

Univerza v Ljubljani
Fakulteta za *matematiko in fiziko*



Oddelek za fiziko

MODELSKA ANALIZA I

Poročila opravljenih vaj

Jernej CUCEK
28172019

MENTOR: Dr. Simon Čopar
Ljubljana, 2017/2018

Kazalo

1. Model vožnje skozi semafor: variacijska metoda	7
1.1. Uvod	8
1.2. Naloga	8
1.3. Brezdimenzijska oblika	8
1.4. Rešitev brez enega robnega pogoja	9
1.5. Rešitev z omejitvijo končne hitrosti	10
1.6. Funkcional z višjimi potencami pospeška	11
1.7. Kvadrat hitrosti v funkcionalu	12
1.8. Periodična rešitev	13
Literatura	14
2. Linearno programiranje	17
2.1. Uvod	18
2.2. Naloga	18
2.2.1. Minimizacija vnosa kalorij	19
2.2.2. Minimizacija vnosa maščob	21
Literatura	21
3. Numerična minimizacija	25
3.1. Uvod	26
3.2. Naloga	26
3.3. Thomsonov problem	27
3.4. Problem optimalne vožnje skozi semafor	30
Literatura	30
4. Populacijski modeli	35
4.1. Uvod	36
4.2. Naloga	36
4.3. Zajci in lisice	37
4.4. Laser	41
4.5. Epidemija	44
Literatura	45
5. Modeli kemijskih reakcij	49
5.1. Naloga	50
5.2. Model binarne reakcije	51
5.3. Reakcija z več stopnjami	51
5.4. Kemijske ure	55
Literatura	58
6. Luščenje modelskih parametrov: linearni modeli	61
6.1. Naloga	62
6.2. Farmakološki model	63
6.3. Varčni model preslikave visokoločljivostnega magnetnega spektrometra	64
6.4. Rentgenski absorpcijski spekter kadmija v rastlini <i>C.Thalpsi</i>	66

Literatura	66
7. Luščenje modelskih parametrov: razdelčni in nelinearni modeli	71
7.1. Naloga	72
7.2. Farmakološki model	72
7.3. Čistilnost ledvic	73
7.3.1. Enorazdelčni model z linearizacijo	73
7.3.2. Enorazdelčni model z ozadjem	74
7.3.3. Dvorazdelčni modeli	75
7.4. Korozija med kovino in elektrolitom	75
Literatura	76
8. Generatorji slučajnih števil	81
8.1. Naloga	82
8.2. Generatorji gaussovskih slučajnih števil	83
8.3. Generator naključnih smeri v prostoru	87
8.4. Generator dipolnega sevanja fotonov	87
8.4.1. Porazdelitev časov oddaj nalog iz Modelske analize	89
A. Kolmogorov-Smirnov testi	93
B. χ^2 testi	95
Literatura	95
9. Integracije z metodo Monte Carlo	99
9.1. Naloga	100
9.2. Volumen in vztrajnostni moment skupnega preseka treh enakih pravokotnih valjev	100
9.2.1. Homogeno telo	100
9.2.2. Telo s spremenljivo gostoto	104
9.3. Pobeg fotonov iz krogle	106
9.4. Model nevtronskega reflektorja	107
Literatura	108
10. Metropolisov algoritem	113
10.1. Naloga	114
10.2. Molekularna verižnica	115
10.3. Dvodimenzionalni Isingov model	116
10.4. Problem trgovskega potnika	119
Literatura	119
11. Stohastični populacijski modeli	123
11.1. Naloga	124
11.2. Statistika časov izumrtja populacije	124
11.3. Matrika prehodov	127
11.4. Stohastični model zajcev in lisic	128
Literatura	129
12. Spektralna analiza in filtriranje	133
12.1. Naloga	134
12.2. Določitev frekvenčnega spektra signalov	135
12.2.1. Diskretna Fourierova transformacija	135
12.2.2. Hitra Fourierova transformacija	136
12.2.3. Frekvenčni spekter signalov val2.dat in val3.dat	136

12.3. Dekonvolucija zašumljenih signalov	139
12.4. Čiščenje Lincolnove slike	141
Literatura	141
13. Metoda maksimalne entropije in linearne napoved	145
13.1. Naloga	146
13.2. Uvod	146
13.3. Frekvenčni spekter vzorcev val2.dat, val3.dat in co2.dat	147
13.3.1. Ločljivost metode	150
13.4. Linarne napovedi	152
13.4.1. Vpliv šuma na linearno napoved	153
Literatura	153

1. Model vožnje skozi semafor: variacijska metoda

Povzetek

Pri tej vaji modeliramo varčno vožnjo skozi semafor in sicer tako, da z variacijsko metodo minimiziramo kumulativni kvadrat pospeška. Preko Euler-Lagrangejeve enačbe dobimo različne analitične rešitve za primere z različnimi robnimi pogoji. Ogledamo si tudi rešitve za drugačne funkcione, ki poleg pospeška vključujejo tudi omejitve v hitrosti.

1.1. Uvod

Varčno vožnjo lahko definiramo s pogojem, da je pospeševanja in zaviranja čim manj. To lahko dosežemo z minimizacijo kumulativnega kvadrata pospeška. Iščemo optimalni režim vožnje v situaciji, ko poskušamo razdaljo do semaforja prevoziti ravno v trenutku, ko se prižge zelena luč.[1]

1.2. Naloga

- Prepiši problem v brezdimenzijsko obliko.
- Variacijski račun daje možnost, da določimo ekstrem tudi brez (enega) robnega pogoja: temu je ekvivalentna zahteva $d\mathcal{L}/d\dot{y} = 0$ v drugi robni točki. Kakšna je torej optimalna vožnja pri poljubni končni hitrosti?
- Kaj dajejo še druge možnosti izbire funkcionala, na primer višje sode potence pospeška? Zanimiv praktični pomen ima limita, ko gre eksponent proti neskončnosti.
- Kakšna je pa optimalna vožnja, če poleg pospeška tudi velikost hitrosti omejimo s kvadratičnim členom v funkcionalu? Koliko je zdaj prostih parametrov problema?
- Zanimiv problem je tudi iskanje periodične rešitve, kjer mora biti končna hitrost enaka začetni. Razisči dobljeno družino rešitev. Kako bi zagotovil zveznost pospeška na periodičnem stiku?

Komentar k reševanju

Naloga je rešena v programske okoljih »Mathematica« in »Python«, pri slednjem je povsem zadostovala uporaba knjižnice `numpy`.

1.3. Brezdimenzijska oblika

Pri problemu vožnje imamo opraviti predvsem s hitrostjo $v(t)$ in njenimi časovnimi odvodi, katere obtežimo z različnimi koeficienti. Vse te člene, vključno s potjo s in časom t , lahko minimiziramo ali maksimiziramo, kar je odvisno od potreb modela, ki si ga zamislimo. Ker problem rešujemo variacijsko s pomočjo Euler-Lagrangejevega multiplikatorja preko enačbe

$$\frac{d}{dt} \frac{\partial \mathcal{L}}{\partial \dot{v}} - \frac{\partial \mathcal{L}}{\partial v} = 0 , \quad (1.1)$$

je potrebno definirati akcijo \mathcal{S} in si izbrati še željeno Lagrangejevo funkcijo \mathcal{L} , da lahko minimiziramo

$$\mathcal{S} = \int_0^T \mathcal{L} dt = \min .$$

Za določitev optimalne vožnje skozi semafor si kot vez izberemo Lagrangejevo funkcijo

$$\mathcal{L} = \dot{v}^2 - \lambda v , \quad (1.2)$$

kjer v drugi člen vpnemo pogoj oddaljenosti avtomobila do semaforja L kot

$$\int_0^T v(t) dt = L . \quad (1.3)$$

Hitrost, ki stoji poleg Lagrangevega multiplikatorja λ , se torej prilagaja razdalji do semaforja in zahtevanem času T v katerem mora avtomobil to razdaljo prevoziti, da ravno ujame prižig zelene luči. Kvadratno odviosnost pospeška v Lagranžijanu pa vzamemo zato, ker linearna funkcija $\dot{v}(t)$ ni primerna, saj ni omejena, mi pa iščemo minimum funkcionala.

Za prepis problema v brezdimenzijsko obliko definiramo hitrost in čas kot

$$y = \frac{v}{L/T} \quad , \quad x = \frac{t}{T}$$

in zapišemo Lagranžian in vez z brezdimenzijskimi spremenljivkami

$$\mathcal{L} = \dot{y}^2 - \lambda y \quad , \quad \int_0^1 y(x) dx = 1 .$$

Vidimo lahko, da hitrost $y = 1$ ustreza enakomerni hitrosti, pri kateri bi prišli do semaforja ravno ob pravem času $x = 1$. Tak izraz za \mathcal{L} lahko vstavimo v Euler-Lagrangeeve enačbo (1.1) in dobimo rešitve

$$2\ddot{y} + \lambda = 0 \quad \rightarrow \quad \dot{y}(x) = -\frac{\lambda}{2}x + A \quad \rightarrow \quad y(x) = -\frac{\lambda}{4}x^2 + Ax + B .$$

Za konec iz robnih pogojev določimo še konstanti A in B ter Lagrangejev multiplikator. Iz pogoja za začetno hitrost $y(0) = y_0$ dobimo B . Ob času $t = T$ oziroma $x = 1$ pa imamo $y(1) = y_T$, od koder izpostavimo A . Lagrangev multiplikator λ določimo z brezdimenzijsko obliko pogoja (1.3).

1.4. Rešitev brez enega robnega pogoja

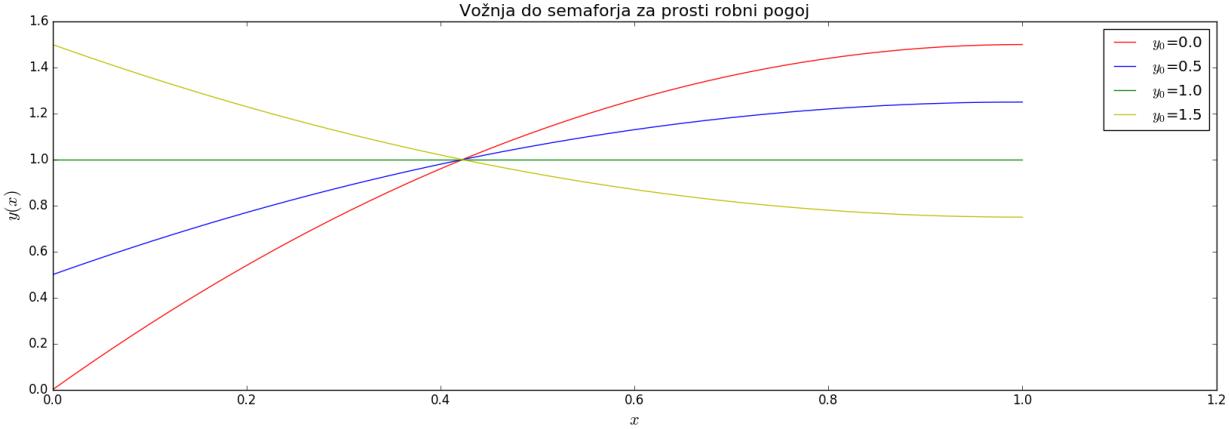
Končo rešitev pri tako izbranih robnih pogojih dobimo ob upoštevanju $d\mathcal{L}/d\dot{y} = 0$ v drugi robni točki, ko dosežemo semafor. Iz začetnega pogoja $y(0) = y_0$ določimo B , pri $x = 1$ pa dobimo A kot

$$2\ddot{y}(1) = 0 \quad \rightarrow \quad \dot{y}(1) = -\frac{\lambda}{2}x + A = 0 \quad \rightarrow \quad A = \frac{\lambda}{2} .$$

Lagrangev multiplikator $\lambda = 6(1 - y_0)$ določimo preko brezdimenzijske oblike pogoja (1.3) in že lahko zapišemo

$$y(x) = 3 \left(x - \frac{x^2}{2} \right) (1 - y_0) + y_0 .$$

Tako y_0 je edini parameter, ki nam določa družino rešitev problema. Nekaj rešitev za različne vrednosti parametra je prikazanih na sliki 1.1. Za primer, ko je $y_0 = 1$, dobimo ravno črto, katere integral poda pot, ki je skladna s pogojem (1.3). Ko je $(1 - y_0) = 0$ nas torej vožnja z nespremenjeno hitrostjo pripelje skozi semafor ob ravno pravem času. Za večje in manjše vrednosti y_0 se krivulje prilagodijo, tako, da se površina pod krivuljo hitrosti ohranja, končne hitrosti za različne y_0 pa se med seboj razlikujejo, kot to dopušča dinamični robni pogoj. Vse tri krivulje se neodvisno od izbire parametra y_0 sekajo v vozliščni točki, kjer je $y(x) = 1$.

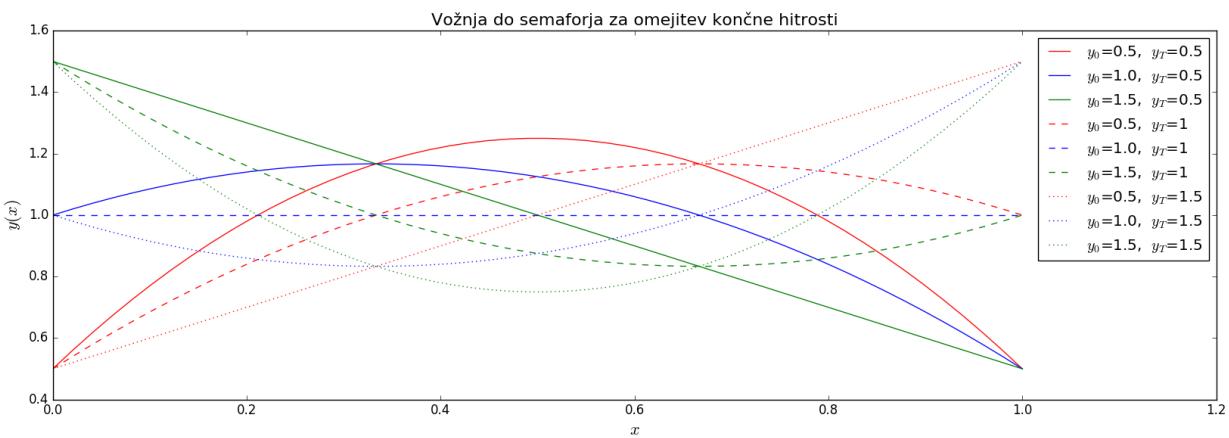


Slika 1.1.: Graf odvisnosti hitrosti od časa za primer brez končnega robnega pogoja in za različne vrednosti začetne hitrosti y_0 . Videti je, da je v tem primeru robni pogoj zares dinamičen, saj se končne hitrosti za različne y_0 med seboj razlikujejo. Na vsaki krivulji grafu je prisotna vsaj ena točka, kjer je $y(x) = 1$. Izgleda, da je ta neodvisna od izbire parametra y_0 . Opazi se še, da se ohranjanje površine pod krivuljami, ki predstavljajo prepotovano pot, kar je v skladu s pogojem (1.3).

1.5. Rešitev z omejitvijo končne hitrosti

Iz takih robnih pogojev spet določimo konstanti A in B ter Lagrangejev multiplikator. Iz pogoja za začetno hitrost dobimo $y(0) = y_0 = B$. Ob času $t = T$ oziroma $x = 1$ pa za končno hitrost dobimo $y(1) = y_T = -\frac{\lambda}{4} + A + B$ od koder izpostavimo $A = \frac{\lambda}{4} + y_T - y_0$. Lagrangev multiplikator $\lambda = 12(2 - y_T - y_0)$ določimo z brezdimenzijsko obliko pogoja (1.3). Končo družino rešitev problema pri tako izbranih robnih pogojih tako določata parametra y_0 in y_T .

Izračuni za nekaj različnih vrednosti parametrov so prikazanih na sliki 1.2. Tokrat hitrost doseže izbrano končno vrednost fiksnega robnega pogoja, po kateri poti to točko dosežemo pa je odvisno od začetne hitrosti y_0 , da se le ohranjanje površine pod krivuljami, ki predstavljajo prepotovano pot v skladu s pogojem (1.3). Tokrat so vozlišča prisotna v posamezni družini krivulj z enakim y_T ali y_0 .



Slika 1.2.: Graf odvisnosti hitrosti od časa za primer z različnimi fiksnnimi končnimi robnimi pogoji y_T in za različne vrednosti začetne hitrosti y_0 . Videti je, da v tem primeru hitrost doseže izbrano končno vrednost, poteki krivulj pa so odvisni od y_0 in med seboj razlikujejo. V družini krivulj z enakim y_T ali y_0 je prisotna vozliščna točka, v kateri se pripadajoče krivulje sekajo. Tudi tu je opaziti, da se ohranjanje površine pod krivuljami, ki predstavljajo prepotovano pot v skladu s pogojem (1.3).

1.6. Funkcional z višjimi potencami pospeška

Za primer višje potence absolutne vrednosti pospeška se omejimo le na sode eksponente, ki tudi brez absolutne vrednosti garantirajo obstoj minimuma. V tem primeru si kot vez izberemo Lagrangejevo funkcijo oblike

$$\mathcal{L} = |\dot{y}|^{2p} - \lambda y , \quad (1.4)$$

kjer je $p = 1, 2, \dots$ pozitivno celo število. Euler-Lagrangeeva enačba poda zvezo

$$\lambda + 2p(2p-1)\dot{y}^{2p-2}\ddot{y} = 0.$$

od koder z integriranjem dobimo izraz za pospešek

$$\dot{y}(x) = \left(A - \frac{\lambda}{2p} x \right)^{\frac{1}{2p-1}}$$

in nato še za hitrost

$$y(x) = \frac{1-2p}{\lambda} \left(A - \frac{\lambda}{2p} x \right)^{\frac{2p}{2p-1}} + B .$$

Ob upoštevanju dinamičnega robnega pogoja $d\mathcal{L}/d\dot{y} = 0$ in začetnega pogoja lahko izračunamo še konstanti

$$A = \lambda/2p \quad \text{in} \quad B = y_0 - \frac{1-2p}{\lambda} \left(\frac{\lambda}{2p} \right)^{\frac{2p}{2p-1}} ,$$

s pogojem (1.3) pa

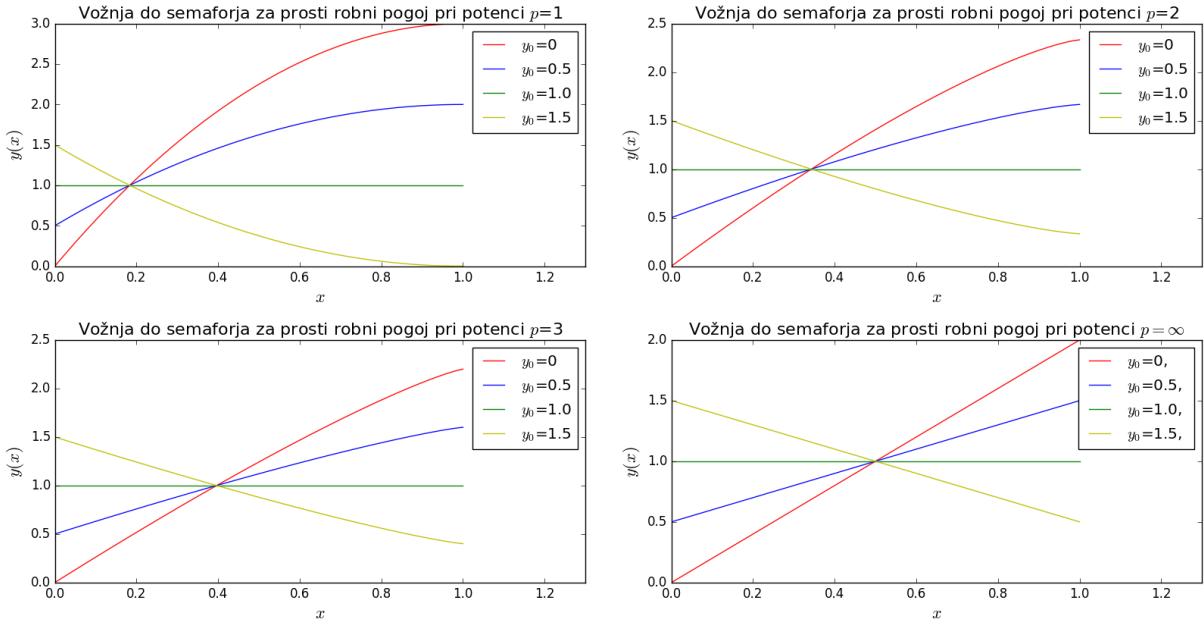
$$\left(\frac{\lambda}{2p} \right)^{\frac{1}{2p-1}} = (1-y_0) \frac{4p-1}{2p-1} .$$

Tako za naš primer dobimo končno obliko odvisnosti hitrosti

$$y(x) = \frac{(4p-1) \cdot (1-y_0)}{2p-1} \left(1 - (1-x)^{\frac{2p}{2p-1}} \right) + y_0 ,$$

ki se v limiti $p \rightarrow \infty$ prepiše v $y(x) = 2x(1-y_0) + y_0$. Limitna rešitev je torej enakomerno pospešeno gibanje, sama velikost pospeška pa je odvisna od začetne vrednosti y_0 .

Izračuni za različne y_0 in različne robne pogoje so prikazani na sliki 1.3. Opaziti je, da se z večanjem potence p vse bolj približujemo ravni črti, torej vedno bolj omejujemo spremembo hitrosti.



Slika 1.3.: Graf odvisnosti hitrosti od časa za primer 1.4 brez končnega robnega pogoja in pri različnih vrednostih začetne hitrosti y_0 za štiri primere potence p . Z večanjem potence p se krivulje vse bolj približujejo ravni črti, ki je dosežena v $p \rightarrow \infty$.

1.7. Kvadrat hitrosti v funkcionalu

Če poleg pospeška s kvadratičnim členom v funkcionalu omejimo tudi hitrost

$$\mathcal{L} = \dot{y}^2 + \mu^2 y^2 - \lambda y \quad (1.5)$$

dobimo dodatni prosti parameter μ . Z vstavitvijo v (1.1) dobimo enačbo

$$\ddot{y} - \mu y + \frac{\lambda}{2} = 0 ,$$

ki nam z integriranjem poda izraz za hitrost.

$$y(x) = \frac{\lambda}{2\mu^2} + A e^{\mu x} + B e^{-\mu x} .$$

Ob upoštevanju dinamičnega robnega pogoja $d\mathcal{L}/d\dot{y} = 0$ in začetnega pogoja izračunamo še konstanti

$$A = \left(y_0 - \frac{\lambda}{2\mu^2} \right) \frac{1}{1 + e^{2\mu}} \quad \text{in} \quad B = \left(y_0 - \frac{\lambda}{2\mu^2} \right) \frac{e^{2\mu}}{1 + e^{2\mu}} ,$$

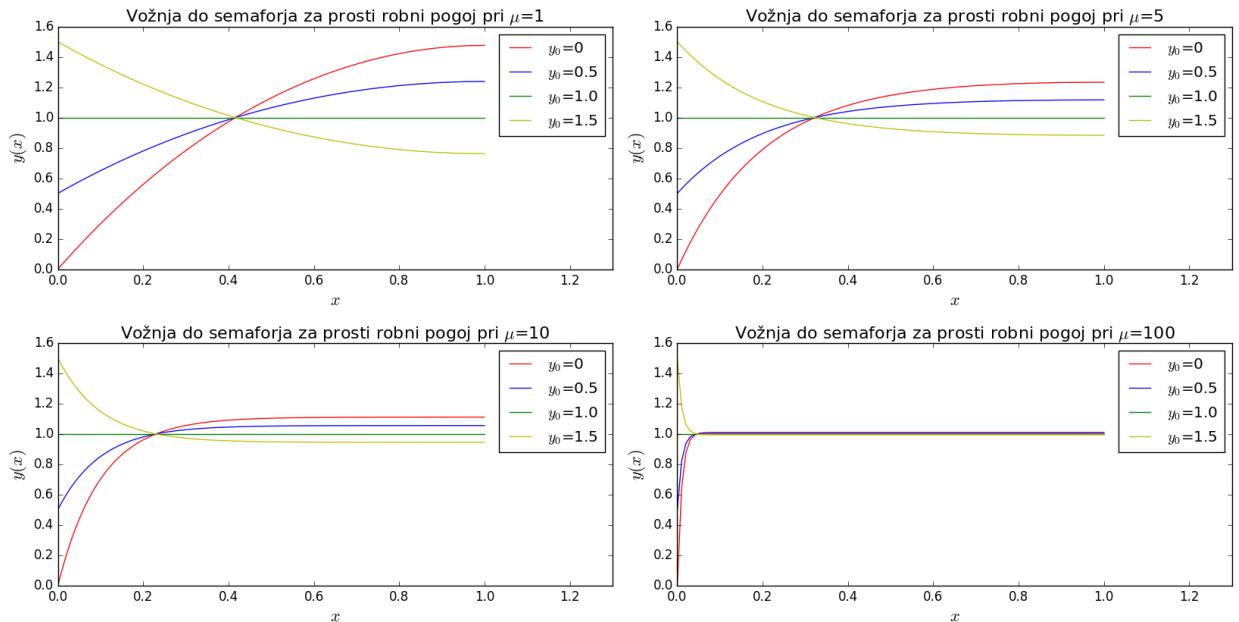
s pogojem (1.3) pa

$$\frac{\lambda}{2\mu^2} = \frac{\mu - y_0 \tanh \mu}{\mu - \tanh \mu} .$$

Tako za naš primer dobimo končno obliko odvisnosti hitrosti

$$y(x) = \frac{\mu - y_0 \tanh \mu}{\mu - \tanh \mu} + \left(y_0 - \frac{\mu - y_0 \tanh \mu}{\mu - \tanh \mu} \right) \frac{e^{\mu x} + e^{-\mu x} e^{2\mu}}{1 + e^{2\mu}} .$$

Izračuni za različne y_0 in različne robne pogoje so prikazani na sliki 1.4. Rezultati za majhno vrednost parametra μ so precej podobni izračunom za primer (1.2) z dinamičnim robnim pogojem. Pri večjih μ hitrost z vedno večim pospeškom pade na konstantno vrednost $y(x) = 1$, saj minimiziranje hitrosti preglaši minimizacijo pospeška.



Slika 1.4.: Graf odvisnosti hitrosti od časa za primere z različnimi multiplikatorji μ kvadratičnega člena hitrosti v funkcionalu (1.5) in za različne vrednosti začetne hitrosti y_0 ob prostem končnem robnem pogoju. Opaziti je mogoče, da se hitrost z večanjem μ hitreje približa vrednosti $y = 1$, pri kateri nato vztraja do semaforja.

1.8. Periodična rešitev

Za primer periodični primer, ko nas zanima vožnja skozi več semaforjev, je potrebno kot začetni robni pogoj ene periode nastaviti končno hitrost predhodne periode $y_{i+1}(0) = y_i(1)$. Vsakemu časovnemu intervalu med semaforjema tako ustreza rešitev, ki upošteva

$$\int_{T_i}^{T_{i+1}} v(t) dt = L_i . \quad (1.6)$$

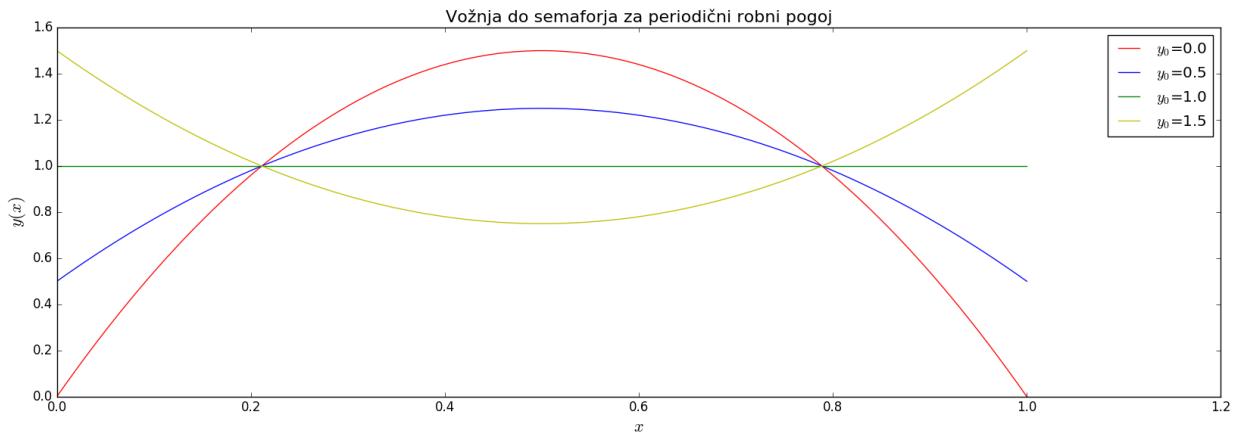
Za odsek prvega semaforja $i = 0$, ko je $T_0 = 0$, dobimo rešitev že izpeljane brezdimenzijske oblike z vrednostmi koeficientov

$$A = \frac{\lambda}{4} , \quad B = y_0 \quad \text{in} \quad \lambda = 24(1 - y_0) ,$$

ki jo zapišemo kot

$$y(x) = 6x(1-x)(1-y_0) + y_0 .$$

Izračuni za različne y_0 oz. različne robne pogoje so prikazani na sliki 1.5. Krivulje, ki opisujejo časovno odvisnost hitrosti, so okrog časa $t = 0.5$ simetrične parabole z istimi vrednostmi v začetni in končni točki.



Slika 1.5.: Graf odvisnosti hitrosti od časa za primer z istimi fiksнимi končnimi robnimi pogoji $y_T = y_T$ in za različne vrednosti začetne hitrosti y_0 . Sami poteki krivulj so odvisni od y_0 in se med seboj razlikujejo vendar je videti, da v periodičnem primeru hitrost res doseže končno vrednost enko začetni. Površine pod krivuljami, ki predstavljajo prepotovanje pot v skladu s pogojem (1.3), se ohranjanju.

Literatura

- [1] Spletne strani z gradivom predmeta, ogled 26.10.2017,
<http://predmeti.fmf.uni-lj.si/modelska>

2. Linearno programiranje

Povzetek

Pri tej vaji se preiskusimo pri aplikaciji metode linearnega programiranja. Pri tem želimo iz podatkov o posameznih živilih sestaviti smiselno dieto. Pri dieti omejimo vnos maščob ali energije ob različnih omejitvah kombinacij hrani in količine skupne zaužite hrane oziroma posamezne hrane iz seznama.

2.1. Uvod

Med tipične primere, ki jih lahko učinkovito rešimo z metodami linearnega programiranja, sodi sestavljanje diet za hujšanje, zdravljenje ali športne aktivnosti. Za dani nabor živil določamo njihove količine, pri čemer moramo zadostiti različnim omejitvam. Med drugim moramo zagotoviti priporočene dnevne odmerke mineralov, vitaminov in hranilnih snovi, omejiti pri vnos maščob, ogljikovih hidratov ter telesu škodljivih snovi, hkrati pa zagotoviti, da energijska vrednost ustreza zahtevam posameznika. Vnos vsake izmed hranilnih snovi je linearna funkcija količin živil in je načinko določena z njihovo sestavo. Od vrste diete pa je odvisno, katere parametre omejimo in katere minimiziramo.[1]

2.2. Naloga

- Datoteka **tabela – zivil.dat**[2] vsebuje podatke o energijski vrednosti ter vsebnosti maščob, ogljikovih hidratov, proteinov, kalcija in železa v nekaj živilih. Minimiziraj število kalorij, če je priporočen minimalni dnevni vnos 70 g maščob, 310 g ogljikovih hidratov, 50 g proteinov, 1000 mg kalcija ter 18 mg železa. Upoštevaj tudi, da naj dnevni obroki količinsko ne presežejo dveh kilogramov hrane. Upoštevate lahko še minimalne vnose za vitamin C (60 mg), kalij (3500 mg) in natrij (2400 mg), ki so tudi na voljo v tabeli.
- Kako se rezultat razlikuje, če zahtevamo minimalno 2000 kcal in namesto energije minimiziramo vnos maščob?
- Ker rešujemo poenostavljen problem z malo parametri na živilo, so lahko rezultati nerealistični. Lahko z omejitvijo količine posameznih živil v obroku izboljšaš uravnovešenost prehrane?
- (neobvezno) Poišči cene živil in poskusi namesto kalorij minimizirati ceno. Kako se varčevanje odraža na zdravi prehrani?

Komentar k reševanju

Linearno programiranje: Linearni optimizacijski problem formuliramo kot optimizacijo funkcije

$$f(x_j) = \sum_j a_{0j}x_j = ekstrem \quad (2.1)$$

pod pogoji

$$\sum_j a_{ij}x_j \begin{cases} = & b_i \\ \leqslant & b_i \\ \geqslant & b_i \end{cases} . \quad (2.2)$$

Iskane spremenljivke x_j so v večini paketov privzeto omejene na pozitivne vrednosti. Rutine za linearne programiranje so v različnih zbirkah med drugim na voljo pod imeni **simplex** (Numerical Recipes), **LinearProgramming** (Mathematica), **linprog** (matlab) in v knjižnici **GLPK**, ki pride tudi s samostojnjim programom **glpsol**. Slednji sprejema vrsto različnih formatov za deklaracijo linearnih problemov (na primer **CPLEXLP**). V Pythonu je linearne minimizacijska metoda na voljo v modulu **scipy.optimize**, ki jo kličemo s funkcijo **linprog**. V nalogi je uporabljena ravno slednja metoda. [1]

2.2.1. Minimizacija vnosa kalorij

Sprva sem minimiziral le število vnešenih kalorij ob navedenih omejitvah na 70 g maščob, 310 g ogljikovih hidratov, 50 g proteinov, 1000 mg kalcija ter 18 mg železa, in dobil rezultat:

Pomfri	= 493.805314989 g
Zelje	= 1312.18433152 g
Brokoli	= 200.853706617 g
Pomaranca	= 706.970890978 g
Kakav	= 57.2237524539 g
skupna masa živil	= 2771.03799656 g
skupna energija živil	= 1318.89399795 kcal

Nato sem račun ponovil z dodanimi omejitvami za vitamin C (60 mg), kalij (3500 mg) ter natrij (2400 mg), in dobil:

Pomfri	= 373.935433016 g
Piscanec	= 55.122187512 g
Mleko	= 80.4263749339 g
Sir_edamec	= 29.7256888281 g
Marmelada	= 252.065470282 g
Med	= 7.42440008418 g
Kakav	= 97.3768851427 g
Radenska	= 2228.19610572 g
skupna masa živil	= 3124.27254552 g
skupna energija živil	= 1428.09942597 kcal

Ker je bila potrebna količina hrane prevelika, sem za oba primera izračun ponovil z dodano omejitvijo količine hrane na dva kilograma. Novi rezultat za prvi primer je:

Pomfri	= 373.969956178 g
Sir_edamec	= 32.9541670636 g
Solata	= 106.975655288 g
Pomaranca	= 1370.37165342 g
Marmelada	= 12.7800314056 g
Kakav	= 102.948536644 g
skupna masa živil	= 2000.0 g
skupna energija živil	= 1394.76220003 kcal

in

Pomfri	= 314.824757623 g
Piscanec	= 23.2623389083 g
Mleko	= 194.560167574 g
Sir_edamec	= 39.7868367868 g
Solata	= 78.7190641534 g
Marmelada	= 262.869067719 g
Kakav	= 104.34902403 g
Sol	= 1.39641717133 g
Radenska	= 980.232326034 g
skupna masa živil	= 2000.0 g
skupna energija živil	= 1474.45192066 kcal

za primer z vitaminom C in kalijem ter natrijem.

Ker so rezultati še vedno nerealistični sem še enkrat računal še z grobimi omejitvami posameznih živil: žita, krompir in ostalo podobno hrano z veliko ogljikovimi hidrati sem omejil do 200g, meso jajca in sir do 400g, zelenjavno in sadje od 100g do 200g, sladkarije do 20g ali 50g, pijače do 200g ali 500g ter sol, maslo in olje do 10g. Po zagonu algoritma za prvi primer dobimo:

Jabolko	= 100.0 g
Pomfri	= 100.0 g
Mleko	= 50.0 g
Sir_edamec	= 62.3125620035 g
Maslo	= 2.72551586742 g
Zelje	= 100.0 g
Grozdje	= 100.0 g
Jagode	= 100.0 g
Torta	= 48.6668662158 g
Nutella	= 50.0 g
Banana	= 100.0 g
Kokice	= 7.64385225795 g
Paradižnik	= 200.0 g
Paprika	= 100.0 g
Korenje	= 100.0 g
Pomaranca	= 100.0 g
Marmelada	= 50.0 g
Med	= 50.0 g
Kakav	= 80.7011503747 g
Radenska	= 497.950053281 g
skupna masa živil	= 2000.0 g
skupna energija živil	= 1738.14867165 kcal

Za primer z zahtevami po vitaminu C in kaliju ter natriju pa dobimo:

Jabolko	= 100.0 g
Pomfri	= 100.0 g
Mleko	= 50.0 g
Sir_edamec	= 69.6990342893 g
Grozdje	= 100.0 g
Torta	= 90.2518414249 g
Nutella	= 30.2270174353 g
Banana	= 100.0 g
Kokice	= 35.7142924612 g
Korenje	= 100.0 g
Kumara	= 192.4610995 g
Pomaranca	= 29.3068511579 g
Marmelada	= 50.0 g
Med	= 50.0 g
Kakav	= 75.8234728001 g
Sol	= 2.06573642046 g
Radenska	= 500.0 g
skupna masa živil	= 1675.54934549 g
skupna energija živil	= 1766.90381632 kcal

2.2.2. Minimizacija vnosa maščob

Rešitve za minimizacijo maščob pri vseh zgornjih omejitvah hrani in zahtevi po 2000 kcal energije so:

Mleko	= 157.17250912 g
Sir_edamec	= 14.4001785109 g
Kruh_bel	= 210.981604852 g
Solata	= 246.651746855 g
Korenje	= 508.011980548 g
Marmelada	= 154.305681687 g
Kakav	= 48.9344145357 g
Sol	= 1.50337207273 g
Cabernet	= 658.038511818 g
skupna masa živil	= 2000.0 g
skupna energija živil	= 2000.0 kcal
skupne maščobe živil	= 2520.76336275 g

Ko dodamo omejitve za razumne količine posamezne hrane dobimo:

Jabolko	= 100.0 g
Mleko	= 50.0 g
Sir_edamec	= 62.6948463907 g
Kruh_bel	= 59.1722472662 g
Rdeca_pesa	= 100.0 g
Solata	= 100.0 g
Pivo	= 341.373156378 g
Grozdje	= 100.0 g
Torta	= 103.342121613 g
Krompir	= 72.2419061191 g
Banana	= 100.0 g
Korenje	= 100.0 g
Kumara	= 200.0 g
Marmelada	= 50.0 g
Med	= 50.0 g
Kakav	= 56.3222625889 g
Sol	= 2.23644479844 g
Radenska	= 71.7301142018 g
Cabernet	= 200.0 g
Traminec	= 80.8869006444 g
skupna masa živil	= 2000.0 g
skupna energija živil	= 2000.0 kcal
skupne maščobe živil	= 4679.11684359 g

Čeprav meni postane precej boj pester se očitno ne moremo znebiti vnosa alkohola, zato sem dodal dodaten kriterij, ki omeji količino alkoholnih pijač na 300g in dobil meni, ki za omejitev vnosa maščob preferira vino pred pivom:

Mleko	= 50.0 g
Sir_edamec	= 51.4297256458 g
Kruh_bel	= 38.8235110077 g
Cokolada	= 20.0 g
Fizol	= 89.8447674668 g
Rdeca_pesa	= 100.0 g
Solata	= 100.0 g
Grozdje	= 100.0 g
Torta	= 123.668984794 g
Krompir	= 96.4529033732 g
Banana	= 100.0 g
Korenje	= 100.0 g
Kumara	= 200.0 g
Marmelada	= 47.3152294472 g
Med	= 50.0 g
Kakav	= 34.7836941923 g
Sol	= 2.29273365172 g
Radenska	= 395.388450422 g
Cabernet	= 200.0 g
Traminec	= 100.0 g
skupna masa živil	= 2000.0 g
skupna energija živil	= 2000.0 kcal
skupne maščobe živil	= 5015.96578493 g

Literatura

- [1] Spletne strani z gradivom predmeta, ogled 20.10.2017,
<http://predmeti.fmf.uni-lj.si/modelska>
- [2] Spletne strani z gradivom predmeta, ogled 20.10.2017,
<http://predmeti.fmf.uni-lj.si/modelska/podatki>

3. Numerična minimizacija

Povzetek

Tokrat nadgradimo znanje iz prejšnjih dvah vaj in se spoznamo z numerično minimizacijo nelinearnih problemov. Pri prvi nalogi rešujemo Thomsonov problem, kjer je potrebno poiskati razporeditev nabojev na sferi, in sicer tako, da nam ta poda najnižjo elektrostatično energijo. V drugi nalogi se vrnemo k problemu varčne vožnje skozi semafor iz prve vaje in ga poskusimo rešiti z minimizacijo akcije, pri čemer zahtevamo izpolnitev zahtev, ki jih podajo izbrane vezi in zahteve po omejitvah hitrosti.

3.1. Uvod

Pri nelinearni minimizaciji iščemo minimum funkcije $f(\vec{r})$ več neodvisnih spremenljivk, ki jih združimo v vektor \vec{r} . Po najpreprostejši zamisli zaporedoma minimiziramo funkcijo po vsaki spremenljivki, vendar ta postopek praviloma zelo počasi konvergira. Rajši uporabimo bolj zahtevne, vendar hitrejše postopke. Zavedati se moramo, da so koordinate minimalne točke po katerikoli metodi določene samo z natančnostjo $\sqrt{\epsilon}$, če je ϵ relativna natančnost podajanja vrednosti funkcije f . [1]

3.2. Naloga

- Thomsonov problem: na prevodno kroglo nanesemo enakih (klasičnih) nabojev. Kako se razmestijo po površini? Zahtevamo seveda minimum elektrostatične energije. Uporabi katero od minimizacijskih metod, npr. Powellovo ali n-dimenzionalni simpleks (amebo)
- Problem optimalne vožnje skozi semafor, ki smo ga spoznali pri nalogi 1, lahko rešujemo tudi z numerično minimizacijo, če časovno skalo diskretiziramo.

Lagrangianu

$$\int (dv/dt)^2 dt - \lambda \int v dt \quad (3.1)$$

lahko dodamo omejitev hitrosti v obliki členov $\exp(\beta(u - u_{lim}))$, če hočemo (približno) zagotoviti $u \leq u_{lim}$. Izpolnitev pogoja je toliko ostrejša, kolikor večji β vzamemo. Poskusil lahko tudi druge omejitvene funkcije, na primer kakšno funkcijo s polom. Za iskanje Lagrangeovega množiliatorja lahko uporabiš bisekcijo ali kakšno drugo vgrajeno metodo za iskanje ničel na funkciji $l(\lambda) = \int v(\lambda, t) dt$, kjer je $v(\lambda, t)$ rezultat minimizacije pri izbranem λ .

Komentar k reševanju

Kratka referenca metod iz *Numerical Recipes*:

- odprogram `amoeba` implementira Nelder-Mead algoritmom. Je preprost za uporabo, zahteva samo podprogram `func` za izračun funkcije f , je zelo robusten, vendar tudi počasen in približen. Načeloma ga lahko uporabimo tudi za negladke funkcije. Temelji na krčenju hipertetraedra (simpleksa), s katerim objamemo minimum, na področje z velikostjo tolerance.
- Podprogram `powell` predpostavlja zveznost minimizacijske funkcije in terja celo vrsto spremmljevalnih procedur (po vrsti najprej minimizacijsko funkcijo `func`, nato `linmin` za 1D minimizacijo, ta pa `f1dim` in `brent`). Metoda določi konjugirane smeri v prostoru, nato pa vzdolž njih ujame minimum. Zato najde rezultat zelo natančno in hitro.
- Še nekoliko hitrejše metode, na primer metodo konjugiranih gradientov Fletcher-Reeves-Polak-Ribiere (`frprmn`) ali metodo s spremenljivo metriko Davidon-Fletcher-Powell (`dfpmin`), uporabimo, kadar lahko podamo tudi odvode funkcije, ki jo minimiziramo. Delujejo podobno Newtonovi metodi za iskanje ničel funkcije.

V Pythonu podoben nabor minimizacijskih metod na voljo v modulu `scipy.optimize` s funkcijo `minimize(method='Nelder-Mead'|'Powell'|'L-BFGS-B'|...)`. V nalogi so uporabljene navedene tri metode modula `scipy.optimize`, ki so natančneje predstavljene v [3]. [1]

3.3. Thomsonov problem

Pri reševanju problema nanašamo diskretne točkaste naboje z $q_i = q_j = 1$ na enotsko kroglo z radijem $r_0 = 1$. Položaje na krogli lahko opišemo z dvema koordinatama sferičnega sistema, $\theta \in [0, \pi]$ in $\phi \in [0, 2\pi]$. Elektrostatična energija, ki jo poskušamo minimizirati se računa po enačbi

$$V_{ij}(r) = C \cdot \frac{q_i q_j}{r_{ij}} \quad (3.2)$$

kej brez izgube splošnosti nastavimo še $C = \frac{1}{4\pi\epsilon_0} = 1$. Potencialna energija med vsakim takim parom enotskih nabojev je tako odvisna le od medsebojne razdalje med njima. V sferičnih koordinatah razdaljo med pari nabojev računamo kot

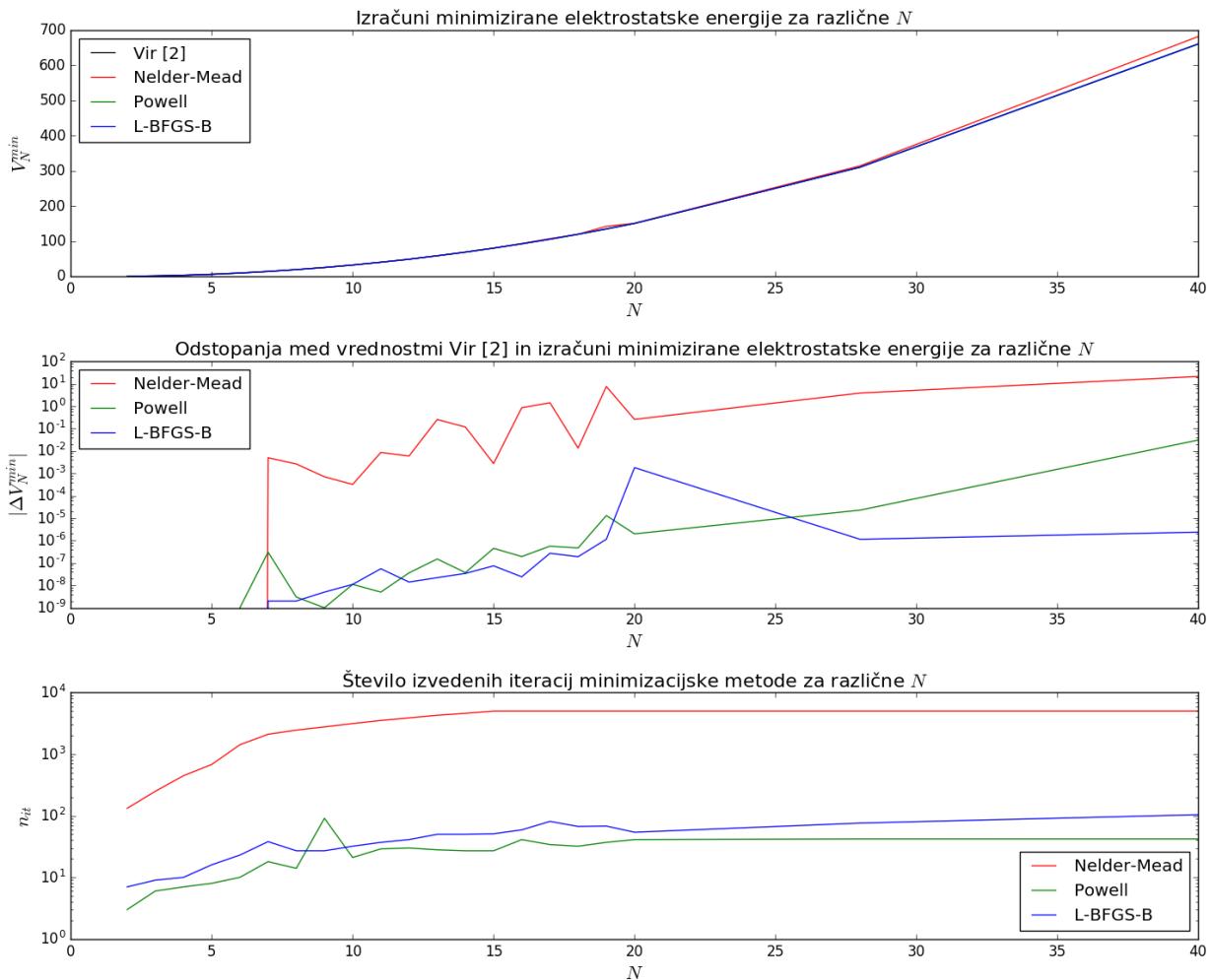
$$r_{ij} = \sqrt{r_i^2 + r_j^2 - 2r_i r_j (\cos \theta_i \cos \theta_j + \sin \theta_i \sin \theta_j (\cos \phi_i \cos \phi_j + \sin \phi_i \sin \phi_j))}. \quad (3.3)$$

Skupna potencialna energija sistema je tako vsota elektrostatičnih energij vseh parov N nabojev na krogli

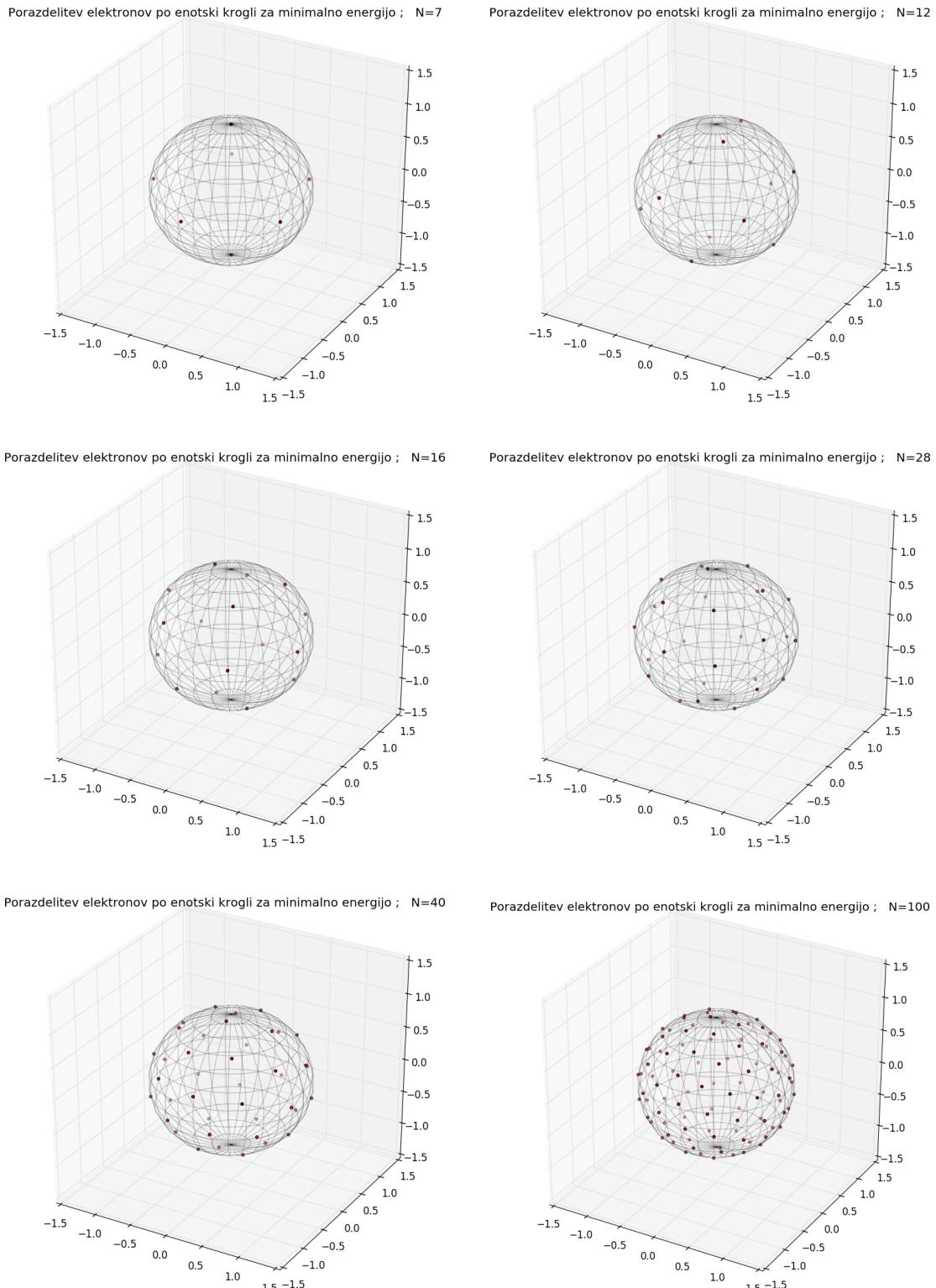
$$V_N = \sum_{i < j} V_{ij} = \sum_{i < j} \frac{1}{r_{ij}}. \quad (3.4)$$

Minimizacijo sem izvedel s tremi različnimi v komentarju omenjenimi metodami in pri več različnih številih naključno nanešenih nabojev N . Nekaj izračunanih položajev porazdelitev nabojev za izbrane N je prikazanih na sliki 3.2, izračuni tako dobljenih energij pa so predstavljeni na sliki 3.1, kjer so prikazana tudi odstopanja izračunov od vrednosti povzetih iz [2]. Meja tolerance za računanje je bila nastavljena na vrednost 10^{-9} . Ker je pri vseh metodah število iteracij omejeno na 5000, z naraščanjem N niso vse metode dosegle željene natančnosti minimizacije. Dodatno naj omenim, da je metoda 'Powell' računanje prekinila, ko je doseglala določeno število izračunov vrednosti V_N ; to je omejilo število maksimalnih iteracij na vrednost 42. Število izvedenih iteracij za izračun V_N^{min} s posamezno metodo v odvisnosti od števila nanešenih nabojev N , je prikazano na spodnjem grafu slike 3.1.

Za majhne N so vse tri izbrane metode podale dober rezultat. Najbolje se je obnesla metoda 'Powell', ki je za izračun potrebovala tudi najmanj iteracij. Sledi ji metoda 'L – BFGS – B' in nato 'Nelder – Mead', ki je za izračun potrebovala največ časa. Z večanjem števila nabojev pa je metoda 'Powell' ni mogladohajati ostalih dveh. Maksimalno število iteracij je doseglala že pri primeru $N = 7$. Metoda 'Powell' je začela odpovedovati pri $N = 20$, metoda 'L – BFGS – B' pa pri $N = 470$.



Slika 3.1.: Vrednosti minimizirane elektrostatske energije za različne N po viru [2] za primerjavo z izračuni s tremi izbranimi metodami (zgoraj). Odstopanja izračunov od vrednosti iz [2] za različne N (sredina). Število vseh potrebnih iteracij za izračun zgoraj navedenih vrednosti energij po vsaki od metod (spodaj). Izračuni so izvedeni z izbrano mejo natančnosti 10^{-9} in omejitvijo iteracij na največ 5000. Teh zahtev z naraščanjem številq N niso dosegle vse metode. Metoda 'Nelder – Mead' je odpovedala pri $N = 7$, čeprav še ni dosegla omejitve iteracij. Metoda metoda 'Powell' je bila še kos problemu $N = 19$, za nadaljnje primere pa je računanje prekinila mnogo prej, in sicer, ko je dosegla vgrajeno maksimalno število izračunov vrednosti V_N . Kot najboljša se je izkazala metoda 'L – BFGS – B'.



Slika 3.2.: Položaji nabojev, ki pri različih številih na sfero nanešenih nabojev N zagotavljajo minimalno vrednost njihove medsebojne elektrostatičen energije. Prikazani izbrani primeri optimalnih položajev nabojev so računani po metodi 'L – BFGS – B'.

3.4. Problem optimalne vožnje skozi semafor

Problem vožnje iz prve vaje je zahteval analitično minimizacijo akcije izbranega lagrangiana ob pogoju prepotovane razdalje do semaforja. Tokrat se minimizacije lotimo z numeričnim minimizacijskim algoritmom, zato integrale v akciji (3.1) diskretiziramo in tako računamo ustreerne vsote. Tako definiran izraz mora torej pri optimalnem poteku hitrosti imeti čim manjšo vrednost, ker pa akcijo v taki obliki lahko vedno zminimiziramo na interval hitrosti z vrednostmi nič, je potrebno nekako upoštevati vez, ki bi to preprečila. V našem primeru je vez prepotovana pot do semaforja L . Da bi minimizator kaznoval za izbiro prenizkih hitrosti tako definiram člen

$$\mathcal{L}_L = e^{\beta \left| \int_0^T v(t) dt - L \right|}, \quad (3.5)$$

ki ga prištejem k (3.1). Vsakršno odstopanje prepotovane poti od zahteve vezi (1.3), je na tak način pri minimizaciji kaznovano, saj veča vrednost minimizacijske funkcije. Z višanjem vrednosti parametra β lahko določamo strogost kriterija vezi.

Podobno ravnamo v primeru, ko želimo na intervalu omejiti vrednosti hitrosti vožnje. Za ta namen definiramo člen, ki kaznuje izbiro prevelikih hitrosti

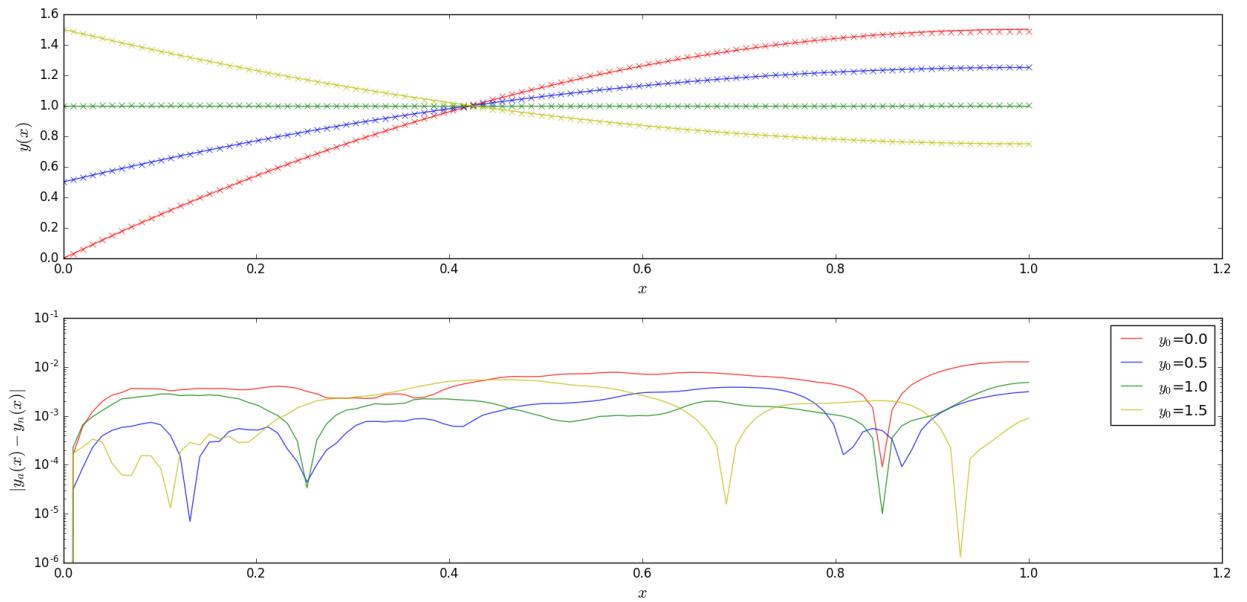
$$\mathcal{L}_{v_{max}} = e^{\lambda(v - v_{max})}$$

in analogno člen, ki kaznuje premajhne hitrosti

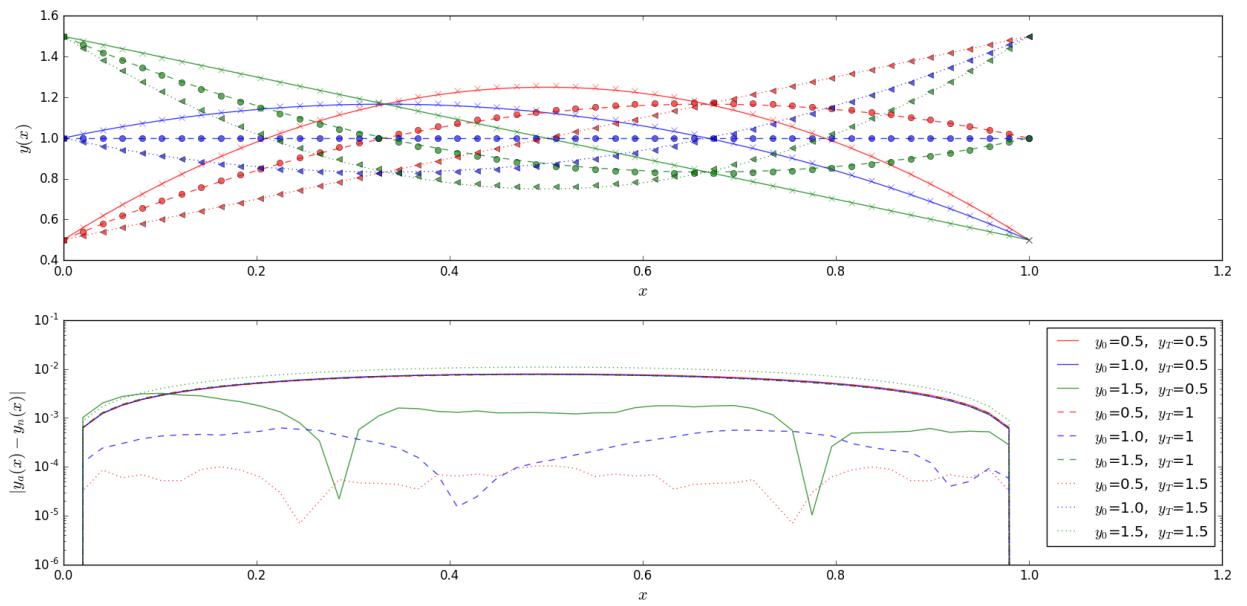
$$\mathcal{L}_{v_{min}} = e^{\lambda(v_{min} - v)}.$$

Takšna člena ravno tako prištejemo k (3.1). Če želimo čim bolj zadostiti pogoju $v_{min} < v < v_{max}$ ju obtežimo s toliko večjim parametrom λ in s tem minimizacijski algoritem prisilimo v izogibanje izbire neustreznih hitrosti, ki ponovno minimizacijski funkciji tako večajo vrednost. V nadaljevanju so prikazani rezultati minimizacije tako definiranih funkcij.

Nalogo sem tokrat reševal le z uporabo metode 'L – BFGS – B', ki se je v primeru Thompsonovega problema odrezala najbolje. Da bi se prepričal o ustreznosti zgornjih predpostavk sem vrednosti poračunane s tako spisanim algoritmom v brezdimenzijski obliki, najprej primerjal z analitičnimi rešitvami iz prejšnje vaje; to sem storil za primere prostih, omejenih in periodičnih končnih robnih pogojev pri različnih vrednostih začetne hitrosti y_0 . Rezultati so prikazani na slikah 3.3-3.5, kjer je videti, da algoritem deluje dokaj spodobno, saj v diskretnih točkah minimizirane vrednosti ne odstopajo od analitičnih za več kot dve decimalni mest. Po preizkušu algoritma sem izvedel še izračune s funkcijo, ki upošteva različno obtežene omejitve hitrosti vožnje. Ti so za tri kategorije robnih pogojev pri različnih začetnih (končnih) hitrostih in vrednostih λ prikazani na sliki 3.6. Metoda je vrnila krivulje, ki so se ob večanju vrednosti parametra λ vedno bolj stiskale v interval $v_{min} < v < v_{max}$ in pri tem vseeno skušale ohraniti zahtevo vezi (1.3) po prepotovani razdalji L .

