Szymon Ociepka *Data wykonania: 17.04.2024*

Bisekcja, Newton-Raphson

Metody numeryczne

1. Wprowadzenie teoretyczne

1.1 metoda Bisekcji

Teoria

Metoda bisekcji, znana również jako metoda podziału na pół, to jedna z najprostszych i najbardziej niezawodnych metod numerycznych stosowanych do rozwiązywania równań nieliniowych.

Podstawowa idea metody bisekcji polega na podziale danego przedziału na dwie równe części (stąd nazwa “bisekcja”, co oznacza “podział na dwie części”). Jeżeli funkcja zmienia znak na danym przedziale, to znaczy, że istnieje tam przynajmniej jedno miejsce zerowe. Metoda bisekcji polega na wyborze punktu środkowego przedziału i sprawdzeniu, w której połowie przedziału funkcja zmienia znak. Następnie proces jest powtarzany dla wybranej połowy przedziału, aż do osiągnięcia wymaganej precyzji.

Jest to metoda iteracyjna, co oznacza, że generuje sekwencję przybliżeń, które zbiegają do prawdziwego rozwiązania. Mimo że metoda bisekcji jest stosunkowo wolna w porównaniu z innymi metodami, takimi jak metoda Newtona czy metoda siecznych, ma tę zaletę, że zawsze zbiega do rozwiązania, pod warunkiem, że początkowy przedział zawiera miejsce zerowe.

Zastosowanie

Aby można było zastosować metodę bisekcji, muszą być spełnione następujące założenia:

- funkcja $f(x)$ jest ciągła w analizowanym przedziale [a, b]
- $f(a)\cdot f(b)<0$ (funkcja przyjmuje różne znaki na końcach przedziału), (1)
- wewnątrz [a, b] znajduje się dokładnie jeden pierwiastek.

Sprawdzamy, czy pierwiastkiem równania jest wybrany punkt $x_1=\frac{a+b}{2}$. (2)

Jeżeli $f(x_1)=0$, algorytm kończy działanie, a punkt x_1 jest szukanym miejscem zerowym.

Jeżeli nie, dopóki nie osiągniemy żądanej dokładności $|a-b|>zadana\,dokładność$, dzielimy przedział [a,b] na dwa mniejsze przedziały: [a,x_1] oraz [x_1,b].

- jeżeli $f(x_1)\cdot f(a)<0$, to b=x_1, (3)
- jeżeli $f(x_1)\cdot f(b)<0$, to a=x_1. (4)

Algorytm kończy działanie po osiągnięciu zadanej dokładności albo po wykonaniu założonej liczby kroków. A poszukiwany pierwiastek równania wynosi $\frac{a+b}{2}$.

1.2 metoda Newtona-Raphsona

Teoria

Metoda Newtona-Raphsona, znana również jako metoda stycznych, to popularna metoda numeryczna używana do szukania pierwiastków równań nieliniowych. Została nazwana na cześć dwóch matematyków: Sir Isaaca Newtona i Josepha Raphsona.

Podstawowym pomysłem metody Newtona-Raphsona jest użycie stycznej do funkcji w danym punkcie jako przybliżenia funkcji w pobliżu tego punktu. Metoda ta jest iteracyjna, co oznacza, że zaczynamy od początkowego przybliżenia, a następnie powtarzamy pewien proces, aby uzyskać coraz to lepsze przybliżenia pierwiastka.

Metoda Newtona-Raphsona jest szybka i efektywna, ale ma też pewne ograniczenia. Na przykład, może nie zbiegać do pierwiastka, jeśli początkowe przybliżenie jest źle wybrane. Mimo to, jest to podstawowe narzędzie w numerycznym rozwiązywaniu równań nieliniowych.

Zastosowanie

Aby można było zastosować metodę Newtona-Raphsona, muszą być spełnione następujące założenia:

- pierwsza i druga pochodna f mają stały znak w przedziale [a, b]
- różne znaki na krańcach przedziału $f(a)\cdot f(b)<0$, (1)
- wewnątrz [a, b] znajduje się dokładnie jeden pierwiastek.

Wybieramy punkt startowy x_1 (a, b, 0 lub 1).

Następnie z punktu x_1 wyprowadzana jest styczna w $f(x_1)$. Odcięta punktu przecięcia stycznej z osią OX jest naszym pierwszym przybliżeniem rozwiązania x_2.

Jeżeli otrzymane przybliżenie nie spełnia warunku stopu, wówczas otrzymany punkt x_2 jest nowym punktem startowym i algorytm powtarzany jest ponownie.

