpowiednio precyzyjny pierwiastek x_0 zostanie znaleziony, lub napotkany zostanie stan błędny.

Warunkami początkowymi, wymaganymi dla poprawnego działania algorytmu jest ciągłość funkcji i zadanym przedziale, oraz przeciwne znaki wartości funkcji na jego krańcach.

Poniżej zaprezentowana jest reprezentacja opisanego algorytmu w formie pseudokodu:

Algorithm 1: Metoda Bisekcji

```
function mbisekcji (f, a, b, δ, ϵ)
Input  : Funkcja f, krańce przedziału a i b, oraz marginesy dokładności δ, ϵ
Output: x_0, f(x_0), liczba iteracji, lub ewentualny błąd
if sign(f(a)) = sign(f(b)) then
    error "funkcja nie zmienia znaku w przedziale [a, b]"
end
x_1 ← 0
it ← 0
while |a − b| > ϵ do
    x_1 ← a + (b−a)/2
    if |x_1| ≤ delta ∨ |f(x_1)| ≤ ϵ then
        return x1, f(x_1), it
    else if f(x_1) · f(a) < 0 then
        b ← x1
    else
        a ← x_1
    end
    it ← it + 1
end
```

2 Metoda Stycznych

W ramach algorytmu początkowo wyznaczana jest styczna w punkcie x_0, a jako pierwsze przybliżenie pierwiastka przyjęte jest trywialne do wyliczenia miejsce zerowe stycznej.

Jeśli wartość bezwzględna wartości funkcji w wyznaczonym miejscu zerowym nie przekracza zadanych liczbami ϵ i δ marginesów błędu, to uzyskana liczba stanowi wynik algorytmu. W przeciwnym wypadku, poprzez poprowadzenie prostej prostopadłej do poprzedniej stycznej i wyznaczenia jej punktu przecięcia z f, obliczana jest kolejna styczna, której miejsce zerowe pozwala uzyskać dokładniejdzy wynik. Operacje te są powtarzane do momentu uzyskania satysfakcjonującego wyniku, bądź napotkania nieprawidłowości w danych.

Dla uzyskania poprawnych wyników algorytmu niezbędne jest spełnienie poniższych warunków:

1. Funkcja f oraz jej pierwsza i druga pochodna są ciągłe w badanym przedziale [a, b]
2. Wewnątrz obszaru przeszukiwania $[a, b]$ znajduje się dokładnie jeden pierwiastek
3. $sign(f(a)) \neq sign(f(b))$
4. Pierwsza i druga pochodna mają stały znak w badanym przedziale [a, b]

Algorithm 2: Metoda Stycznych

function *mstycznych* (f, f' x_0, δ, ϵ, $maxit$)
Input : Funkcja f, jej pochodna f', początek przedziału x_0, marginesy dokładności δ, ϵ, oraz maksymalna liczba iteracji $maxit$
Output: x_0, $f(x_0)$, liczba iteracji, lub ewentualny błąd
$x1 \leftarrow x_0 - 1$
$f_0 \leftarrow f(x_0)$
$it \leftarrow 0$
while $|x_1 - x_0| > \delta \wedge |f_0| > \epsilon$ **do**
    if $it = maxit$ **then**
        error "nie osiągnięto wymaganej dokładności w *maxit* iteracji"
    end
    $f_1 \leftarrow f'(x_0)$
    if $|f_1 < \epsilon|$ **then**
        error "pochodna bliska zeru"
    end
    $x_1 \leftarrow x_0$
    $x_0 \leftarrow x_0 - (f_0/f_1)$
    $f_0 \leftarrow f(x_0)$
    $it \leftarrow it + 1$
end
return x_0, f_0, it

3 Metoda Siecznych

Metoda opiera się na założeniu, że dla dostatecznie małego odcinka funkcja ciągła zmienia się w sposób liniowy. Można w tym wypadku zastąpić krzywą na odcinku $[a, b]$ przez sieczną, a jej punkt przecięcia z osią OX potraktować jako miejsce zerowe. Jeśli błąd uzyskanego miejsca zerowego nie przekracza zadanego marginesu, to algorytm kończy działanie zwracając punkt przecięcia jako wynik. W wypadku, gdy błąd jest zbyt duży, wartość przyliżanej funkcji w miejscu zerowym i jeden z poprzenich punktów użyte są do utworzenia nowej, precyzyjniejszej siecznej. Kroki są powtarzane do momentu uzysknia wystarczająco bliskiego rzeczywistemu wyniku, lub napotkania błędu.