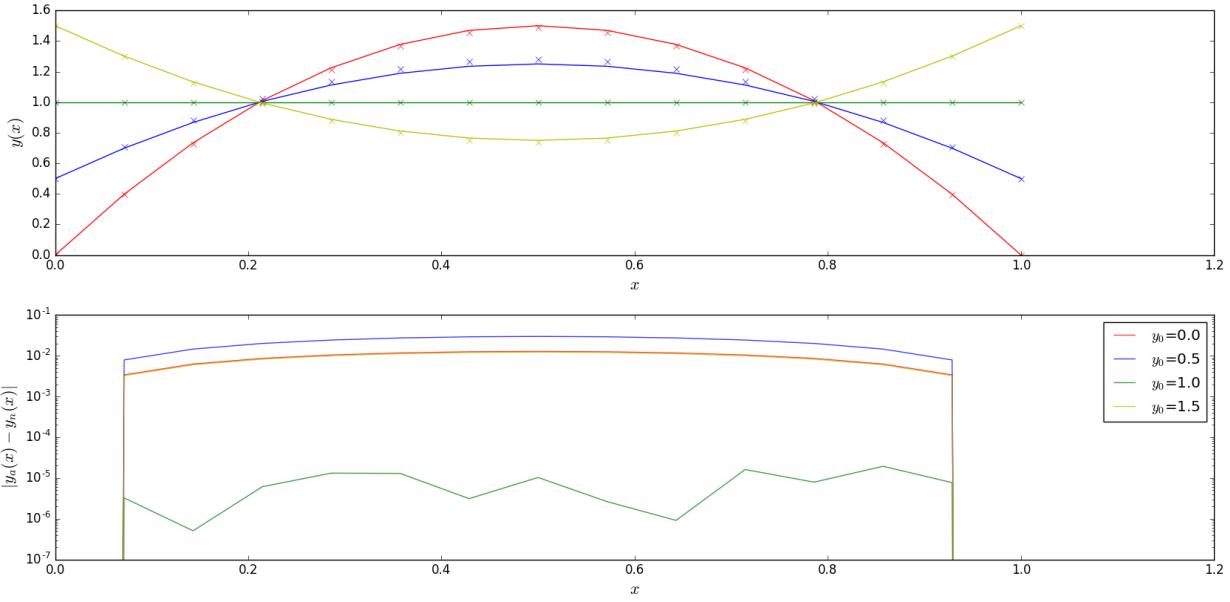


Slika 3.3.: Zgornji graf prikazuje odvisnosti hitrosti od časa za primer brez končnega robnega pogoja in za različne vrednosti začetne hitrosti y_0 . Polna krivulja predstavlja analitični rezultat, 'x' pa diskretne izračune minimizacijskega algoritma v 100 točkah. Odstopanja slednjih od analitičnih vrednosti so za vsako od teh točk poračunana in prikazana na spodnjem grafu, kjer vidimo da rezultati algoritma ne odstopajo od analitičnih za več kot dve decimalni mesti.

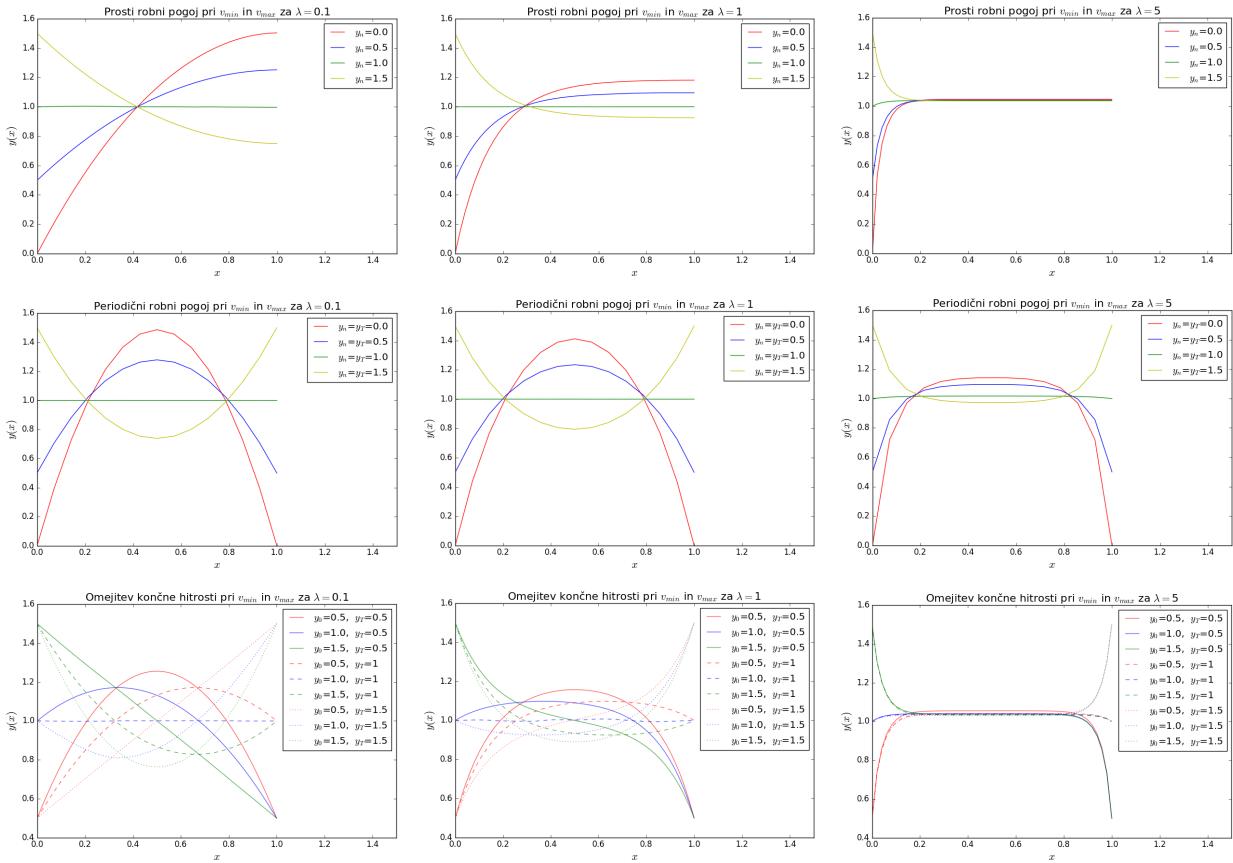


Slika 3.4.: Zgornji graf prikazuje odvisnosti hitrosti od časa za primer z različnimi fiksnnimi končnimi robnimi pogoji y_T in za različne vrednosti začetne hitrosti y_0 . Krivulje predstavljajo analitične rezultate, 'x', 'o' in 'Δ' pa diskretne izračune minimizacijskega algoritma v 50 točkah. Odstopanja slednjih od analitičnih vrednosti so za vsako od teh točk poračunana in prikazana na spodnjem grafu, kjer vidimo da rezultati algoritma ne odstopajo od analitičnih za več kot dve decimalni mesti.

3. Numerična minimizacija



Slika 3.5.: Zgornji graf prikazuje odvisnosti hitrosti od časa za primer z istimi fiksнимi končnimi robnimi pogoji $y_0 = y_T$ in za različne vrednosti začetne hitrosti y_0 . Polna krivulja predstavlja analitični rezultat, 'x' pa diskretne izračune minimizacijskega algoritma v 15 točkah. Odstopanja slednjih od analitičnih vrednosti so za vsako od teh točk poračunana in prikazana na spodnjem grafu, kjer vidimo, da rezultati algoritma za primer izračuna bolj skope diskretizacije od analitičnih odstopajo tudi za več kot dve decimalni mestni.



Slika 3.6.: Grafi prikazujejo diskretne izračune minimizacijskega algoritma v 50 točkah in sicer za tri kategorije robnih pogojev pri različnih začetnih (končnih) hitrostih in vrednostih λ . Krivulje se ob večjanju vrednosti parametra λ vedno bolj stiskajo v interval $v_{min} < v < v_{max}$.

Literatura

- [1] Spletne strani z gradivom predmeta, ogled 20.10.2017,
<http://predmeti.fmf.uni-lj.si/modelska>
- [2] Wikipedia "Thomson problem", ogled 27.7.2019,
https://en.wikipedia.org/wiki/Thomson_problem – 12
- [3] spletna stran z opisi knjižnice "SciPy" programskega jezika Python, ogled 27.7.2019,
<https://docs.scipy.org/doc/scipy/reference/generated/scipy.optimize.minimize.html>

4. Populacijski modeli

Povzetek

Pri vaji preko zveznega pristopa s pomočjo sistemov navadnih diferencialnih enačb modeliramo nekaj diskretnih pojavov, ki jih srečamo v naravi. Pri tem preučimo časovno dinamiko različnih tipov populacij in si ogledamo njihove fazne diagrame. Zanima nas vedenje populacij v odvisnosti od začetnih pogojev ter parametrov, ki jih opisujejo.

4.1. Uvod

Rešujemo sklopljen sistem linearnih enačb prvega reda $\dot{y}_i(x) = f_i(x, y_1, \dots, y_N)$.[1]

4.2. Naloga

- Preuči standardni standardni deterministični **model zajci-lisice** v obliki

$$\begin{aligned}\dot{Z} &= \alpha Z - \beta Z L, \\ \dot{L} &= -\gamma L + \delta Z L.\end{aligned}\tag{4.1}$$

- Nariši in preiči fazni diagram (brezdimenzijska oblika, zastojne točke, stabilnost . . .).
- Oglej si obhodne dobe v odvisnosti od začetnega stanja. Zadošča, da preišeš stanja, v katerih ima ena komponenta ravnovesno vrednost.

- Analiziraj fazni portret za **populacijski model laserja** s konstantnim črpanjem.

$$\begin{aligned}\dot{F} &= -Bf + DAF, \\ \dot{A} &= -CA - EAF + Q.\end{aligned}\tag{4.2}$$

Določi ravnovesno stanje v odvisnosti od moči črpanja. Kako se s tem parametrom spreminja frekvenca in karakteristični čas relaksacijskih oscilacij?

- Model epidemije:** populacijo razdelimo v tri razrede: (D) zdravi in dovzetni, (B) bolni in klicenosni, (I) imuni: nedovzetni in neklicenosni. Bolezen se širi s stiki med zdravimi in bolnimi. Bolnik preide s konstantno verjetnostjo med imune (ozdravi ali umre).

$$\begin{aligned}\dot{D} &= -\alpha DB, \\ \dot{B} &= +\alpha DB - \beta B), \\ \dot{I} &= \beta B.\end{aligned}\tag{4.3}$$

V epidemiji nas zanima njen vrh (maksimalno trenutno število obolelih), čas nastopa maksimuma in celotno število obolelih. S cepljenjem lahko vnaprej preselimo določen del populacije med imune. Kako vpliva delež cepljenih na parametre epidemije? Kako se spremeni potek epidemije, če obolele razdeliš na več razredov?

Komentar k reševanju

Sistem enačb lahko rešujemo s programom *rk4* iz zbirke *Numerical Recipes*, ki uporablja metodo Runge-Kutta 4. reda. V zbirki *GSL* integratorje najdemo v modulu *gsl_odeiv.h*, v Pythonu pa v modulu *scipy.integrate*, kjer ustvarite objekt razreda *ode* in mu tip integratorja nastavite z metodo *set_integrator*.[1]

Naloga je rešena v programskem okolju Python, ob uporabi omenjenega modula z integratorjem *dopri5* ali *dop853*. Vsak izmed zgoraj predstavljenih problemov je računan pri različnih parametrih in začetnih pogojih pri čemer so spremenljivke obravnavane kot zvezne. Rešitve so prikazane na grafih; tako v časovni kot v medsebojni odvisnosti.

4.3. Zajci in lisice

Zanima nas dinamika sistema (4.1) (spremembra števila zajcev in števila lisic glede na medsebojne interakcije), v katerem Z in L predstavlja diskretno število (populacijo) zajcev in lisic. Parameter α predstavlja naravni prirastek zajcev, β umrljivost zajcev na podlagi srečanja z lisicami, γ odstranje lisic in δ prirastek lisic zaradi hranjenja z zajci. Ta model je kar se da preprost in ne upošteva mnogih drugih naravnih dejavnikov, zato ne predstavlja realne situacije.

Ker tak sistem rešujemo v zveznem približku, lahko začetno število lisic določimo iz deleža števila zajcev na način $L = aZ$. To nam omogoči, da z različnimi vrednostmi parametra a preprosteje izbiramo začetne pogoje za računanje populacije lisic in zajcev. Podobno si lahko zmanjšamo še število konstant, saj v enačbi nastopata velikosti obeh populacij v odvisnosti od časa, za kateri nas ne zanima sama vrednost, ampak le njihova relativna dinamika. Problem ima netrivialno zastojno točko pri $Z = \gamma/\delta$ in $L = \alpha/\beta$. Če si želimo velikosti populacij normirati tako, da bo imela zastojna točka koordinate $(1, 1)$, mora torej veljati $\alpha = \beta$ in $\gamma = \delta$.

Dodatno lahko z deljenjem časovnih odvodov dobimo integral sistema

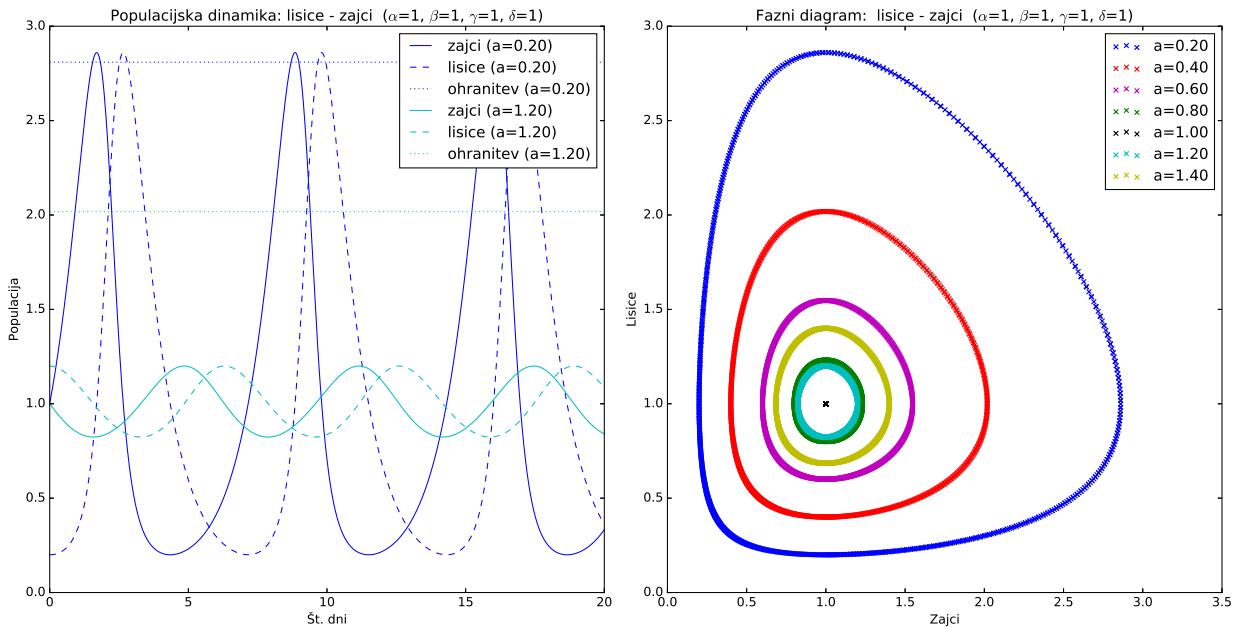
$$\frac{dZ}{dL} = \frac{Z(\alpha - \beta L)}{L(-\gamma + \delta Z)} \quad (4.4)$$

z rešitvami

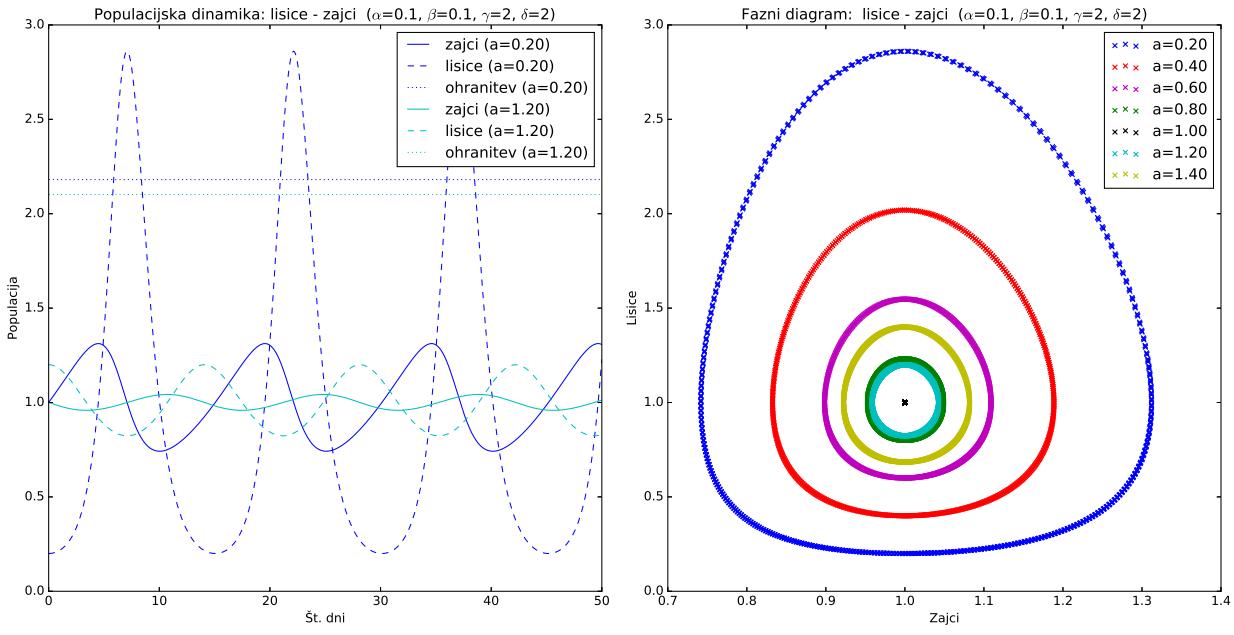
$$\delta Z - \gamma \log(Z) + \beta L - \alpha \log(L) = C, \quad (4.5)$$

ki predstavljajo ohranitveno količino sistema oz. družino krivulja določeno s konstanto C .

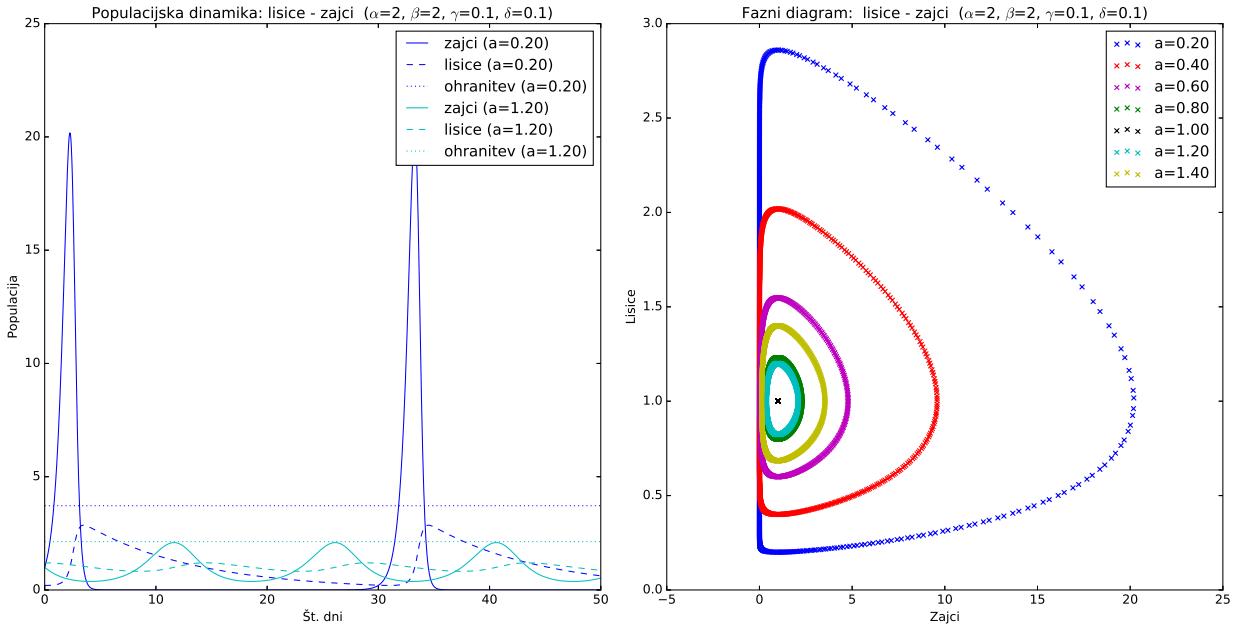
Rezultati računanj za različne vrednosti a začetnih razmerij populacij lisic in zajcev pri različnih parametrih α, β, γ in δ so predstavljeni na slikah 4.1-4.3. Grafi na levi prikazujejo časovno dinamiko velikosti populacije, na desni pa so fazni diagrami populacije zajcev in lisic.



Slika 4.1.: Graf časovne dinamike populacije zajcev in lisic za dve različni začetni vrednosti $a = L/Z$ (levo). Opaziti je periodičnost različnih dožin in amplitud. Črta '...' prikazuje vrednost ohranitvene konstante C . Fazni diagram (desno) je simetričen. Tukaj so začetni parametri sistema enaki in velja $\alpha/\beta = \gamma/\delta = 1$. Zastojna točka ima koordinato $(1, 1)$. dosežemo jo z izbiro $a = 1$.

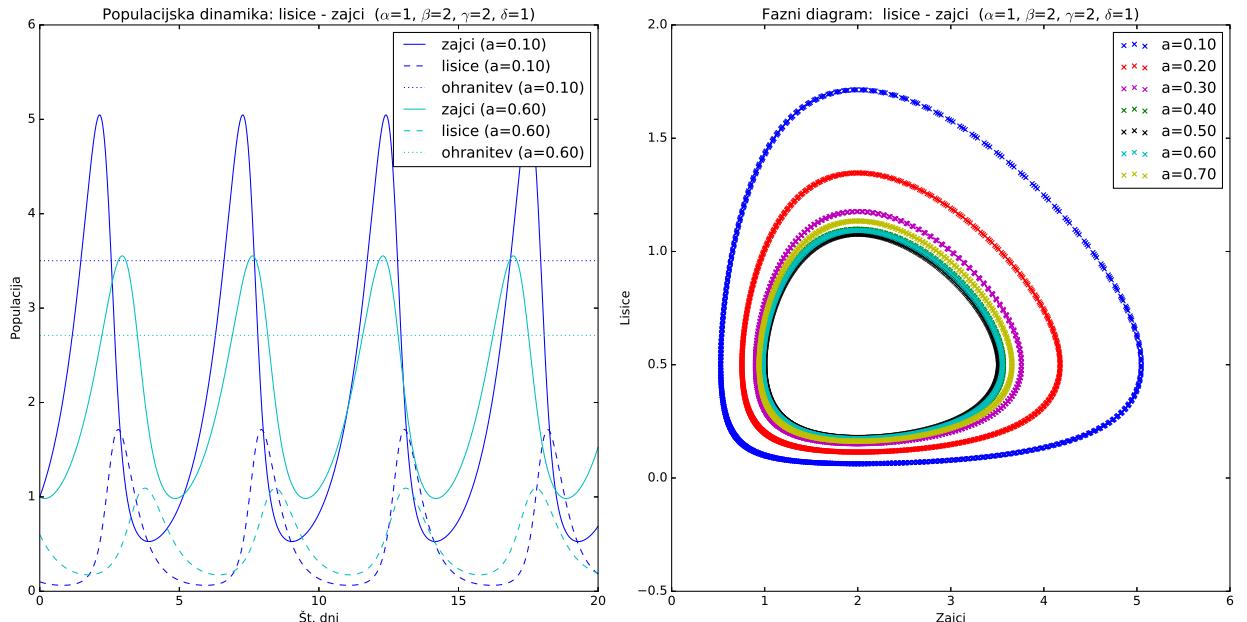


(a)

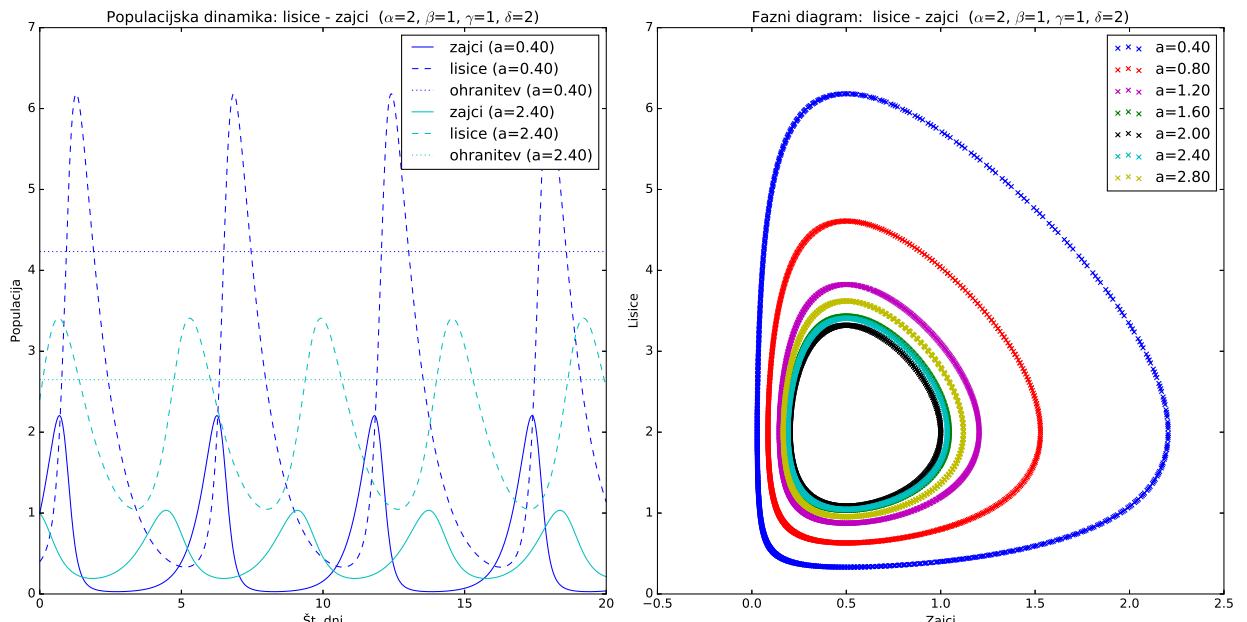


(b)

Slika 4.2.: V primerih (a) in (b) so si začetni parametri sistema med seboj paroma različni. Na grafih časovne dinamike populacije zajcev in lisic za dve različni začetni vrednosti $a = L/Z$ (levo) je opaziti periodičnost različnih dožin in amplitud. Tu črta '...' prikazuje vrednost ohranitvene konstante C . Simetrije faznih diagramov (desno) so za izbrane začetne parametre različne, še vedno pa velja $\alpha/\beta = \gamma/\delta = 1$. Zastojna točka ima tako spet vrednost $(1, 1)$, ki jo dosežemo z izbiro $a = 1$.



(a)

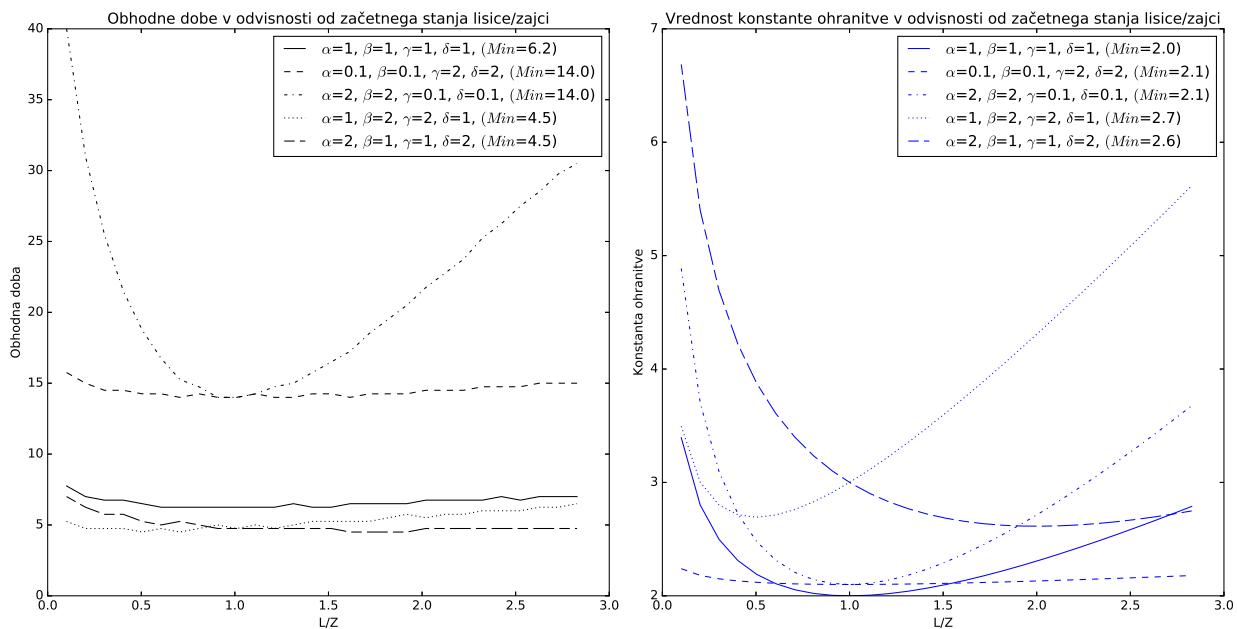


(b)

Slika 4.3.: Na grafih časovne dinamike populacije zajcev in lisic za dve različni začetni vrednosti $a = L/Z$ (levo) je opaziti periodičnost različnih dozin in amplitud. Črta '...' prikazuje vrednost ohranitvene konstante C . Tokrat velja $\alpha/\beta = 0.5$ in $\gamma/\delta = 2$ (a) ter $\alpha/\beta = 2$ in $\gamma/\delta = 0.5$ (b). Vidimo, da se zastojna točka glede na navedene začetne parametre ustrezno premakne, vendar je z nobeno izbiro začetne vrednosti a ne moremo doseči.

Iz prikazanega je lahko razbrati periodično dinamiko populacije s konstantnimi amplitudami, katere ohranitvena količina je res ves čas konstanta. Kot je videti, so oblike krivulj bogate. Velikosti njihovih amplitud in dolžine njihovih periodov so očitno določene z začetnimi vrednostmi a in izbiro ostalih parametrov. Smiselno je torej izrisati odvisnosti dolžin obhodnih dob za različne kombinacije. Prikazane so na grafu (a) slike 4.4. Izkaže se, da za vsako kombinacijo obstaja minimum, v katerem so te najkrajše, njegova pozicija pa je odvisna od vrednosti razmerja a in ostalih parametrov.

Oblike faznih krivulj izražajo izbiro parametrov in so ravno tako raznolike. Položaji zastojnih točk v faznih diagramih sovpadajo z pričakovanimi. Zanimivo je opažanje, da se s spremenjanjem začetne vrednosti populacije zastojnim točkam sprva približujemo in nato spet oddaljimo. Pri posameznih izbirah njihovih vrednostih zastojno točko lahko dosežemo, pri drugih spet ne. Ker je fazni diagram določen iz vrednosti amplitud obeh populacij, na grafu (b) slike 4.4 preverimo odvisnost vrednosti ohranitvene konstante od a za različne kombinacije. Izgleda, da tudi v tem primeru za vsako kombinacijo obstaja minimum, kjer so te najkrajše. Njegova pozicija je ponovno odvisna od vrednosti razmerja a in ostalih parametrov.



Slika 4.4.: Grafa prikazujeta dolžino obhodnih dob (levo) in vrednost ohranitvene konstante (desno) v odvisnosti od parametra $a = L/Z$ za različne kombinacije dinamičnih parametrov. Izgleda, da v vseh primerih obstajajo minimumi, saj pri vseh krivuljah vidimo, da bolj kot razmerje $a = Z/L$ sovpada z vrednostjo α/β , krajši so obhodni časi ter manjša je ohranitvena konstanta C.

4.4. Laser

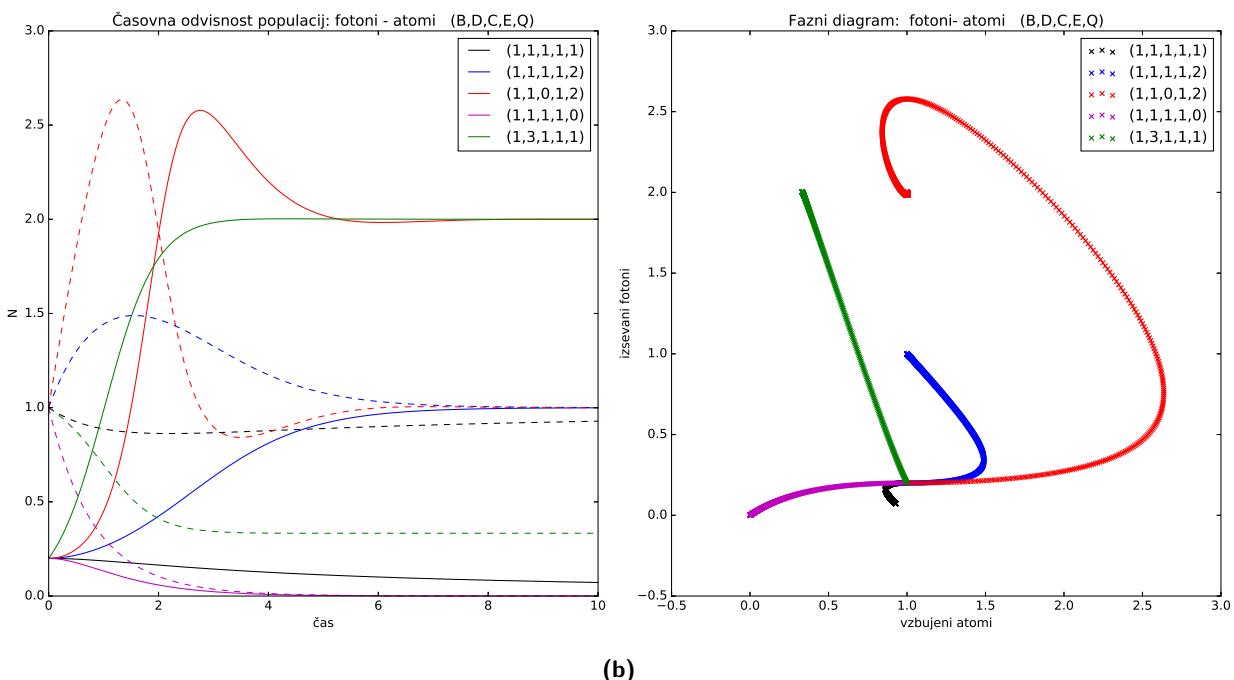
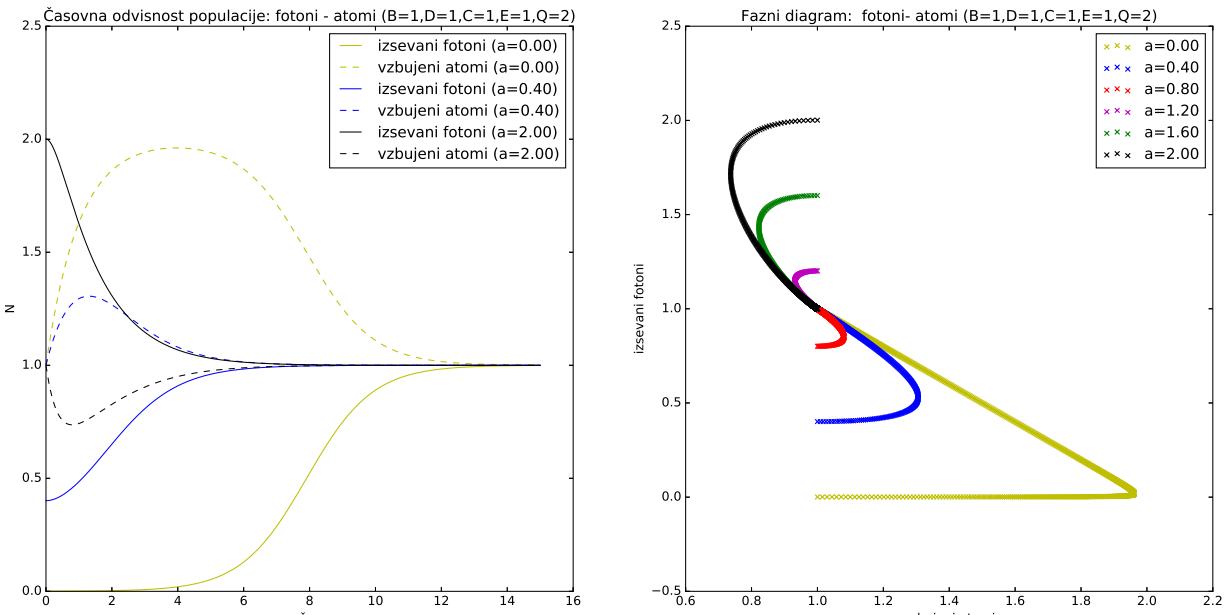
Dinamiko podaja sistem (4.2), kjer A in F predstavlja diskretno število (populacijo) vzbujenih atomov in fotonov, D in E predstavlja Einsteinov koeficient stimulirane emisije vzbujenih atomov, C je koeficient za spontano emisijo, Q koeficient črpanja (vzbujanja atomov), B pa predstavlja prepustnost zrcal.

Tudi tu lahko določimo zastojne točke. Normiramo lahko čas in dve spremenljivki ter se tako znebimo treh parametrov. Zanimiva je zastojna točka, ki predstavlja stabilno delovanje laserja. Fiksiramo jo, če vzamemo $B = E$ (normiranje A), $C = E$ (normiranje F) in $D = E = 1$ (normiranje časa) ter tako dobimo njen položaj v $(1, Q - 1)$. Na grafih (a) slike 4.5 sta za različne začetne pogoje $a = F/A$ in izbiro parametrov $B = D = C = E = 1$ ter $Q = 2$ prikazana časovna dinamika delovanja laserja in fazni diagram. Populaciji atomov in fotonov se za vse zabrane začetne pogoje ustalita pri isti vrednosti in opaziti je konvergenco k ustrezni zastojni točki $(1, 1)$.

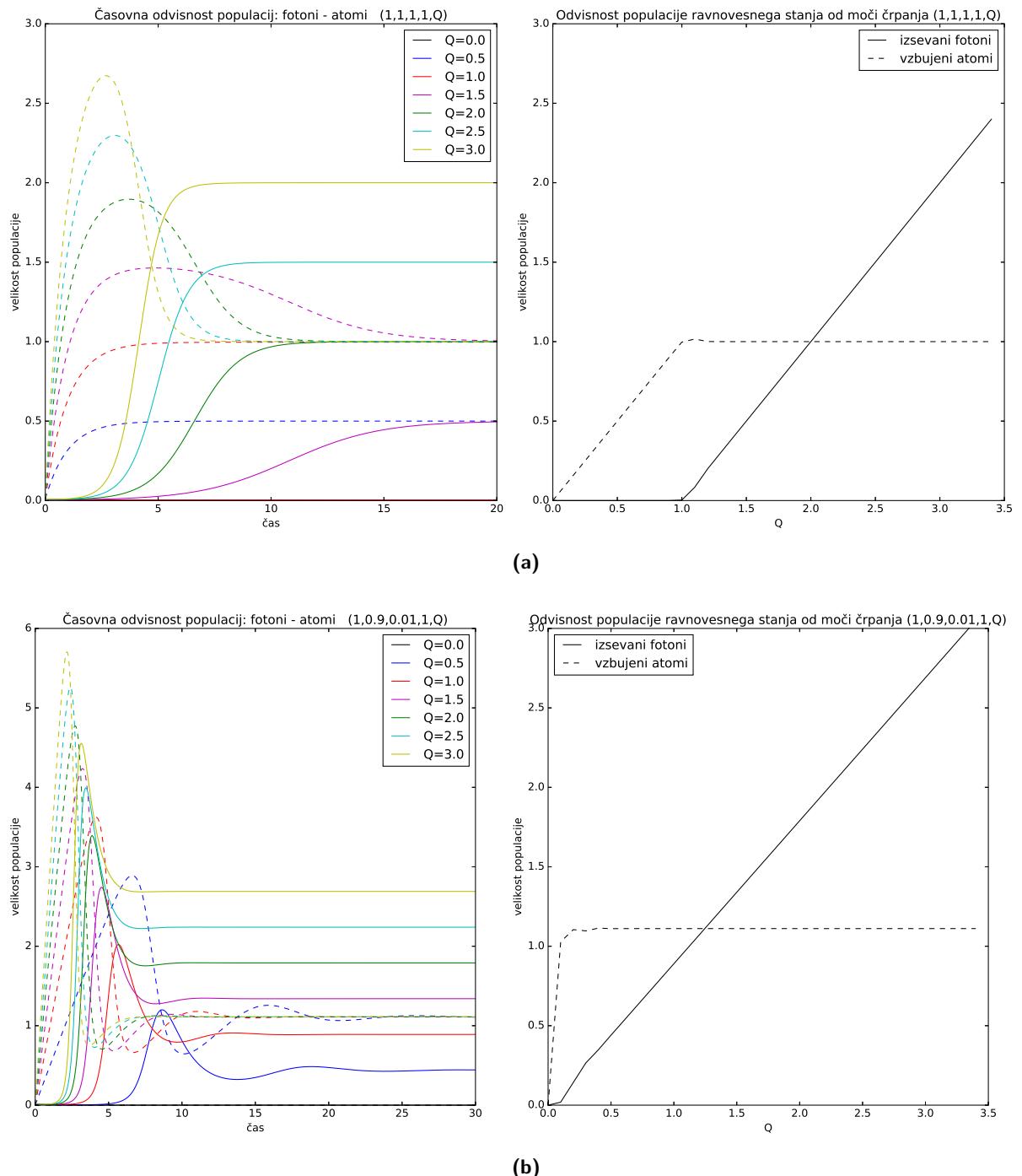
Podobno so na grafih (b) prikazani izračuni za začetno vrednost $a = 0.2$ in različne vrednosti ostalih parametrov. Opaziti je da v tem primeru vse fazne krivulje izhajajo iz točke začetne vrednosti $(1, 0.2)$ in končajo vsaka v svoji zastojni točki, ki ustreza izbranim parametrom. Videti je še, da je v nekaterih primerih izbira parametrov ugodna za vžig laserja, v drugih (ko velja $Q \leq 1$) pa populacija fotonov pada na 0.

Ravovesno stanje fotonov je torej direktno odvisno od moči črpanja Q . Ker nas zanima ali bo prižiganje in delovanje laserja za različne parametre uspešno, si na grafih slike 4.6 ogledamo rezultate izračunov za predvsem takšne začetne pogoje, kjer na začetku ni vzbujenih atomov, ampak jih vzbudimo le s črpanjem; ker pa v enačbi nimamo nobenega črpanja fotonov je pri slednjih potrebno zagotoviti neko začetno vrednost npr $F_0 > 0$.

Dodana sta tudi grafa velikosti populacije fotonov po dosegu ravovesja v odvisnosti od moči črpanja. Takoj po vključitvi laserja na grafih opazimo kratko nihanje, ki pa se kmalu ustali v stabilnih legah. To nihanje se pojavi predvsem pri majhnih vrednostih koeficiente za spontano emisijo C in majhnih močeh črpanja. Z naraščajočim parametrom črpanja ta frekvenca narašča sam relaksacijski čas oscilacij k ravovesju pa se krajsa. Nadaljno povečevanje moči črpanja na sam potek delovanja laserja nima posebnega vpliva. Zaradi močnejšega črpanja je v ravovesni legi le več fotonov.



Slika 4.5.: Grafa (a) prikazuje dinamiko za različne vrednosti začetnih pogojev $a = F/A$. Opaziti je manjše relaksacijske oscilacije pri postavljanju sistema v ravnovesno stanje 1. Ostali parametri so torej izbrani tako, da na faznem diagramu vidimo relaksacijo k zastojni točki (1, 1). Na grafih (b) je začetni pogoj fiksni $a = 0.2$, na ostalih parametrih pa preizkušamo različne kombinacije, ki iz iste začetne točke vodijo do ustrezne zastojne točke. Če vrednost F ne pada na nič laser doseže ravnovesno stanje delovanja. Če črpanja ni ali je premajhno ($Q \leq 1$) laser seveda ugasne.

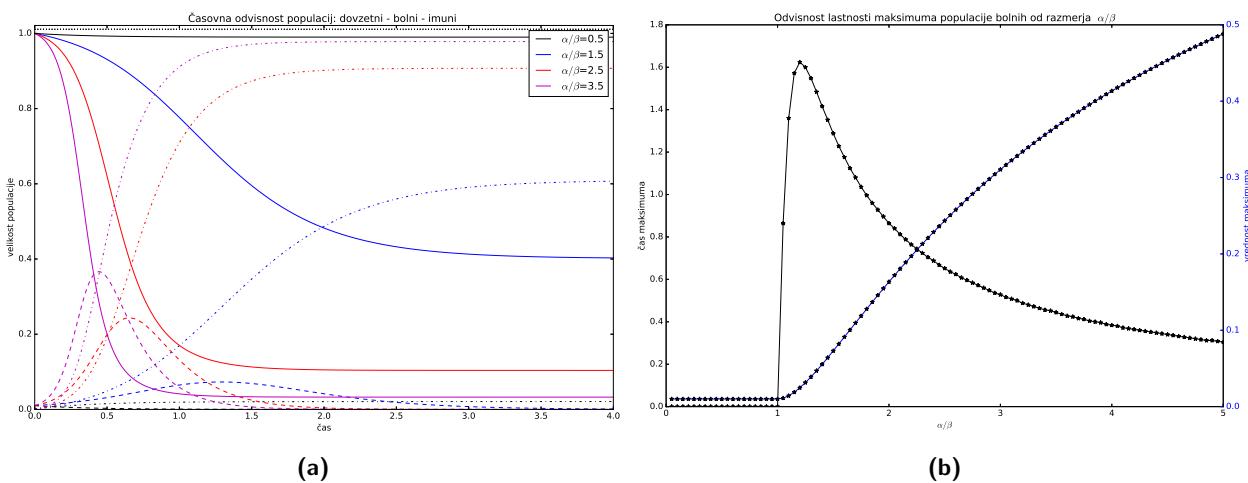


Slika 4.6.: Grafa časovnega razvoja števila atomov in fotonov (levo) in odvisnost vrednosti platoja ravnovesnega stanja od moči črpanja (desno). Začetni vrednosti sta $F_0 = 0.01$ in $A_0 = 0$. Za kombinacijo parametrov (a) je časovnja dinamika lepa. Večja kot je moč črpanja, prej populacija fotonov doseže plato. Za kombinacijo (b) velja isto, le da so zaradi majhne vrednosti koeficiente za spontano emisijo C pri prižigu prisotne oscilacije večje. Njih frekvenca z večanjem Q narašča, vendar zaradi krajše relaksacije tudi prej izzveni.

4.5. Epidemija

Dinamiko epidemije lahko v grobem popišemo s sistemom (4.3), ki upošteva število srečanj med bolnimi in zdravimi osebami ob parametru dovzetnosti obolevanja α ter parametru ozdravljenja oziroma prehajanja med imune β . Tudi tu je privzetih precej predpostavk, ki jih v farmacevtski praksi ponavadi ni za sprejeti, saj v resnici bolnik preide skozi različne stadije bolezni, te pa imajo običajno neko povprečno trajanje, žal pa marsikdo bolezni tudi uspešno ne preboli.

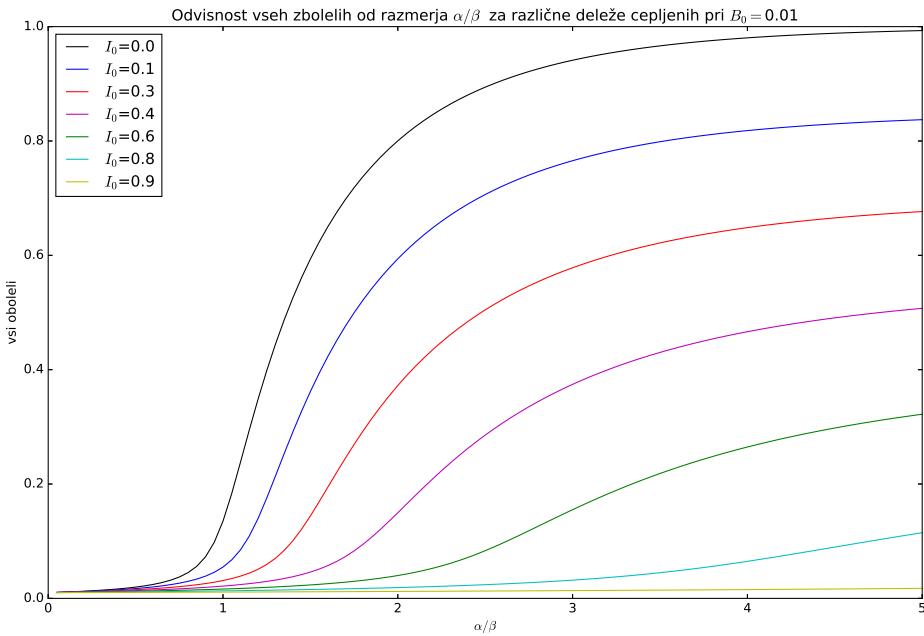
Najprej si poglejmo nekaj primerov poteka epidemije, ki jih prikazuje graf (a) na sliki 4.7. Časovi razvoj je računan za različna razmerja parametrov α/β vendar ob istem začetnem deležu zdravih, bolnih in imunih. Opaziti je, da se populacijska dinamika ustavi pri različnih deležih zdravih, bolnih in imunih za vsako od vrednosti parametrov α/β . Pri velikem parametru dovzetnosti se stvari hitro razvijejo in število prebolelih je visoko. Pri hitrejšem prebolevanju pa epidemija ne pride do pravega izraza, saj je končno število vseh zbolelih majhno. Število ljudi, ki tekom epidemije zbolijo v nekem trenku doseže maksimum in nato spet pade. Položaj in velikost tega maksimuma sta očitno odvisna od razmerja α/β , kar preverimo na grafu (b) iste slike. Od tu sklepamo, da se za zelo dovzetno populacijo vrh pojavi hitro in zajame veliko množico a tudi hitro preide v imunost, tako je možnost za prenos okužbe časovno bolj omejena. Dodatno pa na grafu za nizko vrednost razmerja α/β opazimo odsotnost pojava maksimuma, kar pomeni, da se epidemija ni zgodila.



Slika 4.7.: Graf (a) prikazuje časovni potek epidemije. Začetni deleži zdravih '—', bolnih '—' in imunih '—.' so konstantni, spremojamo pa razmerje parametra dovzetnosti in ozdravitve α/β . Število zdravih se zmanjša na račun obolelih, ki nato preidejo v populacijo imunih. Večja kot je vrednost parametra, hitreje in intenzivneje epidemija steče, medtem ko za majhne vrednosti epidemija ne pride do izraza. Čas viška epidemije in njena intenzivnost sta v odvisnosti od α/β prikazana na grafu (b), od koder se lahko prepričamo o opaženem. Tu je jasno vidna odsotnost pojava maksimuma za nedovzetno oziroma hitro ozdravljivo populacijo. Še več, opazimo tudi odsotnost pojava maksimuma, kar pomeni, da za pri majhnih α/β epidemije ni bilo.

V nadaljevanju preučimo še primere, ko spremojamo začetno populacijo imunih. To lahko razumemo kot nadzorovan cepljenje, saj tako lahko vnaprej preselimo določen del populacije med imune. Graf na sliki 4.8 prikazuje odvisnost vseh zbolelih od razmerja α/β . Krivulje so izrisane za različne začetne vrednosti oz. deleže precepljene populacije. Razbrati je mogoče, da so za velike vrednosti α/β potrebna bolj množična cepljena t.j. če želimo omejiti število vseh ljudi, ki zbolijo, oziroma ohraniti čim več več prebivalstva zdravega, medtem ko je za relativno manjše vrednosti α/β cepljenje neefektivno.

Za konec si ogledamo še dinamiko epidemije bolj raznolike populacije, v kateri obolele razdelimo v več razredov t.j. vsakega z različnim faktorjem ozdravitve β_i . V nadaljevanju je obravnavan primer

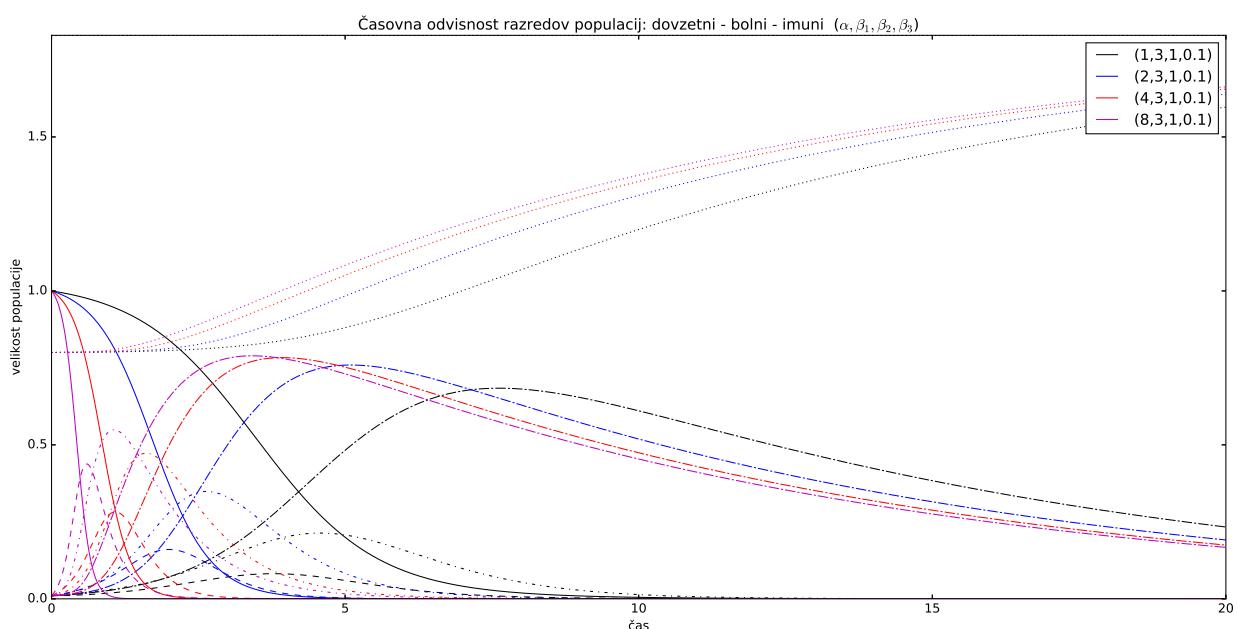


Slika 4.8.: Graf prikazuje število zbolelih v odvisnosti od vrednosti razmerja α/β za različno pocepljene populacije. Delež začetnih prenašalcev bolezni B_0 znaša 1% populacije. Iz prikazanega je mogoče sklepati, da cepljenje ni vedno efektivno temveč zavisi od tipa bolezni.

s tremi razredi

$$\begin{aligned} \dot{D} &= -\alpha D(B_1 + B_2 + B_3), \\ \dot{B}_1 &= +\alpha D(B_1 + B_2 + B_3) - \beta_1 B_1, \\ \dot{B}_2 &= \beta_1 B_1 - \beta_2 B_2, \\ \dot{B}_3 &= \beta_2 B_2 - \beta_3 B_3, \\ \dot{I} &= \beta_3 B_3. \end{aligned} \tag{4.6}$$

Izračuni so prikazani na sliki 4.9. Vidimo, da takšna obravnava bolezni prispeva k bolj enakomernemu trajanju bolezni, saj vsota vseh treh obolelih skupin hitro nastopi in je razpotegnjena daleč v čas. Tu je precepljenost še toliko bolj pomembna.



Slika 4.9.: Graf poteka epidemije za tri razrede bolnega prebivalstva. Delež začetnih prenašalcev bolezni znaša $B_{0i} = 1\%$ in delež cepljene populacije $I_0 = 45\%$. Opaziti je daljše trajanje epidemije. Vsota populacije bolnih je razpotegnjena v čas, saj dovzetni zbolijo ter prenašajo okužbo dalje ljudem, ki bolezen počasi prebolevajo.

Literatura

- [1] Spletne strani z gradivom predmeta, ogled 20.10.2017,
<http://predmeti.fmf.uni-lj.si/modelska>

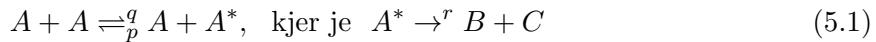
5. Modeli kemijskih reakcij

Povzetek

Pri vaji rešujemo sisteme navadnih diferencialnih enačb, ki opisujejo dinamiko koncentracij reaktantov sodeljujočih v izbranih kemijskih reakcijah. Če je mogoče sisteme rešujemo tako direktno kot v stacionarnem približku; s slednjim dosežemo zmanjšanje števila prostih parametrov ki opisujejo kinematike. Ogledamo si tri različne večstopenjske reakcije, pri čemer spremojamo razmerja začetnih vrednosti koncentracij in razmerja samih hitrosti reakcij.

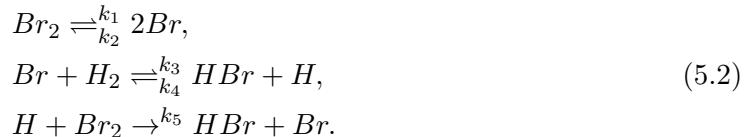
5.1. Naloga

1. Za **model binarne reakcije**



in velja $q/p = 1000$, integriraj sistem eksaktne in v aproksimaciji stacionarnega stanja za $r/qA(0) = 10, 1, 0.1!$ [1]

2. **Model reakcije** $H_2 + Br_2 \rightleftharpoons 2HBr$ vključuje naslednje stopnje:

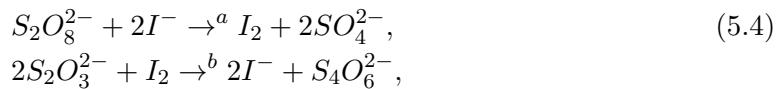


Določi izraz za hitrost reakcije (npr. $\dot{[HBr]}$) v aproksimaciji stacionarnega stanja in primerjaj z empiričnim izrazom

$$\dot{[HBr]} = \frac{k[H_2][Br_2]^{1/2}}{m + [HBr]/[Br_2]}. \quad (5.3)$$

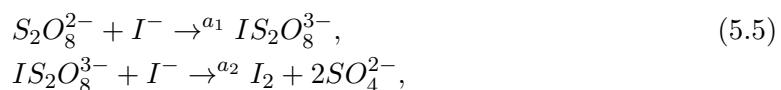
Kako bi najlažje izmeril empirični konstanti k in m ? Skiciraj časovne poteke za $[H_2]/[Br_2] = 100, 1, 0.01$ v začetnem stanju pri $m = 2.5$. Ali kaj pomaga, če že v začetno stanje primešamo mnogo HBr ?[1]

3. **Kemijske ure** so reakcije, ki stečejo s predvidljivim in ponavadi ostrim časovnim zamikom. Primer take reakcije je jodova ura, ki v eni izmed izvedb temelji na ravnotežju naslednjih reakcij:

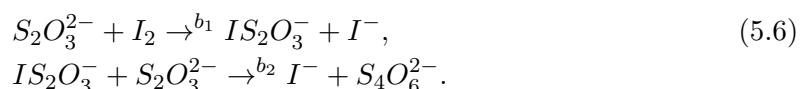


Druga reakcija je bistveno hitrejša od prve, mehanizem merjenja časa pa je enakomerno porabljanje tiosulfata $S_2O_3^{2-}$. Če je persulfat $S_2O_8^{2-}$ v prebitku, lahko za aktivne spremenljivke vzamemo le $[I^-]$, $[I_2]$ in $[S_2O_3^{2-}]$.

Obe zgornji reakciji sta v resnici sosledji dveh binarnih reakcij preko kratkoživega prehodnega stanja. Prva reakcija je sestavljena iz počasne in hitre stopnje



in druga prav tako iz počasne in hitre stopnje



V približku stacionarnega stanja izpelji kinetični zvezi za obe glavni reakciji. Razloži potek celotne reakcije in odvisnost trajanja reakcije od začetne koncentracije tiosulfata. Za razmerje hitrosti glavnih reakcij vzemi $\lambda = 1, 10, 100$. [1]

Komentar k reševanju

Nalogo sem reševal v okolju Python kjer sem v modulu `scipy.integrate` ustvaril objekt razreda `ode` in nastavil tip integratorja `dop853`. V nekaterih delih naloge sem si pri izpeljovah pomagal s programskim paketom Mathematica.

5.2. Model binarne reakcije

Binarni reakciji (5.1) je potrebno najprej poiskati ustrezen sistem linearnih enačb prvega reda. Za določitev hitrosti reakcije potrebujemo le koncentraciji $[A]$ in $[A^*]$, zato bo to sistem dveh enačb z dvema neznankama

$$\begin{aligned} \dot{[A]} &= -p[A]^2 + q[A][A^*], \\ \dot{[A^*]} &= p[A]^2 - q[A][A^*] - r[A^*], \\ \dot{[B]} &= \dot{[C]} = r[A^*]. \end{aligned} \quad (5.7)$$

Sistem je v tej eksaktni obliki precej tog, predvsem zaradi intenzivnejšega prehajanja populacij iz A v A^* , določenega s parametrom $q/p = 1000$; t.j. iz $p \ll q$ sledi $[A^*] \gg [A]$. Na koncentracijo $[A^*]$ pa je pripeta druga časovna konstanta r , ki uravnava tvorbo produktov $[B]$ in $[C]$. Čeprav nas zanima le rezultat v daljšem časovnem obdobju, bo sistem enačb zaradi relativno majhnih koncentracij zahteval integriranje po majhnih korakih.

