

Univerza v Ljubljani
Fakulteta za računalništvo in informatiko
Tržaška 25
Ljubljana

Seminarska naloga

Stohastični model bistabilnega stikala

Avtorji:

Pece Adzиеvski, Borče Paspalovski, Aleš Uršič, Andrej Babič

Mentor:
Miha Mraz

Ljubljana, Januar 2011

Kazalo

| | |
|--|----|
| 1. Uvod..... | 3 |
| 2. Predstavitev projekta DTU-Denmark iz iGEM 2010..... | 5 |
| 3. Matematično modeliranje..... | 12 |
| 3.1 Stohastični model..... | 12 |
| 3.2 Opis parametrov ki nastopajo v enačbi..... | 14 |
| 3.3 Realizacija v Matlabu..... | 14 |
| 4. Simulacija..... | 16 |
| 4.1 Spreminjanje parametra β_0 | 17 |
| 4.2 Spreminjanje parametra β | 18 |
| 4.3 Spreminjanje koeficienta kooperativnosti n | 18 |
| 4.4 Spreminjanje koeficienta represiranja K | 22 |
| 4.5 Spreminjanje parametra δ | 22 |
| 5. Zaključek..... | 25 |
| 6. Viri..... | 25 |

1. Uvod

DNK Procesiranje je alternativna platforma za procesiranje podatkov. Če razmišljamo v računalniškem smislu, prevzamejo nalogo procesorja biološki procesi povezani z DNK, medtem ko prevzame nalogo programa sam DNK zapis.

DNK zapis določa pogoje za izdelavo proteinov in navodila za njihovo sestavljanje. Lahko rečemo, da so proteini izhodni podatki oz. rezultat procesiranja. Izdelavo proteinov je potrebno sprožiti oz. omogočiti. To storijo drugi proteini, katere obravnavamo kot vhodne podatke.

Procesiranje poteka znotraj gostujočih celic. Osnovni gradniki, ki se nahajajo v celicah so:

- DNK zapis
Sestavljen je iz štirih nukleotidnih baz. Kombinacija nukleotidnih baz poljubne dolžine sestavlja unikatni zapis, ki vsebuje navodila za sestavljanje proteinov.
- Kodon
Je enota genetskega zapisa znotraj DNK in RNK. Sestavljen je iz treh zaporednih nukleotidov. Obstaja 4^3 različnih kodonov. Kodon določa ciljno aminokislino ali pa je t.i. stop kodon, ki ustavi prepisovanje DNK v mRNK.
- Promotor
Del DNK zapisa na katerega se veže RNK polimeraza. Ob vezavi se sproži faza transkripcije.
- Operator
Del promotorja, na katerega se vežejo transkripcijski faktorji.
- Transkripcijski faktor
Delimo jih na aktivatorje in represorje. Veže se na operator in s tem pospešuje ali zavira vezavo operatorja na promotor. Posredno vpliva na frekvenco transkripcije mRNK sporočil.

Tvorbo novih proteinov oz. Procesiranje delimo na transkripcijo in translacijo. Celoten postopek nas pripelje od podaje vhodnih podatkov, do prejetja izhodnih.

Transkripcija

Je prva faza oz. začetek procesiranja:

- Transkripcija se začne ko se RNK polimeraza veže na promotor DNK zapisa. To se lahko zgodi ob prisotnosti ali odsotnosti specifičnih transkripcijskih faktorjev.
- RNK polimeraza, ki je vezana na promotor, potuje vzdolž DNK zapisa in genetski zapis prepisuje v mRNK (message RNK). Temu procesu pravimo transkripcija, kar daje ime celotni fazi procesiranja.
- Transkripcija se zaključi, ko RNK polimeraza pride do stop kodona.

Translacija

Je zadnja faza oz. zaključek procesiranja.

- Translacija se začne takoj, ko se zaključi transkripcija. Na molekulo mRNK, ki je nastala v fazi transkripcije, se vežejo ribosomi.
- Vezava ribosomov sproži prevajanje mRNK molekule v zaporedje aminokislin.
- Ko se prevajanje zaključi, imamo ciljni protein.

Intenzivnost tvorbe proteinov lahko reguliramo s transkripcijskimi faktorji. Delimo jih na aktivatorje in represorje. Aktivatorji tvorbo proteinov pospešujejo, vežejo se na operator in s tem pospešijo frekvenco prepisovanja mRNK sporočila. Represorji delujejo obratno in frekvenco mRNK prepisovanja zmanjšujejo.

Intenzivnost tvorbe proteinov lahko spremljamo z vzporednim generiranjem t.i. proteinov zaznave. Uporabljajo se samo za določanje frekvence prepisovanja mRNK sporočil. Primer takega proteina je GFP – zeleni fluorescirajoči protein)

Zgoraj opisan model lahko primerjamo z kombinatoričnimi vezji:

| Kombinatorično vezje | DNK procesiranje |
|----------------------|--|
| Vhodni biti | Proteini, ki delujejo kot transkripcijski faktorji |
| Logika | Zapis DNK, ki določa tvorbo proteinov |
| Izhodni biti | Proteini, nastali iz mRNK sporočila |

Z samimi DNK zapisi lahko sestavimo torej le kombinatorična vezja. Da bi lahko sestavili tudi sekvenčna vezja potrebujemo še pomnilne celice.

2. Predstavitev projekta DTU-Denmark iz iGEM 2010

Biološka stikala

Biološko genetsko stikalo je sistem, ki omogoča celicam, da si "zapomnejo" stanje, ki ga je postavil nek določen signal. Mnogo bioloških procesov obstaja v medsebojno izključujočih stanjih. Celice ali organizmi obstajajo v enem izmed različnih stanj v odvisnosti od nivoja stimulatívnega vhoda. Sprememba v stanju se sproži kot odgovor na fluktuacije stimulatívnega vhoda iz nizkega nivoja na visoki ali obratno. Nivo, ki ločuje nizko in visoko stanje (threshold level), se pogosto ugotavlja s pomočjo regulatornih vezij s pozitivnim ali negativnim povratnim mehanizmom.

Bakteriofagi so tipičen primer naravnih genetskih stikal, zato ker lahko ločijo med dvema različnima življskima cikloma imenovana litični in lizni cikel. Čeprav, ko se enkrat litični cikel sproži, se lizični cikel ne more več ponoviti.

Bistabilna stikala

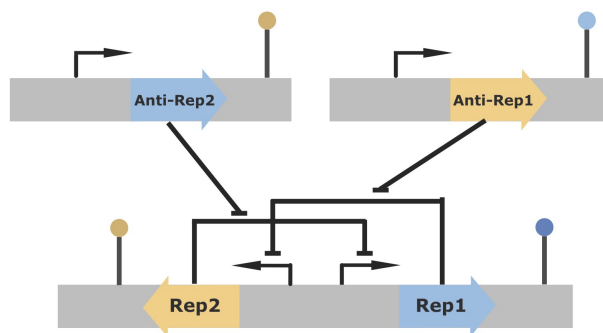
V svetu elektronike in računalništva so bistabilna stikala element z možnostjo pomnjenja in dvema medsebojno izključujočima stabilnima stanjima. Ko je stabilnost enega stanja dosežena, samo intenzivnost vhoda nad pragovnim nivojem, bi spremenila sistem tako, da bi se začel bližati novemu stabilnemu stanju. Bistabilna stikala ohranjajo stanje, če je nivo koncentracije vhoda manjši od pragovnega nivoja. Temu pravimo histerezna zanka.

Bistabilna stikala niso mehanizmi, ki se pojavljajo v naravi. Na primer, ko se pri bakteriofagu inducira litični cikel, se lizični ne more več ponoviti. To je lahko tudi primer robustnega sistema, kjer ko dosežemo neko stanje tudi ostanemo v njem, ne glede na intenzivnost vhoda. Robustna stikala imajo v biološkem svetu pomembno vlogo, saj mehanizmi kot so spreminjanje celic med razvijanjem-rastjo, se genski regulatorni mehanizmi ne smejo spreminjati. To lahko dosežemo z mrežo genov, ki se medsebojno regulirajo z represnimi in aktivacijskimi proteini, ki jih lahko sami kodirajo.

