

MASARYKOVA UNIVERZITA
FAKULTA INFORMATIKY



Rozšíření a refaktORIZÁCIA nástroja BioDiVinE

DIPLOMOVÁ PRÁCA

Martin Demko

Brno, jar 2014

Prehlásenie

Prehlasujem, že táto diplomová práca je mojím pôvodným autorským dielom, ktoré som vypracoval samostatne. Všetky zdroje, pramene a literatúru, ktoré som pri vypracovaní používal alebo z nich čerpal, v práci riadne citujem s uvedením úplného odkazu na príslušný zdroj.

Vedúci práce: RNDr. David Šafránek, Ph.D.

Pod'akovanie

Zhrnutie

Klíčové slova

Obsah

1	Základné pojmy	3
1.1	<i>LTL - Lineárna Temporálna Logika</i>	3
1.2	<i>Büchiho automat</i>	5
1.3	<i>Zákon o mass action kinetike</i>	6
1.4	<i>Michaelis-Mentenovej a Hillova kinetika</i>	7
1.5	<i>Model checking</i>	11
1.5.1	Prevod LTL do BA	14
1.5.2	Farebný model checking	14
2	Biochemický dynamický vstupný model	16
2.1	<i>Abstrakcia</i>	19
3	Východiskový stav a podobné nástroje	20
3.1	<i>BioDiVinE 1.0</i>	20
3.2	<i>PEPMC</i>	20
3.3	<i>RoverGene</i>	20
4	BioDiVinE 1.1	21
5	Implementácia	22
6	Použitie programu	23
7	Case study	24
8	Záver	25

Úvod

Kapitola 1

Základné pojmy

Než začneme zachádzať hlbšie do problematiky tejto práce a opisovať postupy a nástroje v nej použité, je potrebné vysvetliť na počiatku niekoľko pojmov. Tieto sa v práci mnohokrát opakujú a ich včasným uvedením predídeme nepochopiteľnosti textu.

1.1 LTL - Lineárna Temporálna Logika

Temporálna logika obecné je špeciálna vetva logiky zaoberajúca sa logickou štruktúrou výrokov v čase. Je to formalizmus vhodný pre overovanie vlastností formálnych dynamických systémov resp. matematických modelov.

Lineárna temporálna logika (ďalej len LTL) je najjednoduchšia verzia temporálnej logiky, ktorá neumožňuje vetvenie času ani kvantifikátory. Môžeme ju považovať tiež za konkrétny výpočtový kalkulus pracujúci s tzv. formulami, definovanými nasledujúcou syntaxou:

Atomické propozície (ďalej len *AP*)

$A > 0$
 $B \leq 5.834$
 $C \neq \text{"nie"}$
atd'...

Logické operátory

\neg, \vee	— základné logické operátory
$\wedge, \rightarrow, \leftrightarrow, \text{true}, \text{false}$	— odvodené logické operátory

Temporálne operátory

- $X\phi$ - *neXt*, vyjadruje platnosť ϕ v ďalšom stave
- $G\phi$ - *Global*, vyjadruje trvalú platnosť ϕ

- $\mathbf{F}\phi$ - **Future**, vyjadruje platnosť ϕ v niektorom z budúcich stavov
- $\psi\mathbf{U}\phi$ - **Until**, vyjadruje platnosť ψ až do kedy nezačne platiť ϕ
- $\psi\mathbf{R}\phi$ - **Release**, vyjadruje platnosť ϕ , až dokiaľ nezačne platiť ψ a to vrátane tohto bodu. Ak ψ nikdy nezačne platiť, musí ϕ platiť do nekonečna

kde ϕ a ψ sú atomické propozície

Potom platí nasledujúce:

- Ak $p \in AP$, tak p je formula.
- Ak f a g sú formuly, tak $\neg g$, $f \vee g$, $f \wedge g$, $f \rightarrow g$, $f \leftrightarrow g$, $\mathbf{X}g$, $\mathbf{F}g$, $\mathbf{G}g$, $f\mathbf{U}g$ a $g\mathbf{R}f$ sú formuly.

Takto vytvorená LTL formula môže byť splniteľná nekonečnou postupnosťou pravdivých vyhodnotení jednotlivých $p \in AP$. Túto postupnosť si možno predstaviť ako nekonečné slovo w , pre ktoré platí $w = a_0, a_1, a_2, \dots$ a kde a_i je pravdivostná hodnota nejakej $p \in AP$. Nech $w(i) = a_i$ a $w^i = a_i, a_{i+1}, \dots$ je podpostupnosť alebo sufix slova w . Potom formálna definícia relácie splniteľnosti \models medzi slovom w a LTL formulou vyzerá:

- $w \models p$, ak $p \in AP \wedge p = w(0)$
- $w \models \neg\phi$, ak ϕ a ψ sú LTL formuly $\wedge w \not\models \phi$
- $w \models \phi \vee \psi$, ak $w \models \phi \vee w \models \psi$
- $w \models \mathbf{X}\phi$, ak $w^1 \models \phi$
- $w \models \phi\mathbf{U}\psi$, ak $\exists i, i \geq 0 \wedge w^i \models \psi \wedge \forall k, 0 \leq k < i \wedge w^k \models \phi$

Predchádzajúce platí pre základné logické a temporálne operátory, ktoré majú ale dostatočne expresívnu silu, aby s ich pomocou mohli byť zadané ľubovoľné LTL formuly. Ovšem pre uľahčenie zápisu aj čítania, si môžeme dodefinovať rozšírenú paletu operátorov za predpokladu platnosti predchádzajúcich pravidiel:

- $\phi \wedge \psi \equiv \neg(\neg\phi \vee \neg\psi)$, ak ϕ, ψ sú LTL formuly
- $\phi \rightarrow \psi \equiv \neg\psi \vee \phi$,
- $\phi \leftrightarrow \psi \equiv (\phi \rightarrow \psi) \vee (\psi \rightarrow \phi)$,
- $\mathbf{true} \equiv p \vee \neg p$, ak $p \in AP$

- $\text{false} \equiv \neg \text{true}$,
- $\phi \mathbf{R} \psi \equiv \neg(\neg\phi \mathbf{U} \neg\psi)$,
- $\mathbf{F}\phi \equiv \text{true} \mathbf{U} \phi$,
- $\mathbf{G}\phi \equiv \neg \mathbf{F} \neg \phi$,

Napriek tomu, že je LTL tou najprimitívnejšou temporálnou logikou, jej prevod do Büchiho automatu je v najhoršom prípade exponenciálne zložitý. Dôvod tohto prevodu bude vysvetlený v kapitole 1.5. [2]

1.2 Büchiho automat

Automat obecné je matematický model stroja s konečným množstvom pamäte spracováajúci vstup o neznámej veľkosti. Kvôli obmedzeniu pamäte ho nazývame konečným automatom. Vstup sa nazýva slovo a môže byť konečný aj nekonečný. Büchiho automat je potom najjednoduchším konečným automatom nad nekonečným slovom a preto patrí do skupiny ω -automatov.