Wzór rekurencyjny:

$$x_{n+1}=x_n-\frac{f(x_n)}{f'(x_n)} \tag{5}$$

Kryterium stopu:

1. *wartość funkcji w wyznaczonym punickie jest bliska* $0, |f(x_n)|\leq$ *zadana dokładność*.
2. *odległość pomiędzy kolejnymi przybliżeniami jest wystarczająco mała*, $|x_{n+1}-x_n|\leq$ *zadana dokładność*.

2. Opis implementacji numerycznej

Dla usprawnienia programu zaimplementowano dwie funkcje:

- funkcja f, licząca wartość dla naszego punktu x

- funkcja pf, licząca pochodną funkcji f

Przy testach innych funkcji, funkcje te są odpowiednio modyfikowane. W poniższym przykładzie naszą funkcją $f(x)$ jest funkcja $f(x)=(x-1)^2-3$.

```
double f(double x) {//funkcja pierwotna
    return pow(x - 1, 2) - 3;
}

double pf(double x) {//pochodna
    return 2 * x - 2;
}
```

Rysunek 1.1: wartość funkcji oraz jej pochodna

Metoda bisekcja została zaimplementowana w funkcji o nazwie **bisekcja**, która przyjmuję jako parametry kolejno: punkt x, początek przedziału, koniec przedziału, zadaną dokładność.

```
double bisekcja(double x, double a, double b, double dokladnosc) {

    while (fabs(a - b) > dokladnosc) {
        x = (a + b) / 2;
        if (f(x) * f(a) < 0) b = x;
        if (f(x) * f(b) < 0) a = x;
    }

    return x = (a + b) / 2;;
}
```

Rysunek 1.2: implementacja - metoda bisekcji

Metoda Newtona-Raphsona została zaimplementowana w funkcji nazywającej się `newton_raphson`, która przyjmuję jako parametry kolejno: punkt początkowy (początek przedziału) oraz przyjętą dokładność.

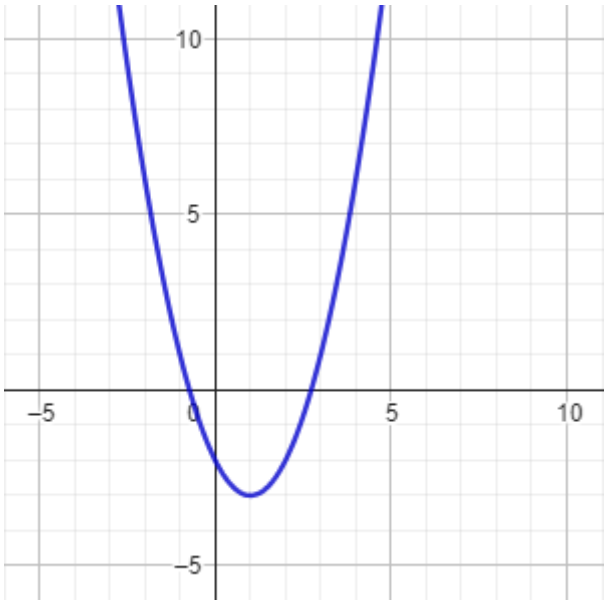
```
double newton_raphson(double x, double precyzja) {
    double x_nastepny = x - f(x) / pf(x);
    while ((fabs(f(x)) > precyzja) && (fabs(x_nastepny - x) > precyzja)) {
        x = x_nastepny;
        x_nastepny = x_nastepny - f(x_nastepny) / pf(x);
    }
    return x_nastepny;
}
```

Rysunek 1.3: implementacja - metoda Newtona-Raphsona

3. Testy kodu numerycznego

Dokładność równa jest 0.0000001

Test I



Rysunek 2.1: wykres funkcji testu I

Funkcja: $f(x)=(x-1)^2-3, x\in[0,9]$

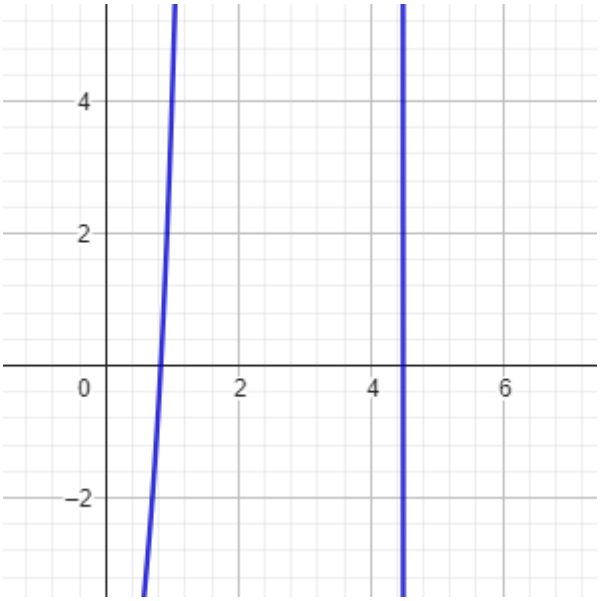
Wynik programu Wolfram Alpha: = 2.7320508

