Obliczenia naukowe - Lista nr 3

Jakub Jaśków

November 18, 2023

1 Metoda bisekcji

Opis

Napisanie funkcji znajdującej pierwiastki równania **metodą bisekcji**.

Rozwiązanie

Opis algorytmu

Metoda bisekcji stosowana jest do przybliżonego rozwiązywania równania $f(x) = 0$ w przedziale $[a, b]$. Funkcja $f(x)$ musi być określona i ciągła w tym przedziale oraz musi posiadać różne znaki na krańcach przedziału $[a, b]$, co gwarantuje istnienie przynajmniej jednego pierwiastka w tym przedziale. Rozwiązanie znajdowane jest za pomocą kolejnych przybliżeń. Z tego powodu należy określić dokładność, z którą chcemy otrzymać pierwiastek funkcji oraz dokładność wyznaczania samej funkcji.

W każdym przybliżeniu algorytm wyznacza środek x_0 przedziału $[a, b]$ jako średnią arytmetyczną krańców. Następnie sprawdzane jest, czy odległość tego środka od krańców przedziału jest mniejsza od założonej dokładności wyliczania pierwiastka. Jeśli tak, to algorytm kończy pracę z wynikiem w x_0. Jeśli nie, to wyznaczana jest wartość funkcji w punkcie x_0 i sprawdza się, czy odległość tej wartości od 0 jest mniejsza od założonej dokładności wyznaczania funkcji. Jeśli tak, to algorytm kończy pracę z wynikiem w x_0.

W przeciwnym razie punkt x_0 dzieli przedział $[a, b]$ na dwie równe połowy: $[a, x_0]$ i $[x_0, b]$. Algorytm za nowy przedział $[a, b]$ przyjmuję tę połówkę, w której funkcja zmienia znak na krańcach i kontynuuje wyznaczanie pierwiastka funkcji.

Z opisu wynika, iż w każdym obiegu szerokość przedziału maleje dwukrotnie. Dzięki temu pierwiastek jest wyznaczany coraz bardziej dokładnie. Po dziesięciu obiegach szerokość przedziału maleje $210 = 1024$ razy.

Algorytm

```
mbisekcji(f, a, b, delta, epsilon)
    u ← f(a)
    v ← f(b)
    e ← b − a
    it ← 0
    if sign(u) = sign(v) then
        return err 1
    end
    while e > epsilon do
        it ← it + 1
        e ← e/2
        c ← a + e
        w ← f(c)
        if |e| < delta or |w| < epsilon then
            return c, w, it, 0
        end
        if sign(w) ≠ sign(u) then
            b ← c
            v ← w
        end
        else
            a ← c
            u ← w
        end
    end
```

Algorithm 1: Metoda bisekcji

Dane:

`f`	–	funkcja $f(x)$ zadana jako anonimowa funkcja (ang. anonymous function),
`a, b`	–	końce przedziału początkowego,
`delta, epsilon`	–	dokładności obliczeń,

Wyniki:

`(r,v,it,err)`	–	czwórka, gdzie
`r`	–	przybliżenie pierwiastka równania $f(x) = 0$,
`v`	–	wartość $f(r)$,
`it`	–	liczba wykonanych iteracji,
`err`	–	sygnalizacja błędu
		0 - brak błędu
		1 - funkcja nie zmienia znaku w przedziale [a,b]

Notatki

1. Punkt środkowy c obliczamy poprzez $c \leftarrow a + (b - a)/2$. Instrukcja $c \leftarrow (a + b)/2$ mogłaby spowodować, że w ekstremalnych przypadkach $c \notin [a, b]$.

2. Pozbywamy się mnożenia przy $f(a)f(c) < 0$, następstwem którego mógł by być nadmiar lub niedomiar, instrukcją $sign(w) \neq sign(u)$.

3. Uwzględniamy trzy możliwości zakończenia algorytmu:
$e > epsilon$ mówi kiedy jest możliwa iteracja w danym przedziale.
Błąd jest dostatecznie mały ($abs(e) < delta$).
$f(c)$ jest dostatecznie bliskie zeru ($abs(w) < epsilon$).

2 Metoda Newtona

Opis problemu

Napisanie funkcji znajdującej pierwiastki równania **metodą Newtona**.

