

# Matematično-fizikalni praktikum: Robni problem lastnih vrednosti

Andrej Kolar-Požun

December 7, 2016

# 1 Uvod

Reševal bom Gelfand-Bratujev robni problem:

$$y'' = -\delta e^y$$

z robnima pogoje  $y(0) = y(1) = 0$  za  $0 < \delta < 3.51$  ima problem rešitvi. Ena izmed njiju je:

$$y(x) = -2\ln\left(\frac{\cosh(\xi(1-2x))}{\cosh(\xi)}\right)$$

Kjer je  $\xi$  rešitev  $\cosh(\xi) = \sqrt{8/\delta}$

Iz enačbe vidimo, da bo naša rešitev konkavna (Drugi odvod je vedno negativen). Za začetni pogoj bom torej tudi vzel neko konkavno funkcijo in sicer  $u(x) = \mu x(1-x)$ . Preizkusil bom  $\mu = 1$  in  $\mu = 16$ .

## 2 Reševanje

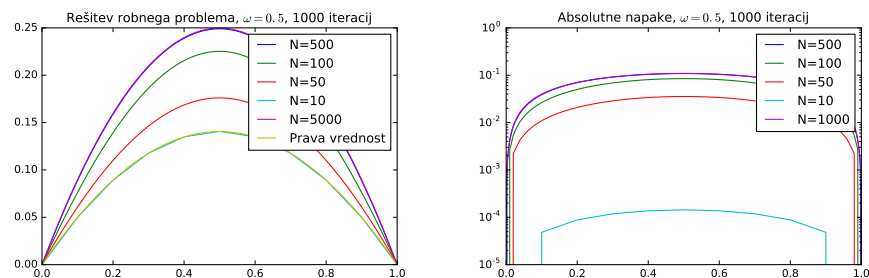
### 2.1 Eksplicitna iterativna metoda

Najprej bom uporabil tole iteracijsko shemo:

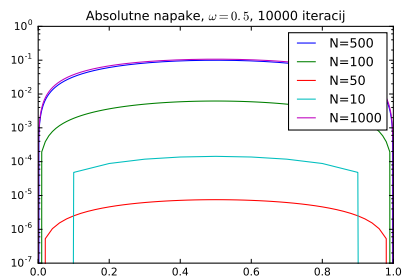
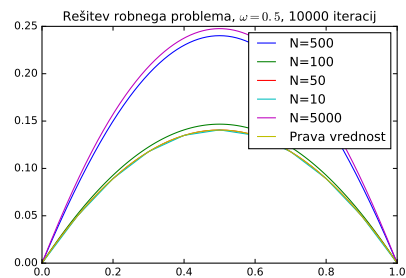
$$u_j^{(n+1)} = \frac{1}{1+\omega} \left[ \frac{1}{2}(u_{j+1}^{(n)} + u_{j-1}^{(n)}) + \omega u_j^{(n)} + \frac{h^2}{2} \delta e^{u_j^{(n)}} \right]$$

Začel bom z  $\mu = 1$  in  $\omega = 0.5$

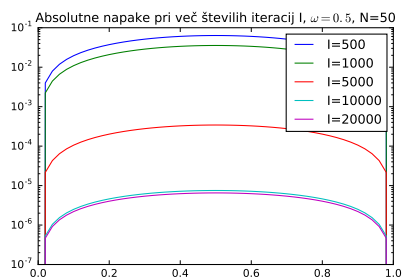
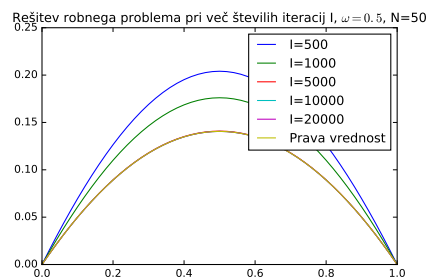
Prva stvar, katero sem opazil je, da je hitrost konvergence zelo odvisna od števila točk.