Poprawne działanie algorytmu wymaga, aby przybliżana funkcja, oraz jej pierwsza pochodna były ciągłe na badanym przedziale, a wartości tej funkcji na krańcach przedziału miały przeciwne znaki. Dodatkowo na zadanym przedziale powinno znajdować się tylko jedno miejsce zerowe.

Algorithm 3: Metoda Siecznych

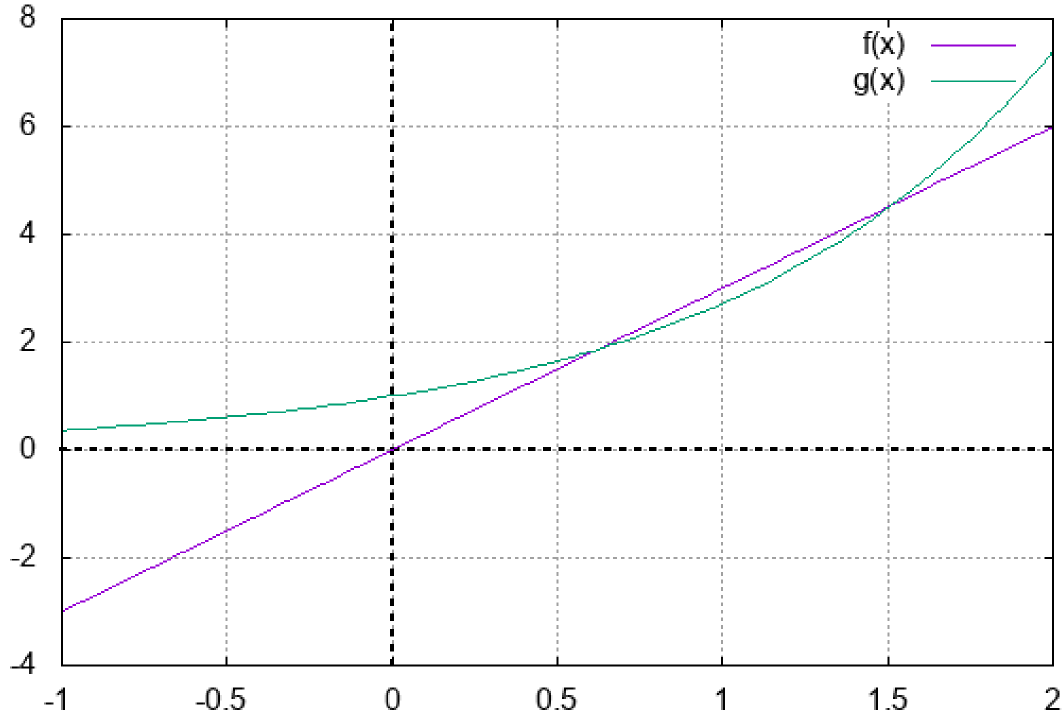
function *msiecznych* (f, x_0, x_1, δ, ϵ, $maxit$)
Input : Funkcja f, krańce przedziału x_0 i x_1, marginesy dokładności δ, ϵ, oraz maksymalna liczba iteracji $maxit$
Output: x_0, $f(x_0)$, liczba iteracji, lub ewentualny błąd
$f_0 \leftarrow f(x_0)$, $f_1 \leftarrow f(x_1)$
$it \leftarrow 0$
while $|a - b| > \epsilon$ **do**
  if $it = maxit$ **then**
    error "nie osiągnięto wymaganej dokładności w *maxit* iteracji"
  end
  if $|f_0| > f_1$ **then**
    $x_0 \leftrightarrow x_1$, $f_0 \leftrightarrow f_1$
  end
  $c \leftarrow x_0 - f_0 \cdot \frac{x_0 - x_1}{f_0 - f_1}$
  $f_c \leftarrow c$
  $x_1 \leftarrow x_0$, $f_1 \leftarrow f_0$
  $x_0 \leftarrow c$, $f_0 \leftarrow f_c$
  $it \leftarrow it + 1$
  if $|x_0 - x_1| < \delta \vee |f_c| < \epsilon$ **then**
    error "nie osiągnięto wymaganej dokładności w *maxit* iteracji"
  end
end
return err, err, err, 1