Dizajn bistabilnega stikala

Najenostavnejši dizajn takšnega biološkega stikala je tak, kjer oba represorska proteina represirata sintezo ostalega proteina. Kadar sta oba represna proteinoma omogočena, bo eno izmed dveh stabilnih stanj opazovano. V enem stabilnem stanju je represor "ena" vklopljen in in represor "dva" izklopljen. Represor "ena" represira represor "dva" kar pomeni, da je represor "ena" samosvoj aktivator. V drugem stabilnem stanju je represor "dva" vklopljen in represor "ena" izklopljen.

V sistemih, kjer so represorji lahko nadzorovani z zunanjimi vhodi npr. Inducerji ali anti-represor proteini, lahko vplivamo na spremembo stanja sistema. To je ilustrirano na naslednjem zgledu:



Slika 1: Enostavno bistabilno stikalo

Navdih za dizajn bistabilnega stikala je skupina na Danskem iskala v naravi. Našli so regulatorni sistem lambda in Gifsy fage. Gifsy fage so zmerne fage, ki jih najdemo v *Salmonella enterica* bakteriji. Imajo celotno gensko organizacijo tipično za lambda faga družino.

Razvoj stikala po korakih

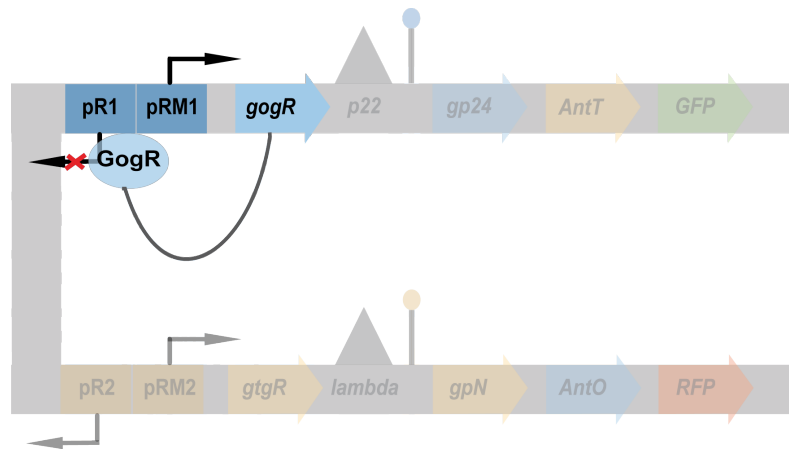
Pri bistabilnem stikalu imamo tri elemente za regulacijo:

1. divergentne promotorje z represorji iz gifsy fage
2. anti represorje iz gifsy fage
3. anti terminatorje iz lambda fage

Vsi ti elementi so vključeni in tako postane naše bistabilno stikalo zelo robustno. Sedaj pa sledi demonstracija konstrukcije bistabilnega stikala.

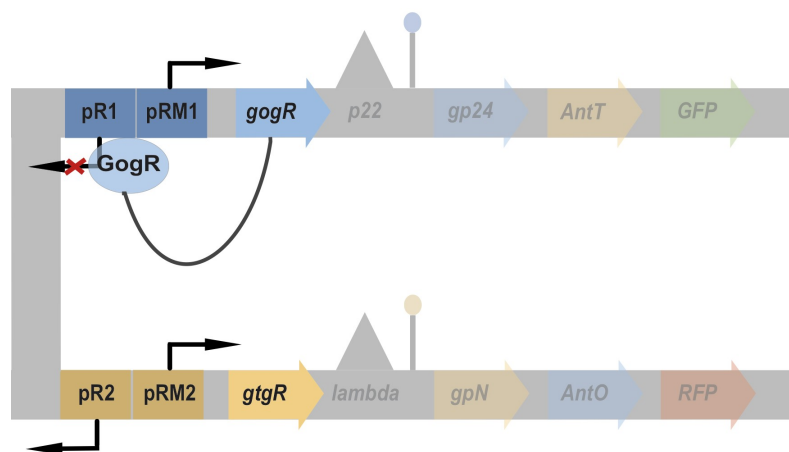
Korak 1: Promotorji in represorji

Divergentni promotorji iz gifsy fage so sestavljeni iz zelo močnih pR promotorjev in manj močnega pRM promotorja, ki uravnava izhod represorja, ki represira pR promotor.



Slika 2: Dva divergentna promotorja pR1 in pRM ter represor GogR so osvetljeni.

Ker divergentna promotorja nimata enake promotorske moči, so se odločili vključiti dve množici divergentnih promotorjev iz gifsy 1 in gifsy 2. To je ilustrirano na Sliki 3.



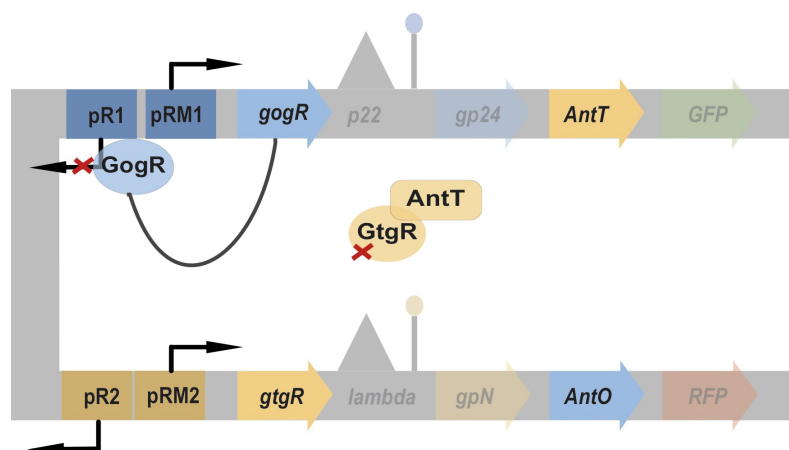
Slika 3: Divergentna promotorja iz gifsy 1 in gifsy 2 fage sta osvetljena. Divergentna promotorja iz gifly 1 sta pR1 in pRM1, iz gifly 2 pa pR2 in pRM2.

Represor GogR iz gifsy 1 fage bo represiral promotorja pR1, ko le-ta vzpostavlja izhod. Represor GtgR iz gifsy 2 fage bo represiral promotorja pR2, ko je ta izražen. Šibkejša pRM1 in pRM2 sta ključna/osnovna promoterja in zato bosta tam konstantno izražena GogR in GtgR. Izražanje (izhod) GogR in GtgR bo rezultat represiranja pR1 in pR2. K temu problemu se bomo vrnili v koraku 2.

Ker sta pR1 in pR2 močna promoterja bosta uporabljena v stikalu za ojačanje izhoda genov iz pRM2 in zato tudi pRM1. Posledično bo pR1 promoter nadziral polovico stikala 2 (na sliki 3 ilustrirano kot spodnji del) in pR2 bo nadziral polovico stikala 1 (na sliki 3 ilustrirano kot zgornji del).

Korak 2: Uvedba anti represorjev

Promoterji pR so lahko derepresirani z dodajanjem anti represorjev v sistem. Ti bodo zavirali efekt represorjev v sistemu. Oba anti represorja AntO in AntT bosta anti represirala GogR in GtgR in istočasno derepresirala pR1 in pR2 promoterja.

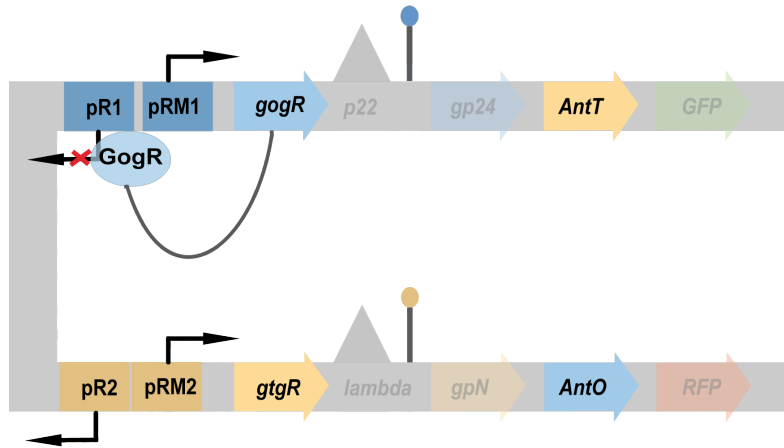


Slika 4: Ko je antirepresor izražen izhod, se bo povezal z represorjem in zato se bo na pR promoterju pojavil izhod. Tukaj je ilustracija gify 2 fage.