Vidimo, da se za  $N=10$  natančnost drastično izboljša napram  $N=100$ . Vseeno nočem imeti za risanje funkcije samo 10 točk, zato bom vajo ponovil z več iteracijami

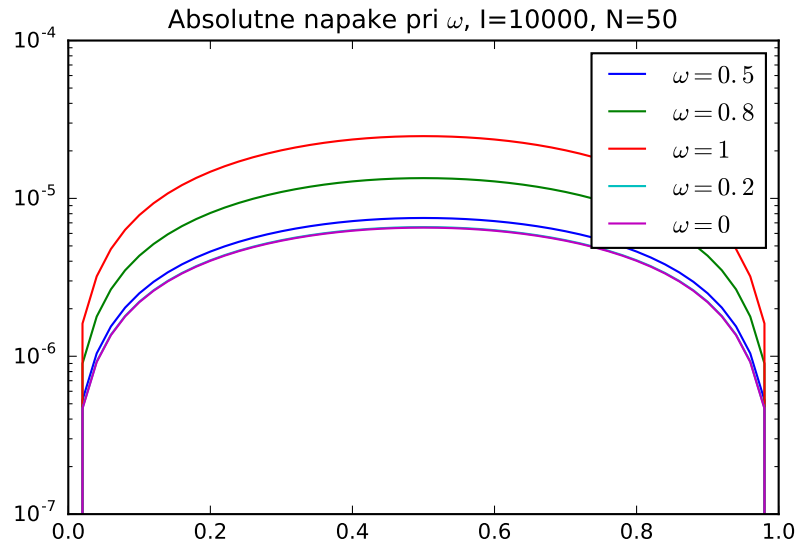


Bolje, opazimo tudi da je zdaj 50 točk natančnejših kot le 10.



Tukaj sem preverjal še optimalno število iteracij. Odločil sem se za 50 ekvidistančnih točk, kar se mi zdi dovolj pa še natančno je bilo. Vidimo, da je izboljšava, ki jo dobimo z 20000 iteracijami namesto s 10000 zelo majhna (traja pa že vidno več časa)

Primerjal bom še različne omega. Zanima me pri kateri bo natančnost največja pri fiksnem številu iteracij(10000).

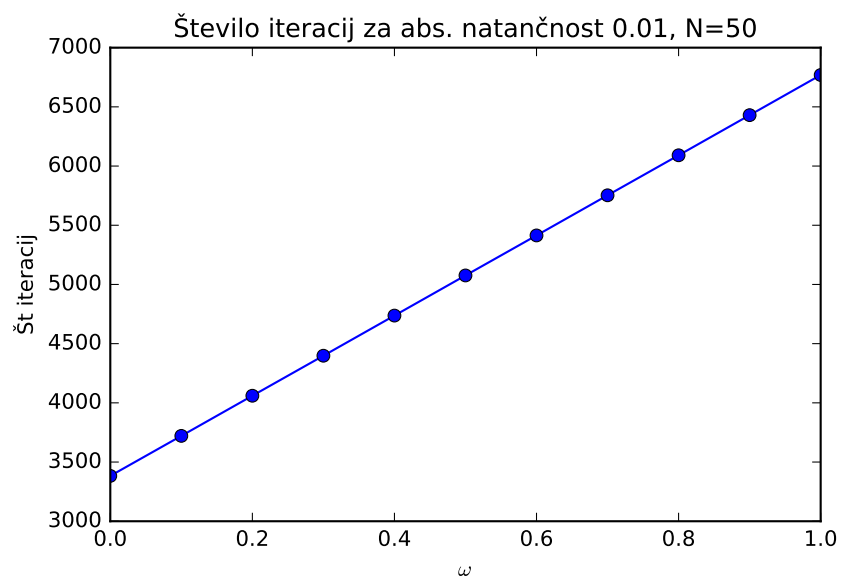


V tem primeru torej izgleda, da se najboljše obnese, če delamo kar brez omega (Metoda bolje deluje za nižje omega, najboljše če je enaka nič). Za negativne omega vrne "not a number" (Najbrž zdivergira)

Vseeno hočem videti, če je kaj prednosti pri večji omega. Zato bom sedaj preveril točno kako hitro konvergira pri različnih omega. In sicer tako, da bom gledal koliko iteracij je potrebnih, da je razlika med aproksimacijo in pravo vrednostjo recimo  $10^{-3}$ . Razliko bom gledal med vsotama vseh 50 vrednosti funkcije na intervalu.

