

Univerza v Ljubljani
Fakulteta za računalništvo in informatiko

Dušan Vučko

**Zasnova hibridnega pristopa za modeliranje
gensko regulatornih omrežij**

DIPLOMSKA NALOGA
NA UNIVERZITETNEM ŠTUDIJU

prof. dr. Miha Mraz
MENTOR

Ljubljana, 2013

© 2013, Univerza v Ljubljani, Fakulteta za računalništvo in informatiko

Rezultati diplomskega dela so intelektualna lastnina Fakultete za računalništvo in informatiko Univerze v Ljubljani. Za objavljanje ali izkoriščanje rezultatov diplomskega dela je potrebno pisno soglasje Fakultete za računalništvo in informatiko ter mentorja.¹

¹V dogovorju z mentorjem lahko kandidat diplomsko delo s pripadajočo izvirno kodo izda tudi pod katero izmed alternativnih licenc, ki ponuja določen del pravic vsem: npr. Creative Commons, GNU GPL.



Št. naloge: 01899/2013

Datum: 05.02.2013

Univerza v Ljubljani, Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Kandidat: **DUŠAN VUČKO**

Naslov: **ZASNOVA HIBRIDNEGA PRISTOPA ZA MODELIRANJE GENSKO
REGULATORNIH OMREŽIJ**

**HYBRID APPROACH FOR THE MODELLING OF GENE REGULATORY
NETWORKS**

Vrsta naloge: Diplomsko delo univerzitetnega študija

Tematika naloge:

Kandidat naj v svojem delu predstavi metode za modeliranje gensko regulatornih omrežij. V nadaljevanju naj kandidat razširi obstoječi deterministični model z možnostjo eksplicitnega modeliranja kompetitivne vezave transkripcijskih faktorjev. Uporabo navedene razširitve naj kandidat predstavi na modeliranju odzivov tipičnih bioloških preklopnih sistemov.

Mentor:

prof. dr. Miha Mraz



Dekan:

prof. dr. Nikolaj Zimic

IZJAVA O AVTORSTVU DIPLOMSKEGA DELA

Spodaj podpisani izjavljam, da sem avtor dela, da slednje ne vsebuje materiala, ki bi ga kdorkoli predhodno že objavil ali oddal v obravnavo za pridobitev naziva na univerzi ali drugem visokošolskem zavodu, razen v primerih kjer so navedeni viri.

S svojim podpisom zagotavljam, da:

- sem delo izdelal samostojno pod mentorstvom prof. dr. Mihe Mraza,
- so elektronska oblika dela, naslov (slov., angl.), povzetek (slov., angl.) ter ključne besede (slov., angl.) identični s tiskano obliko in
- soglašam z javno objavo elektronske oblike dela v zbirki "Dela FRI".

— Dušan Vučko, Ljubljana, marec 2013.

Univerza v Ljubljani
Fakulteta za računalništvo in informatiko

Dušan Vučko

Zasnova hibridnega pristopa za modeliranje gensko regulatornih omrežij

POVZETEK

Pomembno vlogo na področju sintezne biologije imajo gensko regulatorna omrežja (GRO), saj omogočajo realizacijo novih celičnih funkcij. Zaradi pogosto kompleksne dinamike GRO obstaja potreba po njihovem računalniškem modeliranju in simulaciji. V diplomski nalogi predstavimo nov algoritem za modeliranje GRO, imenovan hibridni diskretni algoritem. Ta v sicer deterministični pristop vpelje stohastičnost in bazira na implicitnih pravilih, kar omogoči modularno modeliranje brez potrebe po izpeljavi specifičnih sistemskih enačb. Opišemo deterministični način modeliranja, ki temelji na verjetnostni interpretaciji izražanja genov in uporablja navadne diferencialne enačbe. Predstavimo tudi stohastično modeliranje, ki upošteva stohastičnost izražanja genov in temelji na kemijskih reakcijah. Omenjene tri pristope uporabimo za demonstracijo modeliranja treh eksperimentalno verificiranih GRO - represilatorja in dveh bistabilnih stikal. Vsi modeli uspešno opišejo glavne karakteristike teh GRO.

Ključne besede: gensko regulatorna omrežja, deterministično modeliranje, stohastično modeliranje, hibridni diskretni algoritem, kooperativnost, bistabilnost

University of Ljubljana
Faculty of Computer and Information Science

Dušan Vučko

Hybrid approach for the modelling of gene regulatory networks

ABSTRACT

Gene regulatory networks (GRNs) have an important role in the field of synthetic biology as they make a realization of new cellular functions possible. Because the dynamics of GRNs are often complex, computer modeling and simulation are required. In this thesis, we present a new GRN modeling algorithm, called the hybrid discrete algorithm. It introduces stochasticity into an otherwise deterministic approach and is based on implicit rules that make modular modeling possible, without having to derive specific system equations. We describe a deterministic modeling approach based on probabilistic interpretation of gene regulation that uses ordinary differential equations. We also present stochastic modeling that takes stochasticity of gene expression into account and is based on chemical reactions. We use these three approaches to demonstrate modeling of three GRNs that have been experimentally verified - the repressilator and two bistable switches. All models successfully describe major characteristics of these GRNs.

Key words: gene regulatory networks, deterministic modeling, stochastic modeling, hybrid discrete algorithm, cooperativity, bistability

ZAHVALA

Za pomoč in usmerjanje pri izdelavi diplomske naloge se zahvaljujem mentorju prof. dr. Mihi Mrazu. Za nasvete in dodatno pomoč se zahvaljujem dr. Mihi Moškonu in Tini Lebar. Zahvaljujem se sočlanom in mentorjem slovenske iGEM 2012 ekipe, saj je velik del vsebine naloge posledica idej in izkušenj, pridobljenih ter izmenjanih tekom omenjenega skupinskega projekta. Za možnost nemotenega študija in izdelave te naloge se zahvaljujem staršem.

— Dušan Vučko, Ljubljana, marec 2013.

KAZALO

Povzetek	i
Abstract	iii
Zahvala	v
1 Uvod	1
2 Gensko regulatorna omrežja	5
2.1 Organizacija celice	6
2.2 Izražanje genov	7
2.2.1 Transkripcija	7
2.2.2 Translacija	8
2.3 Reguliranje izražanja genov	9
2.4 Sintezna gensko regulatorna omrežja	11
3 Pristopi k modeliranju gensko regulatornih omrežij	13
3.1 Zakon o vplivu koncentracij	14
3.2 Osnovni matematični opis izražanja gena	15
3.3 Hillove funkcije	17
3.4 Deterministično modeliranje na osnovi verjetnostne interpretacije izražanja genov	21
3.5 Kooperativnost	24
3.6 Primerjava determinističnih in stohastičnih modelov	25
3.7 Stohastično modeliranje	26
3.8 Hibridni diskretni algoritem	28
3.8.1 Objektni model	29

3.8.2	Delovanje algoritma	33
3.9	Določanje vrednosti parametrov	36
4	Zgledi modeliranja s hibridnim diskretnim algoritmom	39
4.1	Represilator	40
4.1.1	Deterministični model	40
4.1.2	Stohastični model	42
4.1.3	Realizacija modela s hibridnim diskretnim algoritmom	43
4.2	Navadno bistabilno stikalo z represorji TAL	47
4.2.1	Deterministični model	48
4.2.2	Stohastični model	51
4.2.3	Realizacija modela s hibridnim diskretnim algoritmom	55
4.3	Bistabilno stikalo s pozitivnima povratnima zankama	56
4.3.1	Deterministični model	58
4.3.2	Stohastični model	60
4.3.3	Realizacija modela s hibridnim diskretnim algoritmom	64
5	Zaključek	67
	Literatura	69

1 Uvod

*"Essentially, all models are wrong,
but some are useful."
– George E. P. Box*

Sintezna biologija je veda, ki se ukvarja z načrtovanjem, izgradnjo in analizo novih, v naravi neobstoječih bioloških sistemov ter funkcionalnosti. To primarno dosega z manipuliranjem delovanja organizmov oz. njihovih posameznih sestavnih delov in logičnim povezovanjem slednjih v nove celote. Sintezna biologija ima izjemen potencial še posebej na področjih, kot so medicina, energija, okolje in prehrana. Primeri prototipnih projektov obsegajo razvoj organizmov za proizvodnjo obnovljivih goriv [1], biosenzorje za odkrivanje prisotnosti toksinov v okolju [2], modificirane celice za kontrolirano proizvodnjo zdravil na obolenem mestu v telesu [3] in detektor pokvarjenosti mesa z namenom zmanjšanja količine nepotrebno zavržene hrane [4].

Celica kot osnovna gradbena in funkcijska enota živih bitij vsebuje genetsko informacijo, nosilec slednje pa je *deoksiribonukleinska kislina* (DNA, angl. *deoxyribonucleic acid*). Sestavni del molekul DNA so zaporedja nukleotidov, ki vplivajo na to, kateri proteini in pri katerih pogojih bodo nastajali znotraj celice. Zaradi njihove vloge v celični dinamiki in posledično organizmu so proteini ključnega pomena - npr. pri človeku sodelujejo pri transportu kisika po krvi, metabolizmu, delovanju mišic in številnih drugih nalogah. Sin-

tezna biologija se za dosego želenega delovanja sistema tipično poslužuje manipuliranja zaporedja DNA, kar ji omogočajo tehnologije za sestavljanje, pomnoževanje, sintezo in sekvenciranje zaporedij DNA (tj. ugotavljanje nukleotidnih zaporedij oz. zgradbe) ter množica biokemijskih, biotehnoloških pa tudi številnih drugih postopkov in tehnologij. Očitno je, da je sintezna biologija izrazito interdisciplinarno področje, ki v klasične vede o življenju vpeljuje inženirski pristop. Posamezen biološki sistem praviloma lahko obravnavamo modularno, kot množico povezanih komponent oz. gradnikov, ki imajo določene lastnosti in se obnašajo po nekih pravilih. Tovrstna abstrakcija vpeljuje tudi prostor za formalno matematično in računalniško obravnavo. Del sintezne biologije tako predstavlja modeliranje in simulacija bioloških sistemov oz. njihovih gradnikov. Omogočata predvidevanje obnašanja načrtovanega sistema in lahko ponudita razlago zakaj, kako ter pod kakšnimi pogoji naj bi ta deloval oz. ne deloval. Neodvisno delovanje osnovnih gradnikov lahko sicer dobro poznamo, a ko te povežemo v večje, kompleksnejše sisteme, dinamika le-teh ni nujno skladna z našimi predvidevanji. Modeliranje tu odigra ključno vlogo, ker omogoči hitrejše in bolj optimalno načrtovanje sistemov z manj eksperimentalnega poskušanja oz. manj popravljanja zasnove na podlagi napak, ki bi jih zaznali šele s posameznimi laboratorijskimi poskusi. Ti so pogosto relativno dragi in zamudni, saj lahko trajajo tudi po več dni. Modeli se sicer lahko z namenom izpopolnjevanja dopolnjujejo z ugotovitvami eksperimentalnih rezultatov - postopek modeliranja in eksperimentalnega dela je prepleten.

Pomemben gradnik sistemov v sintezni biologiji zavzema načrtovanje in izdelava *gensko regulatornih omrežij* (GRO) (angl. *gene regulatory network* - GRN). Posamezno gensko regulatorno omrežje sestavlja množica različnih zaporedij DNA, njihovih pripadajočih produktov - npr. proteinov - in drugih biokemijskih zmesi. Med njimi potekajo biokemijske interakcije (npr. nek protein pospešuje ali zavira tvorbo nekega drugega proteina v omrežju), tako da skupaj tvorijo logično povezano celoto. GRO lahko izkazujejo določene karakteristike, npr. oscilacije koncentracij določenih proteinov ali produkcijo proteina samo ob prisotnosti ustreznih molekul. Z njimi je moč realizirati celično logiko, kjer je lahko visoko logično stanje predstavljeno z npr. visoko koncentracijo nekega proteina. Vstavitev umetnih GRO v celice tako omogoča doseganje številnih novih celičnih funkcij oz. spremembo prvotnega delovanja celice.

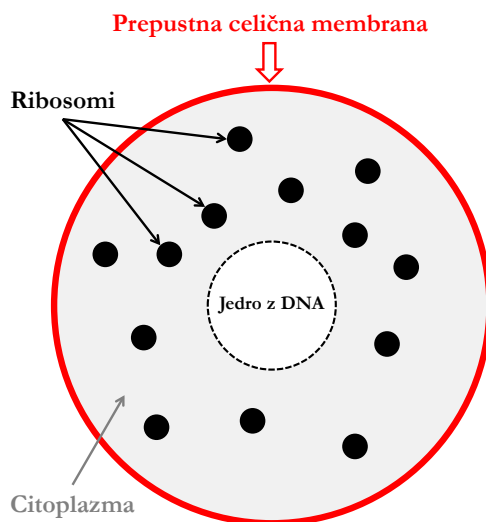
V diplomski nalogi je glavna pozornost posvečena modeliranju GRO. V drugem poglavju so opisane biološke osnove izražanja genov ter koncepti delovanja GRO. V tretjem

poglavju opišemo matematične pristope k modeliranju GRO in ozadje teh pristopov. Predstavljen je deterministični in stohastični način modeliranja, predstavimo pa tudi nov modelirni algoritem, ki omogoča modularno snovanje GRO in vpeljuje stohastičnost v sicer deterministični pristop. V četrtem poglavju so predstavljeni zgledi modeliranja in analize nekaterih eksperimentalno realiziranih GRO. Z rezultati običajnega determinističnega in stohastičnega modeliranja primerjamo rezultate, dobljene z uporabo prej omenjenega novega algoritma. V petem poglavju podamo zaključni pregled rezultatov ter možne smernice za nadaljnje delo.

Velik del vsebine te naloge izhaja iz dela, opravljenega tekom projekta iGEM 2012. V omenjenem okviru je bila realizirana prva različica hibridnega diskretnega algoritma (opisanega v tretjem poglavju), prav tako pa so bili narejeni tudi modeli in opravljena analiza obeh bistabilnih stikal, predstavljenih v četrtem poglavju. Vso omenjeno delo je javno dostopno na [\[3\]](#).

2 Gensko regulatorna omrežja

Da bi razumeli, kako delujejo gensko regulatorna omrežja (GRO), je najprej potrebno razumeti potek genskega izražanja, tj. procesa, pri katerem na osnovi genetske informacije oz. nukleotidnega zaporedja DNA nastajajo genski produkti, npr. proteini. Natančna obravnava izražanja genov iz biološke perspektive je obširna tema, ki presega obseg te naloge. Več o njej si bralec lahko prebere v [5–7]. V tem poglavju so opisani glavni, splošno veljavni koncepti, ki ignorirajo specifične razlike med vrstami organizmov oz. celic, npr. med prokariontskimi in evkariontskimi celicami. Vendar pa s tem ni nič narobe - opisani koncepti namreč zadoščajo za izpeljavo splošno veljavnih pristopov modeliranja GRO. Na tak način lahko dobimo modele, ki zadovoljivo opisujejo glavne lastnosti dinamike GRO v realnem eksperimentalnem okolju oz. v celici. Prvi razlog za to je, da lahko številne razlike med posameznimi tipi celic zajamemo že zgolj z različnimi vrednostmi parametrov modela, ki so določene na podlagi eksperimentalnih rezultatov. Poleg tega se glavne lastnosti dinamike GRO pogosto ohranijo kljub velikim razlikam v vrednostih parametrov. Domena analize GRO je ugotoviti, kako vrednosti posameznih parametrov vplivajo na dinamiko GRO.



Slika 2.1 Poenostavljena shema strukture celice. Vse celice imajo prepustno membrano, citoplazmo z ribosomi in genetski zapis v obliki molekul DNA. Ribosomi skrbijo za sintezo proteinov; kateri proteini se bodo tvorili pa je v splošnem odvisno od nukleotidnega zaporedja DNA.

2.1 Organizacija celice

Vsi živi organizmi so sestavljeni iz celic, ki predstavljajo njihovo osnovno strukturno in funkcijsko biološko enoto [8]. Obstajajo različne vrste celic, ki se razlikujejo tako po strukturi kot funkciji, vsem pa je skupno, da imajo celično membrano, citoplazmo in genetsko informacijo v obliki zapisa DNA [5]. Celična membrana je ovoj, ki ločuje vsebino celice od okolja in omogoča selektivno prepuščanje določenih snovi v celico in iz nje. Membrana obkroža citoplazmo, ki jo sestavljata celična tekočina citosol ter organeli oz. notranje komponente celice, ki opravljajo različne naloge. Med temi komponentami pomembno vlogo igrajo ribosomi, ki skrbijo za sintezo proteinov na način, ki je natančneje opisan v nadaljevanju poglavja. Pomembna komponenta v številnih celicah so še proteasomi, ki so potrebni za razgradnjo oz. degradacijo proteinov v aminokisline (sicer sestavne gradnike proteinov). Celice vsebujejo jedro z zapisom DNA. Jedro evkariontskih celic je v nasprotju z jedrom prokariontskih celic (imenovanem nukleoid) obdano z membrano oz. jedrno ovojnico, ki ga ločuje od citoplazme. Shemo organizacije celice prikazuje slika 2.1.

2.2 Izražanje genov

Osnovni sestavni del zapisa DNA so organske molekule, imenovane nukleotidi [9]. Molekulo DNA sestavljata dve zaporedji oz. verigi nukleotidov v obliki dvojne vijačnice. Sestavni del nukleotida DNA je ena izmed dušikovih baz, ki je lahko adenin (A), gvanin (G), citozin (C) ali timin (T). Nukleotidi ene verige tvorijo povezave oz. bazne pare z nukleotidi nasprotno verige; pri tem A vedno nastopa v paru s T, C pa v paru z G. Genetska informacija je določena z vrstnim redom nukleotidnega zaporedja. Slika 2.2 prikazuje shemo zaporedja baznih parov v molekuli DNA.

Aminokisline so molekule, ki se tvorijo glede na zaporedje DNA. Proteini so molekule, ki jih sestavljajo zaporedja različnih aminokislin. Proteini igrajo pomembno vlogo pri številnih celičnih procesih. Vir [7] kot glavne kategorije proteinov navaja:

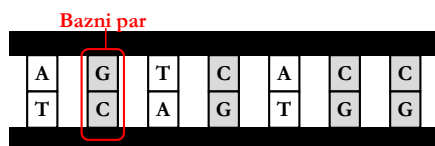
1. proteine, udeležene pri izražanju genov in regulaciji izražanja genov;
2. motorične proteine, ki sodelujejo pri gibanju;
3. proteine za obrambo organizma pred boleznimi;
4. metabolične encime, ki pospešujejo hitrost kemičnih reakcij;
5. proteine za komunikacijo med celicami;
6. transportne proteine za npr. transport glukoze kot vira energije v celico.

Zapis DNA je razdeljen na gene - podzaporedja DNA, ki pri izražanju genov vodijo v nastanek genskega produkta oz. proteina [7]. Izražanje genov je torej proces, pri katerem na osnovi genetske informacije nastane protein oz. v splošnem genski produkt. Izražanje genov sestavljata dva glavna koraka, ki jih upodablja slika 2.3:

1. transkripcija, kjer se na osnovi zaporedja DNA tvorijo molekule *sporočilne ribonukleinske kisline* (mRNA, angl. *messenger ribonucleic acid*);
2. translacija, kjer se na osnovi strukture molekule mRNA tvorijo zaporedja aminokislin, ki sestavljajo končni protein.

2.2.1 Transkripcija

Transkripcija je proces, pri katerem se ustvari prepis oz. kopija zaporedja DNA v ribonukleinsko kislino (RNA). Zaporedje DNA ki pri tem ostane nespremenjeno. Pri večini



Slika 2.2 Shematična predstavitev dela zaporedja DNA, ki podaja genetsko informacijo. Nukleotidi dveh verig tvorijo zaporedje baznih parov. Pri tem A vedno tvori par s T, C pa z G.



Slika 2.3 Izražanje genov vodi v sintezo genskega produkta, kot je protein. Pri transkripciji encim polimeraza RNA opravi prepis zaporedja DNA v zapis mRNA. Pri translaciji ribosomi omogočijo, da se na osnovi zaporedja mRNA sintetizira veriga aminokislin, ki predstavlja končni protein.

genov RNA vsebuje informacijo za sintezo proteina v obliki zaporedja aminokislin - tako RNA imenujemo mRNA oz. sporočilna RNA [7]. Zaporedje DNA, ki je podvrženo transkripciji, je omejeno na eni strani z nukleotidnim zaporedjem, imenovanim promotor, na drugi strani pa z zaporedjem, imenovanim terminator. V mRNA se torej prepíše območje med promotorjem in terminatorjem. Promotor, zaporedje za prepis in terminator skupaj predstavljajo gen, prikazan na sliki 2.4.