Kot je prikazano na sliki 4 bo izražen antirepresor AntO antirepresiral GogR in zato povzročil derepresiranje pR1 promotorja. Kot posledica tega bodo geni polovice stikala 2 ojačani. GtgR bo potem represiral pR2. V tej točki je imamo še vedno šibko izražanje GogR iz pRM promotorja. S tem problemom se bomo ukvarjali v koraku 5.

Korak 3: Uvedba anti terminatorjev in ustreznih "nut-site"

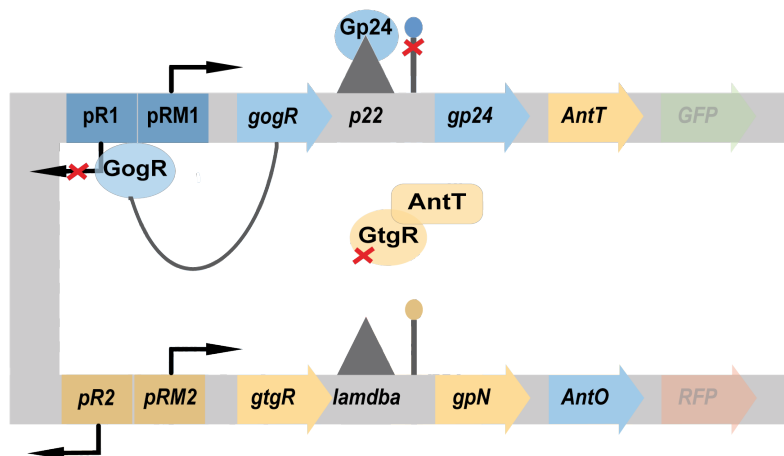
Da se izognemo pomankljivostim sistema bomo uvedli terminator na nižji stopnji represorjev. Poglejte si sliko 5.



Slika 5: Lokacija terminatorjev je prikazana v korelaciji s promotorji in represorji. Anti represorju je izhod onemogočen, ko je terminator funkcionalen.

Če hočemo upravljati s terminatorji in zato tudi s anti represorji, moramo uvesti anti terminatorje, ki so locirani na nižjem nivoju terminatorjev. Anti terminatorji se lahko vežejo na "nut-site" lociran na višjem nivoju terminatorja. Ko se anti terminator veže na "nut-site" se omogoči transkripcija, ki gre mimo terminatorja.

Anti terminator GpN in njegov ustrezeni "nut-site" sta del lambda fage regulatornega sistema. Anti terminator Gp24 in njegov ustrezeni "nut-site" sta del p22 fage regulatornega sistema.



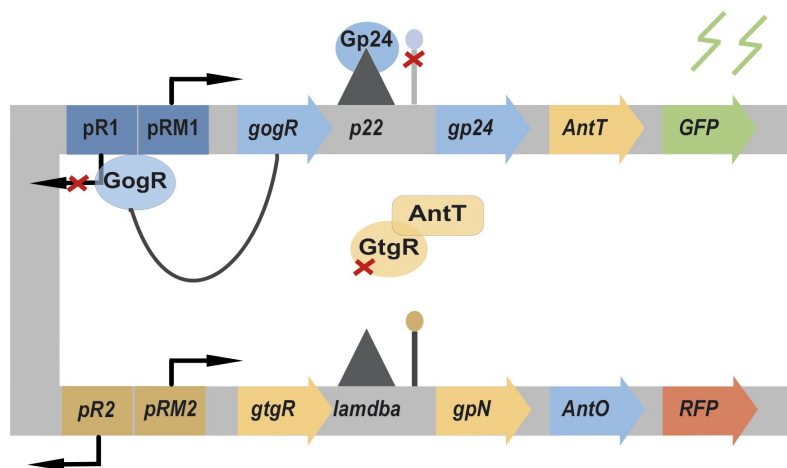
Slika 6: Anti terminator iz polovice stikala 1 ima aktiven izhod in zato se anti represorju AntT aktivira izhod. Rezultat tega obnašanja je stabilnost polovice stikala 1.

Anti terminatorji in "nut-site" pripomorejo k stabilnosti aktivnega stanja sistema. Če je bila transkripcija omogočena za nadaljevanje v anti represor lociran na neaktivni polovici stikala, lahko stikalo spremeni stanje kdaj koli. Terminator zagotavlja, da se to ne zgodi. Anti terminator aktivnega stanja je izražen, kar pomeni, da dovoljuje nadaljevanje transkripcije prek terminatorja.

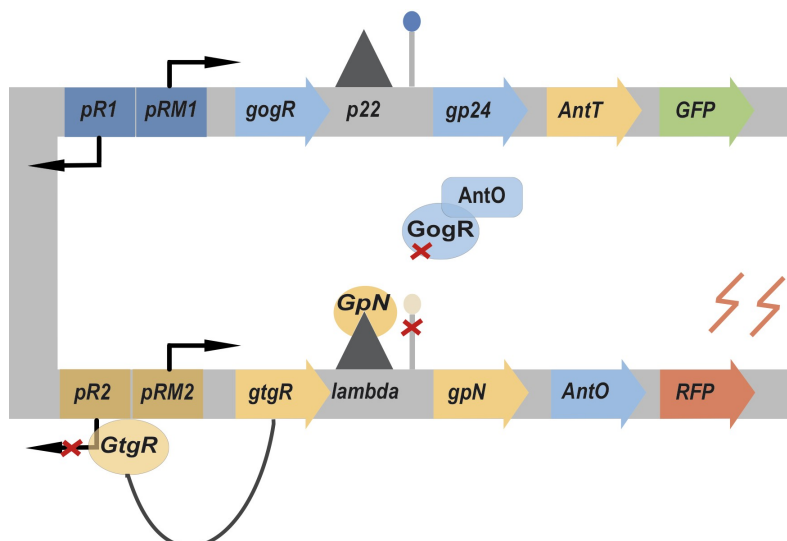
Korak 4: Končni dizajn

Na sliki 7 je prikazan končni dizajn stikala, ko je polovica stikala 1 stabilna. GogR represor represira pR1 promoterja in zato je izraz genov na nižjem nivoju pRM2 promoterja izključno nadziran s strani tega šibkega promotorja. Izražen GtgR bo anti represiran s strani AntT in geni na nižjem nivoju "nut-site" ne bodo izraženi zaradi terminatorja.

Izhod na anti terminatorju Gp24 naredi omogoči transkripcijo skozi terminator in zato je izhod na reporterju GFP izražen.



Slika 7: Končni dizajn stikala. Prva polovica stikala je v stabilnem stanju in reporter GFP daje izhod.



Slika 8: Končni dizajn stikala. Druga polovica stikala je v stabilnem stanju iz zato reporter RFP daje izhod.

Korak 5: Vzbujanje sistema

Sedaj, ko lahko ohranjamo stikalo v obeh stanjih stabilno, so potrebni mehanizmi za spreminjanje stanja. Zato je ekipa uvedla dva vhodna plazmida prikazana na sliki 9. Plazmida vsebujeta anti terminatorje in anti represorje za obe stanji stikala, ki sta postavljena pred dva različna vzbudljiva promotorja.



Slika 9: Vhodna plazmida, ki vsebujeta vzbudljiva promotorja, anti terminator primeren za prvo polovico stikala in anti represor primeren za drugo polovico stikala.

Ko je plazmid za stanju 1 vzbujen, je stikalo prisiljeno preiti v stanje 1. To se zgodi zato, ker se generirata anti terminatorski in anti represorski proteini tega stanja. Anti represor naredi represor stanja 2 neaktiven, rezultat tega pa je povečana transkripcija skozi stanje 1. Istočasno anti terminator prepusti nadaljevanje transkripcije skozi terminator stanja 1. Tam so tudi locirani geni za isti anti represor in anti terminator. Tako je torej z dovolj intenzivnim vzbujanjem stikalo zmožno menjati stanje.