$\omega$	št. iteracij
0	3383
0.1	3721
0.2	4060
0.3	4398
0.4	4737
0.5	5076
0.6	5414
0.7	5753
0.8	6091
0.9	6430
1	6769

Še grafično:



Odvisnost je torej linearna

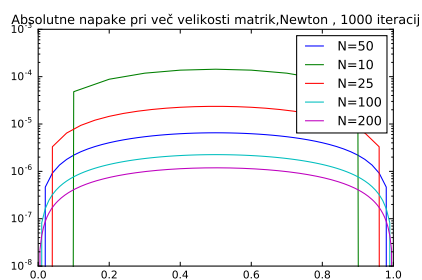
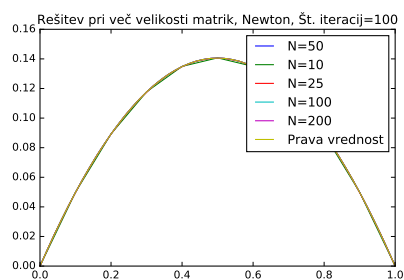
## 2.2 Newtonova metoda

Zdaj bom preizkusil še Newtonovo metodo. Newtonova metoda se uporablja za iskanje ničel, zato morem moj sistem enačb prepisat v obliko  $F(x) = 0$ :

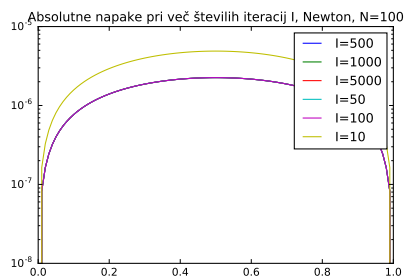
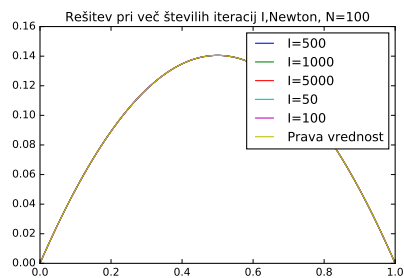
$$-\frac{u_{j+1} - 2u_j + u_{j-1}}{h^2} + f(u_j) = 0$$

rešujem sistem enačb:  $J(u^{(n)})\Delta u^{(n)} = -F(u^{(n)})$  Kjer je J jacobijeva matrika F-a.

Dobim naslednji približek:  $u^{(n+1)} = u^{(n)} + \Delta u^{(n)}$



Vidimo, da je metoda primerljivo natančna prejšnji metodi, vendar zdaj ne prevlada več za  $N=50$  ampak dalj časa raste z  $N$ -jem

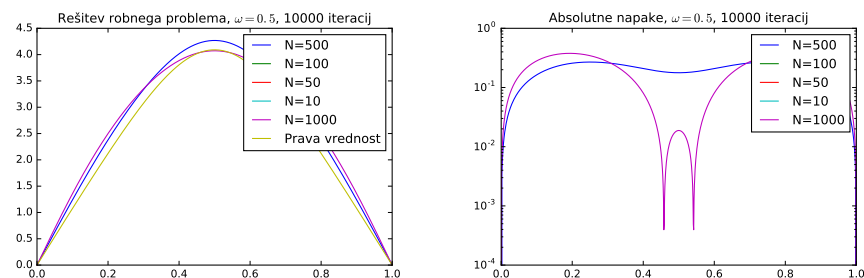


Natančnost najprej raste z večanjem števila iteracij, a se hitro ustavi pri neki točki, po kateri dodatne iteracije ne izboljšajo natančnosti.

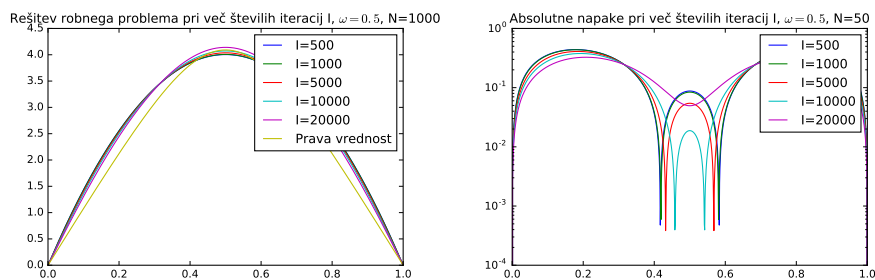
## 2.3 $\mu = 16$

Opremljen s temi dvema metodama bom zdaj poizkusil najti še drugo rešitev robnega problema. Vem, da ima ta višje funkcijske vrednosti in bom zato začel z višjim začetnim približkom in sicer  $u_0 = 16x(1 - x)$