Za transkripcijo je nujno potrebna polimeraza RNA. To je encim, ki se veže na promotor gena, nato pa drsi po DNA in sintetizira mRNA. Ko polimeraza RNA doseže terminator, sprostí DNA in nastalo mRNA. Nastala mRNA je enojna veriga in ima enako zaporedje kot prepisani del DNA, le da v njej namesto timina (T) nastopa baza uracil (U).

2.2.2 Translacija

Translacija je tvorba proteina oz. zaporedja aminokislin v citoplazmi na osnovi nukleotidnega zaporedja mRNA. Za translacijo je potreben encim ribosom. mRNA ima vezavno mesto za ribosom, kamor se slednji veže pred začetkom translacije. Nato se ribosom premika po mRNA in tvori zaporedje aminokislin; pri tem je vsaka aminokislina določena s trojico nukleotidov mRNA, ki ji pravimo kodon. Translacija poteka od t.i. start kodona, ki določa prvo aminokislino, do stop kodona, kjer ribosom sprostí DNA in sproducirano zaporedje aminokislin. Organizacijo mRNA prikazuje slika 2.5.



Slika 2.4 Osnovna struktura gena. Promotor je zaporedje DNA, kamor se veže polimeraza RNA in začne transkripcijo, tj. prepis zaporedja DNA v mRNA. Transkripcija se konča, ko polimeraza doseže terminator.



Slika 2.5 Shema organizacije mRNA. Translacijo vrši ribosom, ki se najprej veže na ustrezno vezavno mesto in nato pomika po mRNA. Vsak kodon - trojica nukleotidov - se prevede v aminokislino, ki ustreza nukleotidnemu zaporedju kodona. Tako nastaja veriga aminokislin, ki ustreza vrstnemu redu kodonov. Prva aminokislina je določena s start kodonom, translacija pa se konča, ko ribosom naleti na stop kodon.

2.3 Reguliranje izražanja genov

Eden od mehanizmov za regulacijo izražanja genov so transkripcijski faktorji. To so proteini, ki vplivajo na vezavo polimeraze RNA na promotor in s tem na samo transkripcijo gena. Transkripcijski faktorji nastopajo v dveh oblikah:

1. *aktivatorji* so transkripcijski faktorji, ki pospešujejo vezavo polimeraze na promotor in s tem učinkovito povečujejo transkripcijo oz. njeno intenzivnost;
2. *represorji* so transkripcijski faktorji, ki zavirajo (represirajo) vezavo polimeraze na promotor in s tem zavirajo transkripcijo oz. zmanjšujejo njeno intenzivnost.

Rezultat vezave aktivatorja je zvišanje števila produciranih molekul mRNA v enoti časa, rezultat vezave represorja pa zmanjšanje števila produciranih mRNA molekul v enoti časa. Transkripcijski faktorji aktivacijo ali represijo izvajajo tako, da se vežejo na specifično zaporedje DNA v ali ob promotorju gena, ki ga regulirajo. Tako zaporedje DNA za vezavo transkripcijskega faktorja imenujemo vezavno mesto. Vezavnih mest je lahko več za različne transkripcijske faktorje in lahko obsegajo različno število ponovitev - s tem se doseže večja učinkovitost vezave. Če je ob promotorju prisotno tako vezavno mesto za aktivator kot tudi za represor, se lahko izkaže, da ima vezava represorja večji učinek na transkripcijo - kljub vezanemu aktivatorju bo že manjša količina vezanega represorja vodila v občutno represijo transkripcije. Z drugimi besedami, represija ima lahko prioriteto pred aktivacijo. V odnosu do transkripcijskih faktorjev je potrebno iz logičnega vidika razlikovati dva tipa promotorjev, ki jih bomo uporabljali v nadaljevanju

naloge. To so:

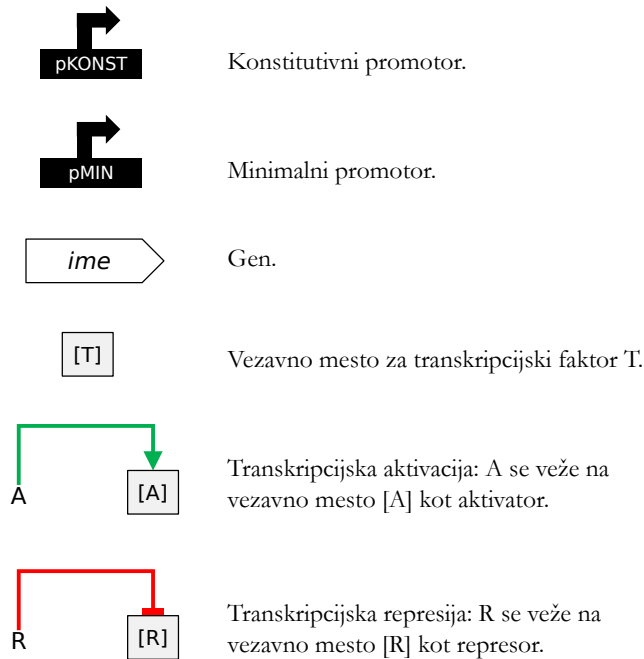
1. *konstitutivni promotorji* in
2. *minimalni promotorji*.

Konstitutivni promotor je promotor, pri katerem transkripcija genov pod njim poteka pri konstantni oz. neregulirani hitrosti že ob odsotnosti transkripcijskih faktorjev. Geni, vezani na konstitutivni promotor, bodo podvrženi t.i. konstitutivni transkripciji. Represija konstitutivnega promotorja bo normalno povzročila opazno znižanje produkcije mRNA. Idealno bi z vezavo zadostne količine represorja želeli doseči popolno represijo gena pod konstitutivnim promotorjem, tj. ničelno efektivno hitrost transkripcije. Vendar v praksi ne glede na količino vezanega represorja pogosto še vedno prihaja do produkcije nekega nivoja mRNA. Ta nivo je sicer lahko izredno nizek napram nerepresirani transkripciji, a lahko odločilno vpliva na logiko delovanja gensko regulatornega omrežja. Pojav, pri katerem konstitutivne transkripcije ne moremo zmanjšati pod določen minimum kljub večji količini represorja, imenujemo *puščanje*.

Minimalni promotor za povečanje intenzivnosti transkripcije zahteva vezavo aktivatorja. Brez prisotnosti aktivatorja je transkripcija neizrazita oz. v idealnem primeru nična. Vezan represor lahko intenzivnost transkripcije pri minimalnem promotorju v splošnem še dodatno zmanjša v primerjavi s stanjem neprisotnosti aktivatorja, vendar spet le do nekega minimuma oz. puščanja. Večanje koncentracije aktivatorja po drugi strani seveda ne pomeni, da lahko intenzivnost transkripcije zvišujemo v neskončnost, saj se prej ali slej doseže nek maksimum.

Pomembno odkritje na področju transkripcijskih faktorjev so *efektorji TAL* (angl. *transcription activator-like (TAL) effectors*) [10]. Gre za proteine, ki so sposobni prepoznave in vezave na specifično zaporedje DNA. S fuzijo represorske domene na vezavno domeno efektorja TAL lahko ustvarimo represorje TAL, s fuzijo aktivatorske domene pa aktivatorje TAL [3]. Posebej uporabna je možnost izdelave efektorjev TAL, ki so sposobni prepoznavati nova zaporedja DNA, torej nova vezavna mesta za reguliranje transkripcije.

Izražanje genov lahko reguliramo tudi na druge načine. Pri npr. post-transkripcijski regulaciji vplivamo na mRNA in s tem na translacijo. Genski produkti oz. molekule, kot so npr. *siRNA* (angl. *short-interfering RNA*) in *miRNA* (angl. *microRNA*) lahko povzročijo inhibicijo translacije obstoječe specifične mRNA. Ta mehanizem imenujemo *interferenca RNA* [7].



Slika 2.6 Grafični simboli pogostih gradnikov gensko regulatornih omrežij.

2.4 Sintezna gensko regulatorna omrežja

Opisane celične fenomene in lastnosti lahko izkoristimo za izdelavo novih, v naravi neobstoječih zaporedij DNA - torej tudi gensko regulatornih omrežij z novo dinamiko izražanja proteinov. To omogoča tehnologija izolacije, cepljenja, lepljenja in pomnoževanja posameznih sekvenc DNA oz. tehnologija rekombinantne DNA, ki se na koncu vnese v ciljni organizem, npr. bakterijsko celico. Cepljenje DNA na specifičnih mestih v zaporedju omogočajo restriksijski encimi, lepljenje kompatibilnih koncev DNA pa encimi, imenovani ligaze DNA. Dodatne možnosti nudi tehnologija sinteze zaporedij DNA. Slika 2.6 podaja grafične simbole, ki so v nadaljevanju naloge uporabljeni kot gradniki gensko regulatornih omrežij.

3

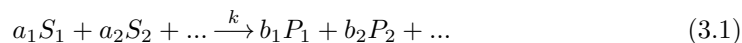
Pristopi k modeliranju gensko regulatornih omrežij

Cilj modeliranja gensko regulatornih omrežij (GRO) je pridobiti formalen opis posameznega omrežja, ki je idealno zmožen natančnega napovedovanja njegovega obnašanja, izhajajočega iz interne strukture DNA in drugih molekularnih interakcij. V praksi se to običajno izkaže za nedosegljiv cilj, saj nimamo na voljo vseh potrebnih podatkov za povsem natančen opis. Kljub temu pa lahko izdelamo model, ki bo vodil v uporabne zaključke in ustrezno podal glavne karakteristike GRO v realnem celičnem okolju. Ko tekom eksperimentalne realizacije na voljo postanejo nove informacije, jih lahko vključimo v model in ga tako izboljšamo oz. po potrebi tudi povsem spremenimo, če se začetne hipoteze izkažejo za neustrezne. V vsakem primeru lahko dano omrežje modeliramo na več različnih načinov. Pri tem velja, da bolj natančna formalizacija ni nujno tudi bolj uporabna - iščemo model, ki bo dovolj dober za namen modeliranja in analize sistema. Tipično nas zanima dinamika spremenljivk - npr. koncentracij mRNA in proteinov - pod določenimi pogoji, npr. ob prisotnosti ali odsotnosti določenih molekularnih vhodov. Kljub različnim pristopom k modeliranju običajno vedno izhajamo iz matematičnega opisa izražanja genov, ki primarno temelji na empiričnih ugotovitvah.

3.1 Zakon o vplivu koncentracij

Dinamiko gensko regulatornega omrežja lahko opišemo s kemijskimi reakcijami oz. enačbami.

V splošnem lahko vsako kemijsko reakcijo zapišemo z izrazom



Pri tem so $S_1, S_2, \dots$ reaktanti, $P_1, P_2, \dots$ produkti, parametri $a_1, a_2, \dots, b_1, b_2, \dots$ pa stehiometrični koeficienti. k je konstanta hitrosti kemijske reakcije. Pomembno predpostavko v številnih modelih GRO predstavlja *zakon o vplivu koncentracij* (angl. *law of mass action*), ki pravi, da je hitrost kemijske reakcije proporcionalna produktu koncentracij reaktantov [11, 12]. Z drugimi besedami to pomeni, da je verjetnost kemijske reakcije proporcionalna verjetnosti medsebojnega trka reaktantov. Zakon pogosto odraža eksperimentalne razmere v primeru dobro premešane raztopine. Za splošno reakcijo (3.1) je hitrost reakcije v pogojena z izrazom

$$v = k \prod_i [S_i]^{a_i}. \quad (3.2)$$

Pri tem $[\cdot]$ označuje koncentracijo. Eksponent posameznega reaktanta imenujemo *red reaktanta*. Za enostavno reakcijo, kjer se reaktant A porabi in pretvori v produkt B, kar zapišemo kot



je hitrost reakcije pogojena z izrazom

$$v = k \cdot [A] \quad (3.4)$$

in ima za ta primer enote $\text{mol l}^{-1} \text{s}^{-1}$, kjer en mol znaša $6,02 \cdot 10^{23}$ molekul. Podobno je za reakcijo



hitrost reakcije enaka

$$v = k \cdot [A] \cdot [B]. \quad (3.6)$$

Upoštevajoč hitrost reakcije lahko zapišemo diferencialne enačbe, ki podajajo spreminjanje koncentracij skozi čas:

$$\frac{d[A]}{dt} = \frac{d[B]}{dt} = -k \cdot [A] \cdot [B], \quad (3.7)$$

$$\frac{d[C]}{dt} = k \cdot [A] \cdot [B]. \quad (3.8)$$

Enačbo (3.7) lahko interpretiramo kot porabljanje reaktantov A in B s hitrostjo k , enačbo (3.8) pa kot nastajanje produkta C s hitrostjo k . Konstanto hitrosti kemijske reakcije lahko po potrebi (npr. zaradi spremenljivega okolja) nadomestimo s funkcijo spremenljivk, ki vplivajo na njeno vrednost [11]. Zakon o vplivu koncentracij je osnova za izdelavo modelov, ki temeljijo na navadnih diferencialnih enačbah (NDE) (angl. *ordinary differential equations* - ODE).

3.2 Osnovni matematični opis izražanja gena

Če stanje posameznega gena pod nekim promotorjem obravnavamo binarno, je gen *aktiven*, če se izraža (na osnovi njegovega zapisa DNA se producira genski produkt oz. protein) in *neaktiven*, če se ne. V nadaljevanju naloge je z izražanjem gena vedno mišljena transkripcija gena v mRNA in translacija mRNA v končni protein, ki ga gen kodira. Ali bo nek gen aktiven ali ne, je odvisno od stanja promotorja gena. Če stanje promotorja obravnavamo binarno, velja, da je gen aktiven, če je tudi njegov promotor *aktiven*. Če je promotor *neaktiven*, so neaktivni tudi vsi geni pod njim - njihovo izražanje dosega kvečjemu nivo puščanja. Minimalni promotor je aktiven, če ima ob sebi (na vezavnem mestu) vezan aktivator. V primeru vezanega represorja ali odsotenga aktivatorja je neaktiven. Konstitutivni promotor je aktiven, če je *prost* - to pomeni, da na vezavna mesta ob njem ni vezan noben transkripcijski faktor (niti aktivator niti represor). Aktiven je tudi, če je ob njem vezan aktivator (brez represorja), ki učinkovito še pohitri transkripcijo. Konstitutivni promotor je neaktiven, če je ob njem vezan represor.

Če predpostavimo, da je promotor vedno aktiven, lahko izražanje gena opišemo z reakcijami:



Reakcija (3.9) opisuje produkcijo mRNA, torej transkripcijo (prazna leva stran reakcije pomeni, da se pri njej ne porablja noben reaktant). Reakcija (3.10) predstavlja translacijo - produkcijo proteina P iz nastale mRNA. Zaradi translacije same se mRNA ne porablja, zato nastopa tako kot reaktant kot tudi produkt. Reakciji (3.11) in (3.12) podajata

degradacijo mRNA in proteina P (prazna desna stran reakcije pomeni, da pri njej ne nastaja noben produkt). Če predpostavljamo zakon o vplivu koncentracij, lahko pišemo:

$$\frac{d[mRNA]}{dt} = k_1 - q_1 \cdot [mRNA], \quad (3.13)$$

$$\frac{d[P]}{dt} = k_2 \cdot [mRNA] - q_2 \cdot [P]. \quad (3.14)$$

Konstanta k_1 določa intenzivnost oz. hitrost transkripcije, k_2 hitrost translacije, q_1 hitrost degradacije mRNA in q_2 hitrost degradacije nastalega proteina. Graf poteka koncentracij mRNA in proteina v odvisnosti od časa prikazuje slika 3.1. V splošnem velja, da je:

$$\frac{d[mRNA]}{dt} = \text{hitrost produkcije mRNA} - \text{hitrost degradacije mRNA}, \quad (3.15)$$

$$\frac{d[P]}{dt} = \text{hitrost produkcije P} - \text{hitrost degradacije P}. \quad (3.16)$$

Formalno gledano se mRNA poleg v reakciji (3.9) producira še v reakciji (3.10), kjer pa se tudi porablja, zato se učinek te reakcije na mRNA izniči in tako pridemo do enačbe (3.13):

$$\frac{d[mRNA]}{dt} = k_1 + k_2 \cdot [mRNA] - k_2 \cdot [mRNA] - q_1 \cdot [mRNA] = \quad (3.17)$$

$$= k_1 - q_1 \cdot [mRNA]. \quad (3.18)$$

Prvi člen predstavlja produkcijo mRNA v reakciji (3.9), drugi člen produkcijo mRNA v reakciji (3.10), tretji člen porabljanje mRNA v reakciji (3.10) in četrti člen degradacijo mRNA v reakciji (3.11). V reakcijah (3.9) - (3.12) bi lahko po potrebi eksplicitno upoštevali tudi druge parametre, npr. polimerazo RNA kot pogoj za transkripcijo.

Sistem enačb (3.13) in (3.14) določa *stabilno stanje*, če velja:

$$\frac{d[mRNA]}{dt} = 0, \quad (3.19)$$

$$\frac{d[P]}{dt} = 0. \quad (3.20)$$

Hkrati mora biti pri tem izpolnjen pogoj, da je hitrost vsaj ene kemijske reakcije, ki opisuje sistem, različna od nič, sicer ne gre za stabilno stanje, pač pa za termodinamično ravnovesje [12]. V stabilnem stanju se koncentracije kemijskih zvrsti s časom ne spreminjajo več. Iz enačb (3.13) in (3.19) sledi, da koncentracija mRNA v stabilnem stanju doseže maksimalno vrednost oz. nivo $mRNA_{max} = \frac{k_1}{q_1}$. Podobno koncentracija proteina

v stabilnem stanju (ob predpostavki stabilnega stanja mRNA) doseže maksimalni nivo $P_{max} = \frac{k_1 \cdot k_2}{q_1 \cdot q_2}$.

Nekatera GRO izkoriščajo časovni zamik med transkripcijo in translacijo, npr. za dosego oscilacij koncentracije proteina. V številnih primerih pa je model še vedno dovolj verodostojen, če koraka transkripcije in translacije združimo v en sam korak produkcije proteina na osnovi zapisa DNA. Pri tem se je potrebno zavedati razlike med posameznimi tipi gostiteljskih celic gensko regulatornega omrežja. V prokariontih se lahko translacija začne, še preden je končana transkripcija oz. še preden je sproducirana končna molekula mRNA; po drugi strani je v evkariontih transkripcija časovno in prostorsko ločena od translacije [13]. Vzrok za to je v zgradbi celic, saj mora za začetek translacije kompletna mRNA zapustiti jedro in priti v stik z ribosomi. Kljub temu lahko modeliranje transkripcije in translacije kot enega dogodka tudi v evkariontih nudi povsem zadovoljiv opis delovanja GRO. Ob predpostavki aktivnega promotorja in združitvi transkripcije ter translacije, se število reakcij, ki podajajo izražanje gena, zmanjša na naslednji dve:



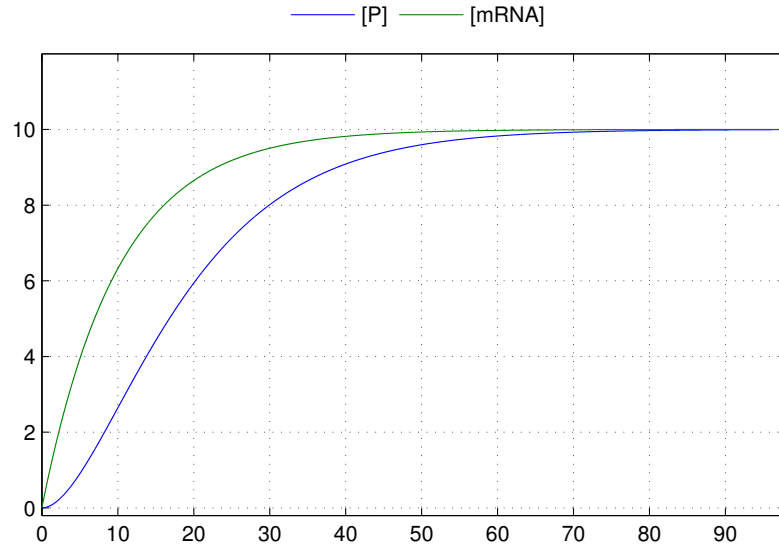
Reakcija (3.21) podaja produkcijo proteina s konstanto hitrosti produkcije k_1 , reakcija (3.22) pa degradacijo proteina s konstanto hitrosti degradacije q_1 . Koncentracija proteina P se s časom spreminja v skladu z diferencialno enačbo

$$\frac{d[P]}{dt} = k_1 - q_1 \cdot [P]. \quad (3.23)$$

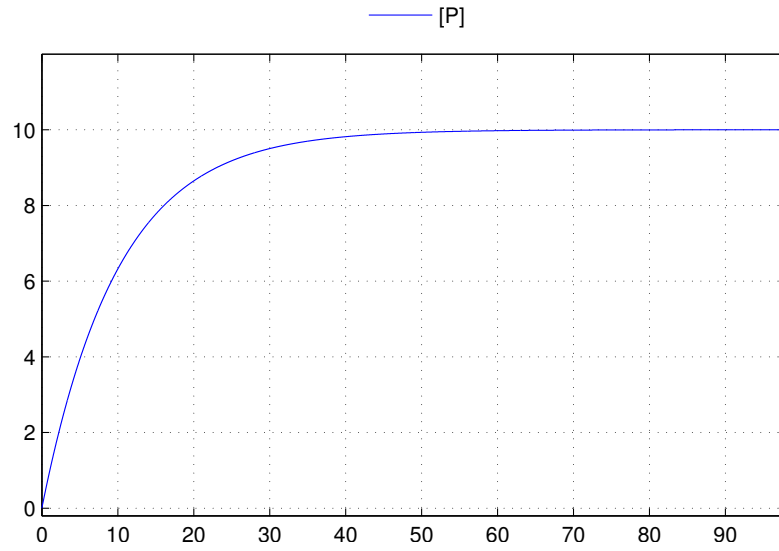
Graf enačbe (3.23) podaja slika 3.2. k_1 je pri tem konstanta hitrosti produkcije proteina, q_1 pa konstanta hitrosti degradacije proteina. V stabilnem stanju $\frac{d[P]}{dt} = 0$ doseže koncentracija proteina P maksimalno vrednost $\frac{k_1}{q_1}$.

