Sprawozdanie - Laboratorium 12
Zastosowanie ekstrapolacji Richardsona do całkowania przy użyciu wzorów Simpsona i Milne'a

Marta Dychała

5 czerwca 2021

1 Wstęp teoretyczny

Całkowanie numeryczne oznacza wykorzystanie metod numerycznych w celu wyznaczenia przybliżenia wartości teoretycznej całki oznaczonej. Właściwość ta sprawia, że całkowanie numeryczne pozwala na wyznaczenie wartości całek oznaczonych nawet dla takich funkcji, dla których pierwotna nie jest opisana za pomocą funkcji elementarnej.

Jedną z rodzin metod całkowania numerycznego są kwadratury Newtona-Cotes'a. W metodach tych do całkowania wykorzystuje się wielomian interpolacyjny Lagrange'a stopnia co najwyżej N, który dla ciągu wartości funkcji podcałkowej $f(x_0), f(x_1), ..., f(x_N)$, będącymi węzłami równoodległymi funkcji jest wyrażony wzorem:

$$f(x_i) = L_N(x) = \sum_{k=0}^{N} \Phi_k(x) F(x_k), \tag{1}$$

gdzie

$$\Phi_k(x) = \prod_{j=0}^{N} \frac{x - x_j}{x_k - x_j}, \qquad j \neq k$$

Najczęściej stosuje się kwadratury niższych rzędów takie jak:

- wzór Simpsona (parabol) - kwadratura NC rzędu 2:

$$S(x) = \sum_{i=0}^{(N/2)-1} \frac{h}{3}(f(x_{2i}) + 4f(x_{2i+1}) + f(x_{2i+2})) \tag{2}$$

- wzór Milne'a - kwadratura NC rzędu 4:

$$S(x) = \sum_{i=0}^{(N/4)-1} \frac{4h}{90}(7f(x_{4i}) + 32f(x_{4i+1}) + 12f(x_{4i+2}) + 32f(x_{4i+3}) + 7f(x_{4i+4})), \tag{3}$$

gdzie h odległość między dwoma węzłami jest zdefiniowana jako:

$$h = \frac{b-a}{N} \tag{4}$$

gdzie a, b to kolejno: dolna i górna granica całkowania, natomiast N to liczba podprzedziałów określona jako:

- dla wzoru Simpsona: $N = 2^{n+1} \implies h = \frac{b-a}{2^{n+1}}$
- dla wzoru Milne'a: $N = 2^{n+2} \implies h = \frac{b-a}{2^{n+2}}$

Parametr n pojawiający się w obydwu wzorach na N wykorzystany jest do ekstrapolacji Richardsona - metody, która teoretycznie jeszcze dokładniej pozwala wyznaczyć wartość całki oznaczonej. W tym celu tworzona jest trójkątna tablica D o wymiarach $n+1 \times n+1$:

$$D = \begin{pmatrix} D_{0,0} & & & & & \\ D_{1,0} & D_{1,1} & & & & \\ D_{2,0} & D_{2,1} & D_{2,2} & & & \\ \vdots & \vdots & \vdots & \ddots & & \\ D_{n-1,0} & D_{n-1,1} & D_{n-2,2} & \cdots & D_{n-1,n-1} & \\ D_{n,0} & D_{n,1} & D_{n,2} & \cdots & D_{n-1,n-1} & D_{n,n} \end{pmatrix}$$

której pierwsza kolumna ($D_{w,0}, \quad w = 0, 1, ..., n$) jest wynikiem kwadratury NC, natomiast pozostałe elementy są wyznaczane rekurencyjnie za pomocą wzoru:

$$D_{w,k} = \frac{4^k D_{w,k-1} - D_{w-1,k-1}}{4^k - 1} \tag{5}$$

W ten sposób najlepszym przybliżeniem wartości całki oznaczonej staje się $D_{n,n}$.