3. Matematično modeliranje

Za pravilno delovanje strukture je potrebno analizirati in upoštevati vse vplive, ki lahko spremenijo željeno delovanje sistema. S pomočjo matematičnega modeliranja na podlagi kemijskih lastnosti okolja, proteinov in ostalih elementov biološkega modela, lahko ugotovimo spreminjanje količin in koncentracij kemijskih vrst v času. Torej matematični model integrira posamezne dele v celoto, saj vsak posamezni gradnik doprinese k večjemu številu interakcij v celici.

V praksi se uporabljajo deterministični modeli in stohastični modeli. Katerega modela bomo izbrali je odvisno od število celic v populaciji. Če imamo opravka z vplivom šuma v eni sami celici se uporablja stohastični model, če pa imamo več celic v sistemu se uporablja deterministični model. V našem primeru smo postavili stohastični model biološkega sistema, ki je prikazan na sliki 1.

3.1. Stohastični model

Dinamika v posamezni celici vključuje interakcijo manjšega števila molekul, kar se lahko izrazi v visoki ravni šuma tudi pri sintezi enega samega proteina. Model, ki ta problem upošteva pri analizi, imenujemo *stohastičen model*.

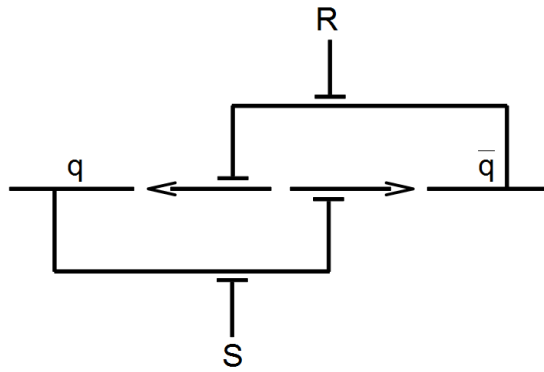
Če imamo definiran deterministični model biološkega sistema, ki vsebuje N vrst, opisan z diferencialnimi enačbami :

$$\frac{d\bar{x}_i}{dt} = f_i(\bar{x}_1, \dots, \bar{x}_N) - g_i(\bar{x}_1, \dots, \bar{x}_N), i=1, \dots, N$$

kjer je $f_i(\bar{x}_1, \dots, \bar{x}_N)$ proces povečanje vrednosti \bar{x}_i , ki pripada vrsti S_i , in $g_i(\bar{x}_1, \dots, \bar{x}_N)$ proces zmanjševanje vrednosti \bar{x}_i , ki pripada vrsti S_i . tuki predstavlja normalno porazdeljena koncentracija vrsti S_i , medtem ko pri stohastičnih modelih x_i predstavlja molekularno število vrsti S_i . Lahko predpostavimo da so povečanje in zmanjšanje molekularnega števila x_i v časovnem intervalu $[t, t + \tau)$ vzorci Poisson – ovih naključnih spremenljivk, z povprečno vrednosjo $f_i(x_1, \dots, x_N)\tau$ in $g_i(x_1, \dots, x_N)\tau$ ustrezno.

Če upoštevamo zgoraj navedenega, dobimo stohastični sistem :

$$x_i(t+\tau) = x_i(t) + P[f_i(x_1(t), \dots, x_N(t))\tau] - P[g_i(x_1(t), \dots, x_N(t))\tau] \quad (1)$$



Slika 10: Model realizacije SR celice

| R | S | D'q | D'q̄ |
|---|---|-----|------|
| 0 | 0 | q | nq |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 |
| 1 | 1 | x | x |

Tabela 1: Pravilnostna tabela SR celice

Deterministični model je opisan z diferencialnimi enačb:

$$\frac{dq}{dt} = \beta_0 + \frac{\beta}{1 + \left(\frac{R + \bar{q}}{K}\right)^n} - \delta q \quad (2)$$

$$\frac{d\bar{q}}{dt} = \beta_0 + \frac{\beta}{1 + \left(\frac{S + q}{K}\right)^n} - \delta \bar{q} \quad (3)$$

Stohastični model dobimo z upoštevanjem enačbi (1) :

– za diferencialni enačbi (2) dobimo:

$$q(t + \tau) = q(t) + P \left[\left(\beta_0 + \frac{\beta}{1 + \left(\frac{R + \bar{q}(t)}{K}\right)^n} \right) \tau \right] - P[\delta q(t)\tau]$$

– za diferencialni enačbi (3) dobimo:

$$\bar{q}(t+\tau) = \bar{q}(t) + P \left[\left(\beta_0 + \frac{\beta}{1 + \left(\frac{S+q(t)}{K} \right)^n} \right) \tau \right] - P [\delta \bar{q}(t) \tau]$$

3.2. Opis parametrov ki nastopajo v enačbi

β_0 – podaja puščanje vrat (leakiness)

β – podaja maksimalno stopnjo transkripcije proteina

K – koeficient represiranja

n – koeficient kooperativnosti ki nam pove koliko enot vhodnega proteina se mora na enkrat vezati na operator

δ – intenzivnost degradacije proteina

τ – časovni korak

S – koncentracijo proteina ki določa SET vhod

R – koncentracijo proteina ki določa RESET vhod

q – koncentracijo proteina ki določa izhod Q

\bar{q} – koncentracijo proteina ki določa izhod \bar{Q}

3.3. Realizacija v Matlabu

```
N = 1200;           % stevilo korakov
beta0 = 0.02*500;   % podaja puščanje vrat (leakiness)
beta = 4*500;       % podaja maksimalno stopnjo transkripcije proteina
n = 1:10;           % koeficient kooperativnosti ki nam pove koliko enot vhodnega
proteina se mora na % enkrat vezati na operator
K = 1*500;          % koeficient represiranja
delta = 1;          % intenzivnost degradacije proteina
tau = 0.1;          % časovni korak
nq0 = 125;          % zacetne koncentracije notq
```

```

q0 = 2125;          % zacetne koncentracije q

S = [ poissrnd(ones(1,N/2-250)*2125) poissrnd(linspace(2125,125,150))
      poissrnd(ones(1,N/2+100)*125) ]; %koncentracijo proteina ki dolo?a SET vhod
R = [ poissrnd(ones(1,N/2+150)*125) poissrnd(linspace(125,2125,150))
      poissrnd(ones(1,N/2-300)*2125) ]; %koncentracijo proteina ki dolo?a RESET vhod

nq = zeros(10,N);
q = zeros(10,N);

nq(:,1) = nq0;
q(:,1) = q0;

for i = 1:10
    for t=2:N

        nq(i,t) = nq(i,t-1) + poissrnd( beta0 + beta./(1 + ((S(t-1) + q(i,t-1))./K).^n(i)).*tau) - poissrnd(delta.*nq(i,t-1).*tau);
        q(i,t) = q(i,t-1) + poissrnd( beta0 + beta./(1 + ((R(t-1) + nq(i,t-1))./K).^n(i)).*tau) - poissrnd(delta.*q(i,t-1).*tau);

    end;
end;

delay = getDelay(nq,mean(q(3,10:700))*0.9,mean(q(3,10:700))*0.1);

%Koncentracija izhodnih proteinov v odvisnosti casa pri koeficient
%kooperativnosti 3
figure(1);
plot(nq(3,:));
hold on;
plot(q(3,:), 'r');
plot(ones(1,N)*mean(q(3,10:700))*0.9, 'g');
plot(ones(1,N)*mean(q(3,10:700))*0.1, 'g');
title('Izhoda Q(rdeca) in notQ(modra) pri n = 3');
xlabel('cas');
ylabel('koncentracija izhodnih proteinov Q in notQ');

%Koncentracija vhodnih proteinov
figure(2);
plot(R); hold on; plot(S, 'r');
title('Vhoda Q(rdeca) in notQ(modra)');
xlabel('cas');
ylabel('koncentracija vhodnih proteinov');

% Cas preklopa pri n = 3:10
figure(3);
plot(3:10,delay(1,3:end));
title('Cas preklopa v odvisnosti od n');
xlabel('koeficient kooperativnosti');
ylabel('cas preklopa');

%3D graf koncentracije proteina Q v casu, s spreminjanjem n
figure(4);
waterfall(q);
title('3D graf koncentracije proteina Q v casu, s spreminjanjem n');
xlabel('cas');
ylabel('koeficient kooperativnosti n');
zlabel('kocentracija proteina Q');

%3D graf koncentracije proteina notQ v casu, s spreminjanjem n
figure(5);
waterfall(nq);
title('3D graf koncentracije proteina notQ v casu, s spreminjanjem n');
xlabel('cas');
ylabel('koeficient kooperativnosti n');
zlabel('kocentracija proteina notQ');

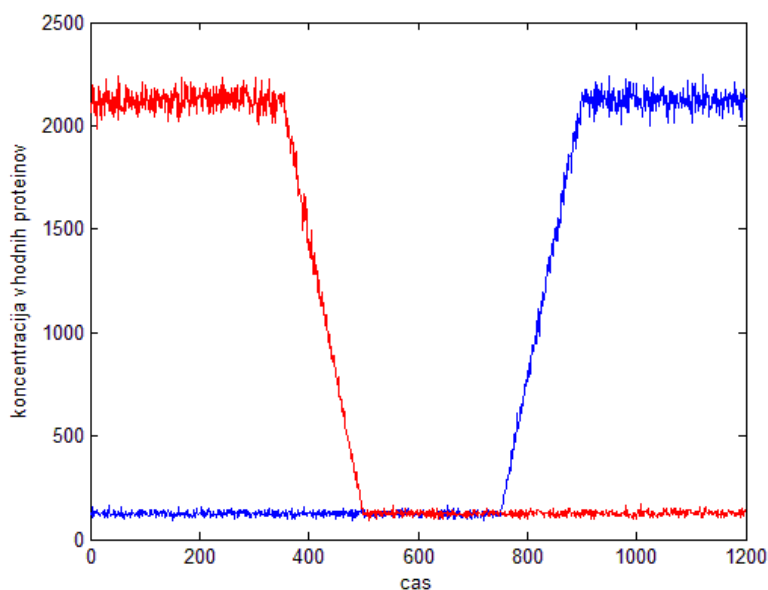
```