3.3 Hillove funkcije

Prejšnje formulacije izražanja genov so predpostavljale, da je promotor vedno aktiven. Ta predpostavka seveda ni vedno izpolnjena, saj promotor prehaja med aktivnim in neaktivnim stanjem glede na prisotnost transkripcijskih faktorjev, torej aktivatorjev in represorjev. Kot izhodišče za modeliranje vpliva transkripcijskih faktorjev na hitrost transkripcije lahko uporabimo *Hillove funkcije* [14]. Hillova funkcija za aktivator je



Slika 3.1 Primer grafa koncentracije mRNA ($[mRNA]$) in koncentracije proteina ($[P]$) v odvisnosti od časa v skladu z enačbami (3.13) in (3.14). Izražanje gena je modelirano z ločitvijo koraka transkripcije in translacije. Konstante hitrosti transkripcije, translacije, degradacije mRNA in degradacije proteina so $k_1 = 1$, $k_2 = 0.1$, $q_1 = 0.1$, $q_2 = 0.1$.



Slika 3.2 Primer grafa koncentracije proteina ($[P]$) v odvisnosti od časa v skladu z enačbo (3.23). Izražanje gena je modelirano tako, da sta transkripcija in translacija združeni v en korak produkcije proteina. Konstanti hitrosti produkcije in degradacije proteina sta $k_1 = 1$, $q_1 = 0.1$.

definirana kot

$$f([A]) = k_1 \cdot \frac{[A]^n}{K_A^n + [A]^n}. \quad (3.24)$$

Pri tem je $[A]$ koncentracija ob promotor vezanega aktivatorja kot spodbujevalca transkripcije. k_1 je konstanta hitrosti transkripcije (če modeliramo transkripcijo in translacijo kot ločena koraka) oz. produkcije (če modeliramo transkripcijo in translacijo kot skupen korak) in določa maksimalno možno vrednost Hillove funkcije. K_A je aktivacijski koeficient in določa potrebno koncentracijo vezanega aktivatorja, da pride - v primeru minimalnega promotorja - do 50% izražanja oz. do 50% aktivacije transkripcije - $f(K_A)$ ima vrednost, ki je polovica maksimalne. n je *Hillov koeficient*, ki podaja nelinearnost oz. strmino funkcije, zato ga imenujemo tudi koeficient nelinearnosti. Za $n = 1$ ima $f([A])$ hiperbolično obliko, z večanjem n pa dobiva sigmoidno. Graf Hillove funkcije za aktivator prikazuje slika 3.3.

Funkcijo $f([A])$ si lahko razlagamo kot verjetnost vezave aktivatorja in s tem verjetnost aktivnega minimalnega promotorja. Funkcija pri neki koncentraciji aktivatorja doseže maksimalno vrednost oz. nasičenje, saj verjetnost vezave aktivatorja ne more preseči vrednosti 1.

Hilova funkcija za represor je definirana kot

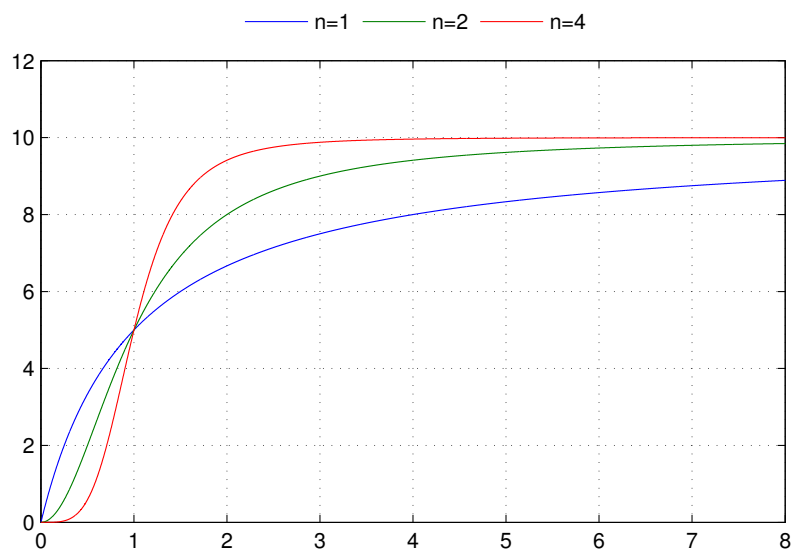
$$f([R]) = k_1 \cdot \frac{K_R^n}{K_R^n + [R]^n}. \quad (3.25)$$

Če govorimo o konstitutivnem promotorju, ki postane neaktiven v primeru vezave zadostne količine represorja, lahko na $f([R])$ gledamo kot na verjetnost vezave represorja oz. verjetnost neaktivnega promotorja. k_1 določa maksimalni nivo transkripcije oz. produkcije - podaja zgornjo mejo funkcije $f([R])$, ki je dosežena ob ničelni koncentraciji vezanega represorja $[R]$. K_R je represijski koeficient in je enak potrebni koncentraciji vezanega represorja za 50% represijo. n je Hillov koeficient oz. koeficient nelinearnosti funkcije. Graf Hillove funkcije za represor prikazuje slika 3.4.

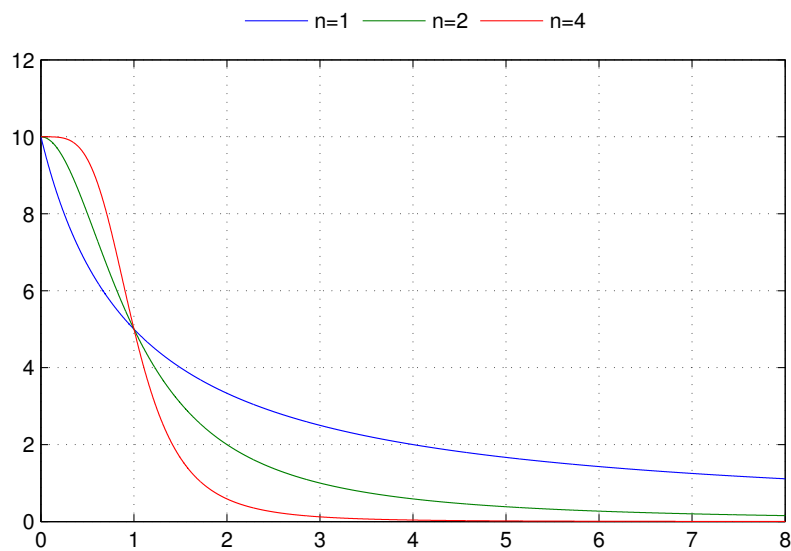
Izražanje gena pod minimalnim promotorjem, ob katerem je vezan kvečjemu aktivator, lahko upoštevajoč Hillovo funkcijo $f([A])$ zapišemo z diferencialno enačbo

$$\frac{d[P]}{dt} = k_1 \cdot \frac{[A]^n}{K_A^n + [A]^n} - q_1 \cdot [P]. \quad (3.26)$$

Izražanje gena pod konstitutivnim promotorjem, ob katerem je vezan kvečjemu represor,



Slika 3.3 Primeri grafov Hillove funkcije za aktivator $f([A])$ v odvisnosti od koncentracije vezanega aktivatorja $[A]$ pri različnih vrednostih koeficienta nelinearnosti n . Pri tem je $k_1 = 10$ in $K_A = 1$. Vse krivulje se z večanjem $[A]$ približujejo zgornji meji $k_1 = 10$.



Slika 3.4 Primeri grafov Hillove funkcije za represor $f([R])$ v odvisnosti od koncentracije vezanega represorja $[R]$ pri različnih vrednostih koeficienta nelinearnosti n . Pri tem je $k_1 = 10$ in $K_R = 1$. Vse krivulje se z večanjem $[R]$ približujejo vrednosti nič.

lahko upoštevajoč Hillovo funkcijo $f([R])$ zapišemo z diferencialno enačbo

$$\frac{d[P]}{dt} = k_1 \cdot \frac{K_R^n}{K_R^n + [R]^n} - q_1 \cdot [P]. \quad (3.27)$$

V obeh enačbah sta transkripcija in translacija združeni v en korak.

3.4 Deterministično modeliranje na osnovi verjetnostne interpretacije izražanja genov

Čeprav so Hillove funkcije lahko uporabne kot aproksimacija empiričnih podatkov, pri nekaterih GRO naletimo na problem, če se ob promotorju nahaja več vezavnih mest za različne transkripcijske faktorje, npr. tako za aktivatorje kot tudi za represorje obenem. V tem primeru potrebujemo bolj splošen pristop modeliranja. Najbolj direkten in splošen način obsega zapis kemijskih reakcij dinamike gensko regulatornega omrežja in nato določitev navadnih diferencialnih enačb na osnovi zakona o vplivu koncentracij. Tak pristop modeliranja je determinističen, tj. brez upoštevanja naključnosti v dinamiki izražanja genov.

Za formulacijo hitrejša in krajša deterministična alternativa temu pristopu sloni na verjetnostni interpretaciji izražanja genov, ki upošteva prehajanje med stanji promotorja. Taka abstrakcija je bolj groba kot opis dinamike z bolj natančnim naborom kemijskih enačb, kljub temu pa lahko nudi ustrezen opis gensko regulatornega omrežja. Če modeliramo transkripcijo in translacijo kot en korak, potem lahko izražanje genov v GRO opišemo z diferencialnimi enačbami oblike

$$\frac{d[P]}{dt} = k \cdot p(\text{aktiven promotor}) + k_b \cdot p(\text{neaktiven promotor}) - q \cdot [P]. \quad (3.28)$$

Pri tem je $[P]$ koncentracija proteina, k konstanta hitrosti produkcije proteina v primeru aktivnega promotorja, k_b konstanta hitrosti puščanja in q konstanta hitrosti degradacije proteina. Zaželjena vrednost k_b je navadno bistveno nižja od k - znaša npr. 10% vrednosti k ali manj. $p(\text{aktiven promotor})$ je verjetnost, da je promotor v aktivnem stanju. $p(\text{neaktiven promotor})$ je verjetnost, da je promotor v neaktivnem stanju. Pri tem velja:

$$p(\text{aktiven promotor}) = 1 - p(\text{neaktiven promotor}). \quad (3.29)$$

Oblika enačbe (3.28) v grobem temelji na determinističnem pristopu, opisanem v [15].

Verjetnost aktivnega promotorja lahko modeliramo s količino, imenovano *delež zasedenosti*¹ (angl. *fractional occupancy*) [12]. Ta podaja zasedenost promotorja oz. vezavnih mest za transkripcijske faktorje in je definirana kot razmerje med vsoto aktivnih in vsoto vseh možnih stanj nekega promotorja:

$$f = \frac{\sum \text{aktivna stanja promotorja}}{\sum \text{vsa stanja promotorja}}. \quad (3.30)$$

Stanja promotorja so v končni obliki enačbe izražena s pomočjo koncentracij transkripcijskih faktorjev, ki vplivajo nanj. Predpostavka tovrstnega modeliranja je binarno obravnavanje stanja promotorja - vezava nekega transkripcijskega faktorja lahko vodi le v aktivno stanje promotorja (kjer prihaja do izražanja genov) ali le v neaktivno stanje promotorja (kjer prihaja le do izražanja, določenega s konstanto hitrosti puščanja k_b). Če upoštevamo enačbo (3.28), lahko pišemo:

$$p(\text{aktiven promotor}) = f, \quad (3.31)$$

$$p(\text{neaktiven promotor}) = 1 - f. \quad (3.32)$$

Če vzamemo za primer gensko regulatorno omrežje z minimalnim promotorjem, enim vezavnim mestom za aktivator in enim genom (slika 3.5), so možna naslednja stanja promotorja:

- U - prosto vezavno mesto: neaktiven promotor;
- U_A - na vezavno mesto je vezan aktivator: aktiven promotor.

Prehajanje med stanjema promotorja opisuje reakcija



A je aktivator, ki se veže na nezasedeno vezavno mesto oz. odcepi od zasedenega vezavnega mesta. Če sta vezava in odcepitev aktivatorja z vezavnega mesta veliko hitrejši od produkcije proteina (transkripcije in translacije), lahko predpostavljamo, da sta hitrosti reakcij - vezava aktivatorja na prosto vezavno mesto in odcepitev od njega, ki rezultira spet v prvotnem stanju - enaki oz. v *ravnovesju*:

$$p \cdot U \cdot [A] = r \cdot U_A. \quad (3.34)$$

¹V času pisanja tega dela nismo našli ustreznega slovenskega prevoda pojma *fractional occupancy*, zato v besedilu uporabljamo prevod *delež zasedenosti*.

To drži v večini primerov. Enačbo lahko preoblikujemo v:

$$\frac{p}{r} = \frac{U_A}{U \cdot [A]} = K. \quad (3.35)$$

Pri tem je K asociacijska konstanta [12]. Delež zasedenosti oz. verjetnost aktivnega promotorja je tako enaka

$$f = \frac{U_A}{U + U_A} = \frac{K \cdot U \cdot [A]}{U + K \cdot U \cdot [A]} = \frac{K \cdot [A]}{1 + K \cdot [A]}. \quad (3.36)$$

To lahko posplošimo na končno obliko

$$f = \frac{K \cdot [A]^n}{K_A^n + K \cdot [A]^n}. \quad (3.37)$$

K_A podaja zahtevano koncentracijo aktivatorja A , pri kateri bo $f = 0,5$. Če je $K = 1$ torej predstavlja mejo za 50% aktivacijo promotorja. n je koeficient nelinearnosti. Z vstavitvijo te enačbe v splošno enačbo (3.28) dobimo končni zapis izražanja gena za dano omrežje:

$$\frac{d[P]}{dt} = k \cdot \frac{K \cdot [A]^n}{K_A^n + K \cdot [A]^n} + k_b \cdot \left(1 - \frac{K \cdot [A]^n}{K_A^n + K \cdot [A]^n}\right) - q \cdot [P]. \quad (3.38)$$

Opazimo lahko, da je enačba (3.37) pomnožena s konstanto hitrosti produkcije k , enaka Hillovi funkciji za aktivator, če je $K = 1$.

Na enak način lahko modeliramo gensko regulatorno omrežje, kjer se ob konstitutivnem promotorju z enim genom nahaja vezavno mesto za represor (slika 3.6). Možna stanja promotorja so v tem primeru:

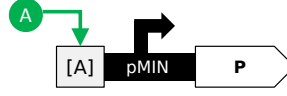
- U - prosto vezavno mesto: aktiven promotor;
- U_R - na vezavno mesto je vezan represor: neaktiven promotor.

Prehajanja med stanjema promotorja opisuje reakcija

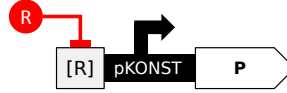


R je represor, ki se veže na nezasedeno vezavno mesto oz. odcepi od zasedenega vezavnega mesta. Če predpostavljamo ravnovesje zgornje reakcije kot posledico zelo hitre vezave in odcepitve napram produkciji proteina, lahko pišemo

$$p \cdot U \cdot [R] = r \cdot U_R. \quad (3.40)$$



Slika 3.5 Gensko regulatorno omrežje z enim genom, ki se izraža, ko je minimalni promotor aktiven. Minimalni promotor je aktiven, če je na vezavno mesto ob njem vezan aktivator, sicer je neaktiven.



Slika 3.6 Gensko regulatorno omrežje z enim genom, ki se izraža, ko je konstitutivni promotor aktiven. Konstitutivni promotor je aktiven, če je vezavno mesto ob njem prosto oz. nanj ni vezan represor.

Velja torej:

$$\frac{p}{r} = \frac{U_R}{U \cdot [R]} = K. \quad (3.41)$$

K je ponovno asociacijska konstanta. Delež zasedenosti je tako enak

$$f = \frac{U}{U + U_R} = \frac{U}{U + K \cdot U \cdot [R]} = \frac{1}{1 + K \cdot [R]}. \quad (3.42)$$

Če izraz posplošimo, dobimo

$$f = \frac{K_R^n}{K_R^n + K \cdot [R]^n}. \quad (3.43)$$

Končno diferencialno enačbo, ki podaja izražanje gena za dano omrežje, dobimo z vstavitvijo f v splošno enačbo (3.28), kjer je f verjetnost aktivnega promotorja. Tako omrežje predstavlja logično negacijo - ob odsotnosti represorja kot vhodnega signala je koncentracija proteina visoka, ob prisotnosti represorja pa nizka. Spet velja, da je enačba (3.43), pomnožena s konstanto hitrosti produkcije k , enaka Hillovi funkciji za represor, če je $K = 1$. Delež zasedenosti si zato lahko razlagamo kot posplošitev Hillovih funkcij na primere z več vezavnimi mesti oz. na primere, kjer na nek promotor lahko vpliva več različnih transkripcijskih faktorjev.

3.5 Kooperativnost

Pomembno vlogo v številnih GRO igra *kooperativnost* oz. *kooperativna vezava* transkripcijskih faktorjev [12]. Gre za vezavo, kjer vezava enega transkripcijskega faktorja vpliva na afiniteto oz. verjetnost vezave drugih transkripcijskih faktorjev. Pri pozitivni kooperativnosti vezava enega transkripcijskega faktorja poveča verjetnost vezave drugega

transkripcijskega faktorja, pri negativni kooperativnosti pa vezava enega transkripcijskega faktorja zmanjša verjetnost vezave drugega transkripcijskega faktorja. Pogost mehanizem kooperativnosti je vezava transkripcijskih faktorjev kot *oligomerov* - molekul, ki jih sestavlja več *monomerov*, tj. posameznih molekul. Tipični primeri oligomerov so npr. dimeri (sestavljani iz dveh enot oz. monomerov) in trimeri (sestavljani iz treh monomerov). Z drugimi besedami, nekateri transkripcijski faktorji postanejo aktivni šele kot oligomeri, po *oligomerizaciji* (povezavi oz. združitvi) posameznih monomerov, ki nastajajo pri translaciji.

Kooperativna vezava se na grafu poteka koncentracije proteinov v odvisnosti od časa kaže kot sigmoidna krivulja namesto običajne hiperbolične. Pojav sigmoidne krivulje označujemo s pojmom *funkcijska kooperativnost* - pomembno je namreč poudariti, da sigmoidna krivulja kot indikator nelinearnosti v GRO še ne pomeni avtomatsko molekularne kooperativnosti, tj. kooperativnosti transkripcijskih faktorjev v kemijskem smislu [16]. Eden od razlogov je, da modeli GRO temeljijo na številnih poenostavitvah, saj mnogi procesi, ki so del npr. transkripcije in translacije, niso eksplicitno modelirani.

V do sedaj opisanih determinističnih modelih smo uporabljali eksponent koncentracij transkripcijskih faktorjev, imenovan koeficient nelinearnosti n . Ta koeficient - v primeru Hillovih funkcij imenovan Hillov koeficient - je merilo funkcijske kooperativnosti in mu zato v kontekstu modelov pravimo tudi funkcijska kooperativnost. Slednja sicer lahko odraža tudi molekularno kooperativnost. Koeficient nelinearnosti $n = 1$ pomeni odsotnost kooperativnosti. Vrednosti n nad 1 pomenijo pozitivno kooperativnost, vrednosti pod 1 pa negativno kooperativnost.

