Stanisław Wilczyński*

Pracownia z analizy numerycznej
Sprawodzdanie do zadania **P.1.12**

Wrocław, 7 listopada 2015

Spis treści

1. Wstęp

Obliczanie całek jest problemem, który jest nie tylko teoretyczny, ale i ważny ze względu na swoje zastosowania w praktyce w dziedzinach takich jak fizyka,chemia, budownictwo, czy analiza ryzyka finansowego. Obliczenie jednej czy dwóch prostych całek nie jest problem dla przeciętnych studentów pierwszego roku dowolnych studiów ścisłych, jednak w przypadku poważniejszych obliczeń wymagana jest pomoc komputera. Niestety jednak przy każdym użyciu takiego sprzętu musimy liczyć się z możliwym błędem związanym np. ze zjawiskiem utraty cyfr znaczących. Powszechnie znane metody Simpsona czy trapezów są dokładne jednak są bardzo obliczeniowo - wymagają między innymi obliczania wartości całkowanej funkcji w wielu punktach. Poszukamy sposobu na inne radzenie sobie z takimi obliczeniami. W tym sprawozdaniu zajmiemy się całką $\int_0^1 t^n e^t \mathrm{d}t$. Zostaną zaprezentowane dwie metody do obliczania wartości tej całki dla kolejnych $n \in \mathbb{N}$.

* *E-mail*: opos1@onet.eu

2. Matematyczne podstawy

Niech (y_n) będzie ciągiem zdefiniowanym następująco:

$$y_n = \int_0^1 t^n e^t \mathrm{d}t$$

Najpierw sprawdzimy że ten ciąg jest malejący:

$$\begin{aligned} y_n - y_{n-1} &= \int_0^1 t^n e^t \mathrm{d}t - \int_0^1 t^{n-1} e^t \mathrm{d}t = \int_0^1 t^{n-1}(t-1)e^t \mathrm{d}t \leqslant 0, \\ &\text{bo } (t-1) \leqslant 0 \,\text{dla każdego } t \in (0,1) \end{aligned}$$

Teraz pokażemy, że (y_n) zbiega do 0. Z twierdzenia o wartości średniej ([1],strona 102):

$$1 \int_0^1 t^n \mathrm{d}t \leqslant y_n \leqslant e \int_0^1 t^n \mathrm{d}t \tag{1}$$

Otrzymujemy więc:

$$\frac{1}{n+1} \leqslant y_n \leqslant \frac{e}{n+1} \tag{2}$$

Zauważmy że

$$\lim_{n\to\infty} \frac{1}{n+1} = \lim_{n\to\infty} \frac{e}{n+1} = 0$$

więc z twierdzenia o trzech ciągach([1],strona 6) i nierówności 2 mamy:

$$\lim_{n\to\infty} y_n = 0$$

Następnie znajdujemy rekurencyjny związek między y_n i y_{n-1}. Stosując całkowanie przez części otrzymujemy:

$$y_n = \int_0^1 t^n e^t \mathrm{d}t = t^n e^t|_{t=0}^{t=1} - \int_0^1 n t^{n-1} e^t \mathrm{d}t = e - n \int_0^1 t^{n-1} e^t \mathrm{d}t = e - n y_{n-1} \tag{3}$$

Obliczamy jeszcze

$$y_0 = \int_0^1 e^t \mathrm{d}t = e - 1 = 1.718281828459045... \tag{4}$$

3. Metoda I - naiwna

3.1. Opis i działanie metody I

W tej metodzie wykorzystamy związek rekurencyjny(3) do obliczania kolejnych wartości, tzn mając dane y_{n-1} wyznaczamy y_n. Za pomocą załączonego programu *program.jl* policzyno pierwsze 21 wyrazów naszego ciągu(w arytmetyce z podwójną precyzją). Niech $\tilde{y_n}$ oznacza obliczoną wartość wyrazów ciągu. Otrzymane wyniki prezentujemy w tabelce(w tej i każdej następnej tabelce rzeczywisty wynik y_n jest podawany do 16 miejsca po przecinku):

n	$\tilde{y_n}$	y_n	liczba tych samych cyfr znaczących w y_n i $\tilde{y_n}$
0	1.7182818284590452	1.7182818284590452	16
2	0.7182818284590451	0.7182818284590452	15
4	0.4645364561314058	0.4645364561314071	14
6	0.3446845416469490	0.3446845416469873	13
8	0.2743615330158317	0.2743615330179760	11
10	0.2280015152934535	0.2280015154864418	9
12	0.1950999056863769	0.1950999311608206	7
14	0.1705190649530128	0.1705237013017674	4
16	0.1503481618374036	0.1514608855385011	2
18	-0.204253561558256	0.1362398909775906	0
20	-129.2637081328594	0.1238038307625699	0