Koncentracija vzbujenih molekul $[A^*]$ se sicer hitro spreminja, vendar je v primerjavi z $[A]$ ves čas majhna. Sistem lahko zato rešujemo v približku stacionarnega stanja, kjer te hitre in majhne spremembe zanemarimo in dodamo pogoj $\dot{[A^*]} = 0$, ki da za vrednost $\dot{[A^*]}$ pogoj

$$p[A]^2 - q[A][A^*] - r[A^*] = 0 \quad \Rightarrow \quad [A^*] = \frac{p[A]^2}{q[A] + r}. \quad (5.8)$$

Tako nam za prvi del reakcije ostane le ena diferencialna enačba z eno neznanko

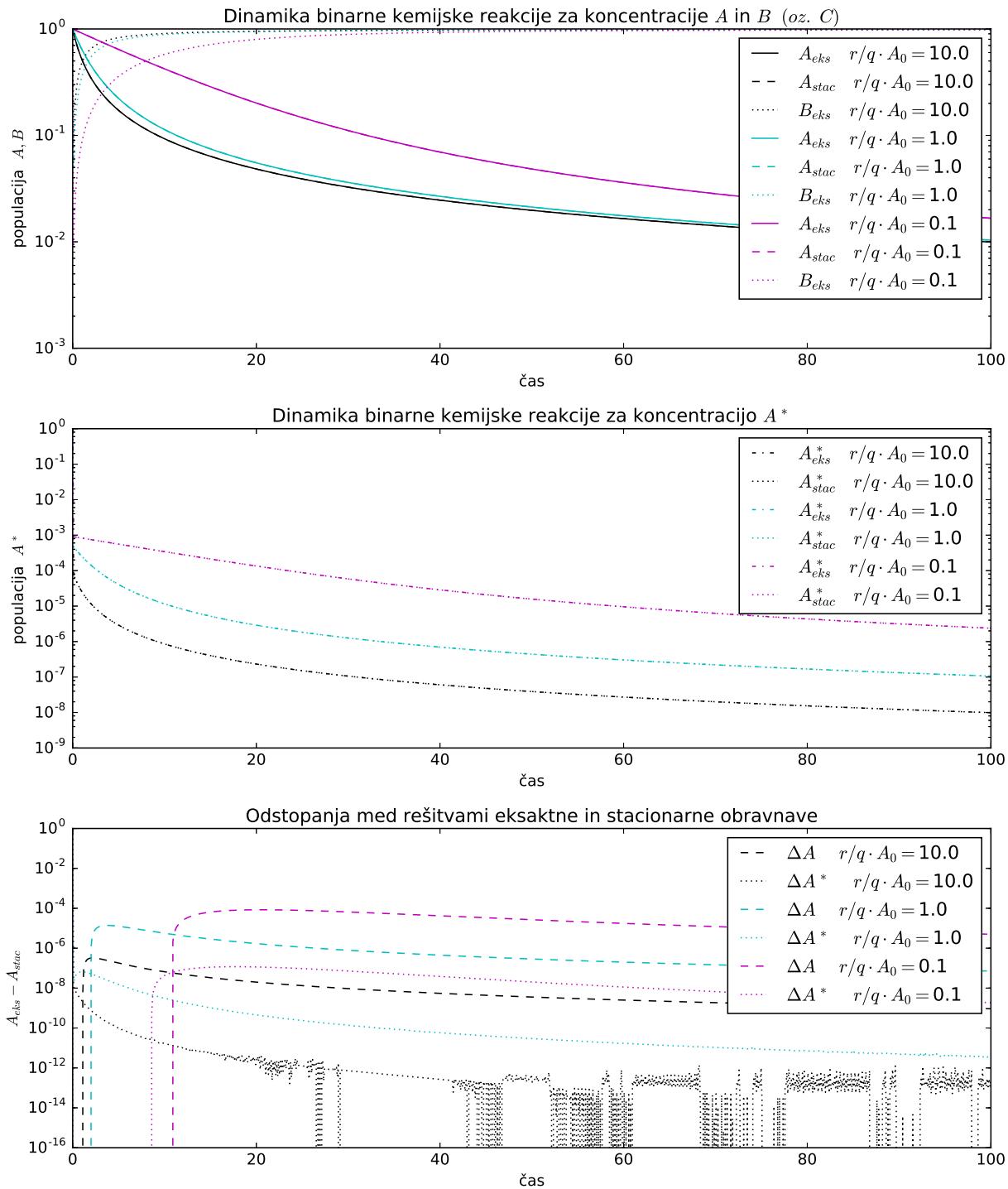
$$\dot{[A]} = -p[A]^2 \left(1 - \frac{q[A]}{q[A] + r} \right) = -\frac{p[A]^2 r}{q[A] + r}. \quad (5.9)$$

Izračuni potekov reakcije preko eksaktnega reševanja in računanja s stacionarnim približkom so za začetna pogoja $[A]_0 = 1$ in $[A^*]_0 = 0$ prikazani na zgornjih grafih slike 5.1. Tam na oku ne vidimo nobene razlike med rezultati obeh metod, saj je izbrani integrator dovolj dober. Za smiselno primerjavo obeh pristopov je na spodnjem grafu slike izrisana razlika med izračuni. Vidno je odstopanje na nekaj decimalnih mest, ki se z večanjem vrednosti $r/qA(0)$ še manjša.

5.3. Reakcija z več stopnjami

Pri tem delu naloge reakcijo (5.2) najprej prevedemo v sistem

$$\begin{aligned} \dot{[\text{Br}_2]} &= -k_1[\text{Br}_2] + k_2[\text{Br}]^2 - k_5[\text{H}][\text{Br}_2] \\ \dot{[\text{Br}]} &= k_1[\text{Br}_2] - k_2[\text{Br}]^2 - k_3[\text{Br}][\text{H}_2] + k_4[\text{HBr}][\text{H}] + k_5[\text{H}][\text{Br}_2] \\ \dot{[\text{H}_2]} &= -k_3[\text{Br}][\text{H}_2] + k_4[\text{HBr}][\text{H}] \\ \dot{[\text{H}]} &= k_3[\text{Br}][\text{H}_2] - k_4[\text{HBr}][\text{H}] - k_5[\text{H}][\text{Br}_2] \\ \dot{[\text{HBr}]} &= k_3[\text{Br}][\text{H}_2] - k_4[\text{HBr}][\text{H}] + k_5[\text{H}][\text{Br}_2] \end{aligned} \quad (5.10)$$



Slika 5.1.: Grafa zgoraj in na sredini prikazujeta časovni potek reakcije izračunan eksaktno in v stacionarnem približku. Na spodnjem grafu je v logaritemski skali prikazana razlika med rešitvami obeh metod. Opaziti je, da je napaka za vrednosti A^* manjša in večanjem vrednosti $r/qA(0)$ pri obeh vrednostih pada.

Da se znebimo množice parametrov takoj uporabimo približek stacionarnega stanja

$$[\dot{\text{Br}}] = [\dot{\text{H}}] = 0, \quad (5.11)$$

ki nam poda zvezi

$$([\text{Br}] + [\text{H}]) = k_1[\text{Br}_2] - k_2[\text{Br}]^2 = 0 \Rightarrow [\text{Br}] = \sqrt{\frac{k_1}{k_2}[\text{Br}_2]} \quad (5.12)$$

$$[\dot{\text{H}}] = k_3[\text{Br}][\text{H}_2] - k_4[\text{HBr}][\text{H}] - k_5[\text{H}][\text{Br}_2] = 0 \Rightarrow [\text{H}] = \frac{k_3[\text{H}_2]\sqrt{\frac{k_1}{k_2}[\text{Br}_2]}}{k_4[\text{HBr}] + k_5[\text{Br}_2]}. \quad (5.13)$$

Naprej iz kemijske reakcije sklepamo

$$[\dot{\text{H}}_2] = [\dot{\text{Br}}_2] = -2[\dot{\text{HBr}}] \quad (5.14)$$

in dobimo

$$\begin{aligned} [\dot{\text{HBr}}] &= k_3[\text{Br}][\text{H}_2] - k_4[\text{HBr}][\text{H}] + k_5[\text{H}][\text{Br}_2] \\ &= \frac{2k_3k_5}{k_4}\sqrt{\frac{k_1}{k_2}}\frac{[\text{H}_2]\sqrt{[\text{Br}_2]}}{k_5/k_4 + [\text{HBr}]/[\text{Br}_2]}. \end{aligned} \quad (5.15)$$

S primerjavo tega izraza z empiričnim izrazom (5.3) sklepamo, da mora veljati

$$k = \frac{2k_3k_5}{k_4}\sqrt{\frac{k_1}{k_2}} \quad \text{in} \quad m = \frac{k_5}{k_4}. \quad (5.16)$$

Rešujemo sistem (5.14) in (5.15) iz treh diferencialnih enačb prvega reda. Na grafih slike 5.2 je prikazana odvisnost koncentracij reaktantov od časa pri različnih začetnih pogojih.

Zanimivo je predvsem obnašanje sistema za reakcijo $\text{H}_2 + \text{Br}_2 \rightleftharpoons 2\text{HBr}$ pri različnih vrednostih parametra m . Opazujemo lahko iz časovnih odvodov 5.14 izpeljani relaciji

$$\frac{[\dot{\text{HBr}}]}{[\dot{\text{H}}_2]} = \frac{d\text{HBr}}{d\text{H}_2} = -2 \quad \text{in} \quad \frac{[\dot{\text{Br}}_2]}{[\dot{\text{H}}_2]} = \frac{d\text{Br}_2}{d\text{H}_2} = 1, \quad (5.17)$$

ki ju integriramo z začetnim pogojem $[\text{HBr}]_0 = 0$. Rezultat integracije nam poda zvezi

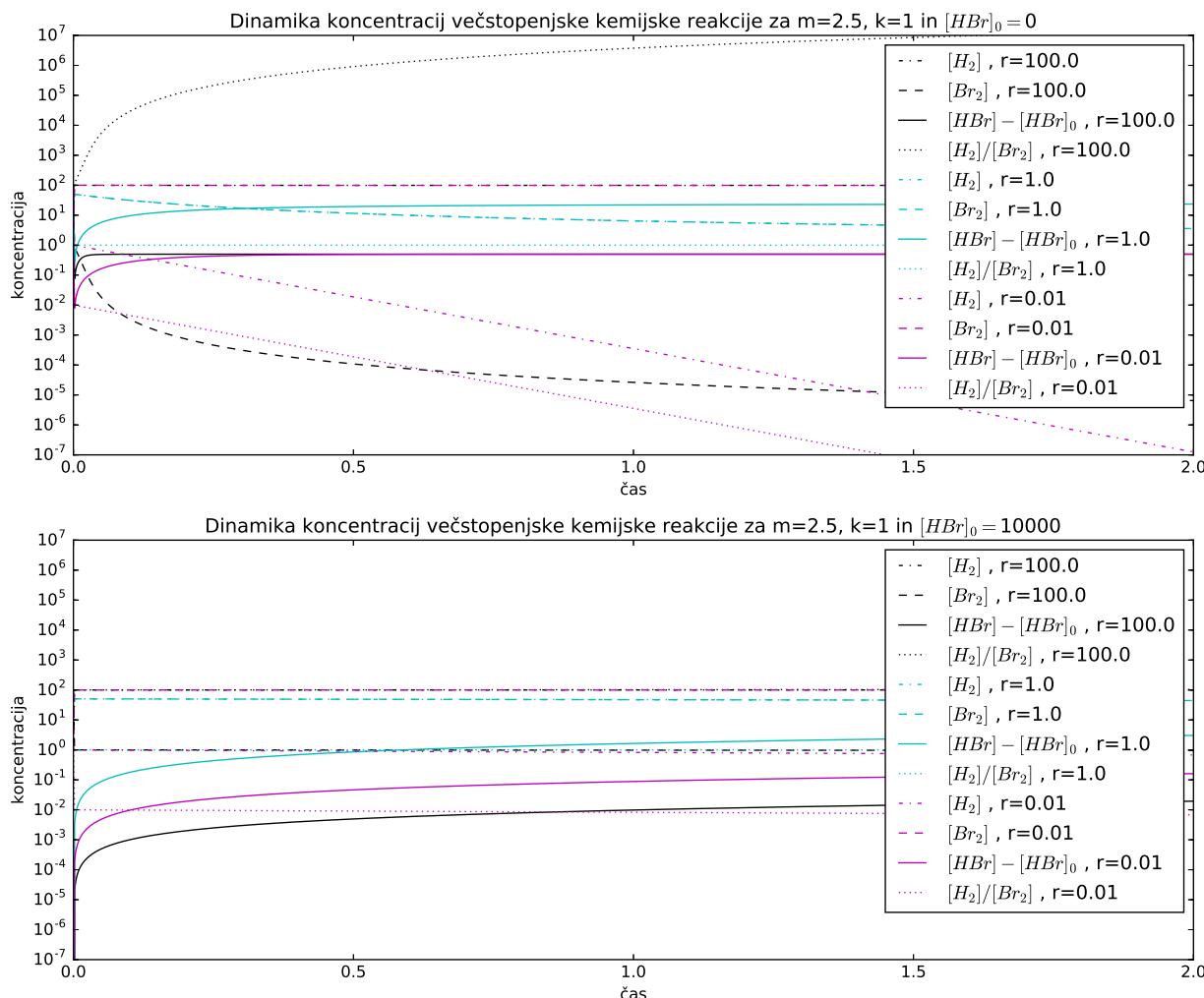
$$[\text{HBr}] = -2([\text{H}_2] - [\text{H}_2]_0) \quad \text{in} \quad [\text{Br}_2] = [\text{H}_2] + ([\text{Br}_2]_0 - [\text{H}_2]_0). \quad (5.18)$$

Če sedaj opazujemo primera, kjer začetni koncentraciji $[\text{H}_2]_0$ in $[\text{Br}_2]_0$ izmenično pošljemo k velikim vrednostim, ju odvod relativno malo spremeni in lahko iz relacije za časovno odvisnost koncentracij z meritavmi določimo vrednost konstante m . Za $[\text{H}_2]_0 \ll [\text{Br}_2]_0$ opazujemo vodik

$$t([\text{H}_2]) = -8/\sqrt{1 + [\text{H}_2]} + 4m \arctan \sqrt{1 + [\text{H}_2]} - C \quad (5.19)$$

in za $[\text{H}_2]_0 \gg [\text{Br}_2]_0$ opazujemo brom

$$t([\text{Br}_2]) = -4(m-2)(\sqrt{[\text{Br}_2]} - \sqrt{[\text{Br}_2]_0}) + 8[\text{Br}_2]_0(1/\sqrt{[\text{Br}_2]} - 1/\sqrt{[\text{Br}_2]_0}). \quad (5.20)$$



Slika 5.2.: Grafa spremenjanj koncentracij $[H_2]$, $[Br_2]$ in $[HBr]$ s časom, če je začetna koncentracija $[HBr]_0$ enaka 0 (zgoraj) in $[HBr]_0 = 10000$ (spodaj); slednjo vrednost izračunani $[HBr]$ odštejem. Pri obeh grafih so krivulje izrisane za tri različne vrednosti parametra $r = [H_2]_0/[Br_2]_0$. Videti je, da reakcija hitreje doseže ravnovesje pri neenakih razmerjih r , opazimo pa lahko tudi, da reakcija poteka najhitreje ob višji koncentraciji vodika napram bromu. Koncentracija končnega produkta $[HBr]$ je največja za vrednost $r = 1$. Na spodnjem grafu s primešanim $[HBr]_0$ je opaziti, počasnejšo dinamiko reaktantov. V primerjavi s prejšnjim grafom je reakcija vidno počasnejša, poleg tega pa primes na hitrost bolj vpliva pri reakciji z večjim r , torej je situacija obratna kot v primeru brez primesi $[HBr]_0$.

5.4. Kemijske ure

Pri zadnjem delu naloge reakcijo (5.4) najprej prevedemo v sistem diferencialnih enačb, ki se s približki poenostavi v

$$\begin{aligned}\dot{[S_2O_8^{2-}]} &= -a[S_2O_8^{2-}][I^-] \\ \dot{[I^-]} &= -a[S_2O_8^{2-}][I^-]^2 + b[S_2O_3^{2-}][I_2]^2 \\ \dot{[I_2]} &= +a[S_2O_8^{2-}][I^-]^2 - b[S_2O_3^{2-}][I_2]^2 \\ \dot{[S_2O_3^{2-}]} &= -b[S_2O_3^{2-}][I_2]\end{aligned}\quad (5.21)$$

Če poskusimo računati s tem pristopom je dobijena kinematika napačna, ker bi pomenilo, da reakcija steče samo, ko se po naključju vse tri molekule hkrati srečajo; kar bi bilo v resnici zelo počasno. Rezultati so predstavljeni na grafih slike 5.3. Ker pa sta posamezni reakciji v resnici sosledji dveh binarnih reakcij, ki potekata preko kratkoživih prehodnih stanj (5.5) in (5.6), ju prevedemo na sistem

$$\begin{aligned}\dot{[S_2O_8^{2-}]} &= -a_1[S_2O_8^{2-}][I^-] \\ \dot{[I^-]} &= -a_1[S_2O_8^{2-}][I^-] - a_2[IS_2O_8^{3-}][I^-] + b_1[S_2O_3^{2-}][I_2] + b_2[IS_2O_3^-][S_2O_3^{2-}] \\ \dot{[IS_2O_8^{3-}]} &= a_1[S_2O_8^{2-}][I^-] - a_2[IS_2O_8^{3-}][I^-] \\ \dot{[I_2]} &= a_2[IS_2O_8^{3-}][I^-] - b_1[S_2O_3^{2-}][I_2] \\ \dot{[2SO_4^{2-}]} &= a_2[IS_2O_8^{3-}][I^-] \\ \dot{[S_2O_3^{2-}]} &= -b_1[S_2O_3^{2-}][I_2] - b_2[IS_2O_3^-][S_2O_3^{2-}] \\ \dot{[IS_2O_3^-]} &= b_1[S_2O_3^{2-}][I_2] - b_2[IS_2O_3^-][S_2O_3^{2-}] \\ \dot{[S_4O_6^{2-}]} &= b_2[IS_2O_3^-][S_2O_3^{2-}]\end{aligned}\quad (5.22)$$

Sedaj najprej upoštevamo približek stacionarnega stanja $[IS_2O_8^{3-}] = [IS_2O_3^-] = 0$ ter končna produkta $[2SO_4^{2-}]$ in $[S_4O_6^{2-}]$ odmislimo; saj nas kot mehanizem merjenja časa zanima le porabljjanje $[S_2O_3^{2-}]$. Sistem (5.22) se tako poenostavi v

$$\begin{aligned}\dot{[S_2O_8^{2-}]} &= -a_1[S_2O_8^{2-}][I^-] \\ \dot{[I^-]} &= -a_1[S_2O_8^{2-}][I^-] - a_2[IS_2O_8^{3-}]_0[I^-] + b_1[S_2O_3^{2-}][I_2] + b_2[IS_2O_3^-]_0[S_2O_3^{2-}] \\ \dot{[I_2]} &= a_2[IS_2O_8^{3-}][I^-] - b_1[S_2O_3^{2-}][I_2] \\ \dot{[S_2O_3^{2-}]} &= -b_1[S_2O_3^{2-}][I_2] - b_2[IS_2O_3^-]_0[S_2O_3^{2-}]\end{aligned}\quad (5.23)$$

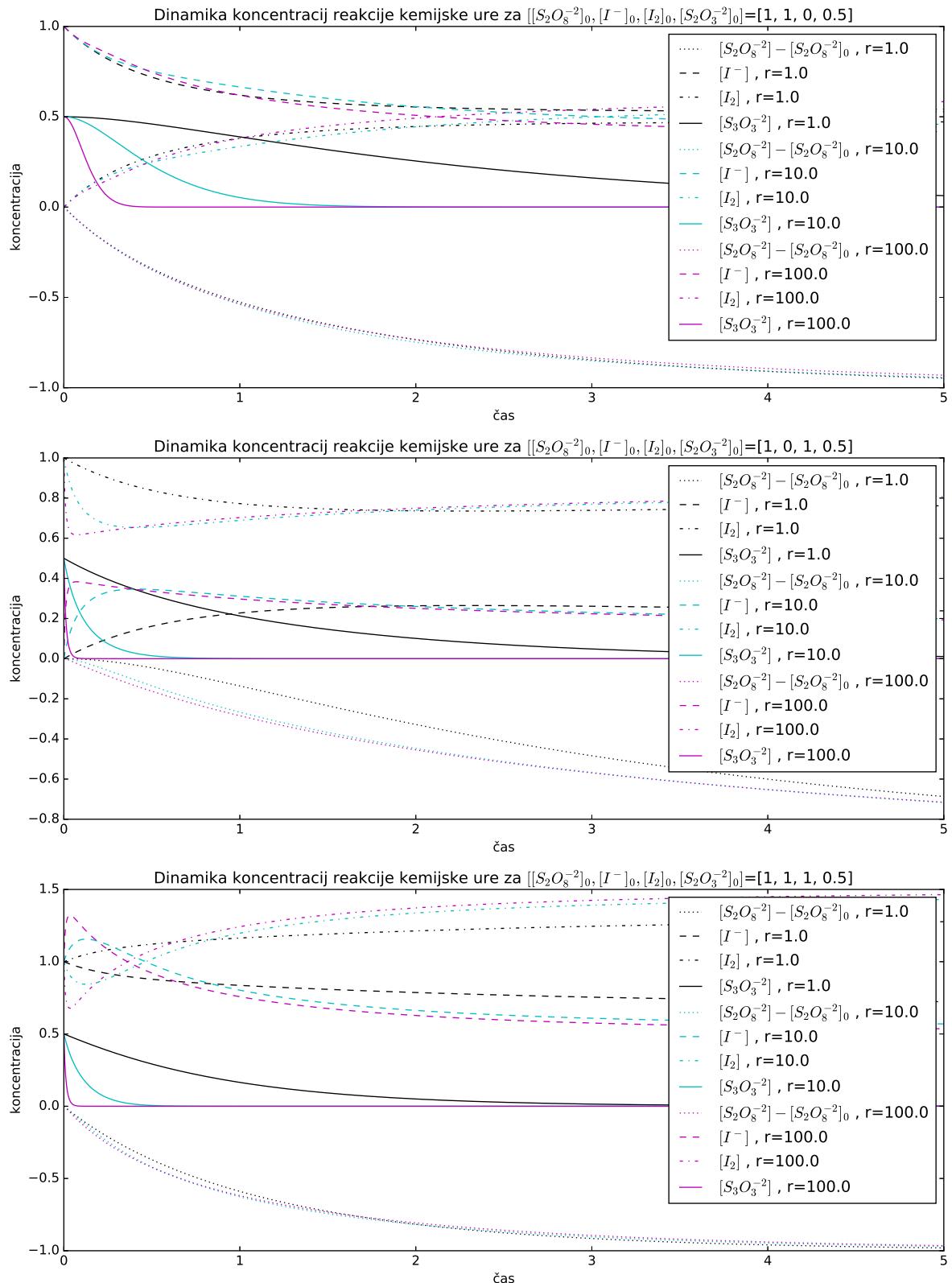
s konstantami $a_1 \ll a_2$, $b_1 \ll b_2$, $[IS_2O_3^-]_0$ in $[IS_2O_8^{3-}]_0$. Sistem se nadalje z zvezama

$$a_2[IS_2O_8^{3-}]_0[I^-] = a_1[S_2O_8^{2-}][I^-] \quad \text{in} \quad b_2[IS_2O_3^-]_0[S_2O_3^{2-}] = b_1[S_2O_3^{2-}][I_2] \quad (5.24)$$

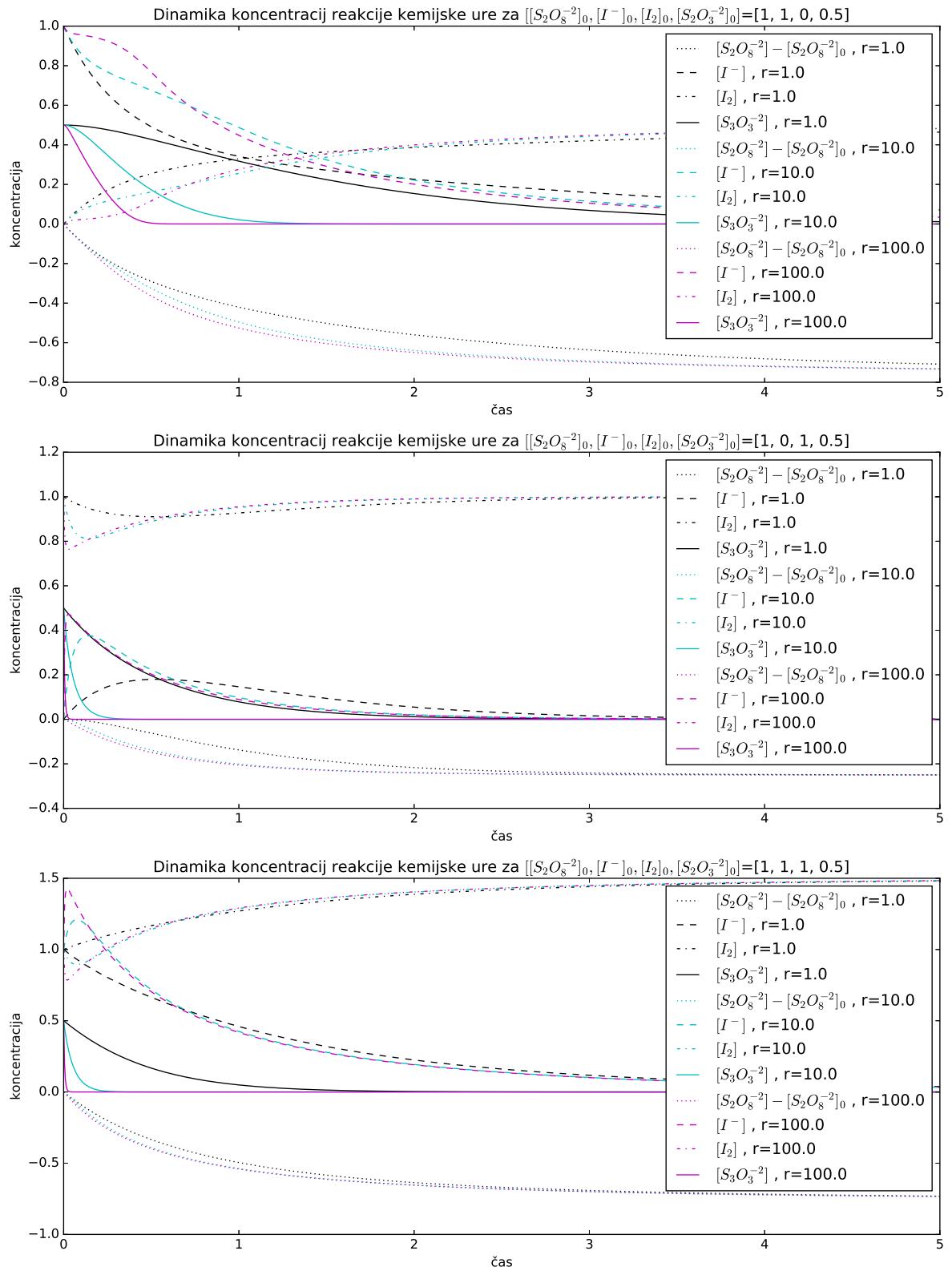
prevede v obliko

$$\begin{aligned}\dot{[S_2O_8^{2-}]} &= -a_1[S_2O_8^{2-}][I^-] \\ \dot{[I^-]} &= -2a_1[S_2O_8^{2-}][I^-] + 2b_1[S_2O_3^{2-}][I_2] \\ \dot{[I_2]} &= a_2[IS_2O_8^{3-}][I^-] - b_1[S_2O_3^{2-}][I_2] \\ \dot{[S_2O_3^{2-}]} &= -2b_1[S_2O_3^{2-}][I_2]\end{aligned}\quad (5.25)$$

s konstantama $a_1 \ll b_1$. Rezultati so predstavljeni na grafih slike 5.4.

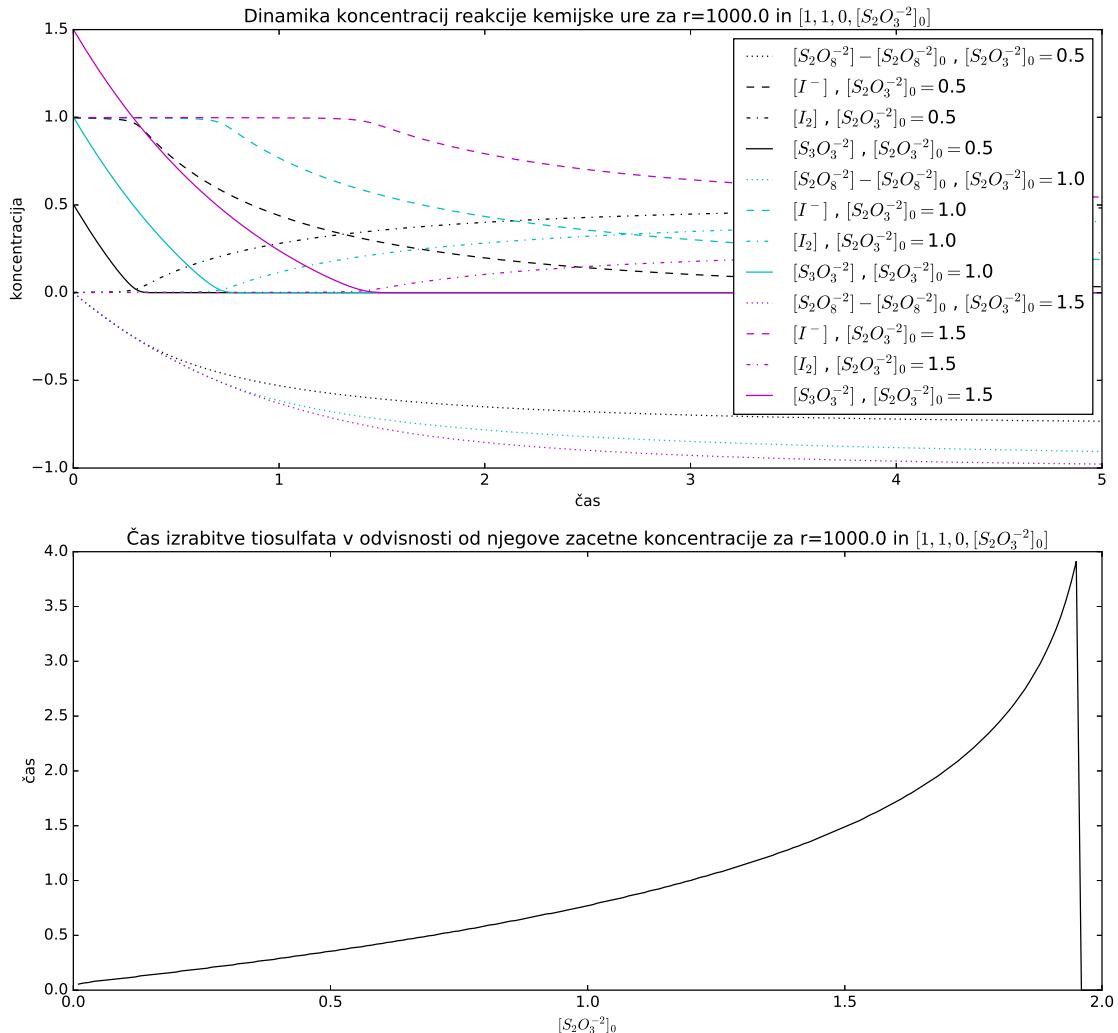


Slika 5.3.: Grafi prikazujejo kinematike koncentracij reaktantov za nekaj izbranih kombinacij začetnih pogojev in sicer vsak še pri treh razilčnih vrednosti razmerja hitrosti reakcij $r = b/a$. Računane so preko sistema (5.21). Koncentracije $[S_2O_3^{2-}]$ vedno padajo proti ničli, dokler se pa to ne zgodi, $[I^-]$ teži k višjim vrednostim, $[I_2]$ pa k nižjim.



Slika 5.4.: Grafi prikazujejo kinematike koncentracij reaktantov za nekaj izbranih kombinacij začetnih pogojev in sicer vsak še pri treh razilčnih vrednostih razmerja hitrosti reakcij $r = b/a$. Računane so preko sistema (5.25). Koncentracije $[S_2O_3^{2-}]$ vedno padajo proti ničli, dokler se pa to ne zgodi, $[I^-]$ teži k višji vrednosti ravnovesja, $[I_2]$ pa k nižji. Najzanimivejši je primer kombinacije začetnih pogojev prikazan na zgornjem grafu. Tu sta joda že od začetka v ravnovesju, dokler se koncentracija tiosulfata toliko ne zniža, da se $[I_2]$ začne nabirati in $[I^-]$ porabljati. Za večje vrednosti r je poraba hitrejša in prehodi ostrejši.

Nazadnje lahko analiziramo čas izrabe vsega začetnega reaktanta $[S_2O_3^{2-}]$ tekom reakcije kemiske ure v odvisnosti od njegove začetne koncentracije. Graf kinematike koncentracij reaktantov za razilčne začetne vrednosti $[S_2O_3^{2-}]_0$ in graf časa porabe $[S_2O_3^{2-}]$ v odvisnosti od njegove začetne koncentracije, sta na sliki 5.5. Iz teh grafov razberemo, da s spremembo $[S_2O_3^{2-}]_0$ lahko uravnavamo čas začetka porabljanja/nastajanja joda. Odvisnost ima eksponentno obliko.



Slika 5.5.: Graf (zgoraj) prikazuje kinematiko koncentracij reaktantov pri treh razilčnih vrednostih začetne koncentracije $[S_2O_3^{2-}]_0$. Večanje začetne koncentracije omejuje manjšanje I^- oz. nabiranje I_2 , ki se za manjše začetne koncentracije tiosulfata prej zganeta z mesta. Graf (spodaj) prikazuje odvisnost za čas začetka porabe/nastajanja joda od $[S_2O_3^{2-}]_0$ in ta odvisnost ima eksponentno obliko. Nadaljnje večanje koncentracije bi najverjetneje povzročilo kopičenje tiosulfata, na koncentracijo joda pa to ne bi vplivalo.

Literatura

- [1] Spletne strani z gradivom predmeta, ogled 20.10.2017,
<http://predmeti.fmf.uni-lj.si/modelska>

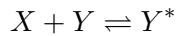
6. Luščenje modelskih parametrov: linearni modeli

Povzetek

Pri tej vaji se z uporabo linearnih metod preizkusimo v analizi različnih merjenih podatkov. Najprej analiziramo farmakološke podatke, kjer spoznamo, da lahko linearizacija problema v nekaterih primerih izboljša rezultate fita in poda manjše negotovosti. V drugi nalogi poskusimo na podlagi podatkov izdelati linearni model, ki bi čim bolje opisal preslikavo med podatki iz gorišča in podatki tarče; za namen rekonstrukcije trajektorij delcev v spektrometru. Pri zadnji nalogi na podlagi meritvev standardov dveh snovi izdelamo linearen model, za določitev odstotnega deleža le teh v dveh meritvah rastlinskih vzorcev.

6.1. Naloga

- V farmakologiji merijo odziv tkiv na različne reagente. Za večino teh pojavov lahko privzemo, da gre za reakcijo, kjer spremljamo vezavo molekul reagenta X na receptorje Y v tkivu.



V stacionarnem stanju dobimo zvezo

$$y = \frac{y_0 x}{x + a}, \quad (6.1)$$

kjer pomeni y_0 nasičeni odziv tkiva in a koncentracijo, potrebno za odziv, ki je enak polovici nasičenega.

Iz merskih podatkov v datoteki `farmakoloski.dat`[2], določi parametra y_0 in a . Napaka v meritvi odziva je v vsem področju enaka trem enotam. Zvezo lahko lineariziramo. Pazi, kako se pri tem transformirajo napake.[1]

- Za uporabo visokoločljivostnega magnetnega spektromетra potrebujemo preslikavo, ki iz izmerjenih količin rekonstruira parametre trajektorije delcev, potrebne za izračun energije in drugih kinematičnih količin. V datoteki `thtg - xfp - thfp.dat`[2] najdete kalibracijske podatke s stolpci ϑ_{tg} (disperzijski kot na tarči v stopinjah) ter x_{fp} (položaj v goriščni ravnini v milimetrih) in ϑ_{fp} (kot v goriščni ravnini v stopinjah). Natančnosti meritev kotov so velikostnega reda miliradianov, položajev pa okrog milimetra. Sestavi varčni model za preslikavo

$$(x_{fp}, \vartheta_{fp}) \mapsto \vartheta_{tg}.$$

Uporabiš lahko na primer najnižje potence x_{fp} in ϑ_{fp} ali pa kakšne druge funkcije teh dveh spremenljivk.[1]

- Podrobnosti v profilu rentgenskih absorpcijskih robov so odvisne od kemijske okolice elementa. Teorijske napovedi profila še niso dovolj natančne in zanesljive, zato si pri analizah snovi pomagamo s standardi. V datoteki `CdL3_linfit.norm`[2] so zbrani širje absorpcijski spektri kadmija na robu L_3 iz študije, kako ta kovina učinkuje na rastline. V prvih dveh vzorcih so izolirane celične stene iz krovne plasti in iz sredice listov rastline C . Thlaspi, ki je znan hiperakumulator težkih kovin. Zadnja dva spektra sta dobljena na standardih, kompleksih Cd sulfata z glutationom (GSH) in pektinom: v prvem je Cd vezan izključno na žveplo, v drugem na kisik. V listnih vzorcih dopuščamo obe vrsti vezave, vemo pa, da sta prispevka obeh v spektru linearno sestavljena. Določi odstotno razmerje vezi $Cd - O$ in $Cd - S$ v obeh listnih vzorcih.[1]

Komentar k reševanju

Naloga je rešena v Pythonu. Pri fitanju je uporabljena vgrajena funkcija iz modula `scipy.optimize`, ki jo kličemo z `curve_fit`. Ta omogoča prilaganje modelske funkcije k merjenim podatkom tudi z upoštevanjem napake v meritvi. Za faktorizacijo matrik je uporabljena funkcija iz modula `numpy.linalg`, ki jo kličemo z `svd`.

6.2. Farmakološki model

Pri fitanju farmakoloških podatkov iz prve naloge, je potrebno zvezo (6.1) najprej linearizirati

$$\frac{1}{y} = \frac{a}{y_0} \cdot \frac{1}{x} + \frac{1}{y_0}$$

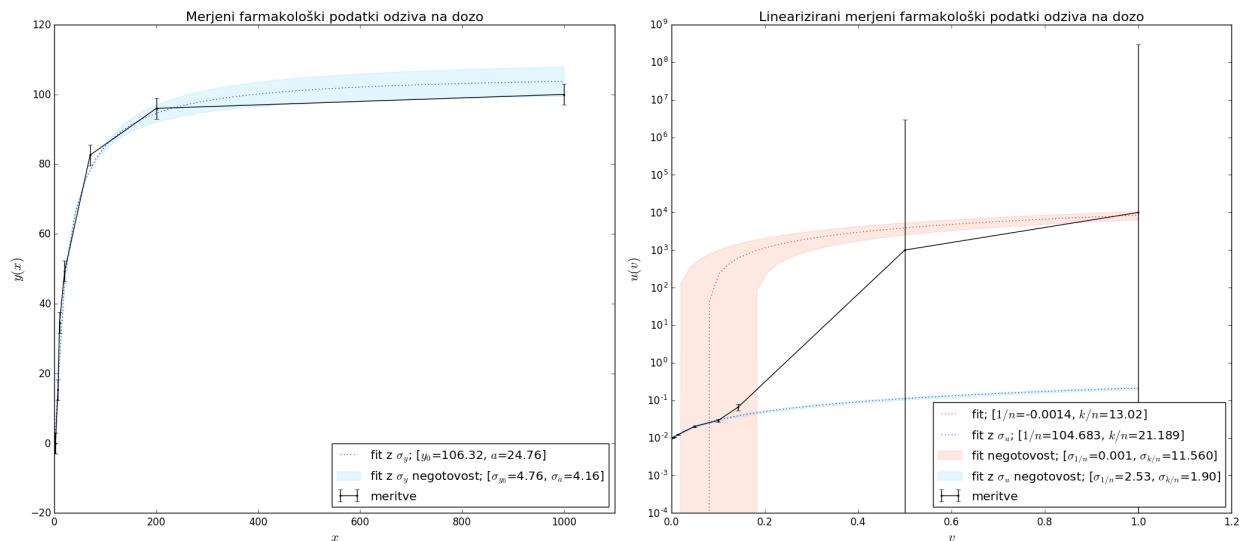
Po uvedbi novih spremenljivk $u = 1/y$ in $v = 1/x$ in koeficientov $n = 1/y_0$ in $k = a/y_0$, dobi zveza (6.2) očitno linearno obliko

$$u = k \cdot v + n . \quad (6.2)$$

Koeficiente k in n s prilagajenjem fit funkcije določimo iz podatkov, od koder potem izračunamo iskana a in y_0 . Preden pričnemo s fitanjem je zaradi linearizacije potrebno transformirati še napako meritve $\sigma_y = 3$ kot

$$\sigma_u = \sigma_y \cdot \left| \frac{\partial u}{\partial y} \right| = \sigma_y \cdot \left| \frac{1}{y^2} \right| . \quad (6.3)$$

Očitno za zelo majhne vrednosti y napaka σ_u linearizbrane zveze zelo hitro narašča. To lahko vidimo, če podatke iz meritve za oba primera narišemo skupaj z intervali napake, kot je predstavljeno sliki 6.1. Poleg izrisa podatkov so teh grafih prikazani fiti modelske funkcije (6.1) in njene linearizirane oblike (6.2). Slednja je fitana z uporabo vgrajene python funkcijen dva načina, in sicer najprej brez z upoštevanja napak meritve in nato še z upoštevanjem. Rezultati izluščenih parametrov nasičenega odziva tkiva y_0 in koncentracije a so očitno boljši za linearizirano obliko, saj sta σ_{y_0} in σ_a najmanjši, vendar le ob upoštevanju napak σ_u pri fitanju. V kolikor namreč `curve_fit` apliciramo brez poznanih σ_u , je rezultat popolnoma napačen in je omenjeno fit funkcijo bolje uporabiti direktno na nelineariziranem modelu.



Slika 6.1.: Graf direktnih meritev odziva tkiva na reagent (levo) in graf istih podatkov v linearizirani obliki (desno); za oba primera so prikazane tudi napake meritve σ_y in σ_u . Na podatke sta z uporabo `curve_fit` prilagojeni modelski funkciji (6.1) in (6.2). Slednjo prilagajamo tako z upoštevanjem informacije o σ_u , kar je prikazano v 'modri' barvi, in brez, ki je v 'rdeči'. Prilaganje lineariziranega modela z upoštevanjem napak je dalo najboljše rezultate z najmanjšo negotovostjo obeh izluščenih parametrov $y_0 = 1/n$ in $a = k/n$. Vrednost statističnega testa odstopanj med merjenimi vrednostmi in napovedmi tako dobljenega modela (iz fita ki upošteva napake) je $\chi^2 = 26.08$ za y_0 in a ; oziroma $\chi^2 = 6.24$ za $k = 0.00955 \pm 0.00023$ in $n = 0.202 \pm 0.017$ lineariziranega modela.

6.3. Varčni model preslikave visokoločljivostnega magnetnega spektrometra

V primeru podatkov kalibracije spektrometra je modelska funkcija preslikave (2) bolj kompleksna. Še vedno imamo linearno parametrizacijo, ki je v splošnem oblike

$$f(x) = a_1 \cdot f_1(x) + a_2 \cdot f_2(x) + a_3 \cdot f_3(x) + \dots + a_M \cdot f_M(x) , \quad (6.4)$$

vendar je število parametrov M takega modela preštevilčno, za udobno aplikacijo algoritma `curve_fit`. Da bi izluščili optimalne parametre modela (6.4) z veliko parametri in pri tem iskali najvarčnejši model, lahko za namen linearne regresije zbranih meritve po viru [2] definiramo matriko A z elementi

$$A_{ij} = \left[\frac{f_j(x_i)}{\sigma_{y_i}} \right] , \quad (6.5)$$

kjer i stoji za zaporedno število točke posamezne meritve x_i z negotovostjo σ_{x_i} in $j = 1, 2, \dots, M$ za indeks parametra izbranega modela. Ta matrika modela mora biti povezana z izmerki y_i v točkah x_i . Slednje pospravimo v vektor \vec{b} z elementi

$$b_i = \frac{y_i}{\sigma_{y_i}} . \quad (6.6)$$

Da bi od tu dobili parametre a_j je potrebno matriko A najprej faktorizirati. Za ta namen lahko uporabimo algoritem `svd`, ki matriko razcepi v tri matrike $A = USV^\dagger$. Stolpci U_j , diagonalni elementi S_{jj} in stolci V_j skupaj z vektorjem \vec{b} podajo vektor \vec{a} iskanih parametrov a_j kot

$$\vec{a} = \sum_{j=1}^M \left(\frac{U_j \cdot \vec{b}}{S_{jj}} \right) V_j . \quad (6.7)$$

Izračunamo lahko tudi vektor negotovosti tako določenih posameznih parametrov kot

$$\sigma_{a_j}^2 = \sum_{k=1}^M \left(\frac{V_{kj}}{S_{kk}} \right)^2 . \quad (6.8)$$

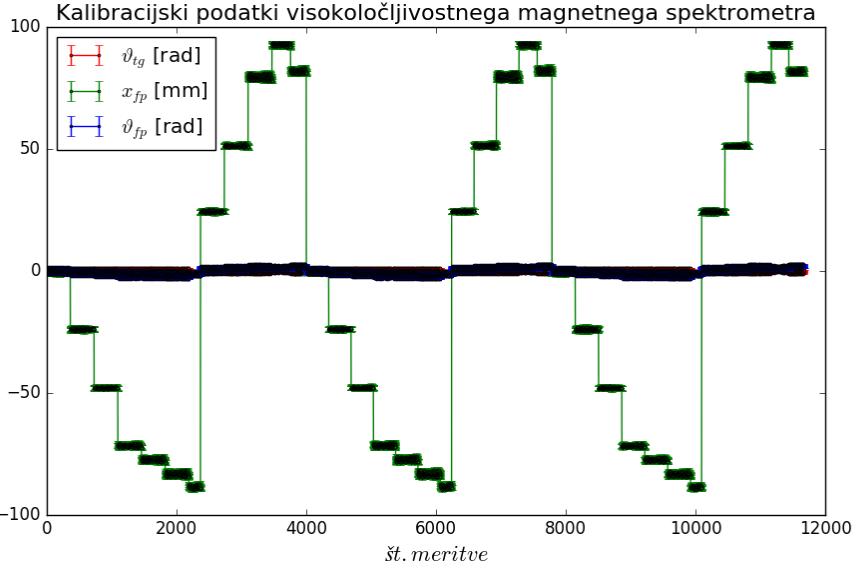
Dobro mero za ustreznost tako dobljenih parametrov dobimo s χ^2 testom, ki v točkah x_i izmerjene y_i podatke primerja z v istih točkah preko modela (6.4) določenimi vrednostmi \tilde{y}_i kot

$$\chi^2 = \sum_i \left(\frac{y_i - \tilde{y}_i}{\sigma_{y_i}} \right)^2 . \quad (6.9)$$

Spisano lahko preverimo na lineariziranem farmakološkem modelu, kjer tako dobimo vrednosti $n = 0.0096 \pm 0.0002$ in $k = 0.202 \pm 0.017$ ob $\chi^2 = 26.08$. Od tu dolocimo $y_0 = 1/n = 104.68 \pm 2.48$ in $a = k/n = 21.2 \pm 1.9$ in vidimo, da se vrednosti dobro ujemajo rezultati funkcije `curve_fit` prikazanimi na sliki 6.1. Prav tako se seveda ujema test $\chi^2 = 26.08$. Tako spisanemu algoritmu lahko torej zaupamo.

Za primer kalibracij spektrometra si najprej pogledamo podatke, ki so prikazani na sliki 6.2. Videti je, da gre za meritev ϑ_{tg} pri dvajstih različnih goriščnih razdaljah x_{fp} in kotih ϑ_{fp} . Iščemo torej model, ki bo za vsak par x_{fp} in ϑ_{fp} podal vrednost kota ϑ_{tg} , ki mora biti čim bližje izmerjeni vrednosti. Za linearno kombinacijo členov modelske funkcije izberem potence spremenljivk x_{fp} in ϑ_{fp} , lahko pa dodamo tudi potence raznih mešanih členov

$$M(x_{fp}, \vartheta_{fp}) = a_0 + a_1 \cdot f_1(x_{fp}) + a_2 \cdot g_1(\vartheta_{fp}) + a_3 \cdot h_1(x_{fp}, \vartheta_{fp}) + \dots , \quad (6.10)$$



Slika 6.2.: Graf kalibracijskih podatkov disperzijskega kota na tarči ϑ_{tg} v radianih, položaja v goriščni ravnini x_{fp} v milimetrih in kota v goriščni ravnini ϑ_{fp} v radianih. Natančnosti meritev kotov so velikostnega reda miliradianov, položajev pa okrog milimetra.

kjer so funkcije v posameznih členih potence

$$\begin{aligned} f_k(x_{fp}) &= (x_{fp})^k \\ g_k(\vartheta_{fp}) &= (\vartheta_{fp})^k \\ h_k(x_{fp}, \vartheta_{fp}) &= (x_{fp} \cdot \vartheta_{fp})^k \\ o_k(x_{fp}, \vartheta_{fp}) &= (x_{fp} + \vartheta_{fp})^k \\ p_k(x_{fp}, \vartheta_{fp}) &= (x_{fp} - \vartheta_{fp})^k \\ r_k(x_{fp}, \vartheta_{fp}) &= (\vartheta_{fp}/x_{fp})^k \end{aligned}$$

za $k = 1, 2, \dots, m$. V tem primeru dobimo matriko A z vrsticami oblike

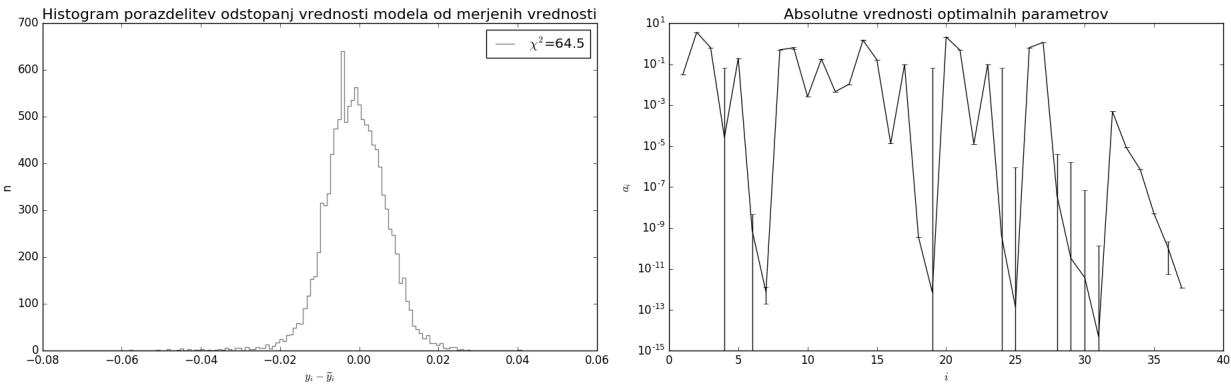
$$A_i = \left[\begin{array}{ccccccc} \frac{1}{\vartheta_{tgi}} & \frac{f_1(x_{fp_i})}{\sigma_{\vartheta_{tgi}}} & \frac{g_1(\vartheta_{fp_i})}{\sigma_{\vartheta_{tgi}}} & \frac{h_1(\vartheta_{fp_i}, x_{fp_i})}{\sigma_{\vartheta_{tgi}}} & \frac{o_1(x_{fp_i})}{\sigma_{\vartheta_{tgi}}} & \frac{p_1(\vartheta_{fp_i})}{\sigma_{\vartheta_{tgi}}} & \frac{r_1(\vartheta_{fp_i}, x_{fp_i})}{\sigma_{\vartheta_{tgi}}} & \dots \end{array} \right],$$

kjer i stoji za zaporedno število posamezne meritve s pripadajočimi negotovostmi. Ta matrika modela je povezana z izmerki ϑ_{tg} , ki jih pospravimo v vektor \vec{b} z elementi

$$b_i = \frac{\vartheta_{tgi}}{\sigma_{\vartheta_{tgi}}}.$$

Sedaj lahko uporabimo algoritem svd, in določimo vektor \vec{a} iskanih parametrov ter vektor njihovih negotovosti $\vec{\sigma}_a^2$. Ustreznost tako dobljenih parametrov preverimo s χ^2 testom. Izračun sem izvedel za funkcije do potenc $k = 6$, kar pomeni, da je bilo potrebno določiti 37 parametrov. Vrednosti teh so poleg histograma odstopanj vrednosti takega modela od merjenih podatkov prikazane na sliki 6.3. Videti je, da so odstopanja od meritev lepo gaussovsko raztresena okrog pravih vrednosti in tudi velikosti niniso prav velike. Sami parametri imajo po večini majhne vrednosti, saj kompenzirajo člene z najvišjimi potencami. Ker imajo nekateri od parametrov precej velike negotovosti, je toliko bolj zaželjena statistična ocena modela. Oceno ustreznosti modela poda reducirani $\frac{\chi^2}{N-M}$ test, kjer

delimo z razliko $N - M$, ki predstavlja število prostostnih stopenj; $N = 11665$ je število izmerkov in $M = 37$ število parametrov v modelu. Želimo, da je vrednost čim bližje ena, vendar v primeru moje modelske funkcije test poda vrednost okrog 60, kar pomeni, da model žal ni najboljši.



Slika 6.3.: Graf kalibracijskih podatkov disperzijskega kota na tarči ϑ_{tg} v radianih, položaja v goriščni ravnini x_{fp} v milimetrih in kota v goriščni ravnini ϑ_{fp} v radianih. Natančnosti meritev kotov so velikostnega reda miliradianov, položajev pa okrog milimetra.

6.4. Rentgenski absorpcijski spekter kadmija v rastlini *C.Thalpsi*

Za primer določanja odstotnega deleža vezi kadmija Cd z žveplom S in kisikom O v rastlinah si najprej pogledamo podatke iz študije, ki so prikazani na sliki 6.4. Glede na standarda lahko sklepamo, da se na krovni plasti listov kadmij veže predvsem na žveplo, ko prodre v sredico pa je večji meri vezan tudi na kisik. Dopustiti moramo torej obe vrsti vezave, zato za posamezen vzorec definiramo model

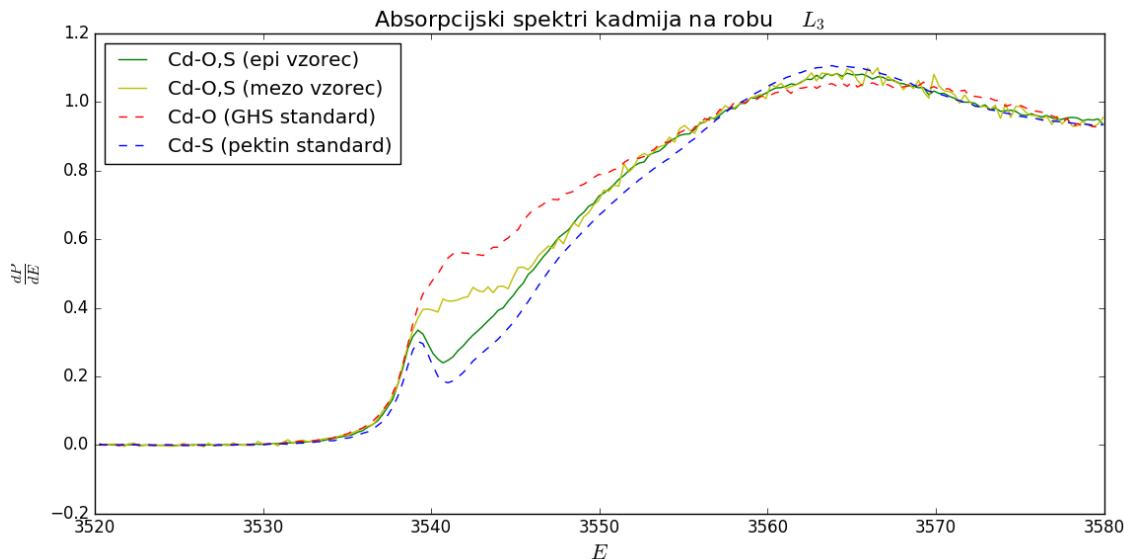
$$M(E) = C_O \cdot F_{GHS}(E) + C_S \cdot F_{pekt}(E),$$

kjer sta C_O in C_S deleža koncentracij vezi $Cd - O$ in $Cd - S$, funkciji F pa predstavljata absorpcijski spekter posameznega standarda. Vsoto tako obteženih vrednosti obeh standardov v vsaki izmerjeni točki E želimo čim bolj približati izmerjenim vrednostim posameznega vzorca lista. Za primer vzorca površine listov torej pričakujemo večji C_S in manjši C_O .

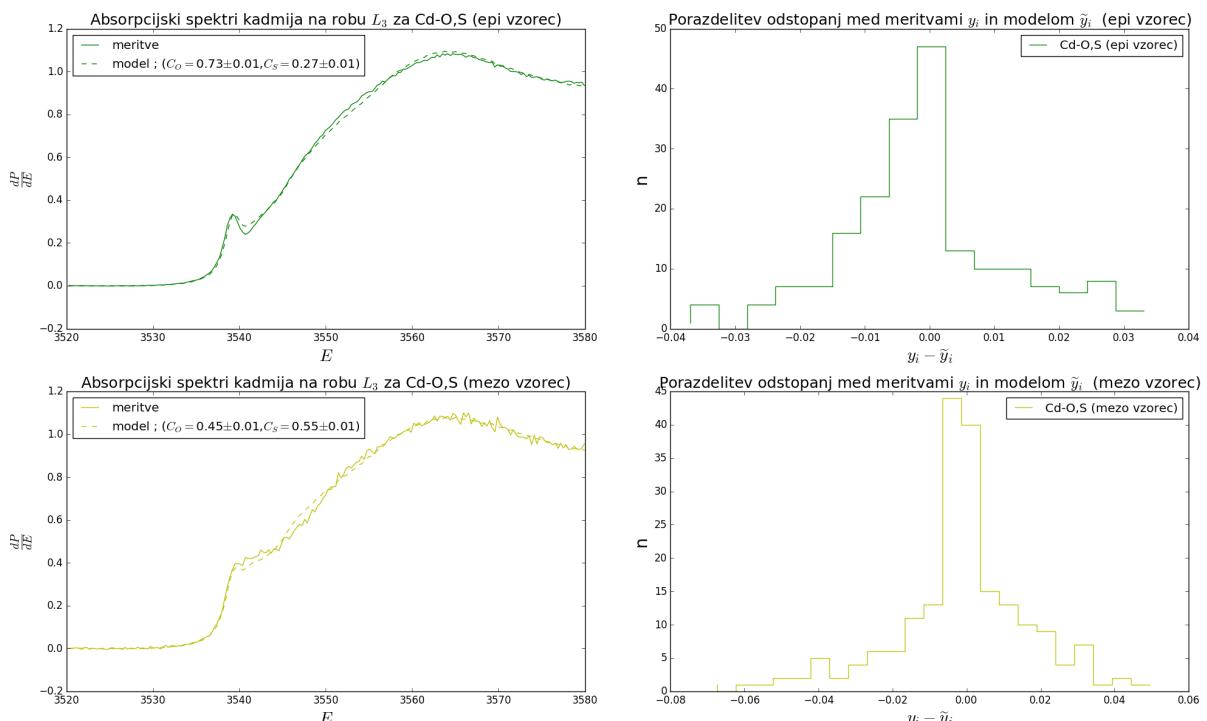
Rezultati takšnega modela so prikazani na sliki 6.5. Izgleda, da se vrednosti modelskih funkcij M pri obeh primerih dobro prilegajo merjenim podatkom. Iz histograma odstopanj modela od merjenih vrednosti v posameznih točkah vidimo, da je fit malenkost boljši za vzorec krovne plasti listov, saj je v tem primeru vrednost največjega odstopanja manjša, prav tako pa je ožja tudi širina porazdelitve odstopanj. Samo razmerje med deležem vezi pri vsakem vzorcu določimo kot

$$R_{O/S} = \frac{C_O}{C_S}.$$

Za vzorec krovne plasti listov dobimo iz fita vrednosti $C_O = 0.73 \pm 0.01$ in $C_S = 0.27 \pm 0.01$, ki nam podata razmerje $R_{O/S} = 2.68 \pm 0.1$, za vzorec iz sredine listov pa $C_O = 0.45 \pm 0.01$ in $C_S = 0.55 \pm 0.01$, ki nam podata razmerje $R_{O/S} = 0.80 \pm 0.03$. Tu je treba opozoriti, da tokrat v datoteki ni bilo na voljo nobenih napak meritev vzorcev in standardov, zato tudi niso upoštevane pri luščenju parametrov. Predstavljeni negotovosti rezultatov so tako zagotovo podcenjene, zato jih gre jemati s precej rezerve.



Slika 6.4.: Štirje absorpcijski spektri kadmija Cd na robu L_3 . V prvih dveh vzorcih so izolirane celične stene iz krovne plasti (epi) in iz sredice (mezo) listov rastline *C. Thlaspi*. Spektra sta dobljena na standardnih, kompleksih Cd sulfata z glutationom (GSH) in pektinom: v prvem je Cd vezan izključno na žveplo, v drugem na kisik. V listnih vzorcih dopuščamo obe vrsti vezave.



Slika 6.5.: Levo sta prikazana absorpcijska spektra kadmija Cd na robu L_3 iz krovne plasti (zgoraj) in iz sredice (spodaj) listov ter primerjana z izračuni v istih točkah energije žarkov po modelu (6.4). Desno poleg spektrov sta prikazana histograma odstopanj vrednosti meritov in izračunov po fitanem modelu za točke pri istih energijah žarkov. Iz spektrov in histogramov je videti da je fit boljši za primer vzorca iz krovne plasti.

Literatura

- [1] Spletne strani z gradivom predmeta, ogled 20.10.2017,
<http://predmeti.fmf.uni-lj.si/modelska>
- [2] Spletne strani z gradivom predmeta, ogled 20.10.2017,
<http://predmeti.fmf.uni-lj.si/modelska/podatki/>
- [3] Verjetnost v fiziki, Simon Širca, DMFA, 2016

7. Luščenje modelskih parametrov: razdelčni in nelinearni modeli

Povzetek

Pri tej vaji se spoznamo še z nelinearno metodo luščenja modelskih parametrov. V treh nalogah poskusimo oblikovati različne modele s katerimi fitamo podatke, pri tem pa skušamo izbiro s testom χ^2 statistično upravičiti izbiro najboljšega med njimi. V prvem delu ponovno analiziramo farmakološke podatke on ugotovimo, da je zanje ustreznejša nelinearna parametrizacija. V drugi nalogi se odločamo med eno in dvo-razdelčnim modelom čistilnosti ledvic, pri zadnji nalogi pa skušamo ujeti karakteristiko meritve in določiti parametre korozije na izmerkih, ki nimajo podanih negotovosti.

7.1. Naloga

1. Farmakološki model iz Naloge 6 lahko razširimo z vpeljavo novega parametra p ,

$$y = \frac{y_0 x^p}{x^p + a^p}. \quad (7.1)$$

kjer pomeni y_0 nasičeni odziv tkiva in a koncentracijo, potrebno za odziv, ki je enak polovici nasičenega.

Iz merskih podatkov v datoteki `farmakoloski.dat`[2], določi parametre y_0 , a in p . Razišči statistično upravičenost dodatnega parametra.[1]

2. Poišči najboljšo vrednost za čistilnost ledvic iz kliničnih podatkov v datoteki `ledvice.dat`[2] uporabo enorazdelčnega in dvorazdelčnega modela ter primerjaj rezultate. Ali je dodatek aditivne konstante (“ozadje” pri štetju razpadov) statistično upravičen? Poskusih lahko tudi s funkcijo $\exp(-\lambda\sqrt{t})$, ki jo izvedemo iz bolj zapletenih modelov. Spremenljivka t v podatkih je čas na sredi vsakega merilnega intervala.

Pri dvorazdelčnem modelu lahko za začetni približek vzameš eksponentni konstanti v razmerju 1:10.[1]

3. Parametre korozije določajo iz $U-I$ diagrama med kovino in korozivnim elektrolitom. Modelske nastavki lahko v najpreprostejši obliki zapišemo

$$I = I_0 \left[\exp\left(\frac{U}{U_a}\right) - \exp\left(-\frac{U}{U_c}\right) \right]. \quad (7.2)$$

Določi parametre I_0 , U_a in U_c iz meritve v tabeli. Podatki so v datoteki `korozija.txt`[2].

