

11. naloga: Reševanje PDE z metodo Galerkina

Gašper Jalen 28181038

1 Uvod

Pri opisu enakomernega laminarnega toka viskozne in nestisljive tekočine po dolgi ravni cevi pod vplivom stalnega tlačnega gradienta p' se Navier-Stokesova enačba poenostavi v Poissonovo enačbo

$$\nabla^2 v = \Delta v = -\frac{p'}{\eta},$$

kjer je v vzdolžna komponenta hitrosti, odvisna samo od koordinat preseka cevi, η pa je viskoznost tekočine. Enačbo rešujemo v notranjosti preseka cevi, medtem ko je ob stenah hitrost tekočina enaka nič. Za pretok velja Poiseuillov zakon

$$\Phi = \int_S v \, dS = C \frac{p' S^2}{8\pi\eta},$$

kjer je koeficient C odvisen samo od oblike preseka cevi ($C = 1$ za okroglo cev). Določili bomo koeficient za polkrožno cev z radijem R . V novih spremenljivkah $\xi = r/R$ in $u = v\eta/(p'R^2)$ se problem glasi

$$\Delta u(\xi, \phi) = -1, \quad u(\xi = 1, \phi) = u(\xi, 0) = u(\xi, \phi = \pi) = 0,$$

$$C = 8\pi \iint \frac{u(\xi, \phi) \xi \, d\xi \, d\phi}{(\pi/2)^2}.$$

Če poznamo lastne funkcije diferencialnega operatorja za določeno geometrijo¹ se reševanje parcialnih diferencialnih enačb včasih lahko prevede na razvoj po lastnih funkcijah. Da bi se izognili računanju lastnih (za ta primer Besselovih) funkcij in njihovih ničel, ki jih potrebujemo v razvoju, lahko zapišemo aproksimativno rešitev kot linearno kombinacijo nekaj poskusnih (*trial*) funkcij

$$\tilde{u}(\xi, \phi) = \sum_{i=1}^N a_i \Psi_i(\xi, \phi), \quad (1)$$

za katere ni nujno, da so ortogonalne, pač pa naj zadoščajo robnim pogojem, tako da jim bo avtomatično zadoščala tudi vsota (1). Ta pristop nam pride prav v kompleksnejših geometrijah, ko je uporabnost lastnih funkcij izključena in potrebujemo robustnejši pristop. Približna funkcija \tilde{u} seveda ne zadosti Poissonovi enačbi: preostane majhna napaka ε

$$\Delta \tilde{u}(\xi, \phi) + 1 = \varepsilon(\xi, \phi).$$

Pri metodi Galerkina zahtevamo, da je napaka ortogonalna na vse poskusne funkcije Ψ_i ,

$$(\varepsilon, \Psi_i) = 0, \quad i = 1, 2, \dots, N.$$

¹Spomni se na primer na vodikov atom v sferični geometriji, kjer smo imeli $\hat{L}^2 Y_{lm}(\vartheta, \phi) = \hbar^2 l(l+1) Y_{lm}(\vartheta, \phi)$ in $\hat{L}_z Y_{lm}(\vartheta, \phi) = m\hbar Y_{lm}(\vartheta, \phi)$.

V splošnem bi lahko zahtevali tudi ortogonalnost ε na nek drug sistem utežnih (*weight*) oziroma testnih (*test*) funkcij Ψ_i . Metoda Galerkina je poseben primer takih metod (*Methods of Weighted Residuals*) z izbiro $\Psi_i = \Psi_i$. Omenjena izbira vodi do sistema enačb za koeficiente a_i

$$\sum_{j=1}^N A_{ij} a_j = b_i, \quad i = 1, 2, \dots, N, \quad (2)$$

$$A_{ij} = (\Delta \Psi_j, \Psi_i), \quad b_i = (-1, \Psi_i),$$

tako da je koeficient za pretok enak

$$C = -\frac{32}{\pi} \sum_{ij} b_i A_{ij}^{-1} b_j.$$

Za kotni del poskusne funkcije obdržimo eksaktne funkcije $\sin((2m+1)\phi)$, Besselove funkcije za radialni del pa nadomestimo s preprostejšimi funkcijami $\xi^{2m+1}(1-\xi)^n$. Pozor: indeks i pomeni seveda dvojni indeks (šteje obenem m in n)