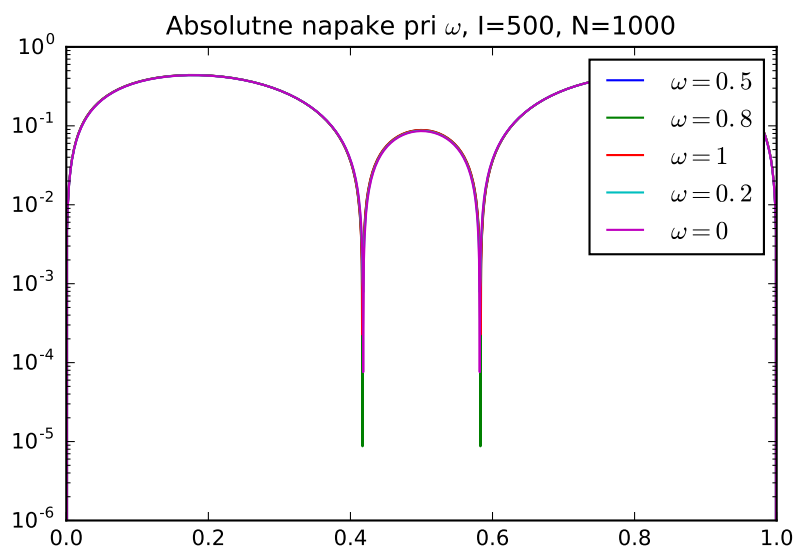
Najprej poskusimo z našo prvo iterativno shemo:



Napaka je vsekakor večja kot pri drugi rešitvi. Na sredini ima za  $N=1000$  tudi neko čudno obliko. Na sliki je viden le  $N=1000$  in  $N=500$ . Razlog za to je, ker je za manjše  $N$  pravzaprav metoda divergirala, in je seveda nisem mogel narisati

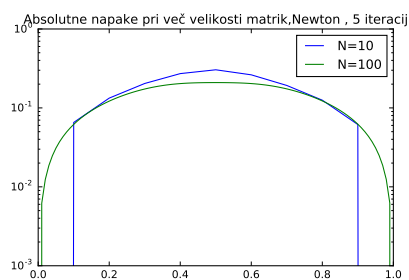
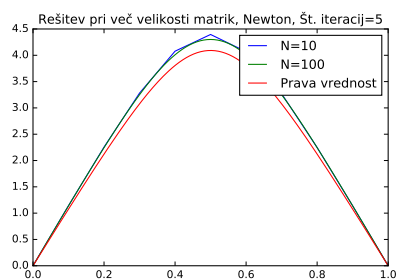


Vidimo, da neka razlika z napakami pri različnih številih iteracij je. Razlike so samo na manjšem območju nekje na sredini intervala



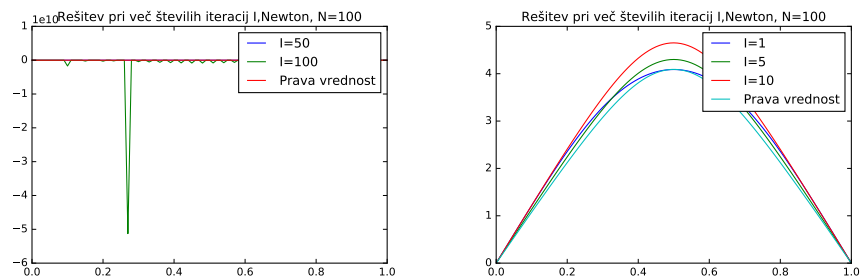
Tukaj pa vidimo, da je rešitev skorajda neodvisna od izbire omega. Pri omega = 0.8 imamo sicer nekoliko večjo natančnost vendar je ta "večja natančnost" na tako majhen območju, da v bistvu ni razlike. To je še ena razlika od prve rešitve, kjer je izbira različne omega spremenila napako metode.