Rozwiązanie

Opis algorytmu

Metoda **Newtona** (stycznych) wyszukująca pierwiastki funkcji opiera się na *linearyzacji funkcji*, czyli zastąpieniu $f(x)$ funkcją liniową, będącą sumą dwóch początkowych składników we wzorze Taylora dla f. Zbieżność metody stycznych jest kwadratowa, co czyni ją (średnio)szybszą od metod **bisekcji** czy siecznych. Gdy przybliżenia otrzymane za pomocą metody **Newtona** stają się bliskie pierwiastka, staje się ona na tyle szybko zbieżna, że kilka przybliżeń pozwala osiągnąć maksymalną dokładność. Metoda stycznych nie zawsze jest zbieżna, więc na ogół stosuje się ją w razem z innymi metodami zbieżnymi globalnie. Kłopotliwa jest także konieczność obliczenia pochodnej, co może sprawiać problemy dla niektórych funkcji.

Algorytm

mstycznych(f, p_f, x_0, *delta*, *epsilon*, *maxit*)
- $v \leftarrow f(x_0)$
- **if** $|v| < epsilon$ **then**
  - **return** $x_0, v, 0, 0$
- **end**
- **if** $|p_f(x_0)| < epsilon$ **then**
  - **return** err 2
- **end**
- **for** $it \leftarrow 1$ **to** $maxit$ **do**
  - $x_1 \leftarrow x_0 - (v/p_f(x_0))$
  - $v \leftarrow f(x_1)$
  - **if** $|x_1 - x_0| < delta$ ***or*** $|v| < epsilon$ **then**
    - **return** $x_1, v, it, 0$
  - **end**
  - $x_0 \leftarrow x_1$
- **end**
- **return** err 1

Algorithm 2: Metoda Newtona

Dane:

`f, pf`	–	funkcja $f(x)$ oraz pochodna $f'(x)$ zadane jako anonimowe funkcje,
`x0`	–	przybliżenie początkowe,
`delta, epsilon`	–	dokładności obliczeń,
`maxit`	–	maksymalna dopuszczalna liczba iteracji,

Wyniki:

`(r,v,it,err)`	–	czwórka, gdzie
`r`	–	przybliżenie pierwiastka równania $f(x) = 0$,
`v`	–	wartość $f(r)$,
`it`	–	liczba wykonanych iteracji,
`err`	–	sygnalizacja błędu 0 - metoda zbieżna 1 - nie osiągnięto wymaganej dokładności w `maxit` iteracji 2 - pochodna bliska zeru

Notatki

Na początku algorytmu sprawdzamy:
1. Wartość funkcji dla przybliżenia początkowego jest wystarczająco bliska zeru - zwracany jest wynik.
2. Pochodna jest bliska zeru - niemożliwe jest wtedy zastosowanie metody.

Następnie wyznaczane są kolejne przybliżenia zera funkcji, które są dane rekurencyjnym wzorem

$$x_{n+1} := x_n - \frac{f(x_n)}{f'(x_n)}$$

. Algorytm kończy się kiedy znaleziony pierwiastek mieści się w dokładności, lub odległości pomiędzy przybliżeniami są wystarczająco niewielkie. Błąd zwracamy jeżeli żadne z powyższych warunków nie zostanie spełniony.

3 Metoda siecznych

Opis

Napisanie funkcji wyznaczającej pierwiastki równania $f(x) = 0$ metodą **siecznych**.

Rozwiązanie

Opis algorytmu

Metoda **siecznych** powstała na skutek próby eliminacji obliczania pochodnej dla metody Newtona. Aby tego dokonać $f'(x)$ zastąpiono ilorazem różnicowym:

$$f'(x) \approx \frac{f(x_n) - f(x_{n-1})}{x_n - x_{n-1}}$$

.

Równość ta, wynikająca z definicji pochodnej, skutkuje właśnie metodą siecznych

$$x_{n+1} := x_n - f(x_n)\frac{x_n - x_{n-1}}{f(x_n) - f(x_{n-1})}$$

$x_{n+1} = x_n + x_{n-1}$ więc potrzebujemy n_0 oraz n_1, jednak każde nowe x_{n+1} wymaga jednej wartości funkcji f. Metoda **siecznych** zbiega wolniej niż metoda Newtona, natomiast każdy jej krok wymaga obliczenia tylko **jednej** wartości funkcji, zamiast dwóch - $f(x)$ i $f'(x)$.

