Trzeci Projekt Labowy

Metody Numeryczne 2021/2022, grupy 1 i 2

Termin oddania: do 30 stycznia 2022 włącznie

Parabole tańczą, tańczą, tańczą tańczą
Tańczą, tańczą, tańczą parabole

pewna niemądra przyśpiewka znaleziona na YT

Wprowadzenie

W *metodzie siecznych* rozwiązywania jednowymiarowych równań postaci $f(x) = 0$, gdzie $f: \mathbb{R} \to \mathbb{R}$ jest pewną funkcją, konstruowany jest ciąg $(x_n)_{n=1}^{\infty}$ przybliżeń rozwiązania, w którym x_{n+2} jest miejscem zerowym prostej przechodzącej przez punkty $\big(x_{n+1}, f(x_{n+1})\big)$ oraz $\big(x_n, f(x_n)\big)$.

Co się stanie, jeśli zamiast prostych pomyślimy o parabolach? Precyzyjniej rzecz ujmując, konstruujemy ciąg przybliżeń rozwiązania w następujący sposób: zaczynamy od dowolnych $x_1, x_2, x_3 \in \mathbb{R}$, a następnie dla $n = 4, 5, 6, \ldots$ za x_{n+3} przyjmujemy miejsce zerowe paraboli przechodzącej przez punkty $\big(x_{n+2}, f(x_{n+2})\big)$, $\big(x_{n+1}, f(x_{n+1})\big)$ oraz $\big(x_n, f(x_n)\big)$.

Trzeci projekt dotyczy implementacji wyżej opisanego podejścia. Znajdowanie równania stosownej paraboli to rzecz jasna interpolacja wielomianowa – dla polepszenia własności numerycznych, powinna zostać przeprowadzona przy użyciu algorytmu różnic dzielonych, tzn. szukaną parabolę chcemy opisać jako

$$y(x) = f[x_{n+2}] + f[x_{n+2}, x_{n+1}] \cdot (x - x_{n+2}) + f[x_{n+2}, x_{n+1}, x_n] \cdot (x - x_{n+2})(x - x_{n+1}). \quad (1)$$

W powyższej równości $f[\cdot]$ to odpowiednie różnice dzielone (patrz wykład dotyczący interpolacji wielomianowej).

Równość (1) nie jest najwygodniejsza w kontekście wyznaczania miejsca zerowego równania kwadratowego. Okazuje się, że można ją przepisać jako

$$y(x) = f[x_{n+2}] + C(x - x_{n+2}) + f[x_{n+2}, x_{n+1}, x_n] \cdot (x - x_{n+2})^2. \quad (2)$$

gdzie $C = f[x_{n+2}, x_{n+1}] + f[x_{n+2}, x_n] - f[x_{n+1}, x_n]$. Stąd

$$x_{n+3} = x_{n+2} - \frac{C \pm \sqrt{\Delta}}{2f[x_{n+2}, x_{n+1}, x_n]}, \quad (3)$$

gdzie $\Delta = C^2 - 4f[x_{n+2}]f[x_{n+2}, x_{n+1}, x_n]$.

Konkrety

W ramach niniejszego projektu należy przekuć opisane wyżej podejście funkcję przybliżającą rozwiązanie równania $f(x) = 0$. Pisząc bardziej precyzyjnie, należy zaimplementować funkcję

```
parasolve(f, init, eps=1e-3, N=100)
```

która zwraca x_{n^*}, przy czym

- $(x_n)_{n\in\mathbb{N}}$ jest ciągiem określonym rekurencyjnym wzorem (3), przy czym funkcja f jest zadana przez argument `f`, natomiast wyrazy początkowe (x_1, x_2, x_3) są przekazywane poprzez trójkę („tupla”) `init`.

- n^* to najmniejszy indeks taki, że $|f(x_{n^*})| \leqslant$ `eps`, o ile ów indeks nie przekracza `N`, gdyż w takim wypadku kładziemy $n^* =$ `N` (tzn. `N` jest maksymalną liczbą iteracji). Formalnie

$$n^* := \min\Big(\{\mathtt{N}\} \cup \{n \leqslant \mathtt{N} \colon |f(x_n)| \leqslant \mathtt{eps}\}\Big).$$

Jeśli $|f(x_{n^*})| >$ `eps`, użytkownik powinien zostać o tym poinformowany (np. poprzez wyświetlenie komunikatu o osiągnięciu maksymalnej liczby iteracji).

Za kod, który realizuje powyższe postulaty, otrzymuje się **co najmniej 1 punkt**. Kolejne punkty można uzyskać zgodnie z poniższymi wytycznymi:

1. parabola może mieć dwa miejsca zerowe – powyższy opis daje dwie możliwości wyboru x_{n+3} w zależności od x_{n+2}, x_{n+1}, x_n. Należy samodzielnie zdecydować, która jest rozsądniejsza i swój wybór uzasadnić w osobnym dokumencie, zwanym dalej „dokumentem X". Słuszny wybór wraz z sensowną motywacją skutkuje dodaniem **1 punktu**.

2. równość (3) dobrze wyraża nasze zamiary, ale w implementacji lepiej skorzystać z równoważnego sformułowania
$$x_{n+3} = x_{n+2} - \frac{2f[x_{n+2}]}{C \mp \sqrt{\Delta}}. \tag{4}$$
Dlaczego (4) może mieć numeryczną przewagę nad (3)? Poproszę o komentarz w dokumencie X. (*Podpowiedź:* ma to coś wspólnego z poprzednim punktem. Z którego rozwiązania należy skorzystać w (3), a z którego w (4)?) Rozsądne uzasadnienie i skorzystanie w implementacji z (4) (a nie (3)) skutkuje dodaniem **1 punktu**.

3. równość (4) może zaprowadzić nas w świat nierzeczywistych liczb zespolonych – *it's not a bug, it's a feature!* W ten sposób metoda może zostać użyta do poszukiwań zespolonych rozwiązań równania $f(x) = 0$. Oczywiście, ma to sens tylko wtedy gdy f możemy przykładać również do liczb zespolonych – **jest to nasze dodatkowe założenie projektowe.** W swoim rozwiązaniu należy zatem zadbać o obsługę (potencjalnych) zespolonych wyrazów ciągu $(x_n)_{n\in\mathbb{N}}$ – jeśli tak jest, rozwiązanie uzyskuje jeszcze **1 punkt**.

4. w dokumencie X należy przedstawić dowód równoważności między (1) a (2) oraz między (3) a (4), za co można otrzymać **1 punkt** (po pół punktu za każde uzasadnienie).

Uwagi końcowe

Tradycyjnie można korzystać z dobrodziejstw pakietów `numpy` oraz `scipy`. Obsługa pierwiastków z liczb zespolonych też może zostać przeprowadzona przy użyciu jakiegoś pakietu (np. `cmath`).