Wynik zaimplementowanych metod:

```
bisekcja x0 = 2.7320508
     N-R x0 = 2.7320508
```

Rysunek 3.1: fragment konsoli testu I

Test II



Rysunek 2.2: wykres funkcji testu II

Funkcja: $f(x)=-2x^6+10x^5-5x^4+2x^3-2x^2+10x-9, x\in[3,5]$

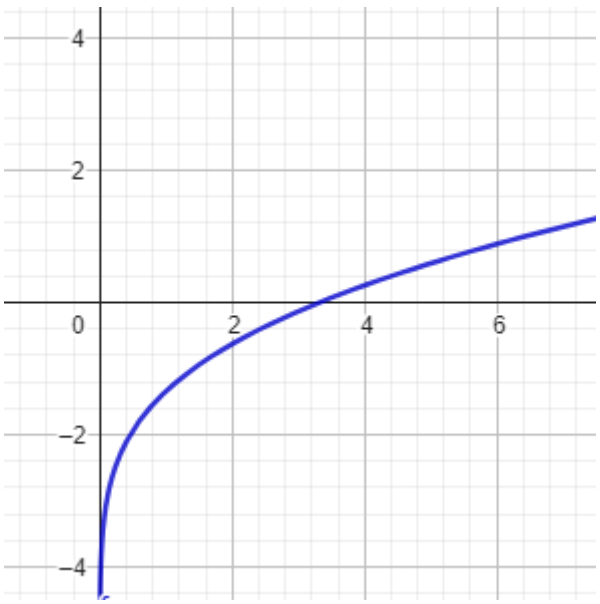
Wynik programu Wolfram Alpha: = 4.4917784

Wynik zaimplementowanych metod:

```
bisekcja x0 = 4.4917787
    N-R x0 = 4.4917787
```

Rysunek 3.2: fragment konsoli testu II

Test III



Rysunek 2.3: wykres funkcji testu III

Funkcja: $f(x)=\log(x)+\sqrt{x}-2.345, x\in[1,6]$

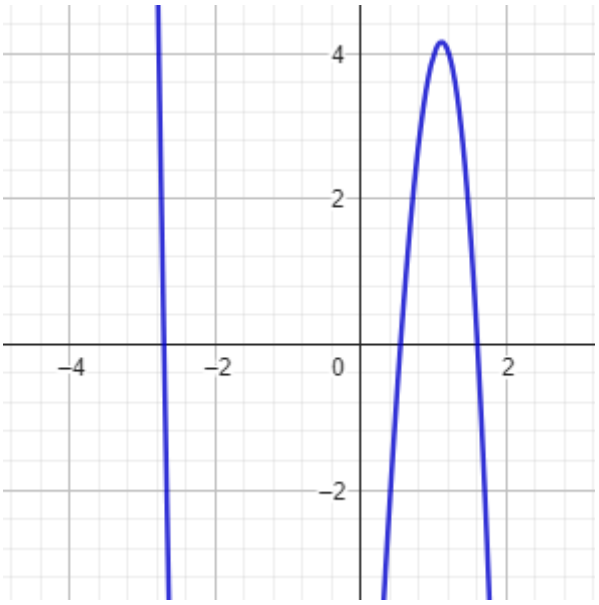
Wynik programu Wolfram Alpha: = 2.2941401

Wynik zaimplementowanych metod:

```
bisekcja x0 = 2.2941401
     N-R x0 = 2.2941401
```

Rysunek 3.3: fragment konsoli testu III

Test IV



Rysunek 2.4: wykres funkcji testu IV

Funkcja: $f(x)=-3.93x^3-2.2x^2+19.357x-9.246, x\in[-4,0.5]$

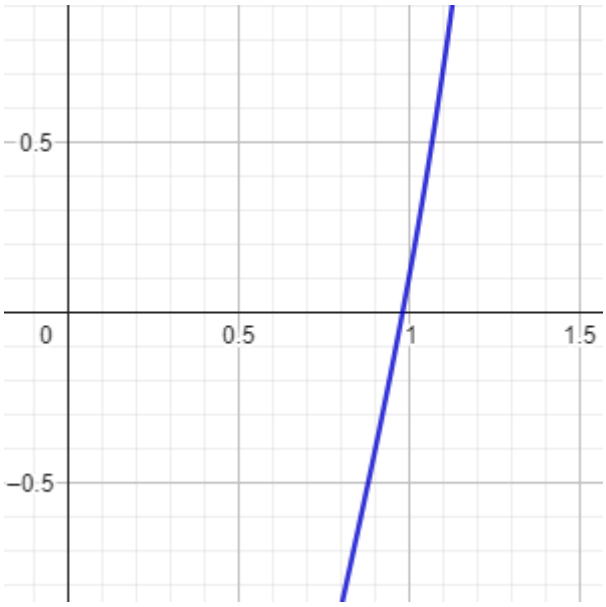
Wynik programu Wolfram Alpha: = -2.7035338