Algorytm

`msiecznych`(*f*, x_0, x_1, *delta*, *epsilon*, *maxit*)
    $f_{x_0} \leftarrow f(x_0)$
    $f_{x_1} \leftarrow f(x_1)$
    for $it \leftarrow 1$ **to** $maxit$ **do**
        if $|f_{x_0}| > |f_{x_1}|$ **then**
            $x_0 \leftrightarrow x_1$
            $f_{x_0} \leftrightarrow f_{x_1}$
        end
        $s \leftarrow (x_1 - x_0)/(f_{x_1} - f_{x_0})$
        $x_1 \leftarrow x_0$
        $f_{x_1} \leftarrow f_{x_0}$
        $x_0 \leftarrow x_0 - (f_{x_0} \times s)$
        $f_{x_0} \leftarrow f(x_0)$
        if $|x_1 - x_0| <$ *delta* ***or*** $|f_{x_0}| <$ *epsilon* **then**
            return $x_0, f_{x_0}, it, 0$
        end
    end
    return *err* 1

Algorithm 3: Metoda siecznych

Dane:

`f`	–	funkcja $f(x)$ zadana jako anonimowa funkcja,
`x0, x1`	–	przybliżenia początkowe,
`delta, epsilon`	–	dokładności obliczeń,
`maxit`	–	maksymalna dopuszczalna liczba iteracji,

Wyniki:

`(r,v,it,err)`	–	czwórka, gdzie
`r`	–	przybliżenie pierwiastka równania $f(x) = 0$,
`v`	–	wartość $f(r)$,
`it`	–	liczba wykonanych iteracji,

`err`	–	sygnalizacja błędu
		0 - metoda zbieżna
		1 - nie osiągnięto wymaganej dokładności w `maxit` iteracji

Algorytm metody przedstawionej powyżej przestawia wartości x_0 i x_1, kiedy wymaga tego utrzymanie nierówności $|f_{x_0}| \leq |f_{x_1}|$. Algorytm kończy się kiedy znaleziony pierwiastek mieści się w dokładności, lub odległości pomiędzy przybliżeniami są wystarczająco niewielkie. Błąd zwracamy jeżeli żadne z powyższych warunków nie zostanie spełniony.

4 zad

Opis

Wyznaczanie pierwiastka równania $\sin x = -(\frac{1}{2}x)^2$ z użyciem metod:
1. bisekcji (na przedziale początkowym $[1.5, 2.0]$,
2. Newtona (dla przybliżenia początkowego $x_0 = 1.5$,
3. siecznych (dla przybliżeń początkowych $x_0 = 1.0, x_1 = 2.0$.

Gdzie $\delta = \frac{1}{2}10^{-5}$, $\epsilon = \frac{1}{2}10^{-5}$.

Rozwiązanie

Podstawienie funkcji $f(x) = \sin x - (\frac{1}{2}x)^2$, jej pochodnej $f'(x) = \cos x - \frac{1}{2}x$ do metod zaimplementowanych w zadaniach 1-3. $maxIt = 32$.

Wyniki

Metoda	Miejsce zerowe (x_0)	Wartość funkcji ($f(x_0)$)	Liczba iteracji	Błąd
Bisekcji	1.9337539672851562	-2.7027680138402843e-7	16	0
Newtona	1.933753779789742	-2.2423316314856834e-8	4	0
Siecznych	1.9337539405015145	-2.3487103129049558e-7	5	0

Table 4: Pierwiastki równania $f(x) = \sin x - (\frac{1}{2}x)^2$ obliczone przy pomocy implementacji metod z zadań 1-3.

Wnioski

Na tabeli powyżej bardzo dobrze widać różnicę w liczbie iteracji potrzebnych do wyznaczenia pierwiastków równania. Wartości te zgadzają się z naszymi przewidywaniami, bowiem metoda bisekcji posiada zbieżność liniową, metoda Newtona kwadratową natomiast metoda siecznych ok ≈ 1.62 (nadliniowo). Metoda bisekcji nie jest, jak mogło by się zdawać, najwolniejsza, ale "najstabilniejsza". Znalazła ona wartość najbliższą zeru, oraz jest stabilna globalnie. Dla danych podanych w zadaniu metoda Newtona oraz metoda siecznych zbiegają szybciej niż metoda bisekcji, lecz dla innej funkcji lub innych przedziałów obserwacje te mogą się różnić.

5 zad

Opis

Przy użyciu metody bisekcji znajdź wartość x, dla której przecinają się wykresy funkcji $y = 3x$ oraz $y = e^x$.

Rozwiązanie

Znalezienia miejsc zerowych funkcji $f(x) = e^x - 3x$ przy użyciu metody bisekcji stworzonej w zadaniu pierwszym.
Dokładności:
$\delta = 10^{-4}$
$\epsilon = 10^{-4}$
Wybrane przedziały: $[0, 1]$, $[1, 2]$.