Formálne je konečný automat \mathcal{A} päťica $(\Sigma, Q, \Delta, Q_0, F)$, pre ktorú platí:

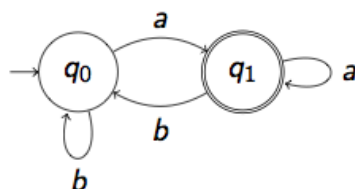
- Σ je konečná abeceda
- Q je konečná množina stavov
- $\Delta \subseteq Q \times \Sigma \times Q$ je relácia nazývaná prechodová funkcia
- $Q_0 \subseteq Q$ je podmnožina množiny stavov, nazývaná iníciaľne stavy
- $F \subseteq Q$ je podmnožina množiny stavov, nazývaná akceptujúce stavy

Príklad jednoduchého automatu je daný na obrázku 1.1.

Automat nad konečným slovom akceptuje toto slovo, ak po prejdení posleného znaku slova zodpovedajúceho prechodu medzi dvoma stavmi, je tento posledný stav v množine F . Avšak automat nad nekonečným slovom nemôže nikdy prejsť cez posledný znak. Preto takýto automat akceptuje nekonečné slovo len v prípade, že počas prechádzania slova je aspoň jeden stav navštívený nekonečne často a zároveň tento stav patrí aj do množiny F .

Automaty môžeme ešte rozlíšiť na deterministické a nedeterministické. Deterministický automat má jednoznačne určené prechody medzi stavmi. Tým sa myslí, že zo stavu $q \in Q$ sa pod znakom $s \in \Sigma$ dá prejsť maximálne do jedného stavu $q' \in Q$. Naproti tomu nedeterministické automaty

umožňujú prechod zo stavu q pod slovom s do stavov $Q' \subseteq Q$. Našťastie existuje algoritmus prevodu nedeterministického konečného automatu na deterministický, ale iba nad konečným slovom. Nevýhodou je ale veľký nárast počtu stavov. [3]



Obr. 1.1: Jednoduchý deterministický automat $\mathcal{A} : \Sigma = \{a, b\}$, $Q = \{q_0, q_1\}$, $\Delta = \{(q_0, a, q_1), (q_0, b, q_0), (q_1, a, q_1), (q_1, b, q_0)\}$, $Q_0 = \{q_0\}$, $F = \{q_1\}$

1.3 Zákon o mass action kinetike

Tento zákon vyjadruje základné pravidlo fungovania chemických reakcií. Konkrétne rýchlosť s akou chemické substancie, či už veľké makromolekuly alebo malé ióny, do seba narážajú a interagujú za tvorby nových chemických látok. Predpokladajme, že substráty A a B spolu reagujú za vzniku novej látky, produktu C ,



Rýchlosť tejto reakcie predstavuje rýchlosť tvorby produktu C , a síce $\frac{d[C]}{dt}$, ktorá stelesňuje počet kolízií za jednotku času medzi reaktantami A a B a zároveň pravdepodobnosť, že tieto kolízie majú dostatočnú energiu na prekonanie aktivačnej energie reakcie. To samozrejme závisí po prvé na koncentrácii reaktantov, ale aj na ich tvare a veľkosti a tiež na teplote a pH roztoku. Kombináciou týchto a aj ďalších faktorov dostávame:

$$\frac{d[C]}{dt} = k[A][B] \quad (1.2)$$

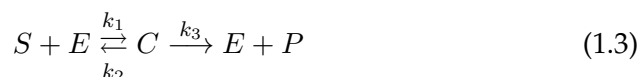
Identifikácia vzťahu 1.2 s rovnicou 1.1 sa nazýva zákon mass action kinetiky (z ang. law of mass action) a konštanta k je potom rýchlostná konštanta (z ang. rate constant) tejto reakcie.

V skutočnosti nejde o zákon ako taký. Nie je to neporušiteľné pravidlo ako Newtonov gravitačný zákon, ale skôr veľmi užitočný model, ktorý však nemusí byť vo všetkých prípadoch validný. [6]

1.4 Michaelis-Mentenovej a Hillova kinetika

Je nutné začať od enzýmovej kinetiky, pretože práve tá bola hlavným katalyzátorom pre objavenie nových spôsobov modelovania chemických reakcií. Tiež je dobré si uvedomiť, prečo tomu tak bolo. Enzýmová kinetika totiž patrí medzi niekoľko prípadov, v ktorých použitie mass action kinetiky nie je validné. Podľa nej sa so zvyšujúcou koncentráciou substrátu S zvyšuje rýchlosť reakcie zlučovania s enzýmom E lineárne. Zatiaľ čo v *in-vivo* prípade táto rýchlosť postupne konverguje k určitému maximu, cez ktoré sa nedá dostať ani dodatočným zvýšením koncentrácie substrátu S .

Model, ktorý vysvetľoval túto odchýlku od zákona o mass action kinetike bol prvý krát prezentovaný v roku 1913 nemeckým biochemikom Leonardom Michaelisom a kanadskou fyzičkou Maud Mentenovou (z toho kinetika *Michaelis-Mentenovej*). V ich reakcii premieňal enzým E substrát S na produkt P v dvoch fázach. Prvá bola reverzibilná reakcia zlučovania S s E za vzniku enzým-substrátového komplexu C a druhá predstavovala rozpad komplexu C za vzniku produktu P a uvoľnenia nezmeneného enzýmu E (viď rovnicu 1.3).



Je dôležité si všimnúť, že druhá reakcia nie je reverzibilná.