Sedaj bom drugo rešitev poizkusil poiskati še z Newtonovo metodo:

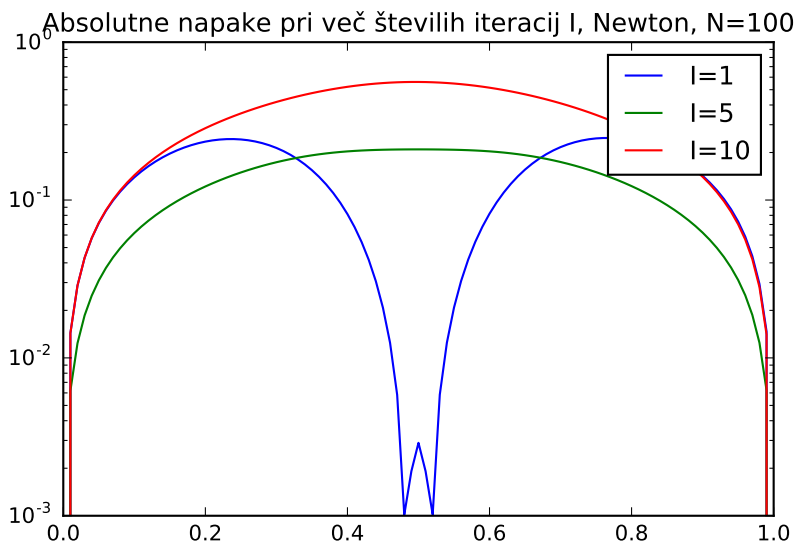


Tudi tukaj je za veliko  $N$ -jev (in večje število iteracij) metoda divergirala. Napaka pa je tudi veliko večja kot pri prvi rešitvi.





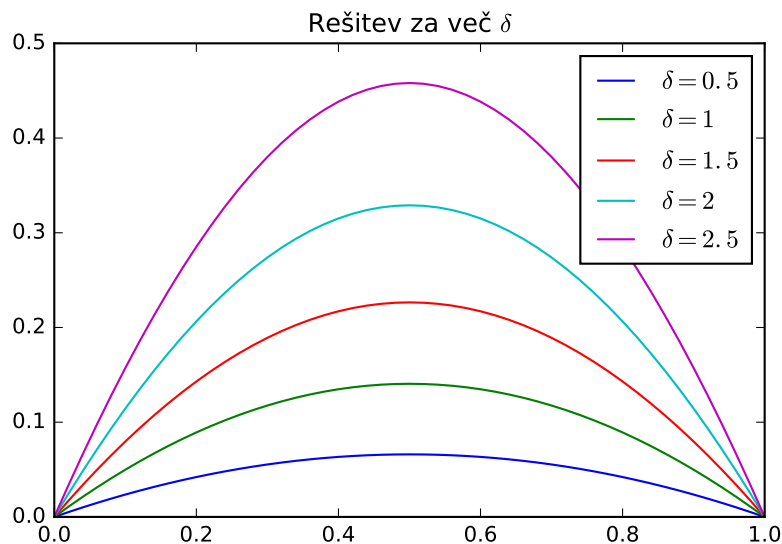
Na levi vidimo čudno obnašanje metode pri višjem številu iteracij (pri še višjem divergira). Na desni vidimo rešitve pri večjem številu iteracij



1 iteracija je že konkurentna po natančnosti večjem številu iteracij. Po 10 pa začne napaka samo rasti.

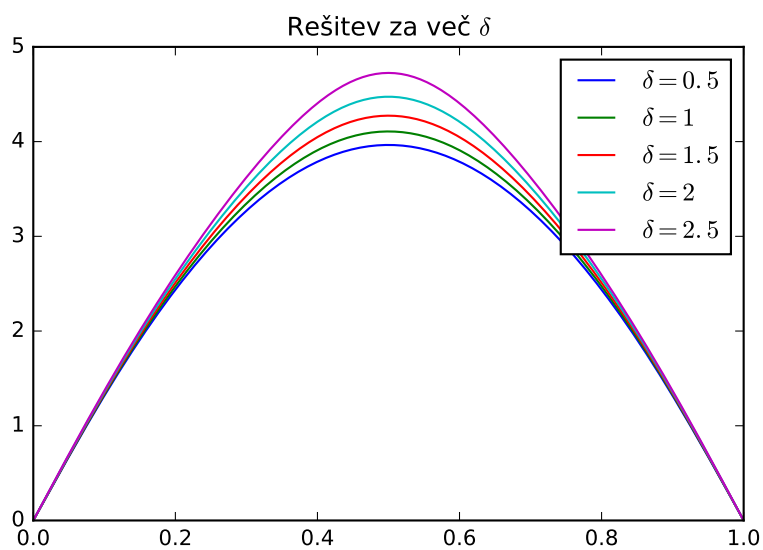
## 2.4 več $\delta$

Sedaj si bom pogledal še, kako rešitve izgledajo pri več različnih možnostih za parameter  $\delta$ . Uporabil bom prvo iterativno shemo z  $N=50$ ,  $\omega = 0$  in številom iteracij 10000, kot se je najbolj obneslo do sedaj. Najprej bom poskusil najti prvo rešitev z začetnim približkom  $u_0 = x(1 - x)$ :



Z večanjem delte je funkcija bolj izbočena in seže višje.