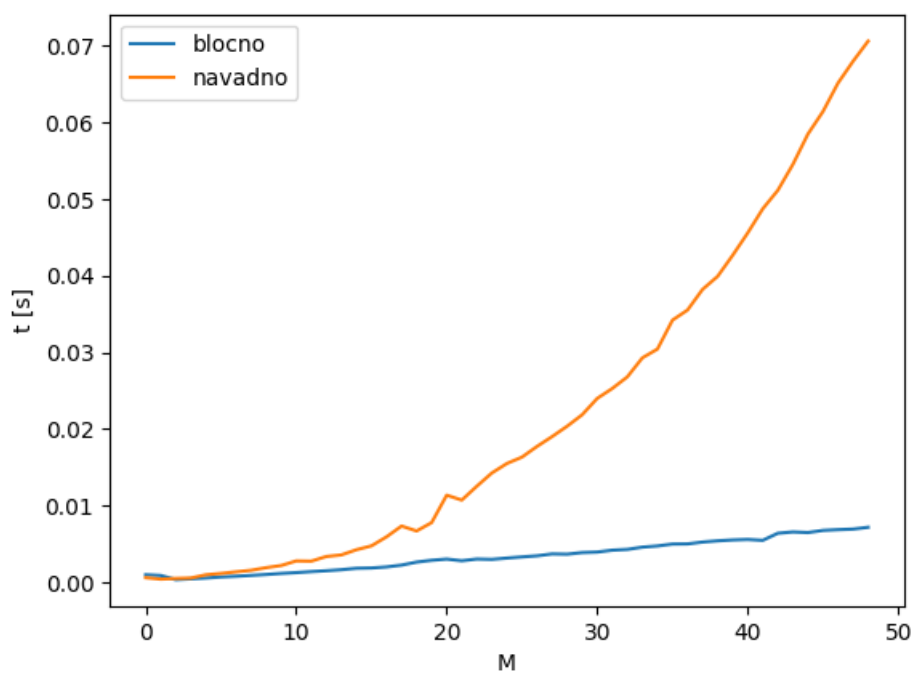
2 Izračun C

Pri izračunu koeficienta C sem moral reševati sistem enačb $\mathbf{A}\mathbf{a} = \mathbf{b}$. Pri tem sem uporabljal funkcijo `numpy.linalg.solve`. Ker je matrika A bločno diagonalna lahko sistem rešujemo po blokih. Sprva je potrebno zapisati elemente matrike A in vektroja \mathbf{b} , ki jih podata enačbi

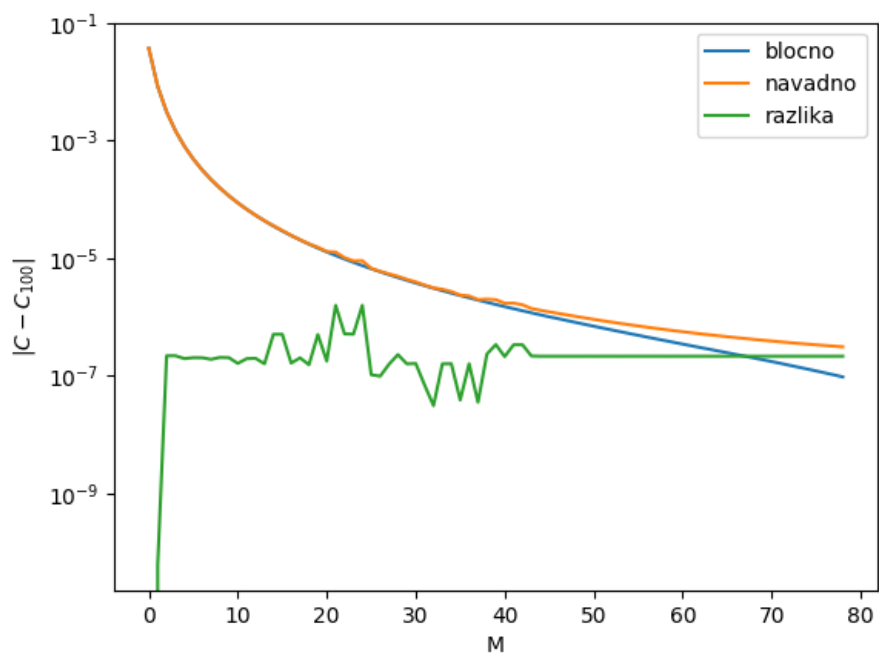
$$A_{(m'n')(mn)} = -\frac{\pi}{2} \delta_{m'm} \frac{nn'(3+4n)}{2+4m+n+n'} B(n+n'-1, 3+4m), \quad (3)$$

$$\mathbf{b}_{(m'n')} = -\frac{2}{2m'+1} B(2m'+3, n'+1), \quad (4)$$

pri čemer si poljubno izberemo nek M in N (M določa število blokov, N pa velikost), po katerih indeksa n in m tečeta in določata dimenzijo matrike. Ker ne poznam delovanja funkcije `numpy.linalg.solve` sem preveril hitrost obeh načinov reševanja sistema enačb (navadno in reševanje sistema za vsak blok na diagonali matrike). Pričakovano je bilo bločno reševanje precej hitrejše. Hkrati sem preveril tudi natančnost reševanja. Ker točne vrednosti koeficienta ne poznam, sem za obe metodi preveril kako se približujeta neka referenčni vrednosti izračunani za večja N in M.