Merska napaka v toku je konstantna, napetosti privzamemo za točne. Ujemanje lahko izboljšamo, če dodamo v nastavek še popravek $U \rightarrow (U - U_0)$, ker meritve ne gredo točno skozi izhodišče.[1]

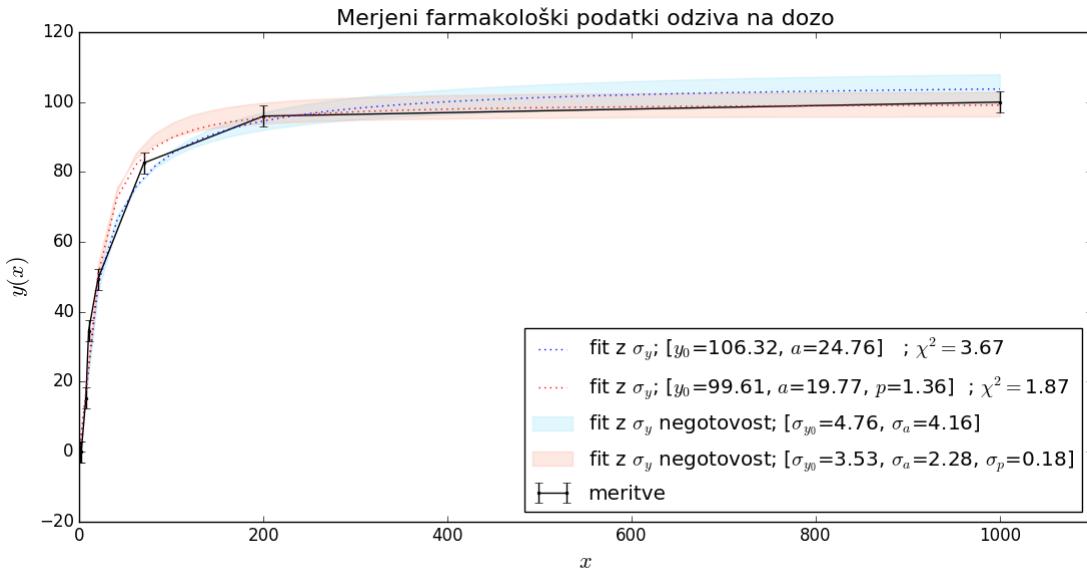
Komentar k reševanju

Splošne nelinearne probleme rešujemo z uporabo numeričnih minimizacijskih metod, od katerih je posebej robustna Levenberg-Marquardtova metoda. V *Numerical Recipes* je na voljo v funkciji `mrgmin`, ki kliče pomožne rutine `mrgcuf`, `gaussj` in `covsrt`. V `scipy.optimize` je treba nastaviti `method = 'lm'`. Podobno jo lahko izberemo v Mathematici in v matlabu. V tej nalogi je bila uporabljenega vgrajena funkcija iz modula `scipy.optimize`, ki jo kličemo z `curve_fit` in omogoča aplikacijo Levenberg-Marquardtove metode.

7.2. Farmakološki model

Pri fitanju farmakoloških podatkov iz prve naloge prejšnje vaje, model (6.1) z vpeljavo novega parametra p razširimo v (7.1) in ob uporabi funkcije `curve_fit` za oba primera izluščimo iskane parametre.

Graf meritve in obeh modelskih funkcij z različnima številoma parametrov prostih parametrov si lahko ogledamo na sliki 7.1, od koder je očitno, da sta prilagojeni funkciji modelov različni. Metodi lahko med seboj primerjamo z uporabo reduciranega testa χ^2 , kjer (6.9) delimo z razliko $N - M$, ki predstavlja število prostostnih stopenj; N je število izmerkov in M število parametrov v modelu. Fit je najboljši takrat, ko je vrednosti reduciranega χ^2 blizu ena; če je njegova vrednost večja je modelska funkcija neoptimalna, če pa je manjša od ena se model dobro prilagaja, vendar to nakazuje, da podatke pre-prilagodi. V našem primeru je po tem kriteriju prilaganje modela razširjenega s parmetrom p dalo najboljši rezultat z najmanjšimi negotovostmi in vrednostjo statističnega testa, ki je bližje ena.



Slika 7.1.: Graf direktnih meritev odziva tkiva na reagent s fitoma modelov (6.1) (modra) in (7.1) (rdeča), ki oba upoštevata napake izmerkov. V legendi so prikazane vrednosti izluščenih parametrov, njihove negotovosti ter reducirane χ^2 vrednosti. Prilagajanje modela razširjenega s parmetrom p je dalo boljši rezultat z manjšimi negotovostmi in vrednostjo statističnega testa, ki je bližje ena.

7.3. Čistilnost ledvic

V primeru kliničnih podatkov za čistilnost ledvic, ki sestojijo iz časovnih intervalov v katerih je izmerjeno število razpadov radioaktivnih delcev N snovi v krvi, nas v grobem zanimata dva modela. Prvi je enorazdelčni s katerim aproksimiramo delovanje ledvic, ki kot rezervoar krvi le to neprestano prečiščujejo. Za neko koncentracijo c sevalca v krvi čistilnost ledvic z uporabo enorazdelčnega modela opisemo z

$$\dot{c} = -\frac{\epsilon\phi}{V} \cdot c = -\lambda \cdot c , \quad (7.3)$$

kjer je ϵ čistilna sposobnost ledvic, ϕ fluks krvi skozi ledvice in V prostornina krvi v ledvicah.

Naprednejši model čistilnosti je dvorazdelčni, ki poleg ledvic dopušča prehajanje sevalca še v okolico ledvic. Opišemo ga z

$$\begin{aligned} \dot{c}_1 &= -\lambda \cdot c_1 - \mu(c_1 - c_2) \\ \dot{c}_2 &= \quad +\mu(c_1 - c_2) , \end{aligned} \quad (7.4)$$

kjer konstanta μ opisuje pretok snovi med ledvicami in preostalim delom telesa.

7.3.1. Enorazdelčni model z linearizacijo

Z integracijo izraza za enorazdelčni model (7.3) dobimo časovno odvisnost koncentracije v obliki

$$c(t) = A e^{-at} . \quad (7.5)$$

Poskusimo pa lahko tudi z korensko odvisnostjo od časa

$$c(t) = A e^{-a\sqrt{t}} . \quad (7.6)$$

Podobno kot pri farmakološčem modelu prejšnje vaje pa lahko pri fitanju zgornji zvezi najprej lineariziramo v

$$\tilde{c} = \ln c(t) = -at + \ln(A) \quad (7.7)$$

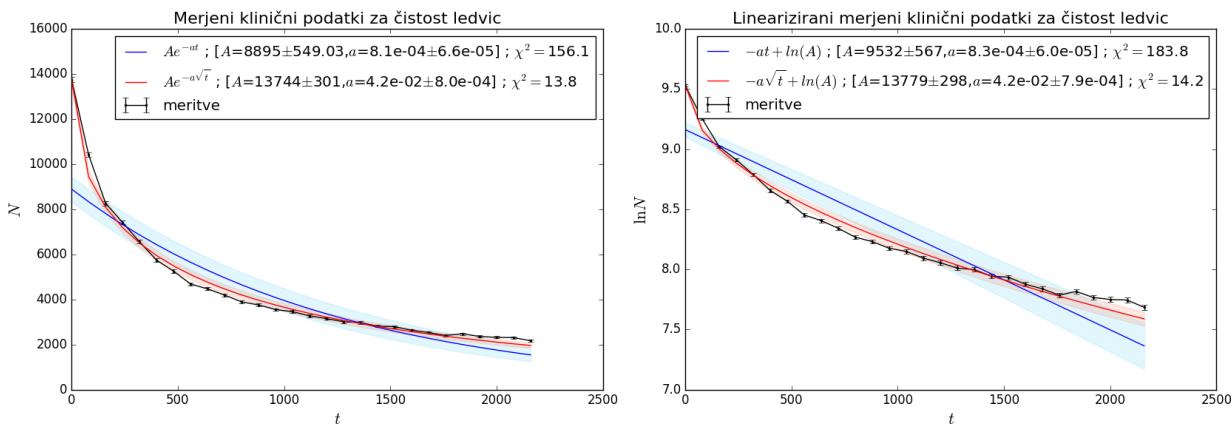
za navadno časovno odvisnost in v

$$\tilde{c} = \ln c(t) = -a\sqrt{t} + \ln(A) \quad (7.8)$$

z korensko odvisnost od časa. Takšni zvezi v upanju po boljšem rezultatu prilagodimo lineariziranim podatkom. Preden pričnemo s fitanjem pa je zaradi linearizacije potrebno transformirati še Poissonovo napako meritve sunkov $\sigma_c = \sqrt{N}$ kot

$$\sigma_{\tilde{c}} = \sigma_c \cdot \left| \frac{\partial \tilde{c}}{\partial c} \right| = \sigma_c \cdot \left| \frac{1}{c} \right|. \quad (7.9)$$

Rezultati prilagajanja zgornjih modelov k podatkom so prikazani na sliki 7.2. Pri tem je bila uporabljen Levenberg-Marquardtova metoda ob ustreznem upoštevanju napak meritve. Prilagajanje nelineariziranega modela s korensko odvisnostjo časa (7.6) je dalo najboljše vrednosti reduciranega χ^2 testa. Z linearizacijo 7.7 se fit ne izboljša. Linearizirana modela se tokrat odrežeta malenkost slabše od svoje nelinearizirane oblike.



Slika 7.2.: Graf direktnih meritev čistosti ledvic (levo) in graf istih podatkov v linearizirani obliki (desno); za oba primera so prikazane tudi napake meritiv σ_c in $\sigma_{\tilde{c}}$. Na podatke so z uporabo `curve_fit` in ob upoštevanju informacije o negotovostih meritiv prilagojene različne enorazdelčne funkcije. Prilagajanje nelineariziranega modela s korensko odvisnostjo časa (7.6) je glede na reducirani χ^2 test podalo najboljše rezultate. Tokrat linearizacija modelov ne izboljša fita.

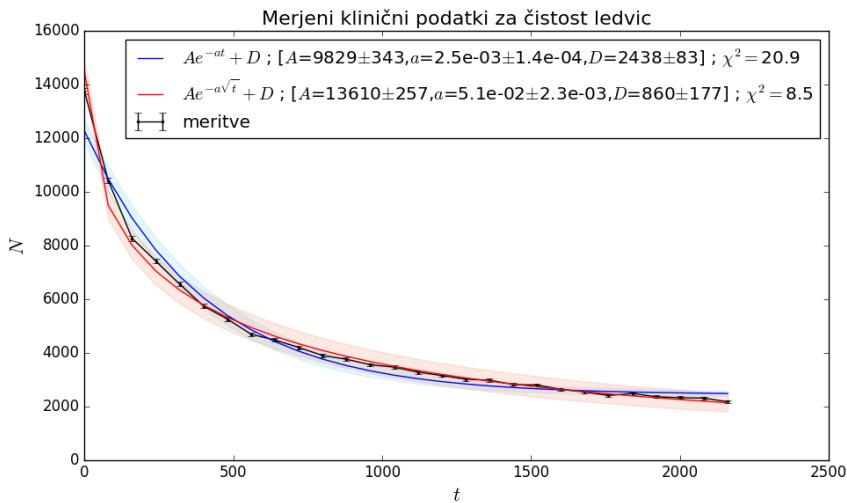
7.3.2. Enorazdelčni model z ozadjem

Oba prejšnja enorazdelčna modela lahko dopolnimo z konstantnim prispevkom sevanja ozadja v obliki

$$c(t) = Ae^{-at} + D, \quad (7.10)$$

$$c(t) = Ae^{-a\sqrt{t}} + D. \quad (7.11)$$

Rezultati prilagajanja k takima modeloma k podatkom so prikazani na sliki 7.3. Iz vrednosti reduciranega χ^2 testa vidimo, da konstanta izboljša fit v obeh primerih.



Slika 7.3.: Graf fitov direktnih meritov čistosti ledvic za primera enodelčnih modelov z upoštevanjem ozadja. Prilagajanje modela s korensko odvisnostjo časa (7.11) je glede na reducirani χ^2 test podalo boljše rezultate od modela (7.10).

7.3.3. Dvorazdelčni modeli

Podobno iz dvorazdelčnega modela (7.4) dobimo odvistnosti koncentracij, pri katerih lahko ravno tako lahko upoštevamo korensko odvisnost časa in prisotnost ozadja

$$c(t) = Ae^{-at} + Be^{-bt}, \quad (7.12)$$

$$c(t) = Ae^{-a\sqrt{t}} + Be^{-b\sqrt{t}}. \quad (7.13)$$

$$c(t) = Ae^{-at} + Be^{-bt} + D, \quad (7.14)$$

$$c(t) = Ae^{-a\sqrt{t}} + Be^{-b\sqrt{t}} + D, \quad (7.15)$$

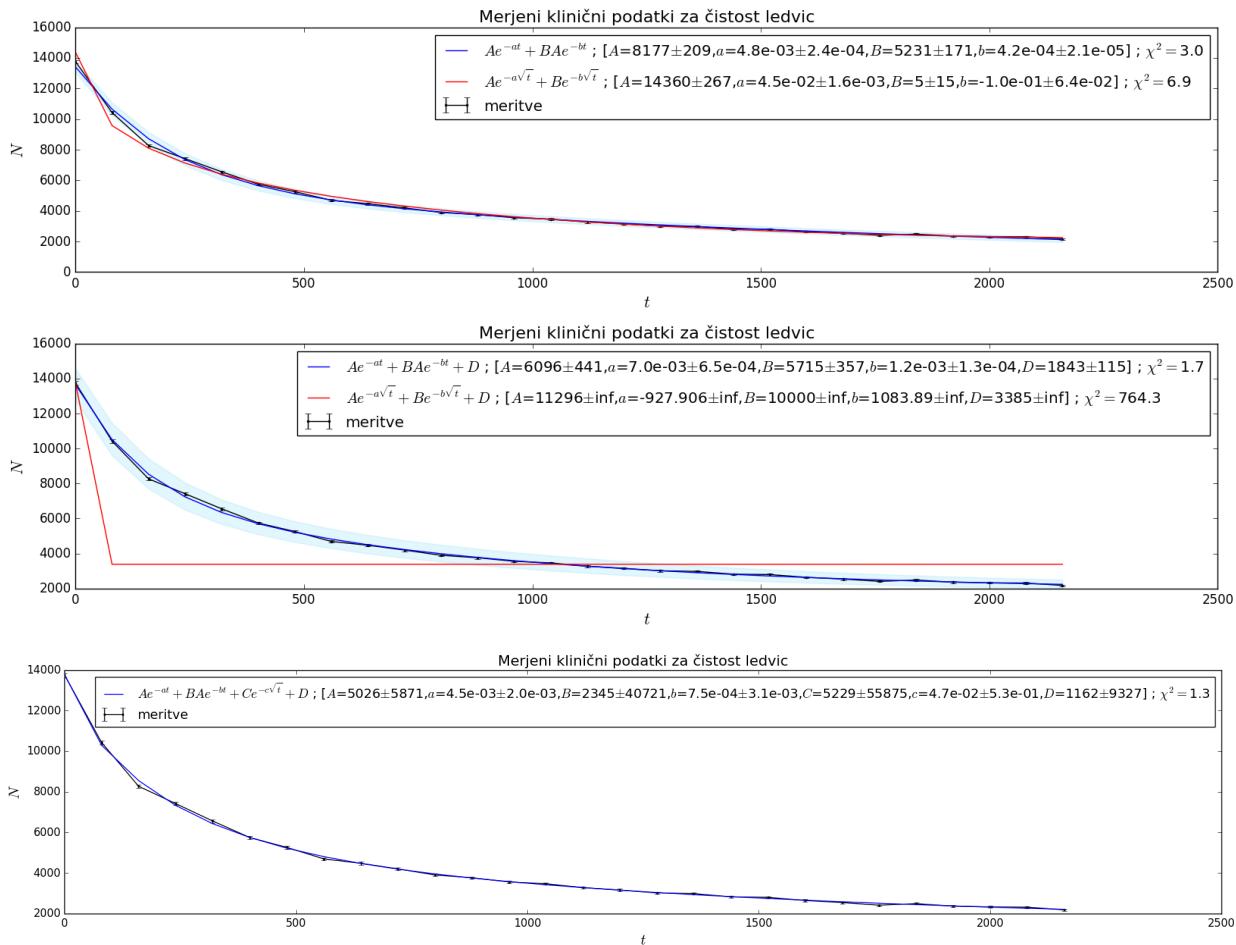
$$c(t) = Ae^{-at} + Be^{-bt} + Ce^{-c\sqrt{t}} + D. \quad (7.16)$$

Rezultati prilagajanja k tem modelom k podatkom so prikazani na sliki 7.4. Večinoma razširitev na dvorazdelčni model prinese izboljšavo v vrednosti reduciranega χ^2 testa, razen za primer (7.15), ko mi kljub ugibanju začetnih približkov parametrov ni uspelo doseči konvergenco fita. Videti je tudi, da po statističnem kriteriju najboljši fit modela (7.16) vrne parametre s precej negotovimi vrednostmi. Za najboljši model se tako izkaže (7.14), ki poda vrednosti parametrov $A = 6096 \pm 441$, $a = 0.007 \pm 0.0007$, $B = 5715 \pm 357$, $b = 0.0012 \pm 0.0001$ in $D = 1843 \pm 115$ ob reduciranem $\chi^2 = 1.7$. Ugotovili smo, da gre pri čiščenju ledvic najverjetneje za prehajanje snovi v krvi skozi ledvice ob dodatni difuziji v telo, detektor pa tekom eksperimenta izmeri tudi nezamerljiv delež sevalnega ozadja.

7.4. Korozija med kovino in elektrolitom

Pri zadnji nalogi si bomo primer meritve korozije pri različnih vrednostih električne napetosti U in električnega toka I . K merjenim podatkom prilagajamo modelski nastavek (7.2) z namenom, da bi določili parametre I_0 , U_a in U_c . Rezultati prilagajanja modela k podatkom so prikazani na sliki 7.5 pri čemer se najprej upošteval, da je meritve napetosti in toka eksaktne, nato pa nastavil negotovost merjenega toka na arbitarni konstantni vrednost $\sigma_I = 10^{-4}$ in $\sigma_I = 10^{-3}$.

Ker meritve ne gredo točno skozi izhodišče lahko poskusimo model prilagoditi še s premikom $U \rightarrow$

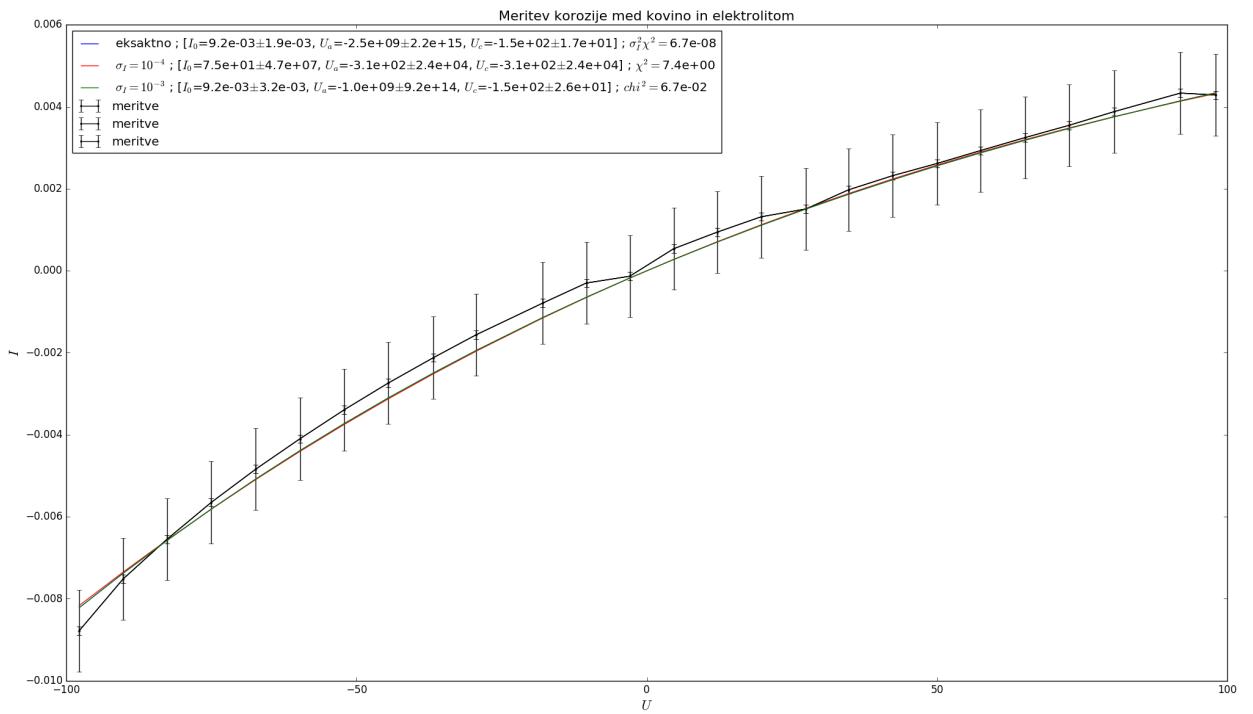


Slika 7.4.: Graf direktnih meritev čistosti ledvic (levo) in graf istih podatkov v linearizirani obliki (desno); za oba primera so prikazane tudi napake meritev σ_c in $\sigma_{\tilde{c}}$. Na podatke so z uporabo `curve_fit` in ob upoštevanju informacije o negotovostih meritev prilagojene različne enorazdelčne funkcije. Prilagajanje nelineariziranega modela s korensko odvisnostjo časa (7.6) je glede na reducirjan χ^2 test podalo najboljše rezultate. Najslabši je preprosti model (7.5), vendar se ta linearizacija izboljša, medtem ko korenskega modela linearizacija ne izboljša.

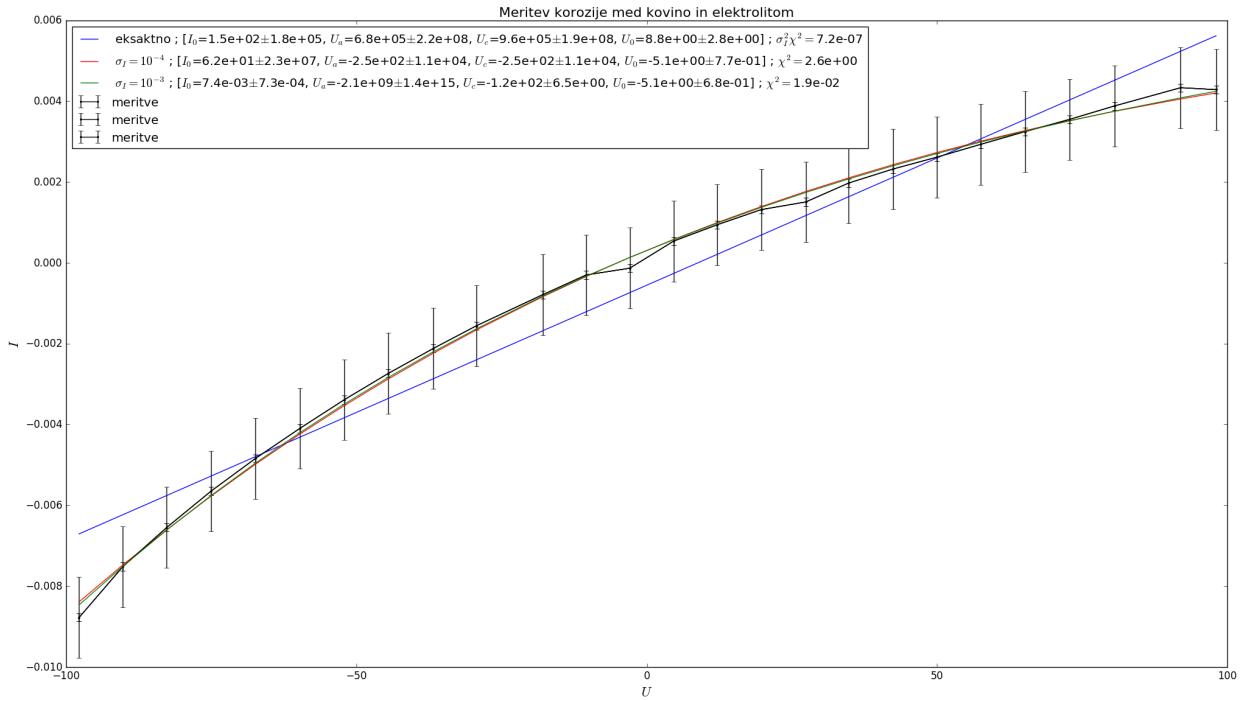
$(U - U_0)$. S tem (7.2) preide v obliko

$$I = I_0 \left[\exp \left(\frac{U - U_0}{U_a} \right) - \exp \left(-\frac{U - U_0}{U_c} \right) \right]. \quad (7.17)$$

Kako pristop z dodanim četrtem parametrom U_0 upošteva podatke lahko vidimo na sliki 7.6. Ponovno sem najprej upošteval, da je meritev napetosti in toka eksaktna, nato pa nastavil negotovost merjenega toka na arbitrarne konstantne vrednosti $\sigma_I = 10^{-4}$ in $\sigma_I = 10^{-3}$. Zaključim lahko, da premik kvalitativno precej izboljša fit, vendar tako določeni parametri še vedno niso uporabni, saj mi ni uspelo najti načina kako zmanjšati njihove negotovosti.



Slika 7.5.: Graf meritev korozije s tremi različnimi primeri fitanja za katere so prikazane tudi napake meritev $\sigma_I = 10^{-4}$ in $\sigma_I = 10^{-3}$. Na podatke je z uporabo `curve.fit` brez in ob upoštevanju arbitrarne negotovosti meritev prilagojena funkcija (7.2). Na oku kaže, da dodajanje negotovosti k fitanju nima posebnega vpliva, saj se vse tri fit krivulje skoraj prekrivajo. V vseh treh primerih pa so parametri kljub spodobnjemu poteku krivulje zelo slabo določeni.



Slika 7.6.: Graf meritev korozije s tremi različnimi primeri fitanja za katere so prikazane tudi napake meritev $\sigma_I = 10^{-4}$ in $\sigma_I = 10^{-3}$. Na podatke je z uporabo `curve.fit` brez in ob upoštevanju arbitrarne negotovosti meritev prilagojena funkcija (7.17). Videti je, da tokrat fit z dopuščanjem negotovosti lepše ujame robne točke iz začetka in konca meritve, čeprav pa so v vseh treh primerih parametri kljub izboljšanemu poteku krivulje ponovno zelo slabo določeni.

Literatura

- [1] Spletne strani z gradivom predmeta, ogled 20.10.2017,
<http://predmeti.fmf.uni-lj.si/modelska>
- [2] Spletne strani z gradivom predmeta, ogled 20.10.2017,
<http://predmeti.fmf.uni-lj.si/modelska/podatki/>

8. Generatorji slučajnih števil

Povzetek

Pri tej nalogi se spoznamo z generatorji (psevdo)naključnih/slučajnih števil in z njihovo uporabo pri generiranju različnih večdimenzionalnih porazdelitev. Nato s statističnimi testi, kot so na primer χ^2 test, test Kolmogorov-Smirnova ter z računanjem različnih normiranih momentov prevrimo njihovo učinkovitost ozziroma ocenimo naključnost. V zadnjem delu vaje s statističnim testom Kolmogorov-Smirnova pogledamo kumulativne porazdelitve časov oddaje nalog pri predmetu Modelska analiza.

8.1. Naloga

1. Med generatorji gaussovskih slučajnih števil najdemo Box-Mullerjev generator in konvolucijski generator (6 prištetih in 6 odštetih naključnih števil iz intervala $[0, 1)$). Preizkusi oba generatorja s testom Kolmogorov-Smirnova ter χ^2 -testom. Izvedi teste za različne velikosti vzorcev (100, 1000, 10000, ...) in primerjaj rezultate. Za oba primera lahko gledaš tudi porazdelitev statistike. Izmeri tudi časovno učinkovitost algoritmov.[1]
2. Pri dipolnem sevanju porazdelitev fotonov po prostorskem kotu ni enakomerna, pač pa je sorazmerna $\sin^2 \vartheta$. Sestavi generator naključnih smeri v prostoru in generator dipolnega sevanja ter ju preizkusi. Namesto razdelitve v predalčke lahko spreminjaš tudi nekaj osnovnih momentov, npr. $\langle \cos \vartheta \rangle$, $\langle \cos^2 \vartheta \rangle$ oziroma splošno $\langle Y_{lm} \rangle$ za najnižje krogelne funkcije. S kakšno potenco padajo variacije v momentih, ko povečujemo velikost vzorca?[1]
3. Datoteke `mod_tmxx_yyy.dat`[2] vsebujejo čase oddaje nalog iz Modelske analize (xx–letnik 10,11,13,14; yyy – številka naloge) v formatu `d : hh : mm`, merjeno od četrtnika ob polnoči. Minus v podatkih pripada številki dneva – $-1 : 20 : 30$ pomeni sredo ob pol devetih zvečer. S testom Kolmogorov-Smirnova preveri, če so kumulativne porazdelitve za posamezne naloge med seboj statistično enake in če se letniki statistično razlikujejo med seboj. S kolikšno verjetnostjo lahko določimo iz katerega leta prihajajo podatki za izbrano nalogo? Če je potrebno, upoštevaj, da se pogoji za nekatere naloge razlikujejo (novoletni prazniki, zadnja naloga semestra,...) in jih izloči iz statistike.[1]

Komentar k reševanju

Za generatorje slučajnih števil lahko uporabiš in med seboj primerjaš vgrajeni generator izbranega programskega jezika, generatorje iz *Numerical Recipes* in druge, bolj sofisticirane generatorje iz numeričnih knjižnic. Preizkusil lahko tudi druge izvore (psevdo)naključnih števil (`/dev/(u)random`, kompresija, enkripcija podatkov, statični šum, ipd.). Kot primer slabega generatorja lahko izbereš t.i. "kalkulatorski" generator: $x_n + 1 = \{(x_n + \pi)^5\}$, kjer {} pomeni mantiso (lomljeni del) števila.

Ker je analiza podatkov izvedena v Python-u s pomočjo vgrajenih funkcij knjižnice `stats`, je večino časa v uporabi v knjižnici `numpy` vgrajeni generator naključnih števil iz modula `random`. Poleg tega so preizkušeni tudi generatorji iz okolja ROOT, katerih rezultate je bilo za namen vzporedne analize potrebno izvoziti v `txt` datoteko.

8.2. Generatorji gaussovskih slučajnih števil

Najprej se posvetimo analizi generatorjev (psevdo)naključnih števil, ki vrnejo naključna števila enakomerno razporejena po intervalu $[0, 1]$. Uporabimo naslednje generatorje:

1. `random.rand()` in `random.randrange()` generatorja vsebovana v Python-u. Oba sta implementirana v C-ju in temeljita na Mersenne Twister generatorju s periodo $2^{19937} - 1$. Prvi vrača enakomerno in drugi normalno porazdeljena števila.
2. `TRandom1` v ROOT-u vsebovan generator, ki je malenkost počasnejši in sloni na RANLUX algoritmu s periodo 10^{171} .[3]
3. `TRandom2` hitrejši v ROOT-u vsebovan generator Tausworthe generator of L'Ecuyer, s krajšo periodo 10^{26} .[3]
4. `TRandom3` v ROOT-u vsebovan generator, ki ravno tako temelji na Mersenne Twister algoritmu, s periodo okrog 10^{6000} .[3]
5. Poleg teh generatorjev sem za primerjavo vključil še statistiko izžrebanih naključnih števil iz nacionalne igre na srečo Loto.[4]

Za zacetek velja preiskusiti časovno učinkovitost algoritmov. V Python-u se za ta namen poslužim funkcije `cProfile.run()` v okolju ROOT pa klicem `TStopwatch()`. Pri generiranju $N = 10^6$ dogodkov sta najhitrejša `TRandom3` in `TRandom3` generatorja ki za to porabita 0.24 sekunde in 0.25 sekunde, sledita jima `random.rand()` in `TRandom1` z 0.028 in 0.033 sekunde, najpočasnejši pa je generator normalne porazdelitve `random.randrange()`, ki potrebuje 0.096 sekunde.

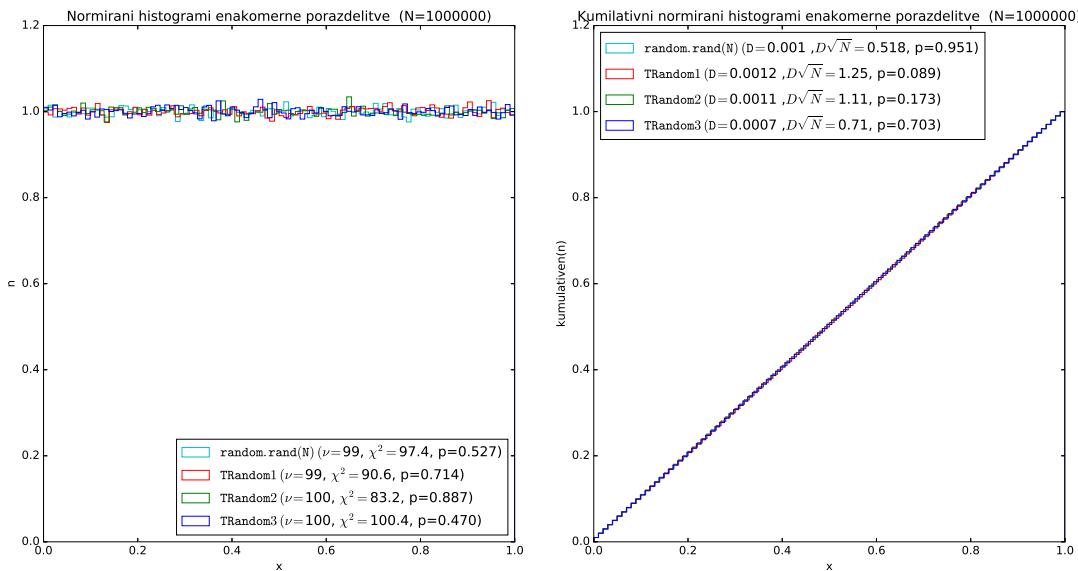
Tako generirana naključna števila v nadaljevanju preiskusimo s testoma χ^2 in Kolmogorov-Smirnov (K-S)¹ ter jih tako lahko primerjamo med seboj. Omenjena testa sta vgrajena v knjižnici `stats` okolja Python. Prvega klicem kot `chisquare()` drugi pa obstaja v dveh inačicah `kstest()` in `ks_2samp()`; preiskusiti gre obe različici K-S testa, saj prva vnos primerja z podano funkcijo, slednja pa dva podana vnosa med seboj.

Na sliki 8.1 sta histograma z rezultati obeh testov za vse štiri numericne generatorje števil enakomerne porazdelitve po intervalu $[0, 1]$. Pričakovani rezultat za χ^2 enakomerne naključne porazdelitve je enak vrednosti ν , ki jo določa število prostostnih stopenj oziroma število predalčkov histograma. Pri tem si želimo verjetnosti čim bližje vrednosti $p = 0.5$. Premajhna vrednost (npr. manj kot 5%) pomeni, da lahko generator zavrzemo z majhno stopnjo tveganja, saj števila niso porazdeljena dovolj enakomerno. Podobno pa prevelika vrednost (npr. več kot 95%) pomeni, da je porazdelitev preveč enakomerna, saj so odstopanja manjša od pričakovane statistične napake. Oboje namreč kaže, da števila niso naključna. V primeru K-S testa temu ustreza kritična vrednost $D\sqrt{N} = 0.8276$.

Iz podatkov testoma χ^2 lahko zaključimo, da je izmed preizkušanih generatorjev najboljši `TRandom3`. Izkaže se ko moramo hitro generirati veliko naključnih števil z dobre kvilitete. Sledi mu tudi malce počasnejši `random.rand()`. To ugotovitev potrdi tudi K-S test. V nadaljevanju bosta zgolj zaradi udobnosti uporabljeni algoritmi, ki sta že vsebovana v Python-u.

Kot zahteva naloga nas zanimajo generatorji, ki podajajo števila v normalni oziroma gaussovski porazdelitvi. Med generatorji gaussovskih slučajnih števil najdemo v navodilu naloge omenjen Box-Mullerjev generator, katerega porazdelitvi generiramo iz enakomernih porazdelitev `a1` in `a2` preko funkcije `randn()` ter imata oznaki `z1` in `z2`. Podobno iz enakomernih porazdelitev `a1, ..., a12` s konvolucijskim generatorjem generiramo porazdelitev in ji dam oznako `K1`. Oba generatorja torej gaussovsko porazdelitev generirata iz števil, ki so generirana po enakomerni porazdelitvi v intervalu $[0, 1]$. Tako generirana gaussovská števila ponovo testiramo z omenjenima statističnima testoma, lahko pa ju preko njiju primerjam tudi s porazdelitvijo števil iz vgrajenega generatorja `randn()`, ki jo označimo z `a` oziroma `norm`.

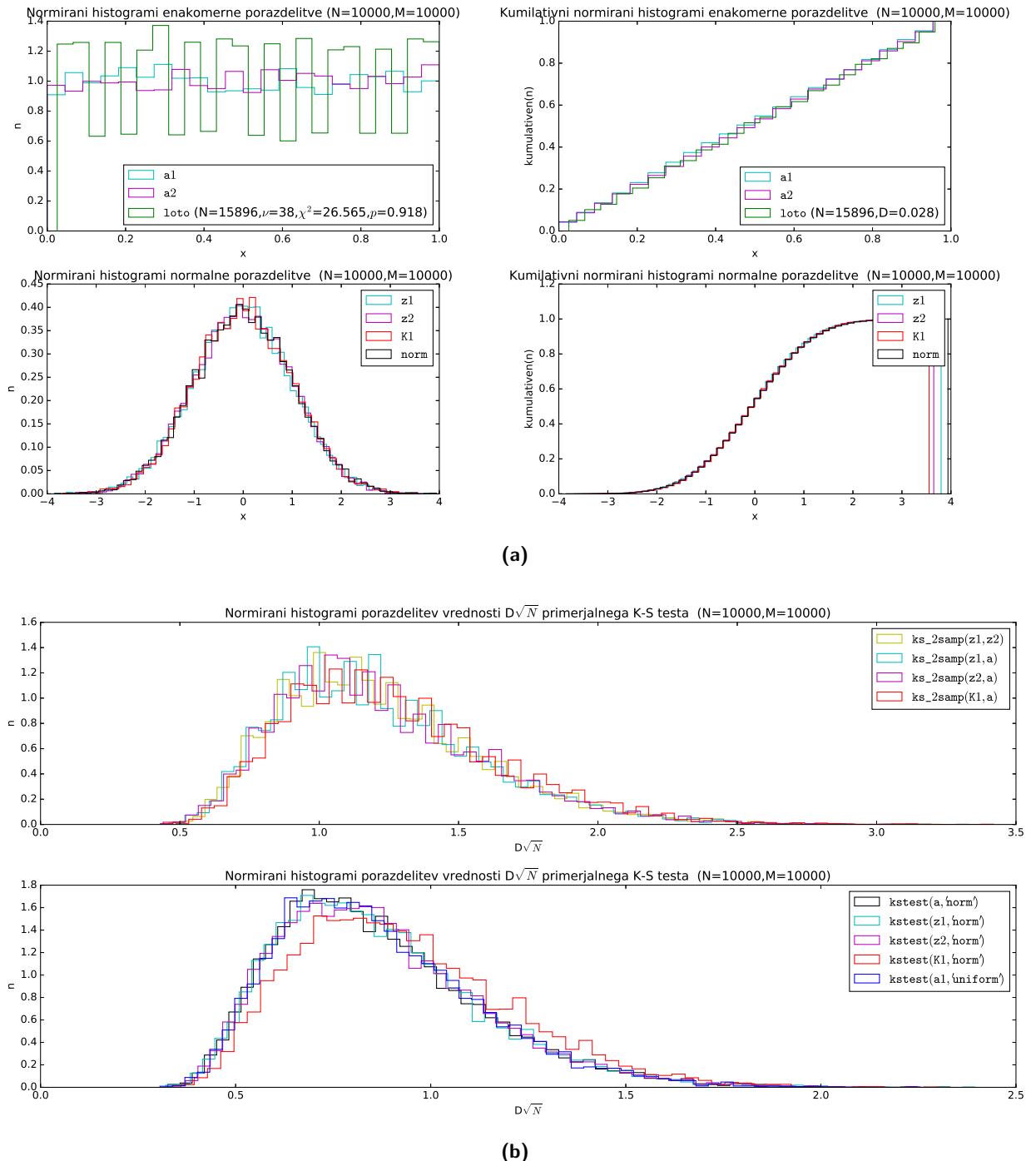
¹Oba omenjena testa sta opisana v literaturi [[2]].



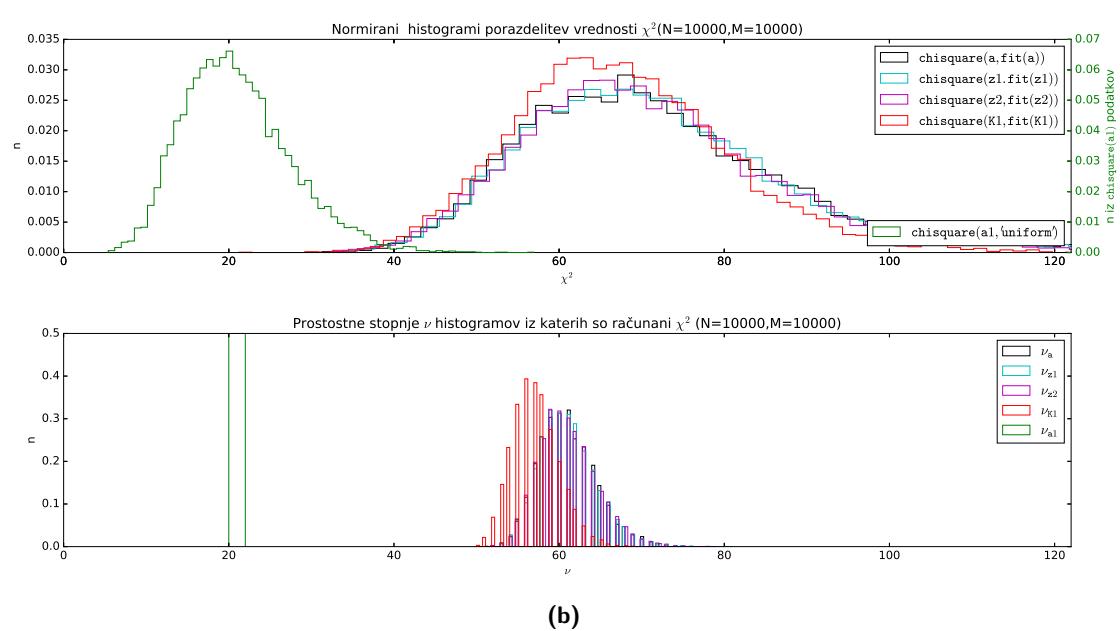
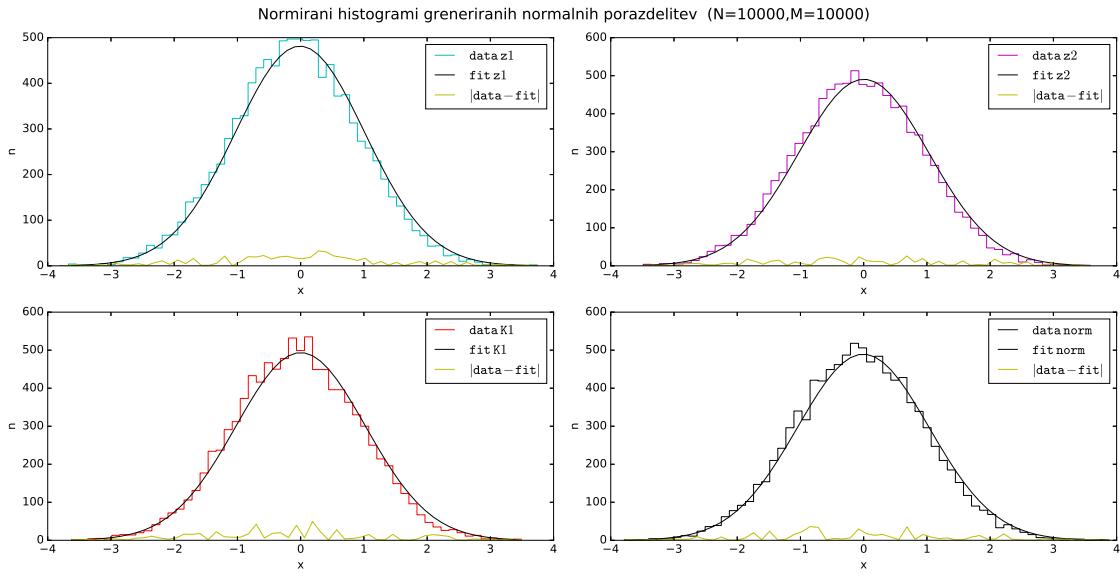
Slika 8.1.: Na levih histogramih normirane porazdelite generiranih naključnih števil in kumulativnimi histogrami na desni med samimi generatorji ne vidimo večjih odstopanj. Razliko med njimi pokažeta testa χ^2 in K-S ki favorizirata generator **TRandom3**.

Testi so izvedeni na vzorcih z velikostmi $N=100, 1000, 10.000$ in ponovljeni 10.000-krat. Tako dobimo porazdelitev same statistike rezultatov obeh testov. Histogrami in rezultati statistike vzorcev veilkosti $N=100, 1000$ se nahajajo v dodatku naloge, histogrami generiranih števil za vzorec velikosti $N=10.000$ pa so prikazani na histogramih (a) slike 8.2. Tu je normiranim histogramom enakomernih porazdelitev a1 in a2 v intervalu $[0, 1]$, dodana še normirana porazdelitev dogodkov žrebanih številk igre loto vse od leta 1991 dalje. Iz nakomernih porazdelitev so generirane gaussovske z_1, z_2 in K_1 prikazane na histohramih pod njimi. dodana pa jim je še porazdelitev **norm** vgrajenega generatorja. Na histogramih (b) slike 8.2 sta prikazani porazdelitvi statistike K-S testov dobljene s uporabo funkcij **kstest()** in **ks_2samp()**; še vedno želimo imeti optimalno vrednost $D\sqrt{N} = 0.8276$. Izkaže se, da se za naš namen bolje obnese metoda K-S primerjave z uporabo **kstest()** - to je predvsem razvidno iz grafov manjših vzorcev z $N=100$ in 1000 , ki se nahajajo v dodatku na slikah A.1 in A.2.

Rezultate testa χ^2 je bilo težje dobiti. Za vsako porazdelitev sem najprej poiskal ustrezno krivuljo, ki je predstavljal funkcijo porazdelitve verjetnosti analiziranega vzorca. Preko nje sem nato izračunal željene vrednosti stolpcov histograma za primer gaussovke in jih z uporabe funkcije **chisquare()** primerjal z vzorcem. Porazdelitve primera s fiti ter porazdelitvami so prikazane na (a) histograih slike 8.3. Porazdelitev statistike χ^2 testa in porazdelitve iz avtomatskega predalčkanja izhajajočih prostostnih stopenj ν so na (b) histograih slike 8.3. Izkaže se, da so generatorji med seboj dokaj primerljivi Iz slik B.1 in B.2 v dodatku, na katerih so histogrami manjših vzorcev z $N=100$ in 1000 , ugotovimo, da je z večanjem vzorca porazdelitev statistike χ^2 testa vedno bolj gaussovsko.



Slika 8.2.: Na histogramih (a) je normirana enakomerna porazdelitvama `a1` in `a2` dodana še normirana porazdelitev dogodkov žrebanih številk igre loto vse od leta 1991 dalje, ki že na oko izstopa. Naključnost igre bi se omogoče dalo še izboljšati. Pod njimi so iz enakomernih porazdelitev generirane gaussovskie `z1`, `z2` in `K1`, dodana pa jim je še porazdelitev `norm` vgrajenega generatorja. Na histogramih (b) sta prikazani porazdelitvi statistike K-S testov dobljene z uporabo funkcij `kstest()` in `ks_2samp()`; še vedno želimo imeti optimalno vrednost $D\sqrt{N} = 0.8276$ opazen je pa k večjim vrednostim razpotegnjjen rep. Izkaže se, da daje uporabo `kstest()` za naš namen boljše rezultate.



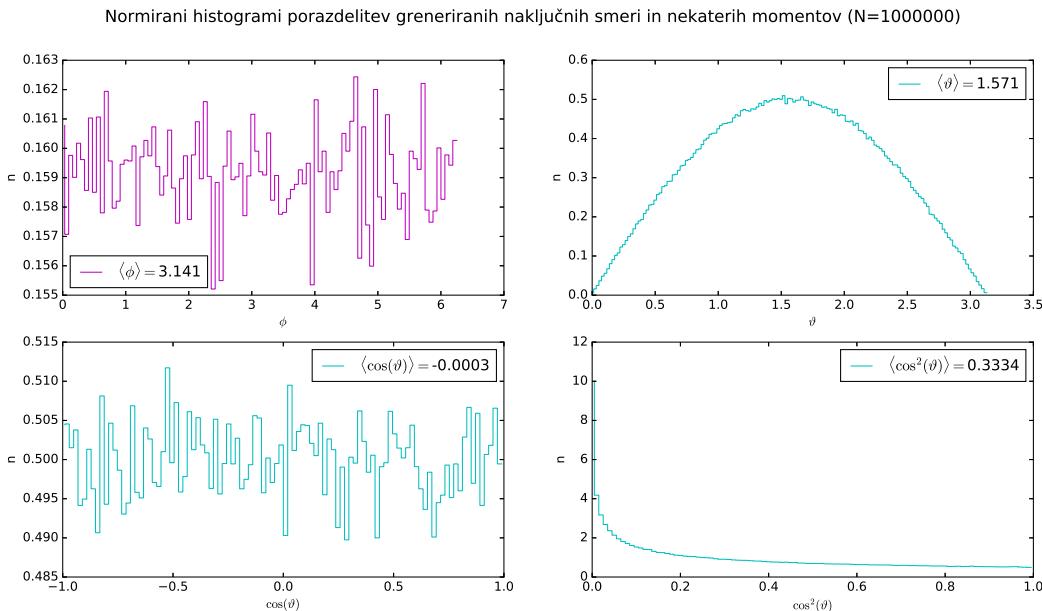
Slika 8.3.: Na (a) histogramih so prikazane porazdelitve primera s porazdelitvami ter pripadajočimi fiti. Porazdelitve statistike χ^2 testa so na zgornjih (b) histogramih slike, pod njimi pa porazdelitve iz avtomatskega predalčkanja izhajajočih prostostnih stopenj ν , katerih vrednosti se naj bi ujemale z njimi. Porazdelitev statistik je prilično gausovska.

8.3. Generator naključnih smeri v prostoru

Z naključnimi števili lahko generiramo tudi slučajne smeri v prostoru. Zanimata nas torej kota ϑ in ϕ . Algoritem je enostaven, saj vemo, da mora biti porazdelitev verjetnosti po spremenljivkah φ in $\cos \vartheta$ enakomerna. Če sta a_1 in a_2 dve naključni števili iz intervala $[0, 1)$, lahko prostorski kot parametriziramo kot

$$\phi = 2\pi a_1 \quad \text{in} \quad \vartheta = \arccos(2a_2 - 1) \quad (8.1)$$

Zdaj si lahko zopet pogledamo enakomernost porazdelitve s histogramom oziroma predalčkanjem vrednosti. samo. Enakomernost porazdelitve lahko preverimo najprej z nekaj osnovnimi momenti in sicer z $\langle \phi \rangle, \langle \vartheta \rangle, \langle \cos \vartheta \rangle, \langle \cos^2 \vartheta \rangle$. Histogrami vzorcev velikosti $N = 10^6$ s pripisanimi vrednostmi momentov so prikazani na sliki 8.4. Enakomernost pa lahko preverimo še z sferičnimi funkcijami Y_m^l prikazanimi na sliki 8.5. Zanimiva je tudi odvisnos kvalitete generirane porazdelitve od velkosti vzorca, ki si jo lahko ogledamo na grafih slike 8.6.



Slika 8.4.: Idealno si želimo vrednosti $\langle \phi \rangle = 3.141593$, $\langle \vartheta \rangle = 1.570796$, $\langle \cos \vartheta \rangle = 0$ in $\langle \cos^2 \vartheta \rangle = 0.333\dots$. Iz slike vidimo, da se jim z našim generatorjem zadvoljivo približamo.

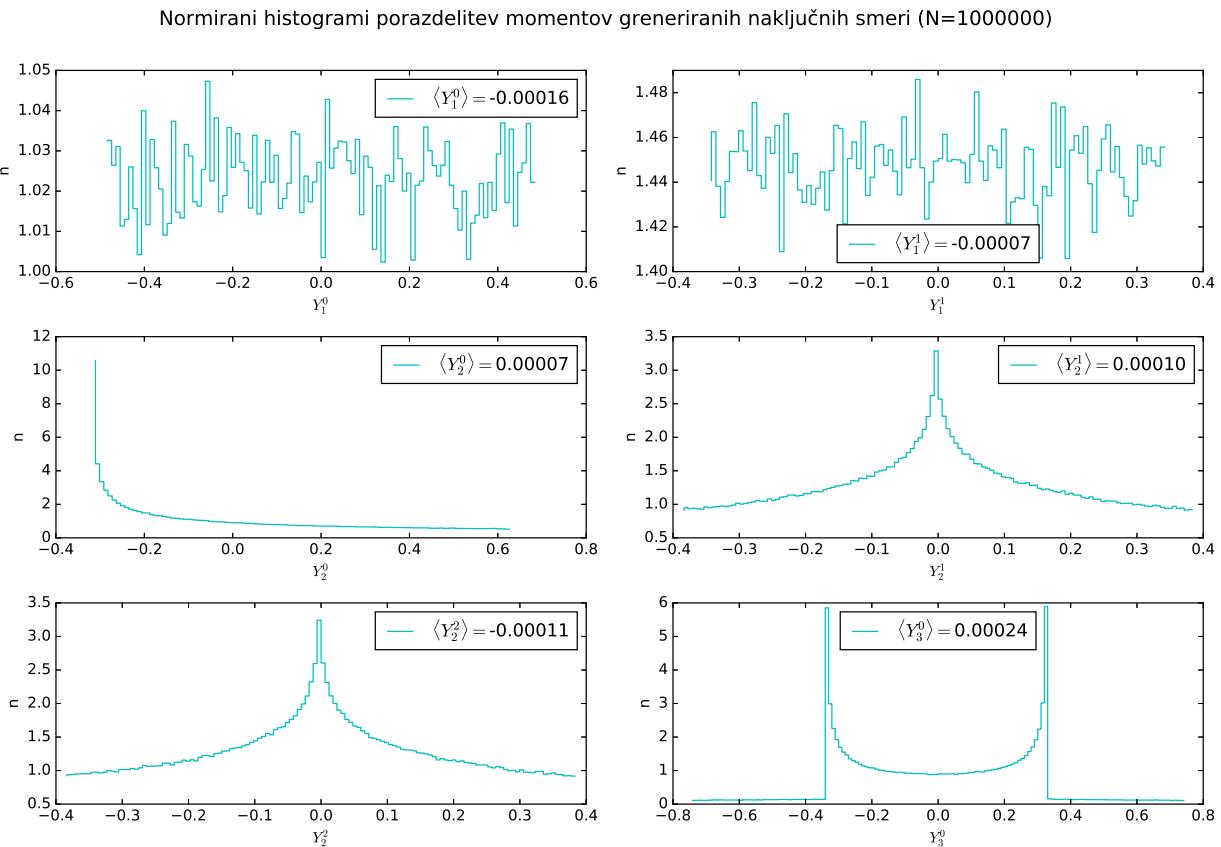
8.4. Generator dipolnega sevanja fotonov

Sevanje dipolne antene nam da odlet fotonov iz dipola, ki ni enakomeren po vsem prostorskem kotu, saj je več sevanj v ravnini, ki jo dipol prebada kot pa na osi dipola. Pri dipolnem sevanju je porazdelitev verjetnosti po kotu ϑ torej bolj zapletena. Kot ϕ ostane isti kot v prejšnjem primeru

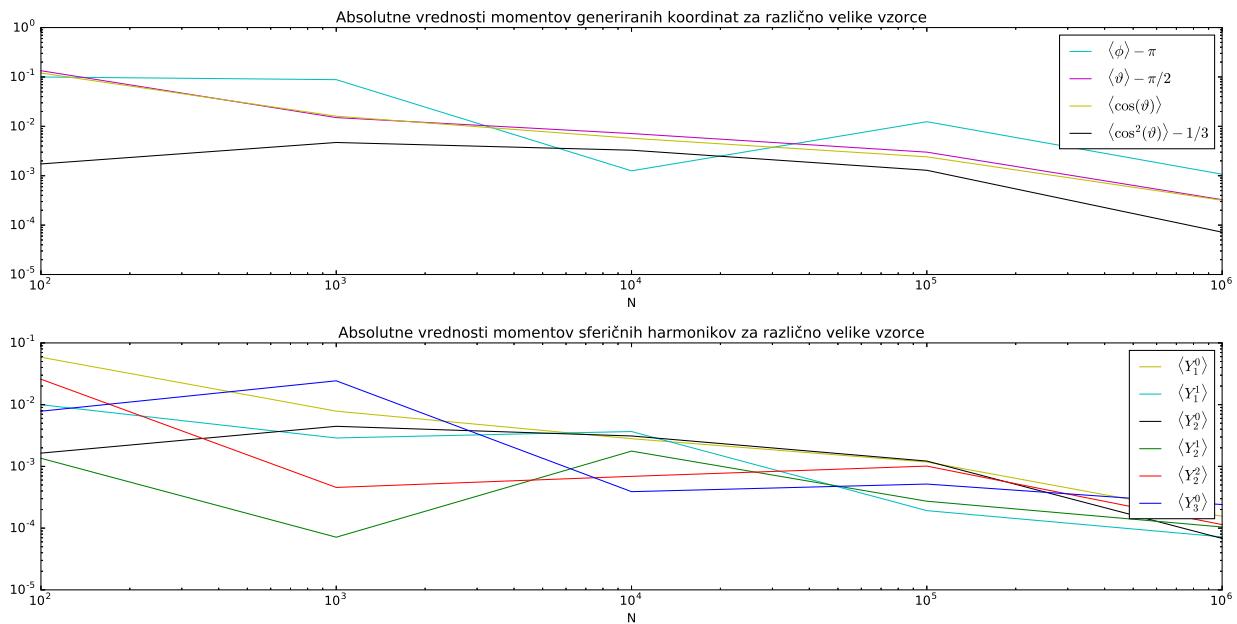
$$\phi = 2\pi a_1, \quad (8.2)$$

kota ϑ ne moremo izraziti kar direktno, vemo pa da velja

$$P = C \sin^3 \vartheta = C(1 - \cos^2 \vartheta)(\cos \vartheta) = C \left(\cos \vartheta - \frac{\cos \vartheta^3}{3} \right) \quad (8.3)$$



Slika 8.5.: Idealno si želimo vrednosti 0 za vse momente. Naš generator naključno porostorsko porazdelitev izvede zadovljivo.

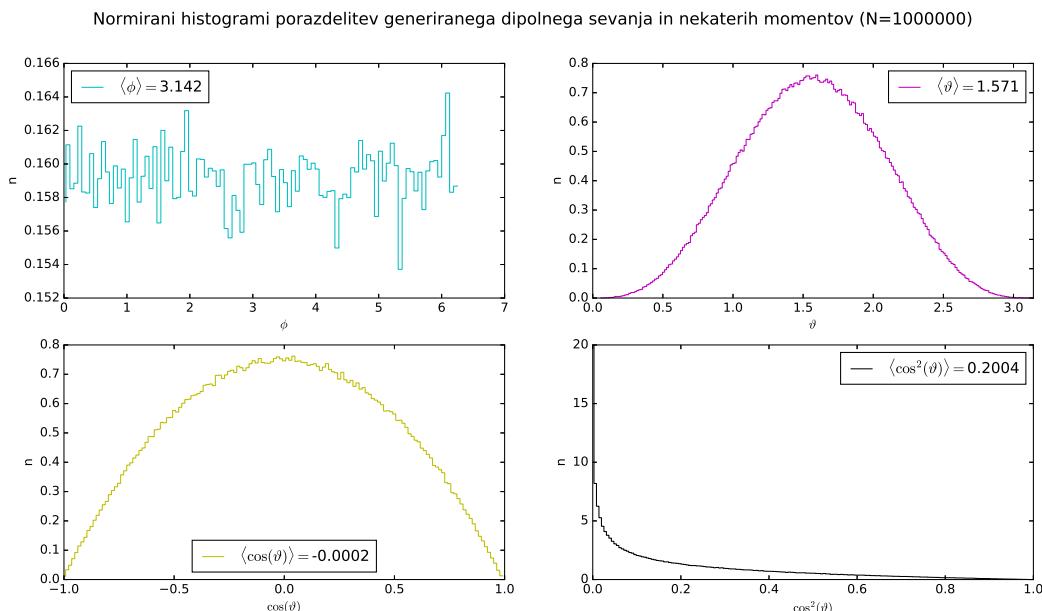


Slika 8.6.: Z večanjem velikosti N generiranega vzorca se počasi približujemo idealnim vrednostim momentov enakomerne prostorske porazdelitve.

če upostevamo, da je a_2 enakomerno razporejen, dobimo enačbo za ϑ .

$$\cos^3 \vartheta - 3 \cos \vartheta + 4a_2 - 2 = 0 \quad (8.4)$$

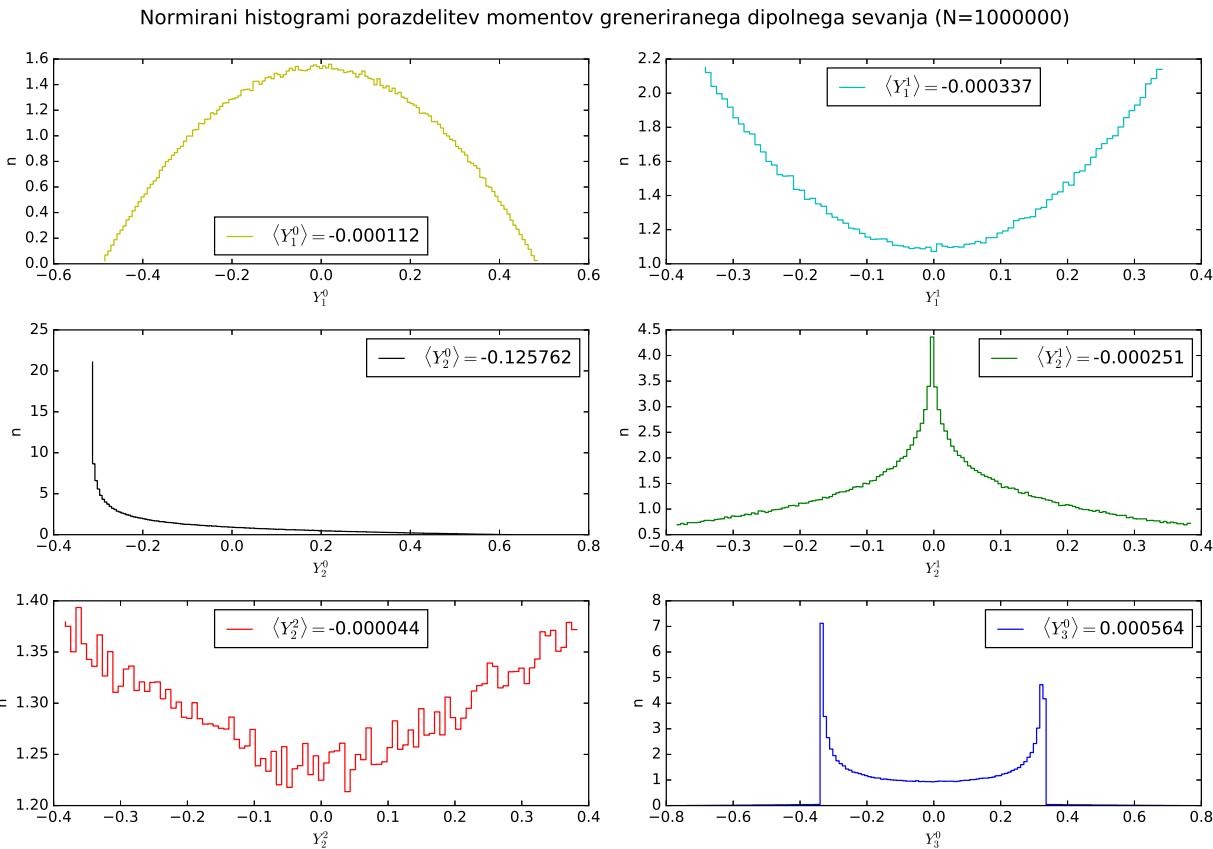
Da izrazimo $\cos \vartheta$ s pomočjo r_2 bi morali rešiti polinom tretje stopnje. Enačba mora vedno imeti eno rešitev za $\cos \vartheta$ na intervalu $[-1, 1]$, ki je žal analitično ne moremo dobiti, a lahko do nje vseeno pridemo z numerično metodo, npr. uporabo Python-ove knjižnice `scipy` z modulom `optimize`, ki vsebuje funkcijo `fsolve()`. Ko tako dobimo porazdelitve po kotih, jih lahko spet preverimo preko istih osnovnih momentov, kot v prejšnji nalogi, le njihove pričakovane vrednosti so nekoliko drugačne. Rezultati izračunov z vzorcem $N = 10^6$ števil so prikazani na histogramih slik 8.7 in 8.8. Odvisnost kvalitete generirane porazdelitve od velkosti vzorca, si lahko ogledamo na grafih slike 8.9.