Existujú dva spôsoby ako analyzovať túto rovnicu a obe sú si veľmi podobné. Ide o rovnovážnu aproximáciu a aproximáciu kvázistacionárneho stavu. Začneme aplikáciou zákona o mass action na tieto reakcie, čo nám vo výsledku dá nasledujúce diferenciálne rovnice vyjadrujúce rýchlosti zmien jednotlivých chemických látok:

$$\frac{d[S]}{dt} = k_2[C] - k_1[S][E], \quad (1.4)$$

$$\frac{d[E]}{dt} = (k_2 + k_3)[C] - k_1[S][E], \quad (1.5)$$

$$\frac{d[C]}{dt} = k_1[S][E] - (k_2 + k_3)[C], \quad (1.6)$$

$$\frac{d[P]}{dt} = k_3[C]. \quad (1.7)$$

Všimnite si, že platí $\frac{d[C]}{dt} + \frac{d[E]}{dt} = 0$ a preto

$$[E] + [C] = [E_0], \quad (1.8)$$

kde $[E_0]$ je absolútna koncentrácia enzýmu v reakcii. [7]

A. Rovnovážna aproximácia

V pôvodnej analýze Michaelis a Mentenová predpokladali, že substrát je v neustálej rovnováhe s enzým-substrátovým komplexom, a teda že platí

$$k_1[S][E] = k_2[C]. \quad (1.9)$$

Potom na základe platnosti vzťahov 1.8 a 1.9, môžeme zdefinovať nasledujúci vzťah:

$$[C] = \frac{[E_0][S]}{K_s + [S]}, \quad (1.10)$$

kde $K_s = \frac{k_2}{k_1}$. Z toho zase vypláva, že rýchlosť reakcie V , respektíve rýchlosť tvorby produktu P môže byť zadaná takto:

$$V = \frac{d[P]}{dt} = k_3[C] = \frac{k_3[E_0][S]}{K_s + [S]} = \frac{V_{max}[S]}{K_s + [S]}, \quad (1.11)$$

kde $V_{max} = k_3[E_0]$ je maximálna reakčná rýchlosť a predstavuje prípad, keď všetky molekuly enzýmu sú naviazané na substrát.

Pri malej koncentrácii substrátu je rýchlosť reakcie lineárna, ak táto koncentrácia nepresiahne celkové množstvo enzýmu. Avšak pri väčších koncentráciách substrátu je rýchlosť reakcie limitovaná množstvom enzýmu a disociačnou konštantou (v našom prípade k_3). Ak sa koncentrácia substrátu približuje hodnote K_s , znamená to, že reakčná rýchlosť je rovná polovičke svojho maxima.

Poznamenajme však, že vzťah 1.9 v reálnych podmienkach takmer nikdy neplatí a rýchlosť reakcie je tak ovplyvnená aj disociačnou konštantou enzým-substrátového komplexu v smere spätného rozpadu (v našom prípade k_2). Práve z tohto dôvodu tu hovoríme o aproximácii. [7]

B: Aproximácia kvázistacionárneho stavu

Inou alternatívou analýzy tejto enzymatickej reakcie je práve aproximácia kvázistacionárneho stavu (z ang. Quasi-steady state approximation), ktorá je v dnešnej dobe aj najpoužívanejšia. Jej tvorcovia, George Briggs a J.B.S. Haldane predpokladali, že rýchlosť tvorby enzým-substrátového komplexu aj jeho disociácia sú si od počiatku rovné. Takže platí $\frac{d[C]}{dt} = 0$.

Aby sme dali tejto aproximácii správny matematický základ, je vhodné zadefinovať nasledujúce bezrozmerné ¹ premenné:

$$\begin{aligned}\sigma &= \frac{[S]}{[S_0]}, & \chi &= \frac{[C]}{[E_0]}, & \tau &= k_1[E_0]t, \\ \kappa &= \frac{k_2 + k_3}{k_1[S_0]}, & \epsilon &= \frac{[E_0]}{[S_0]}, & \alpha &= \frac{k_2}{k_1[S_0]},\end{aligned}\quad (1.12)$$

s pomocou ktorých dostaneme systém iba dvoch diferenciálnych rovníc:

$$\frac{d\sigma}{d\tau} = -\sigma + \chi(\sigma + \alpha), \quad (1.13)$$

$$\epsilon \frac{d\chi}{d\tau} = \sigma - \chi(\sigma + \kappa). \quad (1.14)$$

V porovnaní s koncentráciou substrátu nejakej reakcie je koncentrácia enzýmu vo väčšine prípadov oveľa menšia. Táto skutočnosť pekne odzrkadľuje efektívnosť enzýmov ako katalyzátorov chemických reakcií. Preto je ϵ veľmi malé, typicky v rozmedzí od 10^{-2} do 10^{-7} . Napriek tomu je reakcia 1.14 rýchla a tiež rýchlo spadá do rovnováhy, v ktorej zostáva aj keď sa hodnota premennej σ mení. Preto uvažujeme túto aproximáciu ako $\epsilon \frac{d\chi}{d\tau} = 0$. Toto tvrdenie je ekvivalentné tomu úvodnému, že $\frac{d[C]}{dt} = 0$.

Potom z tejto aproximácie vyplýva:

$$\chi = \frac{\sigma}{\sigma + \kappa}, \quad (1.15)$$

$$\frac{d\sigma}{d\tau} = -\frac{q\sigma}{\sigma + \kappa}, \quad (1.16)$$

kde $q = \kappa - \alpha = \frac{k_3}{k_1[S_0]}$. Rovnica 1.16 popisuje rýchlosť prírastku substrátu a je pomenovaná zákon Michaelis-Mentenovej (z ang. Michaelis-Menten law). V znení pôvodných premenných tento zákon vyzerá takto:

$$V = \frac{d[P]}{dt} = -\frac{d[S]}{dt} = \frac{k_3[E_0][S]}{[S] + K_m} = \frac{V_{max}[S]}{[S] + K_m}, \quad (1.17)$$

kde $K_m = \frac{k_2 + k_3}{k_1}$. Vzťahy 1.11 a 1.17 sú si už na prvý pohľad veľmi podobné. Jediným rozdielom sú konštanty K_s a K_m . Jedná sa o dva podobné výsledky na základe rôznych predpokladov.