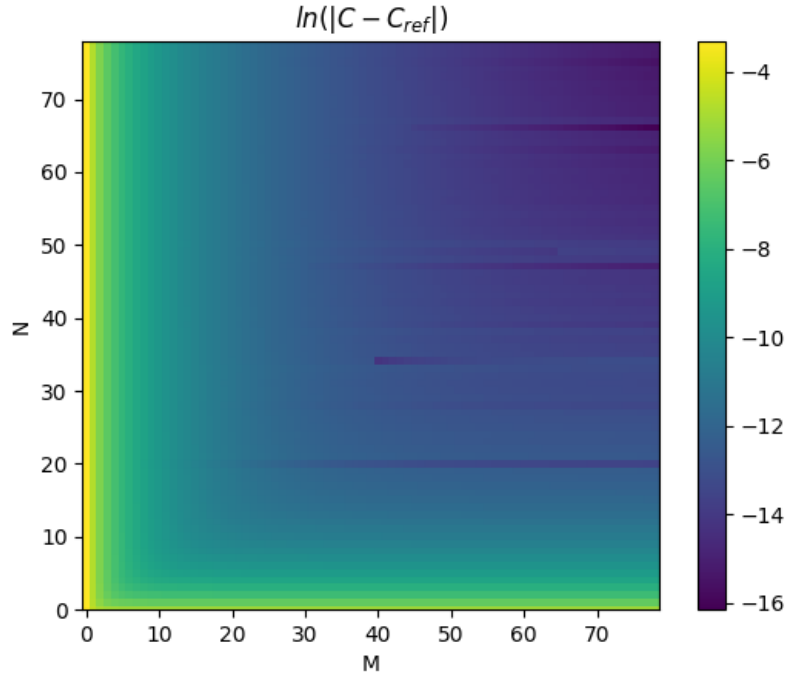


Slika 1: Časovna zahtevnost bločnega in navadnega reševanja sistema enačb za različno število blokov M (pri konstantni velikosti blokov $N=40$).



Slika 2: Razlika med izračunano vrednostjo koeficienta za nek M v primerjavi z izračunom pri $M = 100$, za obe metodi in razlika med obema metodama (pri konstantni velikosti blokov $N=40$).

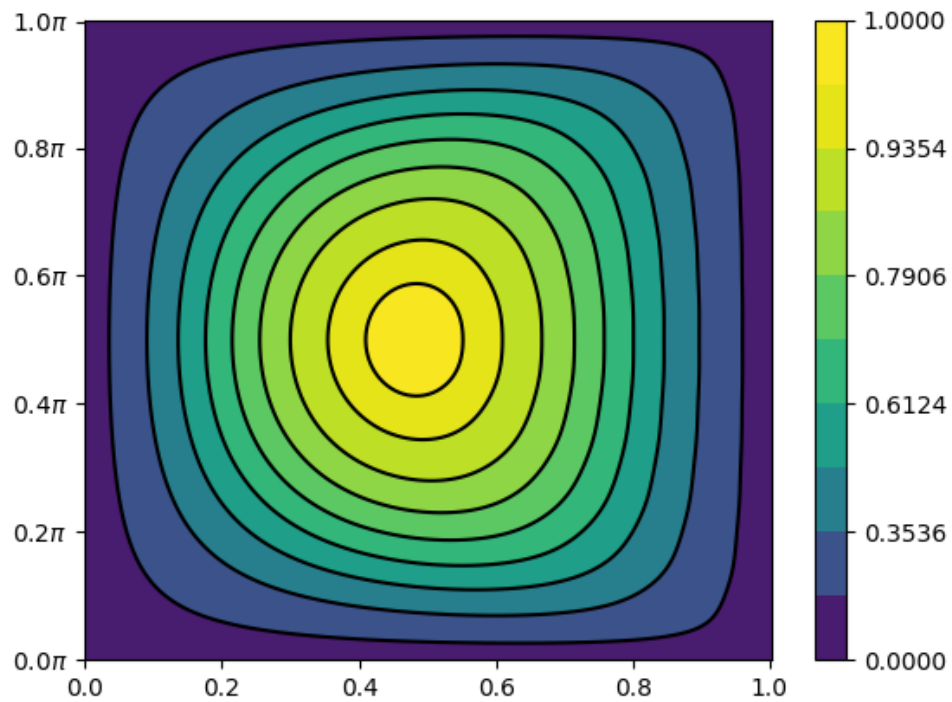
Pri primerjavi metod opazimo, da se različno hitro približujeta svojima referenčnima vrednostima, razlika med njima pa ostaja v logaritemski skali približno konstantna. S temi podatki sicer ni mogoče določiti kateri način reševanja je boljši, zato sem zaradi hitrosti nalogo reševal z bločnim načinom. Pri $M = N = 100$ sem dobil za koeficient C vrednost $C = 0.7577218702419775$. Na koncu sem prikazal, kako se pri večanju N in M , vrednost C približuje tej referenčni vrednosti.