Sedaj pa še z začetnim približkom  $u_0 = 16x(1 - x)$ . Tokrat bom za N vzel 1000:



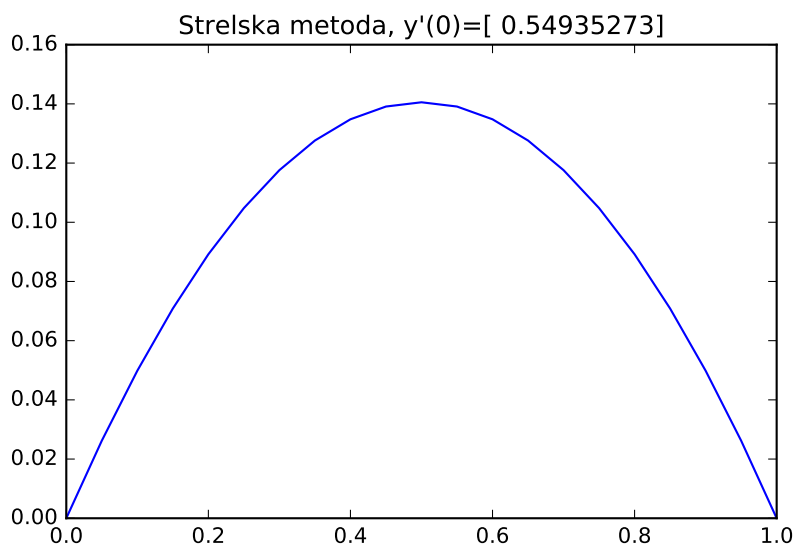
Spet je z večanjem delte je funkcija malce bolj izbočena in seže malce višje.

### 3 Dodatna naloga

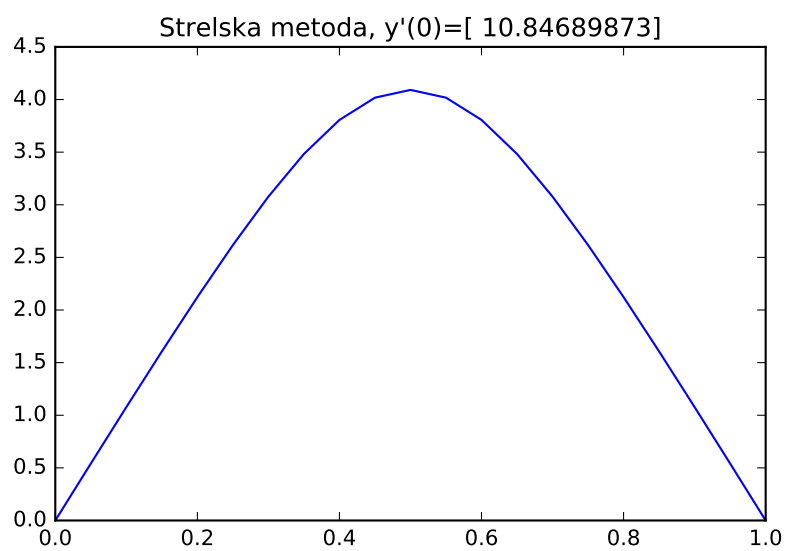
Sedaj bom problem poskusil rešiti še s strelsko metodo. Za reševanje z nekimi začetnimi pogoji bom uporabil metodo Runge Kutta s prilagodljivo velikostjo koraka. Naši robni pogoji so  $y(0) = y(1) = 0$ . Runge Kutta bom pognal s pogojem  $y(0) = 0$  in  $y'(0) = a$ , kjer je  $a$  neka uganjena vrednost odvoda. Runge Kutta bom potem pognal do  $y(1)$  in pogledal kakko blizu sem robnemu pogoju  $y(1)=0$ . a bom potem spreminjal toliko časa, dokler ne bo  $y(1)$  že blizu ničle. a bom poiskla s pomočjo scipy-jeve funkcije fsolve.