1. Existuje viacero spôsobov, ako systém diferenciálnych rovníc previesť na bezrozmerný. O tom však táto práca nepojednáva. Ďalšie informácie ohľadom tejto problematiky je možné nájsť v [8].

Tak ako zákon o mass action kinetike aj Michaelis-Mentenovej zákon 1.17 nie je platný univerzálne. Je však veľmi užitočný a používaný, pretože koeficient K_m je experimentálne dobre pozorovateľný a teda aj ľahko merateľný narozdiel od individuálnych rýchlostných konštánt jednotlivých chemických substancií v reakcii. [7]

C: Enzýmová spolupráca

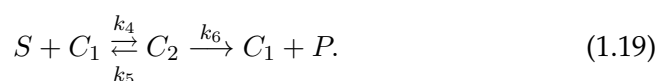
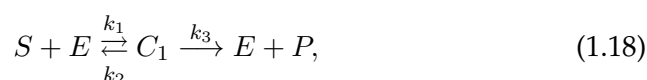
Reakčná rýchlosť mnohých enzýmov nemá klasickú hyperbolickú krivku, tak ako je predpokladané pri použití Michaelis-Mentenovej kinetiky, ale miesto toho má sigmoidálny charakter. To je spôsobené vlastnosťou týchto enzýmov, ktorá im umožňuje viazať na seba viacero substrátov naraz a zároveň tým ovplyvniť obtiažnosť tohto viazania. Táto vlastnosť sa nazýva kooperácia alebo súčinnosť, či spolupráca.

Tieto enzýmy majú viac ako len jedno viazacie miesto (z ang. binding site) a pri naviazaní prvej molekuly dochádza k zmene konformácie vzniknutého komplexu, čo môže ovplyvniť naviazanie ďalšej molekuly pozitívne, ale aj negatívne. Pre každú novú molekulu tento proces pokračuje rekurzívne.

Predpokladajme, že enzým E dokáže viazať až dve molekuly substrátu S , takže sa môže nachádzať v troch rôznych stavoch:

1. voľná molekula enzýmu (E),
2. komplex s jednou naviazanou molekulou substrátu (C_1),
3. komplex s dvoma naviazanými molekulami substrátu (C_2).

Potom samotné reakcie vyzerajú nasledovne:



Použitím zákona o mass action kinetike dostaneme najskôr päť diferenciálnych rovníc a po úprave len tri. Následným uplatnením aproximácie kvázistacionárneho stavu dostaneme:

$$[C_1] = \frac{K_2[E_0][S]}{K_1K_2 + K_2[S] + [S]^2}, \quad (1.20)$$

$$[C_2] = \frac{[E_0][S]^2}{K_1K_2 + K_2[S] + [S]^2}, \quad (1.21)$$

kde $K_1 = \frac{k_2+k_3}{k_1}$, $K_2 = \frac{k_5+k_6}{k_4}$ a $[E_0] = [E] + [C_1] + [C_2]$. Reakčná rýchlosť potom vyzerá takto:

$$V = k_3[C_1] + k_6[C_2] = \frac{(k_3K_2 + k_6[S])[E_0][S]}{K_1K_2 + K_2[S] + [S]^2}. \quad (1.22)$$

Ak budeme teraz pre ukážku uvažovať prípad pozitívnej spolupráce, tak naviazanie prvej molekuly S bude relatívne pomalé, ale naviazanie druhej molekuly S už bude rýchlejšie. Tento jav môžeme vyjadriť ako $k_4 \rightarrow \infty$ a zároveň $k_1 \rightarrow 0$, zatiaľ čo k_1k_4 je konštantná hodnota. V tomto prípade ale naopak platí, že $K_1 \rightarrow \infty$ a $K_2 \rightarrow 0$, zatiaľ čo K_1K_2 je tiež konštantné. Po aplikácii týchto nových obmedzení na vzťah 1.22 dostávame:

$$V = \frac{k_6[E_0][S]^2}{K_m^2 + [S]^2} = \frac{V_{max}[S]^2}{K_m^2 + [S]^2}, \quad (1.23)$$

kde $K_m^2 = K_1K_2$ a $V_{max} = k_6[E_0]$.

Obecne sa dá povedať, že ak enzým dokáže viazať n molekúl substrátu, existuje tiež n rovnovážnych konštánt K_1, \dots, K_n , pre ktoré bude platiť $K_n \rightarrow 0$ a $K_1 \rightarrow \infty$, zatiaľ čo K_1K_n bude stále konštantna a obecná rovnica reakčnej rýchlosti je

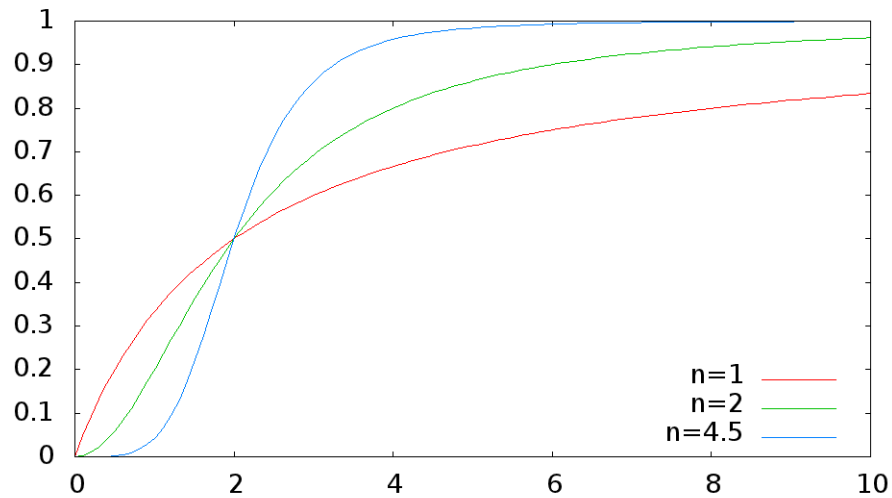
$$V = \frac{V_{max}[S]^n}{K_m^n + [S]^n}, \quad (1.24)$$

kde $K_m^n = \prod_{i=1}^n K_i$. Táto rovnica je známa ako Hillova rovnica alebo Hillova kinetika a konštantna K_m^n vyjadruje koncentráciu, pri ktorej je rýchlosť reakcie v polovičke svojho maxima, tj. $\frac{V_{max}}{2}$. Typický faktor n , vyjadrujúci strmosť reakčnej krivky (obr. 1.2) býva menší ako skutočný počet viazacích miest na enzýme, často to dokonca nie je ani celé číslo. Stojí za zmienku, že v prípade $n = 1$ zodpovedá Hillova kinetika Michaelis-Mentenovej kinetike, preto náš nástroj ponúka iba možnosť zadania Hillovej rovnice (viď kapitolu 2.B). [7]

1.5 Model checking

Model checking alebo overovanie modelov je automatizovaná technika formálnej verifikácie špecifikovaných vlastností konečného stavového systému. Hlavnou výzvou tejto problematiky je úspešne si poradiť s problémom explózie stavového priestoru (z ang. state space explosion).