3.6 Primerjava determinističnih in stohastičnih modelov

Do sedaj obravnavani pristopi modeliranja GRO so bili deterministični in niso upoštevali stohastičnosti, ki je prisotna pri izražanju in regulaciji genov. Deterministični modeli lahko ustrezno opišejo povprečno obnašanje populacije celic ali časovno povprečje obnašanja posamezne celice [17]. Ne morejo pa ustrezno zajeti stanja posamezne celice, ki se lahko precej razlikuje od povprečnega stanja populacije, čeprav so celice genetsko identične in se nahajajo v enakem okolju. Čeprav vse celice v populaciji vsebujejo enake genske konstrukte, se lahko zgodi, da odražajo različno dinamiko; v nekaterih se npr. nek protein opazno izraža, v drugih pa ne. Vzrok za to je stohastičnost izražanja genov,

ki je posledica šuma pri procesu transkripcije in translacije. Za ustrezno modeliranje na nivoju posameznih celic zato potrebujemo stohastične modele, ki upoštevajo omenjeno stohastičnost biokemijskih procesov in so posledično natančnejši.

Obstajajo pogoji, ko se napovedi determinističnih in stohastičnih modelov ujemajo. Prvi pogoj za to so visoke koncentracije mRNA in proteinov ter veliki volumni celic; drugi pogoj so hitri prehodi med stanji promotorja [15].

Slabost stohastičnih modelov je večja računska zahtevnost. Poleg tega za razliko od determinističnih modelov, opisanih z navadnimi diferencialnimi enačbami, ne omogočajo neposredne uporabe analitičnih tehnik [18].

3.7 Stohastično modeliranje

Pri stohastičnem modeliranju izhajamo iz opisa sistema oz. gensko regulatornega omrežja v obliki kemijskih reakcij. *Stanje sistema* v nekem trenutku N je vektor, ki vsebuje število molekul posameznih v reakcijah nastopajočih kemijskih zvrsti, npr. nekega proteina ali transkripcijskega faktorja. Vsaka reakcija ima še dva atributa, *funkcijo propenzitete* (angl. *propensity function*) in *stehiometrični vektor* [11]. Funkcija propenzitete a opisuje hitrost reakcije. Stehiometrični vektor s pove, koliko enot oz. molekul posamezne kemijske zvrsti je bilo v neki reakciji proizvedenih (produkti) ali porabljenih (reaktanti). Ko se zgodi neka reakcija, se stanje sistema posodobi glede na stehiometrični vektor te reakcije. Za primer obravnavajmo preprosto reakcijo, ki naj predstavlja cel sistem:



Stanje sistema je določeno z vektorjem $N = [N_A, N_B]$, kjer je N_A število molekul A, N_B pa število molekul B. Funkcija propenzitete reakcije je $a = k_1 \cdot N_A$. Ker se v reakciji porabi (odšteje) ena enota A in proizvede (prišteje) ena enota B, je stehiometrični vektor reakcije $s = [-1, 1]$. Če je začetno stanje sistema npr. $N = [1, 2]$, bo po izvedbi dane reakcije novo stanje sistema $N' = N + s = [0, 3]$.

Naj bo $P(N, t)$ verjetnost, da je sistem ob času t v stanju N . *Kemijska glavna enačba* (KGE) (angl. *chemical master equation* - CME) podaja časovni potek za $P(N, t)$ in je definirana kot [11]:

$$\frac{dP(N, t)}{dt} = \sum_i \left(P(N - s_i, t) \cdot a_i(N - s_i) - P(N, t) \cdot a_i(N) \right). \quad (3.45)$$

KGE obsega diferencialno enačbo za vsako stanje, v katerem se lahko sistem znajde, kar pogosto pomeni neskončno število enačb. Zato tipično za reševanje KGE kot osnovo stohastične simulacije uporabljamo numerične postopke, npr. *stohastični simulacijski algoritem* (angl. *stochastic simulation algorithm* - SSA), imenovan tudi *Gillespiejev algoritem*. Algoritem se poslužuje uporabe dveh slučajnih spremenljivk:

- naslednje reakcije za izvedbo R ,
- časa do naslednje reakcije T .

Algoritem se premika po času od izvedbe ene reakcije do naslednje. Pri tem je verjetnost izbire naslednje reakcije proporcionalna njeni propenziteti. Verjetnostna porazdelitev T je določena po formuli [11]:

$$P(0 \leq T \leq t) = 1 - e^{-a \cdot t}. \quad (3.46)$$

Pri tem je

$$a = \sum_i a_i. \quad (3.47)$$

Algoritem se izvaja do nekega maksimalnega časa t_{max} in obsega izvajanje naslednjih korakov [11]:

1. določi začetno stanje sistema (vektor N) in nastavi čas t na vrednost 0;
2. za dano stanje izračunaj propenziteto reakcij $a_i(N)$;
3. določi vrednost T (čas do naslednje reakcije) in R (naslednja reakcija, ki se bo zgodila);
4. povečaj čas simulacije na $t = t + T$;
5. posodobi vektor stanj sistema - upoštevaj, da se je zgodila reakcija R , ki ji pripada stehiometrični vektor s : $N = N + s$;
6. če maksimalni čas izvajanja algoritma še ni bil dosežen, ponovi korak 2.

Posebna verzija algoritma SSA je *zakasneni SSA*, ki podpira uporabo zakasnitev v reakcijah. Implementacija temelji na uporabi čakalne vrste za zakasnjene dogodke. Tako omogoča modeliranje npr. zakasnitve med transkripcijo in translacijo. Zakasneni SSA je

uporabljen v brezplačnem modelirnem orodju *SGN Sim* [19]. Primer zakasnjene reakcije je npr.



Produkt B v tem primeru nastane po zakasnitvi 15 časovnih enot.

3.8 Hibridni diskretni algoritem

V tem podpoglavju je podan opis *hibridnega diskretnega algoritma (HDA)* kot nove alternative za modeliranje GRO. Prva verzija algoritma je bila razvita za potrebe tekmovanja iGEM 2012 [3]. Pojem *hibridni* v tem primeru pomeni, da je v sicer deterministični pristop vpeljana tudi stohastičnost izražanja genov v obliki nihanja količine proizvedenih molekul mRNA in proteinov. *Diskretni* pomeni, da algoritem operira s celoštevilskimi koncentracijami molekul mRNA in proteinov - pri tem je vsaka molekula svoj objekt, ki mu algoritem sledi tekom simulacije. Glavni pomen algoritma je v njegovih naslednjih lastnostih:

1. omogoča eksplicitno modeliranje končnega števila ponovitev nekega vezavnega mesta;
2. omogoča eksplicitno modeliranje kompetitivne vezave transkripcijskih faktorjev;
3. omogoča modularno snovanje GRO.

V praksi ima vezavno mesto pogosto večje število ponovitev (npr. 10), kar poveča robustnost oz. verjetnost vezave transkripcijskega faktorja. Deterministično modeliranje z deležem zasedenosti sicer omogoča, da zasedenost vsakega vezavnega mesta obravnavamo kot svoje stanje, ki pomeni ali aktivni ali neaktivni promotor, vendar tak pristop hitro vodi v ogromno število kombinacij možnih stanj. Podobno velja za stohastični pristop s kemijskimi reakcijami. V najboljšem primeru lahko iz vidika uporabnosti modela oz. verodostojnosti napovedi razliko med različnim številom ponovitev vezavnih mest zanemarimo. Po drugi strani nam lahko tovrstno eksplicitno modeliranje ponudi razlago, ki je z drugimi modeli ne bi pridobili. Smiselna je uporaba različnih modelov GRO in primerjava rezultatov z eksperimentalnimi ugotovitvami.

V nekaterih GRO pomembno vlogo igra kompetitivna vezava transkripcijskih faktorjev. Gre za vezavo, kjer dva transkripcijska faktorja - npr. aktivator in represor -

tekmujeta za vezavo na isto vezavno mesto, če sta prisotna istočasno. V splošnem lahko tekmuje poljubno število transkripcijskih faktorjev. HDA omogoča eksplicitno modeliranje kompeticije dveh transkripcijskih faktorjev. Molekule transkripcijskih faktorjev se po vnaprej določenem pravilu razporedijo med razpoložljive (nezasedene) ponovitve vezavnih mest. Pri tem lahko ena molekula oz. entiteta transkripcijskega faktorja zasede eno ponovitev.

HDA je formuliran in implementiran objektno orientirano. GRO tako lahko načrtujemo objektno oz. modularno: vsak objekt predstavlja določen gradnik GRO (npr. promotor, gen ali vezavno mesto), z atributi objekta pa določamo parametre (npr. hitrost degradacije proteina) in relacije (npr. pripadnost vezavnega mesta nekemu promotorju). Tak pristop omogoča hitro in splošno načrtovanje GRO, brez da bi pri tem morali za vsako specifično omrežje posebej formulirati enačbe ali kemijske reakcije za opis dinamike. Slednja je določena z univerzalnimi pravili, ki so implicitna vsem modelom. Pravila vseeno lahko uravnavamo preko parametrov - s tem se doseže možnost prilagoditve modela na npr. eksperimentalne rezultate oz. specifično omrežje ter tako ponudi ustrezna fleksibilnost. HDA ima lastnosti *agentnega modeliranja*, tj. modeliranja, kjer neka pravila interakcije med posameznimi osnovnimi gradniki sistema, t.i. agenti, vodijo v pojav končne dinamike kot posledice teh pravil. Zaradi modularnosti je HDA tudi enostavno razširljiv z novimi gradniki in različnimi tipi dinamike.

Kljub stohastičnim elementom je HDA v osnovi determinističen algoritem, zato predpostavlja visoke molekularne koncentracije (relativno glede na število ponovitev vezavnih mest) in hitro vezavo oz. odcepitev transkripcijskih faktorjev napram hitrosti transkripcije ter translacije. Na vezavna mesta vezani transkripcijski faktorji se obravnavajo (degradirajo itd.) enako kot prosti transkripcijski faktorji.

Algoritem je bil implementiran v jeziku C#.

3.8.1 Objektni model

HDA implementira naslednje razrede, ki služijo kot osnova za postavitve GRO:

- razred *mRNA*, ki predstavlja posamezno molekulo oz. entiteto mRNA - in razred *Protein*, ki predstavlja posamezno proteinsko molekulo oz. entiteto;
- razred *Product*, ki predstavlja genski produkt v obliki seznama mRNA molekul (tipa *mRNA*) nekega gena in seznama pripadajočih proteinov (tipa *Protein*);

- razred *BindingSite*, ki predstavlja vezavno mesto za transkripcijski faktor;
- razred *Promoter*, ki predstavlja promotor;
- razred *Gene*, ki predstavlja gen kot osnovo za nastajanje genskega produkta.

Razreda *mRNA* in *Protein* vsebujeta naslednje attribute (v oklepaju je podan podatkovni tip atributa):

- (*int*) *BirthTime* - čas rojstva, tj. čas oz. časovni korak generiranja objekta (molekule) v simulaciji;
- (*bool*) *Alive* - parameter, ki določa, ali je molekula v omrežju aktivna (v primeru vrednosti *true*) ali pa je označena za degradacijo oz. odstranitev iz seznama molekul (v primeru vrednosti *false*).

Razred *Product* vsebuje naslednje javne attribute:

- (*List<mRNA>*) *mRNAs* - seznam molekul mRNA, ki določajo ta genski produkt; vsaka molekula v seznamu je tipa *mRNA*;
- (*List<Protein>*) *Proteins* - seznam proteinov, ki določajo ta genski produkt; vsaka molekula proteina v seznamu je tipa *Protein*;
- (*double*) *mRNADegradationPercent* - odstotek molekul mRNA, vezanih na ta objekt, ki se degradirajo v vsakem časovnem koraku simulacije;
- (*double*) *ProteinDegradationPercent* - odstotek proteinskih molekul, vezanih na ta objekt, ki se degradirajo v vsakem časovnem koraku simulacije;
- (*int*) *TranslationDelay* - časovni zamik med transkripcijo in translacijo v številu časovnih korakov simulacije (privzeto 0); vsaka molekula iz seznama *mRNAs* je podvržena translaciji po *TranslationDelay* časovnih korakih od njenega rojstnega časa *BirthTime*;
- (*int*) *TranslationRate* - število molekul proteina, ki nastanejo na posamezno molekulo mRNA pri translaciji; privzeta vrednost je 1.

Razred *Product* vsebuje naslednje bistvene javne metode:

- *(void) Degrade()* - izvede degradacijo mRNA in proteinov - iz seznama *mRNAs* se pobriše določeno število objektov mRNA, iz seznama *Proteins* pa določeno število proteinov;
- *(void) DegradeMRNA()* - izvede degradacijo mRNA, tj. odstranitev določenega števila objektov mRNA iz seznama *mRNAs*;
- *(void) DegradeProtein()* - izvede degradacijo proteinov, tj. odstranitev določenega števila proteinskih objektov iz seznama *Proteins*;
- *(void) Translate()* - izvede korak translacije: za vsako aktivno mRNA molekulo iz seznama *mRNAs* zgenerira - če je že pretekla zakasnitev med transkripcijo in translacijo - določeno število proteinov in jih doda v seznam *Proteins*.

Razred *BindingSite* vsebuje naslednje javne attribute:

- *(int) Capacity* - kapaciteta vezavnega mesta, tj. število ponovitev tega vezavnega mesta (na vsako ponovitev vezavnega mesta se lahko veže en objekt transkripcijskega faktorja); ta atribut sicer lahko interpretiramo manj dobesedno kot vpliv vezave enega transkripcijskega faktorja - vpliv atributa je odvisen od vrednosti ostalih atributov oz. parametrov (npr. v nadaljevanju opisanih konstant K_A in K_R), ki določajo končna pravila oz. enačbe transkripcije;
- *(int) AmountOfBoundActivator* - število aktivatorskih molekul oz. objektov vezanih na ponovitve tega vezavnega mesta;
- *(int) AmountOfBoundRepressor* - število represorskih molekul oz. objektov vezanih na ponovitve tega vezavnega mesta.

Razred *BindingSite* vsebuje naslednje javne metode:

- *(void) BindActivators(Activators)* - kot parameter sprejme seznam aktivatorjev (vsak aktivator je tipa *Protein*), ki so kandidati za vezavo in jih ustrezno veže na razpoložljive ponovitve vezavnega mesta;
- *(void) BindRepressors(Repressors)* - kot parameter sprejme seznam represorjev (vsak represor je tipa *Protein*), ki so kandidati za vezavo in jih ustrezno veže na razpoložljive ponovitve vezavnega mesta;

- *(void) BindTFs(Activators, Repressors)* - kot parametre sprejme seznam aktivatorjev in represorjev, ki tekmujejo za vezavo in jih po določenih pravilih veže na razpoložljive ponovitve vezavnega mesta.

Razred *Promoter* vsebuje naslednje bistvene javne attribute (metod ne vsebuje):

- *(PromoterType) Type* - tip promotorja, ki je lahko minimalni (*Minimal*) ali konstitutivni (*Constitutive*);
- *(bool) Enabled* - omogočenost promotorja: v primeru vrednosti *true* (privzeto) je promotor omogočen (lahko teče transkripcija genov pod njim), sicer je onemogočen;
- *(List<BindingSite>) BindingSites* - seznam vezavnih mest, ki pripadajo promotorju (vsako vezavno mesto je tipa *BindingSite*);
- *(int) ActivatorCount* - skupno število na vezavna mesta promotorja vezanih aktivatorskih molekul;
- *(int) RepressorCount* - skupno število na vezavna mesta promotorja vezanih represorskih molekul;
- *(int) KA, (int) KR, (int) ActiveTranscriptionRate, (int) ActivatedConstitutiveTranscriptionRate, (int) LeakyTranscriptionRate* - parametri, ki vplivajo na hitrost transkripcije genov pod promotorjem na način, opisan v nadaljevanju poglavja (hitrost transkripcije se meri v številu generiranih molekul mRNA na časovni korak simulacije);
- *(double) ActivatorExp* - koeficient nelinearnosti oz. funkcijske kooperativnosti za vezane aktivatorje;
- *(double) RepressorExp* - koeficient nelinearnosti oz. funkcijske kooperativnosti za vezane represorje.

Razred *Gene* vsebuje naslednja bistvena javna atributa:

- *(Promoter) GenePromoter* - promotor gena;
- *(int) TranscriptionScaleConstant* - konstanta, s katero je pomnoženo število pri transkripciji generiranih molekul mRNA, ki jih določa gen;

Razred *Gene* vsebuje naslednjo javno metodo:

- (*void*) *Transcribe(mRNAs)* - opravi transkripcijo gena, tj. generira določeno število molekul mRNA in jih doda v seznam *mRNAs*, ki ga sprejme kot parameter.

3.8.2 Delovanje algoritma

Modeliranje s HDA obsega dva glavna koraka:

1. definiramo strukturo gensko regulatornega omrežja kot nabora objektov (promotorjev, genov, vezavnih mest, genskih produktov itd.) - pri tem določimo ustrezne parametre (hitrosti degradacije, transkripcije itd.);
2. izvajamo simulacijo kot končno zaporedje časovnih korakov, pri čemer v vsakem koraku:
 - (a) določimo vpliv morebitnih vhodnih signalov omrežja (npr. prisotnost neke molekule prepreči vezavo transkripcijskega faktorja),
 - (b) kličemo metode za vezavo transkripcijskih faktorjev,
 - (c) kličemo metode za transkripcijo genov in translacijo,
 - (d) kličemo metode za degradacijo mRNA ter proteinov.

Vsako vezavno mesto je modelirano kot svoj objekt tipa *BindingSite*. V vsakem koraku simulacije algoritem porazdeli transkripcijske faktorje na ciljna vezavna mesta. Naj bo B vezavno mesto, določeno s

- kapaciteto C (številom ponovitev vezavnega mesta - atribut *Capacity*),
- številom vezanih aktivatorskih molekul B_A (atribut *AmountOfBoundActivator*),
- številom vezanih represorskih molekul B_R (atribut *AmountOfBoundRepressor*).

Skupno število vezanih aktivatorskih in represorskih molekul ne more presegati kapacitete vezavnega mesta:

$$B_A + B_R \leq C. \quad (3.49)$$

Naj bo A število za vezavo razpoložljivih aktivatorskih molekul in R število za vezavo razpoložljivih represorskih molekul. V primeru kompetitivne vezave aktivatorja in represorja za isto vezavno mesto algoritem enakomerno porazdeli aktivatorske in represorske molekule proporcionalno z njuno koncentracijo po pravilu:

■ če je $A + R < C$, potem: $B_A = A$, $B_R = R$;

■ če je $A + R \geq C$, potem:

$$B_A = C \cdot \frac{A}{A + R}, \quad (3.50)$$

$$B_R = C \cdot \frac{R}{A + R}. \quad (3.51)$$

V primeru nekompetitivne vezave velja $R = 0$ (če se veže samo aktivator) oz. $A = 0$ (če se veže samo represor).

Transkripcija gena pod določenim promotorjem obsega v vsakem koraku simulacije naslednje korake:

1. seštej vezane aktivatorje in represorje vseh vezavnih mest promotorja:

$$A_N = \sum_i B_{A_i}, \quad (3.52)$$

$$R_N = \sum_i B_{R_i}. \quad (3.53)$$

Pri tem je B_{A_i} število vezanih aktivatorskih molekul na vezavnem mestu i , B_{R_i} pa število vezanih represorskih molekul na vezavnem mestu i . Vrednost A_N (število vezanih aktivatorskih molekul) se shrani v promotorjev atribut *ActivatorCount*, vrednost R_N (število vezanih represorskih molekul) pa v atribut *RepressorCount*.

2. Če gre za minimalni promotor, izračunaj celo število molekul mRNA, N'_{mRNA} , ki naj se generirajo v tem koraku simulacije:

$$N'_{mRNA} = b - b \cdot \frac{R_N^n}{K_R^n + R_N^n} + k \cdot \frac{A_N^m}{K_A^m + A_N^m + R_N^n}. \quad (3.54)$$

Pomen parametrov je naslednji:

- b je hitrost transkripcije ob odsotnosti transkripcijskih faktorjev (podaja jo promotorjev atribut *LeakyTranscriptionRate*);
- k je hitrost aktivirane transkripcije (podaja jo promotorjev atribut *ActiveTranscriptionRate*);
- K_R in K_A podajata - kot v ostalih determinističnih modelih - potrebno količino vezanega aktivatorja oz. represorja za določeno stopnjo aktivacije oz. represije (določena sta s promotorjevima atributoma KR in KA);

- n in m sta koeficienta nelinearnosti (določena sta s promotorjevima atributoma *RepressorExp* in *ActivatorExp*).

3. Če gre za konstitutivni promotor, izračunaj celo število molekul mRNA, N'_{mRNA} , ki naj se generirajo v tem koraku simulacije:

$$N'_{mRNA} = b + k - k \cdot \frac{R_N^n}{K_R^n + R_N^n} + \alpha \cdot \frac{A_N^m}{K_A^m + A_N^m + R_N^n}. \quad (3.55)$$

Pomen parametrov je naslednji (določajo jih enaki atributi kot za minimalni promotor):

- b je minimalna hitrost transkripcije (puščanje), ki poteka tudi ob popolni represiji konstitutivnega promotorja;
- k je hitrost nerepresirane transkripcije (ko ni vezan noben transkripcijski faktor);
- α je hitrost aktivirane transkripcije (določa jo promotorjev atribut *ActivatedConstitutiveTranscriptionRate*);

Pomen ostalih parametrov je enak kot za minimalni promotor.