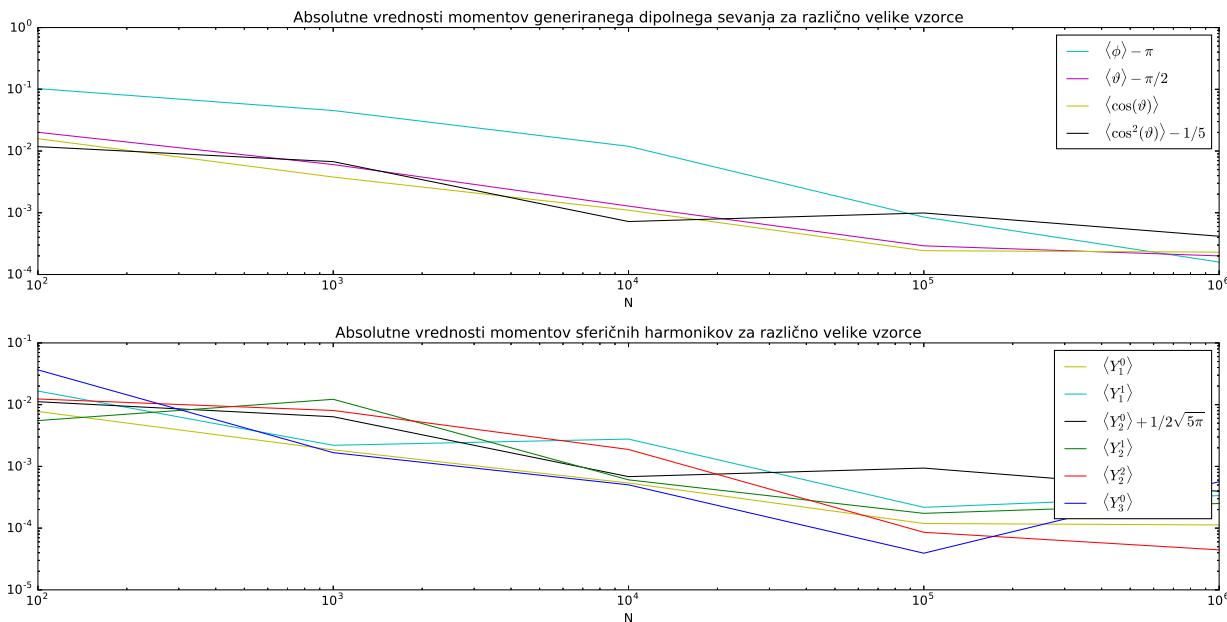
Slika 8.7.: Opazimo, da je na polih izhaja manj fotonov kot proti ekvatorju. To je naš dipolni sevalec. Idealno si želimo vrednosti $\langle \phi \rangle = 3.141593$, $\langle \theta \rangle = 1.570796$, $\langle \cos \vartheta \rangle = 0$ in $\langle \cos^2 \vartheta \rangle = 0.2$. Iz slike vidimo, da se jim z našim generatorjem in vzorcem izbrane velikosti N zopet zadvoljivo približamo.

8.4.1. Porazdelitev časov oddaj nalog iz Modelske analize

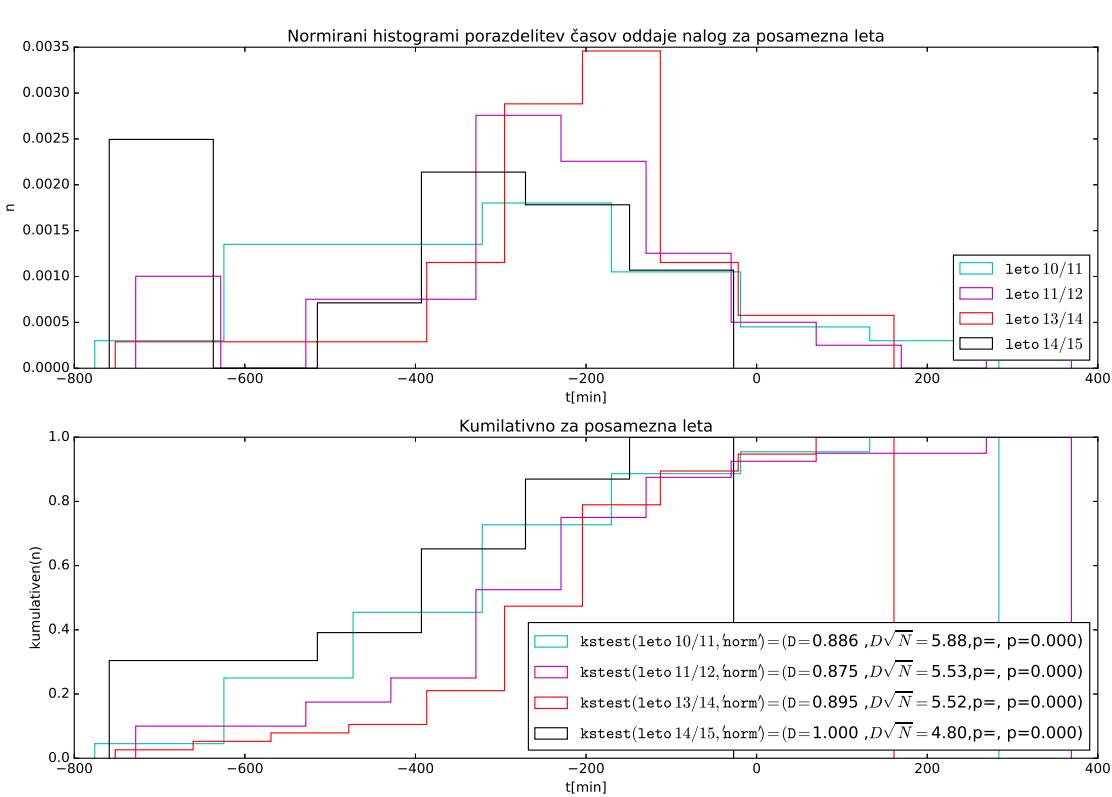
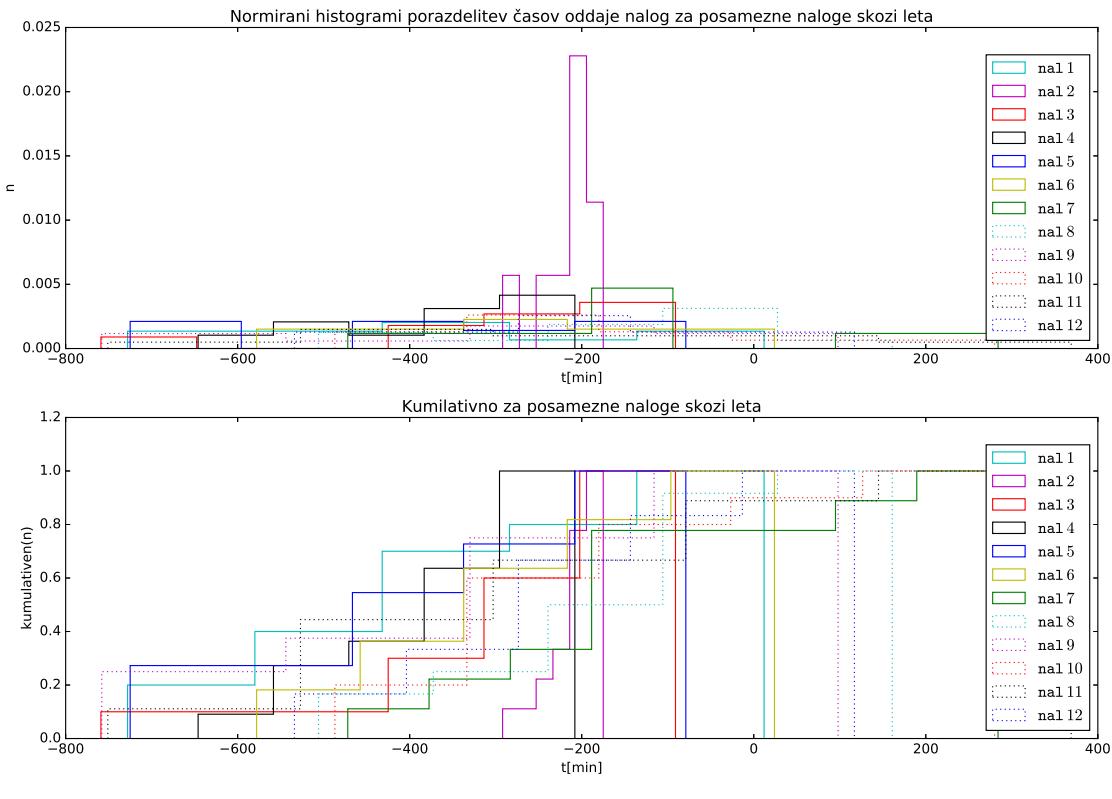
Pri zadnjem delu naloge preverimo kumulativne porazdelitve časov oddajanja poročil opravljenih vaj pri predmetu Modelska analiza. Iz danih datotek preberemo vrednosti časov oddaj, jih pretvorimo v minute in nazadnje ločimo po letnicah oziroma naloga. V tem koraku je bilo potrebno izločiti podatke, ki pripadajo zadnji nalogi semestra; t.j. trinajsti nalogi, za katero ni dovolj podatkov v vsakem letu. Iz analize so naknadno izvzeti še osameli dogodki, ki so se zgodili 800 minut pred in po roku oddaje. Želimo se prepričati, če so trendi kumulativnih porazdelitev časov oddaj nalog čez leto razlčni. Histogrami časov oddaje in njihove kumulativne porazdelitve so prikazane na sliki 8.10. Pokaže se, da so podatki porazdeljeni nekako naključno, vendar z subtilnim gaussovskim trendom, ki ima vrh 200 minut pred rokom oddaje. Vsekakor je to čas, ki ga sam z oddajo te vaje ne bom ulovil!



Slika 8.8.: Idealno si želimo vrednosti 0 za vse momente razen za $\langle Y_2^0 \rangle = 0.12616$. Naš generator naključno porostorsko porazdelitev izvede zadvoljivo.

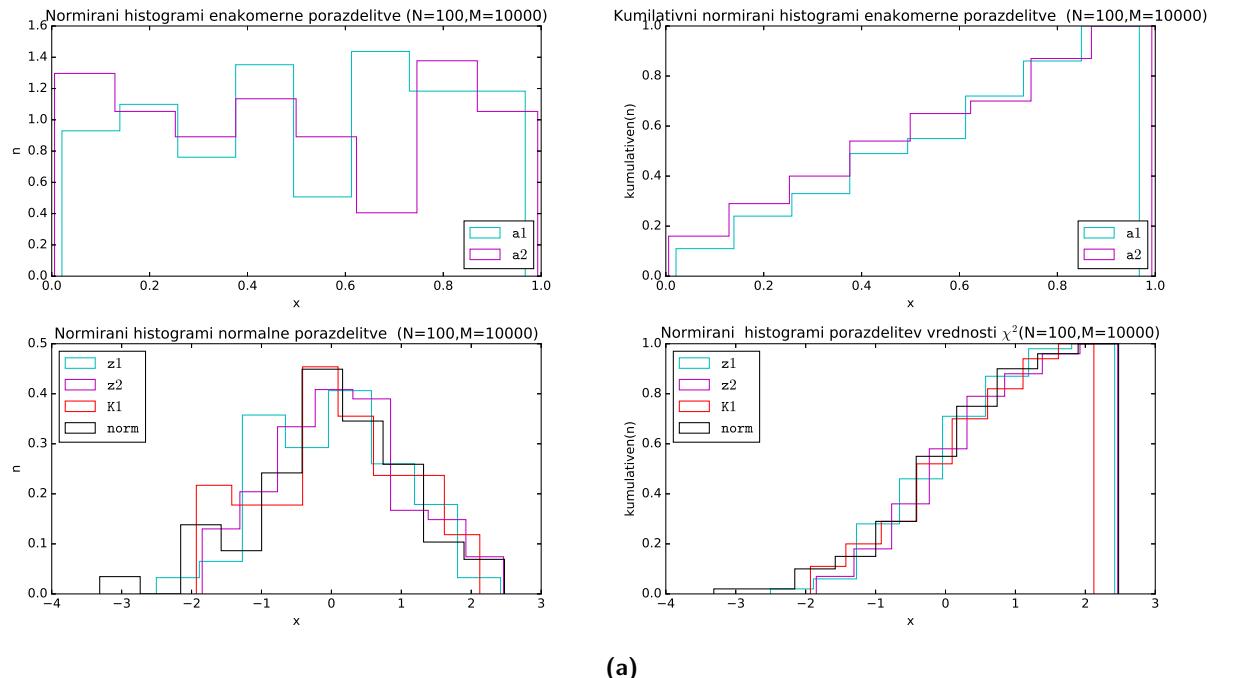


Slika 8.9.: Z naraščanjem velikosti N vzorca se počasi približujemo idealnim vrednostim momentov enakomerne prostorske porazdelitve.

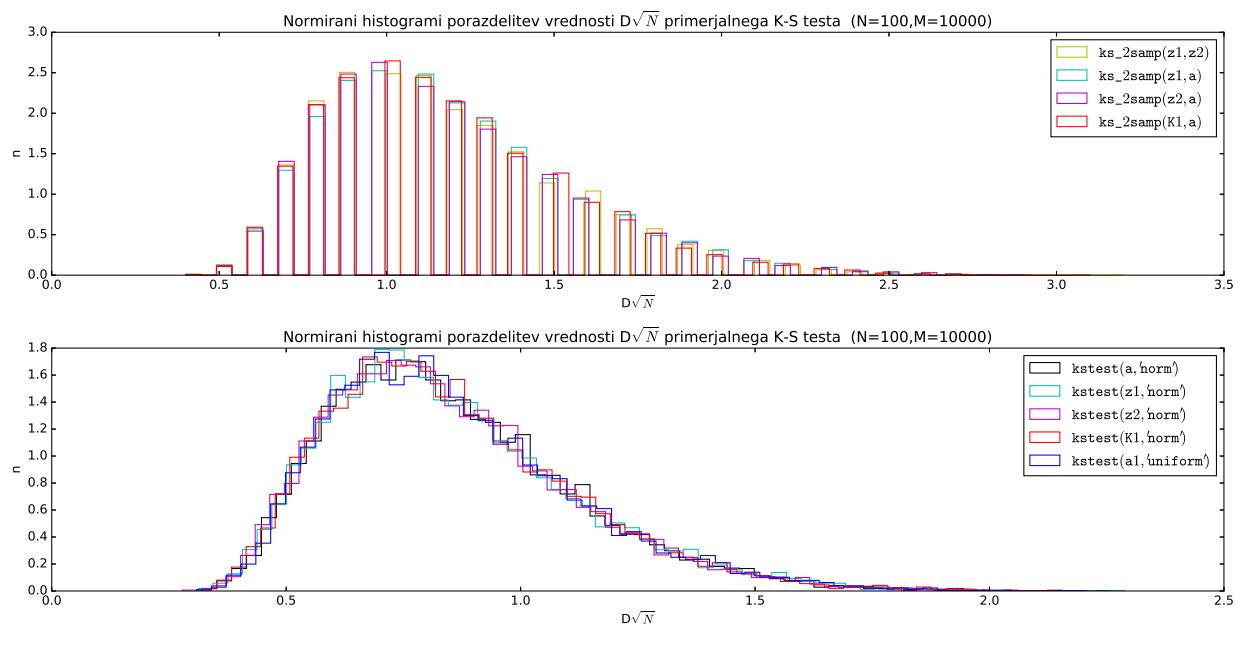


Slika 8.10.: Podatki histogramov časov so ločeni po nalogah (a) oziroma letnikih (b). Iz histogramov so izvzeti osameli dogodki, ki so se zgodili 800 minut pred in po roku oddaje. Podatki so porazdeljeni nekako naključno, vendar z subtilnim gaussovskim trendom, ki ima vrh 200 minut pred rokom oddaje. Podatki K-S testa veljajo za primerjavo dobljenih porazdelitev z normalno porazdelitvijo.

A. Kolmogorov-Smirnov testi



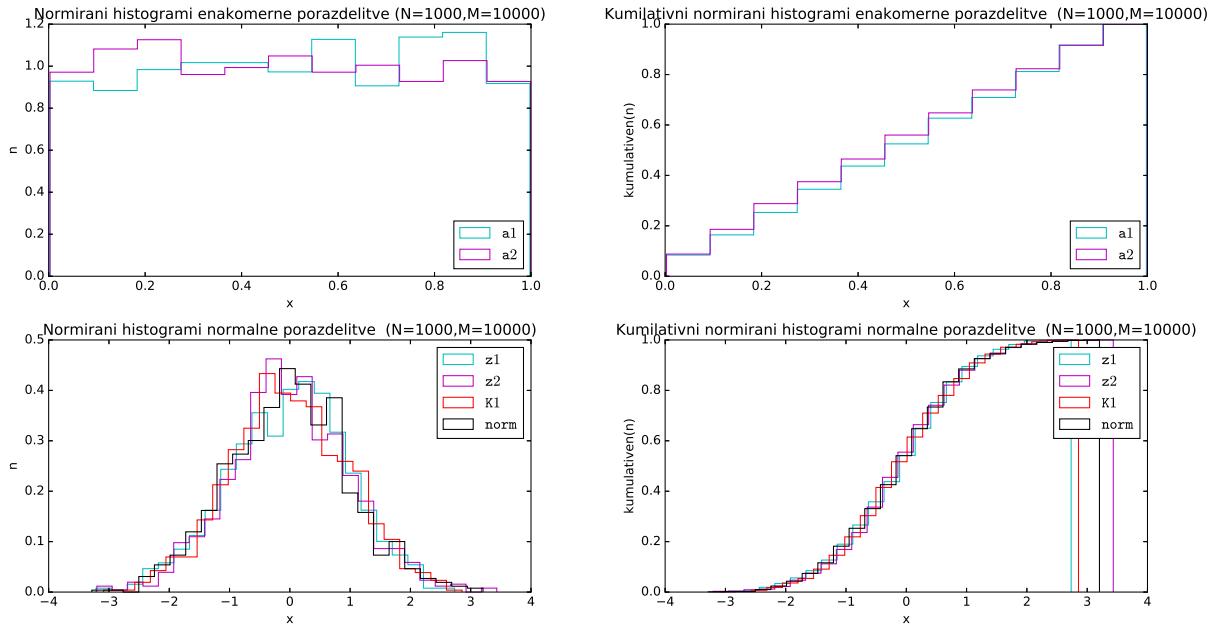
(a)



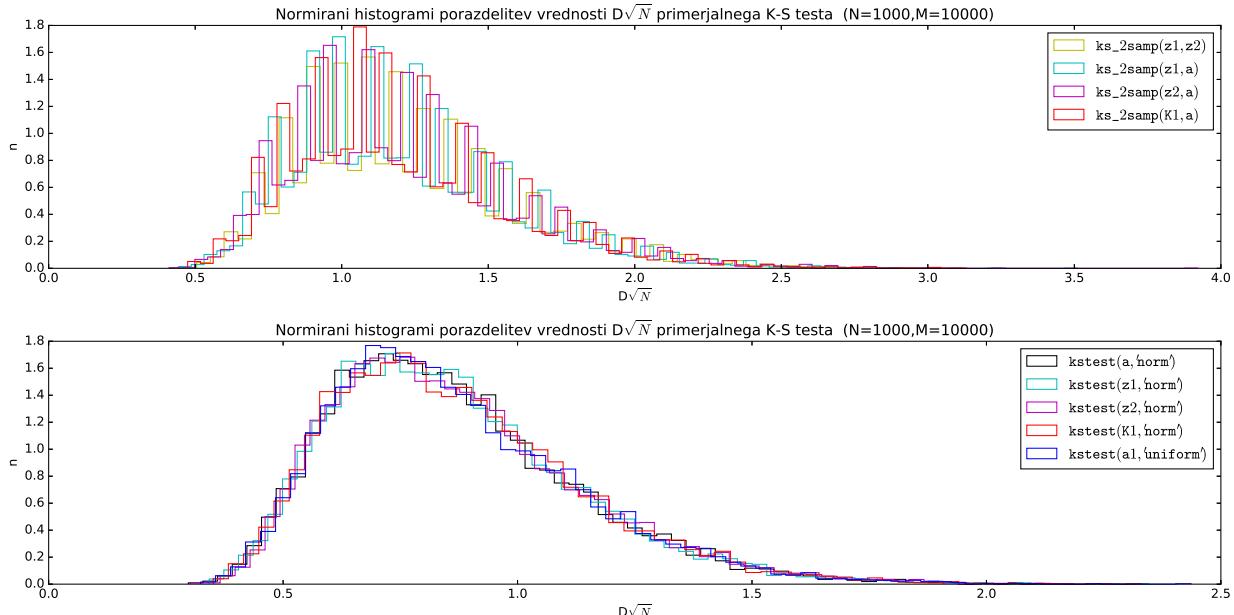
(b)

Slika A.1.

A. Kolmogorov-Smirnov testi



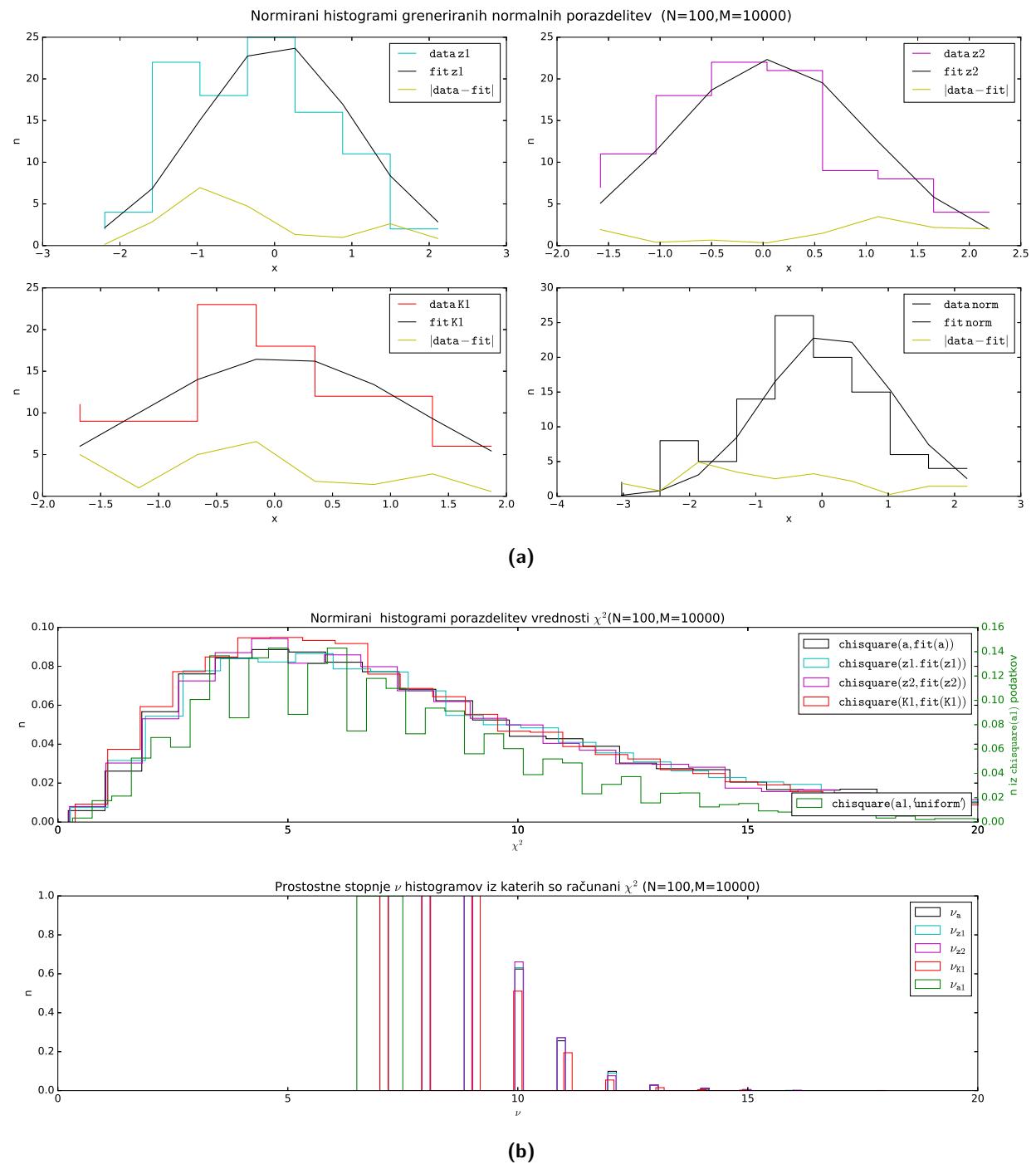
(a)



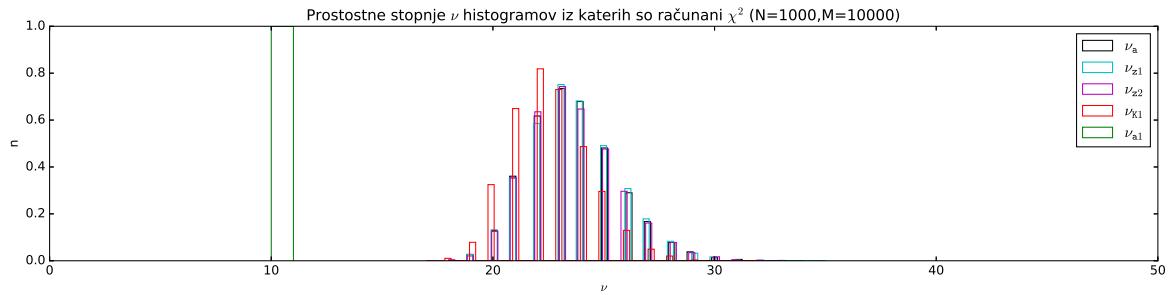
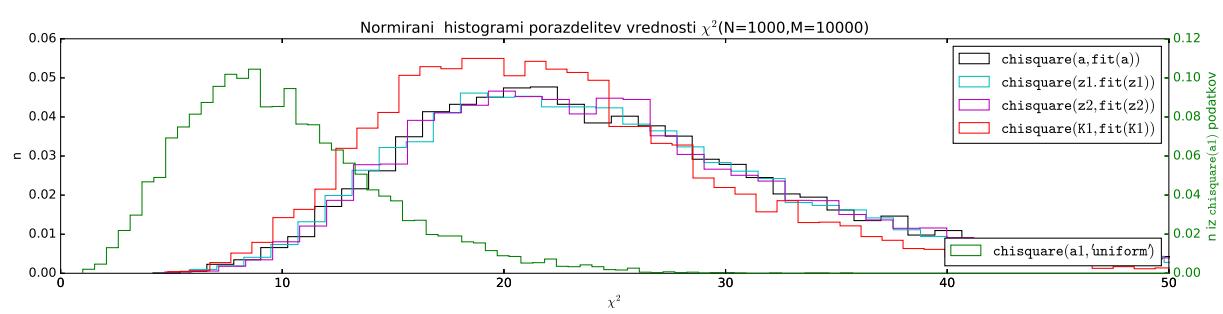
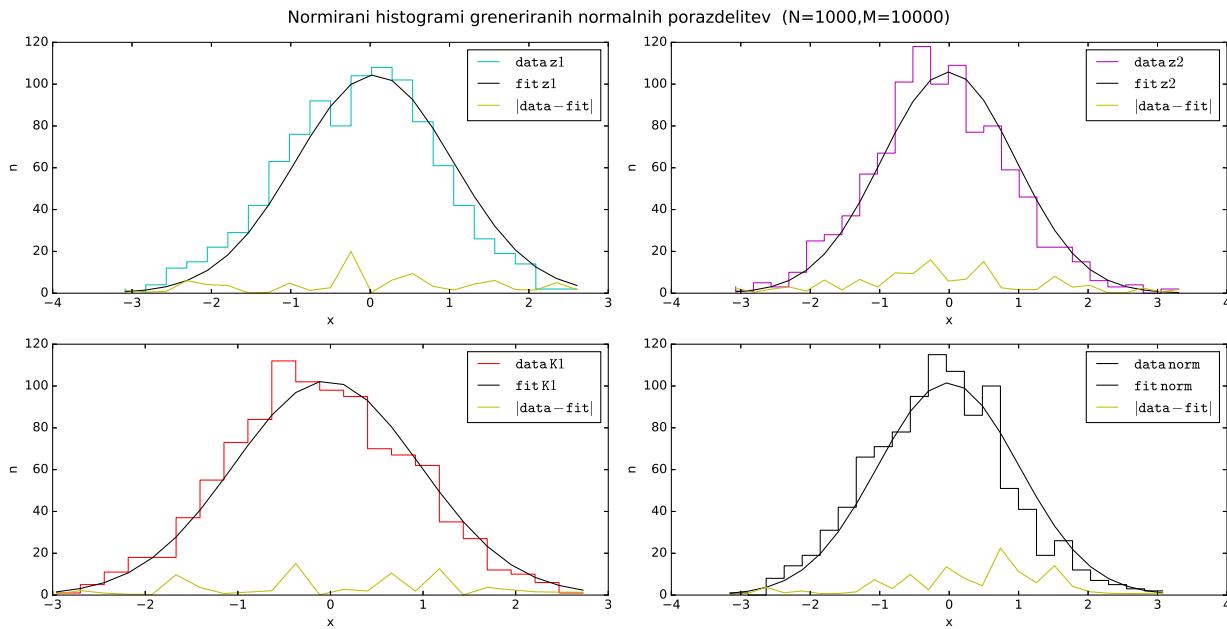
(b)

Slika A.2.

B. χ^2 testi



Slika B.1.



(b)

Slika B.2.

Literatura

- [1] Spletne strani z gradivom predmeta, ogled 30.11.2017,
<http://predmeti.fmf.uni-lj.si/modelska>
- [2] Spletne strani z gradivom predmeta, ogled 30.11.2017,
<http://predmeti.fmf.uni-lj.si/modelska/podatki/>
- [3] ROOT Reference Guide, ogled 30.11.2017,
<https://root.cern.ch/doc/master/classTRandom.html>
- [4] Spletne strani loterije slovenija z objavljeno statistiko igre loto 30.11.2017,
<http://www.loterija.si/loto-rezultati>
- [5] Verjetnost v fiziki, Simon Širca, DMFA, 2016

9. Integracije z metodo Monte Carlo

Povzetek

Pri tej vaji implememetiramo metode generiranja naključnih števil iz prejšnje naloge. V prvem delu vaje generatorje najprej uporabimo pri integracijski metodi Monte Carlo (MC). Ta nam z uporabo naključnih števil omogoča izračune zahtevnih večdimenzionalnih določenih integralov s zapletenimi robnimi pogoji, pri katerih računski algoritmi pogosto zatajijo zaradi prevelike računske zahtevnosti ob uporabi standardnih numeričnih metod. Metodo MC smo pri prvem delu nalogi uporabili za izračun prostornine in vztrajnostnega momenta geometrično zapletenejšega telesa. V drugem delu naloge je obravnanvan model okroglega sevalca žarkov gama pri katerem nas zanima kakšna je verjetnosti izstopa oziroma absorbcije fotona v njem. Pri zadnjem tretjem delu vaje pa preučimo dva modela sisanja nevronov pri prehodu skozi reflektor različnih debelin, ki mu določimo prepustnost in kotno karakteristiko sisanja netronov v prostor.

9.1. Naloga

- Oglej si presek treh medsebojno pravokotnih valjev z enakimi polmeri, katerih osi so poravnane s koordinatnimi osmi. Izračunaj prostornino in vztrajnostni moment dobljenega telesa
 - če je gostota konstantna,
 - če se gostota spreminja z razdaljo od središča kot $\rho = (r/r_0)^p$ in je r_0 polmer telesu očrtane krogle, r pa razdalja od središča telesa. Kak učinek ima vrednost parametra p ?

Oceni natančnost rezultatov.[1]

- V krogli se rojevajo žarki gama. Njihova povprečna prosta pot v snovi, iz katere je krogle, je enaka radiju krogle. Kolikšen delež fotonov uide iz krogle? Kako se verjetnost pobega spreminja z razmerjem povprečne proste poti in radija krogle?[1]
- Model nevtronskega reflektorja: tok nevronov vpada pravokotno na ploščo, v kateri se nevroni sipljejo in nič ne absorbirajo, pri čemer je njihova prosta pot enaka polovici debeline plošče. V poenostavljenem modelu privzamemo, da se sipljejo samo naprej in nazaj, in to z enako verjetnostjo. Kakšna je porazdelitev po številu sipanj? Kolikšna je prepustnost reflektorja? Oceni natančnost. Nekoliko bolj realen je model z izotropnim sipanjem. Z njim preveri, koliko so rezultati poenostavljenega modela uporabni.[1]
 - (neobvezno) Določi kotno porazdelitev odbitih in prepuščenih nevronov ter odvisnost prepustnosti od debeline plošče v modelu z izotropnim sipanjem.

Komentar k reševanju

Računanje je izvedeno v Python-u s, kjer je večino časa v uporabi v knjižnici `numpy` vgrajeni generator naključnih števil iz modula `random`. Natančneje, pri reševanju naloge je uporabljen generator `np.random.rand()`.

9.2. Volumen in vztrajnostni moment skupnega preseka treh enakih pravokotnih valjev

Telo je skupni presek treh valjev z enakim polmerom r in ga je mogoče spraviti v kocko s stranicami dolžine $a = 2r$ in s središčem v izhodišču koordinatnega sistema. Točka $\vec{r} = (x, y, z)$ pripada telesu, če je znotraj obih valjev hkrati, kjer veljajo pogoji

- $x^2 + y^2 < 1$
- $x^2 + z^2 < 1$
- $y^2 + z^2 < 1$

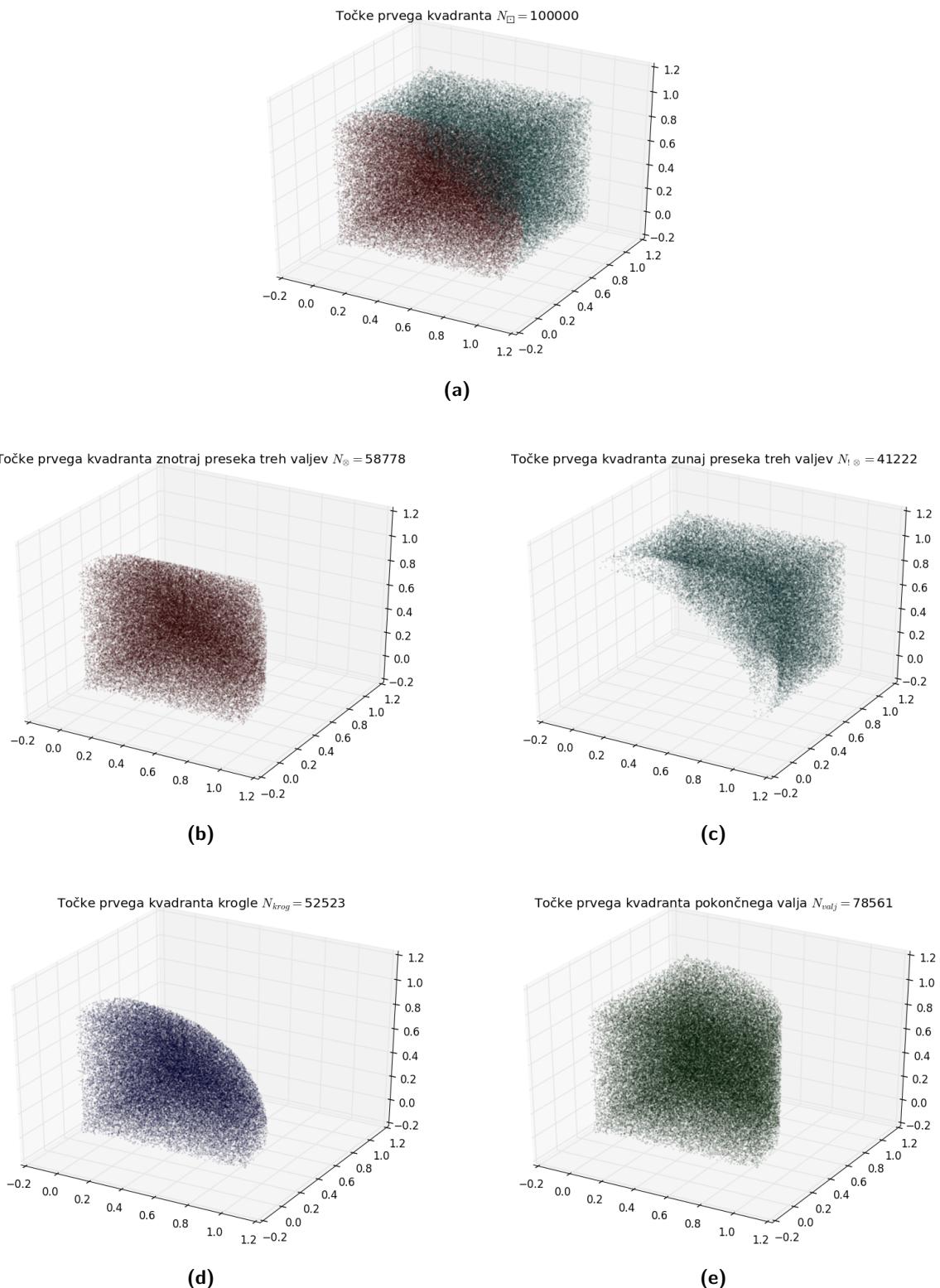
Pri tako izbranem koordinatnem sistemu je zaradi simetrije težišče vedno v izhodišču, potrebno pa je izračunati volumen, porazdelitev mase in od tu še vztrajnostni moment telesa.

9.2.1. Homogeno telo

Za izračun volumena ter mase telesa z enakomerno gostoto naključno izberemo točke znotraj kocke N_{\square} in preštejemo, kolikšen delež le teh N_{\otimes} ustrezata zgornjim pogojem. Ta delež je enak razmerju prostornin telesa V_{\otimes} in kocke V_{\square} ter je enak njunemu razmerju mas

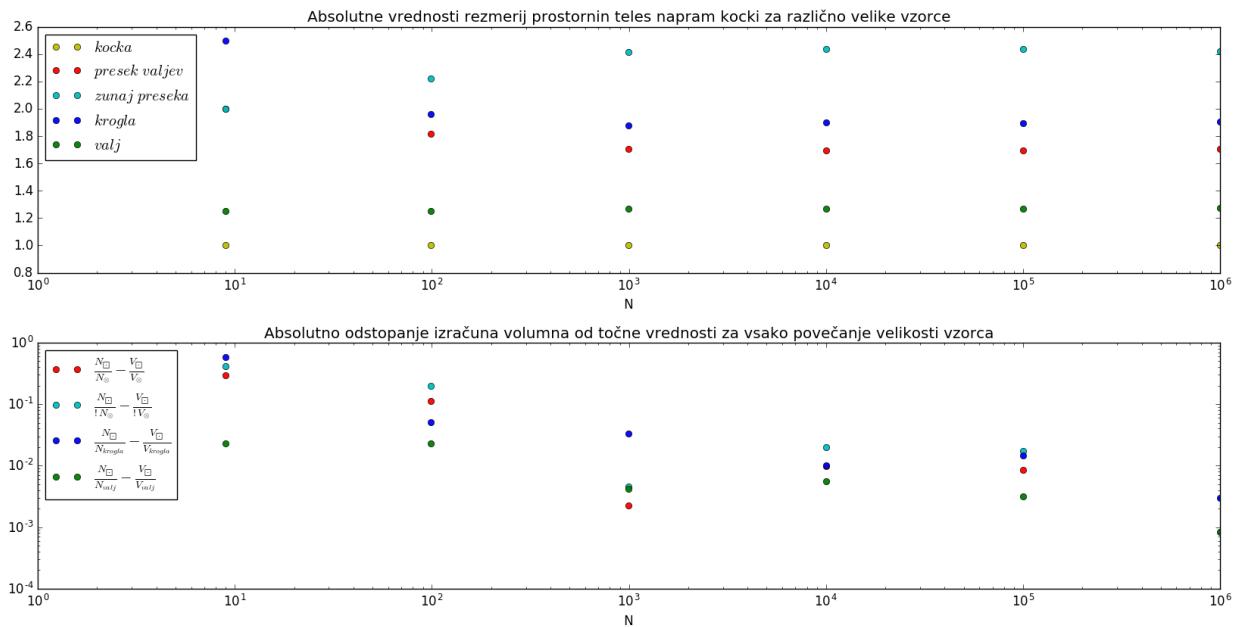
$$\frac{N_{\square}}{N_{\otimes}} = \frac{V_{\square}}{V_{\otimes}} = \frac{m_{\square}}{m_{\otimes}} . \quad (9.1)$$

Podobno lahko preverimo razmerje postopek še za telesa z znanim volumnom, kot sta krogla in valj, kjer prav tako beležimo število točk, ki ustrezajo koordinatnim kriterijem.



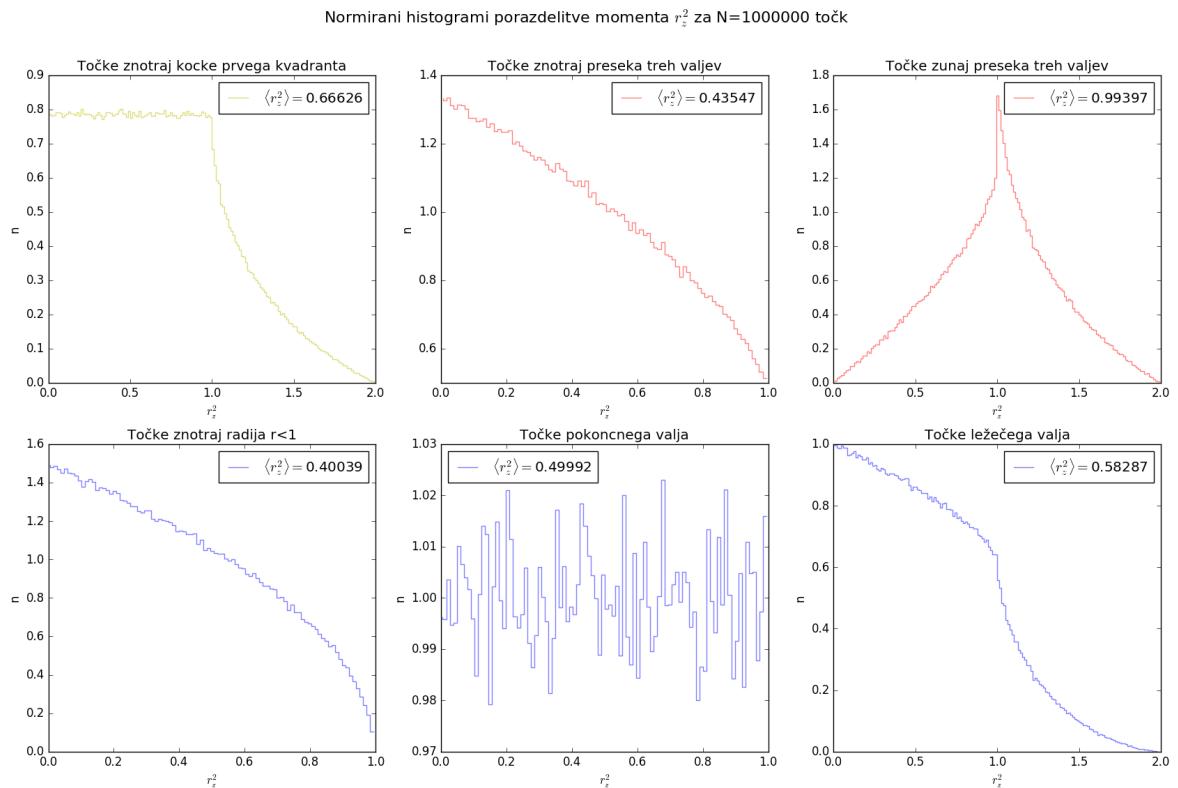
Slika 9.1.: Prikazi polnjenja prostora s točkami za koordinatne kriterije različnih teles. Točno razmerje volumnov kocka/krogla ima vrednost 1.9098.. (izračunan 1.9039..) in kocka/valj 1.2732.. (izračunan 1.2728..). Izračunano razmerje kocka/presek valjev znaša 1.7013.., kjer pri vzorcu $N = 10^5$ rezultatu lahko zaupamo na dve decimalki.

Naključno točko znotraj krogle generiramo s tremi naključnimi števili neodvisnih koordinat. Izžrebani parametri za koordinate x_i , y_i in z_i vsake točke morajo biti naključna števila porazdeljena med 0 in 1, radij od izhodišča do točke pa lahko izrazimo kot $r_i = \sqrt{x_i^2 + y_i^2 + z_i^2}$. Ker so telesa simetrična glede na izhodišče, si je računanje mogoče olajšati s tem, da se omejimo le na enega od osmih kvadrantov prostora. Prikazi takšnega pristopa izbere točk za $a = 2$ oziroma $r = 1$ so vidni na sliki 9.1, rezultati izračuna razmerij tako določenih volumnov pa na sliki 9.2. Opaziti je, da z večanjem vzorca točk negotovost razmerja volumnov pada, vendar pa z vsakim nadaljnji velikostnim redom narašča računska zahtevnost, s čimer se daljsa čas računanja.

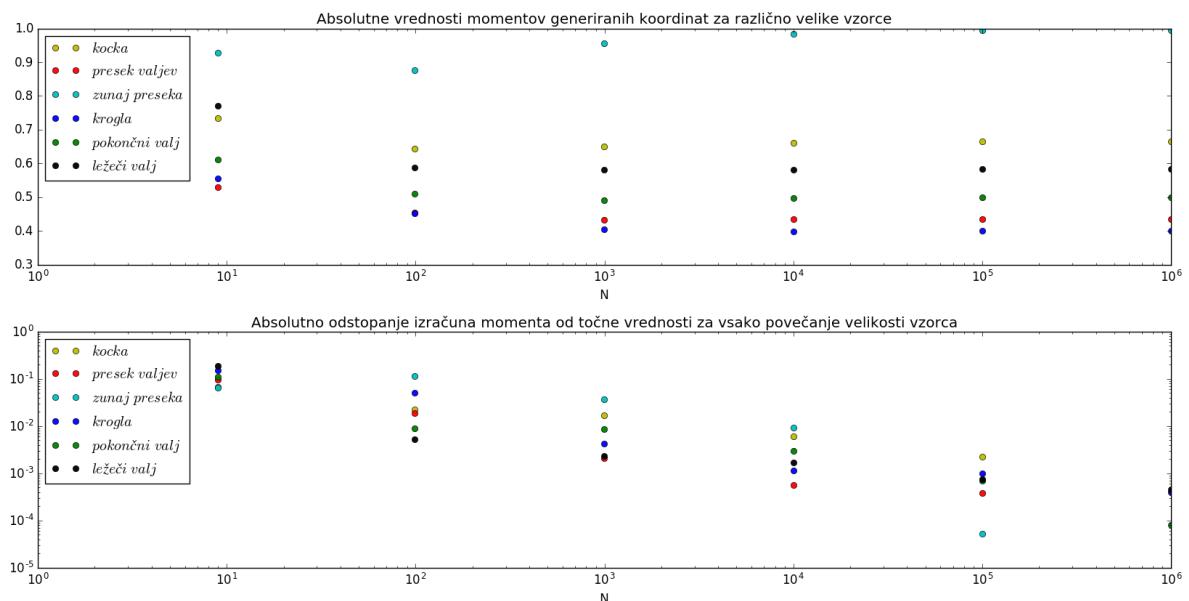


Slika 9.2.: Grafa prikazujata vrednosti razmerij volumna teles napram kocki in odstopanja teh izračunov od pravih vrednosti volumna; oboje v odvisnosti od velikosti vzorca naključno generiranih točk. Pri majhnem številu točk vrednosti seveda močno nihajo, a se z večanjem vzorca točk z vsakim velikostnim redom natančnost izboljšuje. Za presek valjev so odstopanja računana s primerjavo na novo izračunanega rezultata in rezultata iz izračuna z velikostni red manjšim vzorcem.

Za izračun vztrajnostnega moment definiranega kot $J = mr^2$ lahko v našem približku zapišemo, kot da bi imeli sistem točkastih teles $J = \sum_i m_i r_i^2$, kjer je m_i masa izžrebane točke in r_i oddaljenost točke od osi vrtenja. Moment homogenega telesa je definiran kot $J = m \sum_i r_i^2$ torej ga lahko izračunamo kot povprečje $\langle r^2 \rangle$, kjer seštejemo razdalje do osi vrtenja vseh naključno izbranih točk, ki ležijo znotraj telesa, nato pa to delimo z njihovim številom. Tudi v tem primeru lahko tako določen moment presek valjev nato preko razmerja primerjamo z znanimi vrednostimi za kocko ali drugo enostavno telo. Rezultati za tako obravnavan primer vrtenja okrog simetrijske osi z kartezičnega koordinatnega sistema so prikazani na sliki 9.4.



Slika 9.3.: Histogrami porazdelitve točk po prvem kvadrantu različnih homogenih teles glede na vrednost r_z^2 kvadrata oddaljenosti od osi z . Vztrajnostni moment telesa je izračunan kot povprečna vrednost kvadrata oddaljenosti od osi z za vse točke. Točne vrednosti vztrajnostnih momentov so 0.6666.. za kocko, krogla ima vrednost 0.4, ležeči valj 0.5833.. in pokončni valj 0.5. Izračunu za presek treh valjev pri vzorcu $N = 10^5$ lahko torej zaupamo na tri decimalke.

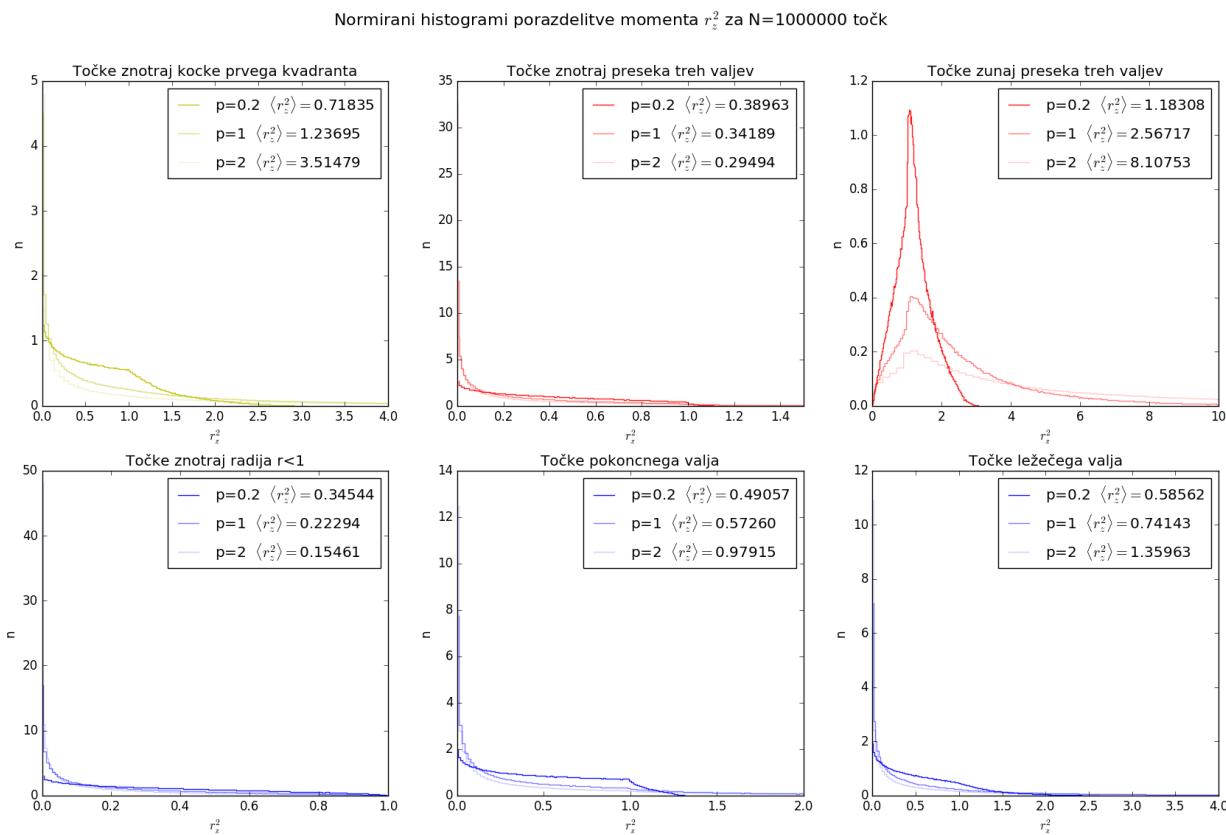


Slika 9.4.: Grafa prikazujata vrednosti vztrajnostnih momentov teles in odstopanja teh izračunov od pravih vrednosti momenta; oboje v odvisnosti od velikosti vzorca naključno generiranih točk. Tudi v tem primeru se z večanjem vzorca natančnost izboljšuje. Za presek valjev so odstopanja računana s primerjavo na novo izračunanega rezultata in rezultata iz izračuna z velikostni red manjšim vzorcem.

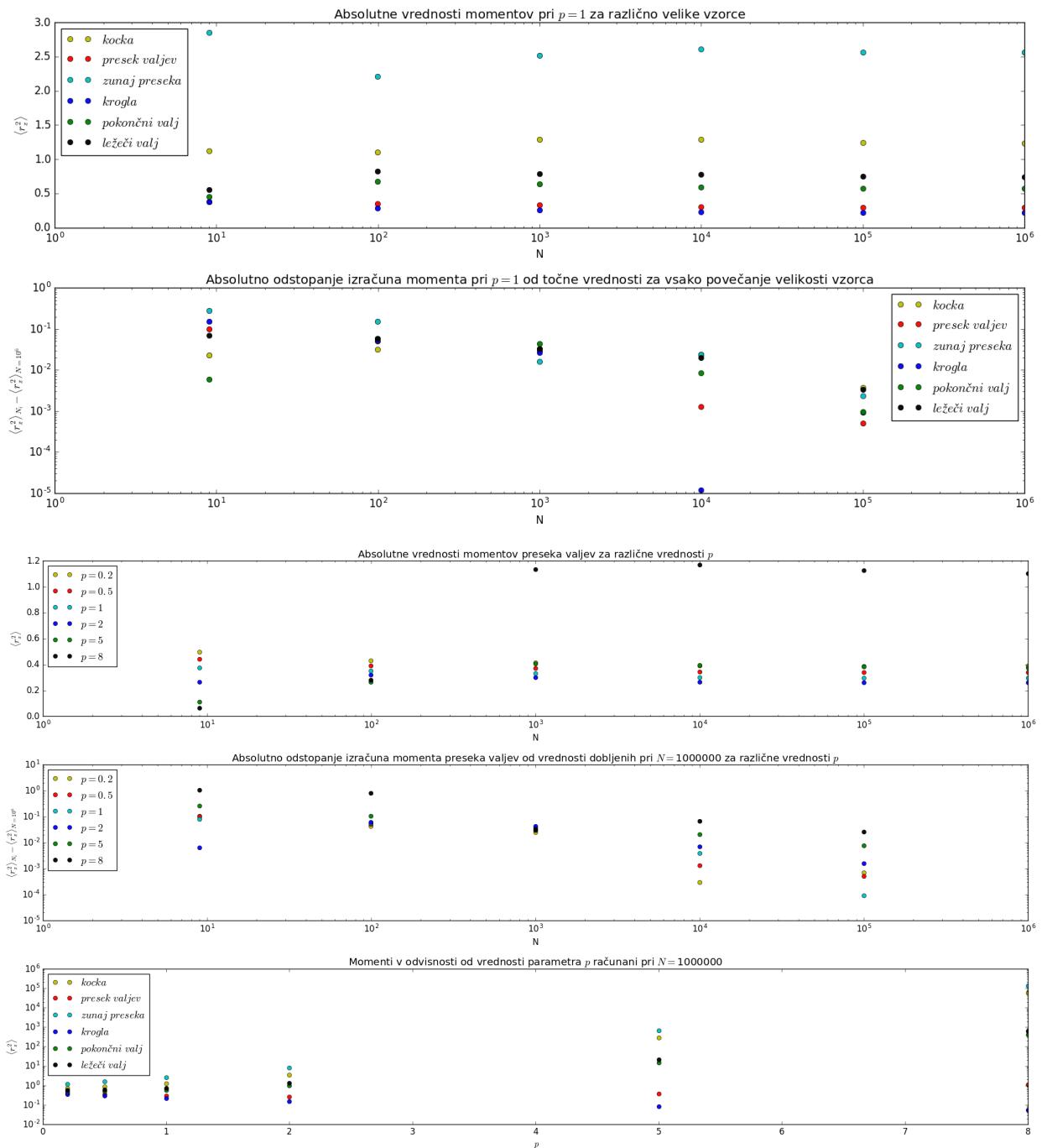
9.2.2. Telo s spremenljivo gostoto

V tem primeru računamo na podoben način kot v primeru homogenega telesa, le da moramo naključne točke upoštevati neenakovredno, saj enakomerne porazdelitev po prostornini ne pomeni več tudi enakomerne porazdelitve po masi. Točke, ki so generirane na enak način kot prej, zato obtežimo skladno z odvisnostjo $\rho = (r/r_0)^p$.

Rezultati računov za $r_0 = 1$, različne velikosti vzorca in različne vrednosti parametra p so prikazani na sliki 9.6. Vztrajnostni momenti so večji kot v primeru homogenega telesa. Točke za katere velja $r < r_0$ z večanjem parametra postajajo manj pomembne k doprinosu k vztrajnostnemu momentu medtem ko točke z $r > r_0$ postajajo pomembnejše. To se lepo opazi iz primerjave vrednosti vztrajnega momenta krogla, ki ima vse točke manjše izhodišču manj oddaljene kot r_0 , in pa momenta točk, ki so zunaj preseka treh valjev pa konstantno narašča - medtem ko za točke iz preseka treh valjev moment sprva pada in doseže nek minimum, potem pa začne naraščati.



Slika 9.5.: Histogrami porazdelitve točk po prvem kvadrantu različnih nehomogenih teles glede na vrednost r_z^2 kvadrata oddaljenosti od osi z . Vztrajnostni moment telesa je izračunan kot povprečna vrednost kvadrata oddaljenosti od osi z za točke obtežene skladno z $(r/r_0)^p$. Za vsako telo je histogram izrisan pri treh različnih vrednostih parametra p . Z večanjem slednjega se porazdelitev z veliko točkami $r > r_0$ razleže proti večjim vrednostim r_z^2 in vrednost vztrajnega momenta narašča, medtem, ko se za telesa z točkami pri $r < r_0$ porazdelitev tudi bolj skoncentrira v bližini izhodišča. Za kroglo in telo preseka valjev z naraščanjem parametra vztrajnostni moment pada, saj za vse ali večino točk velja $r < r_0$.



Slika 9.6.: Vsi grafi razen spodnjega prikazujejo vrednosti vztrajnostnih momentov teles in odstopanja teh izračunov od rezultata iz izračuna z velikostnim redom vzorca 10^6 ; oboje v odvisnosti od velikosti vzorca naključno generiranih točk. Tudi v tem primeru se z večanjem vzorca natančnost izboljšuje vendar počasneje za večje vrednosti parametra p . Spodnji graf prikazuje odvisnost vrednosti vztrajnih momentov teles od velikosti parametra p , od koder je opaziti, da z večanjem parametra narašča tudi vztrajnostni moment - vendar le za telesa, ki imajo točke za katere velja $r > r_0$. Za kroglo z točkami $r < r_0$ pa vztrajnostni moment z večanjem parametra pada, medtem, ko za presek treh valjev moment sprva pada, nad neko vrednostjo parameta pa prične ponovno naraščati, saj takrat prevladajo točke z $r > r_0$.

9.3. Pobeg fotonov iz krogle

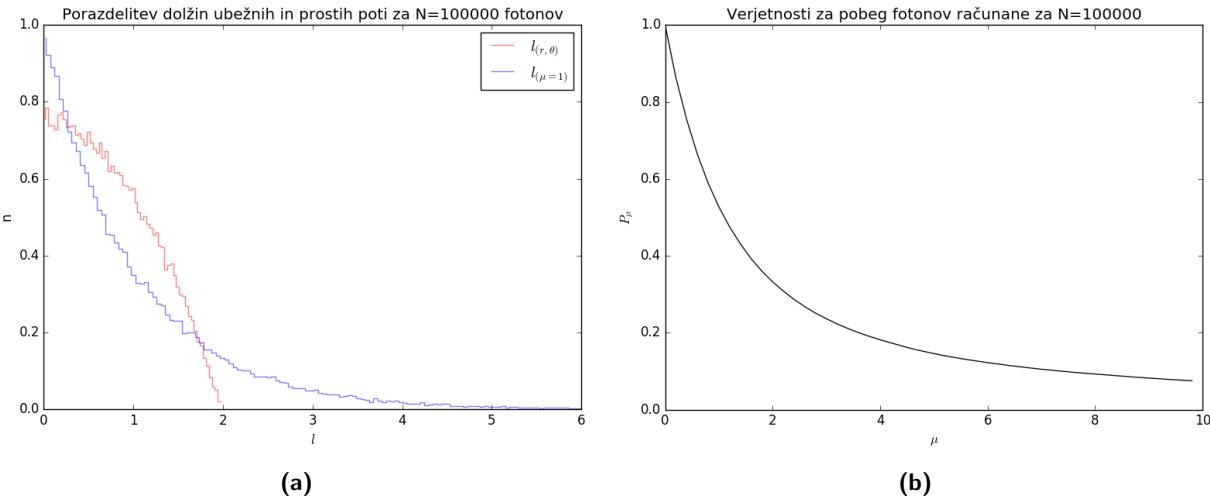
Zaradi krogelne simetrije lahko pot vsakega fotona, ki nastane v krogli, opišemo z razdaljo r od središča krogle do mesta nastanka in kotom θ med radialno premico krogle, ki gre skozi točko nastanka in premico v smeri pogloba fotona. S tem dvema podatkom lahko izračunamo pot fotona iz mesta nastanka do roba krogle, pri tem pa lahko zaradi simetrij koordinate rojevanj fotonov v krogli generiramo kar nekje na polmeru iz središča proti robu krogle in hkrati brez izgube splošnosti privzamemo, da je radij krogle enak $r_0 = 1$. Z uporabo trigonometrije po nekaj korakih dobimo zvezo za dolžino poti fotona v brezdimenzijski obliki

$$l_i(r, \theta) = -r_i \cos \theta_i + \sqrt{1 - r_i^2 \sin^2 \theta_i} . \quad (9.2)$$

Za izračun porazdelitve poti $\frac{dP}{dl}$ generiramo več naključnih točk r_i in smeri θ_i ter tako dobimo različne dolžine, ki jih morajo fotoni prepotovati, da lahko zapustijo kroglo. Medtem so same poti, ki jih fotoni dejansko opravijo odvisne od koeficienteja μ snovi iz katere je kroga. Te so porazdeljene po zakonu $\propto e^{-\mu l(\mu)}$, kjer velja $\mu = r_0/\lambda$ in λ predstavlja povprečno prosto pot. Za generiranje teh poti torej uporabimo zvezo

$$l_i(\mu) = -\frac{\log(1 - P_i)}{\mu} . \quad (9.3)$$

Obe porazdelitvi razdalj sta prikazani na sliki 9.7(a).