Proces overovania modelov pozostáva z nasledujúcich podúloh:



Obr. 1.2: Faktor n Hillovej kinetiky pre veľkosti 1, 2 a 4.5 ($K_m = 2$)

Modelovanie Prvou úlohou je prevedenie skúmaného systému do formálneho matematického modelu, ktorý bude akceptovaný vybraným overovacím nástrojom. V niektorých prípadoch je to ľahká úloha, avšak v iných je potreba použiť vhodné abstrakcie za účelom odstránenia irelevantných detailov alebo naopak zvýraznenia istých črtov skúmaného systému.

Špecifikácia Ešte pred samotným overovaním, je nevyhnutné špecifikovať vlastnosti, ktoré má skúmaný systém spĺňať, pretože práve tie budeme ďalej overovať. Špecifikáciou sa myslí použitie nejakého vhodného formalizmu. Typicky napríklad temporálnej logiky, ktorá umožňuje skúmať správanie systému v čase. My budeme používať LTL (viď 1.1).

Overovanie Model checking dokáže overiť, či model vyhovuje danej špecifikácii, ale nedokáže rozhodnúť, či daná špecifikácia pokrýva všetky vlastnosti, ktorým skúmaný systém vyhovuje. Toto je veľmi významný problém formálneho overovania modelov.

Výsledkom overovania modelu je buď tvrdenie áno (v zmysle model spĺňa vlastnosť) alebo nie (model nespĺňa vlastnosť) a v tomto prípade by mal použitý nástroj poskytnúť možnosť trasovania chyby (z ang. error trace). Táto chybová trasa grafom sa obvykle používa ako protipríklad k overovanej vlastnosti. S jej pomocou môžeme lep-

šie pochopiť dôvod a miesto vzniku chyby a upraviť systém podľa toho. [4]

Prvé algoritmy pre overovanie modelov používali ako formalizmus matematického modelu daného systému Kripkeho štruktúru. Je to obdoba nedeterministického konečného automatu (viď kapitolu 1.2), ktorého stavy sú označené výrazmi z množiny 2^{AP} , kde AP sú atomické propozície formule f (viď kapitolu 1.1). Všetky tieto stavy sú akceptujúce.

Neskôr sa používal μ -kalkulus, ale v súčasnosti sú najrozšírenejším formalizmom automaty. Konkrétne budeme pojednávať o použití Büchiho automatu (viď kapitolu 1.2). Tento je veľmi vhodný, pretože sa s jeho pomocou dá vyjadriť rovnako model systému, ako aj po vynaložení určitého výpočtového úsilia, špecifikácia vlastnosti.

Prvou fázou tvorby modelu systému je vytvorenie Kripkeho štruktúry, ktorá sa ale veľmi jednoducho prevedie na automat \mathcal{A} . Špecifikáciu vlastnosti môžeme vyjadriť ako automat \mathcal{S} . Potom sú oba tieto automaty vytvorené nad rovnakou abecedou $\Sigma = 2^{AP}$. Overenie modelu teraz znamená jednoducho zistiť, či platí $\mathcal{L}(\mathcal{A}) \subseteq \mathcal{L}(\mathcal{S})$, kde $\mathcal{L}(\mathcal{A})$ je jazyk zodpovedajúci automatu \mathcal{A} a $\mathcal{L}(\mathcal{S})$ je jazyk zodpovedajúci automatu \mathcal{S} (Jazyk je množina všetkých slov, ktoré je možno vygenerovať príslušným automatom.). To znamená, že každé chovanie modelovaného systému, určeného jazykom $\mathcal{L}(\mathcal{A})$ sa nachádza medzi povolenými chovaniami, určenými jazykom špecifikácie $\mathcal{L}(\mathcal{S})$.

Toto tvrdenie môžeme preformulovať. Nech $\overline{\mathcal{L}(\mathcal{S})}$ je komplement k $\mathcal{L}(\mathcal{S})$, potom pri overovaní modelu dokazujeme, či platí $\mathcal{L}(\mathcal{A}) \cap \overline{\mathcal{L}(\mathcal{S})} = \emptyset$. Ak je tento prienik neprázdny, predstavuje to chovanie, ktoré je protipríkladom k overovanej vlastnosti. Použiť predchádzajúcu formuláciu nám umožňuje vedomosť, že Büchiho automaty sú uzavreté na prienik a doplnok (komplement). Vďaka tomu môžeme urobiť nasledovné:

$$\begin{aligned}\emptyset &= \mathcal{L}(\mathcal{A}) \cap \mathcal{L}(\mathcal{S}') = \mathcal{L}(\mathcal{A} \cap \mathcal{S}') = \mathcal{L}(\mathcal{M}), \\ \text{kde } \mathcal{M} &= \mathcal{A} \cap \mathcal{S}' \text{ a } \mathcal{S}' = \mathcal{S}(\neg\phi), \\ \text{kde } \phi &\text{ je LTL formula a } \neg\phi \text{ jej negácia,}\end{aligned}$$

potom \mathcal{S}' je komplement $\mathcal{S}(\phi)$ a v skutočnosti dokazujeme prázdnosť jazyka $\mathcal{L}(\mathcal{M})$ (Pre úplnosť treba dodať, že $\mathcal{S}(\phi)$ je automat \mathcal{S} zkonštruovaný zo špecifikácie vlastnosti ϕ).