2 Zadanie do wykonania

2.1 Opis problemu

W czasie zajęć laboratoryjnych zadaniem było obliczenie wartości całki:

$$I = \int_0^1 f(x)dx \quad (= -0.186486896) \tag{6}$$

na dwa sposoby - wykorzystując w tym celu wzór Simpsona lub Milne'a. Funkcja podcałkowa f(x) ze wzoru (6) była określona wzorem:

$$f(x) = \ln(x^3 + 3x^2 + x + 0.1)\sin(18x) \tag{7}$$

Następnie dla każdej z dwóch metod należało przeprowadzić ekstrapolację Richardsona, tworząc tablicę D. Jako wartość parametru n przyjęto $n = 8$. Do zrealizowania zadania pierwsze kolumny tablicy D dla przypadków wzoru Simpsona oraz Milne'a wypełniono zgodnie ze wzorami (2) oraz (3), natomiast jej pozostałe elementy za pomocą wzoru (5).

Otrzymane wyniki dla obydwu metod zapisano do pliku tekstowego. Program potrzebny do zrealizowania zadania napisano w czystym $C++$ z wykorzystaniem typu double.

2.2 Wyniki

W celu sprawdzenia efektu, jaki daje ekstrapolacja Richardsona na otrzymany wynik, poniżej porównano ze sobą wartości otrzymane za pomocą czystych kwadratur (pierwsza kolumna tablicy D, w tabelach oznaczona jako $D_{w,0}$) oraz kwadratur z zastosowaniem ekstrapolacji Richardsona (główna przekątna tablicy D, oznaczona jako $D_{w,w}$).

Każdy z wierszy poniższych tabel odpowiada zwiększonej liczbie węzłów użytych do całkowania numerycznego.

Tabela 1: Wartości całek uzyskane za pomocą metody Simpsona bez i wraz z ekstrapolacją Richardsona

w	$D_{w,0}$	$D_{w,w}$
0	-0.0971410499	-0.0971410499
1	0.4083851989	0.5768939485
2	-0.2209681412	-0.4979290236
3	-0.1880063997	-0.1547412817
4	-0.1865747211	-0.1872519207
5	-0.1864922827	-0.1864826455
6	-0.1864872311	-0.1864869011
7	-0.1864869169	-0.1864868960
8	-0.1864868973	-0.1864868960

W czystej metodzie Simpsona nie udało się po 9 iteracjach wyznaczyć całki z zadowalającą dokładnością. Dopiero zastosowanie ekstrapolacji Richardsona sprawiło, iż dokładna wartość całki została wyznaczona po 7 iteracjach.

Tabela 2: Wartości całek uzyskane za pomocą metody Milne'a bez i wraz z ekstrapolacją Richardsona

w	$D_{w,0}$	$D_{w,w}$
0	0.4420869488	0.4420869488
1	-0.2629250305	-0.4979290236
2	-0.1858089502	-0.1375818946
3	-0.1864792758	-0.1892838357
4	-0.1864867868	-0.1864321619
5	-0.1864868943	-0.1864871836
6	-0.1864868960	-0.1864868957
7	-0.1864868960	-0.1864868960
8	-0.1864868960	-0.1864868960

Przeciwna sytuacja ma miejsce w przypadku metody Milne'a - zastosowanie ekstrapolacji zmniejszyło skuteczność metody. Dla czystego wzoru Milne'a wartość całki z wymaganą dokładnością otrzymano już dla $w = 6$, natomiast wraz z zastosowaniem ekstrapolacji Richardsona należało wykonać jedną iterację więcej. Jest to o tyle zaskakujące, że wraz z zastosowaniem ekstrapolacji Richardsona można było by się spodziewać poprawy wyniku.

