Obliczenia Naukowe
Lista 3
Laboratoria

Piotr Szyma

25 listopada 2017

1 Zadanie 1

1.1 Opis problemu

Zadanie polegało na zaimplementowaniu algorytmu rozwiązującego równanie $f(x) = 0$ METODĄ BISEKCJI.

1.2 Analiza

Dane: $f, a, b, \delta, \epsilon$
Wynik: $x, f(x), i, err$
$u \leftarrow f(a)$; $v \leftarrow f(b)$
$e \leftarrow b - a$
if $sgn(u) = sgn(v)$ **then**
| return (0, 0, 0, 1);
end
while *true* **do**
| $i\ = i + 1$
| $e \leftarrow \frac{e}{2}$
| $c \leftarrow a + e$
| $w \leftarrow f(c)$
| **if** $|e| < \delta\ or\ |w| < \epsilon$ **then**
| | return $(w, f(x), i, 0)$;
| **end**
| **if** $sgn(w) \neq sgn(u)$ **then**
| | $b \leftarrow c$; $v \leftarrow w$
| **else**
| | $a \leftarrow c$; $u \leftarrow w$
| **end**
end

Jeśli f jest funkcją ciągłą w przedziale $[a, b]$ i jeśli $f(a)f(b) < 0$, a więc f zmienia znak w $[a, b]$ to ta funkcja musi mieć zero w (a, b). Jest to konsekwencja własności Darboux funkcji ciągłych. Metoda bisekcji korzysta z tej własności. Jeśli $f(a)f(b) < 0$, to obliczamy $c = \frac{a+b}{2}$ i sprawdzamy, czy $f(a)f(c) < 0$. Jesli tak, to f ma zero w $[a, c]$; w przeciwnym wypadku jest $f(c)f(b) < 0$; wtedy pod a podstawiamy c. W obu przypadkach przedział $[a, b]$, dwa razy krótszy od poprzedniego, zawiera zero funkcji f, więc postępowanie można powtórzyć.

Warto zwrócić uwagę na pewne istotne elementy algorytmu. Po pierwsze punkt środkowy c nie jest obliczany standardowo, tj. $c = \frac{a+b}{2}$, ponieważ nową wielkość lepiej jest obliczać dodając do poprzedniej małą poprawkę, czyli $c = a + \frac{b-a}{2}$ Kolejnym ważnym czynnikiem jest sprawdzanie znaku - tj. złym sposobem jest sprawdzanie go poprzez $wu < 0$, lepiej $sgn(u) \neq sgn(w)$. Nie można również zapomnieć o pewnej tolerancji błędu, zarówno od strony argumentów δ, jak i wartości ϵ. Warto również zastosować ograniczenie M maksymalnej ilości kroków. W tym wypadku jednak w treści zadania zostały nałożone inne ograniczenia.

1.3 Implementacja

Moja implementacja algorytmu w języku Julia znajduje się w module `MyModule` znajdującym się w pliku załączonym do tego sprawozdania.

2 Zadanie 2

2.1 Opis problemu

Zadanie polegało na zaimplementowaniu algorytmu rozwiązującego równanie $f(x) = 0$ METODĄ STYCZNYCH.

2.2 Analiza

Dane: $f, x_0, M, \delta, \epsilon$
Wynik: $x, f(x), i, err$
$v \leftarrow f(x_0)$;
if $|v| < \epsilon$ **then**
| return (0, 0, 0, 1);
end
do
| $k += 1$
| $x_1 \leftarrow x_0 - \frac{v}{f'(x_0)}$
| $v \leftarrow f(x_1)$
| **if** $|x_1 - x_0| < \delta$ *or* $|v| < \epsilon$ **then**
| | return $(x_1, v, i, 0)$;
| **end**
| $x_0 \leftarrow x_1$
while $k < M$;

Metoda Newtona polega na linearyzacji wykresu w oparciu o wzór Taylora, tj.

$$f(x) = f(c) + f'(c)(x-c) + \frac{1}{2!}f''(c)(x-c)^2 + \dots$$

$$l(x) = f(c) + f'(c)(x-c)$$

Jest możliwa do zastosowania na przedziale spełniającym następujące warunki:

1. w przedziale tym powinien znajdować się dokładnie jeden pierwiastek
2. wartości funkcji na jego krańcach muszą być różnych znaków
3. pierwsza i druga pochodna funkcji w tym przedziale mają mieć stały znak

Na początku wybierany jest punkt x_0, następnie wyliczana jest styczna do funkcji w punkcie x_0. (do wyliczenia stycznej do funkcji potrzebna jest znajomość pochodnej funkcji) Wybierany jest kolejny punkt, x_1 - punkt przecięcia stycznej z osią OX. Punkt x_1 staje się nowym bazowym, na którym wykonujywana jest operację analogiczną, jak do x_0. Wyznaczana jest kolejna styczna, jej przecięcie z osią odciętych staje się kolejnym punktem. Wyznaczając kolejne punkty, punkt przecięcia zbliża się do punktu zerowego. Operacja jest powtarzana aż do osiągnięcia wymaganej dokładności.
Metoda Newtona jest metodą o lokalnej zbieżności - aby móc wyznaczyć poprawne miejsce zerowe, należy dobrać odpowiednie parametry startowe - punkty początkowe x_0 oraz x_1.