Wyniki

Punkty przecięcia:
0.619140625 dla przedziału $[0, 1]$,
1.5120849609375 dla przedziału $[1, 2]$.

	x_0	x_1
Przedział	$[0, 1]$	$[1, 2]$
Wartość	0.619140625	1.5120849609375
Niedokładność $\lvert f(x_i) \rvert$	$9.066320343276146 \times 10^{-5}$	$7.618578602741621 \times 10^{-5}$
Liczba iteracji	9	13

Table 5: Miejsca zerowe funkcji $f(x) = e^x - 3x$ obliczone za pomocą metody bisekcji.

Wnioski

Wiedza z analizy matematycznej jest niezbędna przy doborze parametrów początkowych do poszczególnych metod szukania pierwiastków równania. Jeśli spełnimy wszystkie jej założenia, metoda bisekcji gwarantuje nam znalezienie "jakiegoś" pierwiastka równania. Jeżeli natomiast zależy nam na znalezieniu wszystkich rozwiązań, musimy skorzystać z naszej wiedzy na temat analizy matematycznej. Analiza samego problemu pomoże nam dobrać odpowiedni przedział początkowy, co pozwoli na znaczne przyśpieszenie obliczeń w przypadku zastosowania metody stycznych czy metody siecznych.

6 zad

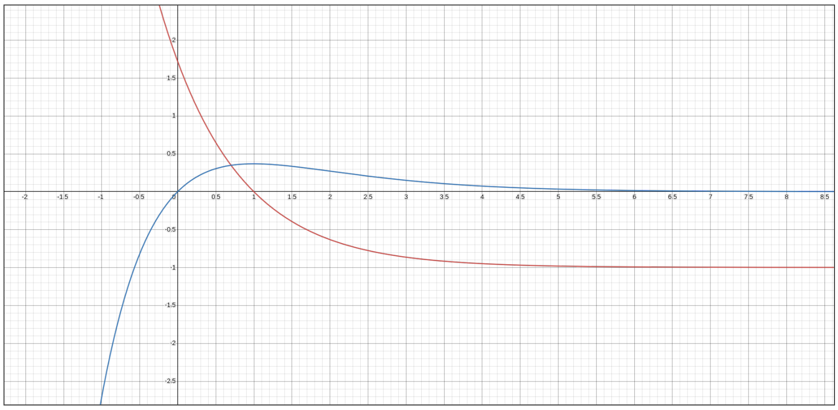
Opis

Znajdź pierwiastki funkcji $f_1(x) = e^{1-x} - 1$ i $f_2(x) = xe^{-x}$ przy pomocy metod bisekcji, stycznych oraz siecznych z dokładnością: $\delta = 10^{-5}$, $\epsilon = 10^{-5}$.

Rozwiązanie

Zaimplementuj wymienione wyżej funkcje (przy użyciu metod z zadań 1-3) oraz oblicz ich pochodne. Przeprowadzona została analiza funkcji f_1 i f_2, które zostały przedstawione na wykresie 1, na podstawie której dobrano parametry początkowe.

Desmos | Graphing Calculator https://www.desmos.com/calculator



1 of 2 11/18/23, 12:29

Figure 1: Wykresy funkcji $f(x) = e^{1-x} - 1$ oraz $g(x) = xe^{-x}$ wykonane w kalkulatorze graficznym Desmos

Wyniki

Łatwo zauważyć, że poprawnymi rozwiązaniami dla f_1 i f_2 jest odpowiednio -1 i 0.

Przedział	r	$f(r)$	Liczba iteracji
		f_1	
$[0.0, 1.5]$	1.0000076293945312	-7.6293654275305656e-6	16
$[0.5, 3.0]$	0.9999923706054688	7.629423635080457e-6	16
$[-4.0, 4.0]$	1.0	0.0	3
$[0.0, 100.0]$	0.9999990463256836	9.536747711536009e-7	22
$[-10.0, 2000.0]$	1.0000018030405045	-1.803038878978036e-6	27
		f_2	
$[-0.5, 1.0]$	-7.62939453125e-6	-7.629452739132958e-6	16
$[-0.25, 1.5]$	-7.62939453125e-6	-7.629452739132958e-6	15
$[-1.0, 6.0]$	-3.814697265625e-6	-3.814711817567984e-6	18
$[-1.5, 100.0]$	49.25	2.010958004139294e-20	1
$[-5.0, 1000.0]$	497.5	4.318056675122884e-214	1

Table 6: Miejsca zerowe f_1 i f_2 obliczone za pomocą metody bisekcji.