Slika 9.7.: (a) Histograma simulirane porazdelitve dolžin poti potrebnih za pobeg fotonov naključno nastalih v volumnu krogle (rdeča) in prepotovanih poti teh fotonov (modra). Poteka krivilj predstavlja verjetnostni porazdelitvi za ubežne dolžine iz vseh točk v krogi $\frac{dP}{dl}$ in za prepotovane poti skladno z povprečne proste poti $\mu e^{-\mu l(\mu)}$. (b) Graf odvisnosti verjetnosti za pobeg od vrednosti μ . Verjetnosti so računane z vzorcem generiranih poti fotonov velikosti $N = 10^5$. Z večanjem razmerja $\mu = r_0/\lambda$ oziroma krajanjem povprečne proste poti λ je verjetnost za pobeg fotonov vse manjša.

Verjetnost pobega zato definiramo kot

$$P = \int \frac{dP}{dl} \cdot \mu e^{-\mu \cdot l} dl , \quad (9.4)$$

kjer je $\mu = r_0/\lambda$ in λ predstavlja povprečno prosto pot fotona v snovi krogle. Ker želimo ugotoviti kakšna je verjetnost, da nastali fotom pobegne iz krogle, potem v njej simuliramo nastajanje fotonov z različnimi ubežnimi potmi in te poti primerjamo z verjetnostjo za pobeg, ki je odvisna od njihove povprečne proste poti v snovi. Če je določena ubežna pot krajša kot določena prosta pot, štejemo

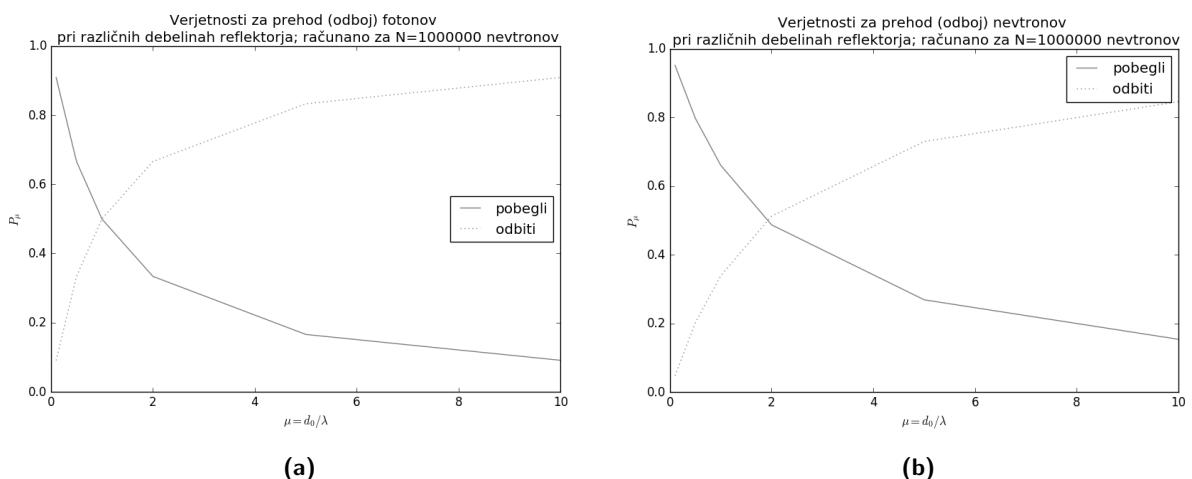
foton za absorbiran, v nasprotnem primeru pa ga prištejemo k pobeglim. Iz razmerja dobimo verjetnost pobega kot $P_{pobegli} = 1 - P_{abs} = 1 - \frac{N_{abs}}{N_{vsi}}$. Za vrednost $r_0/\lambda = \mu = 1$ in vzorec velikosti $N = 10^5$ dobimo verjetnost za pobeg fotna nastalega nekje v krogi $P(\mu = 1) = 0.5286$. Če vrednost povprečne proste poti povečamo pa verjetnost za pobeg iz krogla seveda narašča k vrednosti 1, saj imajo takrat skoraj vsi fotoni precej dobro možnost premagati razdaljo premera krogla. Odvisnost verjetnosti za pobeg od vrednosti μ je prikazana na sliki 9.7(b).

9.4. Model nevtronskega reflektorja

Za konec si poglejmo še model nevtronskega reflektorja v katerem se siplje žarek nevronov, zanima pa nas predvsem prepustnost reflektorja. Žarek vpada pravokotno na reflektor v katerega notranjosti se nato nevtroni sipljejo po poenostavljenem modelu naprej-nazaj ali po realnejšem izotropnem modelu. V obeh primerih lahko za generacijo dolžin prepotovanih poti med posameznimi sisanji nevronov ponovno uporabimo zvezo (9.3), pri čemer za izotropni model z naključno izbiro kota sisanja vsako generiramo pot še projeciramo na vpadno smer $l_{ix} = l_i(\mu) \cdot \cos \theta_i$.

Vsak delec se siplje toliko časa dokler ne zapusti reflektorja na vpadni ali prepustni strani. Pri tem si beležimo število njegovih sisanj in stran izstopa iz reflektorja. Ker nas zanima tudi odvisnost od debeline pregrade, lahko sproti beležim prehode s kriteriji za različne izbrane debeline reflektorja in tako simulacijo za posemezen delec ustavim, šele ko delec zapusti najdebelejši primer reflektorja (ozioroma izstopi na vpadni strani). Za kotno porazdelitev pa si shranjujem končne vrednosti vseh kotnih projekcij simuliranih poti.

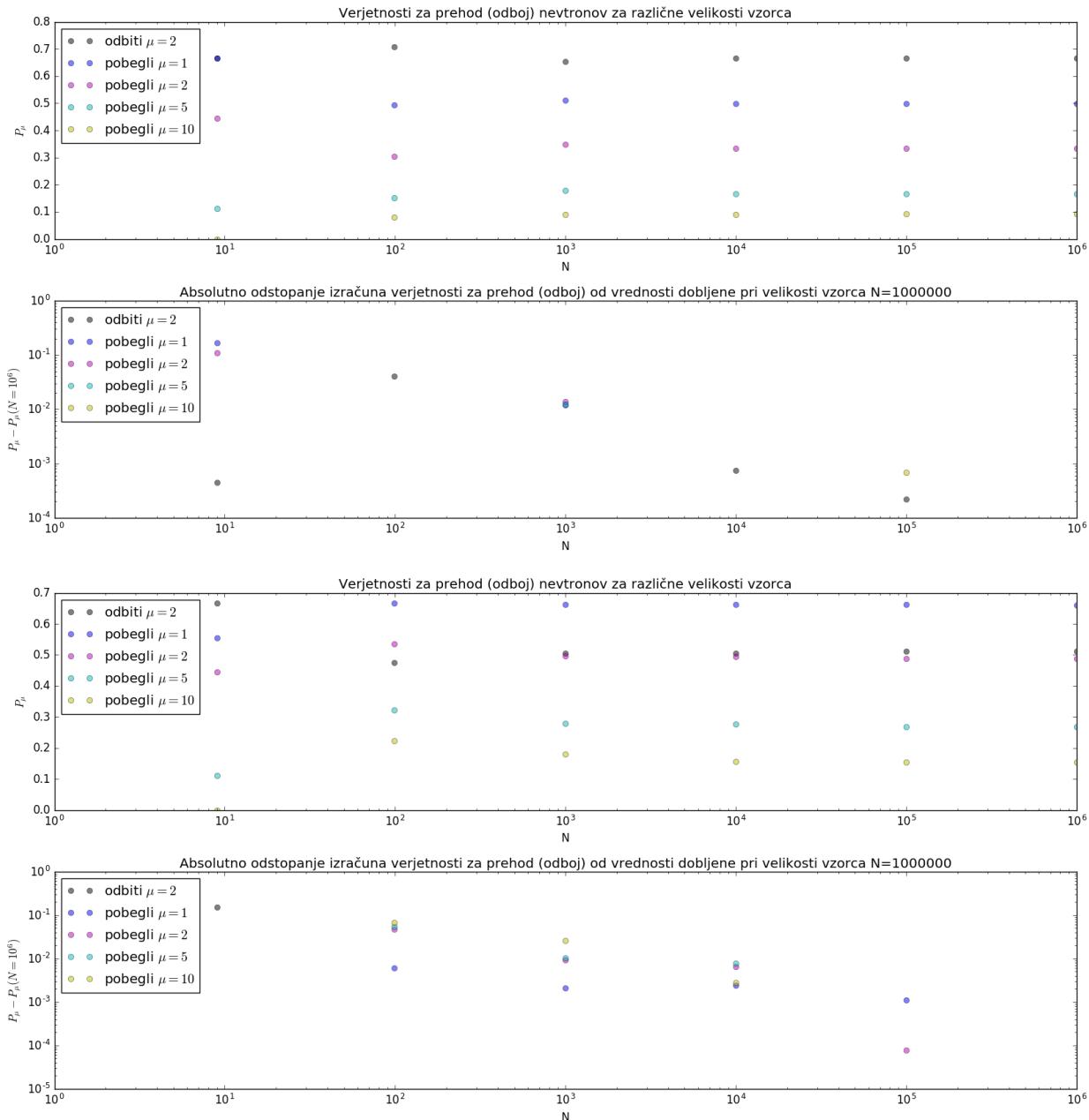
Na sliki 9.8 je prikazana verjetnost za nevronov prehod reflektorja s prosto potjo $\lambda = 1$ za različne debeline d_0 stene pri uporabi vzorca velikosti $N = 10^6$. Na sliki 9.9 pa je prikazana odvisnost izračuna teh verjetnosti od različnih velikosti vzorca nevronov za prehod reflektorja za različne debeline reflektorja. Prepustnost z upoštevanjem modela sisanja naprej-nazaj je zelo blizu rezultatu izotropnega modela sisanja. Z večanjem vzorca seveda natančnost izračuna narašča, ampak, ker je izotropni model računsko bolj zahteven, saj ga omejuje dodatna generacija naključnih števil in izbirnih kriterijev, je tudi model s sisanjem naprej in nazaj primeren za opis prepustnosti reflektorja.



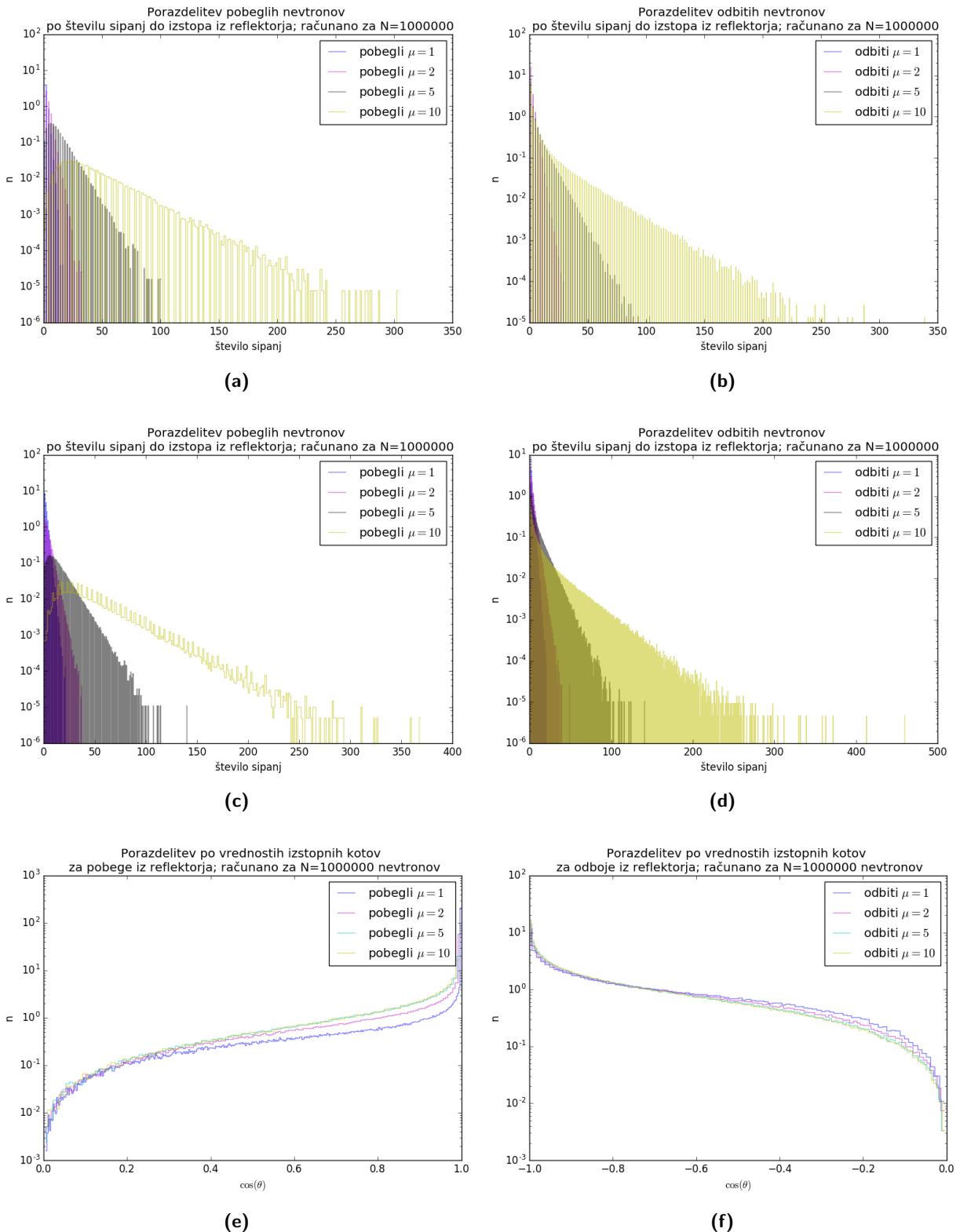
Slika 9.8.: Graf verjetnosti za prehod (odboj) nevronov pri vpadanju na reflektor za model sisanja naprej-nazaj (a) in za izotropni model sisanja. Verjetnosti so računane z vzorcem generiranih poti fotonov velikosti $N = 10^6$. Z večanjem razmerja $\mu = r_0/\lambda$ oziroma debeline d_0 reflektorja pri isti povprečni prosti poti λ je verjetnost za pobeg nevronov vse manjša.

Histogrami porazdelitev po številu sisanj iz obeh modelov sisanja in po izstopnem kotu za izotropni model so za različne debeline reflektorja prikazani na sliki 9.10. Same oblike porazdelitev števila sisanj so si precej podobne za oba obravnavana modela, le v modelu z izotropnim sisanjem

so višja števila sipanj bolj verjetna, vendar je vseeno moč zaključiti, da velike razlike med njima ni opaziti. Pri obeh modelih se z debelejšo steno porazdelitev raztegne proti večjim številom sipanj, vrh porazdelitve pa se odmakne od verjetnosti, da bi nevron brez sipanja prešel skozi reflektor. Porazdelitve po izstopnem kotu za izotropni model kaže v prid majhnim kotom. Z debeljenjem stene pa se povdari asimetrija med odbitimi in prepuščenimi nevroni, ki začnejo prehajati reflektor pri vedno manjših kotih hkrati pa na drugi stani pri večjih kotih opazimo manj odbojev.



Slika 9.9.: Zgornja grafa pripadata modelu sipanja naprej-nazaj, spodnja pa izotropnemu modelu sipanja. Grafi prikazujejo izračunane vrednosti verjetnosti za prehod (odboj) nevronov skozi reflektorje različnih debelin in odstopanja teh izračunov od rezultata iz izračuna z vzorcem velikostnega reda $N = 10^6$; oboje v odvisnosti od velikosti vzorca naključno generiranih točk. Tudi v tem primeru se z večanjem vzorca natančnost.



Slika 9.10.: Levo so prikazani normirani histogrami porazdelitev za prepuščene delce, desno pa za odbite delce; oboje pri različnih debelinah reflektorja. Histograma (a) in (b) pripadata modelu sipanja naprej nazaj. Porazdelitvi ne odstopata veliko v primerjavi izotropnega sipanja prikazane na grafih (c) in (d), kjer je pri slednjih opaziti večje verjetnosti za veliko število sipanj. Z večanjem razmerja $\mu = r_0/\lambda$ oziroma odebeltvijo d_0 stene je verjetnost za večkratno sipanje vse večja in verjetnost za prehod nevronov skozi reflektor vse manjša. Histograma (e) in (f) prikazujeta porazdelitev prepuščenih in odbitih nevronov po izstopnem kotu iz reflektorja. Vse vrednosti so računane z vzorcem generiranih poti (smeri) nevronov v velikosti $N = 10^6$.

Literatura

- [1] Spletne strani z gradivom predmeta, ogled 20.10.2017,
<http://predmeti.fmf.uni-lj.si/modelska>

10. Metropolisov algoritem

Povzetek

Pri tej vaji se spoznamo z Metropolisovim algoritmom. V prvi nalogi ga preizkusimo na molekularni verigi, ki jo opišemo z potencialno in prožonostno energijo njenih posameznih členov. Z algoritmom je mogoče poiskati njeno ravnovesno stanje in preučiti vpliv temperature nanjo. Pri drugi nalogi simuliramo Isingov model mreže spinov, kjer z metropolisovim algoritmom ponovno poiščemo ravnovesje in raziščemo energijo, magnetizacijo, magnetno susceptibilnost in specično toploplotne spinske mreže. Pri dodatni nalogi poskusimo algoritom uporabiti za reševanje problema trgovskega potnika, kjer je potrebno najti najkrajšo pot, ki povezuje vse podane očke.

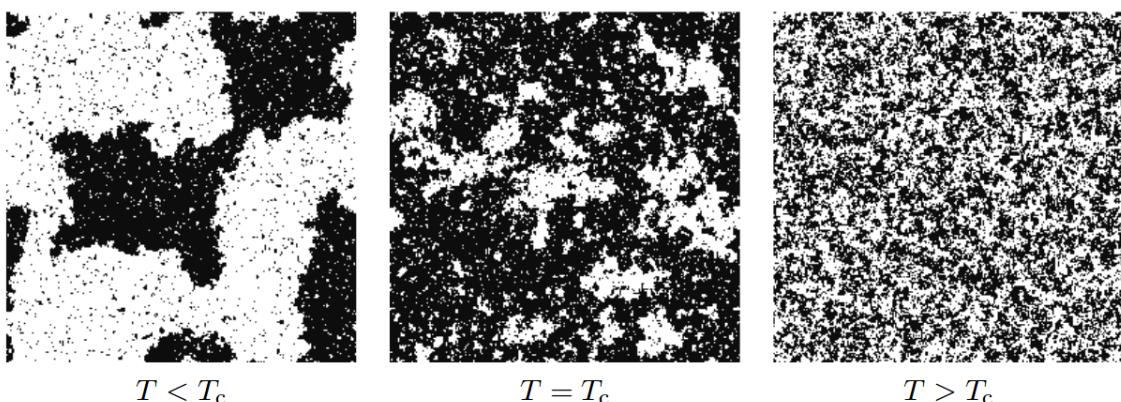
10.1. Naloga

- Molekularna verižnica :** 17 členkov dolga nitkasta molekula je obešena za oba konca. Vsak členek se lahko povesi od ničelne lege na poljubnega od 19 nivojev in si s tem zmanjša potencialno energijo za eno enoto na nivo. Če pa s tem prenategne vezi do sosedov, plača s prožnostno energijo, ki je za vsakega sosedja enaka kvadratu razlike v nivojskem številu. Določi ravnovesno energijo v odvisnosti od temperature. Za poteze lahko uporabiš spremembo za en nivo na izbranem slučajnem mestu.
- Za **Isingov model** določi povprečno energijo $\langle E \rangle$ in lastno magnetizacijo $\langle S \rangle$ odvisnosti od temperature. $S = \sum_i^N s_i$ je celotna magnetizacija sistema. Oglej si tudi spinsko susceptibilnost in specifično toploto pri različnih jakostih magnetnega polja.
Isingov model feromagnetne in antiferomagnetne snovi v dveh dimenzijah v približku dveh stanj opišemo s Hamiltonovim operatorjem

$$\mathcal{H} = -J \sum_{\langle ij \rangle} s_i s_j - H \sum_i s_i, \quad J = \pm 1. \quad (10.1)$$

kjer je $s_i = \pm 1$ in vsota teče le po vezeh $\langle ij \rangle$ med najbližnjimi sosedji. Če ni zunanjega polja ($H = 0$), temperatura T_c faznega prehoda pri feromagnetu zadošča enačbi

$$\sinh \frac{2J}{k_B T_c} = 1 \implies T_c \approx 2.269185 \frac{J}{k_B}. \quad (10.2)$$



Slika 10.1.: Obnašanje spinske mreže pri različnih temperaturah.[1]

- (neobvezno) Analiziraj zanesljivost Metropolisovega algoritma na **modelu trgovskega ponika**, tako da rezultate primerjaš s točnimi rezultati pri majhnem številu mest. Tablico razdalj lahko kar izžrebaš.

Komentar k reševanju

Računanje je izvedeno v Python-u s, kjer je večino časa v uporabi v knjižnici `numpy` vgrajeni generator naključnih števil iz modula `random`. Natančneje, pri reševanju naloge je uporabljen generator `np.random.randint()`.

10.2. Molekularna verižnica

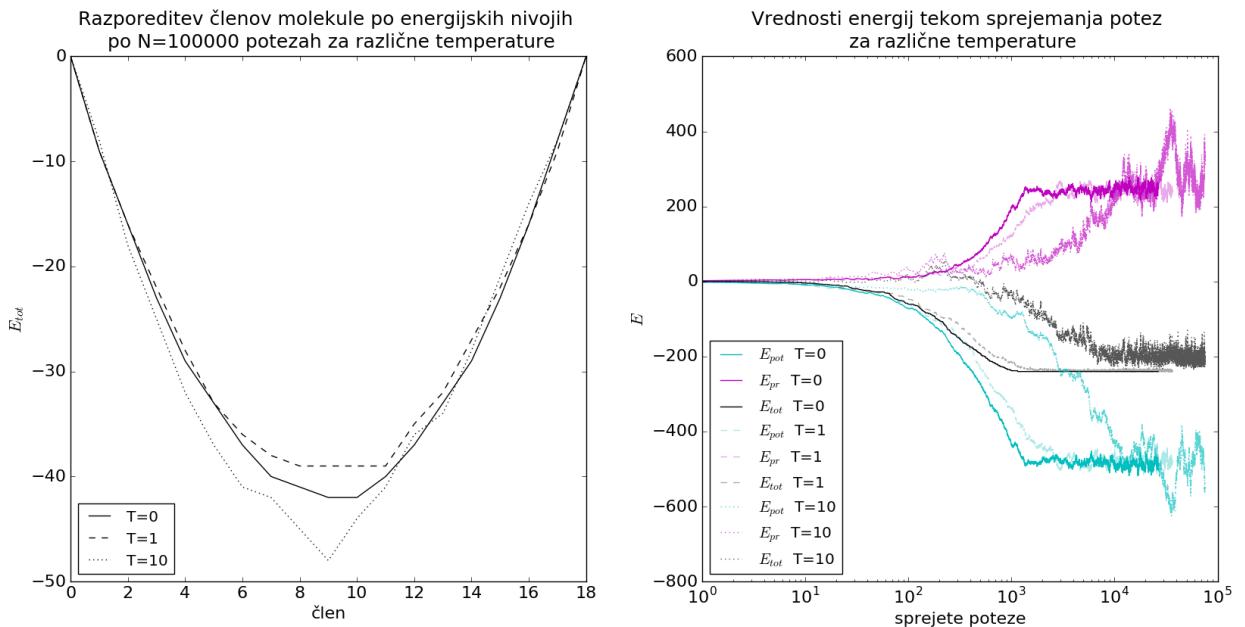
Energijo verižice lahko zapišemo kot vsoto potencialne in prožnostne energije

$$E_{tot} = E_{pot} + E_{pr} = \sum_1^{17} mgh_i + \frac{k}{2} \left(\sum_1^{16} (h_i - h_{i+1})^2 + \sum_2^{17} (h_i - h_{i-1})^2 \right)$$

kjer so h_i energijski nivoji posameznih členkov označenih z $1 \leq h_i \leq 20$, z $i = 1, 2, \dots, 17$. Vrednosti koeficientov lahko izberemo kot $m = g = k = 1$.

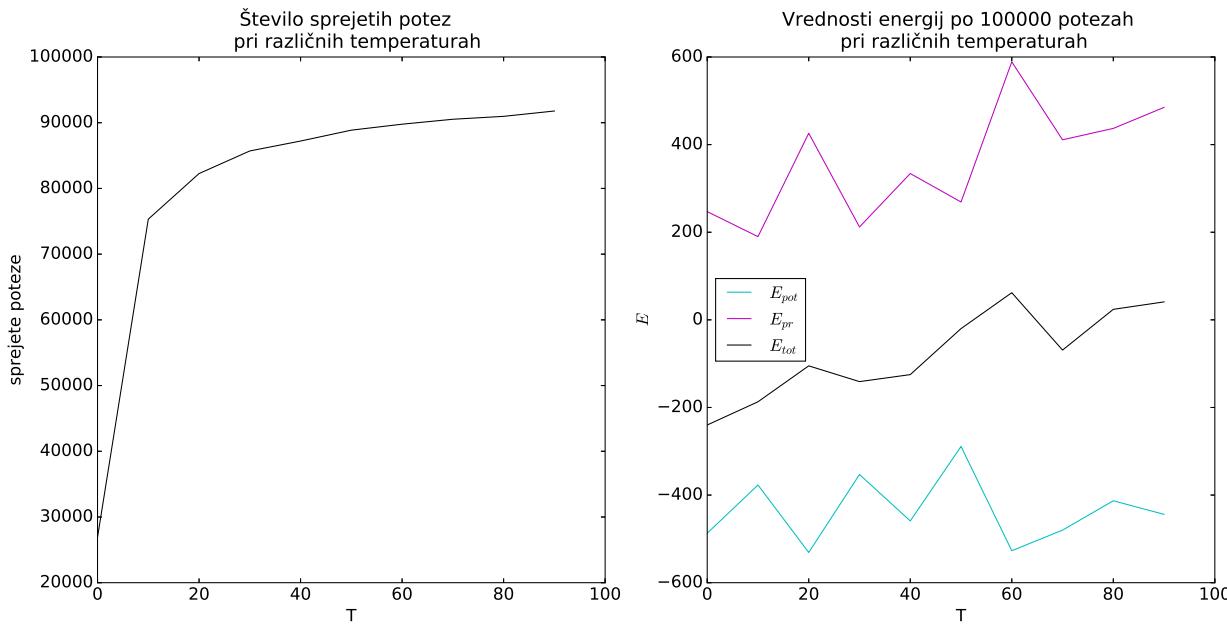
Naloge se lotimo tako, da pri vsaki potezi računamo le spremembo energije stanja naključno izbranega i člena verige za potezo $h_i \rightarrow h'_i$. Če je $\Delta E_{tot} < 0$ potezo sprejmemo, če je $\Delta E_{tot} > 0$ potezo sprejmemo z verjetnostjo $p = e^{-\Delta E_{tot}/T}$, kjer je T temperatura. Pri tem torej verižnici naključno nižamo oziroma višamo energijo.

Sledеči izračuni so izvedeni za različne vrednosti temperature in za vsako temperaturo je generiranih 10^5 potez. Tekom izvajanja potez veriga ostaja pripeta, njen največji odmik pa je na sredini. Energija verižnice se sčasoma ustali pri neki ravovesni vrednosti vendar tekom korakov precej opleta okrog te vrednosti, tako da moramo gledati povprečje. Ko takšno verigo segrevamo, sprejemamo vedno več potez, ki ne predstavljajo znižanja energije, s tem veriga postajata vedno manj gladka. Odvisnost ravovesne energije verižnice za različne temperature in konvergiranje k ravovesni energiji verižnice pri različnih temperaturah sta prikazana na sliki 10.2. Opazimo, da energija potrebuje nekaj časa, da doseže ravovesje, ta čas pa se z višanjem temperature daljša.



Slika 10.2.: Na levi sliki so prikazana stanja členov verižnice po izvedeni zadnji potezi. Vidi se, da se s padcem energije na levi sliki verižnica povesi do nizkih stanj. Z višanjem temperature postaja verižnica vedno bolj nepravilne oblike, kar narekuje, da pri še višjih temperaturah izgubi obliko in so členi precej naključno razporejeni. Na desni sliki so prikazane vrednosti prožnostne in potencialne energije ter njune vsote tekom sprejemanja potez. Da energija doseže ravovesje potrebuje nekaj časa, na koncu pa okrog ravovesja opleta. Višje temperature zahtevajo več sprejetih potez za dosego ravovesne lege okrog katerih potem tudi močneje opleta.

Če izračunamo odvisnosti energij od temperature lahko opazimo da te z višanjem temperature naraščajo, pri tem pa je pri pogoju $\Delta E_{tot} > 0$ z verjetnostjo $p = e^{-\Delta E_{tot}/T}$ sprejetih vedno več za dosego minimalne energije neugodnih potez; pričakovati je, da se z višanjem temperature $T \rightarrow \infty$ sprejmejo vse poteze. Oboje je razvidno iz slike 10.3.



Slika 10.3.: Iz levega grafa je videti da z naraščanjem temperature sprejmemo vedno več potez od 10^5 izvedenih. Iz desnega grafa pa je moč razbrati, linearno odvisnost količin od temperature. Energiji verige s temperaturo naraščata in vedno bolj nihata okoli srednje vrednosti, prav tako niha tudi njuna vsota a je nekoliko manj raztresena. Zaradi višanja temperature se členi verige vse bolj neurejeno gibajo in imajo več prostosti, zato se tudi energija veča.

10.3. Dvodimenzionalni Isingov model

Pri modeliranju feromagnetne in antiferomagnetne snovi nas zanima povprečna lastna magnetizacija snovi in povprečna energija v odvisnosti od temperature. Celotna magnetizacija snovi je $\langle S \rangle = \frac{1}{L^2} \sum_i^N s_i$ kjer je $s_i = \pm 1$ spin posameznega delca oz. kvadratka mreže, ki ponazarja magnetizacijo \downarrow ali \uparrow . Povprečno energijo pa izračunamo kot $\langle E \rangle = \frac{1}{L^2} \langle \mathcal{H} \rangle$. Od tu pa lahko izračunamo še magnetno susceptibilnost

$$\chi = \frac{\langle S^2 \rangle - \langle S \rangle^2}{Nk_B T}$$

in specifično toploto

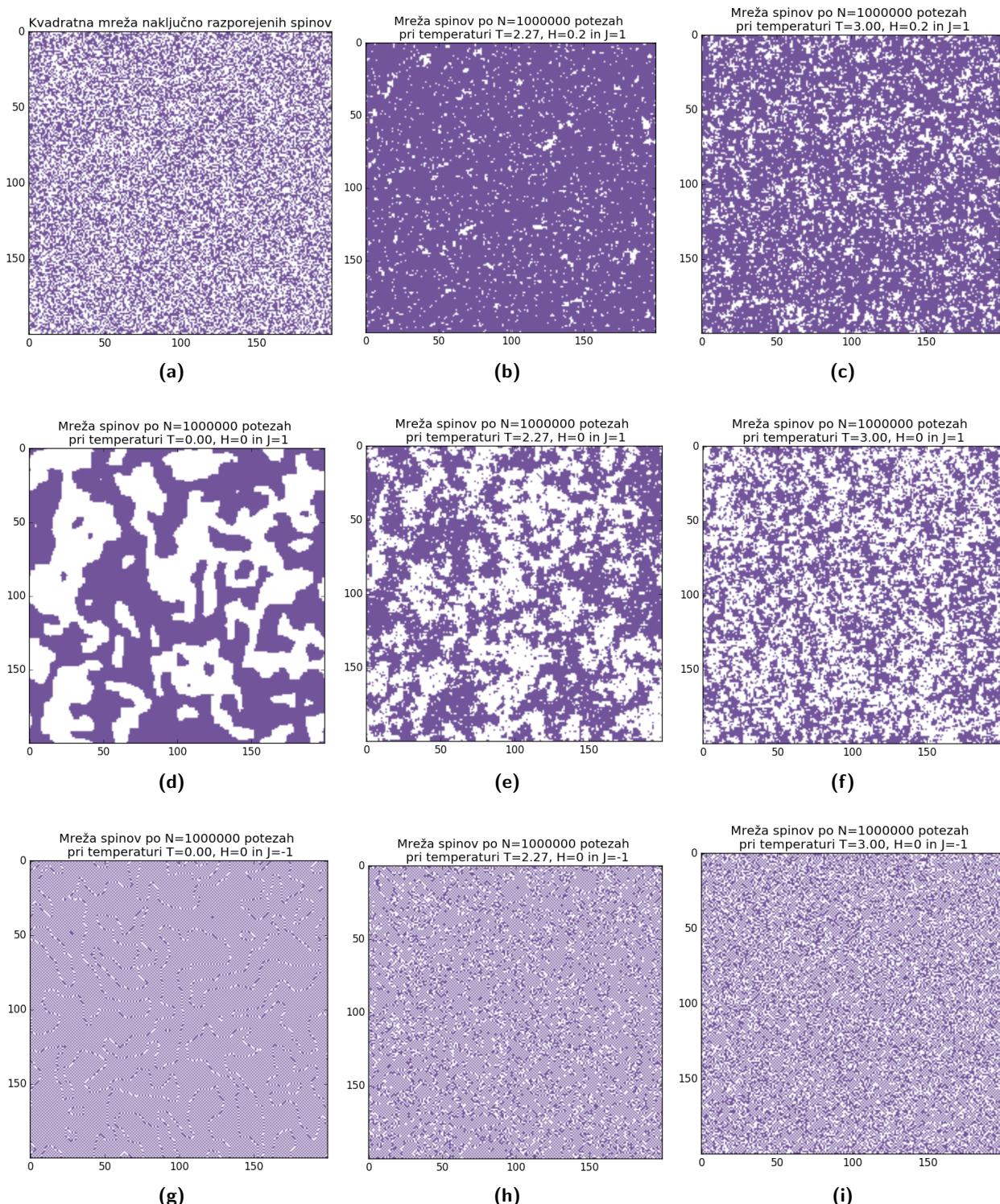
$$c_v = \frac{\langle E^2 \rangle - \langle E \rangle^2}{Nk_B T^2}$$

pri različnih jakostih zunanjega magnetnega polja. Pri izračunih sta sklopitvena konstanta dveh spinov J in boltzmanova konstanta k_B ponovno zaradi preprostosti postavljeni na 1.

Računi so izvedeni za kvadratne mreže različnih velikosti. Vsote tečejo le po vezeh $\langle ij \rangle$ med štirimi najbližjimi sosedji, robni pogoji pa so ciklični. Začnemo lahko z naključno urejeno mrežo spinov ali pa z enakomerno mrežo spinov, kjer so vsi spinii enako obrnjeni. Posamezne delce katerim spin obračam izbiram z naključno in sicer kordinato x in y posebej, poteze pa so sprejete kadar energijo sistema zmanjšajo oziroma po načelu verjetnosti, kot je bilo to izvedeno pri verižnici.

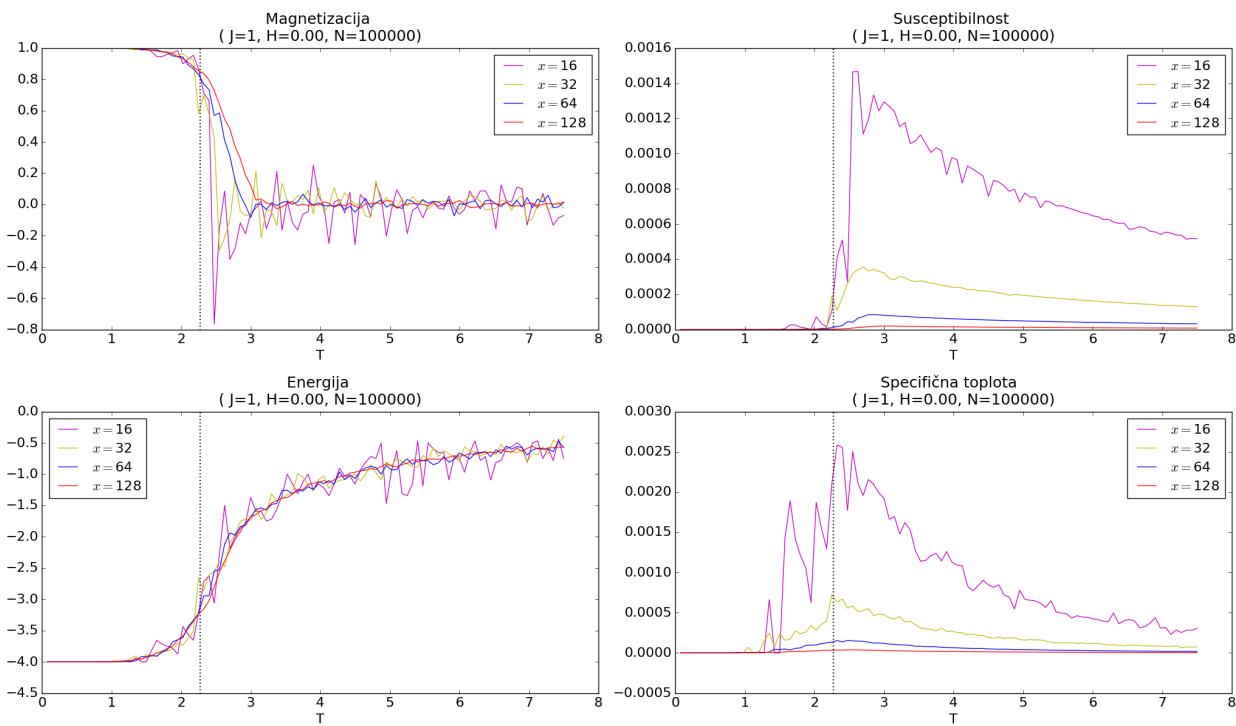
Najprej sem izračunal porazdelitev mreže spinov za tri različne temperature za antiferomagnetno in feromagnetno snov v odsotnosti magnetnega polja. Pri slednji snovi sem porazdelitev izračunal še za primer prisotnosti magnetnega polja. V vsakem primeru sem izvedel 10^6 potez zasuka naključno izbranih spinov. Rezultati izračunov so prikazani na sliki 10.4. Iz naključne porazdelitve se po vsaki potezi stanje sistema bolj približuje stanju z spinskimi domenami; te so očitno vidne za primer feromagneta, pri antiferomagnetu pa se domene pojavijo v obliki mejnih linij. Z višanjem temperature se za obe snovi domene drobijo in gredo pri kritični temperaturi skozi fazni prehod. Urejenost antiferomangeta je podobna šahovnici, kjer se spinii uredijo tako, da težijo k ničelni magnetizaciji.

Pri antiferomagnetu torej nimamo tako izrazitega faznega prehoda. Tudi vklop magnetnega polja na antiferomagnet ni imel posebnega učinka, rezultati za feromagnet pa kažejo, da spini silijo v preferenčno smer polja.



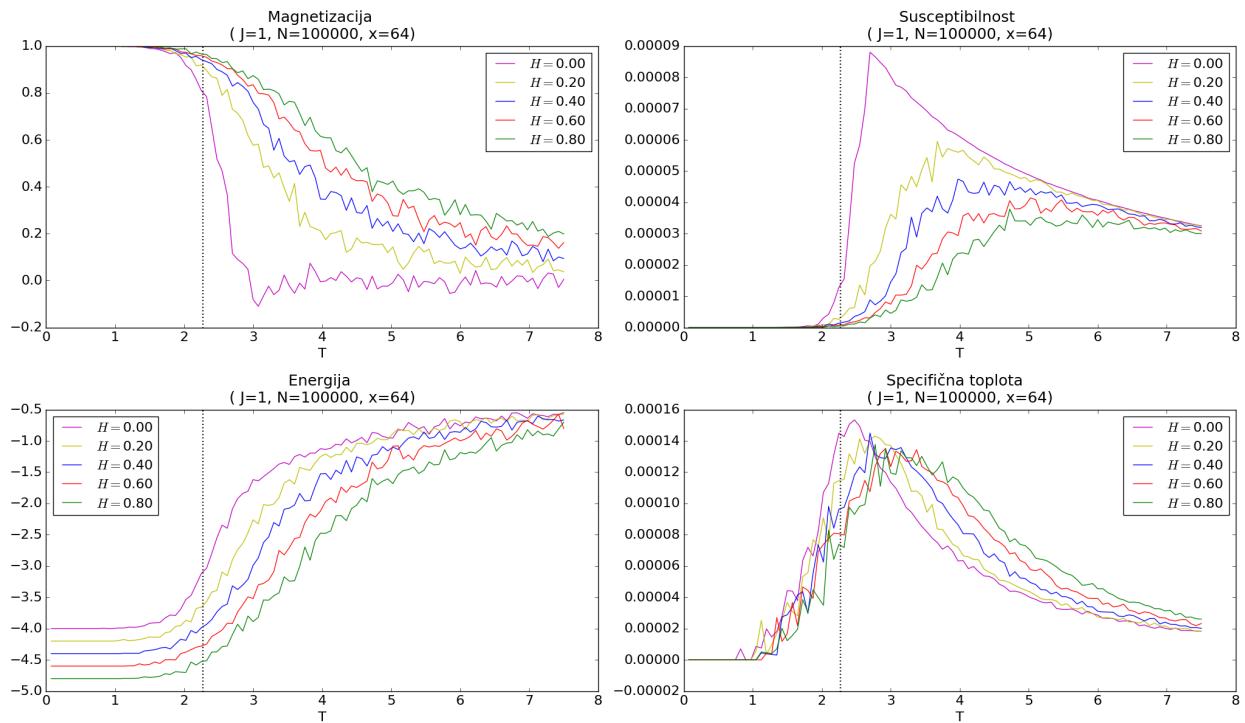
Slika 10.4.: Rezultati feromagnetske ($J = 1$) in antiferomagnetske ($J = -1$) spinske mreže naključnih spinov velikosti 200×200 (a) po simulaciji za tri različne temperature v prisotnosti magnetnega polja (b,c) in odsotnosti magnetnega polja (d-i). Za obe snovi opazimo tvorjenje domen, ki se z višanjem temperature vedno bolj drobijo. Prisotnost polja sili spine v preferenčno smer - pri podkritični temperaturi bi se vsi spini uredili v smer polja.

Nadalje sem pri računanju energije, magnetizacije in obeh iz njiju izpeljanih količin v odvisnosti od temperature uporabil feromagnetne spinske mreže različnih velikosti. Začetno stanje mreže je bila enaka usmeritev vseh spinov na katerih je bilo izvedenih 10^5 zasukov naključnih spinov, ki pod že omenjenimi sprejemnimi pogoji potez sistem spravijo v ravnovesno energijo. Rezultati simulacij v odsotnosti magnetnega polja so prikazani na sliki 10.5. Iz grafov je opaziti relaksacijo magnetizacije in energije sistema h neki končni vrednosti. Z višanjem temperature energija narašča magnetizacija pa je maksimalna na začetku, ko so vsi spini enako orientirani. Hiter porast oziroma padec se zgodi v bližini kritične temperature. Končna relaksacija obeh količin pa je hitrejša a bolj neenakomerna za mreže manjših dimenzij, saj je tu manj možnih kombinacij usmeritve spinov. Susceptibilnost in specifična toplota sprva narasteta in dosežeta maksimum pri kritični temperaturi ter z višanjem temperature preko kritične vrednosti pričneta padati. Tudi pri teh dveh količinah je učinek izrazitejši za manjše mreže.



Slika 10.5.: Grafi štirih opazovanih količin v odvisnosti od temperature za nekaj feromagnetnih spinskih mrež različnih populacij spinov x^2 in pri odsotnem magnetnem polju. Kritična temperatura je označena s črtano črto. Videti je da energija začetnega stanja vztraja pri najnižji vrednosti dokler ni dosežena kritična temperatura, nato pa z višanjem temperature naraste. Podobno se vede magnetizacija sistema, ki pa sprva vztraja pri najvišji vrednosti in nato hitro pada na vrednost nič. Fazni prehod je ožji za manjše mreže. Susceptibilnost in specifična toplota začneta z ničelno vrednostjo in pri kritični temperaturi hitro narasteta nato pa pričneta padati. Maximima obeh količin dosežeta višjo vrednost za manjše mreže, kjer so relativni efekti posameznih domen verjetno izrazitejši.

Za primer ko je prisotno magnetno polje so bile simulacije izvedene le za mreže s številom spinov 64^2 . Temperaturne odvisnosti vseh štirih količin so računane za pet različnih jakosti magnetnega polja. Rezultati so prikazani na sliki 10.6. Tudi tokrat energija s segrevanjem narašča, ko preseže kritično temperaturo. Podobno je z magnetizacijo, ki pa se pri močnem zunanjem magnetnem polju bolj upira temperaturnim učinkom; čeprav še vedno z zamikom sledi razbitju domen, saj z višanjem temperature sistem postaja manj dovzet za zunanje polje. Za višja polja je torej pričakovati premik kritične temperature k višjim vrednostim, kar potrdita še temperaturni odvisnosti susceptibilnosti in specifične topolote, kjer je videti premik položaja maksimumov.



Slika 10.6.: Grafi štirih opazovanih količin v odvisnosti od temperature za sprva uniformno feromagneto spinsko mrežo s populacijo spinov 64^2 pri petih različnih vrednostih jakosti magnetnega polja. Kritična temperatura je označena s črtkano črto. Energija začetnega stanja se vede v vseh primerih polja zelo podobno a ima za močnejše polje manjše vrednosti. Podobno kot pri odsotnosti polja se vede tudi magnetizacija sistema, vendar se temperaturno območje faznega prehoda za močnejša polja razširi, čeprav višanjem temperature sistem vseeno ostaja manj dovzet za zunanje polje. Z naraščanjem jakosti polja pa se k višjim temperaturam zamika tudi položaj maksimuma susceptibilnosti in specifične topote, kar kaže na zamik vrednosti kritične temperature.

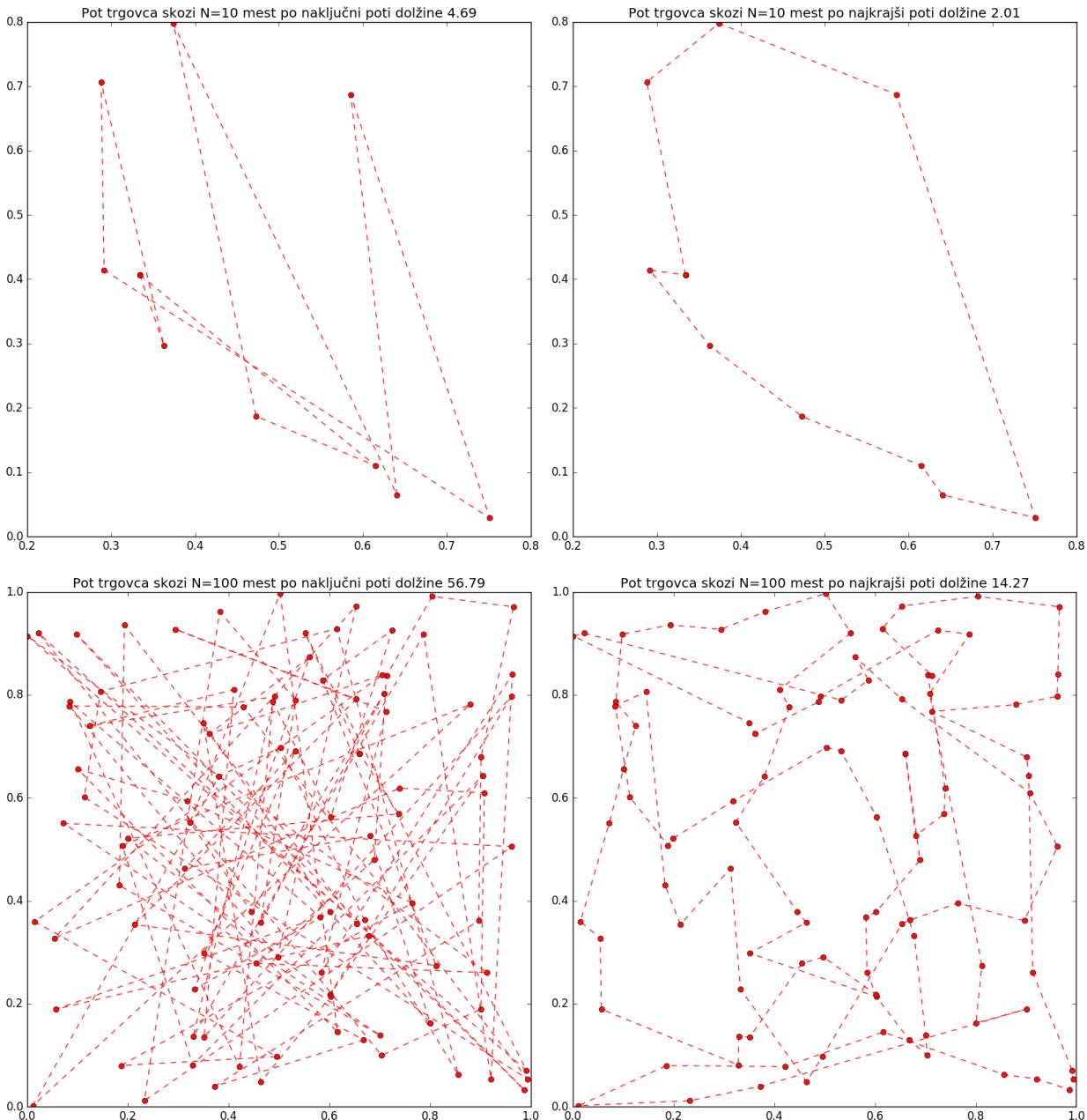
10.4. Problem trgovskega potnika

Za reševanje problema trgovskega potnika z metropolisovim algoritmom je potrebno definirati cenovno funkcijo. Ker gre pri tem problemu za iskanje najkrajše poti, ki povezuje nabor mest, je cenovna funkcija lahko kar najkrajša vsota dolžin med zaporedno povezanimi točkami

$$E = L = \sum_{i=1}^N \sqrt{(x_i - x_{i+1})^2 + (y_i - y_{i+1})^2} .$$

Problem rešujemo tako, da točke najprej naključno povežem in nato naključno spremojam vrstni red obiska mesta. Če se nova pot izkaže za krajšo jo obdržimo, drugače pa jo sprejmemo z določeno verjetnostjo in se tako izognemo lokalnemu minimumu; podobno kot smo to počeli pri prejšnjih nalogah.

Na sliki 10.7 so prikazani izračunani rešitvi za dve različni števili naključno generiranih mest. Očitno je, da je po 10^5 potezah končna pot precej krajša, kot če jo potnik ubere čez mesta po naključju. Za malo število mest je iz slike mogoče sklepati, da je najdena rešitev verjetno optimalna. Že pri povečanju na 100 mest pa se vidi, da metoda ni našla najugodnejše rešitve, saj potnik opravi kar precej dolgih poletov do mest, ki so očitno precej bližja tistim v katerih je predhodno že bil. Za iskanje optimalne rešitve večjega nabora mest, bi bilo potrebno algoritem bolj sofisticirano prilagoditi problemu.



Slika 10.7.: Grafa naključnega potovanja potnika skozi mesta (levo) in grafa rešitev preprsto apliciranega metropolisovega algoritma za dve različni števili naključno generiranih mest. Očitno je da so poti pridobljene z algoritmom krajše. Pri primeru desetih mest je rešitev skoraj optimalna, medtem ko je za primer stotih mest mogoče najti boljšo rešitev.

Literatura

- [1] Spletne strani z gradivom predmeta, ogled 20.10.2017,
<http://predmeti.fmf.uni-lj.si/modelska>

11. Stohastični populacijski modeli

Povzetek

Pri tej vaji ponovno obiščemo populacijske modele, kjer jih tokrat obravnavamo diskretno. S takim stohastičnim pristopom zaradi uporabe diskretnih vrednosti pri spremnjanu števila osebkov v populaciji problem modeliramo realnejše saj ne dopuščamo obstoja polovice osebka. V nalogah obravnavamo v gorbem dva stohastična modela. Eksponentni model odmiranja populacije in model, ki upošteva tudi rojstva članov populacije. Pri izračunih napravimo statistiko časovnih razvojev populacije in opazujemo čas izumrtja, ter preučimo vplive časovnega koraka in velikosti populacije nanj.

11.1. Naloga

1. Napravi statistiko časov izumrtja za preprosti eksponentni model ($\beta = 1.0/\text{enota časa}$) za majhno ($N=25$) in veliko ($N=250$) populacijo. Ali je ta čas odvisen od velikosti časovnega koraka? Primerjaj z modelom, ki vključuje rojstva in smrti ($\beta_r = 4\beta$, $\beta_s = 5\beta$)![1]
2. Za zgornji model poišči matriko prehodov in z njo oceni čas izumrtja. Za preprost eksponentni model izumiranja lahko zapišeš enačbe za verjetnostno porazdelitev v odvisnosti od časa. Kako se efektivni odmik te porazdelitve spreminja s časom?[1]
3. V stohastičnem modelu zajci-lisice določi povprečno življensko dobo sistema, če začnemo v ravnovesnem stanju. Za boljšo primerjavo med rezultati uporabi stacionarno stanje z 200 zajci in 50 lisicami in razmerje rodnost/smrtnost= 5/4 za zajce in obratno za lisice.[1]

Komentar k reševanju

Računanje je izvedeno v Python-u s, kjer je večino časa v uporabi v knjižnici `numpy` vgrajeni generator naključnih števil iz modula `random`. Natančneje, pri reševanju naloge je uporabljen generator števil po Poissonovi porazdelitvi `np.random.poisson()`.

11.2. Statistika časov izumrtja populacije

V kontrastu z prejšnjo vajo iz populacijskih modelov tokrat modeliramo spremembe na diskreten način. Izhajamo iz preprostega eksponentnega modela gibanja populacije

$$\dot{N} = -\beta N \quad \rightarrow \quad N = N_0 e^{-\beta t}, \quad (11.1)$$

kjer imajo vsi člani zaprtega sistema neko povprečno življensko dobo, vendar posamično umirajo naključno z različnimi lastnimi življenskimi dobami. Ravno to naključnost vpeljemo diskretno in vsako smrt simuliramo tako, da v izbranem časovnem koraku za $N_{t+dt} = N_t - \Delta N$ z naključnim generatorjem izvedemo Poissonov žreb iz porazdelitve

$$\Delta N = \frac{\lambda^k}{k!} e^{-\lambda}, \quad (11.2)$$

kjer k stoji za smrti. Časovna enota skriva v povprečni vrednosti populacije $\lambda = N_{(t)}\beta\Delta t$. Žrebe za spremembo začetne populacije izvajamo toliko časa dokler populacija ne izumre. Število časovnih korakov potrebnih za izumrtje shranjujemo v histogram. Za različne vrednosti časovnega koraka Δt dobimo različne porazdelitve časov izumrtja.

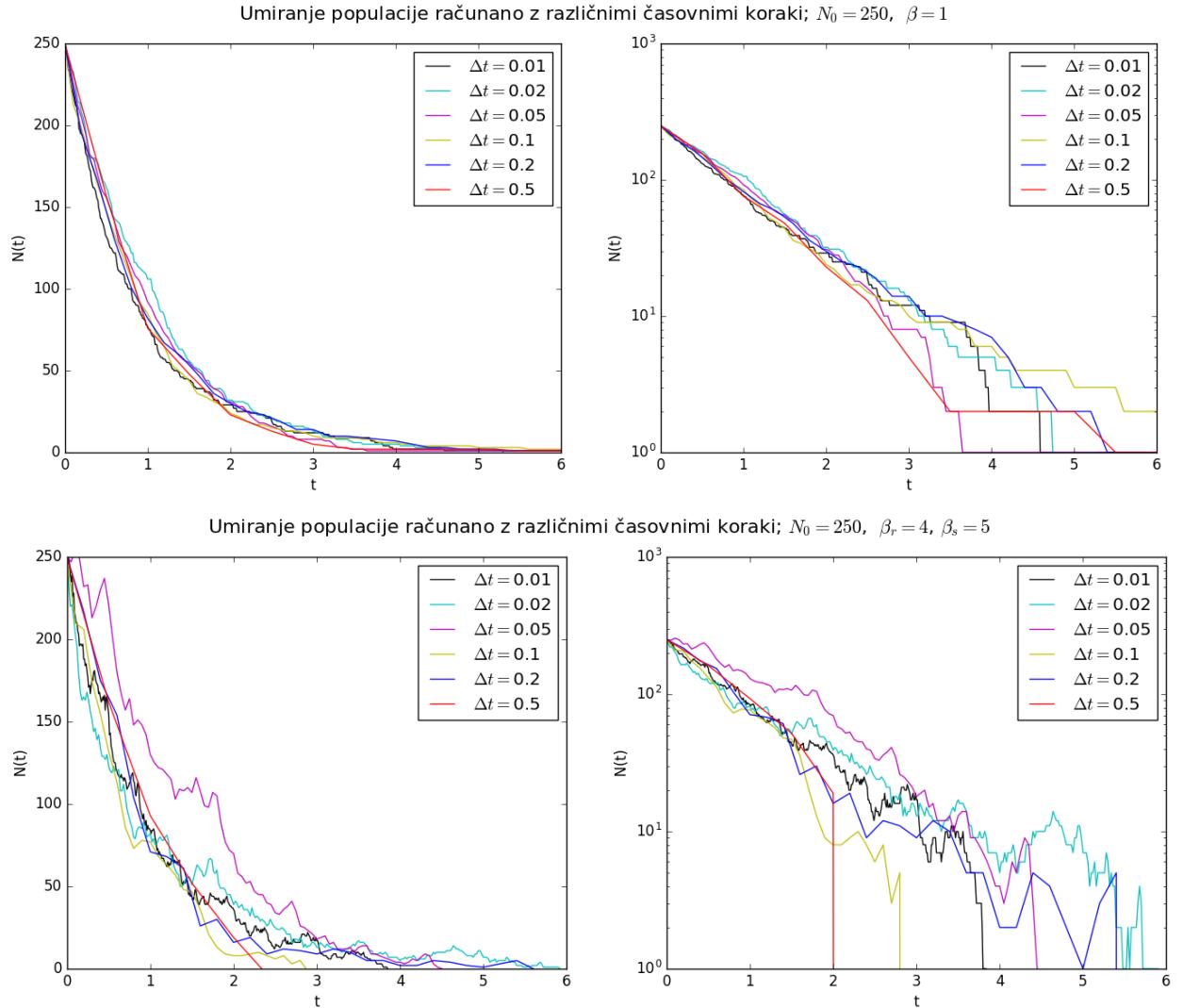
Namesto preprostega modela izumiranja pa lahko obravnavamo model, ki poleg smrti vključuje tudi rojstva. Dinamiko takšnega modela opišemo z

$$\dot{N} = -\beta_s N + \beta_r N, \quad (11.3)$$

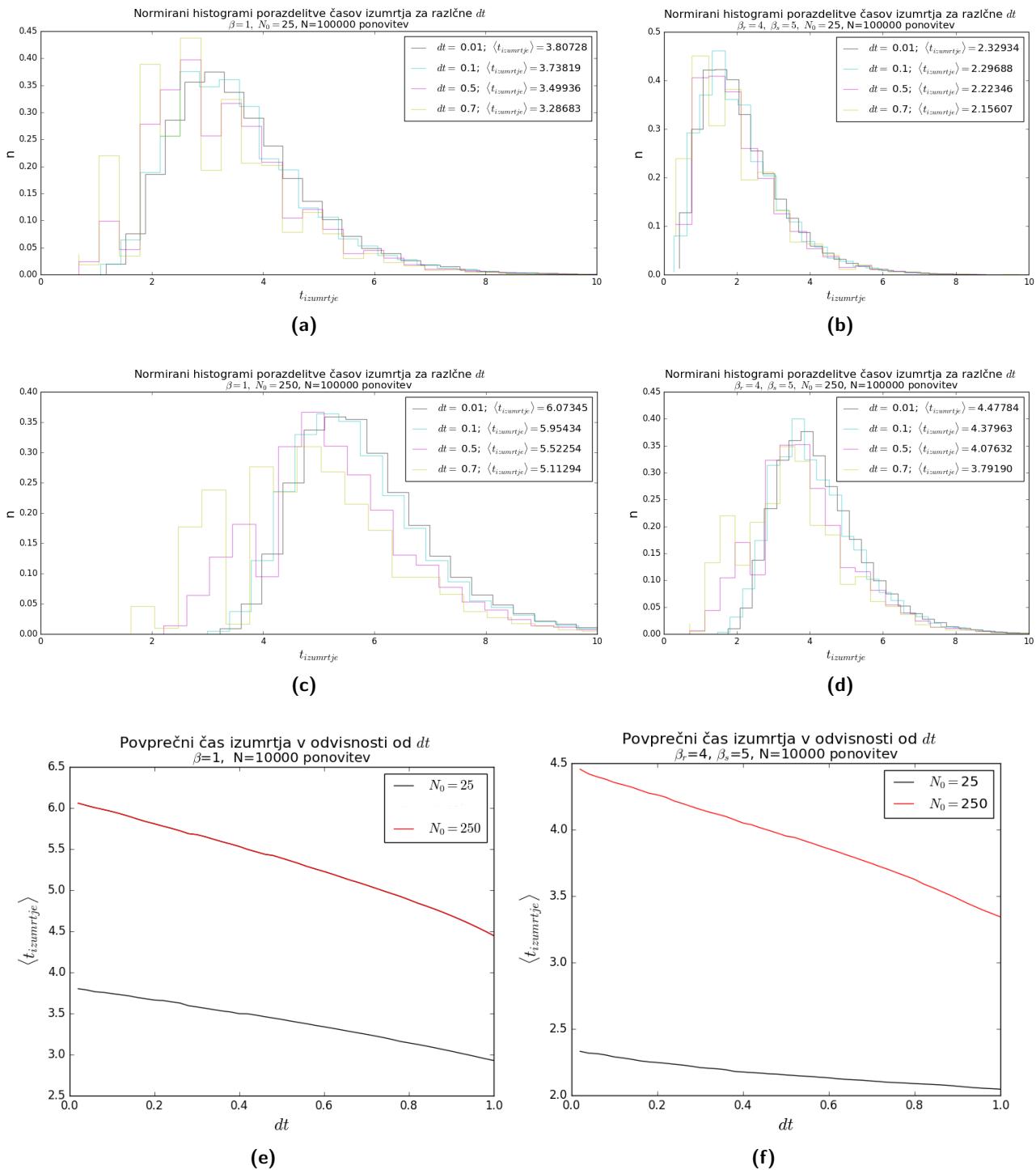
kjer koeficient β_s opisuje smrti in koeficient β_r rojstva.