4. Simulacija

S pomočjo simulacije smo hoteli raziskati kako spremembe določenih parametrov vplivajo na delovanje modela. Simulacija smo izvedeli v petih scenarijih. V prvem scenariju opazujemo obnašanje modela pri spreminjanju parametra β_0 , ki določa puščanje vrat, v drugem scenariju spreminjamo β maksimalno stopnjo transkripcije proteina, v tretjem koeficienta kooperativnosti n , v četrtem koeficienta represiranja K , ter v zadnjem intenzivnost degradacije proteina. Začetni parametri so povzeti po članku [1]. V vseh teh primerih smo skušali določiti interval vrednosti za ustreznih parameter v katerem model pravilno deluje.

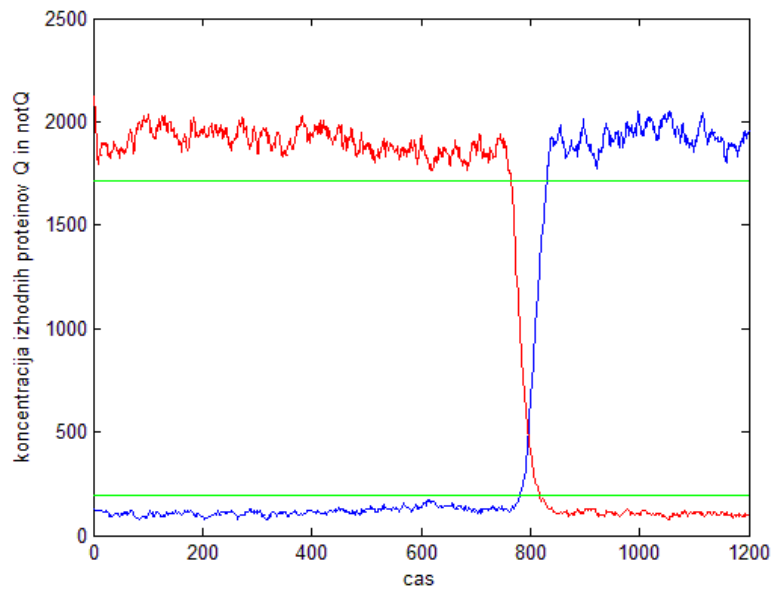
Osnovni parametri so $\beta_0 = 0.02 \mu\text{M}/\text{min}$, $\beta = 4 \mu\text{M}/\text{min}$, $K = 1 \mu\text{M}$, $n = 3$, $\delta = 1 \text{ min}^{-1}$. V članku [1] je navedeno da približno 500 molekul v celici *E. coli* vodi k koncentraciji $1 \mu\text{M}$. Začetno koncentracijo molekul za $q(0) = 2125$, in za $\bar{q}(0) = 125$.

Pri vseh primerih smo uporabili vhoda ki sta prikazana na sliki 11. Vhod S je podan z rdečo barvo, vhod R pa modro barvo.



Slika 11: Koncentracija vhodnih proteinov v odvisnosti časa

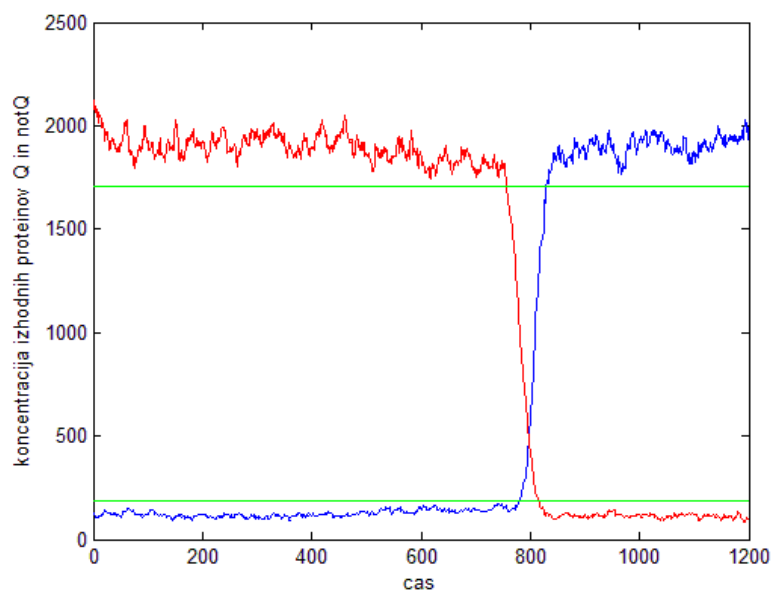
Na sliki 12. je prikazana simulacija pri uporabi osnovnih parametrov. Z rdečo barvo je podana koncentracija izhodnega proteina q , ter z modro je podana koncentracija izhodnega proteina \bar{q} . Zeleni črti določata $10\%C_H$ in $90\%C_H$. C_H – visoka koncentracija.



