

1 Fresnelov integral

Avtor: Luka Bajić

1.1 Opis problema

Cilj je poiskati vrednost Fresnelovega kosinusa

$$C(x) = \int_0^x \cos\left(\frac{\pi t^2}{2}\right) dt$$

za poljubno vhodno realno število x . Pomagamo si lahko s pomožnima funkcijama

$$f(z) = \frac{1}{\pi\sqrt{2}} \int_0^\infty \frac{e^{-\frac{\pi z^2 t}{2}}}{\sqrt{t}(t^2 + 1)} dt$$
$$g(z) = \frac{1}{\pi\sqrt{2}} \int_0^\infty \frac{\sqrt{t} e^{-\frac{\pi z^2 t}{2}}}{t^2 + 1} dt$$

Zveza med pomožnima funkcijama in Fresnelovim kosinusom je sledeča:

$$C(x) = \frac{1}{2} + f(x)\sin\left(\frac{\pi x^2}{2}\right) - g(x)\cos\left(\frac{\pi x^2}{2}\right)$$

Želena natančnost izračunanega rezultata je $5 * 10^{-11}$.

1.2 Opis rešitve

Ker velja zveza $C(-x) = -C(x)$, lahko algoritem implementiramo tako, da se vedno izvaja na intervalu $[0, x]$ in če $x < 0$, dobljen rezultat pomnožimo z -1 .

Za $|x| \leq 1.5$ lahko uporabimo kar potenčno vrsto, ki konvergira že za dovolj majhne n -je, npr. $n = 14$:

$$C(x) = \sum_{n=0}^{\infty} \frac{(-1)^n \left(\frac{1}{2}\pi\right)^{2n} x^{4n+1}}{(2n)!(4n+1)}$$

Avtorji [1] zagotavljajo natančnost na 16 decimalok, kar zadošča našim zahtevam, prav tako pa je časovna zahtevnost konstantna za vsak x na intervalu $[-1.5, 1.5]$.

Za $|x| > 1.5$ pa se moramo poslužiti numeričnih algoritmov za računanje vrednosti integralov. V nadaljevanju predstavimo dva pristopa: računanje pomožnih funkcij z uporabo Gauss-Laguerrovih kvadratur, ki se izkaže kot počasen in ne dovolj natančen pristop, ter računanje integrala z adaptivnim Simpsonovim pravilom.

1.2.1 Gauss-Laguerrove kvadrature

Gauss-Laguerrove kvadrature se uporabljajo za aproksimacijo integralov oblike

$$\int_0^\infty e^{-x} f(x) dx$$

Z uporabo substitucije $y = \frac{\pi z^2}{2}$ lahko pomožne funkcije za Fresnelov integral preoblikujemo v zgornjo obliko na sledeč način:

$$f(z) = \frac{1}{\pi\sqrt{2}} \int_0^\infty e^{-yt} \frac{1}{\sqrt{t}(t^2 + 1)} dt$$

$$f(z) = \frac{1}{\pi\sqrt{2}} \int_0^\infty e^{-yt} \frac{\sqrt{t}}{(t^2 + 1)} dt$$

pri čemer $\frac{1}{\sqrt{t}(t^2+1)}$ in $\frac{\sqrt{t}}{(t^2+1)}$ predstavljata $f(x)$ v zgoraj omenjeni obliki. Tako preoblikovani pomožni funkciji nato aproksimiramo z n vozli in n utežmi, ki jih pridobimo s takoimenovanim Golub-Welschovim algoritmom, tako, da skonstruiramo tridiagonalno matriko v kateri so diagonalni elementi enaki $1, 3, 5, \dots, 2n-1$, elementi nad in pod diagonalo pa $1, 2, 3, \dots, n-1$. Vrednosti vozlov so kar lastne vrednosti te matrike, uteži pa izračunamo iz prvih komponent pripadajočih lastnih vektorjev kot $w_i = (x_{i,1})^2$.

1.2.2 Adaptivno Simpsonovo pravilo

Ideja adaptivnih pravil je, da na intervalu $[a, b]$, na katerem računamo integral, uporabimo rekurziven postopek za računanje vrednosti v levem in desnem podintervalu. Te podintervale pa določamo glede na napako, torej tam kjer je integral "težje" izračunati, izvedemo več razpolavljanj, dokler ne dosežemo želene tolerance.