4 Zadanie 4

Wyniki			
Metoda	x_0	$f(x_0)$	Liczba iteracji
Bisekcji	1.9337539672851562	$-2.7027680138402843 \cdot 10^{-7}$	15
Stycznych	1.933753779789742	$-2.2423316314856834 \cdot 10^{-8}$	4
Siecznych	1.933753644474301	$1.564525129449379 \cdot 10^{-7}$	4

Powyższy wykres prezentuje złożoności obliczeniowe poszczególnych metod. Metoda bisekcji do osiągnięcia wyniku potrzebowała aż piętnastu iteracji w odróżnieniu od czterech, jaki wystarczyły do uzyskania wyniku w pozostałych metodach. Ta obserwacja pokrywa się z teoretycznymi zbieżnościami metod. Bisekcja ma bowiem zbieżność kwadratową, a pozostałe meody zbieżność liniową. Dalsza analiza metody bisekcji pozwala również stwierdzić, że oprócz niesatysfakcjonującej złożoności metoda ta przynosi również obarczone największym błędem wyniki. Stwierdzenie to jednak nie jest całkowicie poprawne, gdyż ze względu na sposób działania metody bisekcji, jej wynikiem jest wartość najbliższa nie zeru, a zadanej w parametrze dokładności. Odmienne zachowanie prezentują metody stycznych i siecznych, które zbiegały do zera szybciej, co pozwoliło im osiągnąć dalszą od dokładności, a bliższą faktycznemu zeru wartość. Taki rezultat jest jednak wynikiem odpowiednio dobranych parametrów i funkcji, a nie ogólnej zasady.

5 Zadanie 5



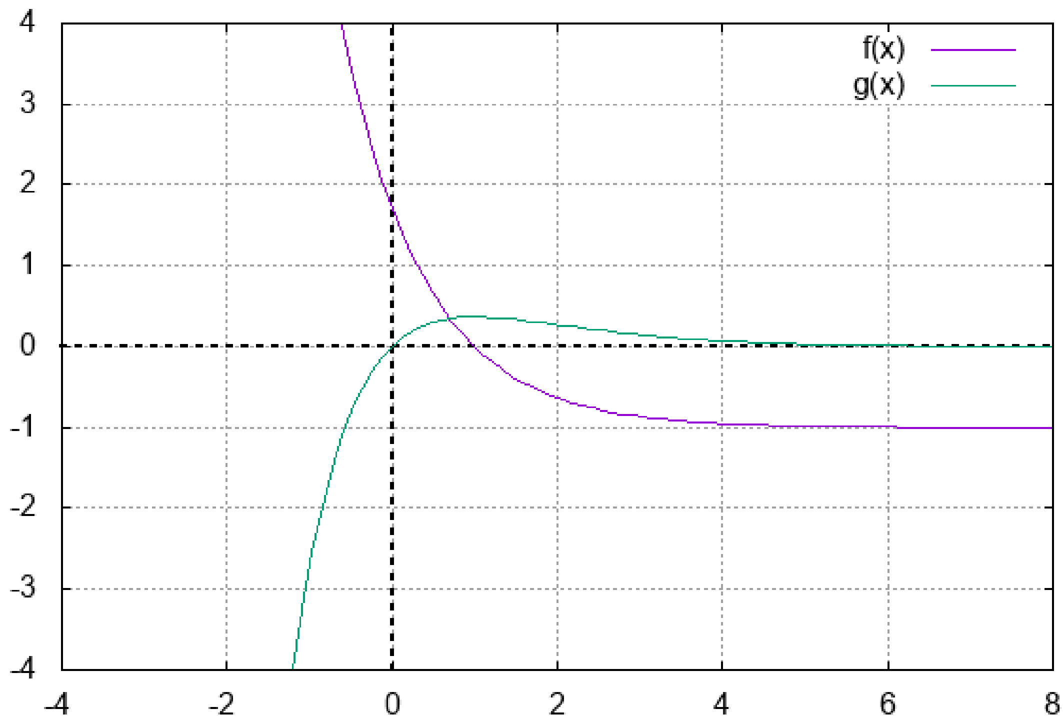
Rys. 1. Wykresy $f(x) = 3x$, $g(x) = e^x$