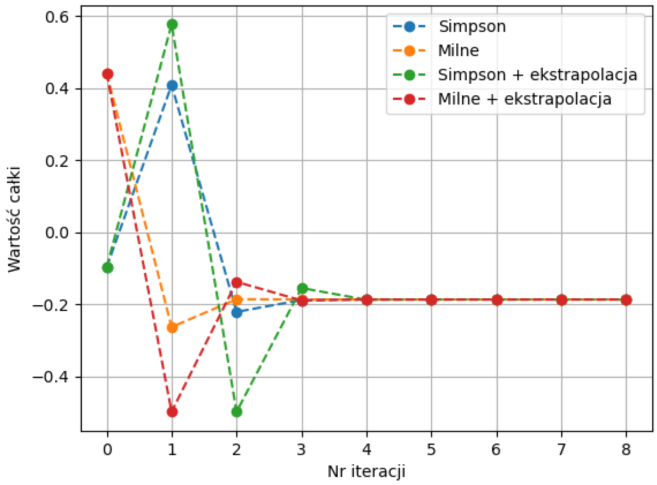
Ponieważ zastosowanie czystej metody Simpsona nie dało wyniku z zadowalającą dokładnością, zwiększono liczbę iteracji, a dodatkowe wyniki dla tej metody zebrano do poniższej tabeli:

Tabela 3: Wartości całek uzyskane za pomocą metody Simpsona bez i wraz z ekstrapolacją Richardsona dla $w = \{9, 10, 11, 12\}$

w	$D_{w,0}$	$D_{w,w}$
9	-0.1864868961	-0.1864868960
10	-0.1864868960	-0.1864868960
11	-0.1864868960	-0.1864868960
12	-0.1864868960	-0.1864868960

Okazuje się, że dla czystej metody Simpsona należało użyć 10 iteracji, aby uzyskać wynik z zadaną dokładnością.

Ponadto w celu porównania zmian wartości wraz z numerem iteracji dla każdej z metod całkowania, zestawiono poniższy wykres wygenerowany za pomocą biblioteki Pythona Matplotlib:



Rysunek 1: Porównanie zmian wartości obliczonej całki po danym kroku dla różnych metod całkowania numerycznego. Przerywane linie zostały dodane w celu bardziej czytelnego zobrazowania zmian wartości.

Na powyższych wykresach można zauważyć, iż w początkowej fazie algorytmu w metodach Simpsona i Milne'a wartości całek zmieniają się gwałtownie. Co więcej, zmiana wartości całki w metodach Milne'a przebiega na ogół łagodniej niż w metodach Simpsona. Wartym zauważenia jest także fakt, iż dla obydwu metod zastosowanie ekstrapolacji Richardsona sprawiło, że wartości całek zmieniały się gwałtowniej w początkowych iteracjach, by potem jednak osiągnąć wartość z żądaną dokładnością.

3 Wnioski

Dla całki z zajęć laboratoryjnych najbardziej skuteczna okazała się czysta metoda Milne'a - odpowiednie przybliżenie udało się znaleźć już po $w = 6$ iteracjach. Metoda Simpsona, nawet z zastosowaniem ekstrapolacji Richardsona okazała się być mniej skuteczna, bowiem i tak wymagała ona $w = 7$ iteracji.

W przypadku całkowania numerycznego za pomocą kwadratur Newtona-Cotes'a ważne okazuje się dobranie odpowiedniej metody. W naszym przypadku okazało się bowiem, iż za pomocą metody Milne'a szybciej znaleziono akceptowalne rozwiązanie niż przy metodzie Simpsona. Ważne jest także dobranie liczby iteracji potrzebnej do wyznaczenia wartości całki - dla przypadku z zajęć laboratoryjnych liczba $w = 8$ iteracji dla czystej metody Simpsona okazała się niewystarczająca, a właściwą wartość dla tej metody można było znaleźć dopiero po $w = 10$ iteracjach.

W celu poprawienia zbieżności wyników można posłużyć się ekstrapolacją Richardsona, w której pomimo że znajdowane wartości całki początkowo bardziej odbiegają od właściwej, to możliwe jest - przy dobraniu odpowiedniej metody - szybsze znalezienie zadowalającego przybliżenia. Słowem kluczem staje się tutaj „odpowiedniej" - w przypadku metody Simpsona ekstrapolacja Richardsona faktycznie pomogła szybciej znaleźć wymagane przybliżenie - już po 7 iteracjach zamiast 10 przy „czystej" metodzie. Niestety, w przypadku metody Milne'a już nie przyspieszyła znalezienia właściwej wartości, a nawet wręcz opóźniła o jedną iterację.