Slika 12: Koncentracija izhodnih proteinov v odvisnosti časa

4.1. Spreminjanje parametra β_0

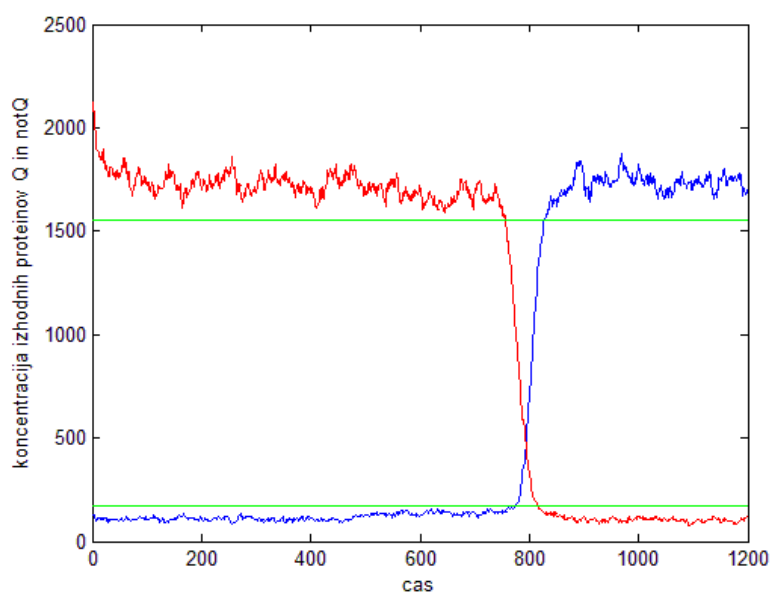
β_0 je parameter ki določa puščanje vrat. V našem primeru je dovoljena vrednost na intervalu (0,0.022). Če je podana vrednost večja kot 0.022 model ne deluje pravilno ker vrednost koncentracije pade v prepovedanem območju tudi takrat ko ni preklopa. Po drugi strani ta parameter ne vpliva na čas preklopa. Primer mejne vrednosti $\beta_0=0.022$ je podan na naslednjem grafu:



Slika 13: Koncentracija izhodnih proteinov v odvisnosti časa, pri $\beta_0 = 0.022$

4.2. Spreminjanje parametra β

S spreminjanjem parametra β spreminjamo maksimalno stopnjo transkripcije proteina. V našem primeru to bi pomenilo spreminjanje maksimalne vrednosti proteina na izhodu. Na grafu to bi pomenilo rastezanje y osi. Če povečamo vrednost parametra β je treba ustrezno povečati izhod ki je v visoko stanje zato da bo začetna vrednost ustrezala visoko stanje. Če pa parametar zmanjšamo dobimo da je $\min(\beta)=3.6\mu\text{M}/\text{min}$. Primer mejne vrednosti $\beta=3.6$ je podan na naslednjem grafu:



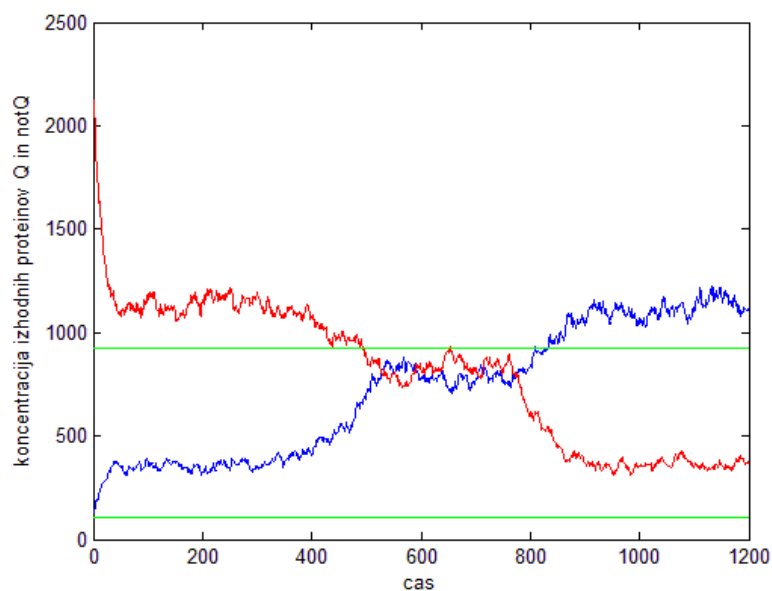
Slika 14: Koncentracija izhodnih proteinov v odvisnosti časa, pri $\beta = 3.6$

4.3. Spreminjanje koeficienta kooperativnosti n

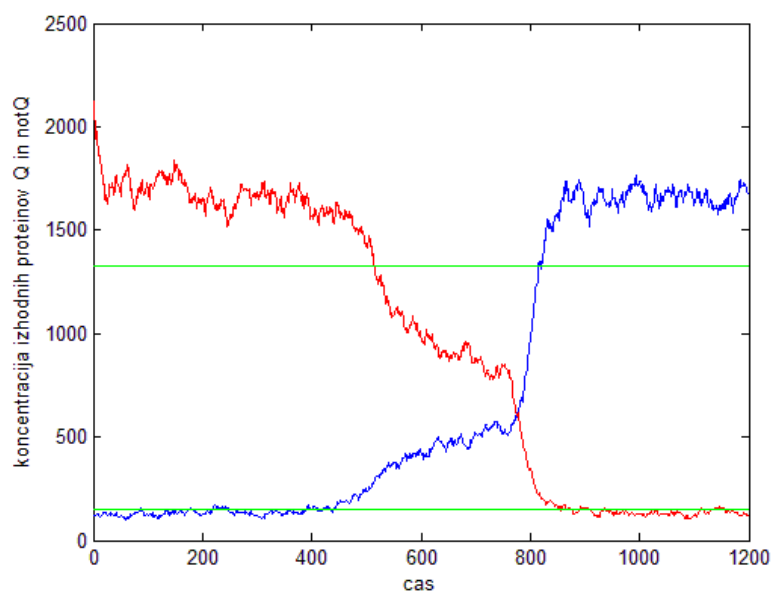
Koeficient kooperativnosti n pomeni koliko enot vhodnega proteina se mora naenkrat vezati na operator. Na sliki 15 in sliki 16 je prikazan odziv na vhoda iz slike 11. Kot lahko opazimo model pri vrednosti za $n = 1$ in $n = 2$ ne deluje, saj koncentracija proteina Q pada pod $90\%CH$ tudi ko ni preklopa. To se zgodi ko gre koncentracija vhodnega proteina S v nizko stanje, torej oba vhoda sta neaktivna. V tem primeru naj bi stikalo ohranjalo stanje, vendar to v primeru ko je $n = 1$ ili $n = 2$, se ne zgodi. Če pa povečamo $n = 3$ ali več vidimo da je delovanje pravilno. Ta primer je prikazan na sliki 17.

S povečanjem n – ja smo opazili da se zmanjšuje tudi čas preklopa. Na sliki 18. je prikazan graf ki podaja čas preklopa v odvisnosti od koeficienta kooperativnosti n. V grafu n se giblje med 3 in 10 ker za $n = 1$ in $n = 2$ model ne deluje.

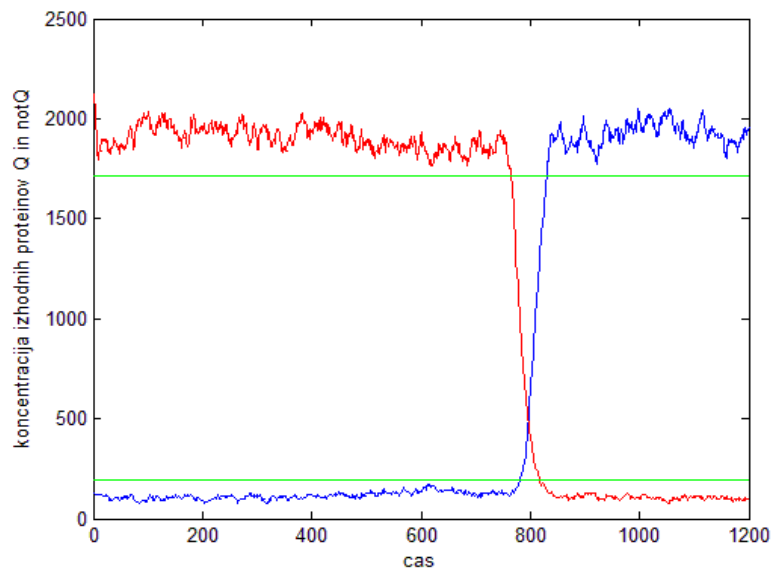
Na sliki 19. ter sliki 20. sta prikazana 3D grafa ki podajata koncentracijo izhoda q in \bar{q} , v odvisnosti od koeficienta kooperativnosti n .