2.3 Implementacja

Moja implementacja algorytmu w języku Julia znajduje się w module `MyModule` znajdującym się w pliku załączonym do tego sprawozdania.

3 Zadanie 3

3.1 Opis problemu

Zadanie polegało na zaimplementowaniu algorytmu rozwiązującego równanie $f(x) = 0$ METODĄ SIECZNYCH.

3.2 Analiza

Dane: $f, a, b, M, \delta, \epsilon$
Wynik: $x, f(x), i, err$
$f_a \leftarrow f(a)$; $f_b \leftarrow f(b)$
do
  $k \leftarrow k + 1$
  if $|f_a| > |f_b|$ **then**
    $a \leftrightarrow b$; $f(a) \leftrightarrow f(b)$
  end
  $s \leftarrow \frac{b-a}{f_b - f_a}$
  $b \leftarrow a$; $f_b \leftarrow f_a$
  $a \leftarrow a - f_a * s$
  if $|b - a| < \delta \vee |f_a| < \epsilon$ **then**
    return $(a, f_a, k, 0)$;
  end
while $k < M$;
return $(a, f_a, M, 1)$;

Metoda siecznych polega na wyznaczaniu miejsca zerowego w oparciu o wznaczanie punktów będących przecięciem pewnych siecznych funkcji. Algorytm zaczyna obliczenia od dwóch punktów początkowych x_0, x_1. Sieczna funkcji, przechodząca przez te punkty, przecina również oś OX. Z punktu przecięcia x_2 oraz punktu x_1 wyznaczana jest kolejna sieczna. Generowane sieczne przecinają oś odciętych w miejscach zbliżających się do szukanego miejsca zerowego funkcji. Operacja jest powtarzana aż do momentu osiągnięcia oczekiwanej dokładności. Metoda ta jest lokalnie zbieżna.

3.3 Implementacja

Moja implementacja algorytmu w języku Julia znajduje się w module `MyModule` znajdującym się w pliku załączonym do tego sprawozdania.

4 Zadanie 4

4.1 Opis problemu

Zadanie polegało na wyznaczeniu pierwiastka równania $\sin x - (\frac{1}{2}x)^2 = 0$ przy pomocy wcześniej zaprogramowanych metod, z parametrami odpowiednio:

1. dla metody bisekcji z przedziałem $[1.5, 2]$, $\delta = \frac{1}{2} * 10^{-5}$, $\epsilon = \frac{1}{2} * 10^{-5}$
2. dla metody stycznych z $x_0 = 1.5$, $\delta = \frac{1}{2} * 10^{-5}$, $\epsilon = \frac{1}{2} * 10^{-5}$
3. dla metody siecznych z $x_0 = 1$, $x_1 = 2$, $\delta = \frac{1}{2} * 10^{-5}$, $\epsilon = \frac{1}{2} * 10^{-5}$

4.2 Rozwiązanie

Wykorzystując funkcje zaimplementowane w module `MyModule` wyliczyłem pierwiastki za pomocą tych trzech algorytmów.

4.3 Wynik

Wyniki zestawiłem w tabeli poniżej:

Metoda	x	$f(x)$	i	err
Bisekcja	1.9337539672851562	-2.7027680138402843e-7	16	0
Stycznych	1.933753779789742	-2.2423316314856834e-8	4	0
Siecznych	1.933753644474301	1.564525129449379e-7	4	0

Rozwiązując zadanie, udało się wyznaczyć jedno z dwóch miejsc zerowych zadanej funkcji. Porównując wyniki, każda z metod doprowadziła do oczekiwanych rezultatów. Okazało się, że metoda bisekcji wykonała aż 4 razy więcej iteracji.