Výhodou použitia automatov pre vyjadrenie modelu skúmaného systému aj špecifikácie vlastnosti je, že stačí skonštruovať automat pre vlastnosť (viď kapitolu 1.5.1) a automat modelu sa konštruuje za behu samotného overovania pomocou algoritmu pre prienik automatov. To znamená,

že v mnohých prípadoch môžeme vyvrátiť splniteľnosť oveľa skôr, než sa vytvorí celý stavový priestor automatu modelu nájdením prvého protipríkladu. Čo značne šetrí čas a priestor. Táto technika sa nazýva overovanie modelu za behu (z *ang.* on-the-fly model checking). [3]

1.5.1 Prevod LTL do BA

Ako sme už predtým niekoľko krát spomenuli, prevod LTL formule na Büchiho automat nie je primitívny algoritmus. V skutočnosti je príliš rozsiahly pre naše potreby zoznamovania sa s problematikou. Z tohto dôvodu ho tu nebudeme rozoberať, avšak uvedieme niekoľko dôležitých informácií, ktoré sa tohto problému týkajú.

Ešte pred začiatkom sa musí samotná LTL formula ϕ previesť do negatívnej normálnej formy (z *ang.* negation normal form). Najskôr sa preformulujú niektoré temporálne operátory:

- $\mathbf{F} \phi \Rightarrow \mathbf{true} \mathbf{U} \phi$
- $\mathbf{G} \phi \Rightarrow \mathbf{false} \mathbf{R} \phi$

a tiež logické operátory, tak aby zostali iba \wedge , \vee a \neg . V poslednom kroku prípravy sú všetky negácie presunuté dovnútra:

- $\neg(\psi \mathbf{U} \phi) \Rightarrow (\neg\psi) \mathbf{R} (\neg\phi)$
- $\neg(\psi \mathbf{R} \phi) \Rightarrow (\neg\psi) \mathbf{U} (\neg\phi)$
- $\neg(\mathbf{X} \phi) \Rightarrow \mathbf{X} (\neg\phi)$

Ďalej pokračuje dlhý algoritmus [5], ktorého výsledkom je Büchiho automat \mathcal{S} . Jeho konštrukcia má exponenciálnu časovú aj priestorovú zložitosť závislú od veľkosti formule ϕ . Avšak v praxi býva takto skonštruovaný automat ovykle menší.

Je dôležité ešte raz pripomenúť, že pri overovaní modelu vlastne nezisťujeme, či model spĺňa vlastnosť, ale naopak zisťujeme, či model nespĺňa opak vlastnosti. Dôvodom k tomu je, že je výpočtovo ľahšie vytvoriť automat z negácie vlastnosti, ako vytvoriť komplement automatu z pôvodnej vlastnosti, ktorý by mohol mať až dvojnásobne exponenciálnu priestorovú zložitosť. [5]

1.5.2 Farebný model checking

Farebný model checking alebo tiež paralelný sa snaží vysporiadať s parametrizáciou formálneho modelu, ktorá ho rozširuje o nový rozmer. Ak sa

budeme na jednotlivé parametre $p_i \in P$, kde P je množina všetkých neznámych parametrov dívať ako na intervaly alebo skôr ako na množiny hodnôt, vďaka diskretizácii. Potom kombinácia evaluácií všetkých neznámych parametrov tvorí parametrický priestor $\mathcal{P} = \prod_{i=1}^n [\min(p_i), \max(p_i)]$, kde $n = |P|$. Tento rozširuje model systému o parametrizované multi-afinné funkcie $f_i(x, \pi_i)$, kde $\pi_i \in \mathcal{P}$ namiesto bežných multi-afinných funkcií $f_i(x)$, kde $x = (x_1, \dots, x_n)$ je vektor premenných a $f = (f_1, \dots, f_n) : \mathbb{R}^n \rightarrow \mathbb{R}^n$ je vektor multi-afinných funkcií (viď kapitolu 2). Hodnota π_i vyjadruje konkrétnu evaluáciu parametrov v celom modeli a označujeme ju ako farba.

Aby sa zabránilo opakovanému vytváraniu automatu pre každú evaluáciu π_i za účelom overenia modelu, bola vytvorená nová pomocná štruktúra pomenovaná parametrizovaná Kripkeho štruktúra (ďalej len PKS). Tá v sebe tradične uchováva celý stavový priestor modelu, ale navyše s novou informáciou, ktorá rozhoduje pod ktorou farbou, resp. konkrétnou evaluáciou parametrov sa dá prejsť z prechodu s do prechodu s' . Každý prechod musí byť uschopnený aspon pod jednou farbou, ale zároveň môže byť aj pod všetkými. Celý stavový priestor je tak zjednotenie všetkých jednofarebných stavových priestorov.

Vďaka tomu prebieha overovanie všetkých parametrizácií modelu naraz. Výsledkom je najväčšia množina parametrizácií, v ktorých model spĺňa danú vlastnosť. [1]

Kapitola 2

Biochemický dynamický vstupný model

A. Multi-afinný ODE model

Vstupným modelom sa u nás myslí model biochemických reakcií, ktorý je v našom poňatí braný ako po častiach multi-afinný systém diferenciálnych rovníc. Ale začnime od počiatku a postupne sa dopracujeme k tomuto výsledku.

Na základe pravidla o mass action kinetike (viď 1.3) je možné modelovať ľubovoľnú biochemickú reakciu alebo dokonca sústavu takýchto reakcií pomocou sústavy nelineárnych obyčajných diferenciálnych rovníc (Ordinary Differential Equations alebo ODEs) [odkaz na ODE ale nie wiki].

Uvažujme multi-afinný systém vo forme $\dot{x} = f(x)$, kde $x = (x_1, \dots, x_n)$ je vektor premenných a $f = (f_1, \dots, f_n) : \mathbb{R}^n \rightarrow \mathbb{R}^n$ je vektor multi-afinných funkcií. Tieto funkcie sú vlastne polynómy, v ktorých je stupeň premenných x_1, \dots, x_n obmedzený na hodnotu 1. Každá premenná x_i , kde $i \in \{1, \dots, n\}$ predstavuje koncentráciu špecifickej chemickej látky a je interpretovaná ako $\mathbb{R}_+ = \{x \in \mathbb{R} \mid x \geq 0\}$. /*Mozno priklad*/

Z dôvodu, že premenné môžeme vyjadriť len ako nezáporné reálne čísla, je možné tiež spojiť stavový priestor nášho matematického systému obmedziť iba na prvý, resp. kladný kvadrant $\mathbb{R}_+^n = \{x \in \mathbb{R}^n \mid x \geq 0\}$.