x_0	r	$f(r)$	Liczba iteracji	Błąd
		f_1		
-1.0	0.9999922654776594	7.734552252003368e-6	5	0
0.0	0.9999984358892101	1.5641120130194253e-6	4	0
1.0	1.0	0.0	0	0
2.0	0.9999999810061002	1.8993900008368314e-8	5	0
5.0	0.9999996427095682	3.572904956339329e-7	54	0
7.0	0.9999999484165362	5.15834650549607e-8	401	0
8.0	—	—	—	1
13.0	—	—	—	2
		f_2		
-2.0	-1.425500682806244e-9	-1.425500684838296e-9	7	0
-1.0	-3.0642493416461764e-7	-3.0642502806087233e-7	5	0
0.0	0.0	0.0	0	0
0.5	-3.0642493416461764e-7	-3.0642502806087233e-7	5	0
1.0	—	—	—	2
2.0	14.398662765680003	8.036415344217211e-6	10	0
5.0	15.19428398343915	3.827247505782987e-6	9	0
100.0	100.0	3.7200759760208363e-42	0	0

Table 7: Miejsca zerowe f_1 i f_2 obliczone za pomocą metody stycznych.

x_0	x_1	r	$f(r)$	Liczba iteracji	Błąd
			f_1		
-1.0	2.0	1.0000009310146594	-9.310142259355558e-7	7	0
0.5	3.0	0.9999998801054126	1.1989459447470097e-7	6	0
-3.0	4.0	0.9999924734799833	7.526548341019179e-6	1500	0
-2.0	6.0	5.229263398675002	-0.9854368861925255	5	0
10.0	100.0	—	—	—	1
			f_2		
-1.0	0.5	3.201418966654486e-7	3.2014179417463104e-7	7	0
-0.25	1.5	5.662892187393383e-7	5.662888980559498e-7	7	0
2.0	6.0	14.386737398698989	8.12609044213971e-6	12	0
10.0	20.0	20.00090808104888	4.118752554492817e-8	1	0
-2.0	100.0	100.0	3.7200759760208363e-42	1	0

Table 8: Miejsca zerowe f_1 i f_2 obliczone za pomocą metody siecznych.

Wnioski

Po przeanalizowaniu wyników z tabeli 6 możemy stwierdzić, że metoda **bisekcji**:
1. f_1 - metoda ta zbiega do zera nawet dla bardzo dużych przedziałów, jeżeli przyjjmiemy wystarczającą liczbę iteracji.
2. f_2 - należy uważać na dobór przedziałów nawet stosując metodę bisekcji, ponieważ f_1 osiąga zero w nieskończoności. Przyjęcie dużego przedziału początkowego skutkować może zakończeniem obliczeń po jednej iteracji, gdyż wartość będzie dostatecznie bliska wartości zera, natomiast odległa od rzeczywistego pierwiastka f_2.

Po przeanalizowaniu wyników z tabeli 7 możemy stwierdzić, że metoda **Newtona**:
1. f_1 - dla wartości oddalonych od zera liczba iteracji rośnie, co skutkuje rozbieżnością metody dla dużej liczb iteracji. Od pewnego momentu funkcja f_1, bardzo wolno maleje, więc jej pochodna osiąga wartość bliską zeru, co czyni zastosowanie metody Newtona niemożliwym.
2. f_2 - pochodna zeruje się dla $x_0 = 1$. Dla wszystkich wartości przybliżenia początkowego, które są większe od jedynki, metoda nie zbiega do faktycznego zera funkcji, ale znajduje wartość dostatecznie bliską zeru. Dzieje się tak, bowiem $lim_{x \to \infty} f_2(x) = 0$.

Po przeanalizowaniu wyników z tabeli 8 możemy stwierdzić, że metoda **siecznych**:
1. f_1 - dla przybliżeń początkowych oddalonych od zera możemy obserwować rozbieżność(jak w metodzie **Newtona**).
Ciekawym wynikiem jest wartość uzyskana dla $x_0 = -2$ i $x_1 = 6$. Metoda nie znajduje wartości funkcji bliskiej zeru, ale w pewnym momencie warunek $|x_1 - x_0| < delta$ staje się prawdą, kończąc działanie algorytmu. Dzieje się tak, ponieważ różnica między wartościami funkcji f_1 w $x_0 = -2$ i $x_1 = 6$ jest bardzo duża i kolejne przybliżenie wyznaczane jest bardzo blisko x_1.
2. f_2 - analogicznie do metody **Newtona**.