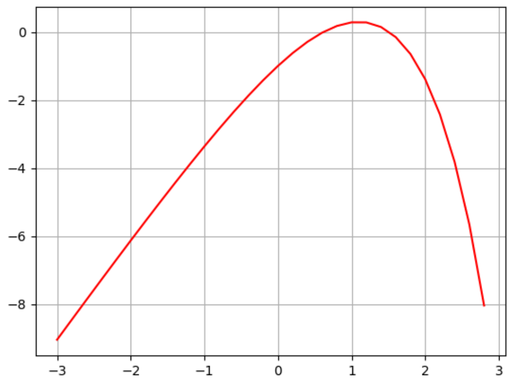
5 Zadanie 5

5.1 Opis problemu

Zadanie polegało na znalazieniu wartości zmiennej x, dla której przecinają się wykresy funkcji $f(x) = 3x$ i $g(x) = e^x$. Wymagana dokładność obliczeń to $\delta = 10^{-4}$, $\epsilon = 10^{-4}$

5.2 Rozwiązanie

W celu wyznaczenia takiego punktu, tj. pary (x, y), zestawiłem ze sobą te funkcje, tj. $f(x) = g(x) \Rightarrow 3x = e^x \Rightarrow 3x - e^x = 0$ i przeanalizowalem nowopowstałą funkcję, tj. $h(x) = 3x - e^x$. Za pomocą biblioteki matplotlib wygenerowałem wykres tej funkcji. Z analizy wykresu doszedłem do tego, że funkcja ta ma dwa miejsca zerowe, jedno wśród argumentów z zakresu $x \in (0.0, 1.0)$ oraz drugie z zakresu $x \in (1.0, 2.0)$. W związku z zaobserowanymi własnościami funkcji $h(x)$ za pomocą funkcji `mbisekcji` odnalazłem miejsca zerowe z dokładnością wymaganą w treści zadania.



Rysunek 1: Wykres $g(x) = 3x - e^x$

5.3 Wynik

Wyniki zestawiłem w tabeli poniżej:

Przedział	x	$f(x)$	i	err
$(0.0, 1.0)$	0.619140625	9.066320343276146e-5	9	0
$(0.1, 2.0)$	1.5120849609375	7.618578602741621e-5	13	0

Porównując (w pakiecie matematycznym) wyniki uzyskane za pomocą metody bisekcji z rzeczywistymi wartościami, możemy stwierdzić, że wartości te są poprawne (z uwzględnieniem błędu). W metodzie bisekcji bardzo ważnym jest, aby odpowiednio dobrać przedziały (warunki początkowe). Próby eksperymentalnego znalezienia miejsc zerowych na przedziale $\langle 0, 2 \rangle$ nie doprowadziły do poprawnego wyniku. (nie spełniał on założenia o różnych znakach wartości na krańcach przedziału)

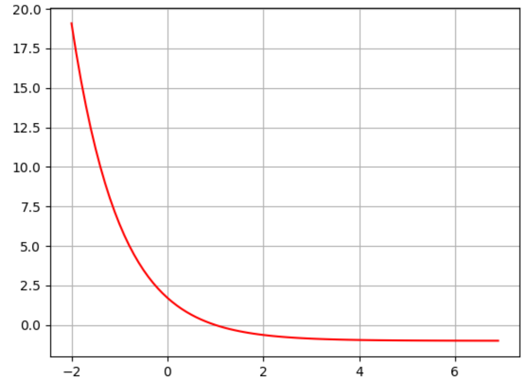
6 Zadanie 6

6.1 Opis problemu

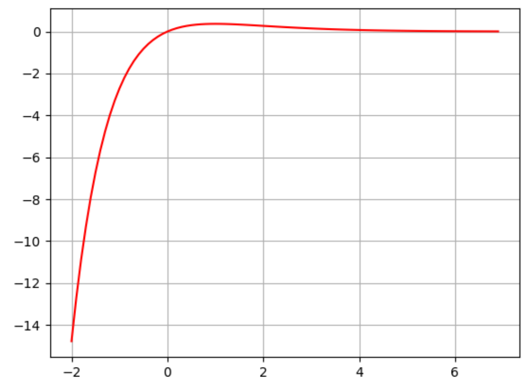
Celem tego zadania było znalezienie miejsc zerowych funkcji $f_1(x) = e^{1-x} - 1$ oraz $f_2(x) = xe^{-x}$ za pomocą wcześniej zaimplementowanych metod przy dokładności $\delta = 10^{-5}$, $\epsilon = 10^{-5}$. Dobrać odpowiednio przedział i przybliżenie początkowe.

6.2 Rozwiązanie

Na samym początku, dla ułatwienia zadania, za pomocą biblioteki `matplotlib` wygenerowałem wykresy funkcji $f_1(x)$ oraz $f_2(x)$.



Rysunek 2: Wykres $f_1(x) = e^{1-x} - 1$



Rysunek 3: Wykres $f_2(x) = xe^{-x}$

Z analizy wykresów (Rysunek 2, Rysunek 3) zaobserwowałem, że poszukiwane miejsca zerowe znajduje się w przypadku $f_1(x)$ przedziale $x \in [0.0, 2.0]$, a w przypadku $f_2(x)$ przedziale $x \in [-1.0, 1.0]$. Krańce tych przedziałów przyjąłem za punkty początkowe w przypadku obliczeń metodą siecznych, tych przedziałów użyłem również do wyliczenia wyniku metodą bisekcji. Za punkt początkowy, w przypadku metody stycznych, przyjąłem -0.5.