Najprej moram mojo diferencialno enačbo 2. reda pretvoriti v sistem enačb prvega reda:

$$\begin{aligned}y' &= p \\ p' &= -e^y\end{aligned}$$

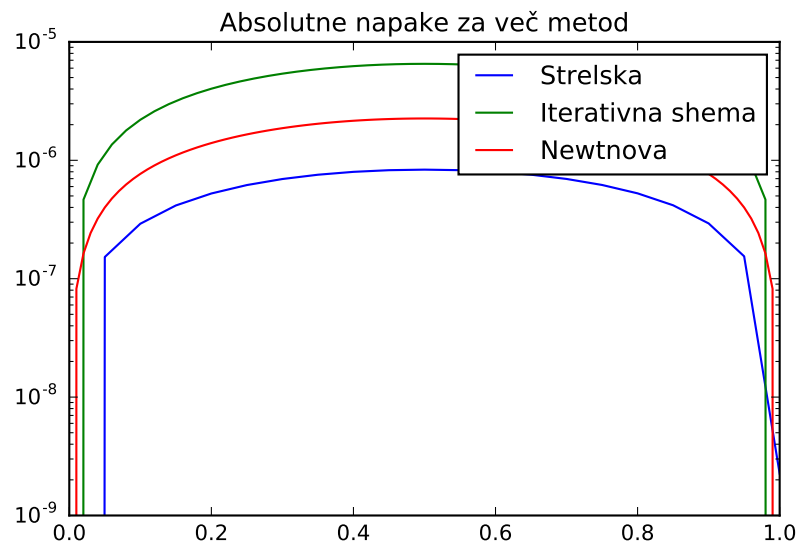


Izgleda, da deluje v redu. Vrnilo je prvo rešitev. Prva uganitev za odvod je bila v tem primeru 1.

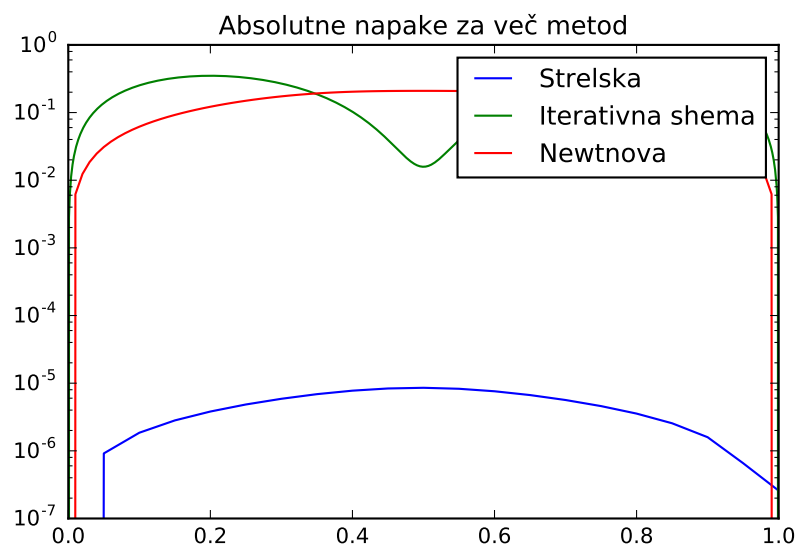


Dobil sem tudi drugo rešitev s tem, da sem za prvi približek za odvod izbral 10, saj vem, da druga rešitev na začetku stremeje narašča

Rad bi naredil samo še zadnjo primerjavo med tremi obdelanimi metodami:



Primerjava za prvo rešitev. Vidimo, da je strelska metoda najbolj natančna, iterativna shema pa najmanj. Kot parametre za iterativno shemo sem vzel  $N=50, I=10000, \omega = 0$ , za Newtonovo pa  $N=100, I=100$



Pri drugi rešitvi je strelska metoda zelo zelo dobra. Saj smo že prej videli, da sta se Newtonova in Iteracijska shema slabo obnesli. Parametre za iteracijsko sem tukaj vzel  $N=1000, I=10000, \omega = 0$ , za Newtonovo pa  $N=100, I=5$