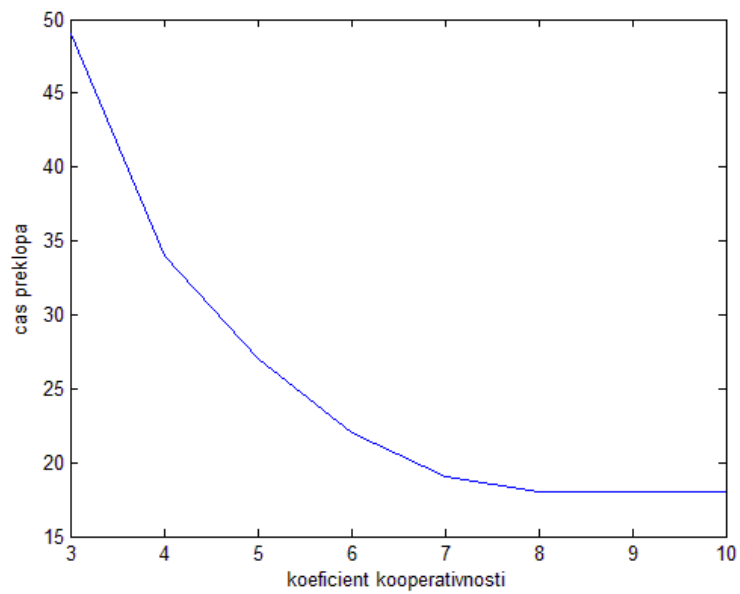
Slika 15: Koncentracija izhodnih proteinov v odvisnosti časa, pri $n = 1$



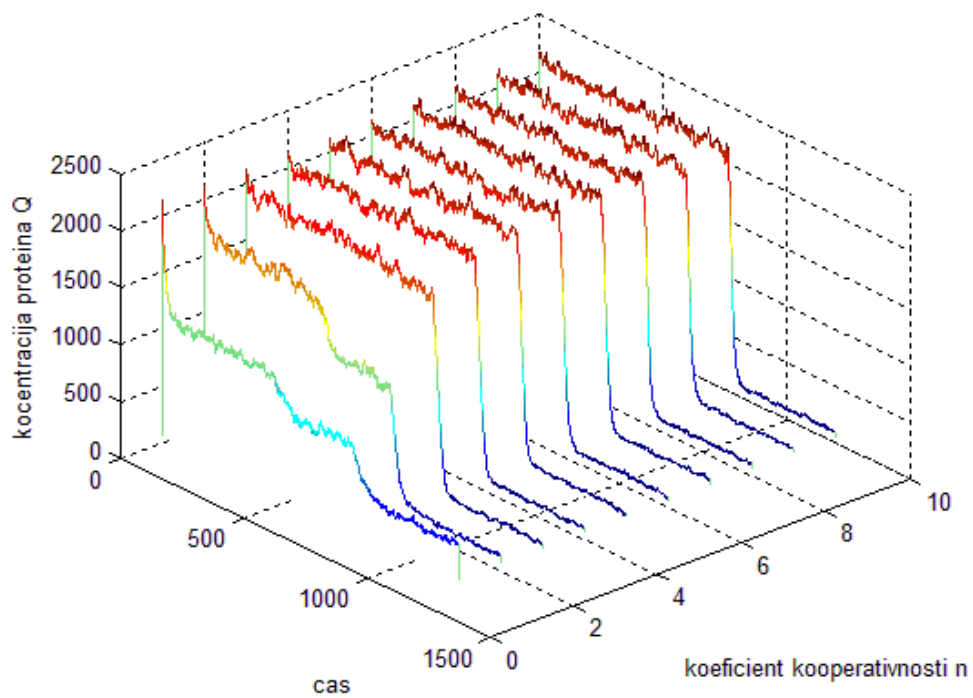
Slika 16: Koncentracija izhodnih proteinov v odvisnosti časa, pri $n = 2$



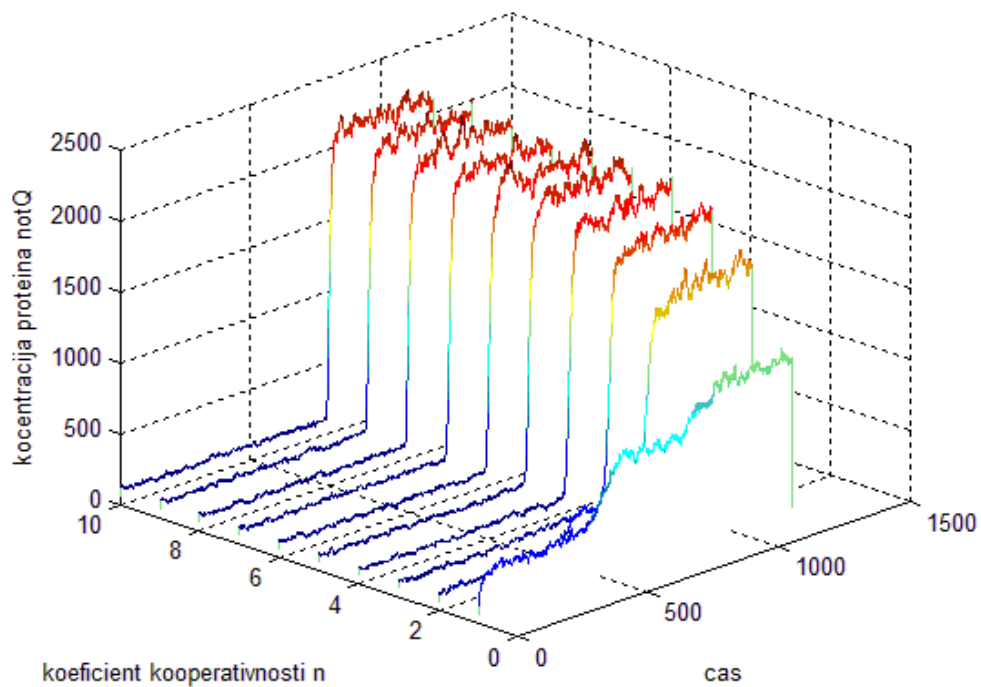
Slika 17: Koncentracija izhodnih proteinov Q in $notQ$ v odvisnosti časa, pri $n = 3$



Slika 18: Čas preklopa v odvisnosti n - ja



Slika 19: 3D graf koncentracije proteina Q v času, s spreminjanjem n

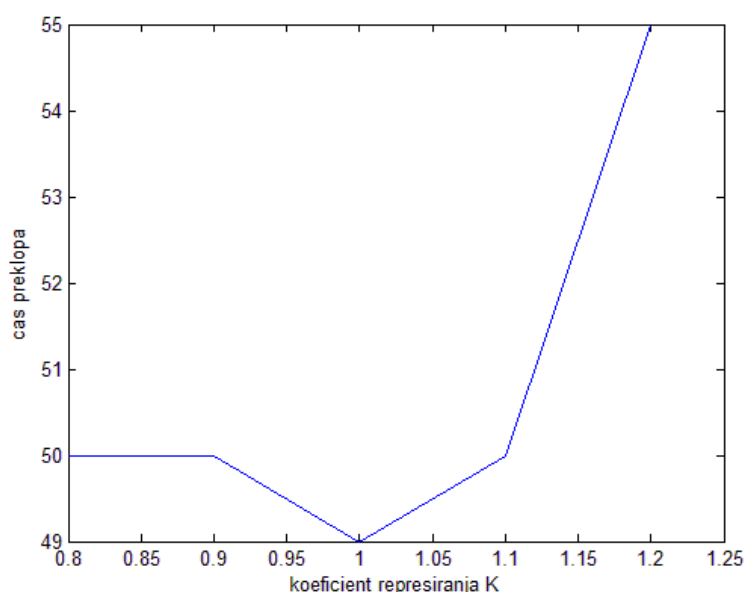


Slika 20: 3D graf koncentracije proteina \bar{Q} v času, s spreminjanjem n

4.4. Spreminjanje koeficienta represiranja K

Spreminjanjem tega parametra močno vpliva na koncentracij izhodnih proteinov q in \bar{q} . Pri testiranju smo določili mejne vrednosti parametra K, pri katerih model še pravilno deluje. K se giblje v intervalu $[0.8, 1.2]$. Če vzamemo vrednost ki je izven tega intervala koncentracije proteinov presežejo meje $90\%C_H$ in $10\%C_H$, tudi ko ni preklopa.

Smo skušali tudi določiti kako spreminjanje parametra K vpliva na čas preklopa. Na sliki 21. je podan graf ki prikazuje kako se spreminja čas preklopa v odvisnosti od parametra K. Kot lahko vidimo je rezultat najboljši pri osnovni vrednosti $K = 1$.