Miejsca zerowe				
Miejsce	Przedział	Wartość	Błąd	Liczba iteracji
x_0	[0, 1]	0.619140625	$9.066320343276146 \cdot 10^{-5}$	8
x_1	[1, 2]	1.5120849609375	$7.618578602741621 \cdot 10^{-5}$	12

Zastosowanie metody bisekcji do znalezienia punktów przecięcia powyższych funkcji prowadziło do napotkania pwenych problemów. Analiza zaprezentowanego wykresu pokazuje, że szukane miejsca zerowe będą od siębie oddalone na stosunkowo niewielki dystans. Ze względu na ten fakt, oraz na początkowe warunki, jakie wybrany dla metody przedział musi spełnić (chociażby wymóg róznicy znaków na jego krańcach), wybranie obu przedziałów nie jest proste. W ich ustaleniu posłużono się wspomnianą wcześniej reprezentacją graficzną funkcji.

Warte zauważenia jest, że dla zadanej w poleceniu funkcji preferowane byłoby użycie metody stycznych, bądź siecznych, dla których dobranie początkowych parametrów funkcji byłoby prostsze.

6 Zadanie 6



Rys. 2. Wykresy $f(x) = e^{1-x} - 1$, $g(x) = xe^{-x}$

$f(x)$ - Metoda Bisekcji				
Przedział	z	$f(z)$	Liczba iteracji	Błąd
[0, 1.5]	1.0000076293945312	$-7.6293654275305656 \cdot 10^{-6}$	15	0
[0.5, 3]	0.9999923706054688	$7.629423635080457 \cdot 10^{-6}$	15	0
[-5, 5]	0.0	1.718281828459045	0	0
[0, 100]	0.9999990463256836	$9.536747711536009 \cdot 10^{-7}$	21	0
[-10, 2000]	1.0000018030405045	$-1.803038878978036 \cdot 10^{-6}$	26	0

$g(x)$ - Metoda Bisekcji				
Przedział	z	$f(z)$	Liczba iteracji	Błąd
[-0.5, 1]	$-7.62939453125 \cdot 10^{-6}$	$-7.629452739132958 \cdot 10^{-6}$	15	0
[-0.25, 1.5]	$-7.62939453125 \cdot 10^{-6}$	$-7.629452739132958 \cdot 10^{-6}$	14	0
[-1, 6]	$-3.814697265625 \cdot 10^{-6}$	$-3.814711817567984 \cdot 10^{-6}$	17	0
[-2, 100]	49.0	2.5690139750480974e-20	0	0
[-5, 1000]	497.5	4.318056675122884e-214	0	0

$f(x)$ - Metoda Newtona				
x_0	z	$f(z)$	Liczba iteracji	Błąd
-1	0.9999922654776594	$7.734552252003368 \cdot 10^{-6}$	5	0
0	0.9999984358892101	$1.5641120130194253 \cdot 10^{-6}$	4	0
1	1.0	0.0	0	0
2	0.9999999810061002	$1.8993900008368314 \cdot 10^{-8}$	5	0
5	0.9999996427095682	$3.572904956339329 \cdot 10^{-7}$	54	0
7	0.9999999484165362	$5.15834650549607 \cdot 10^{-8}$	401	0
13	0	0	0	2

$g(x)$ - Metoda Newtona				
x_0	z	$f(z)$	Liczba iteracji	Błąd
-2	$-1.425500682806244 \cdot 10^{-9}$	$-1.425500684838296 \cdot 10^{-9}$	7	0
-1	$-3.0642493416461764 \cdot 10^{-7}$	$-3.0642502806087233 \cdot 10^{-7}$	5	0
0	0.0	0.0	0	0
0.5	$-3.0642493416461764 \cdot 10^{-7}$	$-3.0642502806087233 \cdot 10^{-7}$	5	0
1	0	0	0	2
2	14.398662765680003	$8.036415344217211 \cdot 10^{-6}$	10	0
100	100.0	$3.7200759760208363 \cdot 10^{-42}$	0	0
200	200.0	$2.767793053473475 \cdot 10^{-85}$	0	0