Zauważamy,że liczba takich samych cyfr znaczących w liczbach y_n i $\tilde{y_n}$ maleje o 1 w przy prawie każdej iteracji. Oznacza to, że wyniki $\tilde{y_n}$ dla $n \geqslant 18$ nie mają nawet jednej takiej samej cyfry jak rzeczywiste wartości y_n. Co więcej ciąg $\tilde{y_n}$ nie jest nawet malejący tak jak y_n. Skąd bierze się tak ogromna niedokładność obliczeń?

3.2. Analiza metody I

Niech d_n oznacza błąd bezwzględny, tzn. $d_n = |\tilde{y_n} - y_n|$. Zgodnie z definicją zawartą w [2] na stronie 48, mówimy, że algorytm jest numerycznie niestabilny, jeśli małe błedy popełnione na początku obliczeń, rosną na tyle, że w następnych krokach algorytmu całkowicie wypaczją wynik obliczeń. Formalnie, aby sprawdzić czy algorytm jest niestabilny numerycznie należy sprawdzić błąd względny czyli w naszym wypadku $\frac{d_n}{y_n}$ i porównać z wartością y_n. Przeprowadzimy analizę błędu bezwzględnego. Zauważmy, że błąd d_2 jest rzędu precyzji naszej arytmetyki DOUBLE, tzn. $d_2 = 10^{-16}$. Co więcej:

$$d_n = |\tilde{y_n} - y_n| = |(e - n\tilde{y_{n-1}}) - (e - ny_{n-1})| = |n(\tilde{y_{n-1}} - y_{n-1})| = nd_{n-1} \tag{5}$$

Powyższy rachunek daje nam tylko przybliżoną wartość błędu, gdyż pomijamy tu niedokładności wynikające z wykonywania operacji arytmetycznych. Mimo wszytsko korzystając z powyższej równości możemy wyliczyć wzór na d_n:

$$d_n = nd_{n-1} = n(n-1)d_{n-2} = ... = \frac{1}{2}n!d_2 \tag{6}$$

Korzystając z tgeo wzoru obliczamy $d_{20} = \frac{1}{2}20!d_2 \approx \frac{1}{2}2,4 \times 10^{18} \times 10^{-16} \approx 120$ i zauważamy, że dla już 20 wyrazu naszego ciągu obliczona wartość nie jest nawet bliska wartości rzeczywistej. Co więcej błąd względny wynosi aż $\frac{d_{20}}{y_{20}} \approx 1044,13$, co oznacza,że wartość d_{20} jest blisko 1000 razy większa niż rzeczywista wartość y_{20}. Oczywiście mając na uwadze wzór 6 wnioskujemy, że zastosowany tutaj algorytm wyliczania y_n jest numerycznie niestabiny.

4. Metoda II - ulepszona

4.1. Opis metody II

Druga metoda jest oparta na algorytmie Millera([3]). Zauważmy, że korzystjąc z nierówności 2 możemy stwierdzić, że ciąg (y_n) jest wolno zbieżny, a co za tym idzie, dla dużych N $y_N \approx e - Ny_N$. Z tego związku możemy policzyć, że $y_N \approx \frac{e}{N+1}$. Stosując oznaczenia z poprzedniego rozdziału kładziemy:

$$\tilde{y_N} = \frac{e}{N+1}$$

Wtedy $d_N \leqslant \frac{e-1}{N+1}$ (nierówność 2) Oczywiście, dzięki naszemu związkowi rekurencyjnemu możemy się cofać, tzn. obczliczyć y_{n-1} w zależności od y_n. Przekształcając równanie 3 otrzymujemy:

$$y_{n-1} = \frac{e - y_n}{n} \tag{7}$$

Za pomocą powyższej równości w załączonym programie *program.jl* obliczamy wartości wyrazów naszego ciągu zaczynając raz od N=20.