Ak uvažujeme o premenných ako o nestabilných chemických látkach, ktoré sami od seba degradujú v čase, môžeme s kl'udom obmedziť náš spojitý stavový priestor \mathcal{D} ešte viac. A síce na n -dimenzionálny obdĺžnik $\mathcal{D} = \prod_{i=1}^n [0, \max_i] \subset \mathbb{R}^n$, kde \max_i je horná hranica koncentrácie premennej x_i . [odkaz na hibi2010 a hibi2009]

B. Po častiach multi-afinný ODE model

Multi-afinný systém diferenciálnych rovníc dokáže pokryť skoro celú mass action kinetiku s jedinou výnimkou. A tou sú homodiméry a reakcie s nimi spojené. Dôvodom je predchádzajúce obmedzenie multi-afinných funkcií f_1, \dots, f_n s ohľadom na stupeň premenných x_1, \dots, x_n .

Teoreticky sme schopný popísať akýkoľvek biochemický model pomocou pravidiel tejto kinetiky. V skutočnosti, ak sa pokúsime formulovať tieto pravidlá pre rozsiahly model, zistíme, že s narastajúcou veľkosťou rastie komplexita týchto pravidiel a navyše k úplnosti modelu je potrebné poznať veľké množstvo čo najpresnejšie vyčíslených parametrov. Práve tento druhý problém môže byť v niektorých prípadoch experimentálne neriešiteľný. Či už ide o veľké enzymatické komplexy, alebo (o látky s veľmi krátkou existenciou / o veľmi rýchlo degradujúce látky).

Práve preto sa ponúkajú možnosti aproximácie, ktoré nie len znižujú systém a tým aj dimenzionalitu matematického modelu, ale zjednodušujú aj výpočtovú zložitosť. Takouto možnosťou je aj aproximácia kvázistacionárneho stavu (viď 1.4.B). Napríklad Michaelis-Mentenovej kinetika (viď. 1.4), či obecnjšia Hillova kinetika (viď. 1.4.C) a tiež sigmoidálne prepínače publikované na konferencii CAV (Grosu et al. 2011) [], ktoré náš nástroj ponúka. Takto redukované diferenciálne rovnice majú formu racionálnych polynómov, získaných ako lineárna kombinácia týchto regulačných funkcií, medzi ktoré patria aj Heavisideove alebo schodové funkcie []. V skutočnosti výsledný matematický model nie je multi-afinný, ale na druhú stranu je ho možné aproximovať v zmysle po častiach multi-afinného systému. A to tak, že nahradíme všetky regulačné funkcie sústavou vhodných po častiach lineárnych rampových funkcií. Tieto sú definované nasledovne:

$$r^+_{\text{coor}}(x_i, \theta_i, \theta'_i, y, y') = \begin{cases} y, & \text{pre } x_i < \theta_i, \\ y + (y' - y) \frac{x_i - \theta_i}{\theta'_i - \theta_i}, & \text{pre } \theta_i < x_i < \theta'_i, \\ y', & \text{pre } x_i > \theta'_i. \end{cases};$$

$$r^+(x_i, \theta_i, \theta'_i, a, b) = r^+_{\text{coor}}(x_i, \theta_i, \theta'_i, a\theta_i + b, a\theta'_i + b);$$

kde $i \in \{1, \dots, n\}$,

$$y = x_j, y' = x'_j; j \in \{1, \dots, n\} \wedge j \neq i,$$

$$\theta_i, \theta'_i \in \mathbb{R}^+, \theta_i < \theta'_i \leq \max_i,$$

$$a, b \in \mathbb{R}.$$

Potom klesajúce rampové funkcie sú definované ako kvantitatívny doplnok rastúcich:

$$r^-_{\text{coor}}(x_i, \theta_i, \theta'_i, y, y') = 1 - r^+_{\text{coor}}(x_i, \theta_i, \theta'_i, y, y')$$

$$r^-(x_i, \theta_i, \theta'_i, a, b) = 1 - r^+(x_i, \theta_i, \theta'_i, a, b)$$

Už zmienené regulačné funkcie majú nasledujúce formy:

$$hill^+(x_i, d, \theta_i, a, b) = a + (b - a) \frac{[x_i]^d}{[\theta_i]^d + [x_i]^d};$$

$$hill^-(x_i, d, \theta_i, a, b) = 1 - hill^+(x_i, d, \theta_i, a, b);$$

$$s^+(x_i, e, \theta_i, a, b) = a + (b - a) \frac{1 + \tanh(e(x_i - \theta_i))}{2};$$

$$s^-(x_i, e, \theta_i, a, b) = 1 - s^+(x_i, e, \theta_i, a, b);$$

$$h^+(x_i, \theta_i, a, b) = a, \text{ ak } x_i < \theta_i; b \text{ inak};$$

$$h^-(x_i, \theta_i, a, b) = 1 - h^+(x_i, \theta_i, a, b);$$

kde $hill^+, hill^-$ sú funkcie Hillovej kinetiky,

s^+, s^- sú sigmoidálne prepínače,

h^+, h^- sú Heavisideove (schodové) funkcie,

$$\theta_i \in \mathbb{R}^+, \theta_i \leq \max_i,$$

$$i \in \{1, \dots, n\},$$

$$a, b \in \mathbb{R}_0^+,$$

$$e, d \in \mathbb{R}^+.$$

Špeciálnym prípadom je recipročná hodnota sigmoidálnej funkcie:

$$s^+(x_i, e, \theta_i, a, b)^{-1} = s^-(x_i, e, \theta_i + \frac{\ln(\frac{a}{b})}{2e}, b^{-1}, a^{-1}),$$

ktorú označujeme ako $s^{+inv}(x_i, e, \theta_i, a, b)$. Potom klesajúcu recipročnú funkciu označíme obdobne ako doplnok rastúcej:

$$s^{-inv}(x_i, e, \theta_i, a, b) = 1 - s^+(x_i, e, \theta_i, a, b).$$

Dôkaz možno nájsť v článku *From cardiac cells to genetic regulatory network* na strane 6 [1].