Algoritem najprej izračuna vrednosti funkcije v robnih točkah a in b , ter v sredinski točki $c = \frac{a+b}{2}$, torej dobimo $f(a)$, $f(b)$ in $f(c)$. Da ocenimo napako, izvedemo še dve razpolavljanji $d = \frac{a+c}{2}$ in $e = \frac{c+b}{2}$ ter izračunamo vrednosti funkcije $f(d)$ in $f(e)$. S tako izračunanimi vrednostmi, lahko napako dobimo kot:

$$S = \frac{h}{12}(f(a) + 4f(d) + 2f(c) + 4f(e) + f(b)) - \frac{h}{6}(f(a) + 4f(c) + f(b))$$

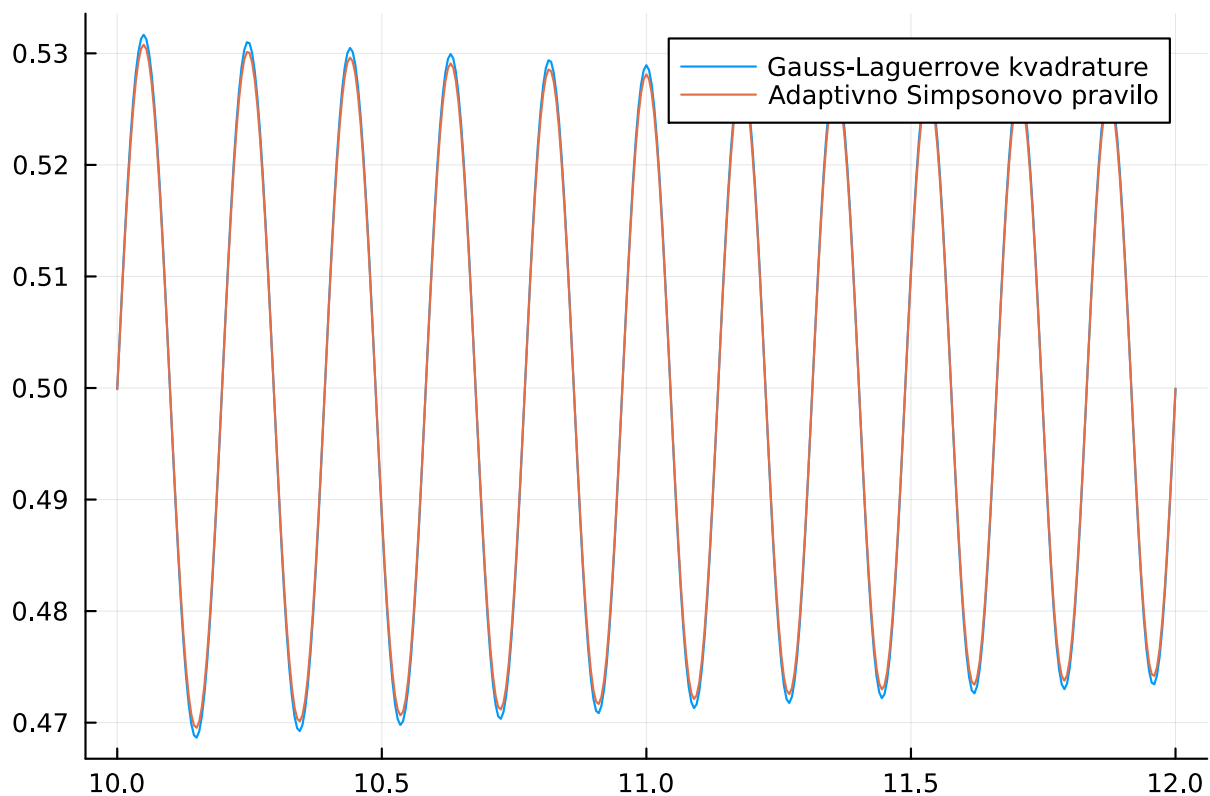
Če je napaka manjša od 15ϵ , kjer je ϵ zelena toleranca, se algoritem zaključi, sicer se rekurzivno izvedeta podalgoritma na intervalih $[a, c]$ in $[c, b]$, končni rezultat pa je vsota teh dveh rekurzivno izračunanih vrednosti. Pozorni moramo biti, da pri rekurzivnem klicu toleranco prepolovimo.

1.3 Primer uporabe

Spodnji izsek kode izriše vrednosti Fresnelovega kosinusa na poljubno izbranem intervalu $[10, 12]$, z uporabo Gauss-Laguerrovih kvadratur in adaptivnega Simpsonovega pravila. Vidimo da tudi za relativno veliko število vozlov, Gauss-Laguerrove kvadrature niso dovolj natančne v primerjavi z adaptivno metodo. Zaradi močnega osciliranja funkcije, tudi za več tisoč vozlov napaka ni boljša od treh decimalnih mest za večino vhodnih podatkov, medtem ko z adaptivnim Simpsonovim pravilom dosežemo zeleno natančnost na vseh testnih primerih.

Ena izmed slabosti tega pristopa pa je, da časovna kompleksnost ni neodvisna od vhodnega podatka.

```
using Fresnel, Plots
x = []
ys = []
ygl = []
for i=10:0.005:12
    push!(x, i)
    push!(ys, fresnel_cos(i))
    push!(ygl, fresnel_cos(i, metoda="gauss_laguerre", n_vozlov=300))
end
plot(x[:], ys[:], label="Gauss-Laguerrove kvadrature")
plot!(x[:], ygl[:], label="Adaptivno Simpsonovo pravilo")
```



1.4 Reference

[1] Alazah, Chandler-Wilde, La Porte: Computing Fresnel Integrals via Modified Trapezium Rules