4. N'_{mRNA} pomnoži s konstatno, ki jo določa atribut *TranscriptionScaleConstant* danega gena.
5. Vpelji stohastičnost transkripcije - povečaj ali zmanjšaj N'_{mRNA} za $\delta\%$, če je $N'_{mRNA} > 0$. δ zavzame naključno vrednost po enakomerni porazdelitvi iz intervala $[-\lambda, \lambda]$. Parameter λ imenujemo *stohastičnost transkripcije* - podana je v odstotkih, ki določajo maksimalni razpon naključnosti v hitrosti transkripcije. Končno število molekul mRNA za generacijo v tem koraku je tako celo število, določeno z izrazom

$$N_{mRNA} = N'_{mRNA} + \delta \cdot N'_{mRNA}. \quad (3.56)$$

Minimalna možna vrednost N_{mRNA} je 0.

6. Generiraj N_{mRNA} novih molekul mRNA.
7. Degradiraj $q_M\%$ mRNA, ki pripada genu. Vrednost q_M podaja atribut *mRNAdegradationPercent* iz razreda *Product*.

Enačba (3.54) temelji na empirični predpostavki, da transkripcija genov pod minimalnim promotorjem ob odsotnosti vseh transkripcijskih faktorjev poteka z relativno nizko hitrostjo b (prvi člen enačbe), vezava represorja pa lahko to hitrost še zmanjša - idealno na vrednost nič (drugi člen enačbe). Idealizirano velja $b = 0$ in za doseg izražanja gena je potrebna prisotnost aktivatorja. Tretji člen enačbe opisuje transkripcijo v primeru prisotnosti aktivatorja, ko se lahko doseže hitrost transkripcije k . Maksimalna hitrost transkripcije je tako $b + k$. Tipično za robusten sistem želimo, da je vrednost k precej večja (vsaj npr. desetkrat) od b , sicer učinek puščanja lahko prepreči doseg zelene dinamike. Tretji člen upošteva tudi možnost sočasne prisotnosti tako vezanega aktivatorja kot represorja (v primeru npr. kompetitivne vezave) - represor zmanjša učinek aktivatorja. S prilagajanjem parametrov lahko dosežemo, da ima vezava represorja ustrezno večji učinek od vezave enake količine aktivatorja - z drugimi besedami, že relativno majhen delež vezanega represorja glede na aktivator lahko opazno zniža hitrost transkripcije.

Tudi enačba (3.55) predpostavlja, da je b znatno manjši kot k . Ob odsotnosti transkripcijskih faktorjev tako poteka hitrost transkripcije genov pod konstitutivnim promotorjem s hitrostjo $b + k$. Zadostna količina vezanega represorja lahko hitrost zmanjša na b (tretji člen enačbe), vezava aktivatorja pa hitrost lahko še poveča (četrti člen enačbe).

Translacija poteka na naslednji način: za vsako aktivno molekulo mRNA se v vsakem časovnem koraku simulacije z verjetnostjo P_T generira T molekul proteina, če je že pretekel časovni zamik med transkripcijo in translacijo. Parametru P_T pravimo *učinkovitost translacije*. Parameter T je določen z atributom *TranslationRate* razreda *Product*. V vsakem časovnem koraku simulacije se tudi degradira $q_P\%$ proteina, ki ga kodira nek gen. Vrednost q_P podaja atribut *ProteinDegradationPercent* iz razreda *Product*.

Zgledi modeliranja s HDA so podani v četrtem poglavju.

3.9 Določanje vrednosti parametrov

Natančnost modeliranja je strogo odvisna od natančnosti uporabljenih parametrov, ki podajajo opis opazovanega sistema. Želimo, da bi model ustrezno aproksimiral dinamiko sistema tudi pod simulacijskimi pogoji, ki jih med njegovo vzpostavitev nismo eksperimentalno preverili. Določanje primernih vrednosti parametrov predstavlja velik izziv - natančna obravnava tega problema presega obseg naloge, zato so v tem podpoglavju opisane zgolj nekatere smernice.

V nekaterih primerih lahko uporabimo testne vrednosti parametrov in nato s simulacijo ugotavljamo, kako se sistem obnaša za različne nabore parametrov. Modeli lahko kljub velikim razponom v vrednostih parametrov zadovoljivo opišejo dinamiko nekaterih GRO. Pri izbiri testnih vrednosti parametrov upoštevamo določene predpostavke, za katere pričakujemo, da bodo izpolnjene - npr., da bo hitrost produkcije proteina višja od hitrosti degradacije. Neznane parametre lahko poskusimo oceniti tudi na podlagi ustreznih znanih parametrov drugih primernih modelov - če npr. ne poznamo hitrosti transkripcije nekega gena, lahko sklepamo, da bo ta podobna hitrosti transkripcije drugega podobnega gena v enakem okolju oz. organizmu. Pri tem si pomagamo z razpoložljivo literaturo in bazami podatkov.

Dejstvo je, da lahko eksperimentalno ovrednotimo le določene parametre - posamezen parameter lahko namreč predstavlja redukcijo celičnih dinamik v eno samo vrednost [20]. Ena možnost za določanje parametrov modela je s prileganjem simulacijskih rezultatov (npr. časovnega poteka koncentracije proteina) z eksperimentalnimi podatki. Pri tem je cilj minimizacija napake (npr. srednje kvadratne napake), ki podaja razhajanje med eksperimentalnimi podatki in podatki simulacije [20, 21]. Optimalno prilegajoče vrednosti parametrov lahko tako iščemo z različnimi optimizacijskimi metodami, npr. genetskim algoritmom.

4 Zgledi modeliranja s hibridnim diskretnim algoritmom

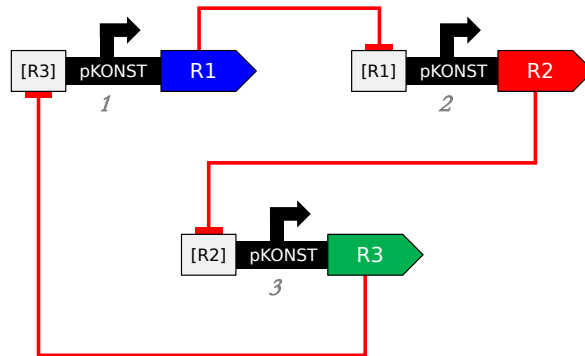
V tem poglavju so podani zgledi modeliranja realnih, eksperimentalno realiziranih gensko regulatornih omrežij. Glavni poudarek je na postavitvi modelov na načine, opisane v 3. poglavju - deterministično z uporabo deleža zasedenosti, stohastično in z uporabo hibridnega diskretnega algoritma (primeri kode so zapisani v jeziku C#). Deterministični modeli temeljijo na diferencialnih enačbah (3.28). Stohastične modele realiziramo kot množico reakcij z orodjem *SGN Sim* [19]. Modeli uporabljajo testne parametre, ki pa so dovolj dobri za ugotavljanje glavnih značilnosti sistemov. Podrobna analiza omrežij - npr. natančno preiskovanje vpliva posameznih parametrov na delovanje - je izpuščena. Namen opisanih modelov je predvsem predvideti, ali obstajajo smiselni pogoji za dosego želene (pričakovane) dinamike omrežja in kakšni v grobem so ti pogoji. V splošnem modeli niso sposobni predvideti npr. natančnega gibanja koncentracije nekega proteina v eksperimentalnem okolju že iz enostavnega razloga, da je znanstveno razumevanje številnih bioloških dinamik nepopolno in jih posledično ni moč povsem natančno zajeti z modelom.

4.1 Represilator

Represilator je gensko regulatorno omrežje, ki lahko izkazuje oscilacije koncentracij proteinov [18, 22]. Neodvisno od biološke realizacije ga lahko obravnavamo kot tri elementarne konstrukte. Vsak konstrukt iz logičnega vidika obsega:

- konstitutivni promotor,
- vezavno mesto za represor,
- gen, ki kodira represor.

Represor, ki se izraža iz gena prvega konstrukta, represira transkripcijo gena na drugem konstruktu. Represor, določen z genom drugega konstrukta, represira transkripcijo gena na tretjem konstruktu. Represor, določen z genom tretjega konstrukta, pa represira transkripcijo gena prvega konstrukta. Pod ustreznimi pogoji koncentracije represorjev oscilirajo. Strukturo represilatorja podaja slika 4.1.



Slika 4.1 Posplošena struktura represilatorja. Represorji treh konstruktov (označenih z 1, 2 in 3) ciklično represirajo gene, kar pod ustreznimi pogoji vodi v oscilacije koncentracij represorjev $R1$, $R2$, $R3$, ki jih kodirajo istoimenski geni.

4.1.1 Deterministični model

Pri determinističnem modelu se najprej osredotočimo na modeliranje prvega konstrukta. Promotor se lahko nahaja v naslednjih stanjih:

- U - prosto vezavno mesto (represor $R3$ ni vezan): aktiven promotor;
- U_R - na vezavno mesto je vezan represor $R3$: neaktiven promotor.

Prehajanje med stanjema podaja reakcija



R_3 je represor $R3$. Ker predpostavljamo ravnovesje, velja

$$U_R = K_1 \cdot U \cdot [R_3]. \quad (4.2)$$

Delež zasedenosti za prvi konstrukt je podan z izrazom

$$f_1 = \frac{U}{U + U_R} = \frac{1}{1 + K_1 \cdot [R_3]}. \quad (4.3)$$

To lahko posplošimo na

$$f_1 = \frac{K_a^{n_1}}{K_a^{n_1} + K_1 \cdot [R_3]^{n_1}}. \quad (4.4)$$

Po enakem načinu dobimo delež zasedenosti za drugi (f_2) in tretji (f_3) konstrukt, kjer je razlika le v nastopajočem represorju:

$$f_2 = \frac{K_b^{n_2}}{K_b^{n_2} + K_2 \cdot [R_1]^{n_2}}, \quad (4.5)$$

$$f_3 = \frac{K_c^{n_3}}{K_c^{n_3} + K_3 \cdot [R_2]^{n_3}}. \quad (4.6)$$

$[R_1]$ je koncentracija represorja $R1$, $[R_2]$ pa koncentracija represorja $R2$. Če zanemarimo puščanje in združimo transkripcijo ter translacijo v en korak, so diferencialne enačbe za represorje sledeče:

$$\frac{d[R_1]}{dt} = k_1 \cdot f_1 - q_1 \cdot [R_1], \quad (4.7)$$

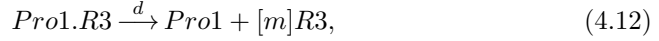
$$\frac{d[R_2]}{dt} = k_2 \cdot f_2 - q_2 \cdot [R_2], \quad (4.8)$$

$$\frac{d[R_3]}{dt} = k_3 \cdot f_3 - q_3 \cdot [R_3]. \quad (4.9)$$

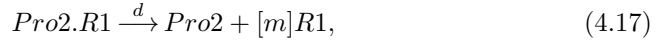
k_1 , k_2 in k_3 so konstante hitrosti produkcije represorjev, q_1 , q_2 in q_3 pa konstante hitrosti degradacije. Slika 4.2 prikazuje rezultat simulacije za podani model z začetnimi koncentracijami represorjev $[R_1] = 5$, $[R_2] = 10$, $[R_3] = 8$.

4.1.2 Stohastični model

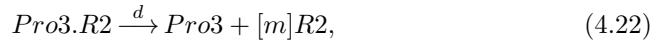
Tudi v stohastičnem modelu transkripcijo in translacijo modeliramo združeno ter zanemarimo puščanje. Reakcije, ki modelirajo prvi konstrukt, so:



Pro1 je promotor prvega konstrukta. Reakcija (4.10) podaja produkcijo represorja *R1*. Reakcija (4.11) podaja multimerično vezavo represorja *R3* na promotor prvega konstrukta (npr. za $m = 2$ se *R3* veže kot dimer oz. formalno gledano kot dve molekuli hkrati), kar spremeni stanje promotorja v neaktivno stanje *Pro1.R3*. Reakcija (4.12) je obratna reakcija reakcije (4.11) in podaja multimerično odcepitev *R3* od promotorja. Reakcija (4.13) podaja degradacijo vezanega represorja, ki rezultira v ponovno prostih enotah promotorja. Reakcija (4.14) opisuje degradacijo prostega represorja *R1*. Po enakem principu lahko postavimo reakcije za drugi konstrukt, kjer represijo izvaja represor *R1*, promotor *Pro2* pa vpliva na produkcijo represorja *R2*:



Enaka logika velja za tretji konstrukt, kjer represor *R2* represira gen *R3* pod promotorjem *Pro3*:



Rezultat simulacije stohastičnega modela prikazuje slika 4.3; začetne koncentracije represorjev so $R1 = 50$, $R2 = 100$, $R3 = 80$.

4.1.3 Realizacija modela s hibridnim diskretnim algoritmom

Pri modeliranju s hibridnim diskretnim algoritmom (HDA) najprej definiramo objekte za genske produkte, ki vsebujejo sezname molekul mRNA za vse tri represorje ter sezname vseh treh represorskih proteinov:

```
// mRNA in proteini za represor R1.
Product repressor1 = new Product();
repressor1.mRNADegradationPercent = 0.45;
repressor1.ProteinDegradationPercent = 0.1;

// mRNA in proteini za represor R2.
Product repressor2 = new Product();
repressor2.mRNADegradationPercent = 0.45;
repressor2.ProteinDegradationPercent = 0.1;

// mRNA in proteini za represor R3.
Product repressor3 = new Product();
repressor3.mRNADegradationPercent = 0.45;
repressor3.ProteinDegradationPercent = 0.1;
```

Atributi podajajo vrednosti degradacijskih konstant za represorske mRNA in proteine. Nato definiramo vezavna mesta, promotorje in gene represilatorja z nekim testnim naborom vrednosti atributov (parametrov):

```
// Vezavna mesta za konstrukte 1, 2, 3.
BindingSite Site1 = new BindingSite();
Site1.Capacity = 100;
BindingSite Site2 = new BindingSite();
Site2.Capacity = 100;
BindingSite Site3 = new BindingSite();
Site3.Capacity = 100;

// Seznam vezavnih mest promotorja 1.
List<BindingSite> List1 = new List<BindingSite>();
List1.Add(Site1);

// Seznam vezavnih mest promotorja 2.
List<BindingSite> List2 = new List<BindingSite>();
```

```
List2.Add(Site2);

// Seznam vezavnih mest promotorja 3.
List<BindingSite> List3 = new List<BindingSite>();
List3.Add(Site3);

// Promotor za konstrukt 1.
Promoter Promoter1 = new Promoter();
Promoter1.Type = PromoterType.Constitutive;
Promoter1.BindingSites = List1;
Promoter1.ActiveTranscriptionRate = 15;
Promoter1.KR = 10;
Promoter1.RepressorExp = 2.0;

// Promotor za konstrukt 2.
Promoter2.Type = PromoterType.Constitutive;
Promoter2.BindingSites = List2;
Promoter2.ActiveTranscriptionRate = 15;
Promoter2.KR = 10;
Promoter2.RepressorExp = 2.0;

// Promotor za konstrukt 3.
Promoter3.Type = PromoterType.Constitutive;
Promoter3.BindingSites = List3;
Promoter3.ActiveTranscriptionRate = 15;
Promoter3.KR = 10;
Promoter3.RepressorExp = 2.0;

// Gen za represor R1.
Gene Repressor1 = new Gene();
Repressor1.GenePromoter = Promoter1;

// Gen za represor R2.
Gene Repressor2 = new Gene();
Repressor2.GenePromoter = Promoter2;

// Gen za represor R3.
Gene Repressor3 = new Gene();
Repressor3.GenePromoter = Promoter3;
```

Časovnega zamika med transkripcijo in translacijo ne upoštevamo, prav tako ne puščanja. Po določitvi začetnih koncentracij represorjev - $R1 = 50$, $R2 = 100$, $R3 = 80$ - izvajamo simulacijo kot serijo časovnih korakov:

```
for (int t = 0; t < 300; t++)
{
    // Vezava represorja R3 na promotor 1.
    Promoter1.BindingSites[0].BindRepressors(repressor3.Proteins);

    // Vezava represorja R1 na promotor 2.
    Promoter2.BindingSites[0].BindRepressors(repressor1.Proteins);

    // Vezava represorja R2 na promotor 3.
    Promoter3.BindingSites[0].BindRepressors(repressor2.Proteins);

    // Transkripcija gena R1.
    Repressor1.Transcribe(repressor1.mRNAs, t);

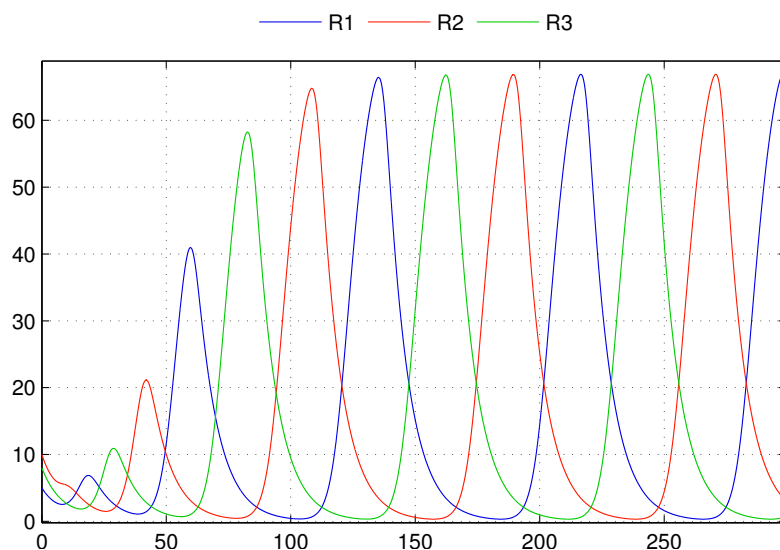
    // Transkripcija gena R2.
    Repressor2.Transcribe(repressor2.mRNAs, t);

    // Transkripcija gena R3.
    Repressor3.Transcribe(repressor3.mRNAs, t);

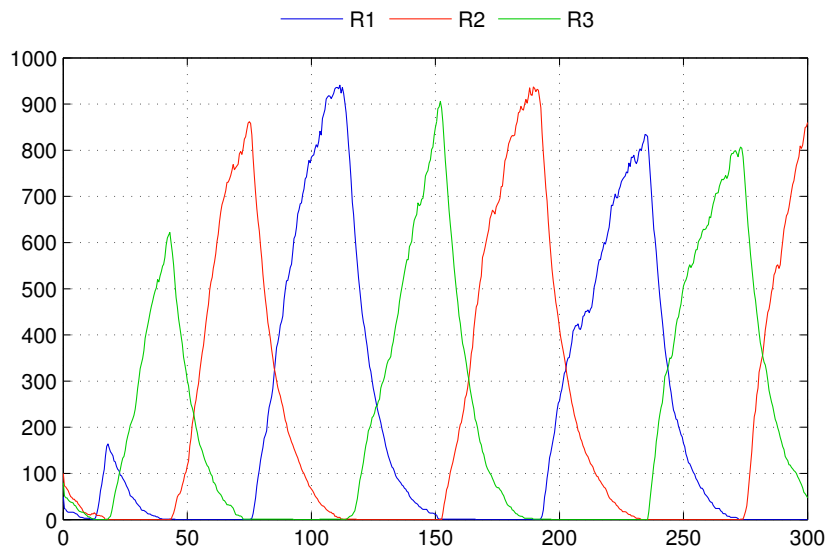
    // Translacija mRNA posameznih represorjev v represorske proteine.
    repressor1.Translate(t);
    repressor2.Translate(t);
    repressor3.Translate(t);

    // Degradacija mRNA in represorskih proteinov.
    repressor1.Degrade();
    repressor2.Degrade();
    repressor3.Degrade();
}
```

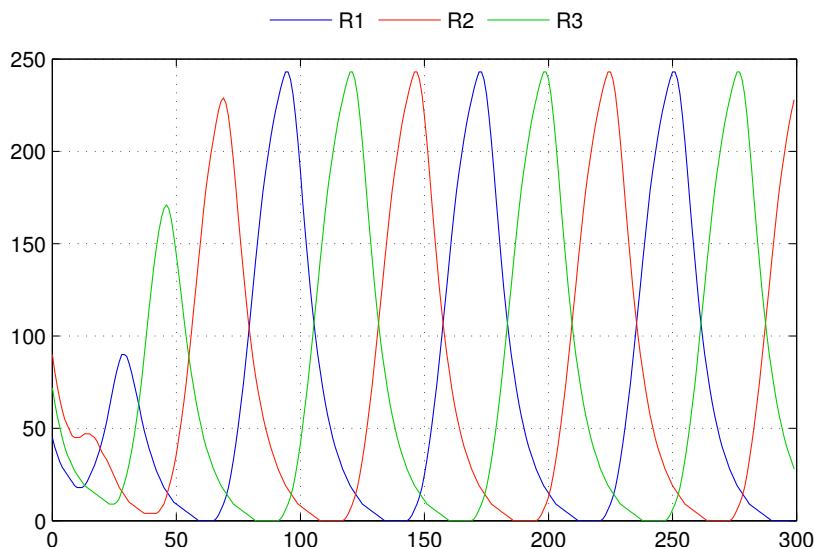
Rezultati simulacije z izključeno stohastičnostjo so prikazani na sliki 4.4 in kažejo oscilacije.