Rezultati takšne diskretne simulacije spremicanja majhne in velike populacije so prikazani na sliki 11.1. Koeficinetti so nastavljeni na $\beta = 1$, $\beta_s = 5$ ter $\beta_r = 4$. Iz grafov je razvidno, da se potek populacije preprostega modela od kompleksnejšega očitno razlikuje, saj so v slednjem prisotne večje fluktuacije. V obeh modelih pa manjšanje časovega koraka uporabljenega pri računanju nakazuje na krajšanje časa izumrtja. Za izvedbo statistike sem simulacijo za vsak primer izvedel $N = 10^5$ krat in čase izumrtja beležil v histogramih, od koder lahko za vzako izbiro parametrov izračunamo povprečni čas izumrtja. Histogrami in odvisnost povprečnega časa izumrtja za različne dolžine uporabljenih časovnih korakov so prikazane na sliki 11.2, in sicer za oba modela in različno veliki populaciji. Rezultati statistike kažejo, da se po pričakovanju s povečanjem začetne populacije vrh porazdelitve

časov izumrtij premakne k daljšim časom pri obeh modelih. So pa nepričakovano krajši povprečni časi izumrtja za izbrani model rojstva-smrti; to je lahko posledica večjih vrednosti $\beta < \beta_s$ kljub upoštevanju rojstev, saj lahko večje fluktuacije kompleksnejšega modela populacijo hitro poženejo v izumrtje. Statistika tudi potrdi ugotovitev, da manjšanje uporabljenega časovnega koraka pri obeh modelih krajša čas izumrtja.



Slika 11.1.: Grafi časovnega poteka populacije z nekim začetnim številom članov N_0 za preprosti eksponentni model (zgoraj) in kompleksnejši model, ki vključuje tudi rojstva (spodaj). Simulacije so računane z uporabo različnih dolžin časovnih korakov, katerih daljšanje privede do hitrejših izumrtij. Razvidno je tudi, da se modela neglede na dolžino časovnega koraka, razlikujeta v intenzivnosti fluktuacij, ki so večje pri modelu z rojstvi.



Slika 11.2.: Izračuni za preprost model (levo) in model rojstva-smrti (desno) za primer dveh različno velikih začetnih populacij N_0 pri različnih dolžinah uporabljenega časovnega koraka dt . Iz primerjave histogramov istega modela a različnih N_0 se po pričakovanju čas izumrtja za večje začetne populacije premakne k večjim vrednostim. Čas izumrtja populacije pri modelu rojstva-smrti pa je nepričakovano krajiš kot pri preprostem modelu. Iz spodnjih dveh grafov povprečnih časov izumrtij za različne korake dt se vidi, da se čas izumrtja z večanjem koraka v vseh primerih krajša, saj tako večamo moč fluktuacije in lahko še hitreje zbijemo populacijo. Vidimo tudi, da uporaba velikega koraka bolj vpliva na povprečni čas izumrtja ko začnemo z večjo začetno populacijo. Skoraj linearna odvisnost od velikosti časovnega koraka pa se po dopolnitvi eksponentnega modela kaj bistveno ne spremeni.

11.3. Matrika prehodov

Verjetnosti, da se v časovnem intervalu rodi ali umre neko število osebkov sta neodvisni verjetnosti. Njih skupen vpliv na populacijo tako lahko izračunamo po Poissonovi porazdelitvi

$$\Delta N = \frac{\lambda_r^h e^{-\lambda_r}}{h!} \cdot \frac{\lambda_s^k e^{-\lambda_s}}{k!}, \quad (11.4)$$

ker h stoji za rojstva in k za smrti v populaciji. Velja še $\lambda_r = N_{(t)} \beta_r \Delta t$ in $\lambda_s = N_{(t)} \beta_s \Delta t$. videti je torej da vspremeba v populaciji odvisna od njene trenutne velikosti.

Za neko populacijo z začetnim številom stanj N_0 lahko torej definiramo vektor (diskretnih) stanj

$$p(t) = \{p_{N_t=0}, p_{N_t=1}, p_{N_t=2}, \dots, p_{N_t=N_0-1}, p_{N_t=N_0}\}, \quad (11.5)$$

ki je velikosti $N_0 + 1$ in opisuje verjetnost za velikost (diskretno) populacije v nekem časovnem trenutku. Vsak element vektorja poda verjetnost, da je sistem v stanju z neko velikostjo populacije, ki začne z velikostjo N_0 in se nato z vsakim časovnim korakom manjša. Tako se verjetnost, da je po času Δt sistem še vedno velikosti $N_t = N_0$ zmanjša in naraščajo verjetnosti za populacije velikosti $N_t = N_0 - 1, N_0 - 2, N_0 - 3, \dots$ vse do izumrtja. Ob času nič ima ta vektor torej obliko čistega stanja

$$p(t_0) = \{0, 0, \dots, 0, 1\},$$

kar pomeni, da je zasedeno le eno (začetno) stanje saj je $N_t = N_0$. Ob času izumrtja pa ima vektor drugo možno obliko čistega stanja

$$p(t_{izumrtje}) = \{1, 0, \dots, 0, 0\},$$

saj je zasedeno le eno (končno) stanje, ko je $N_t = 0$. V časih med začetnim časom in časom izumrtja je možno, da je sistem v stanju z nekimi vmesnimi velikostmi populacije in takrat imamo vektor mešanih stanj.

Prehode v stanja iz večje populacije v manjšo lahko zapišemo z linearo operacijo kot matriko $M(t)$ prehodov med stanji, ki je dimenzije $N_0 + 1 \times N_0 + 1$. Njeni elementi predstavljajo verjetnosti za prehod iz večjega stanja v manjše stanje

$$M_{i,j} = P(j \rightarrow i) \quad (11.6)$$

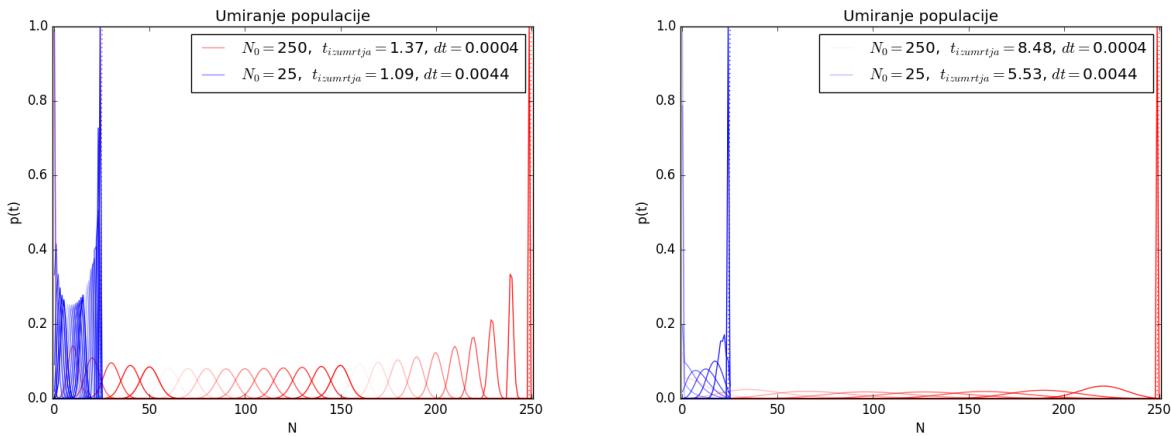
in upošteva enačbo (11.4). Pri eksponentnem modelu je za $\beta N \Delta t \ll 1$ matrika $M(t)$ spodnje trikotna

$$M_{i,j} = \begin{cases} 1 - \lambda_s, & i = j \\ \lambda_s, & i = j - 1 \\ 0, & \text{sicer} \end{cases}, \quad (11.7)$$

saj v sistemu brez rojstev velikost populacije monotono pada. Pri sistemu z rojstvi pa matrika dobri obliko

$$M_{i,j} = \begin{cases} \lambda_r, & i = j + 1 \\ 1 - (\lambda_s + \lambda_r), & i = j \\ \lambda_s, & i = j - 1 \\ 0, & \text{sicer} \end{cases}. \quad (11.8)$$

Da bi dobili časovno evolucijo verjetnosti za velikost populacije v sistemu začnemo iz stanja $p(t_0)$ in vektor iterativno množimo z matriko $M(t)$ dokler ne dosežemo stanja $p(t_{izumrtje})$. Rezultati obeh modelov računani za dve različno veliki populaciji so prikazani na sliki 11.3. Razvidno je, da sistem začne v čistem stanju z delta funkcijo in po korakih preide v mešana stanja, ki imajo neko porazdelitev verjetnosti za zasedenost stanj z populacijami različnih velikosti. Pri obeh modelih je čas izumrtja daljši za primer večje populacije in, kot pričakovano, se izumrtje modela brez rojstev dogodi hitreje.



Slika 11.3.: Izračuni za preprost model (levo) in model rojstva-smrti (desno) za primer dveh različno velikih začetnih populacij N_0 pri različnih dolžinah uporabljenega časovnega koraka dt . Razvidno je, da sistem začne v čistem stanju z delta funkcijo in po korakih preide v mešana stanja, ki imajo neko porazdelitev verjetnosti za zasedenost stanj z populacijami različnih velikosti. Na koncu se porazdelitev spet zoži ko se bliža stanju izumrtja. Iz primerjave različnih modelov je opaziti večjo razmazanost modela rojstva-smrti. Iz primerjave istega modela a različnih N_0 se po pričakovanju čas izumrtja za večje začetne populacije premakne k večjim vrednostim. Čas izumrtja populacije pri modelu rojstva-smrti pa je pričakovano daljši, kot pri preprostem modelu.

Sama porazdelitev pa ima za podana modela v času različne oblike. Pri obeh srednja vrednost leže k ničli in hkrati postaja vse bolj razmazana, dokler ne doseže vrednosti $N_0/2$ in se nato začne zopet nazaj ožati, da doseže končno stanje kot delta funkcija. Vendar ima eksponentni model manšo razmazanost kot model z rojstvi, saj je časovni potek eksponentnega modela kljub naključnim prehodom jasnejši, medtem ko parameter rojstev veča negotovost o časovni evoluciji populacije. Teorija napove časovni potek teh dveh statističnih momentov za populacijo tipa (11.4) kot

$$\bar{\mu} = N_0 \cdot e^{(\beta_r - \beta_s)t} \quad (11.9)$$

in

$$\sigma = N_0 \cdot \frac{\beta_r + \beta_s}{\beta_r - \beta_s} \cdot e^{(\beta_r - \beta_s)t} \cdot [e^{(\beta_r - \beta_s)t} - 1]; \quad (11.10)$$

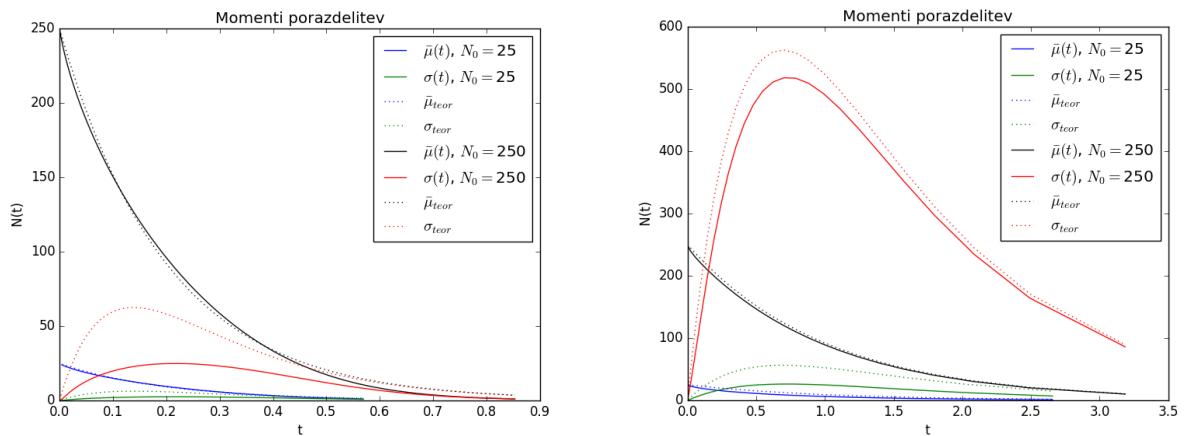
povzeto iz [2]. Spremljanje teh dveh parametrov tekom računanja in primerjava rezultatov s teoretično pričakovanimi je prikazana na sliki 11.4. Videti je, da srednja vrednost med modeloma ne odstopa preveč, le čas izumrtja je za eksponentni model krajši. Sam efektivni odmik pa je za model z rojstvi občutno večji kot pri eksponentnem modelu.

11.4. Stohastični model zajcev in lisic

Pri tej nalogi ponovno preučimo razvoj začetnega stanja populacije zajcev in lisic že obravnavane v četrti vaji. Pri tem se sprememb v populaciji lotimo diskretno in standardni deterministični model zajci-lisice v obliki (4.1) dopolnimo še z naravno smrtno zajcev in lisic. Dinamiko sistema najprej zvezno opišemo kot

$$\begin{aligned} \dot{Z} &= \alpha_r Z - \alpha_s Z - \delta L Z \\ \dot{L} &= \beta_r 4 L - \beta_s L + \gamma L Z, \end{aligned} \quad (11.11)$$

kjer rodnost zajcev nastavimo na $\alpha_r = 5$ in smrtnost $\alpha_s = 4$, ter obratno za lisice $\beta_r = 4$ in $\beta_s = 5$. Uporabimo v diskretno obliko, kjer v izbranem časovnem koraku za spremembo populacije



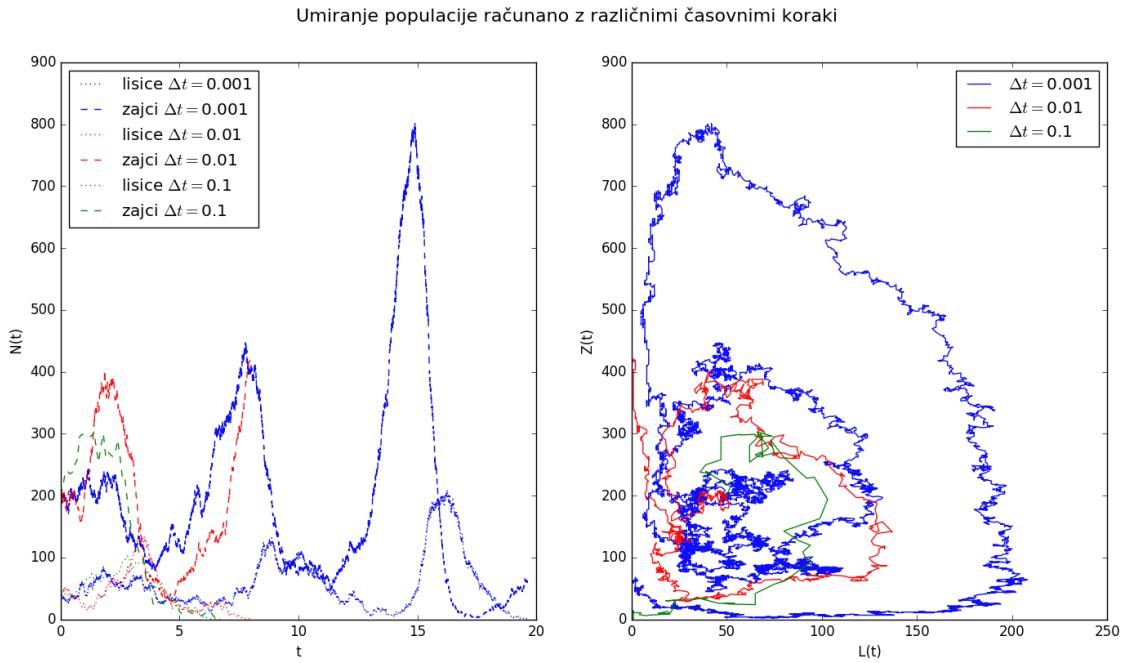
Slika 11.4.: Izračuni za preprost model (levo) in model rojstva-smrti (desno) za primer dveh različno velikih začetnih populacij N_0 pri različnih dolžinah uporabljenega časovnega koraka dt . Videti je, da srednjina vrednost med modeloma ne odstopa preveč, le čas izumrtja je za eksponentni model krajši. Sam efektivni odmik pa je za model z rojstvi občutno večji kot pri eksponentnem modelu. Izračuni obeh modelov tudi po karakteristiki ne odstopajo preveč od teoretičnih pričakovanj za podana β_r in β_s , čeprav se pri modelu z rojstvi časovna evolucija efektivnega odmika bolje ujema z napovedmi.

$N_{t+dt} = N_t - \Delta N$ z naključnim generatorjem izvedemo šest neodvisnih Poissonovih žrebov

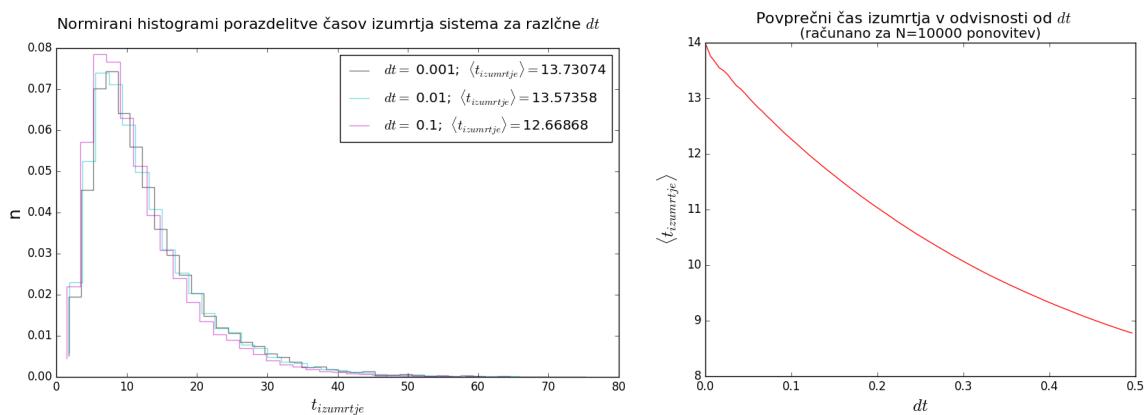
$$\begin{aligned} Z_{t+dt} &= Z_t + P(\alpha_r Z_t dt) - P(\alpha_s Z_t dt) - P(\delta Z_t L_t dt) \\ L_{t+dt} &= L_t + P(\beta_r Z_t dt) - P(\beta_s Z_t dt) + P(\gamma Z_t L_t dt) . \end{aligned} \quad (11.12)$$

V zveznem primeru je možno parametre izbrati tako, da je sistem obstojen in do izumrtja ne pride kot je to lepo videti iz slik 4.1-4.3 četrte vaje. Za začetne velikosti populacije zajcev $Z_0 = 200$ in lisic $L_0 = 50$ in lahko poiščemo ustrezni pogoj za ravnovesno stanje tako, da velja $\gamma = \frac{L_0}{L_0 Z_0} = 0.005$ in $\delta = \frac{Z_0}{L_0 Z_0} = 0.02$.

Izračuni časovnih krivulj z različnimi časovnimi koraki so prikazani na sliki 11.5. Čeprav naj bi bil sistem pri danih parametrih in začetnih pogojih stabilen, ugotovimo, da z diskretizacijo procesa vedno pride do izumrtja vsaj ene izmed populacij in takrat simulacijo ustavimo; natančneje, v kolikor izumrejo najprej zajci, lisice ostanejo brez hrane, če pa najprej izumrejo lisice, se zajci lahko neomejeno namnožijo, saj v enačbah nismo upoštevali omejitve hrane. Kaže, da prav tako kot v prvi nalogi, tudi za ta primer na hitrost izmrtja vpliva izbira dolžine koraka. Če začnemo iz stabilnega stanja računati s prevelikim korakom potem hitro pride do fluktuacijske smrti, medtem ko z manjšanjem časovnega koraka čas izumrtja podaljšamo. O tem se lahko prepričamo tudi iz rezultatov izračuna statistike, ki so prikazani na sliki 11.6. Za izračun povprečno življenske dobe sistema je kot pogoj za izumrtje definiran trenutek, ko ena od populacij pade na nič. Simulacije pa so ponovljene v 10^5 poskusih, za različne časovne korake.



Slika 11.5.: Grafi stohastične časovne dinamike populacije zajcev in lisic (levo) ter fazni diagram (desno) za računanja s tremi različnimi velikostmi koraka dt . Čeprav vedno pride do izumrtja, je za krajše korake moč opaziti neko periodičnost različnih dožin in amplitud, za velike dt pa je fluktuacijska smrt sistema hitra.



Slika 11.6.: Histogram izračunov za sklopljen model populacije lisic in zajcev z rojstvi in smrtmi obhod živalskih vrst za primer populacije v velikosti $Z_0 = 200$ in $L_0 = 50$ pri različnih dolžinah uporabljenega časovnega koraka dt (levo). Na prvi pogled razlike v časih izumrtja za računanje z različnimi dolžinami korakov ni opaziti saj se vse tri izračunane porazdelitve prekrivajo. Iz grafa povprečnih časov izumrtij za različne korake dt (desno) pa je moč videti, da se čas izumrtja z večanjem koraka v vseh primerih krajša, tako kot se je to zgodilo pri prejšnjih dveh modelih.

Literatura

- [1] Spletne strani z gradivom predmeta, ogled 20.10.2017,
<http://predmeti.fmf.uni-lj.si/modelska>
- [2] Verjetnost v fiziki, Simon Širca, DMFA, 2016

Ž

12. Spektralna analiza in filtriranje

Povzetek

Pri tej vaji poskušamo z uporabo Fourierovih transformacij in frekvenčnega filtriranja preučiti nekaj izmerjenih signalov. V prvem delu izračunamo frekvenčni spekter dveh signalov. To storimo najprej brez in nato z uporabo okenskih funkcij; oboje na prebranih intervalih posameznega signalnega vzorca z različnimi dolžinami. V drugem delu naloge na vzorcih različno zašumljenih signalnih potekov ob poznavanju prenosne funkcije izvedemo dekonvolucijo. Pri tem si bolj ali manj uspešno pomagamo tudi z uporabo Wienerjevega filtra. Pri zadnjem delu naloge poskušamo s spoznanimi metodami počistiti razilčno zašumljene Lincolnove podobe podane v `pgm` formatu.

12.1. Naloge

- Signaloma s 512 točkami na datotekah `val2.dat` in `val3.dat`[2] določi frekvenčni spekter. Preskusite različne okenske funkcije. Kako se spremeni spekter, če analiziramo krajše intervale (64, 128,... točk)?[1]
- Signal $u(t)$, ki prihaja v merilno napravo s prenosno funkcijo $r(t)$, se ob dodatku šuma $n(t)$ preoblikuje v

$$c(t) = u(t) * r(t) + n(t) = s(t) + n(t). \quad (12.1)$$

Iz izmerjenega časovnega poteka $c(t)$ bi radi, ob poznavanju odzivne funkcije $r(t)$ in ob nekaterih predpostavkah o šumu $n(t)$, rekonstruirali vpadni signal $u(t)$. N. Wiener je predlagal naslednjo rešitev, ki sledi iz minimizacije napake po metodi najmanjših kvadratov¹. Pred dekonvolucijo je treba transformiranko $C(\omega)$ pomnožiti s filtrom,

$$\Phi(\omega) = \frac{|S(\omega)|^2}{|S(f\omega)|^2 + |N(\omega)|^2}. \quad (12.2)$$

S pomočjo Wienerjevega filtra napravi dekonvolucijo signalov na datotekah `signal0, 1, 2, 3.dat` [2]. Število točk v posameznem signalu je 512. Na zadnjih treh datotekah je signalu primešan šum. Prenosna funkcija je

$$r(t) = \frac{1}{2\tau} \exp(-|t|/\tau), \quad \tau = 16. \quad (12.3)$$

- V novejšem času se vse bolj uveljavljajo metode vrstičnega odbiranja (scanning), pri katerih je slika zabeležena kot 1D signal, dobljen iz točkastega detektorja (n.pr. vrstični elektronski mikroskop). V analitski metodi LA-ICP-MS² ima detektorski sistem končen volumen, zato se pri hitrem odbiranju po vrstici vsebina posameznega piksla prenese še k nadaljnjam točkam. S preprostim razdelčnim modelom pokažemo, da je prenosna funkcija takega sistema eksponentna. Z njo lahko izvedemo dekonvolucijo in razmazanje eksaktno odpravimo. Ko se k razmazanju slike primeša šum, pa lahko motnjo odpravimo le delno, n.pr. z Wienerjevim filtrom.[1]

Poskusiti Lincolnovo podobo, odbrano po stolpcih, ki se v metodi LA-P-MS tradicionalno uporablja za testiranje dekonvolucijskih algoritmov. Slike `lincoln_L30_N00, 10, 30.pgm` najdeš na spletni strani predmeta [2]. V zadnjih dveh datotekah je dodan šum. Prenosna funkcija je

$$r(t) = \frac{1}{\tau} \exp(-t/\tau), \quad \tau = 30, t > 0. \quad (12.4)$$

`pgm` format je P2 različica `ppm` formata, ki shranjuje sivine v tekstovni obliki, na začetku pa ima glavo oblike `P2 x y n`, kjer sta `x` in `y` dimenziji slike, `n = 255` je pa maksimalna vrednost sivine, ki predstavlja belo barvo. Ta format je priporočljiv tudi za izhodne slike, saj ga je enostavno pretvoriti v kateri koli drug format in nazaj s programom `convert` iz paketa `imagemagick`:

```
> converta.pgm>png  
> converta.png -compressnonea.pgm3
```

S tem se izognemo klicanju knjižnic za kodiranje slikovnih formatov. Pazi, po filtriranju se lahko pojavijo vrednosti, ki ne padajo v območje [0,255]. Take presežne vrednosti je treba popraviti pred zapisovanjem izhodne slike.

¹gl. *Numerical Recipes*, pogl. 12.6

²Krajše za ang. Laser ablation with inductively coupled plasma and mass spectrometry.

³Brez –`compressnone` argumenta namesto tekstovnega zapiše binarni format.

Komentar k reševanju

Nalogo sem reševal v programskem okolju »Python«. Pri tem sem se poslužil predvsem knjižnic `scipy` in `numpy`. Podatke sem uvozil s pomočjo funkcije `numpy.loadtxt`, ki elemente stolpca tekstovne datoteke shrani kot elemente vektorja. Računal sem z vgrajnimi funkcijami iz paketa `fftpack` kot so `fft` in `ifft` ter z okenskimi funkcijami iz paketa `scipy.signal`, `signal.gaussian`, `signal.bartlett`, `signal.hann`, `signal.chebwin` in `signal.tukey`.

12.2. Določitev frekvenčnega spektra signalov

12.2.1. Diskretna Fourierova transformacija

Pri numeričnem izračunavanju Fourierove transformacije (FT) neke funkcije f :

$$H(\omega) = \int_{-\infty}^{\infty} h(t) \exp(i2\pi\omega t) dt \quad (12.5)$$

ter njene inverzne transformiranke

$$h(t) = \int_{-\infty}^{\infty} H(\omega) \exp(i2\pi\omega t) df \quad (12.6)$$

funkcijo $h(t)$ običajno diskretno predstavimo s tabelico vrednosti:

$$h_k = h(t_k), \quad t_k = k \cdot \Delta, \quad \text{kjer je} \quad k = 0, 1, 2, 3, \dots N - 1.$$

Tako interval na katerem računamo, razdelimo z N točkami, v katerih nam je vrednost $h(t)$ znana. Ponavadi so razdalje med točkami v katerih računamo enake in pravimo, da smo funkcijo vzorčili z vzorčno gostoto (ali frekvenco) $f_s = 1/\Delta$.

Transformirana funkcija nosi informacije o nekem pojavu, ki jih v praksi želimo imeti; v primeru te vaje npr. izvajamo frekvenčno analizo dveh signalov. Ker je na voljo le tabelo diskretno izmerjenih vrednosti funkcije v določenih točkah t_k , smo v diskretizaciji praktično prisiljeni. Posledično je tudi od tu določena Fourierova transformiranka v frekvenčnem prostoru podana z N diskretnimi vrednostmi, ki ohranjajo le zajeto informacijo. Vsak element diskretne transformiranke izračunamo po formuli:

$$H_n = \sum_{k=0}^{N-1} h(t) \exp(2\pi i kn/N), \quad n = -N/2, \dots N/2. \quad (12.7)$$

Z obratno transformacijo lahko iz H_n rekonstruiramo tudi h_k :

$$h_k = \frac{1}{N} \sum_{n=0}^{N-1} H_n \exp(-2\pi i kn/N), \quad n = -N/2, \dots N/2. \quad (12.8)$$

Za tako definiran vzorec obstaja naravna meja frekvenčnega spektra, imenovana *Nyquistova frekvenca* $\omega_c = 1/2\Delta$. Harmonični val s to frekvenco ima v vzorčni gostoti ravno dva vzorca v periodi. Če ima torej funkcija $h(t)$ frekvenčni spekter omejen na interval $[-\omega_c, \omega_c]$, potem ji klub diskretizaciji z izbranim vzorčenjem nismo odvzeli nič informacije, v kolikor pa se spekter razteza preko teh frekvenc, pride do t.i. pojava *potujitve oz. aliasing-a* signala. Pri slednjem se zunanjji del frekvenčnega spektra preslika v interval preko njegovih mej v točkah $\pm\omega_c$ in tako popači transformirako $H(\omega)$. Samo

potujitev lahko tudi izkoristimo in zaradi enakosti $H_{-n} = H_{N-n}$ indeks v enačbah (12.7) in (12.8) pišemo kot $n = 0, 1, 2, 3, \dots N - 1$. V tem primeru posebna vrednost pri $n = 0$ ustreza frekvenci nič, vrednost pri $n = N/2$ pa ustreza obema mejama $\pm\omega_c$.

Posebej uporabna je še informacija o celotni moči nekega zajetega signala. Ta je neodvisna od reprezentacije in je definirana s Parsevalovo enačbo

$$\sum_{n=0}^{N-1} |h_k|^2 = \frac{1}{N} \sum_{n=0}^{N-1} |H_n|^2 . \quad (12.9)$$

Ko nas bolj zanima, koliko moči je vsebovane v frekvenčni komponenti med ω in $\omega + d\omega$, lahko definiramo enostransko spektralno gostoto moči (PSD⁴)

$$P_n = |H_n|^2 + |H_{N-n}|^2 , \quad (12.10)$$

kjer sta za realne signale h_k prispevka enaka, tako da takrat velja $P_n = 2|H_n|^2$. [3]

12.2.2. Hitra Fourierova transformacija

Diskretno Furierovo transformacijo (DFT⁵) (12.7) pa lahko zapišemo kompaktneje kot

$$H_n = \sum_{k=0}^{N-1} W^{nk} h_k , \quad W = \exp(2\pi i/N) . \quad (12.11)$$

Časovna zahtevnost takšnega pristopa je N^2 , kar je precej potratno. Postopek računanja lahko bistveno pospešimo, če DFT najprej razcepimo. Predpostavimo, da je N deljiv z m . Tedaj lahko vsoto (12.11) razdelimo na m delov, ki vsaka teče po elementih vektorja h_k . Optimalni primer je $N = 2^k$, kjer na vsakem koraku delimo vektor na sode in lihe indekse ter dobimo razcep

$$H_n = H_n^{sod} + W^n H_n^{lih} . \quad (12.12)$$

Gornjo relacijo lahko uporabljamо rekurzivno in če je N enak potenci števila 2, lahko rekurzijo razdrobimo do nizov, ki imajo samo še en člen, zanj pa je transformacija kar identiteta. Za obrat pri eni vrednosti frekvence (pri danem m) je na vsakem koraku rekurzije tako potrebno le eno množenje s potenco W, število vseh korakov pa je $\log_2(N)$. Skupna časovna zahtevnost je torej le še $N \log_2(N)$.

Da ne iščemo pripadnikov niza po vsej tabeli, si podatke še preuredimo. Mogoče je pokazati, da je v prvotni tabeli treba med seboj zamenjati tiste podatke, katerih vrstna števila v binarnem zapisu so obrnjena in v novem redu jemljemo člene kar po vrsti. Tudi potenc W ne izražamo vedno znova s sinusom in kosinusom, pač pa jih računamo z rekurzijo. Tak ali podoben postopek je osnova vseh algoritmov hitre Fourierove transformacije (FFT⁶). [3]

12.2.3. Frekvenčni spekter signalov val2.dat in val3.dat

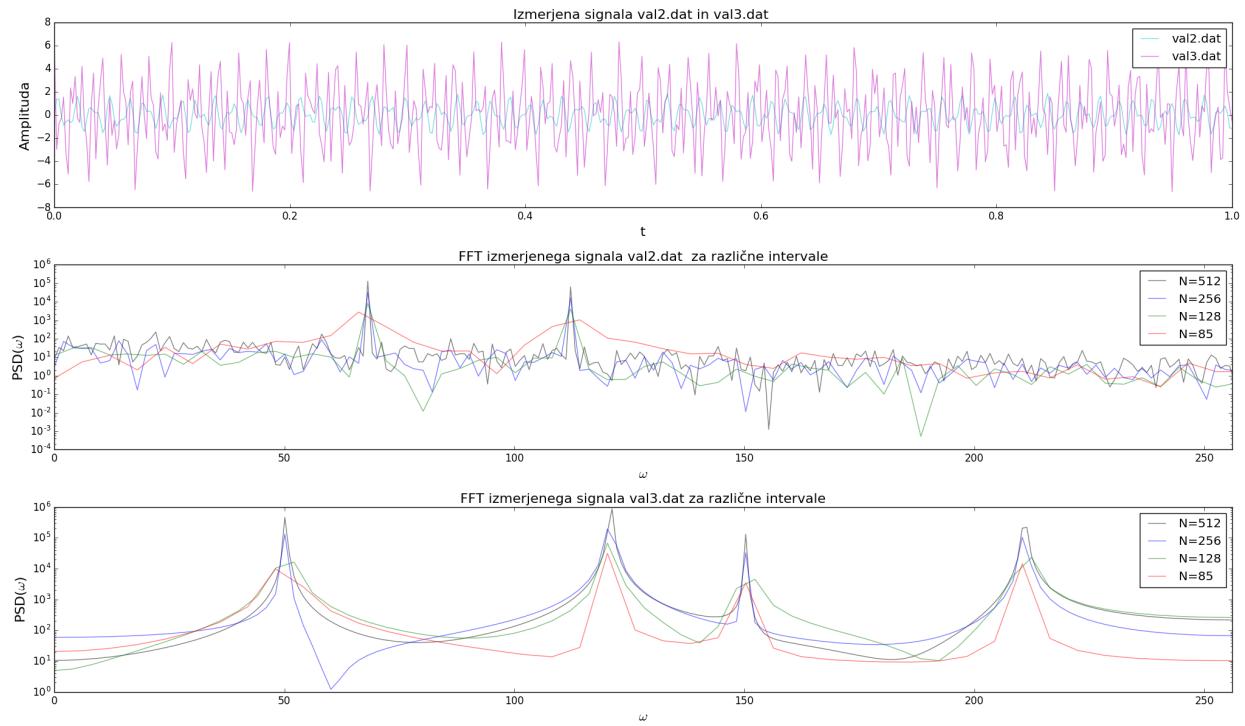
Za oba podana signala vzorčena s 512 točkami bi želeli z uporabo FFT določiti frekvenčni spekter. To storimo z analiziranjem celotnega posamičnega signala in pa tudi krajsih intervalov signala. Pričakovati je, da daljši intervali signala vsebujejo več informacije o vseh prisotnih valovnih nihanjih, kar tudi potrdijo razlike v spektrih za intervale v 64, 128, 256 ter 512 točkah, ki so prikazani na sliki 12.1. Originalna signala podata največje amplitude nosilnih frekvenc ter ohranita tudi več naloženih frekvenc v signalu prisotnega šuma. Z manjšanjem intervala analiziranega vzorca se glavni vrhovi

⁴Krajše za ang. onesided Power Spectral Density.

⁵Krajše za ang. Discrete Fourier Transformation.

⁶Krajše za ang. Fast Fourier Transformation.

spektra ohranijo, čeprav v manjši amplitudi in večji razmazanosti. Očitno se spremeni tudi ostrina deleža frekvenc v prostoru, ki ga pokriva šum. Ta se pri vzorcu `val2.dat` za intervala 64 in 128 še posebej zgladi, saj je prisotne manj informacije za ostro rekonstrukcijo bogatega spektra šuma.

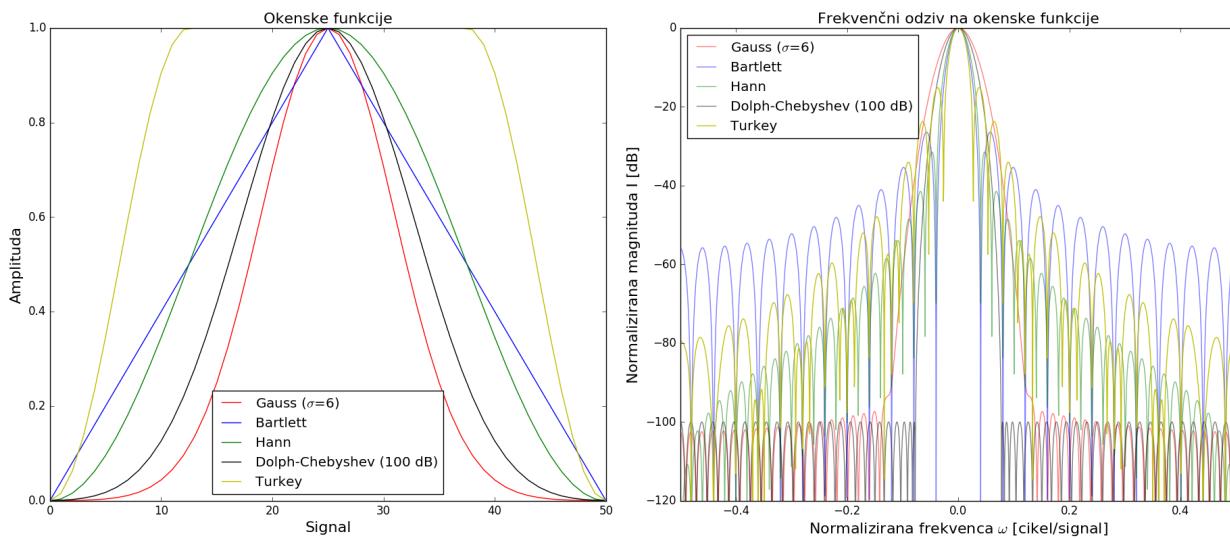


Slika 12.1.: Izmerjena signala `val2.dat` in `val3.dat` prikazana v celoti (zgoraj) in njune fourierove transformacije prikazane za štiri različna vzorčenja. Iz signalov samih ni opaziti lepe periodičnosti, v spektru signalov pa je videti, da so nekatere frekvence v signalu precej bolj pomembne, saj imajo večjo amplitudo. Z manjšanjem vzorca se v frekvenčnem prostoru te amplitude dobro ohranijo, vendar se začne izgubljati ostrina spektra in ostali detajli.

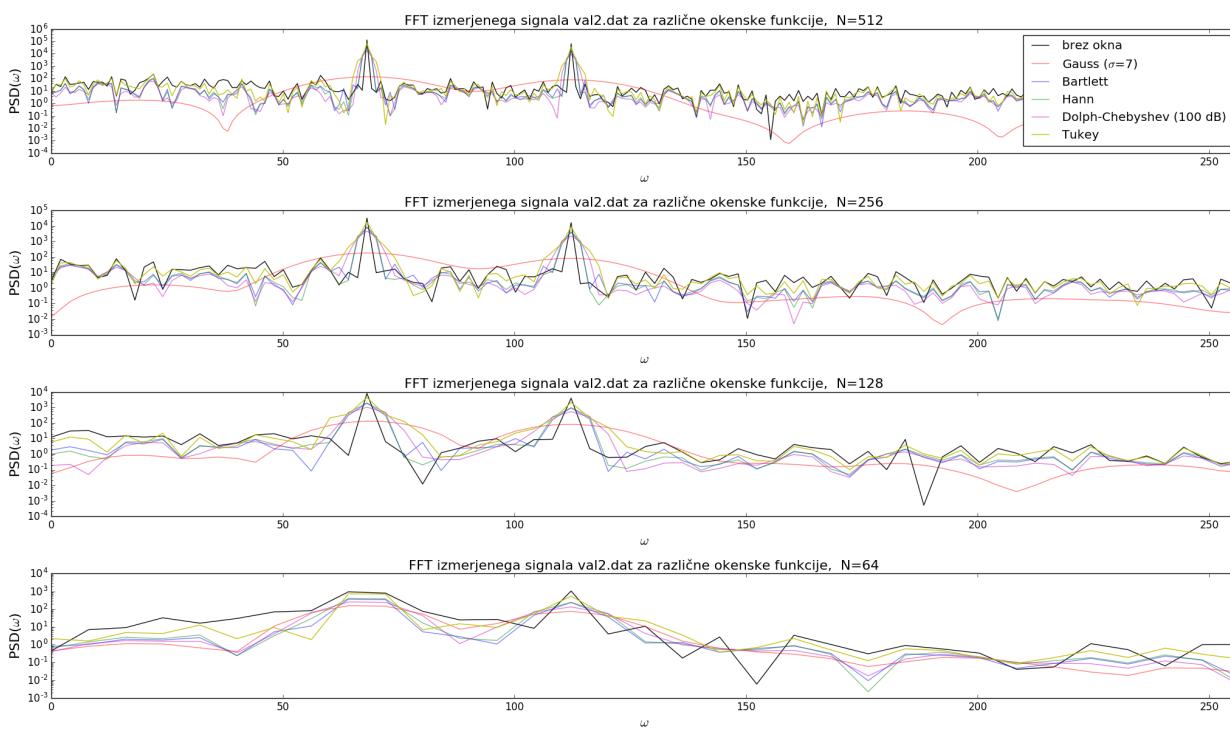
Za potrebe izločitve čim večjega dela šuma ob ohranitvi amplitude nosilnega signala lahko uporabimo okenske funkcije, s katerimi pomnožimo analizirani interval signalnega vzorca. Nabor nekaterih funkcij razvitih za različne potrebe spektralne analize je skupaj z njihovimi spektri izračunanimi s FFT prikazan na sliki 12.2. Čeprav so lahko razlike med oblikami oken v časovnem prostoru majhne, je njihov učinek v frekvenčnem prostoru precej bolj raznolik.

Ob pomnožitvi prej analiziranih intervalov obeh signalov z oknom pa dobimo spektre, ki si jih lahko ogledamo na sliki 12.3 za signal `val2.dat` in sliki 12.4 za signal `val3.dat`. Vsaka od okenskih funkcij ima drugačen učinek na šum in signal, a načeloma vse okenske funkcije ohranijo glavne vrhove nosilnega signala in zmanjšajo šum v njihovi okolini. Še najbolje se pri tem odrežeta⁷ Hannova funkcija in Dolph-Čebišova funkcija, ki dobro zadušita šum v bližini glavnih vrhov, kjub temu pa same špice nekoliko zaoblita; kar se z krajšanjem intervala še bolj izrazi. Najslabše se odreže Gaussovo okno. Slednje skoraj neodvisno od dolžine analiziranega signala povsem zamaže okolico glavnih vrhov. Glede na opaženo je v primeru vzorca `val2.dat` efekt vseh oken nefektiven, saj poleg zaoblenja glavnih vrhov nosilnega signala, še premalo zadušijo šum, da bi bil končni rezultat ostrejša določenost spektra. Pri vzorcu `val3.dat` pa večina oken zelo dobro poreže okolico glavnih vrhov, čeprav se njihova ostrina ravno tako zmanjša. Pri obeh vzorcih se tudi lepo opazi kako manjšanje intervala signala vpliva na povečanje širine vrhov nosilnih frekvenc.

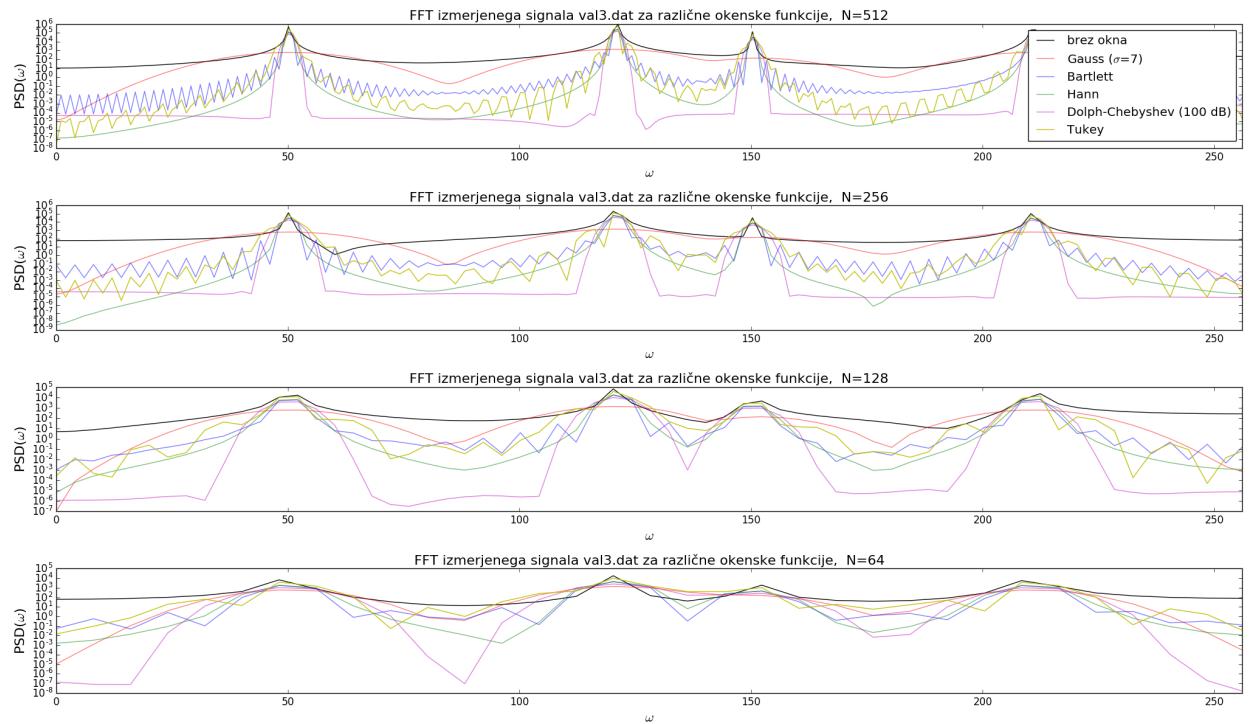
⁷Pun intendend.



Slika 12.2.: Primer okenskih funkcij (levo) in njihovih FFT prikazanih v frekvenčnem prostoru (desno).



Slika 12.3.: Grafi fourierovih transformacij štirih različnih vzorčnih dolžin intervalov signala `val2.dat`. Efekt vseh oken je za ta signal precej neučinkovit, saj premalo zadušijo šum. Manjšanje analiziranega intervala vzorca ima enak efekt pri vseh primerih, pri čemer se izgubi ostrina spektra in ostali detajli. Posebno slabo se odreže uporaba Gausovega okna.



Slika 12.4.: Grafi fourierovih transformacij štirih različnih vzorčnih dolžin intervalov signala val3.dat. Tokrat večina oken zelo dobro poreže okolico glavnih vrhov, čeprav se njih ostrina zmanjša; le uporaba Gausovega okna je še vedno neučinkovita. Pri rezultatih okenskih funkcij Bartlet-a in Tukey-a sta vidna še lepa primera spektralnega puščanja, ki se kaže kot žagasto naložen vzorec ob straneh glavnih vrhov. Manjšanje analiziranega intervala vzorca ima enak efekt kot pri prejšnjem signalu.

12.3. Dekonvolucija zašumljenih signalov

Pri tem delu vaje želimo s pomočjo Wienerjevga filtra (12.2) rekonstruirati meritev $s(t)$; oziroma v merilno napravo z odzivno funkcijo $r(t)$ vpadi signal $u(t)$, pri tem pa imamo podane štiri izmerjene časovne poteke $c(t)$. Slednje razberemo iz datotek **signal0.dat**, **signal1.dat**, **signal2.dat** in **signal3.dat**. V prvi od naštetih datotek se nahaja čisti signal, ostale tri pa imajo signalu dodan šum $n(t)$, kot je to razvidno na sliki 12.5.

Da bi z dekonvolucijo dobili iskani $u(t) = \mathcal{F}^{-1}[U(\omega)]$ se moramo najprej dokopati do vrednosti $U(\omega)$. Do nje lahko na preprost način pridemo le za primer nezašumljenega signala preko zveze za v frekvenčni prostor transformirano konvolucijo

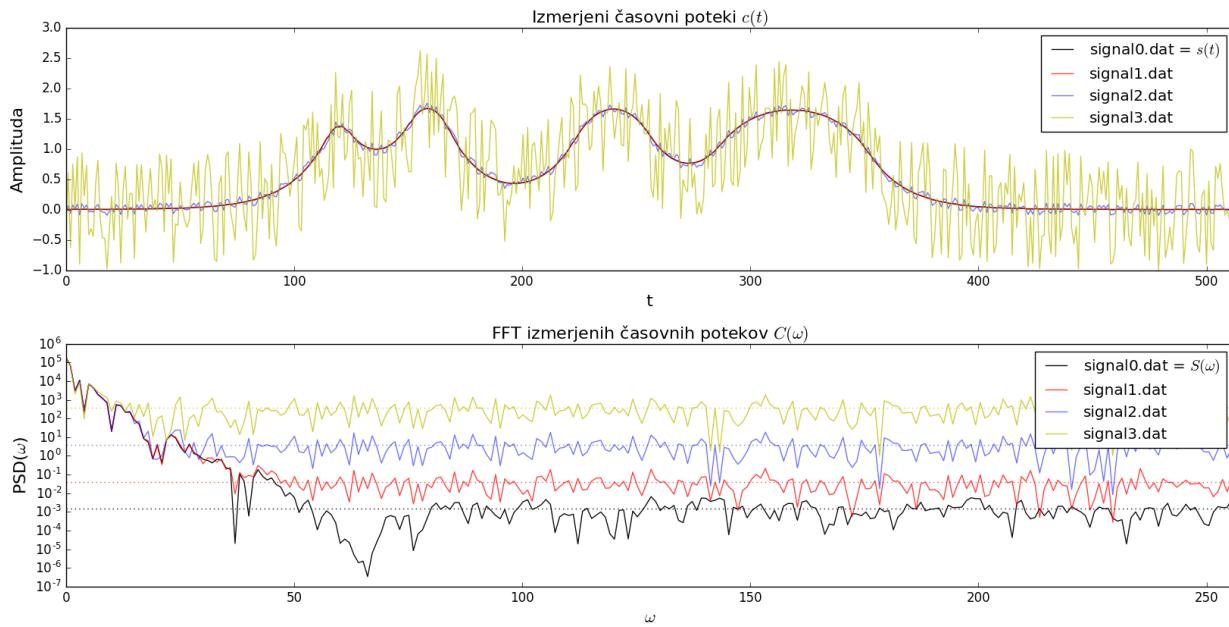
$$U(\omega) = \frac{C(\omega)}{R(\omega)} .$$

Funkcija $R(\omega)$ je seveda FTT prenosne funkcije (12.4) in ima obliko

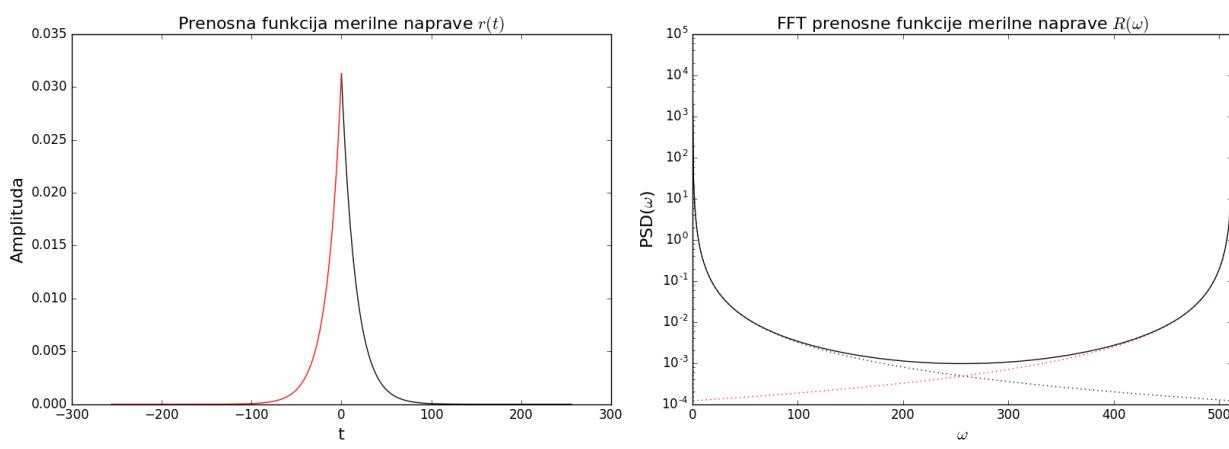
$$R(\omega) = \frac{2(1/\tau)}{(1/\tau)^2 + \omega^2} ,$$

ker pa je prenosna funkcija dvostranska, je potrebno upoštevati še periodičnost prostora, saj naš signal nikoli ne pride do negativnih vrednosti časa. Če hočemo $R(\omega)$ v frekvenčnem prostoru pravilno predstaviti, je potrebo izračunati tudi FFT za negativni del ter ga nato preslikati v desni del frekvenčnega definicijskega območja; kot je to prikazano na sliki 12.6. Z $R(\omega)$ torej delimo spekter časovnega poteka, nato pa spekter z inverzno fourierjevo transformacijo (IFTT⁸) preslikamo nazaj v časovni prostor. Pri ostalih vzorcih je zaradi prisotnosti šuma potrebno opraviti več dela. Pred

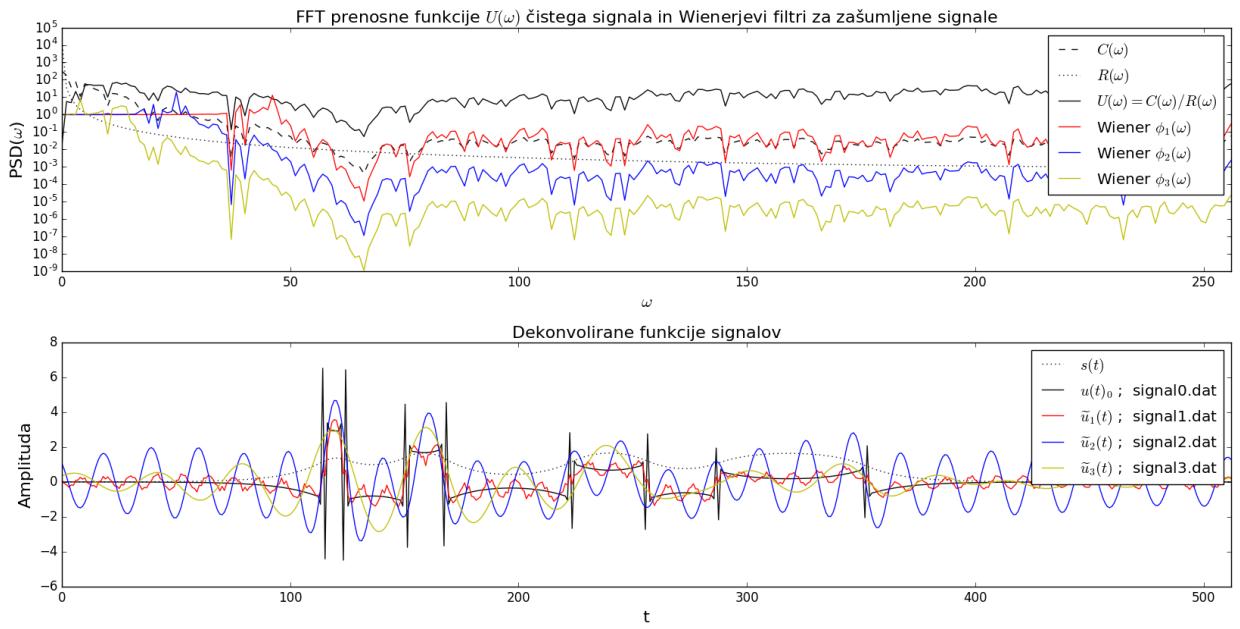
⁸Krajše za ang. Inverse Fast Fourier Transformation.



Slika 12.5.: V 512 točkah izmerjeni časovni potezi štirih signalov (zgoraj) in njihove fourierove transformacije (spodaj). Že iz grafov časovne odvisnosti je opaziti različno zašumljenost. Povsem brez šuma je le signal **signal0.dat**, ki se skoraj prekriva z le malenkost zašumljenim signalom **signal1.dat**. Več šuma je naloženega na signala **signal2.dat** in **signal3.dat**. V močnostnem spektru signalov je moč razbrati, da se intenzitete šuma stopnjujejo po dva velikostna reda. Njiova povprečna vrednost je na grafu močnostnega spektra prikazana z črtanimi črtami.



Slika 12.6.: Prikaz prenosne funkcije računane v 512 točkah (levo) in njene fourierove transformiranke (desno). Izračunati je potrebno tudi spekter za negativne vrednosti in ga nato preslikati v desni del frekvenčnega definicijskega območja ter oba prispevka sešteji. Vsak prispevek je prikazan s črtano črto, njuna končna vsota pa s polno; le slednjo lahko uporabimo za izračun $U(\omega)$.



Slika 12.7.: Prikaz izračuna funkcije $U(\omega)$ (zgoraj) in Wienerjevih filtrov za posamezni zašumljen vzorec. Inverzne fourierove transformiranke $u(t)$ in $\tilde{u}(t)$ si pridobljene z dekonvolucijo nezašumljenega vzorca `signal0.dat` in zašumljenih vzorcev `signal1, 2, 3.dat` (spodaj). S črtkano črto je prikazan nezašumljen izmerjeni časovni potek $c(t)$. Originalni signal $u(t)$ izmerjen z merilno napravo s prenosno funkcijo $r(t)$ je bil torej sestavljen iz štirih kvadratnih pulzov padajoče amplitude in naraščajoče širine. Že majhna prisotnost šuma zelo oteži dekonvolucijo signala $u(t)$, z njegovim večanjem pa slednji postaja nerazpoznaven.

dekonalculacijo moramo transformiranko $C(\omega)$ pomnožiti s filtrom (12.2), ki ga lahko prepišemo v obliko

$$\Phi(\omega) = \frac{|C(\omega)|^2 - |N(\omega)|^2}{|C(\omega)|^2},$$

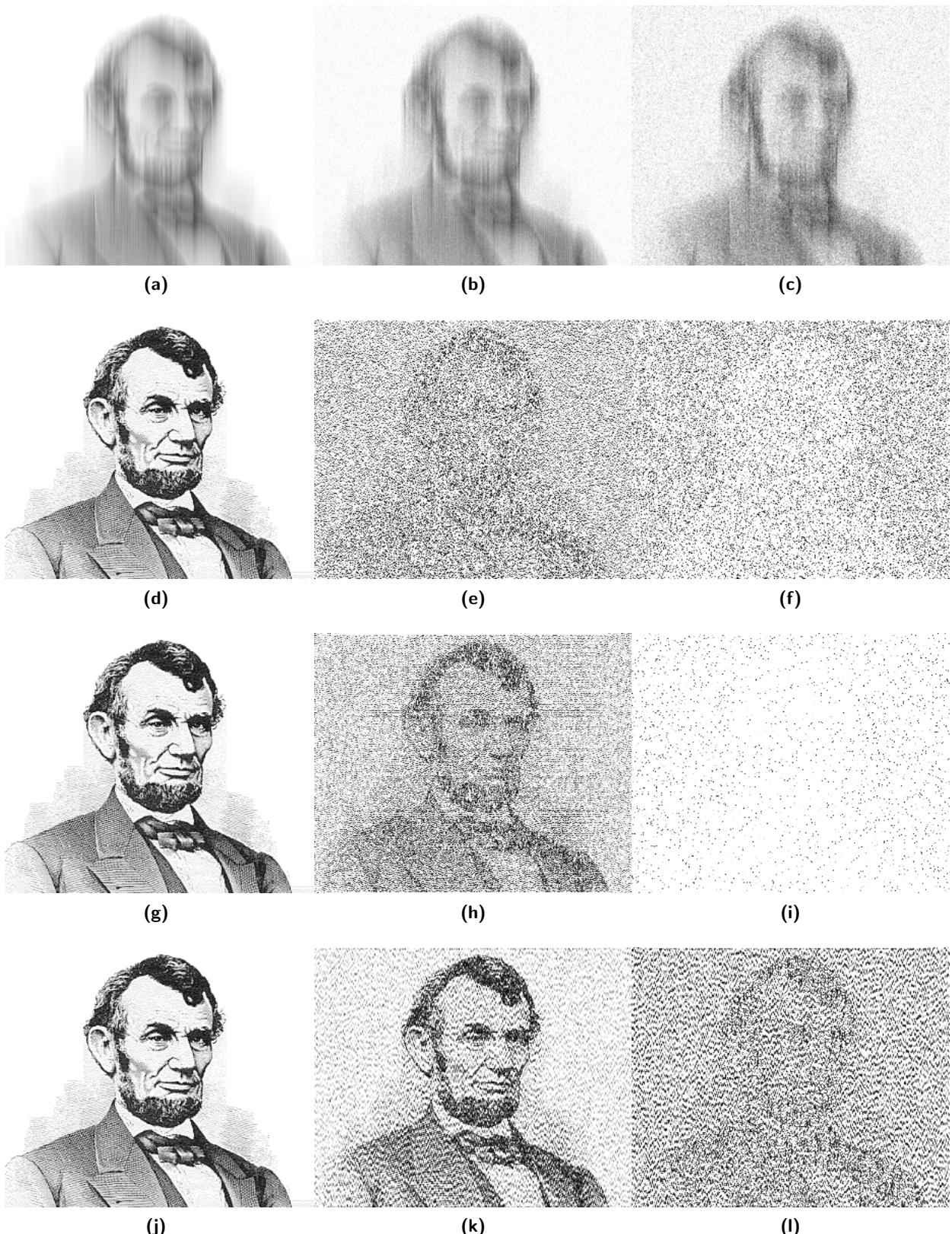
da lahko izračunamo približek

$$\tilde{U} = \Phi(\omega) \frac{C(\omega)}{R(\omega)}.$$

Podan je le $C(\omega)$, medtem, ko je povprečen prispevek šuma $N(\omega)$ potrebno oceniti. Pri tem si pomagamo s spektrom zašumljenih časovnih potekov, kot je prikazano na spodnjem grafu slike 12.5. S takšno uporabo Wienerjevega filtra lahko končno izvrednotimo še $\tilde{u}(t) = \mathcal{F}^{-1}[\tilde{U}(\omega)]$. Tako dobljeni Wienerjevi filtri in dokonalculirani signali $u_i(t)$ so prikazani na sliki ??.