N	$\tilde{y_N}$	y_N	liczba tych samych cyfr znaczących w y_N i $\tilde{y_N}$
20	0.1294419918313831	0.1238038307625699	2
18	0.1362547282435611	0.1362398909775906	4
16	0.1514609340262984	0.1514608855385011	6
14	0.1705237015037999	0.1705237013017674	9
12	0.1950999311619307	0.1950999311608206	11
10	0.2280015154864502	0.2280015154864418	13
8	0.2743615330179761	0.2743615330179760	15
6	0.3446845416469873	0.3446845416469873	16
4	0.4645364561314070	0.4645364561314071	16
2	0.7182818284590452	0.7182818284590452	16
0	1.7182818284590452	1.7182818284590452	16

Porównując tabelki dla metody I i metody II zauważamy, że II jest dokładniejsza o około 5 cyfr. Ponieważ w metodzie II błąd przyobliczaniu pierwszej wartości zależyod tego jakie N wybraliśmy, stwierdzamy, że dla $N \geqslant 20$ dokładność będzie jeszcze lepsza, a dokłądność metody I ulega pogorszeniu w raz ze wzrostem N. Skąd bierze się taka znaczna różnica wyniku dla obu metod?

4.2. Analiza metody II

Podobnie jak w metodzie I przeprowadzimy analizę błędu bezwzględnego. Już w poprzednim rozdziale zauważyliśmy, że jeśli rozpoczynamy działanie naszej metody od N, to

$$d_N \leqslant \frac{e-1}{N+1}$$

Skoro mamy wzór rekurencyjny na wcześniejsze wyrazy ciągu(7) możemy również obliczyć przybliżone błędy bezwzględne:

$$d_{N-1} = |y_{\tilde{N-1}} - y_{N-1}| = \left|\frac{e - \tilde{y_N}}{N} - \frac{e - y_N}{N}\right| = \left|\frac{\tilde{y_N} - y_N}{N}\right| = \frac{d_N}{N} \tag{8}$$

Dalej korzystając z powyższej równości otrzymujemy:

$$d_{N-k} = \frac{(N-k)!}{N!} d_N \tag{9}$$

Dla naszego przypadku $N = 20$ $d_{20} \approx 3 \times 10^{-3}$
Oznaczało to, że już $d_8 = \frac{8!}{20!} d_N \approx 1,6 \times 10^{-14} \times 6 * 10^{-3} \approx 1 \times 10^{-16}$ i jeśli zaglądniemy do tabelki jest to nawet dokładnie błąd, który otrzymaliśmy przy wyliczaniu $\tilde{y_8}$.

5. Podsumowanie

Po przeprowadzeniu doświadczenia, możemy stwierdzić, że metoda I nie nadaje się do obliczania dokładnych wartości wyrazów ciągu (y_n). Jednakże, łatwo zauważyć, że metoda II pozwala nam wyliczyć każdy wyraz ciągu (y_n) z dokładnością do 16 miejsca po przecinku. Korzystając ze wzoru 9 otrzymujemy:

$\mathrm{d}_n = \frac{n!}{(n+k)!} d_{n+k}$ dla każdego $k \in N$ (10)

W takim razie żeby nasz błąd $d_n < 10^{-16}$ wystarczy znaleźć takie k, że $\frac{n!}{(n+k)!} d_{n+k} \leqslant \frac{n!}{(n+k)!} \times \frac{e-1}{n+1} < 10^{-16}$
Oczywiście takie k można znaleźć i co więcej, im większe n tym mniejsze k, tzn. dla $n = 100$ $k = 9$, a dla $n = 1000$ $k = 6$. Podsumowując, aby otrzymać dokładny wynik y_n za pomocą metody drugiej wystarczy rozpoczęcia obliczeń od wyrazu ciągu o indeksie o kilka(-naście) większym. Ponadto, aby zwiększyć dodatkowo dokładność metody II do większej liczb cyfr po przecinku wystarczy wybrać dla danego n większe niż wcześniej k w taki sposób, aby $\frac{n!}{(n+k)!} \times \frac{e-1}{n+1} < 10^p$, gdzie p jest liczbą cyfr po dziesiętnych, które chcemy wyznaczyć.

Literatura

[1] R. Szwarc, Skrypt do wykładu z analizy matematycznej I `http://www.math.uni.wroc.pl/~szwarc/pdf/AnalizaISIM1.pdf`.

[2] D.Kincaid, W.Cheney, Numerical Analysis. Mathematics of scientific computing, Brooks/Cole Publishing Company,Pacific Groove California, 1991.

[3] J.Wimp, Computation with recurrence relations, Boston:Pitman Advanced Publ. Program, 1984.