Slika 21: Čas preklopa v odvisnosti K - ja

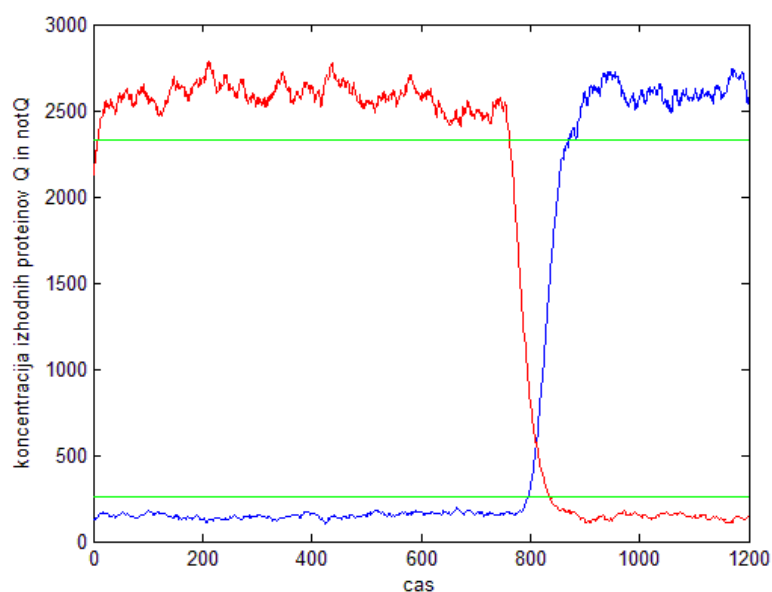
4.5. Spreminjanje parametra δ

Parameter δ določa intenzivnost degradacije proteina. Osnovna vrednost tega parametra je 1, če ga povečamo se degradacija ustreznega proteina poveča, če ga pomanjšamo se degradacija zmanjša. To pomeni da se razteza y - osi pri zmanjšanju parametra, ter skrči pri povečanju. Model deluje pravilno za vrednosti parametra ki so manjši kot 1.3.

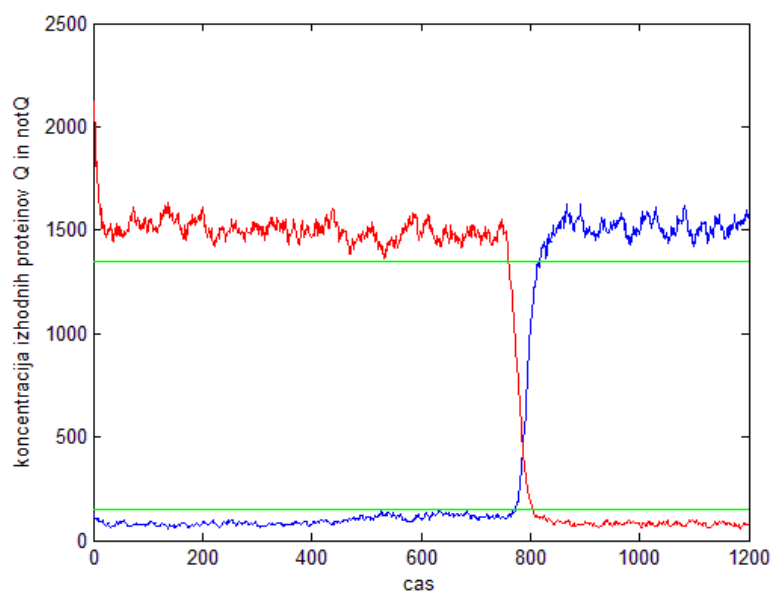
Na sliki 22. in sliki 23 je prikazano delovanje pri $\delta = 0.7$ in $\delta = 1.3$ ustrezno.

Posledično na to da se s spreminjanjem vrednosti parametra, y – os razteza ali skrči, je za pričakovati da bo ta parameter vplival na čas preklopa. Če se os raztegne bo razlika med visoke in nizke koncentracije se poveča, zaradi tega pa se poveča tudi čas preklopa. Če pa se os skrči se zmanjša tudi čas preklopa.

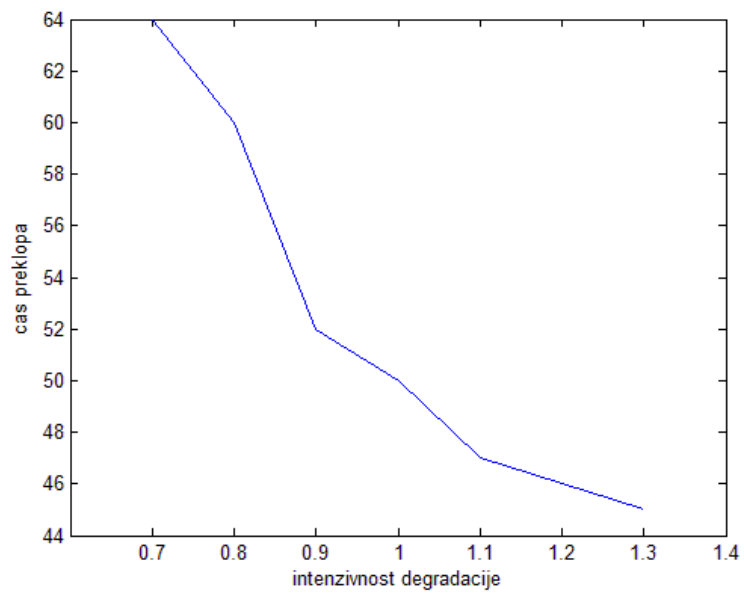
Na sliki 24. je prikazan graf časa preklopa v odvisnosti od parametra δ .



Slika 22: Koncentracija izhodnih proteinov v odvisnosti časa, pri $\delta = 0.7$



Slika 23: Koncentracija izhodnih proteinov v odvisnosti časa, pri $\delta = 1.3$



Slika 24: Čas preklopa v odvisnosti δ

5. Zaključek

Pri analizi tega projekta, ki so ga izvedli Danski študentje, smo se naučili nekatere splošne značilnosti DNK procesiranja, inženirski pristop gradnje nekaterih elementov, ki so analogni obstoječim elektronskim elementom in podrobneje spoznali delovanje tega bistabilnega stikala prek postavitve matematičnega modela in simulacije tega. Poskušali smo najti primerne parametre v modelu, ki so približno podobni realnosti in tudi dajejo najboljše rezultate delovanja bistabilnega stikala. Ugotavljali smo tudi robne vrednosti, ki še omogočajo pravilno delovanje sistema. Vse vrednosti različnih parametrov, ki smo jih testirali ter rezultati simulacije so opisani zgoraj. Ves projekt je le teoretična raziskava, ki nam da občutek, kako se spoprijeti s področjem znanosti, ki je še v povojih. Najtežji del pa bi bila dejanska fizična realizacija tega stikala.

6. Viri

[1] T. Tian and K. Burrage, Stochastic models for regulatory networks of the genetic toggle switch, Proceedings of the National Academy of Sciences, vol.103, pp. 83728377, 2006.

[2] B. Cvetkovič, Analiza možnosti realizacije primitivnih računalniških struktur na osnovi DNK gradnikov diplomska naloga, Univerza v Ljubljani, Fakulteta za računalništvo in informatiko, 2008.

[3] Miha Moškon, Monika Ciglič, Roman Jerala, Nikolaj Zimic, Miha Mraz, Model realizacije funkcionalnosti RS pomnilne celice v biološkem sistemu, Elektrotehniški vestnik XX(Y): 1-6, YEAR

[4] [Bordon, J., Češnovar, R., Petroni, M., Pustoslemšek, R., 2009.](#) Stohastični model preklopnega stikala z uporabo genetskih regulatornih omrežij, seminarska naloga, Ljubljana.

[5] [Mazalin, M., Laharnar, V., Rangus, S., 2009.](#) Stohastični model na podlagi Hillovih enačb, seminarska naloga, Ljubljana.

[6] <http://2010.igem.org/Team:DTU-Denmark>