12.4. Čiščenje Lincolnove slike

Pri čiščenju Lincolnove slike je bilo najprej potrebno ugotoviti v kateri smeri je bila konvolucija s podano enosmerno prenosno funkcijo izvedena. Že na prvi pogled je opaziti, da je pisanje najverjetnejše potekalo v smeri stolpcov. To je potrdil tudi poskus dekonvolucije nezašumljene slike v omenjeni smeri, katerega rezultat je bila popolnoma ostra slika. Nadaljnje je bila izvedena še dekonvolucija zašumljenih slik in sicer s tremi pristopi: direktna dekonvolucija, dekonvolucija z uporabo Wienerjevega filtra, in dekonvolucija s prepisanim visokofrekvenčnim šumom s podatki iz nezašumljenega vzorca. Rezultati dekonvolucije za vse tri podane vzorce so prikazani na sliki 12.8. Videti je, da že minimalna prisotnost šuma zelo popači sliko, z večanjem šuma pa rekonstrukcija postane skoraj nemogoča. Uporaba Wienerjevega filtra pri izločanju šuma le malo pomaga, prepis s nezašumljeno sliko pa malce bolj, vendar v praksi dodatni nezašumljeni signali, s katerimi se da na tak način pogoljufati, niso na voljo.



Slika 12.8.: Originalni vzorci so prikazani na slikah (a-c). V stolpcih pod njimi so prikazani rezultati različnih primerov dekonvolucij slike z vrha posameznega stolpca. Rezultati direktne dekonvolucije so na slikah (d-f) (ter (g) in (j)), kjer je opaziti, da šum onemogoča popolno rekonstrukcijo slike. Ob uporabi Wienerjevega filtra dobimo slike (h) in (i). Takle filter le nekoliko pomaga pri dekonvoluciji, vendar le če je šum dovolj majhen. Pod (k) in (l) sta prikazana še primera dekonvolucije s prepisom frekvenc intervala $\omega[50, 128]$ zašumljenih slik s fekvencami iz istega območja intervala nezašumljenega signala slike (a); s takšno goljufijo visoke frekvence šuma porežemo, zato je rezultat dekonvolucije precej bolj razpoznaven.

Literatura

- [1] Spletne strani z gradivom predmeta, ogled 20.10.2017,
<http://predmeti.fmf.uni-lj.si/modelska>
- [2] Spletne strani z gradivom predmeta, ogled 20.10.2017,
<http://predmeti.fmf.uni-lj.si/modelska/podatki/>
- [3] Spletne strani predmeta "Matematično fizikalni praktikum", ogled 20.10.2017,
<http://predmeti.fmf.uni-lj.si/mafirprak>

13. Metoda maksimalne entropije in linear na napoved

Povzetek

Pri tej vaji poskušamo izračunati močnostni spekter vzorca signala in napovedati njegovo obnašanje v prihidnosti z uporabo metode maksimalne entropije. Pri tem preučimo odvisnost rezultatov glede na izbrano število polov in glede na gostoto vzorčenja signala. Rezultate izračunov močnostnih spektrov po metodi maksimalne entropije primerjamo tudi z rezultati dobljenimi s pomočjo algoritma hitre fourierove transformacije ob uporabi okenskih funkcij. Preverimo tudi kako na učinkovitost spoznane metode vpliva šum.

13.1. Naloga

- Določi frekvenčni spekter signalov iz datotek `val2.dat`, `val3.dat` in `co2.dat` z metodo maksimalne entropije. V slednji datoteki s koncentracijo CO_2 v zraku upoštevaj splošen (linearen) letni trend. Preizkusи delovanje metode v odvisnosti od števila polov in od gostote prikaza. Pri CO_2 si oglej še njihovo lego. Primerjaj natančnost metode z rezultati, ki jih da FFT ob uporabi filtrov. Sestavi tudi testni signal iz vsote sinusnih členov z bližnjimi frekvencami ter razišči ločljivost metode.[1][2]
- Koeficiente metode maksimalne entropije na signalih iz prve naloge uporabi za linearno napoved: koeficiente določi iz prve polovice zapisa signala, drugo polovico pa uporabi za primerjavo z napovedjo. Priporočljivo je majhno število polov. Ugotovi, kako vpliva na napoved šum! Podobno preuči datoteko `borza.dat` in podatke za lunine efemeride v `luna.dat`, oziroma sončni cikel `Wolf_number.dat`. Preveri, ali so vsi koreni preslikavne funkcije znotraj enotnega kroga, in jih po potrebi preslikaj vanj![1][2]

Komentar k reševanju

Priporočene uporabne funkcije iz *Numerical Recipes* so: `memcof`, `evlmem`, `zroots`, `fixrts` in `predic`. Nalogo sem reševal v programskem okolju »Python«. Pri tem sem se poslužil predvsem knjižnic `scipy` in `numpy`. Podatke sem uvozil s pomočjo funkcije `numpy.loadtxt`, ki elemente stolpca tekstopisne datoteke shrani kot elemente vektorja. Spektre FFT sem računal z vgrajeno funkcijo `fft` iz paketa `fftpack` ter z okenskima funkcijama `signal.hann` in `signal.chebwin` iz paketa `scipy.signal`. Matrični sistem sem reševal z `linalg.solve`, ostale enačbe pa s funkcijama `numpy.roots` in `numpy.poly1d`.

13.2. Uvod

Pri metodi maksimalne entropije (MEM¹) želimo opisati spekter moči vhodnega signala $S(t)$. Točke v periodičnem signalu lahko interpretiramo tako, kot da je vsaka naslednja točka meritve določena z linearnimi kombinacijmi točk njegove zgodovine in novih informacij U

$$S_n = - \sum_k a_k S_{n-k} + G \sum_l b_l U_{n-l}, \quad (13.1)$$

kjer so a_k in b_l koeficienti ter G ojačitvena konstanta vhodnega signala. Z ustrezno transformacijo signala lahko močnostni spekter signala opišemo z izrazom

$$PDS(\omega) = |H(z)|^2 = G^2 \frac{|1 + \sum_l b_l z^{-l}|^2}{|1 + \sum_k a_k z^{-k}|^2} \quad (13.2)$$

kjer je $z(\omega) = e^{-i\omega k}$. Pri tem nas ne zanima ‐all zero‐ sistem z vrednostmi koeficientov $a_k = 0$, temveč raje postavimo $b_l = 0$ in dopustimo $a_k \neq 0$ ter dobimo

$$PDS(\omega) = \frac{G^2}{|1 + \sum_k a_k z^{-k}|^2}.$$

S takšno izbiro koeficientov dopustimo obstoj polov, s katerimi lahko določimo frekvence ω kot delta funkcije. Da bi dobili ustrezne a_k , najprej s pomočjo avtokorelačijskih funkcij signala S

$$R(i) = \frac{1}{N-i} \sum_{n=0}^{N-1-i} S_n \cdot S_{n+1} \quad (13.3)$$

¹Krajše za ang. Maximal Entropy Method.

tvorimo Yule Walkerjev sistem

$$\begin{bmatrix} R(0) & R(1) & \dots & R(p-1) \\ R(1) & R(0) & & \\ \vdots & & \ddots & \\ R(p-1) & \dots & & R(0) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_p \end{bmatrix} = - \begin{bmatrix} R(1) \\ R(2) \\ \vdots \\ R(p) \end{bmatrix}, \quad (13.4)$$

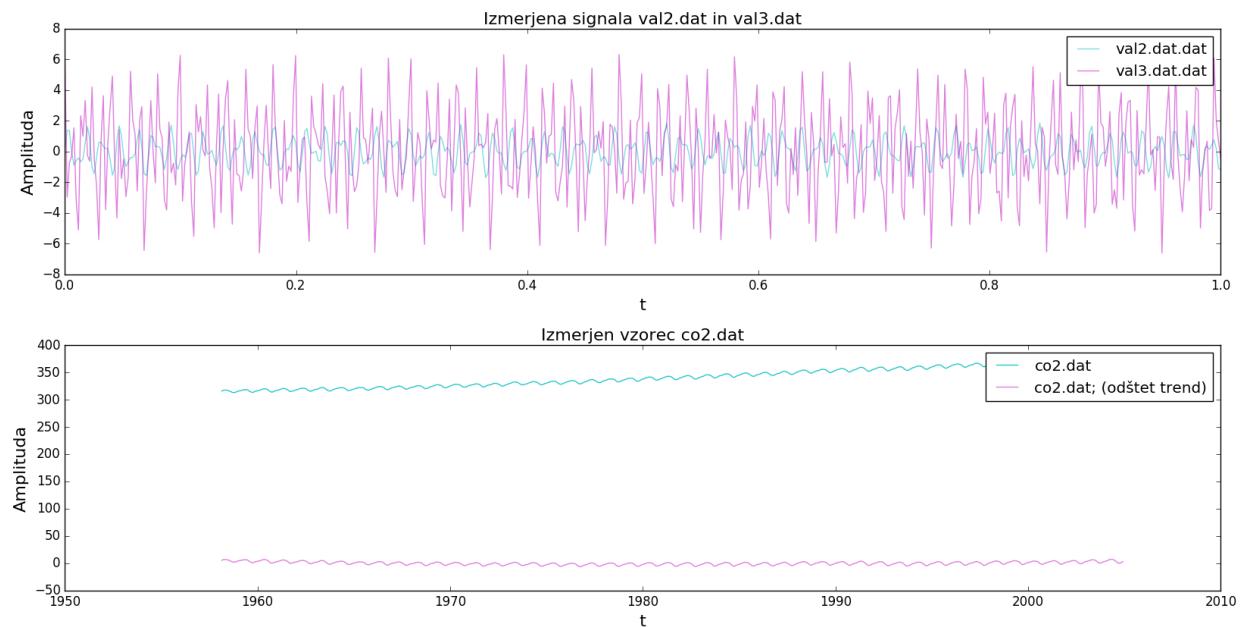
katerega rešitve nam podajo koeficiente s katerimi poiščemo pole polinomske enačbe iz imenovalca v izrazu (13.2)

$$1 + \sum_{k=1}^p a_k z^{-k} = 0.$$

Z ustreznim izbirom reda p polinoma izberemo število polov. Za vsak vrh spektra je potrebno imeti dva pola, katerih vrednosti morajo biti znotraj enotskega kroga $|z| < 1$. Če temu ni tako, pole preko enotske krožnice preslikamo v notranjost območja z želenimi vrednostmi. Da ne kvarimo faze pa to preslikavamo izvedemo po radiju. Dodatno, poli, ki so bolj oddaljeni od enotske krožnice, manj doprinesajo k spektru. Če hočemo spekter dobro opisati lahko torej variramo število polov in najbolj ustrezeni končni rezultat rezultat obdržimo. V nadaljevanju je predstavljena aplikacija MEM metode na treh različnih signalih, rezultati pa so primerjani z uporabo FFT² in okenskih funkcij.

13.3. Frekvenčni spekter vzorcev val2.dat, val3.dat in co2.dat

Iz datotek uvoženi signali so predstavljeni na sliki 13.1. Pri meritvah koncentracije CO_2 v zraku je bilo potrebo odšteti tudi linearni letni trend. Slednji je ocenjen z prilagajanjem linearne funkcije.



Slika 13.1.: Signali za vzorca val2.dat, val3.dat (zgoraj) ter signal meritev koncentracije CO_2 v zraku, kjer je bilo potrebo odšteti tudi linearni letni trend (spodaj).

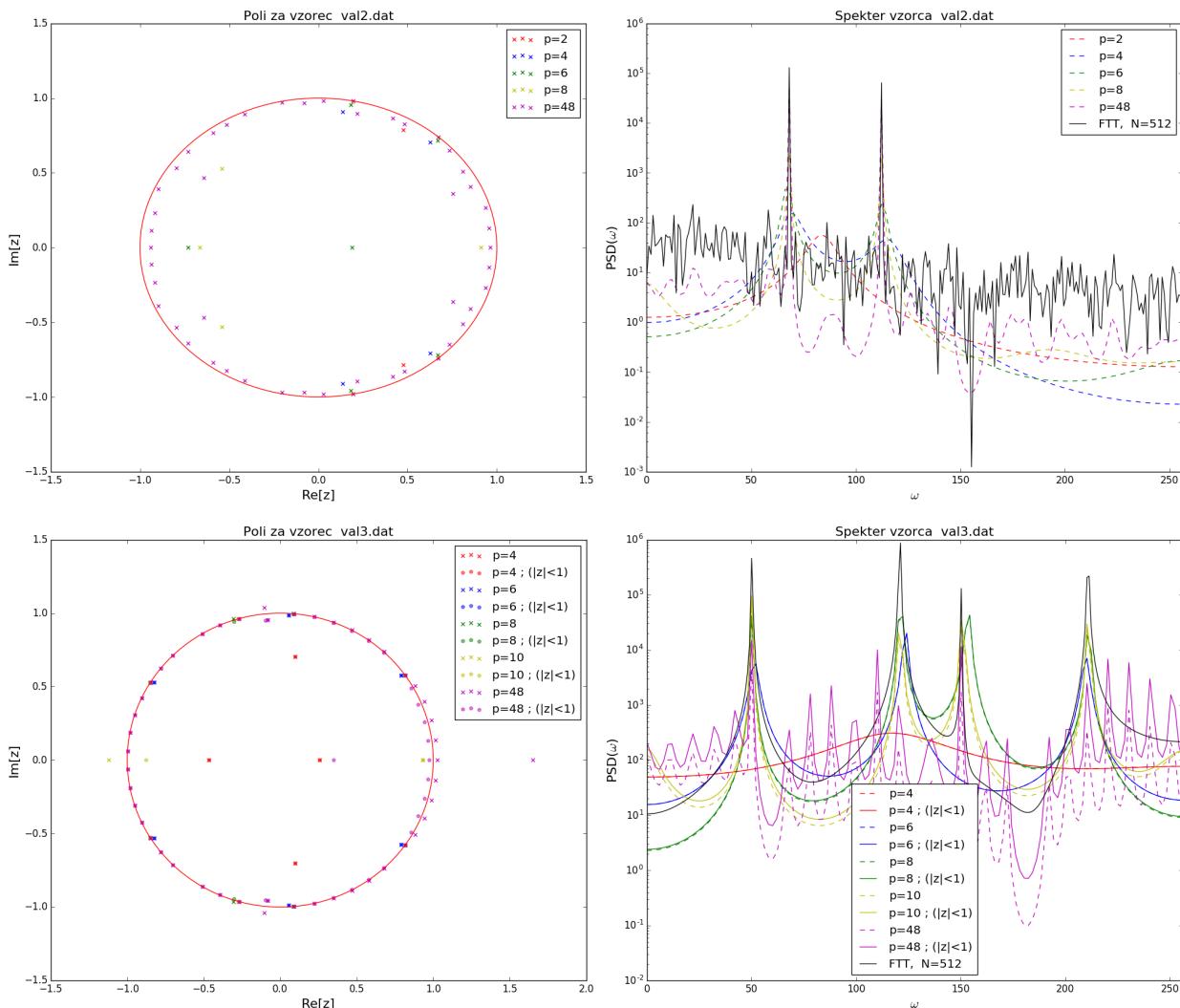
Signale za val2.dat, val3.dat sem obdelal z MEM ob izbiri različnih števil polov, ter rezultate primerjal z izračuni po metodi FFT. Oboje je prikazano na sliki 13.2. Izkaže se, da je razlika med izračuni z različnim p precejšnja. Če je število polov manjše od dvakratnika števila vrhov v analiziranem v spektru, potem metoda MEM popolnoma odpove, saj ne zadane nobenega vrha.

²Krajše za ang. Fast Fourier Transformation.

V kolikor pa s številom polov pretiravamo, začno ti pobirati šum, oziroma v primeru `val3.dat` v spekter dodajati nefizikalne vrhove. Izbira števila polov nam omogoča nadzor nad podrobnostmi, ki so nam v interesu. Čeprav za dva vrhova pri `val2.dat` načeloma potrebujemo štiri pole, poda izračun s šestimi poli večji kontrast med šumom in signalom, a žal ob prisotni manjši anomaliji ob desnem robu spektra.

Če si ogledamo še vrednosti polov, je videti, da so za `val2.dat` vsi poli preslikave znotraj enotske krožnice, pri vzorcu `val3.dat` pa jih je bilo potrebno preslikati v notranjost enotskega kroga. Za slednji vzorec je spekter izračunan tudi za primer pobeglih polov in iz rezultatov je res videti, da dodatna preslikava polov vpliva na resolucijo spektra.

Po primerjavi s FFT lahko sklepamo, da metoda MEM ob pravilni izbiri števila polov pobere precej manj šuma. Pri signalu `co2.dat` sem močnostni spekter računal za primera z in brez odštetega

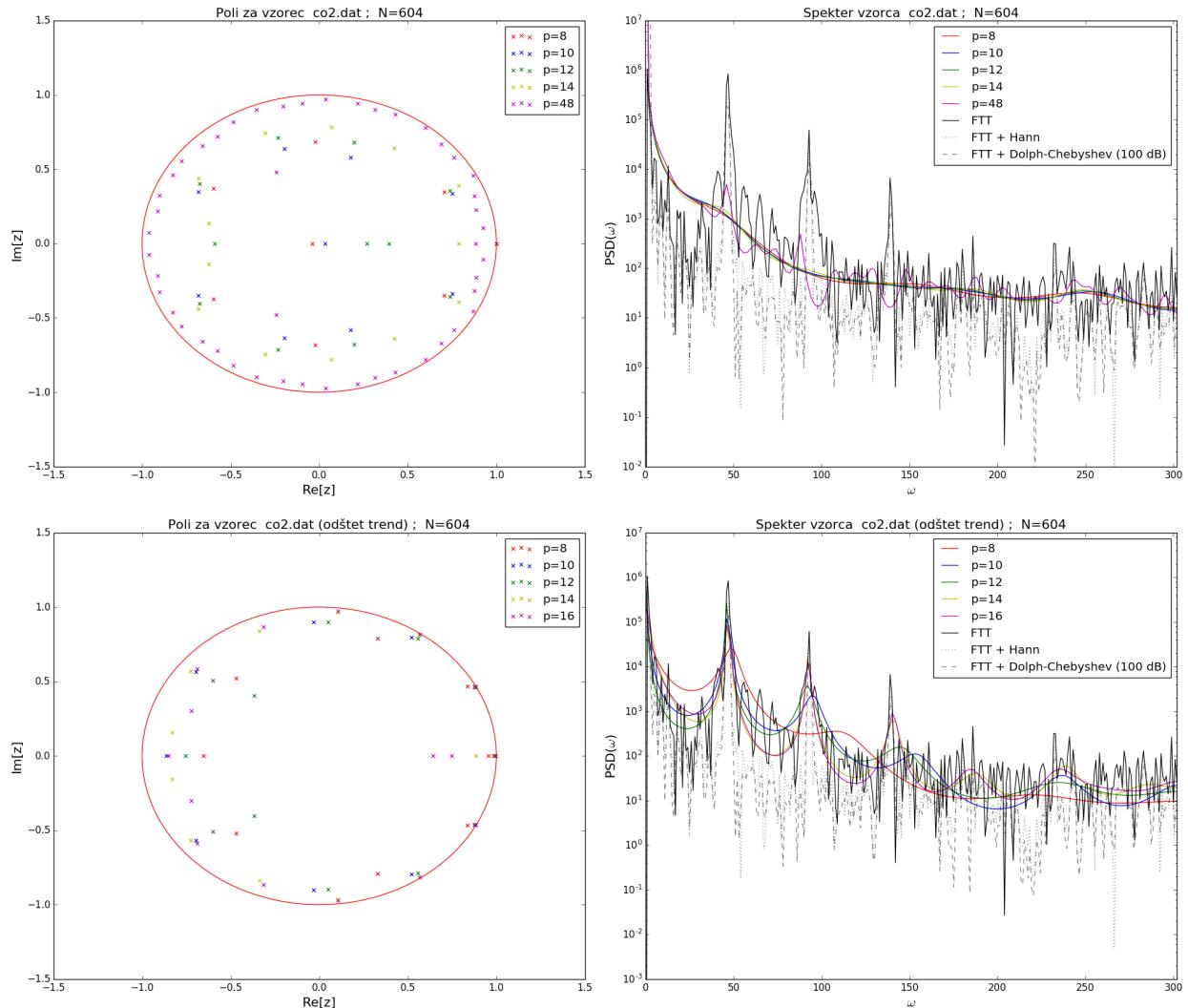


Slika 13.2.: Izračuni polov in močnostnega spektra za signala `val2.dat` (zgoraj) in `val3.dat` (spodaj).

Poli ‘ \times ’ so večinoma znotraj enotske krožnice, le pri vzorcu `val3.dat` jih je bilo neka potrebno preslikati po radijo čez krožnico v notranjost enotskega kroga ‘•’, takšna dodatna preslikava polov pa vpliva na resolucijo spektra. Iz spektrov je opaziti, da metoda MEM ob pravilni izbiri števila polov pobere precej manj šuma kot metoda FFT. Najboljša izbira polov za primer `val2.dat` bi bila $p = 8$, za primer `val3.dat` pa $p = 10$.

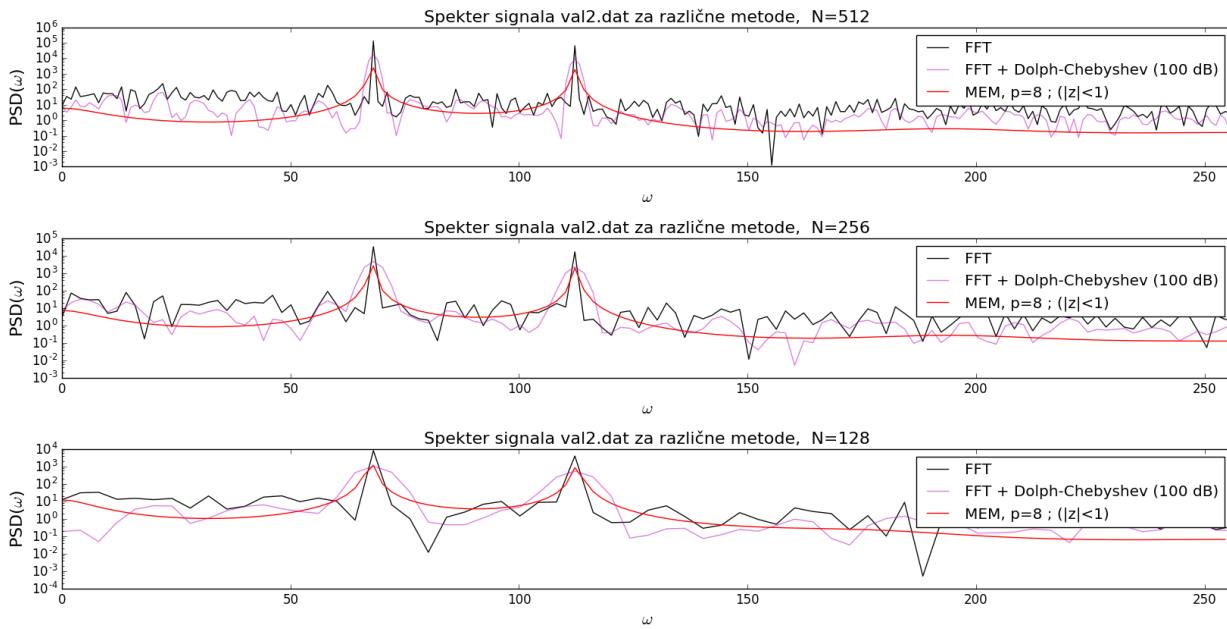
letnega trenda. Rezultati so za obe metodi prikazano na sliki 13.3.

Na koncu lahko preverimo, kako se metodi MEM in FFT obnašata pri različnih gostotah vzorčenja signalov. Poračunati je bilo toej potrebno še spektre za signale z polovičnim in četrtinskim vzorčenjem.

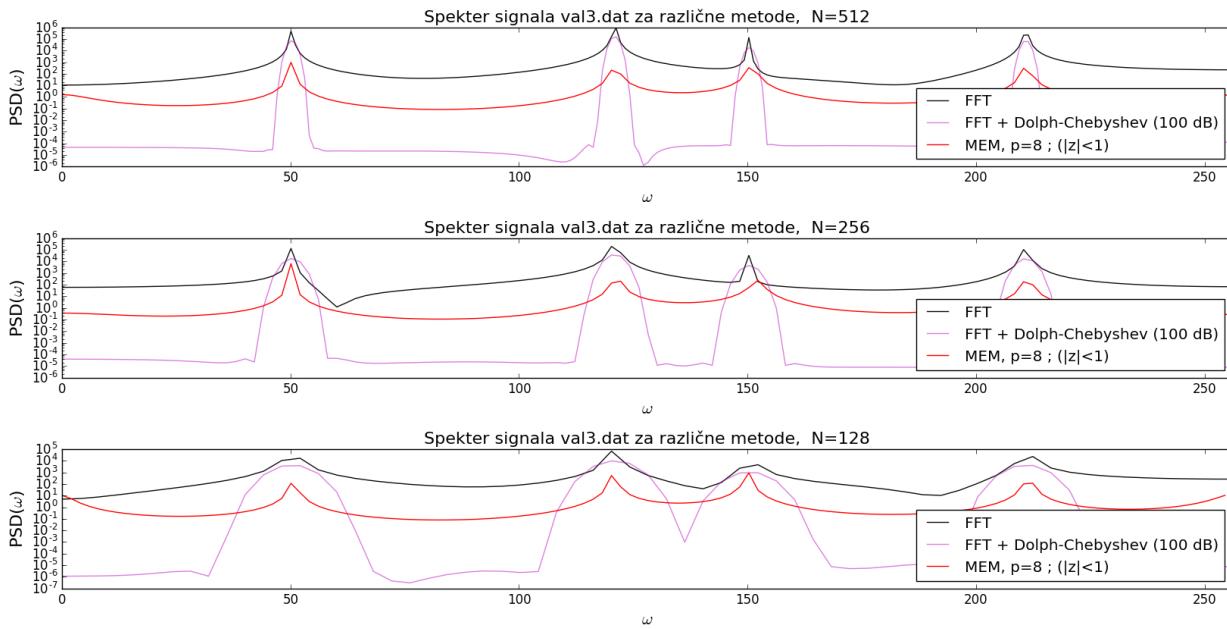


Slika 13.3.: Izračuni polov in močnostnega spektra za originalen signal `co2.dat` (zgoraj) in signal z odštetim letnim trendom (spodaj). Vsi poli so znotraj enotske krožnice. Iz spektrov pa je opaziti, da metoda MEM bolje deluje če odstejemo linearni letni trend. Najboljša izbira polov za ta primer bi bila $p = 14$. Ker MEM ob pravilni izbiri števila polov pobere precej manj šuma kot metoda FFT, je iz grafov lažje razbrati prisotnost zašumljenih manjših vrhov na desni strani spektra. Metodo FFT je moč precej izboljšati z uporabo okenske funkcije; v tem primeru Dolph-Chebyshev okna.

Rezultati za val2.dat, val3.dat in co2.dat so v tem redu prikazani na slikah (13.4), (13.5) in (13.6). Od tu je očitno, da z manjšanjem gostote vzorčenja precej bolj prizadanemo metodo FFT.



Slika 13.4.: Spektri računani po metodah MEM in FFT pri različnih gostotah vzorčenja signalov. Manjšanje gostote vzorčenja ima večji učinek na metodi FFT

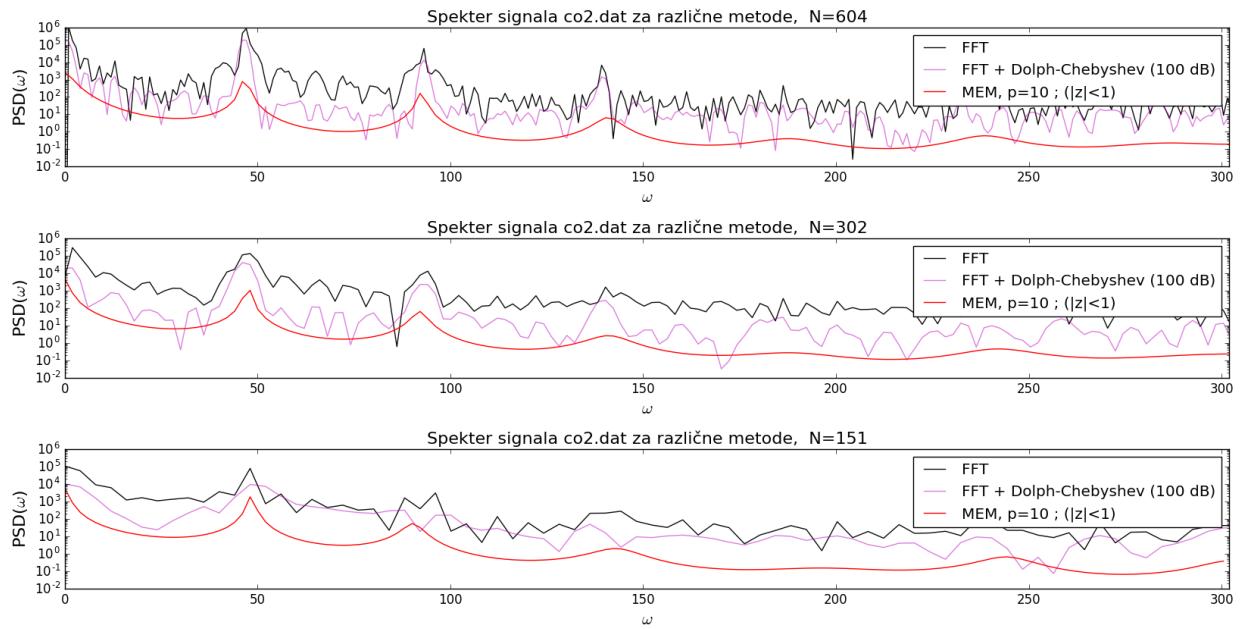


Slika 13.5.: Spektri računani po metodah MEM in FFT pri različnih gostotah vzorčenja signalov. Manjšanje gostote vzorčenja ima večji učinek na metodi FFT

13.3.1. Ločljivost metode

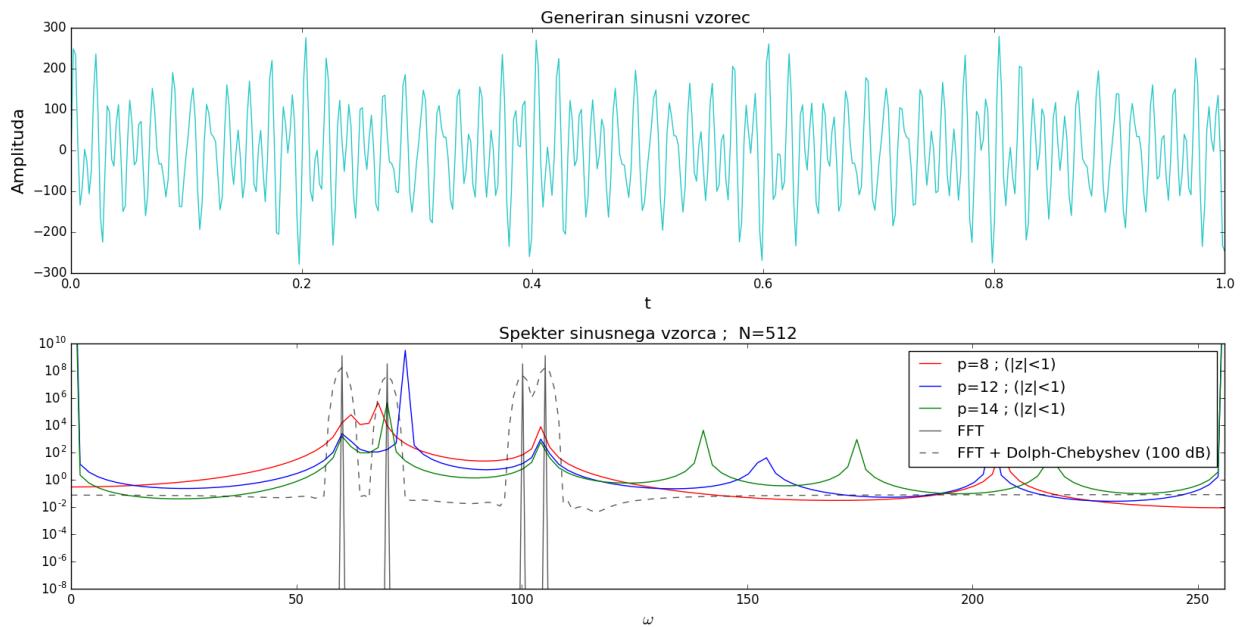
Za preverjanje ločljivosti metode sem generiral signal iz prispevkov sinusnih valovanj z različnimi amplitudami in frekvencami

$$S(t) = 100 \cdot \sin(2\pi \cdot 60 \cdot t) + 50 \cdot \sin(2\pi \cdot 70 \cdot t) + 50 \cdot \sin(2\pi \cdot 100 \cdot t) + 100 \cdot \sin(2\pi \cdot 105 \cdot t) ,$$



Slika 13.6.: Spektri računani po metodah MEM in FFT pri različnih gostotah vzorčenja signalov. Manjšanje gostote vzorčenja ima večji učinek na metodi FFT

da bi ugotovil, spekter katere metode bo bolje zaznal bližnja si vrhova. Generiran signal in iz njega izračunani spektri so prikazani na sliki 13.7. Tak čist signal najbolje razvzrožla metoda FFT, ki ji okenska funkcija ob odsotnosti šuma le zmanjša ločljivost. Metoda MEM ima v primerjavi t FFT slabšo ločljivost, saj so vrhovi bolj razmazani kot pri FFT in v spektru niti ne uspe razločiti dveh najbližjih si vrhov. Namesto tega pole porabi tako, da nekje v spektru ustvari dodatne umetne vrhove. Osto oko opazi še, da MEM v spektru poda povsem napačna razmerja amplitud vrhov.

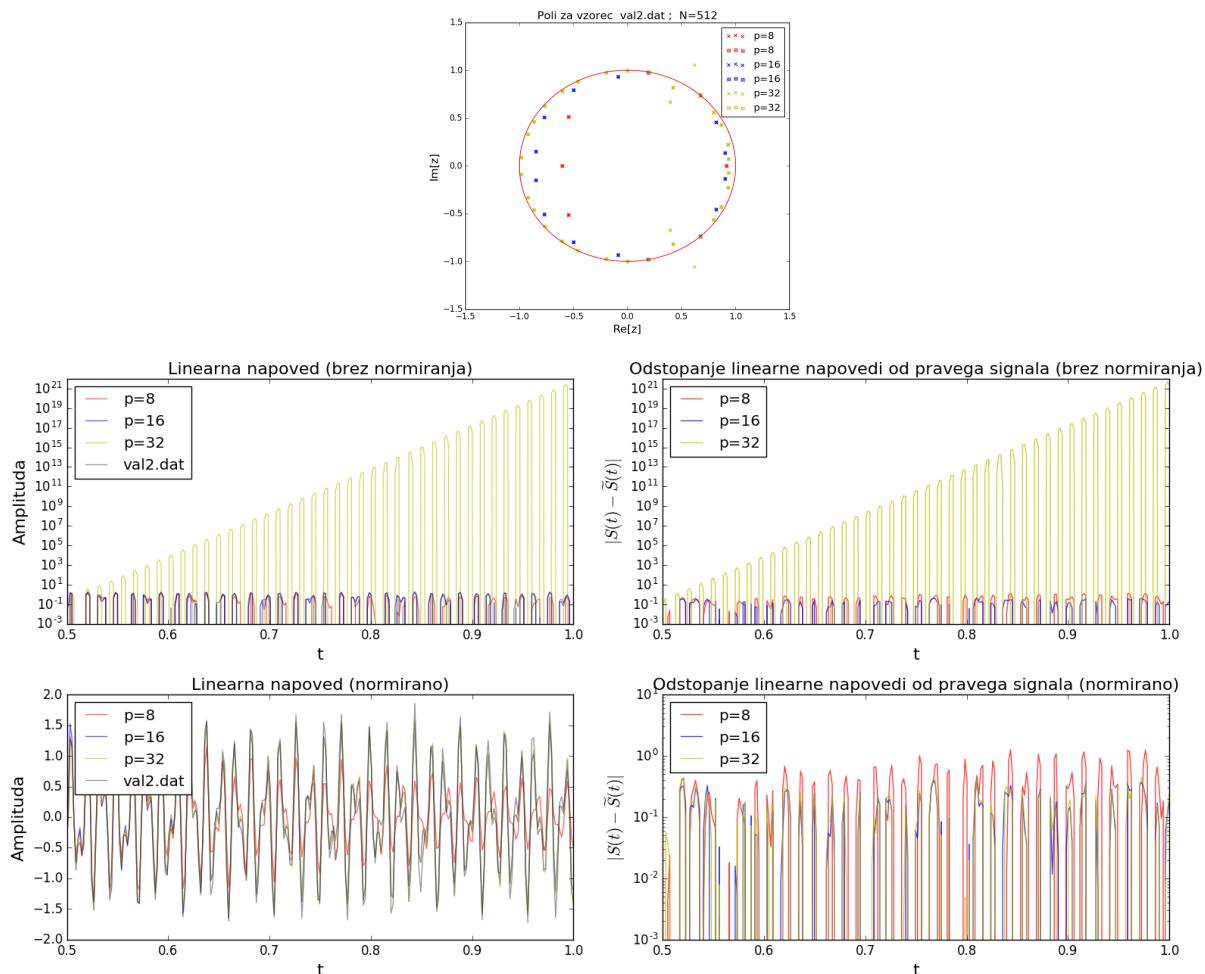


Slika 13.7.: Generiran signal $S(t)$ in iz njega izračunani spektri po metodah FFT in MEM. V primeru odsotnosti šuma se očitno bolje obnese metoda FFT.

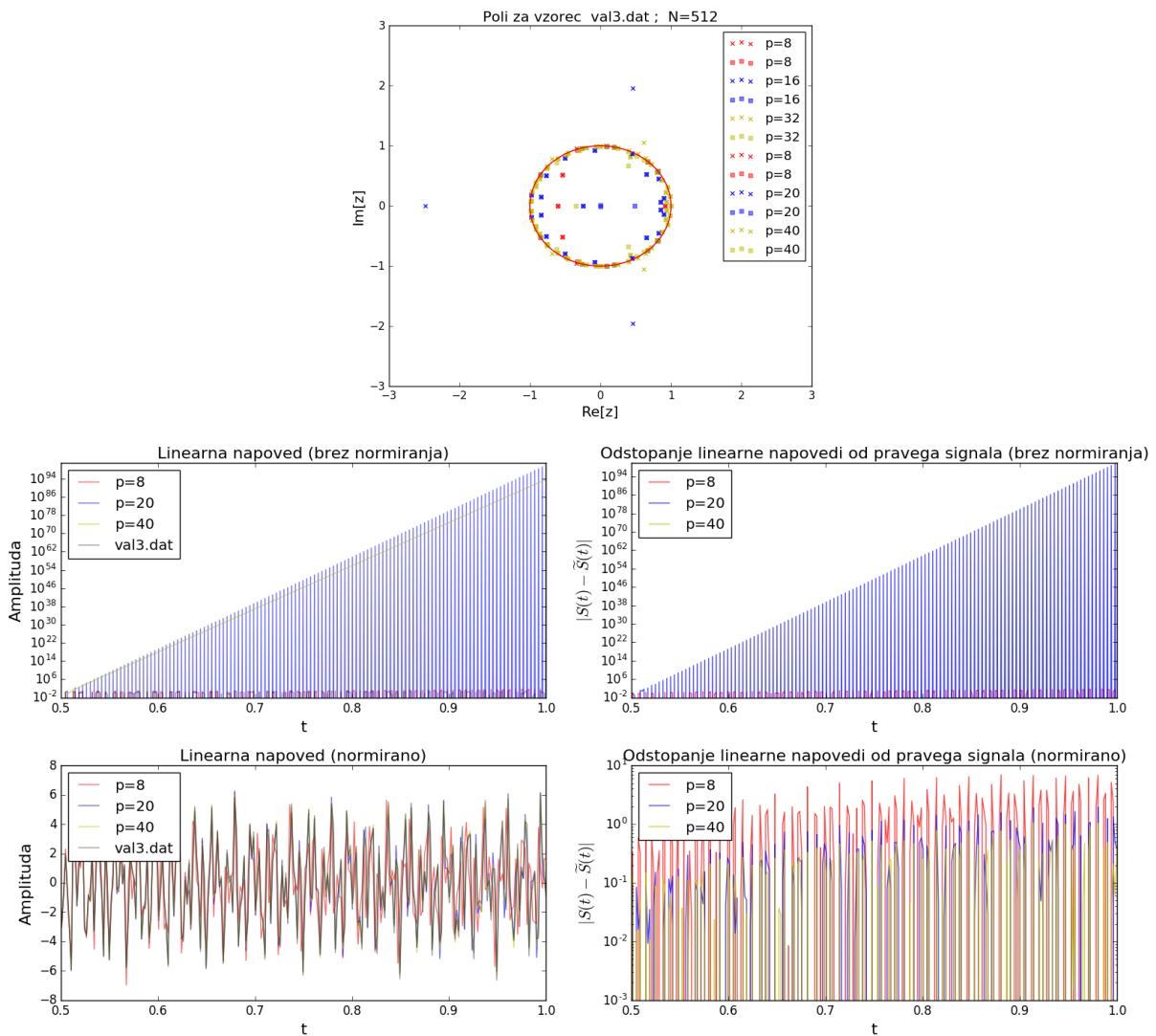
13.4. Linarne napovedi

V prejšnjem delu vaje so bili za vsak signal $S(t)$ izračunani koeficienti a_k . Slednje lahko porabimo za izračun približne napovedi obnašanja signala v prihodnosti $\tilde{S}(t)$. Tak približek signala računamo kar po enačbi (13.1), kjer vse koeficiente b_l postavimo na nič. Za namen primerjanja takšne napovedi z realnostjo, koeficiente a_k izračunamo le iz informacij prve polovice meritov, iz njih določeni $\tilde{S}(t)$ pa nato odštejemo od druge polovice meritov $S(t)$ in preverimo odstopanja.

Poskusi napovedi za signale `val2.dat`, `val3.dat`, `co2.dat`, `luna.dat`, `Wolf_number.dat` in `borza.dat` so po vrsti prikazani na slikah 13.8-13.13. Vidimo, da napovedi niso stabilne, saj amplitudo divergirajo kljub presenetljivo dobremu lovljenju faze v nekaterih od primerov. V primerjavi s prvim delom naloge se tokrat izkaže, da nekoliko večje število polov poda boljše napovedi; seveda pa obstaja tudi neka meja, ko zaradi prevelikega števila polov napoved podivja, vendar prikazani izračuni ostajajo pod to mejo. Posebej pomembna je normalizacija polov v notranjost enotskega kroga, s čimer ohranimo fazo in stabilnost napovedi precej izboljšamo. Pri prvih štirih vzorcih se napovedi še dokaj skladajo z izmerjenim signalom, medtem ko so napovedi za `Wolf_number.dat` in `borza.dat` slabše. Posebej pri slednjem vzorcu je napoved izjemno slaba oz. povsem nemogoča, saj na dolgi rok ni pričakovati posebnih periodičnih nihanj vrednosti delnic in metoda nima čemu slediti.



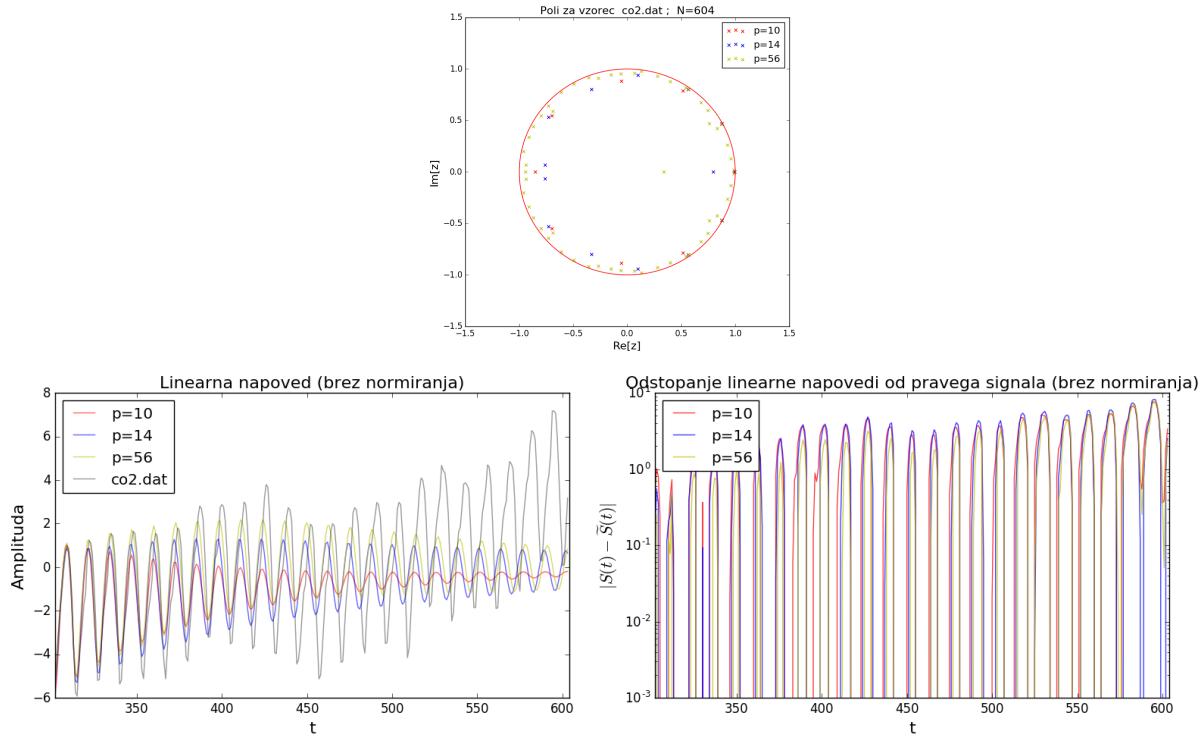
Slika 13.8.: Izračuni polov (zgoraj) in linearnih napovedi (levo) z odstopanjem od realne meritve (desno) za signal `val2.dat` za normirane in nenormirane pole. Poli so večinoma znotraj enotske krožnice, le nekaj jih je bilo potrebno po radiju preslikati čez krožnico v notranjost enotskega kroga, da ne pokvarimo faze. Preslikava polov ima tudi očiten dober vpliv na stabilnost amplitude. Iz oblike signalov na spodnjih grafih je videti spodborno ujemanje, vendar logaritemska skala odstopanj ne kaže na to, da je bila dosežena posebna natančnost.



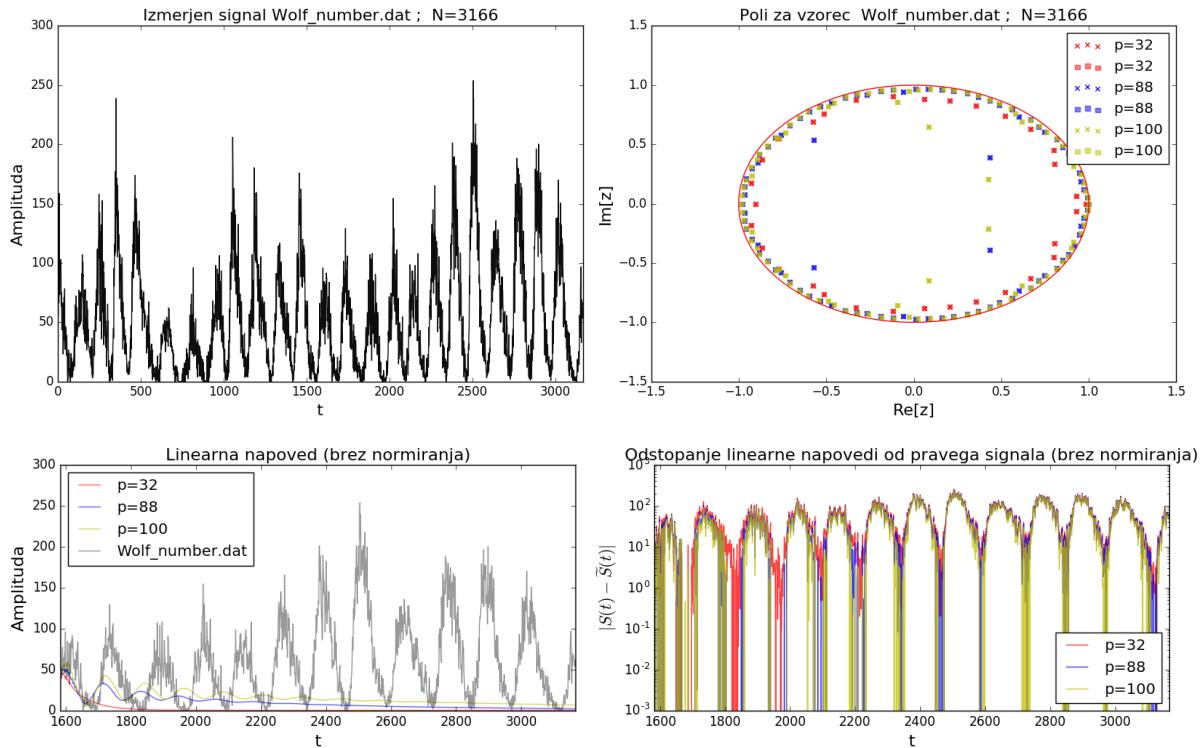
Slika 13.9.: Izračuni polov (zgoraj) in linearnih napovedi (levo) z odstopanjem od realne meritve (desno) za signal val3.dat za normirane in nenormirane pole. Kot v primeru prejšnjega signala 13.8 so poli večinoma znotraj enotske krožnice, le napoved je malce slabša, saj se odstopanja od prave vrednosti s časoma vedno bolj povečujejo.

13.4.1. Vpliv šuma na linearno napoved

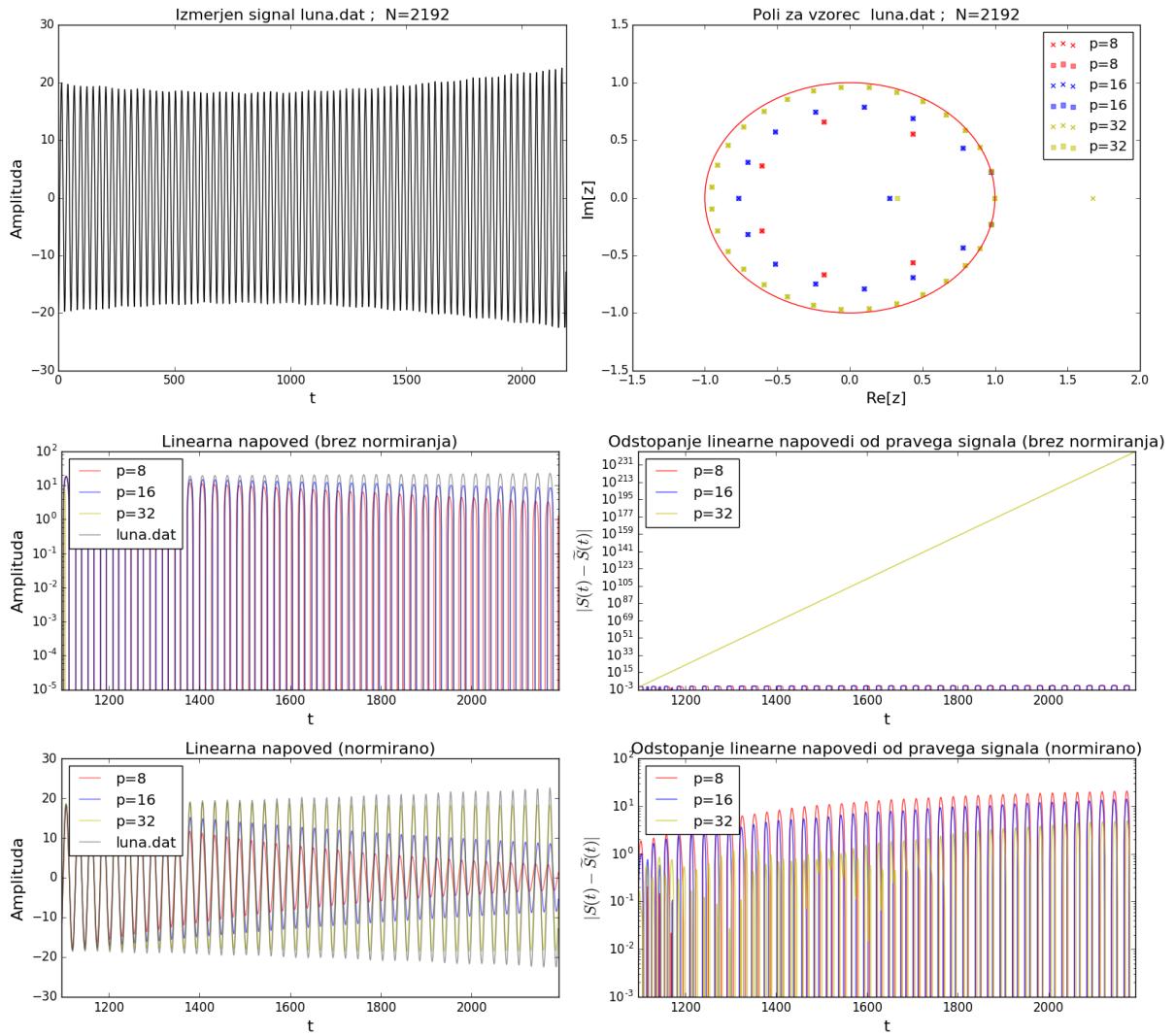
Na koncu preverimo še kakšen vpliv na napoved ima šum. Za ta namen sem uporabil vzorec val2.dat katerega napoved je podala najboljše rezultate. Signalu vzorca sem prištel različna deleža šuma in izračunal napoved. Rezultati so prikazani na sliki 13.14. Videti je da lahko šum napoved povsem pokvari; pri majhnem deležu šuma lahko s povečanjem polov napoved še nekako izboljšamo, ko pa šum preseže neko vrednost napovedovanje z metodo MEM povsem odpove.



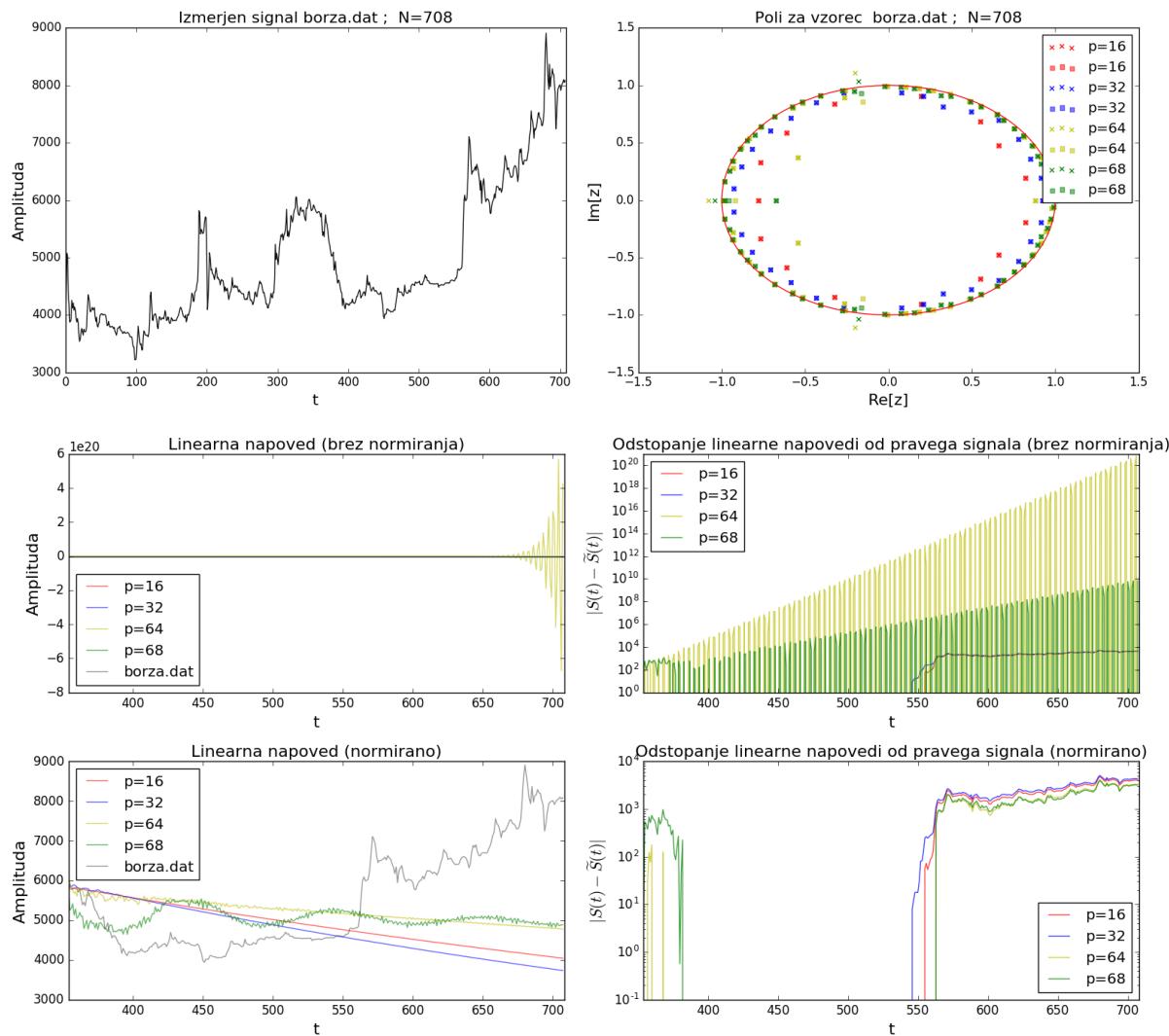
Slika 13.10.: Izračuni polov (zgoraj) in linearne napovedi z odstopanjem od realne meritve (spodaj) za signal `co2.dat` (z odštetim linearnim trendom). Napoved je računana le za nenormirane pole, saj so bili tokrat vsi izračunani poli že znotraj enotske krožnice. Tu je napoved že precej slabša. Sčasoma se izgubljata tako amplituda kot tudi faza napovedi.



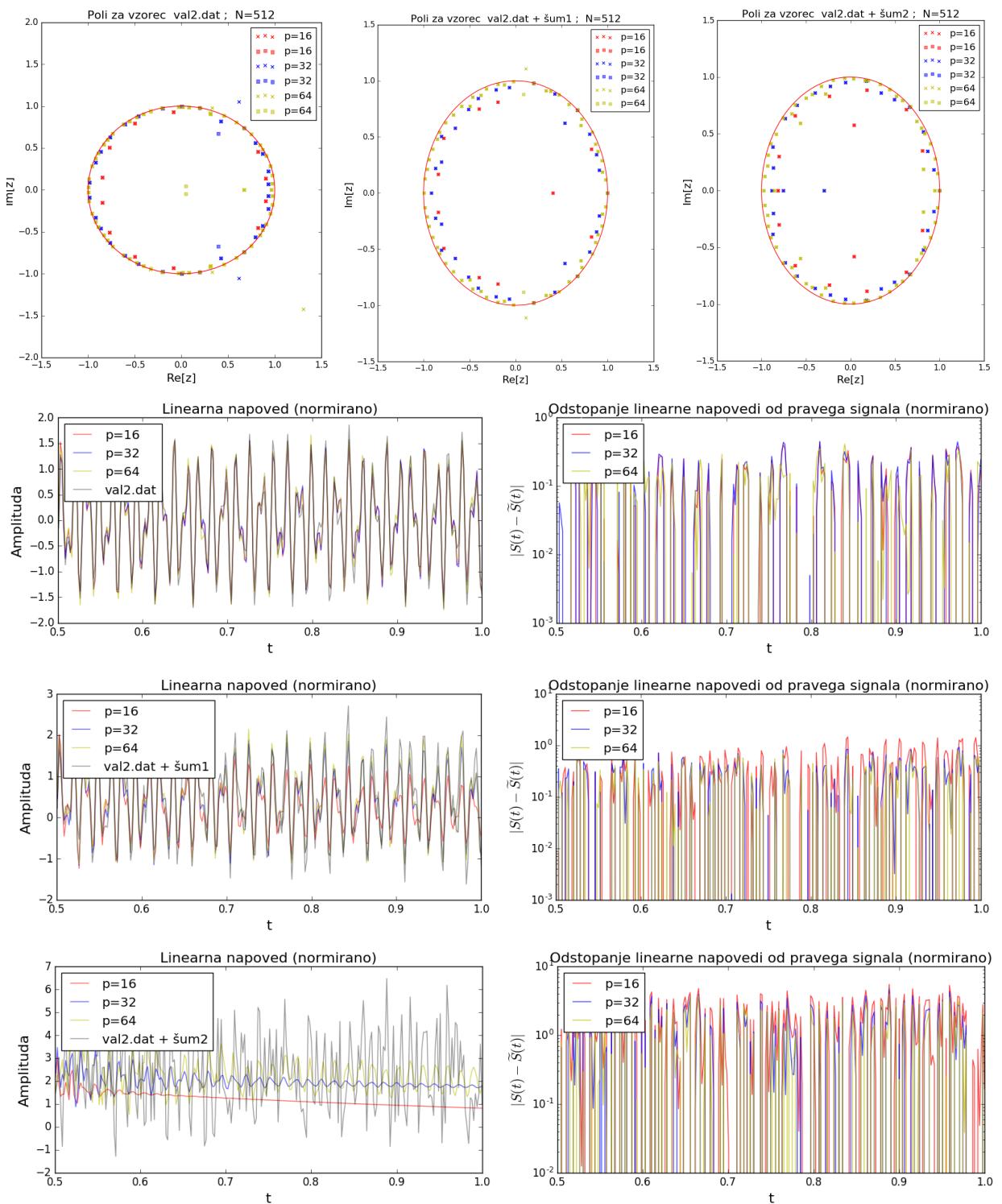
Slika 13.11.: Izris signala in izračuni polov (zgoraj) ter linearna napoved z odstopanjem od realne meritve (spodaj) za vzorec `Wolf_number.dat`. Napoved je računana le za nenormirane pole, saj so bili tudi v tem primeru vsi poli že znotraj enotske krožnice. Tu je napoved že precej slabša. Tako amplituda kot tudi faza napovedi sta zgrešeni kmalu po obetavnem začetku intervala.



Slika 13.12.: Izris signala in izračuni polov za vzorec *luna.dat* za normirane in nenormirane pole. Poli so večinoma znotraj enotske krožnice, preostali so za izboljšanje stabilnosti amplitude in faze vanjo preslikani. Iz oblike signalov na spodnjih grafih je videti spodborno ujemanje faze tudi za majše število polov.



Slika 13.13.: Izris signala in izračuni polov (zgoraj) ter linearna napoved za nenormirane in normirane pole ter odstopanja od realne meritve (spodaj) za vzorec `borza.dat`. Lep primer slabe borzne spekulacije. Tu je napoved najslabša, saj MEM praktično nima čemu slediti. Tako amplituda kot tudi faza originalnega signala sta naključni, zato sta tudi napovedi zgrešeni že v samem začetku intervala. Morda bi se napoved lahko izračunala za precej krajše časovne intervale ob kakih "faznih prehodih" na trgu, kot so prodaja podjetja ali z njim povezani medijski škandali.



Slika 13.14.: Izračuni polov (zgoraj) in linearnih napovedi (levo) z odstopanjem od realne meritve (desno) za čisti signal val2.dat in isti signal z različnima deležema dodanega šuma. Poli so večinoma znotraj enotske krožnice, le nekaj jih je bilo potrebno preslikati po radiju čez krožnico v notranjost enotskega kroga, da ne pokvarimo fazne napovedi. Iz oblike signalov na spodnjih grafih je že na oko videti, da lahko šum napoved povsem pokvari. Pri majhnem deležu šuma lahko s povečanjem polov napoved še malenkostno izboljšamo, ko pa šum preseže neko vrednost dodajanje polov ne pomaga več.

Literatura

- [1] Spletne strani z gradivom predmeta, ogled 20.10.2017,
<http://predmeti.fmf.uni-lj.si/modelska>
- [2] Spletne strani z gradivom predmeta, ogled 20.10.2017,
<http://predmeti.fmf.uni-lj.si/modelska/podatki/>