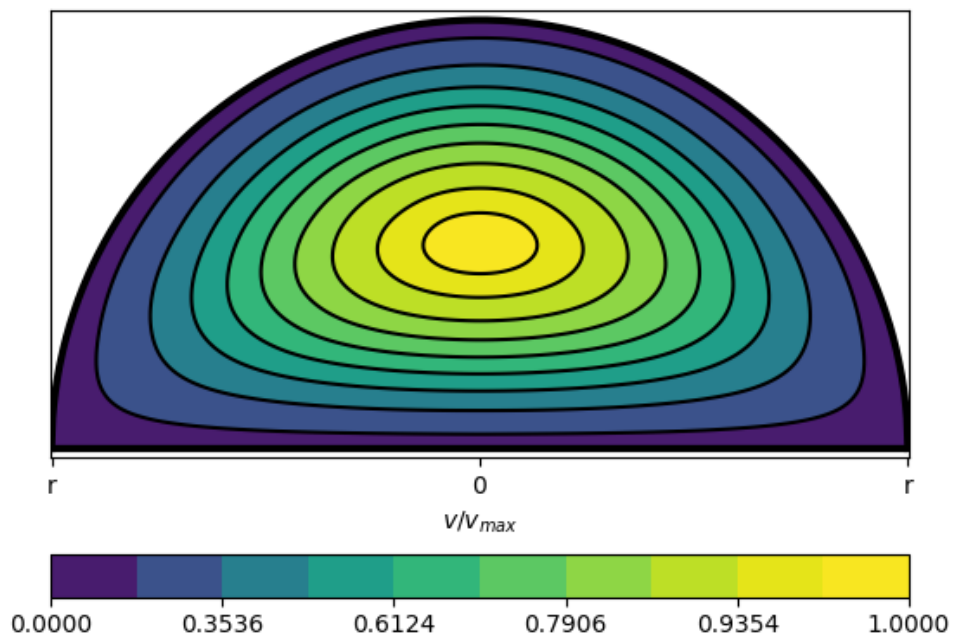
Slika 3: Približevanje referenčni vrednosti C , izračunani pri $N = M = 100$, pri različnih N in M . Prikazana je logaritmirana vrednost razlike med obema koeficientoma.

3 Hitrostni profil

Iz vektorja \mathbf{a} , dobimo tudi koeficiente za funkcije, ki predstavljajo hitrostni profil tekočine, ki se pretaka po polkrožni cevi. Dobil sem sledeče rezultate.



Slika 4: Hitrostni profil za polkrožno cev v kartezičnem koordinatnem sistemu. Prikazana je normirana hitrost v odvisnosti od kota (y-os) in oddaljenosti od izhodišča (radija, x-os).



Slika 5: Realni prikaz hitrostnega profila v polkrožni cevi (v polarnem koordinatnem sistemu). Ponovno je prikazana normirana hitrost.

4 Komentar

Nalogo bi označil, kot uspešno izvedeno, saj sem dobil smiselne rezultate za hitrostni profil v polkrožni cevi. Pri tem sem se naučil uporabljati nov tipa grafa (contour plot), ki se mi zdi za tovrstne predstavitve podatkov precej uporaben. Pri študiji natančnosti bi bilo smiselno pridobiti vrednost koeficienta za polkrožno cev iz kakšnega zunanjega vira, saj bi mi to omogočilo boljšo analizo rezultatov, vendar ga nisem uspel najti. Primerjava rezultatov samih s sabo, ki sem jo opravil, namreč ni najbolj smiselna oziroma, bo verjetno vedno dala dobre rezultate.