Slika 4.2 Koncentracije treh represorjev (R_1 , R_2 , R_3) v odvisnosti od časa za deterministični model represilatorja na osnovi verjetnostne interpretacije izražanja genov v skladu z enačbami (4.7) - (4.9). Začetne koncentracije represorjev so $[R_1] = 5$, $[R_2] = 10$, $[R_3] = 8$. Ostali parametri so: $k_1 = k_2 = k_3 = 10$, $q_1 = q_2 = q_3 = 0,12$; $K_a = K_b = K_c = K_1 = K_2 = K_3 = 1$, $n_1 = n_2 = n_3 = 3$.



Slika 4.3 Koncentracije treh represorjev (R_1 , R_2 , R_3) v odvisnosti od časa za stohastični model represilatorja. Začetne koncentracije represorjev so $R_1 = 50$, $R_2 = 100$, $R_3 = 80$. Ostali parametri so $k = 10$, $q = 0,1$; $a = 10$, $d = 0,01$; $m = 2$. Začetne vrednosti za promotorje so $Pro1 = Pro2 = Pro3 = 10$.



Slika 4.4 Spreminjanje koncentracij treh represorjev ($R1$, $R2$, $R3$) s časom za model represilatorja, implementiran s HDA brez stohastičnosti.

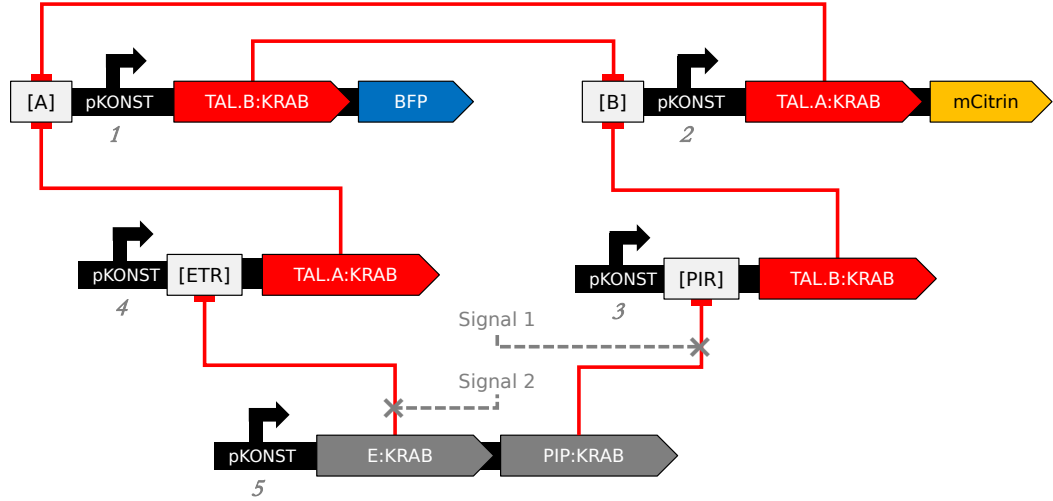
4.2 Navadno bistabilno stikalo z represorji TAL

Z gensko regulatornimi omrežji lahko realiziramo preklopna bistabilna stikala. Značilnost slednjih je, da jih lahko z zunanjimi signali (npr. ustreznimi molekulami) preklapimo v eno izmed dveh stabilnih stanj - z enim signalom v eno stanje, z drugim signalom pa v drugo. Tipično je za eno stabilno stanje značilna visoka koncentracija prvega proteina glede na koncentracijo drugega proteina, za drugo stabilno stanje pa ravno obratno. Pomembna lastnost takih stikal je, da se aktualno stanje obdrži tudi po odstranitvi vhodnega signala, s katerim smo stanje inducirali, npr. po razgraditvi induktorskih molekul - v kolikor se stanje ne obdrži, ni stabilno. Iz logičnega vidika imajo bistabilna genetska stikala torej spomin. Primer njihove aplikativne uporabe je npr. proizvodjanje različnih zdravil na obolenem mestu v telesu - posamezen protein je neko zdravilo, z zunanjimi signali v obliki npr. tablete pa stikalo kontrolirano preklaplja med proizvodjanjem enega od dveh različnih zdravil, kar lahko vodi v bolj kvalitetno terapijo [3].

V tem podglavju je predstavljeno modeliranje bistabilnega stikala, ki za regulacijo transkripcije uporablja represorje TAL [3]. Konceptualno temelji na kontrolirani medsebojni represiji dveh genov, ki se nahajata pod kontrolo vsak svojega promotorja oz. vsak na svojem konstrukt [23]. Stikalo uporablja dva represorja TAL, $TAL.A : KRAB$ in $TAL.B : KRAB$. Prvi se veže na vezavno mesto $[A]$, drugi pa na vezavno mesto $[B]$.

Celotno strukturo in opis stikala podaja slika 4.5. Stanje stikala lahko preklapljam z dvema vhodnima signaloma, signalom 1 in signalom 2.

Pri determinističnem in stohastičnem modelu zaradi poenostavitve ignoriramo več ponovitev vsakega vezavnega mesta in privzamemo le eno.



Slika 4.5 Struktura navadnega bistabilnega stikala z represorji TAL. Obsega pet oštevilčenih konstruktov s konstitutivnimi promotorji. Vsako vezavno mesto obsega 10 ponovitev. Izhodno stanje stikala določata dva reporterska proteina, *BFP* in *mCitrin*. Prisotnost vhodnega signala 1 povzroči, da se represor *PIP : KRAB* odveže od ciljnega vezavnega mesta [*PIR*], kar prekine represijo promotorja 3. Prisotnost vhodnega signala 2 pa povzroči, da se represor *E : KRAB* odveže od ciljnega vezavnega mesta [*ETR*], kar prekine represijo promotorja 4. Z vhodnim signalom 1 želimo inducirati stabilno stanje, kjer bo koncentracija *BFP* visoka, koncentracija *mCitrina* pa nizka. Z drugim vhodnim signalom pa želimo inducirati stabilno stanje, kjer bo ravno obratno. Prisotnost obeh signalov hkrati je logično nedovoljeno oz. nedefinirano stanje.

4.2.1 Deterministični model

Postopek izpeljave determinističnega modela je enak kot pri represilatorju. Ker imamo opravka le s konstitutivnimi promotorji, bo delež zasedenosti za vse enake oblike - razlika je le v nastopajočem represorju. Tako velja:

$$f_1 = \frac{K_r^{n_1}}{K_r^{n_1} + K_1 \cdot [TAL.A : KRAB]^{n_1}}, \quad (4.25)$$

$$f_2 = \frac{K_r^{n_2}}{K_r^{n_2} + K_2 \cdot [TAL.B : KRAB]^{n_2}}, \quad (4.26)$$

$$f_3 = \frac{K_r^{n_3}}{K_r^{n_3} + K_3 \cdot [PIP : KRAB]^{n_3}}, \quad (4.27)$$

$$f_4 = \frac{K_r^{n_4}}{K_r^{n_4} + K_4 \cdot [E : KRAB]^{n_3}}. \quad (4.28)$$

Diferencialne enačbe, ki podajajo spreminjanje koncentracij proteinov stikala v času, so:

$$\frac{d[BFP]}{dt} = k_{BFP} \cdot f_1 + kb_{BFP} \cdot (1 - f_1) - q_{BFP} \cdot [BFP], \quad (4.29)$$

$$\frac{d[mCitrin]}{dt} = k_{Cit} \cdot f_2 + kb_{Cit} \cdot (1 - f_2) - q_{Cit} \cdot [mCitrin], \quad (4.30)$$

$$\frac{d[TAL.B : KRAB]}{dt} = B_{KR1} + B_{KR3} - q_{BKR} \cdot [TAL.B : KRAB], \quad (4.31)$$

$$\frac{d[TAL.A : KRAB]}{dt} = A_{KR2} + A_{KR4} - q_{AKR} \cdot [TAL.A : KRAB], \quad (4.32)$$

$$\frac{d[PIP : KRAB]}{dt} = k_{PIP} - q_{PIP} \cdot [PIP : KRAB], \quad (4.33)$$

$$\frac{d[E : KRAB]}{dt} = k_E - q_E \cdot [E : KRAB]. \quad (4.34)$$

k_i so konstante hitrosti produkcije, kb_i konstante hitrosti puščanja in q_i konstante hitrosti degradacije. $TAL.B : KRAB$ izražata dva gena, ki sta na konstruktih 1 in 3, zato je njegova produkcija (vključujoč puščanje) seštevek $B_{KR1} + B_{KR3}$, kjer je B_{KR1} izražanje gena $TAL.B : KRAB$ konstrukta 1, B_{KR3} pa izražanje gena $TAL.B : KRAB$ konstrukta 3:

$$B_{KR1} = k_{1BKR} \cdot f_1 + kb_{1BKR} \cdot (1 - f_1), \quad (4.35)$$

$$B_{KR3} = k_{3BKR} \cdot f_3 + kb_{3BKR} \cdot (1 - f_3). \quad (4.36)$$

$TAL.A : KRAB$ se producira pri izražanju genov konstruktov 2 (kar podaja A_{KR2}) in 4 (kar podaja A_{KR4}):

$$A_{KR2} = k_{2AKR} \cdot f_2 + kb_{2AKR} \cdot (1 - f_2), \quad (4.37)$$

$$A_{KR4} = k_{4AKR} \cdot f_4 + kb_{4AKR} \cdot (1 - f_4). \quad (4.38)$$

Prisotnost vhodnih signalov modeliramo na naslednji način:

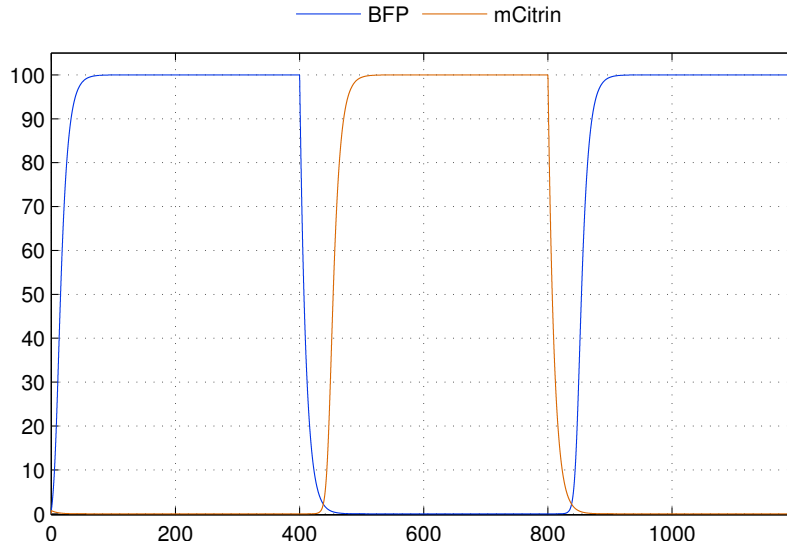
- ob prisotnosti signala 1 se $PIP : KRAB$ hitro degradira (q_{PIP} je npr. 1000-krat večji kot sicer) in posledično skoraj popolnoma izgine iz sistema;
- ob prisotnosti signala 2 se $E : KRAB$ hitro degradira (q_E je npr. 1000-krat večji kot sicer) in posledično skoraj popolnoma izgine iz sistema.

V simulaciji uporabljamo naslednje zaporedje preklapljanja stanj:

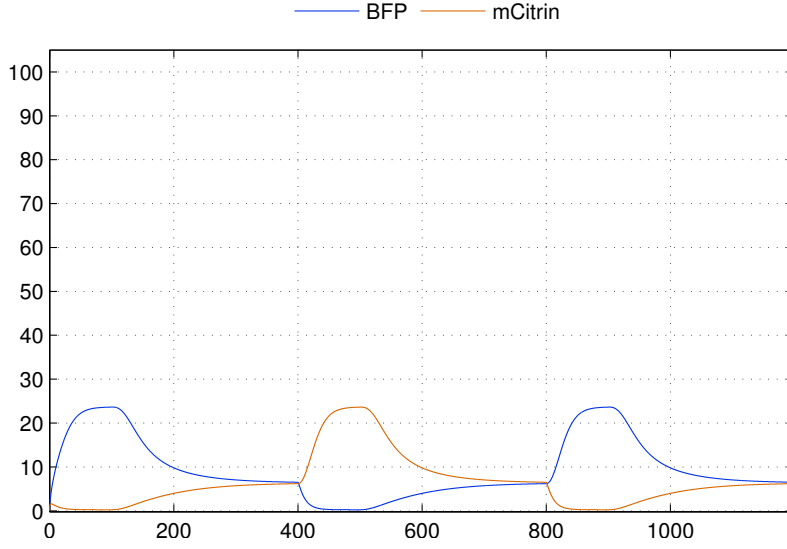
1. ob času $t = 0$ induciramo stanje 1 (dodamo signal 1 in ga umaknemo ob času $t = 100$);

2. ob času $t = 400$ induciramo stanje 2 (dodamo signal 2 in ga umaknemo ob času $t = 500$);
3. ob času $t = 800$ induciramo ponovno stanje 1 (signal 1 umaknemo ob času $t = 900$).

Rezultat simulacije determinističnega modela za testne parametre prikazuje slika 4.6. Značilnost navadnega stikala je slaba odpornost na puščanje, ki že za relativno nizke vrednosti - te so v praktični realizaciji presežene - hitro rezultira v izgubi bistabilnosti. Odpornost na puščanje do neke mere zvišuje večja kooperativnost. Vendar lahko iz strukture efektorjev TAL sklepamo, da se vežejo na DNA neodvisno oz. nekooperativno [3]. Prav tako stikalo ne vsebuje drugih mehanizmov, ki bi vpeljali nelinearnost. Slika 4.7 prikazuje rezultat simulacije, če upoštevamo puščanje in nizko kooperativnost oz. nizke koeficiente nelinearnosti - v tem primeru stikalo ne izkazuje bistabilnosti. Eksperimentalna realizacija stikala ni pokazala bistabilnosti, v skladu z napovedmi modeliranja [3].



Slika 4.6 Preklapljanje navadnega bistabilnega stikala med stanji za deterministični model na osnovi verjetnostne interpretacije izražanja genov ob odsotnosti puščanja ($kb_i = 0$) za kooperativnost vseh represorjev $n_i = 3$. V tem primeru stikalo izkazuje bistabilnost. Prikazana je koncentracija obeh reporterskih proteinov v odvisnosti od časa. Ostali parametri, različni od 1, so $k_i = 10$, $q_i = 0, 1$ in $K_r = 3$. Začetne koncentracije proteinov so enake nič.



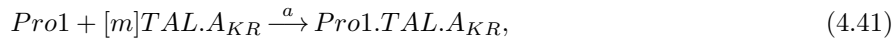
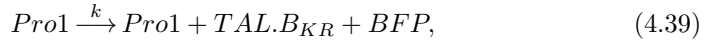
Slika 4.7 Preklapljanje navadnega bistabilnega stikala med stanji za deterministični model na osnovi verjetnostne interpretacije izražanja genov. Prisotno je puščanje $kb_i = 0, 1$. Koeficienti nelinearnosti so $n_i = 1, 2$. V tem primeru stikalo ne izkazuje bistabilnosti. Prikazana je koncentracija obeh reporterskih proteinov v odvisnosti od časa. Ostali parametri, različni od 1, so $k_i = 10$, $q_i = 0, 1$ in $K_r = 3$. Začetne koncentracije proteinov so enake nič.

4.2.2 Stohastični model

Za stohastični model zaradi preglednosti pri hitrostih posameznih reakcij (produkcije, vezave, odcepitve, degradacije) predpostavimo simetričnost - uporabimo enake konstante hitrosti za posamezne tipe reakcij.

Pomen oznak v reakcijah je naslednji: Pro_i predstavlja promotor konstrukta i , $TAL.B_{KR}$ predstavlja $TAL.B : KRAB$, $TAL.A_{KR}$ predstavlja $TAL.A : KRAB$, PIP_{KR} predstavlja $PIP : KRAB$, E_{KR} pa $E : KRAB$. Prvi vhodni signal (signal 1) je označen z I_1 , drugi vhodni signal (signal 2) pa z I_2 . Ostale oznake so intuitivne oz. razložene v nadaljevanju.

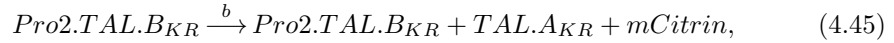
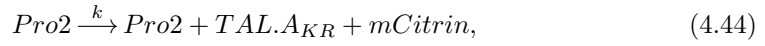
Reakcije za konstrukt 1 so:



Pomen reakcij od vrha navzdol je naslednji:

1. konstitutivno (nerepresirano) izražanje genov pod promotorjem 1;
2. izražanje genov pod represiranim promotorjem 1 (puščanje) - na oz. ob promotor je vezan $TAL.A : KRAB$;
3. vezava $TAL.A : KRAB$ ob promotor 1 - rezultat je represiran promotor $Pro1.TAL.A_{KR}$;
4. odcepitev $TAL.A : KRAB$ od promotorja 1;
5. degradacija vezanega $TAL.A : KRAB$.

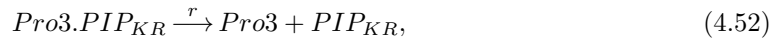
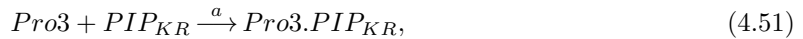
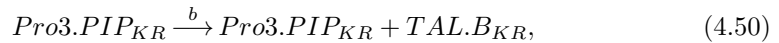
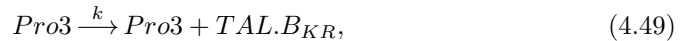
Reakcije za konstrukt 2 so:



Pomen reakcij od vrha navzdol je naslednji:

1. konstitutivno izražanje genov pod promotorjem 2;
2. izražanje genov pod represiranim promotorjem 2 (puščanje) - na oz. ob promotor je vezan $TAL.B : KRAB$;
3. vezava $TAL.B : KRAB$ ob promotor 2 - rezultat je represiran promotor $Pro2.TAL.B_{KR}$;
4. odcepitev $TAL.B : KRAB$ od promotorja 2;
5. degradacija vezanega $TAL.B : KRAB$.

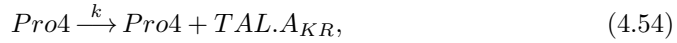
Reakcije za konstrukt 3 so:



Pomen reakcij od vrha navzdol je naslednji:

1. konstitutivno izražanje genov pod promotorjem 3;
2. izražanje genov pod represiranim promotorjem 3 (puščanje) - na oz. ob promotor je vezan $PIP : KRAB$;
3. vezava $PIP : KRAB$ ob promotor 3 - rezultat je represiran promotor $Pro3.PIP_{KR}$;
4. odcepitev $PIP : KRAB$ od promotorja 3;
5. degradacija vezanega $PIP : KRAB$.

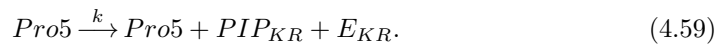
Reakcije za konstrukt 4 so:



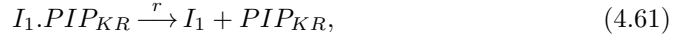
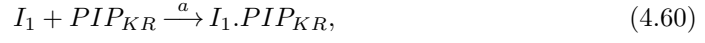
Pomen reakcij od vrha navzdol je naslednji:

1. konstitutivno izražanje genov pod promotorjem 4;
2. izražanje genov pod represiranim promotorjem 4 (puščanje) - na oz. ob promotor je vezan $E : KRAB$;
3. vezava $E : KRAB$ ob promotor 4 - rezultat je represiran promotor $Pro4.E_{KR}$;
4. odcepitev $E : KRAB$ od promotorja 4;
5. degradacija vezanega $E : KRAB$.

Reakcija za konstrukt 5 podaja zgolj konstitutivno izražanje dveh genov pod promotorjem 5, $PIP : KRAB$ in $E : KRAB$:



Vpliv vhodnih signalov podajajo reakcije:



Pomen reakcij od vrha navzdol je naslednji:

1. vezava signala 1 (I_1) na $PIP : KRAB$ povzroči neaktiven $PIP : KRAB$ (poimenovan $I_1.PIP_{KR}$), ki se ne more vezati ob promotor 3;
2. odcepitev signala 1 od $PIP : KRAB$;
3. degradacija neaktivnega $PIP : KRAB$;
4. degradacija signala 1 (signalnih molekul);
5. vezava signala 2 (I_2) na $E : KRAB$ povzroči neaktiven $E : KRAB$ (poimenovan $I_2.E_{KR}$), ki se ne more vezati ob promotor 4;
6. odcepitev signala 2 od $E : KRAB$;
7. degradacija neaktivnega $E : KRAB$;
8. degradacija signala 2.