Teraz už môžeme zdefinovať úplný formát nášho po častiach linearizovaného multi-afinného ODE modelu (ďalej len PMA model z *ang.* piece-wise multi-affine ODE model). PMA model \mathcal{M} je daný ako $\dot{x} = f(x)$, kde x je stále vektor premenných (x_1, \dots, x_n) , ale $f = (f_1, \dots, f_n) : \mathbb{R}^n \rightarrow \mathbb{R}^n$ je tentokrát vektor po častiach multi-afinných funkcií. Nevyhnutnou súčasťou modelu \mathcal{M} je množina prahových hodnôt (z *ang.* threshold) $\theta_m^i \in$

\mathbb{R}^+ spĺňajúca $\min_i = \theta_1^i < \theta_2^i < \dots < \theta_{\eta_i}^i = \max_i$, kde $i \in \{1, \dots, n\}$, $m \in \{1, \dots, \eta_i\}$ a platí, že $\eta_i \geq 2$.

Uvažujme Ω ako časť modelu \mathcal{M} tak, že $\Omega = \prod_{i=1}^n \{1, \dots, \eta_i - 1\}$. Funkcia $g : \mathbb{R}^n \rightarrow \mathbb{R}^+$ je vtedy po častiach multi-afinná, ak je multi-afinná na každom n -dimenzionálnom intervale $(\theta_{j_1}^1, \theta_{j_1+1}^1) \times \dots \times (\theta_{j_n}^n, \theta_{j_n+1}^n)$, kde $(j_1, \dots, j_n) \in \Omega$ a zároveň $\forall i, 1 \leq i \leq n, j_i < \max_i$. Potom dostávame n -dimenzionálny PMA model pozostávajúci z funkcií f v nasledujúcom tvare:

$$f_i(x) = \sum_{j \in I^+} \rho_i^j(x) - \sum_{j \in I^-} \rho_i^j(x),$$

kde $i \in \{1, \dots, n\}$, I^+ a I^- sú konečné množiny indexov, pre ktoré platí $I^+ \cap I^- \neq \emptyset$ a ρ_i^j je ľubovoľná hodnota z

$$\rho(x_l) = \begin{cases} c, & c \in \mathbb{R} \\ p, & p \in \langle g, h \rangle; g, h \in \mathbb{R} \\ x_k, & k \in \{1, \dots, n\} \\ r^*(x_k, \theta_m^k, \theta_{m+1}^k, a', b'), & m \in \{1, \dots, \eta_k - 1\}; a', b' \in \mathbb{R} \\ r^*_{\text{coor}}(x_k, \theta_m^k, \theta_{m+1}^k, x_l, x'_l), & l \in \{1, \dots, n\} \wedge l \neq i \\ s^*(x_k, e, \theta_m^k, a, b), & e \in \mathbb{R}^+; a, b \in \mathbb{R}_0^+ \\ s^*_{\text{inv}}(x_k, e, \theta_m^k, a, b), & \\ \text{hill}^*(x_k, d, \theta_m^k, a, b), & d \in \mathbb{R}^+ \\ h^*(x_k, \text{theta}_m^k, a, b) & \end{cases},$$

kde $*$ $\in \{+, -\}$. Príklad PMA modelu možno nájsť na obrázku ... Do modelu je priamo zavedená aj možnosť zadania konštantnej funkcie (v prípade, že $\rho(x_l) = c$), lineárnej funkcie (v prípade $\rho(x_l) = x_k$) a možnosť parametrizácie prostredníctvom hodnoty kinetického parametru (v prípade $\rho(x_l) = x_k$). Ďalšou nevyhnutnou súčasťou modelu sú iníciaľne podmienky, vyjadrujúce počiatočné koncentrácie jednotlivých látok. Tieto nie sú vyjadrené presne, ale namiesto toho sú určené intervalom, ktorého hranice musia byť z množiny $\{\theta_1^i, \dots, \theta_{\eta_i}^i\}$.

Takto vytvorený model je prakticky použiteľný pre ľubovoľný biologický systém [odkaz na HiBi2010].

2.1 Abstrakcia

Kapitola 3

Východiskový stav a podobné nástroje

3.1 BioDiVinE 1.0

3.2 PEPMC

3.3 RoverGene

Kapitola 4

BioDiVinE 1.1

Kapitola 5

Implementácia

Kapitola 6

Použitie programu

Kapitola 7

Case study

Kapitola 8

Záver

Literatúra

- [1] J. Barnat, L. Brim, A. Krejci, A. Streck, M. Vejnar, and T. Vejpustek. On parameter synthesis by parallel model checking. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 9(3):693–705, May–June 2012.
- [2] Edmund M. Clarke, Jr., Orna Grumberg, and Doron A. Peled. *Model Checking*, chapter 3 Temporal Logics, pages 27–33. MIT Press, 1999.
- [3] Edmund M. Clarke, Jr., Orna Grumberg, and Doron A. Peled. *Model Checking*, chapter 9 Model Checking and Automata Theory, pages 121–140. MIT Press, 1999.
- [4] Edmund M. Clarke, Jr., Orna Grumberg, and Doron A. Peled. *Model Checking*, chapter 1.3 The Process of Model Checking, pages 4–4. MIT Press, 1999.
- [5] Edmund M. Clarke, Jr., Orna Grumberg, and Doron A. Peled. *Model Checking*, chapter 9.4 Translating LTL into Automata, pages 132–138. MIT Press, 1999.
- [6] James P. Keener and James Sneyd. *Mathematical Physiology*, volume 8 of *Interdisciplinary Applied Mathematics*, chapter 1.1 The Law of Mass Action, pages 3–4. Springer-Verlag, 1998.
- [7] James P. Keener and James Sneyd. *Mathematical Physiology*, volume 8 of *Interdisciplinary Applied Mathematics*, chapter 1.2 Enzyme kinetics, pages 5–16. Springer-Verlag, 1998.
- [8] James P. Keener and James Sneyd. *Mathematical Physiology*, volume 8 of *Interdisciplinary Applied Mathematics*, chapter 1.4 Appendix: Math Background, pages 24–30. Springer-Verlag, 1998.

Príloha