$f(x)$ - Metoda Siecznych					
x_0	x_1	z	$f(z)$	Liczba iteracji	Błąd
-1	2	0.9999990036367258	$9.963637706000839 \cdot 10^{-7}$	7	0
0.5	3	0.9999998801054126	$1.1989459447470097 \cdot 10^{-7}$	6	0
-2	6	5.6042543962200835	-0.989990837907685	3	0

$g(x)$ - Metoda Siecznych					
x_0	x_1	z	$f(z)$	Liczba iteracji	Błąd
-1	0.5	$-1.1737426154042664 \cdot 10^{-6}$	$-1.1737439930768023 \cdot 10^{-6}$	7	0
-0.25	1.5	$2.5381552519802256 \cdot 10^{-6}$	$2.538148809756318 \cdot 10^{-6}$	6	0
2	6	14.198625030792476	$9.679697572338075 \cdot 10^{-6}$	11	0
10	20	20.00090808104888	$4.118752554492817 \cdot 10^{-8}$	1	0
-4	400	400.0	$7.660678386856023e \cdot 10^{-172}$	1	0

Jak można wywnioskować z powyższych danych dla $f_1(x)$ metoda bisekcji jest zbieżna do 0 nawet dla bardzo dużych przedziałów. Problem można dostrzec stosując ją dla funkcji $f_2(x)$, która zero osiąga w nieskończoności. Nieodpowiednio dobrany przedział początkowy prowadzi w jej przypadku do znalezienia przez funkcję miejsca zerowego w miejscu znacznie oddalonym od prawidłowego, gdyż wartość funkcji w nim jest wystarczająco bliska zeru dla zadanego marginesu błędu.

Zastosowanie metody Newtona dla $f_1(x)$ pozwala zauważyć, że dla coraz większych od 1 wartości x_0 liczba iteracji potrzebna do wyznaczenia miejsca zerowego, gwałtownie rośnie, aż do momentu, gdy metoda staje się rozbieżna. Anomalia ta jest spowodowana ukazanym na wykresie przebiegiem funkcji - od pewnego momentu maleje ona bardzo wolno, przez co pochodna w punkcie osiąga wartość 0 uniemożliwiając poprawne działanie metody.

W przypadku $f_2(x)$ metoda nie wyznacza miejsca zerowego dla $x_0 = 1$ ze względu na pochodną równą 0 w tym punkcie, dodatkowy problem zaobserwować można dla każdej wartości przybliżenia początkowego większej od 1, gdzie podobnie jak przy metodzie bisekcji wynikiem jest liczba wystarczająco bliska 0 dla zadanego marginesu. Podobnie jak w metodzie bisekcji jest to skutkiem dążenia do 0 w nieskończonosci przez $f_2(x)$.

Obserwacja wyników zastosowania metody siecznych dla $f_1(x)$ pozwala zaobserwować podobne problemy z rozbieżnością jak przy metodzie stycznych. Zjawisko, jakiego nie spotkano stosując wcześniejsze metody można jednak zaobserwować dla początkowych przybliżeń $x_0 = -2$, $x_1 = 6$, dla których wyznacony zestaje punkt, gdzie wartość funkcji jest bliska -1, a nie oczekiwanemu 0. Spowodowane jest to bardzo dużą różnicą między wartościami funkcji w punktach -2 i 6, co w efekcie doprowadza do "ominięcia" faktycznego miejsca zerowego funkcji i wyznaczenia kolejnego przybliżenia blisko 6. Kolejne kroki iteracyjne zmniejszają różnicę między x_0 i x_1, która ostatecznie staje się mniejsza od delty, co kończy wykonywanie algorytmu z niedokładnym wynikiem.

Analogicznie do poprzednich metod, metoda siecznych dla funkcji $f_2(x)$ i odpowiednio odległych przybliżeń wyznacza miejsce zerowe funkcji, gdy dąży ona do nieskończoności.