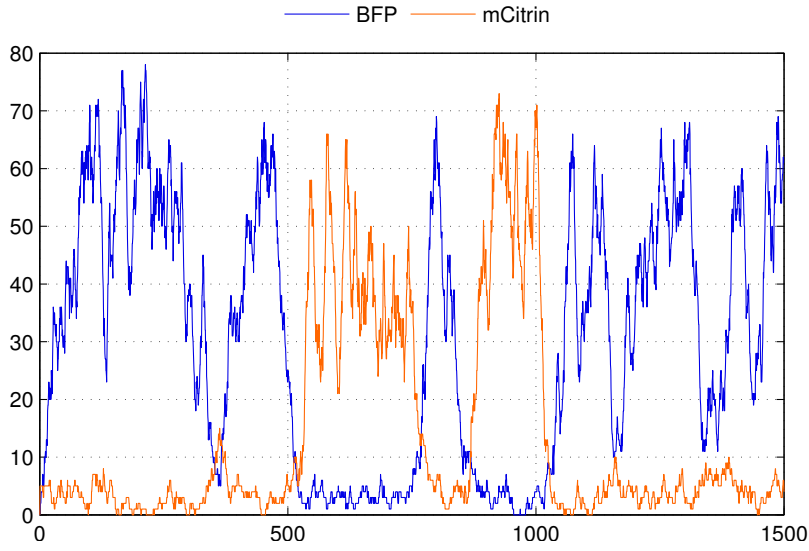
Degradacije proteinov podajajo reakcije:



Rezultat testne stohastične simulacije prikazuje slika 4.8. Vhodni signali so vpeljani na naslednji način:

1. ob času $t = 0$ dodamo 8000 molekul prvega signala (I_1);
2. ob času $t = 500$ dodamo 8000 molekul drugega signala (I_2);
3. ob času $t = 1000$ dodamo 8000 molekul prvega signala (I_1).

Molekule vhodnih signalov se hitro razgradijo in so prisotne le delček časa z namenom induciranja stabilnega stanja.



Slika 4.8 Preklapljanje navadnega bistabilnega stikala med stanji za stohastični model brez kooperativnosti ($m = 1$). Konstanta za puščanje je $b = 0,03$. V tem primeru stikalo ne izkazuje izrazite bistabilnosti, saj pride do neželenih preklapov. Prikazana je koncentracija obeh reporterskih proteinov v odvisnosti od časa. Ostale konstante so $k = a = 1$, $q = 0,1$ in $r = 0,01$. Začetne koncentracije proteinov so enake nič.

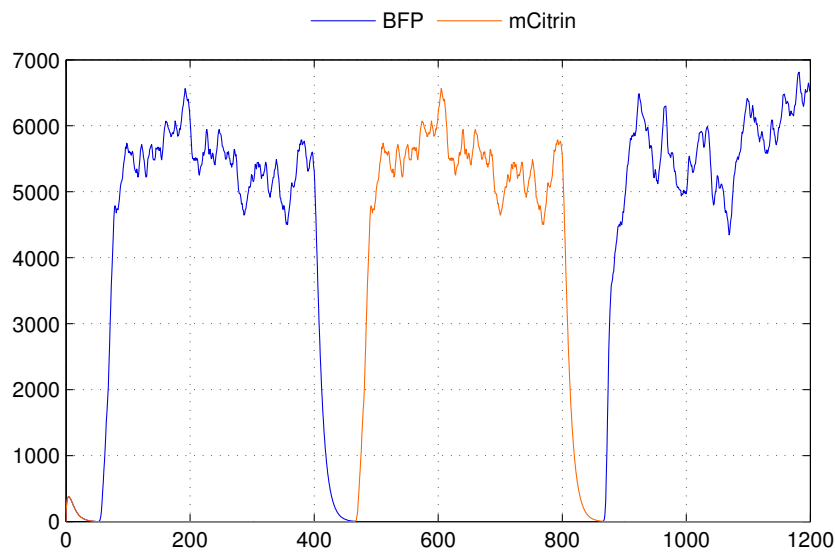
4.2.3 Realizacija modela s hibridnim diskretnim algoritmom

Pri realizaciji s HDA postopamo enako kot pri modeliranju represilatorja. Kapaciteto vezavnih mest nastavimo na 10, saj ima vsako vezavno mesto 10 ponovitev. Vpliv posameznega vhodnega signala na sistem modeliramo npr. tako, da v vsakem koraku simulacije po klicih metod za vezavo transkripcijskih faktorjev preverimo njegovo prisotnost in preprečimo vezavo $PIP : KRAB$ oz. $E : KRAB$ na ciljno mesto z držanjem količine vezanega represorja na vrednosti nič:

```
// Ob prisotnosti signala 1 efektivno zatri vezavo PIP:KRAB.
if (signal1 == 1) Promoter3.BindingSites[0].AmountOfBoundRepressor = 0;

// Ob prisotnosti signala 2 efektivno zatri vezavo E:KRAB.
if (signal2 == 1) Promoter4.BindingSites[0].AmountOfBoundRepressor = 0;
```

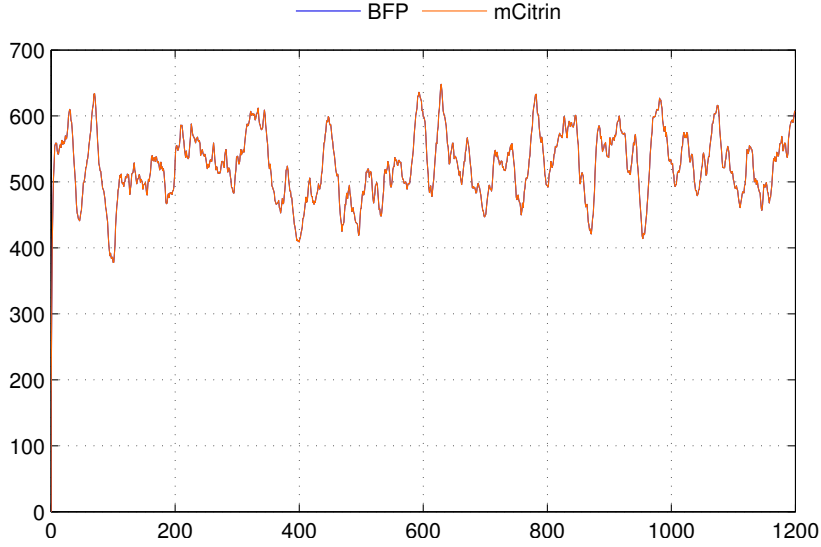
Predpostavljen je enak časovni scenarij preklapljanja kot pri prvem determinističnem modelu. Rezultat simulacije v primeru brez puščanja podaja slika 4.9. Že majhno puščanje je dovolj, da stikalo preneha izkazovati bistabilnost, kar prikazuje slika 4.10.



Slika 4.9 Preklapljanje navadnega bistabilnega stikala med stanji za model s HDA. Prikazana je koncentracija obeh reporskih proteinov v odvisnosti od časa. Puščanja ni, kooperativnost vseh represorjev (koeficient nelinearnosti) je enaka 3. Stikalo v tem primeru izkazuje bistabilnost. Ostali parametri HDA so: $k = 200$ (aktivna transkripcija), $q_M = 0,25$ (odstotek mRNA za degradacijo), $q_P = 0,1$ (odstotek proteinov za degradacijo), $K_R = 1$, $\lambda = 0,75$ (stohastičnost transkripcije), $P_T = 0,75$ (učinkovitost translacije). Začetne koncentracije proteinov so enake nič.

4.3 Bistabilno stikalo s pozitivnima povratnima zankama

Nadgradnja navadnega stikala je bistabilno stikalo s pozitivnima povratnima zankama [3]. Poleg dveh represorjev TAL ($TAL.A : KRAB$ in $TAL.B : KRAB$) uporablja še dva aktivatorja TAL - $TAL.A : VP16$ in $TAL.B : VP16$. Pri tem $TAL.A : KRAB$ in $TAL.A : VP16$ v primeru sočasnega obstoja tekmujeta za vezavo na isto vezavno mesto. Enako velja za $TAL.B : KRAB$ in $TAL.B : VP16$. Stikalo tako za delovanje izkorišča kompetitivno vezavo aktivatorja in represorja za isto vezavno mesto. Celotno strukturo

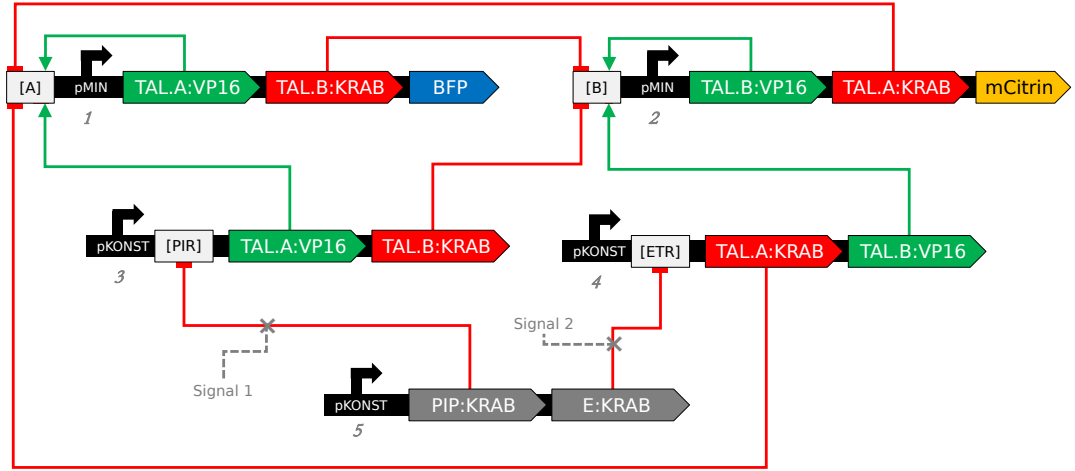


Slika 4.10 Poskus preklapljanja navadnega bistabilnega stikala med stanji za model s HDA v primeru prisotnosti puščanja vsakega promotorja ($b = 20$ - 10% konstitutivne transkripcije), ki povzroči izgubo bistabilnosti. Priказana je koncentracija obeh reporterskih proteinov v odvisnosti od časa. Kooperativnost vseh represorjev (koeficient nelinearnosti) je enaka 3. Ostali parametri HDA so: $k = 200$ (aktivna transkripcija), $q_M = 0,25$ (odstotek mRNA za degradacijo), $q_P = 0,1$ (odstotek proteinov za degradacijo), $K_R = 1$, $\lambda = 0,75$ (stohastičnost transkripcije), $P_T = 0,75$ (učinkovitost translacije). Začetne koncentracije proteinov so enake nič.

in opis stikala podaja slika 4.11.

Pozitivni povratni zanki, ki sta realizirani s povratno vezavo aktivatorjev na prvem in drugem konstrukt, sta pomembni za večjo robustnost stikala napram navadnemu stikalu. Omogočata, da stikalo lahko izkazuje bistabilnost kljub odsotnosti kooperativnosti, tj. za koeficiente nelinearnosti enake ali zelo blizu 1. Teoretično je bilo namreč pokazano, da je bistabilnost možna brez multimerične regulacije, če obstaja vsaj ena regulatorna zanka [24]. Prav tako so bile teoretično odkrite bistabilne arhitekture brez kooperativnosti transkripcijskih faktorjev, ki pa so vsebovale kompetitivno vezavo slednjih [25]. V nadaljevanju opisani modeli napovejo, da stikalo s pozitivnima povratnima zankama lahko izkazuje bistabilnost kljub relativno visokemu puščanju in odsotnosti kooperativnosti. Eksperimentalna realizacija stikala je potrdila bistabilno delovanje [3].

Pri determinističnem in stohastičnem modelu zaradi poenostavitve ignoriramo več ponovitev vsakega vezavnega mesta in privzamemo le eno ponovitev.



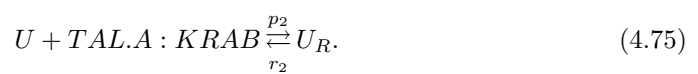
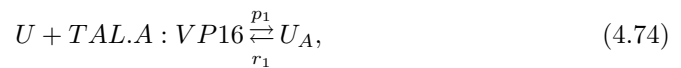
Slika 4.11 Struktura bistabilnega stikala s pozitivnima povratnima zankama. Obsega pet oštevilčenih konstruktov. Vsako vezavno mesto obsega 10 ponovitev. Ciljno vezavno mesto transkripcijskih faktorjev, baziranih na *TAL.A* - tj. aktivatorja *TAL.A : VP16* in represorja *TAL.A : KRAB* - je vezavno mesto *[A]*. Enako je ciljno vezavno mesto za aktivator *TAL.B : VP16* in represor *TAL.B : KRAB* vezavno mesto *[B]*. Izhodno stanje enako kot v že opisanem navadnem stikalu določata dva reporterska proteina, *BFP* in *mCitrine*. Prvo stabilno stanje (visoka koncentracija *BFP*) induciramo s signalom 1, drugo stabilno stanje (visoka koncentracija *mCitrina*) pa s signalom 2. Prisotnost vhodnega signala 1 povzroči, da se represor *PIP : KRAB* odveže od ciljnega vezavnega mesta *[PIR]*, prisotnost vhodnega signala 2 pa povzroči, da se represor *E : KRAB* odveže od ciljnega vezavnega mesta *[ETR]*. Prisotnost obeh signalov hkrati je logično nedovoljeno oz. nedefinirano stanje. Aktivator na prvem in drugem konstruktu se veže ob promotor istega konstrukta in s tem pospešuje lastno transkripcijo - temu pravimo pozitivna povratna zanka.

4.3.1 Deterministični model

Na vezavno mesto *[A]* in *[B]* se lahko vežeta tako aktivator kot represor. Če se osredotočimo na prvi konstrukto (z vezavnim mestom *[A]*), so možna naslednja stanja minimalnega promotorja:

- U - prosto vezavno mesto: neaktiven promotor;
- U_R - na vezavno mesto je vezan represor *TAL.A : KRAB*: neaktiven promotor;
- U_A - na vezavno mesto je vezan aktivator *TAL.A : VP16*: aktiven promotor;

Prehajanje med stanji promotorja podajata reakciji:



V ravnovesju lahko pišemo:

$$U_A = K_1 \cdot U \cdot [TAL.A : VP16], \quad (4.76)$$

$$U_R = K_2 \cdot U \cdot [TAL.A : KRAB]. \quad (4.77)$$

Delež zasedenosti za prvi promotor je podan z izrazom

$$f_1 = \frac{U_A}{U + U_A + U_R}. \quad (4.78)$$

Posplošena oblika pa je določena z izrazom

$$f_1 = \frac{K_1 \cdot [TAL.A : VP16]^{n_1}}{K_a^{n_1} + K_1 \cdot [TAL.A : VP16]^{n_1} + K_2 \cdot [TAL.A : KRAB]^{n_2}}. \quad (4.79)$$

Na enak način lahko izpeljemo delež zasedenosti za drugi promotor:

$$f_2 = \frac{K_3 \cdot [TAL.B : VP16]^{n_3}}{K_a^{n_3} + K_3 \cdot [TAL.B : VP16]^{n_3} + K_4 \cdot [TAL.B : KRAB]^{n_4}}. \quad (4.80)$$

Za promotor tretjega in četrtega konstrukta je rezultat enak kot za navadno stikalo:

$$f_3 = \frac{K_r^{n_5}}{K_r^{n_5} + K_5 \cdot [PIP : KRAB]^{n_5}}, \quad (4.81)$$

$$f_4 = \frac{K_r^{n_6}}{K_r^{n_6} + K_6 \cdot [E : KRAB]^{n_6}}. \quad (4.82)$$

K_a in K_r imajo v vseh simulacijah vrednost 1, enako velja za konstante K_1 - K_6 .

Diferencialne enačbe, ki podajajo spreminjanje koncentracij proteinov stikala v času, so:

$$\frac{d[BFP]}{dt} = k_{BFP} \cdot f_1 + kb_{BFP} \cdot (1 - f_1) - q_{BFP} \cdot [BFP], \quad (4.83)$$

$$\frac{d[mCitrin]}{dt} = k_{Cit} \cdot f_2 + kb_{Cit} \cdot (1 - f_2) - q_{Cit} \cdot [mCitrin], \quad (4.84)$$

$$\frac{d[TAL.B : KRAB]}{dt} = B_{KR1} + B_{KR3} - q_{BKR} \cdot [TAL.B : KRAB], \quad (4.85)$$

$$\frac{d[TAL.A : KRAB]}{dt} = A_{KR2} + A_{KR4} - q_{AKR} \cdot [TAL.A : KRAB], \quad (4.86)$$

$$\frac{d[TAL.B : VP16]}{dt} = B_{VP2} + B_{VP4} - q_{BVP} \cdot [TAL.B : VP16], \quad (4.87)$$

$$\frac{d[TAL.A : VP16]}{dt} = A_{VP1} + A_{VP3} - q_{AVP} \cdot [TAL.A : VP16], \quad (4.88)$$

$$\frac{d[PIP : KRAB]}{dt} = k_{PIP} - q_{PIP} \cdot [PIP : KRAB], \quad (4.89)$$

$$\frac{d[E : KRAB]}{dt} = k_E - q_E \cdot [E : KRAB]. \quad (4.90)$$

Gena $TAL.B : KRAB$ se nahajata na konstruktih 1 in 3; produkcijo za vsakega izmed teh genov podajata člena:

$$B_{KR1} = k_{1BKR} \cdot f1 + kb_{1BKR} \cdot (1 - f1), \quad (4.91)$$

$$B_{KR3} = k_{3BKR} \cdot f3 + kb_{3BKR} \cdot (1 - f3). \quad (4.92)$$

Gena $TAL.A : KRAB$ se nahajata na konstruktih 2 in 4, zato:

$$A_{KR2} = k_{2AKR} \cdot f2 + kb_{2AKR} \cdot (1 - f2), \quad (4.93)$$

$$A_{KR4} = k_{4AKR} \cdot f4 + kb_{4AKR} \cdot (1 - f4). \quad (4.94)$$

Gena $TAL.B : VP16$ se nahajata na konstruktih 2 in 4, zato:

$$B_{VP2} = k_{2BVP} \cdot f2 + kb_{2BVP} \cdot (1 - f2), \quad (4.95)$$

$$B_{VP4} = k_{4BVP} \cdot f4 + kb_{4BVP} \cdot (1 - f4). \quad (4.96)$$

Gena $TAL.A : VP16$ pa se nahajata na konstruktih 1 in 3, zato velja:

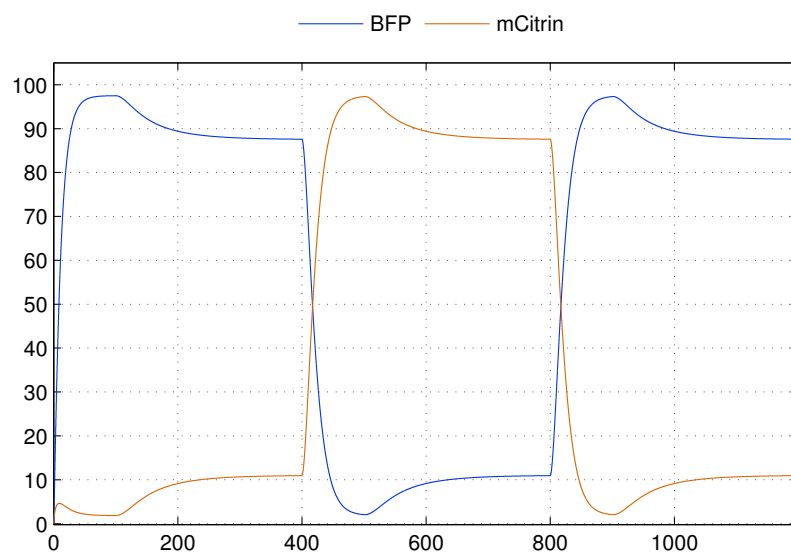
$$A_{VP1} = k_{1AVP} \cdot f1 + kb_{1AVP} \cdot (1 - f1), \quad (4.97)$$

$$A_{VP3} = k_{3AVP} \cdot f3 + kb_{3AVP} \cdot (1 - f3). \quad (4.98)$$

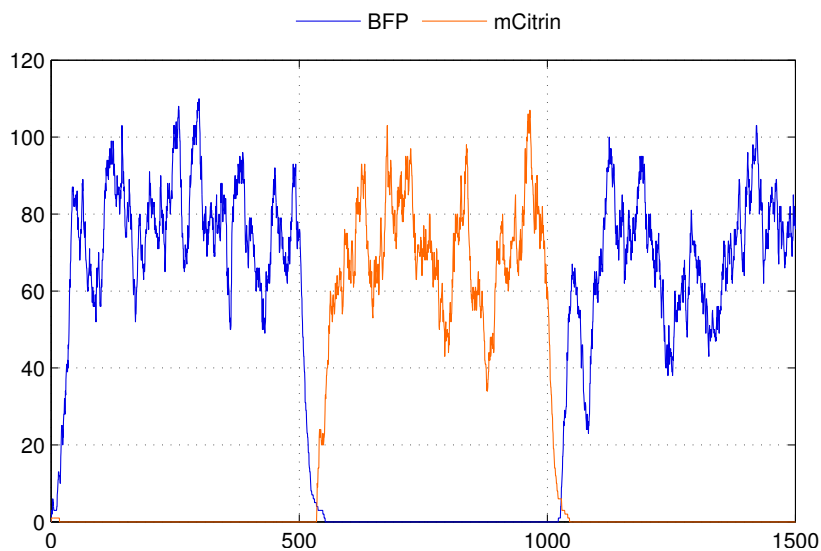
Primer rezultata simulacije za postavljeni model prikazuje slika 4.12.