Wynik zaimplementowanych metod:

```
bisekcja x0 = -2.7035338
     N-R x0 = -2.7035338
```

Rysunek 3.4: fragment konsoli testu IV

Test V



Rysunek 2.5: wykres funkcji testu V

Funkcja: $f(x)=1.12\,x^5-\sqrt{x^7}+7.89\log(x), x\in[0.2, 1.6]$

Wynik programu Wolfram Alpha: = 0.9778528

Wynik zaimplementowanych metod:

```
bisekcja x0 = 0.9879481
    N-R x0 = 0.9879482
```

Rysunek 3.5: fragment konsoli testu V

4. Analiza wyników

W celu głębszej analizy powstała tabela, w której porównane zostały wyniki zaimplementowanych metod z wynikami programu „Wolfram Alpha” na podstawie powyższych testów.

Tabela 1: wyniki przeprowadzonych testów

	Test I	**Test II**	**Test III**	**Test IV**	**Test V**
Bisekcja	2.7320508	4.4917787	2.2941401	-2.7035338	0.9879481
Newton-Raphson	2.7320508	4.4917787	2.2941401	-2.7035338	0.9879482
Wolfram Alpha	2.7320508	4.4917784	2.2941401	-2.7035338	0.9778528

Analizując wyniki z tabeli, można powiedzieć, że obie metody numeryczne zostały dobrze zaimplementowane. Zarówno metoda bisekcji, jak i metoda Newtona-Raphsona dobrze radzą sobie z testowanymi funkcjami. Wyniki różnią się głównie w mało istotnych cyfrach po przecinku, co można zaobserwować w teście nr II dla obydwu metod, gdzie różnica między wynikiem z programu a wynikiem poprawnym jest minimalna. To prawdopodobnie wynik zbyt szerokiego zakresu [a,b], jaki został użyty. Z drugiej strony, w Teście V zaimplementowane metody dają znacznie bardziej błędny wynik, co może być spowodowane niewłaściwym wyborem wartości początkowej x_1.

5. Wnioski

Metoda bisekcji jest jedną z najprostszych i najbardziej niezawodnych metod numerycznych do rozwiązywania równań nieliniowych. Mimo że jest to metoda wolno zbieżna, jej zaletą jest gwarancja zbieżności. Metoda Newtona-Raphsona jest metodą szybko zbieżna, która wykorzystuje pochodną funkcji do znalezienia pierwiastka. Metoda ta jest bardziej skomplikowana niż metoda bisekcji, ale oferuje znacznie szybszą zbieżność, szczególnie dla dobrze uwarunkowanych problemów. Wadą tej metody jest to, że nie zawsze jest zbieżna, a jej zbieżność zależy od początkowego przybliżenia. Implementacje obu metod, zostały poprawnie zaimplementowane. Implementacja była dokładna i niezawodna, co pozwoliło na skuteczne i efektywne rozwiązywanie układów równań liniowych. Wszystkie testy przeszły pomyślnie, co potwierdza poprawność implementacji. Podsumowując, zarówno metoda bisekcji, jak i metoda Newtona-Raphsona są efektywnymi narzędziami do rozwiązywania układów równań nieliniowych, przy czym każda z nich ma swoje mocne oraz słabe strony. Główną wadą metody bisekcji jest to, że jest ona stosunkowo wolna w porównaniu do innych metod, takich jak metoda Newtona-Raphsona. Metoda bisekcji zawsze zbiega do pierwiastka, ale robi to powoli, szczególnie dla funkcji, które są prawie płaskie w pobliżu pierwiastka. Wadą metody Newtona-Raphsona jest to, że nie zawsze zbiega do pierwiastka, szczególnie jeśli punkt startowy jest daleko od rzeczywistego pierwiastka. Ponadto, metoda ta wymaga obliczenia pochodnej funkcji, co może być trudne dla niektórych funkcji.

6. Źródła

- prezentacja z zajęć „lab 7.pdf”
- wolframalpha.com