6.3 Wynik

Wyniki zestawiłem w tabelach poniżej:

Dla $f_1(x) = e^{1-x} - 1$:

Metoda	x	$f(x)$	i	err
Bisekcja	1.0	0.0	1	0
Stycznych	0.9999922654776594	7.734552252003368e-6	5	0
Siecznych	1.0000017597132702	-1.7597117218937086e-6	6	0

Dla $f_2(x) = xe^{-x}$:

Metoda	x	$f(x)$	i	err
Bisekcja	0.0	0.0	1	0
Stycznych	-3.0642493416461764e-7	-3.0642502806087233e-7	4	0
Siecznych	1.744165849924562e-8	1.7441658195034172e-8	18	0

Podczas wyliczania miejsc zerowych funkcji przedstawionych w treści zadania zauważyłem, że wybór punktów w metodzie bisekcji zawsze doprowadzi nas do rozwiązania. Oczywiście wybranie punktów startowych odległych od miejsca zerowego o $x_{00} = x_0 - \delta$ oraz $x_{01} = x_0 + \delta$ w łatwy sposób pozwoli trafić w miejse zerowe. Przykładowe wyniki dla bisekcji dla lekko przesuniętych miejsc zerowych to odpowiednio
dla $f_1 \rightarrow (\mathtt{0.999993896484375}, \mathtt{6.1035342515669555e} - 6, 16, 0)$
dla $f_2 \rightarrow (-\mathtt{3.051757812455591e} - 6, -\mathtt{3.051767125695548e} - 6, 17, 0)$
W kwestii zbieżności - globalna zbieżność odróżnia metodę bisekcji od reszty. W przypadku pozostałych dwóch - metody stycznych oraz metody siecznych, dobór punktów startowych jest istotnym elementem. Metody te są jedynie lokalnie zbieżne, a więc złe punkty początkowe nie doprowadzą do dobrego wyniku. Przykładem źle dobranych punktów startowych są zestawy parametrów startowych przedstawione w kolejnej sekcji.

6.4 Dobór parametrów

Dodatkowym punktem zadania było sprawdzenie zachowania metody Newtona dla pewnych parametrów:

1. dla f_1 gdy $x_0 > 1$
2. dla f_2 gdy $x_0 > 0$
3. dla f_2 gdy $x_0 = 1$

Analiza wywołań:

1. W tym wypadku wartości zwracane przez metodę były akceptowalne do $x_0 = 7.4$. W kolejnych iteracjach testu $x_0 \in [7.6, 12.4]$ pojawił się błąd $err = 1$ (nieosiągnięto dokładności po `maxit` iteracjach), a wartości zwracane przez metodę to `NaN`. Powodem pojawienia się `NaN` tj. (not a number) był fakt, że w pewnym momencie pochodna, bliska zeru, przechodziła przez test $|f'(x0)| < \epsilon$, ale w momencie, gdy algorytm wyliczał $x_1 = x_0 - \frac{v}{f'(x0)}$, to dochodziło do dzielenia przez zero. Dalsze iteracje zwracały błąd $err = 2$ świadczący o pochodnej bliskiej zeru - wtedy warunek $|f'(x0)| < \epsilon$ zwracał `true` i metoda kończyła działanie.
2. Sprawdzając kolejne wywołania metody Netwona dla parametrów $x \in (0, \infty)$ zauważyłem, że do osiągnięcia $x = 1.0$, oscylowały w granicach realnej wartości. Przy $x = 1.0$ funkcja zwróciła $\mathtt{err} = \mathtt{2}$, kolejne obliczenia, dla $x_0 > 1.0$ zwracały wartości znacznie odbiegające od realnej.
3. Po wywołaniu metody Newtona na f_2 z zadanymi paramterami otrzymujemy $(\mathtt{1.0}, -\mathtt{0.0}, \mathtt{0}, \mathtt{2})$. Takie zachowanie (będące odstępstwem od punktów w otoczeniu tego punktu) związane jest z pochodną tej funkcji, tj. $f_2'(x) = -e^{-x}(x-1)$, która w $x_0 = 1$ osiąga swoje miejsce zerowe, a więc styczna $g(x)||OX$, tj. nie spełnia warunków metody Netwona.

Powyższe zestawy pokazują, że metody Netwona i siecznych są jedynie zbieżne lokalnie, a źle dobrane parametry początkowe uniemożliwiają osiągnięcie rozsądnych wyników.