4.3.2 Stohastični model

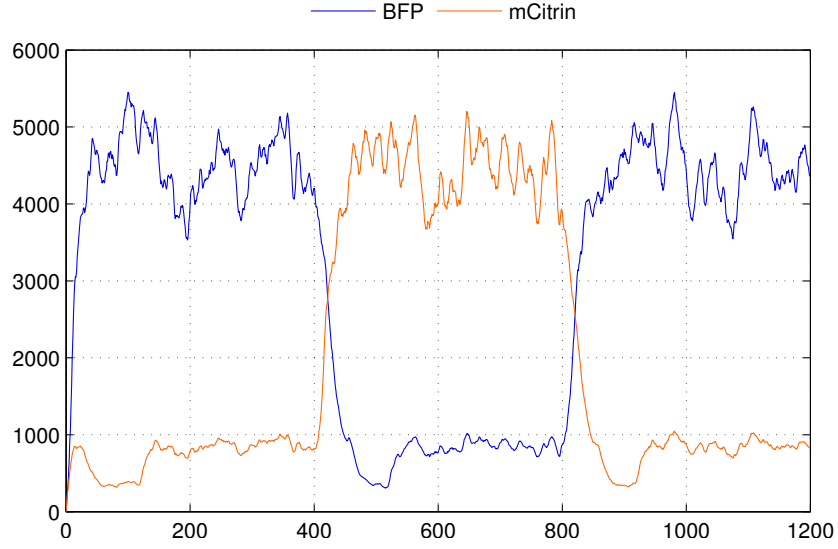
Za stohastični model kot pri navadnem stikalu uporabimo enake konstante hitrosti za posamezne tipe reakcij. Oznake, uporabljene v modelih, nosijo enak pomen kot pri navadnem stikalu. Na novo sta vpeljeni le oznaki $TAL.A_{VP}$ in $TAL.B_{VP}$ - prva pomeni aktivator $TAL.A : VP16$, druga pa aktivator $TAL.B : VP16$.



Slika 4.12 Preklapljanje bistabilnega stikala s pozitivnima povratnima zankama za deterministični model na osnovi verjetnostne interpretacije izražanja genov. Puščanje je $kb_i = 0,25$. Koefficienti nelinearnosti so $n_i = 1,2$. V tem primeru stikalo za razliko od navadnega stikala izkazuje bistabilnost in večjo robustnost. Prikazana je koncentracija obeh reporterskih proteinov v odvisnosti od časa. Ostali parametri, različni od 1, so $k_i = 10$ in $q_i = 0,1$. Začetne koncentracije proteinov so enake nič.

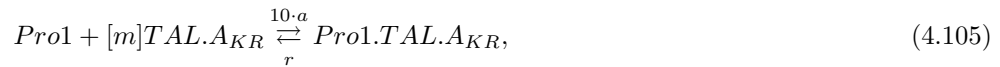
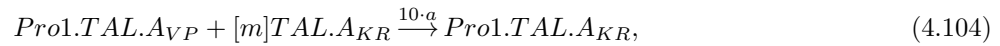
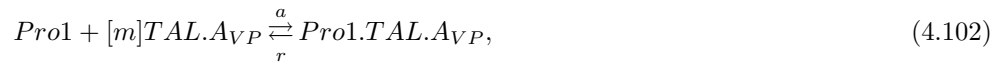
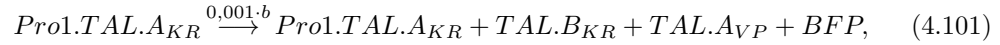
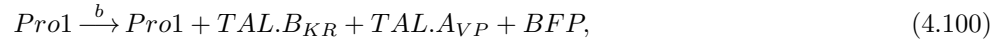


Slika 4.13 Preklapljanje bistabilnega stikala s pozitivnima povratnima zankama za stohastični model. Kljub odsotnosti kooperativnosti ($m = 1$) in puščanju ($b = 0,03$) stikalo izkazuje bistabilnost. Prikazana je koncentracija obeh reporterskih proteinov v odvisnosti od časa. Ostale konstante so $k = a = 1$, $q = 0,1$ in $r = 0,01$. Začetne koncentracije proteinov so enake nič.



Slika 4.14 Preklapljanje bistabilnega stikala s pozitivnima povratnima zankama za model s HDA. Puščanje vsakega promotorja je enako 10% aktivne transkripcije ($b = 20$, $k = 200$). Stikalo deluje bistabilno (za razliko od navadnega stikala, ki pri takem puščanju kljub višji kooperativnosti ni izkazovalo bistabilnosti). Koefficienti nelinearnosti so enaki 1,2. Parametri stohastičnosti so $\lambda = 0,75$ in $P_T = 0,75$. Začetne koncentracije proteinov so enake nič.

Reakcije za konstrukt 1 z minimalnim promotorjem lahko zapišemo na naslednji način:

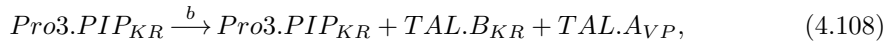


Pomen reakcij od vrha navzdol je naslednji:

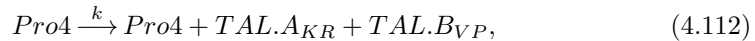
1. aktivirano izražanje genov pod promotorjem 1 - na oz. ob promotor je vezan aktivator $TAL.A_{VP}$;
2. puščanje minimalnega promotorja ob odsotnosti transkripcijskih faktorjev;

3. represirano izražanje genov, ko je vezan represor $TAL.A_{KR}$ - konstanta hitrosti reakcije je v tem primeru še 1000-krat nižja kot v primeru puščanja prostega promotorja (gre za testno vrednost, ki odraža dejstvo, da vezan represor še zmanjša hitrost transkripcije v primerjavi s situacijo, ko ni vezan noben transkripcijski faktor);
4. vezava oz. odcepitev aktivatorja $TAL.A_{VP}$;
5. degradacija vezanega aktivatorja $TAL.A_{VP}$;
6. reakcija, ki odraža večjo moč represije v primerjavi z aktivacijo: če je ob promotor že vezan aktivator, bo vezava represorja potekala 10-krat hitreje od vezave aktivatorja, kar bo efektivno pomenilo večjo vrednost represiranega stanja $Pro1.TAL.A_{KR}$;
7. vezava oz. odcepitev represorja $TAL.A_{KR}$;
8. degradacija vezanega represorja $TAL.A_{KR}$.

Reakcije za konstrukt 2 so enake oblike, le da nastopajo reaktanti oz. produkti konstrukta 2, npr. *mCitrin* namesto *BFP* itd. Reakcije za konstrukta 3 in 4 so enake oblike kot pri navadnem stikalu, upoštevamo pa še nove gene za aktivatorje. Reakcije za konstrukt 3 so tako:



Reakcije za konstrukt 4 pa so:



Reakcija za konstrukt 5 in reakcije za vpliv vhodnih signalov so enake kot za navadno stikalo. Enake so tudi reakcije za degradacije, le da dodamo še degradacijo obeh aktivatorjev TAL:



Slika 4.13 prikazuje primer stohastične simulacije opisanega modela. Časovni scenarij preklapov (vnosa signalov) je enak kot za navadno stikalo.

4.3.3 Realizacija modela s hibridnim diskretnim algoritmom

Pri realizaciji s HDA ni večjih posebnosti. Kapaciteto vezavnih mest tudi tokrat nastavimo na 10, ker ima vsako vezavno mesto 10 ponovitev. Kompetitivno vezavo aktivatorja in represorja realiziramo s klicem metode *BindTFs()*. Gene, ki kodirajo enak protein, a nastopajo na več konstruktih - npr. *TAL.B : KRAB* - definiramo posebej, pri transkripciji pa nastale molekule mRNA za oba gena shranjujemo v isti seznam. Efektivno gre torej za seštevanje hitrosti transkripcije na podoben način kot pri determinističnem modelu na osnovi verjetnostne interpretacije izražanja genov.

Če zanemarimo kodo za časovno preklapljanje (npr. ob času $t = 0$ nastavi signal 1 na vrednost 1), v vsakem koraku simulacije (ob času t) izvajamo naslednje korake:

```
// Na promotorjih 1 in 2 poteka kompetitivna vezava, na promotorjih 3 in 4
// pa vezava represorjev.
Promoter1.BindingSites[0].BindTFs(TAL_A_VP16.Proteins, TAL_A_KRAB.Proteins);
Promoter2.BindingSites[0].BindTFs(TAL_B_VP16.Proteins, TAL_B_KRAB.Proteins);
Promoter3.BindingSites[0].BindRepressors(PIPKRAB.Proteins);
Promoter4.BindingSites[0].BindRepressors(EKRAB.Proteins);

// Vpliv vhodnih signalov.
if (signal1 == 1) Promoter3.BindingSites[0].AmountOfBoundRepressor = 0;
if (signal2 == 1) Promoter4.BindingSites[0].AmountOfBoundRepressor = 0;

// Transkripcija genov - mRNA shranimo v seznam mRNAs
// TAL.B:KRAB na 1. konstrukt:
genTAL_B_KRAB1.Transcribe(TAL_B_KRAB.mRNAs, t);
// TAL.B:KRAB na 3. konstrukt:
genTAL_B_KRAB3.Transcribe(TAL_B_KRAB.mRNAs, t);
// TAL.B:VP16 na 2. konstrukt:
```

```

genTAL_B_VP162.Transcribe(TAL_B_VP16.mRNAs, t);
// TAL.B:VP16 na 4. konstruktu:
genTAL_B_VP164.Transcribe(TAL_B_VP16.mRNAs, t);
// TAL.A:KRAB na 2. konstruktu:
genTAL_A_KRAB2.Transcribe(TAL_A_KRAB.mRNAs, t);
// TAL.A:KRAB na 4. konstruktu:
genTAL_A_KRAB4.Transcribe(TAL_A_KRAB.mRNAs, t);
// TAL.A:VP16 na 1. konstruktu:
genTAL_A_VP161.Transcribe(TAL_A_VP16.mRNAs, t);
// TAL.A:VP16 na 3. konstruktu:
genTAL_A_VP163.Transcribe(TAL_A_VP16.mRNAs, t);
// Ostali geni:
genBFP.Transcribe(BFP.mRNAs, t);
genCitrin.Transcribe(mCitrin.mRNAs, t);
genPIPKRAB.Transcribe(PIPKRAB.mRNAs, t);
genEKRAB.Transcribe(EKRAB.mRNAs, t);

// Translacija. Vsi objekti (BFP, mCitrin itd.) so tipa Product.
BFP.Translate(t);
mCitrin.Translate(t);
TAL_B_KRAB.Translate(t);
TAL_B_VP16.Translate(t);
TAL_A_KRAB.Translate(t);
TAL_A_VP16.Translate(t);
PIPKRAB.Translate(t);
EKRAB.Translate(t);

// Degradacija mRNA in proteinov.
BFP.Degrade();
mCitrin.Degrade();
TAL_B_KRAB.Degrade();
TAL_B_VP16.Degrade();
TAL_A_KRAB.Degrade();
TAL_A_VP16.Degrade();
PIPKRAB.Degrade();
EKRAB.Degrade();

```

Primer rezultata simulacije prikazuje slika 4.14.

5 Zaključek

V diplomski nalogi smo predstavili glavne pristope k modeliranju gensko regulatornih omrežij (GRO) in lastni doprinos v obliki novega algoritma. Potreba po modeliranju GRO izvira predvsem iz njihovega pogosto neintuitivnega oz. težko preverljivega delovanja. Razložili smo osnovne predpostavke in izhodišča modeliranja. Pokazali smo primer determinističnega modeliranja na osnovi verjetnostne interpretacije izražanja genov, kjer dinamiko omrežja formuliramo upoštevajoč aktivna in neaktivna stanja promotorja. Take modele smo zapisali z navadnimi diferencialnimi enačbami. Ker je izražanje genov v osnovi stohastičen proces, smo razložili koncept stohastičnega modeliranja. Kot tretji pristop smo predstavili nov algoritem za modeliranje GRO - hibridni diskretni algoritem. Sposoben je eksplicitno modelirati vezavo transkripcijskih faktorjev in morebitno kompetitivno vezavo, njegova objektna zasnova pa omogoča hitro postavitev modela brez potrebe po sistemskem določanju enačb ali reakcij - dinamika modela je namreč določena z univerzalnimi pravili oz. enačbami, ki so sorodna empirično osnovanim ostalim pristopom, npr. Hillovim funkcijam, a obenem predstavljajo razširitev slednjih. Algoritem je hibriden, ker v sicer deterministično osnovo vpelje stohastične člene. Vsaka molekula

je modelirana kot svoj objekt, zato bi lahko algoritem v prihodnosti razširili z novimi dinamikami, npr. 3D prostorskim gibanjem v celici, difuzijo, ipd.

Podali smo primere modeliranja treh eksperimentalno preverjenih GRO - represilatorja in dveh bistabilnih stikal, ki ju lahko kontroliramo z zunanjimi signali. Aplikativno predstavljajo taka stikala potencial v medicini. Vsa našeta GRO smo modelirali deterministično, stohastično in s hibridnim diskretnim algoritmom - poudarek je bil na osnovni postavitvi modelov, veliko več prostora pa bi lahko namenili še podrobnejši analizi obnašanja in simulaciji. Vsi modeli so v vseh primerih pravilno napovedali glavne karakteristike teh omrežij - oscilacije v primeru represilatorja, slabo odpornost na puščanje navadnega bistabilnega stikala ter večjo robustnost izboljšanega stikala s pozitivnima povratnima zankama. Modeli za navadno stikalo so pokazali, da obstajajo teoretični pogoji za bistabilnost, vendar pa so malo verjetni v praksi. Napoved modelov - torej odsotnost bistabilnosti v realnih celičnih razmerah - se ujema z eksperimentalnimi rezultati [3]. Velik pomen predstavlja predvsem modeliranje drugega, izboljšanega stikala - modeli so napovedali njegovo bistabilno delovanje kljub puščanju in minimalni ali ničelni kooperativnosti transkripcijskih faktorjev. Tudi v tem primeru se je napoved modelov ujemala z eksperimentalnimi rezultati, izboljšano stikalo pa predstavlja sploh prvo eksperimentalno realizacijo bistabilnega stikala brez kooperativnosti [3].

Modele bi lahko skušali izboljšati z upoštevanjem fenomenov, kot sta npr. delitev celic in mutacije DNA. Prav tako bi lahko izražanje genov modelirali natančneje, tj. z upoštevanjem več vmesnih korakov - slabost v tem primeru bi bilo več novih težko določljivih parametrov. Zanimivo bi bilo primerjati obnašanje takih modelov z bolj grobimi. Veliko pozornosti bi lahko namenili modeliranju okolja oz. organizma kot nosilca gensko regulatornega omrežja. Pri vsem tem je potrebno upoštevati, da je v biologiji veliko stvari še neznanih, zato je nujno verificiranje in dopolnjevanje modeliranja z bazičnimi in eksperimentalnimi izsledki.

LITERATURA

- [1] Kevin Bullis. Making Diesel from CO₂ and Sunlight, December 2009. Dostopno na: <http://www.technologyreview.com/news/416622/making-diesel-from-co2-and-sunlight>.
- [2] iGEM 2012 Team Cornell. SAFE BET: The Shewanella Assay for Extended Bio-monitoring of Environmental Toxins, 2012. Dostopno na: <http://2012.igem.org/Team:Cornell>.
- [3] iGEM 2012 Team Slovenia. Switch-IT (Inducible therapeutics), 2012. Dostopno na: <http://2012.igem.org/Team:Slovenia>.
- [4] iGEM 2012 Team Groningen. The Food Warden. It's rotten and you know it!, 2012. Dostopno na: <http://2012.igem.org/Team:Groningen>.
- [5] David L. Nelson and Michael M. Cox. *Lehninger Principles of Biochemistry*. W.H. Freeman, 5th edition, 2008.
- [6] Dale Jeremy W., Malcolm von Schantz, and Nicholas Plant. *From Genes to Genomes: Concepts and Applications of DNA Technology*. Wiley, 3rd edition, 2011.
- [7] Robert J. Brooker, Eric P. Widmaier, Linda E. Graham, and Peter D. Stiling. *Biology*. McGraw-Hill, 2nd edition, 2011.
- [8] Ina Koch, Wolfgang Reisig, and Falk Schreiber. *Modeling in Systems Biology: The Petri Net Approach*. Springer, 2011.
- [9] Cecie Starr, Christine A. Evers, and Lisa Starr. *Biology: Concepts and Applications*. Brooks Cole, 8th edition, 2010.

- [10] Heidi Scholze, Jana Streubel, and Jens Boch. TAL effectors from *Xanthomonas*: design of a programmable DNA-binding specificity. *Bulletin of Insectology*, 64:279–280, 2011.
- [11] Brian Ingalls. *An Introduction to Dynamic Mathematical Modelling in Molecular Systems Biology*. University of Waterloo, 2012.
- [12] Herbert M. Sauro. *Enzyme Kinetics for Systems Biology*. Ambrosius Publishing, 1st edition, 2011.
- [13] Stefan Lorkowski and Paul Cullen. *Analysing Gene Expression: A Handbook of Methods: Possibilities and Pitfalls*. Wiley-VCH, 2003.
- [14] Uri Alon. *An Introduction to Systems Biology: Design Principles of Biological Circuits*. Chapman and Hall/CRC, 2007.
- [15] Mads Kærn, Timothy C. Elston, William J. Blake, and James J. Collins. Stochasticity in gene expression: from theories to phenotypes. *Nature Reviews Genetics*, 6:451–464, 2005.
- [16] Mircea Andrecut, Julianne D. Halley, David A. Winkler, and Sui Huang. A General Model for Binary Cell Fate Decision Gene Circuits with Degeneracy: Indeterminacy and Switch Behavior in the Absence of Cooperativity. *PLoS ONE*, 6, 2011.
- [17] Mads Kærn, William J. Blake, and J.J. Collins. The engineering of gene regulatory networks. *Annual Review of Biomedical Engineering*, 5:179–206, 2003.
- [18] Jeff Hasty, David McMillen, Farren Isaacs, and James J. Collins. Computational studies of gene regulatory networks: *in numero* molecular biology. *Nature Reviews Genetics*, 2:268–279, 2001.
- [19] Andre S. Ribeiro and Jason Lloyd-Price. SGN Sim, a Stochastic Genetic Networks Simulator. *Bioinformatics*, 23(6):777–779, 2007.
- [20] Gabriele Lillacci and Mustafa Khammash. Parameter Estimation and Model Selection in Computational Biology. *PLoS Comput Biol*, 6, 2010.
- [21] I-Chun Chou and Eberhard O. Voit. Recent developments in parameter estimation and structure identification of biochemical and genomic systems. *Mathematical Biosciences*, 219(2):57–83, 2009.

- [22] Michael B. Elowitz and Stanislas Leibler. A synthetic oscillatory network of transcriptional regulators. *Nature*, 403, 2000.
- [23] Timothy S. Gardner, Charles R. Cantor, and James J. Collins. Construction of a genetic toggle switch in *Escherichia coli*. *Nature*, 403(6767):339–42, 2000.
- [24] Stefanie Widder, Javier Macía, and Ricard Solé. Monomeric Bistability and the Role of Autoloops in Gene Regulation. *PLoS ONE*, 4(4), 2009.
- [25] Dan Siegal-Gaskins, Maria K. Mejia-Guerra, Gregory D. Smith, and Erich Gro-tewold. Emergence of Switch-Like Behavior in a Large Family of Simple Biochemical Networks. *PLoS Comput Biol*, 7(5), 2011.