

Univerzita Karlova v Praze  
Matematicko – fyzikální fakulta

DIPLOMOVÁ PRÁCE

Vojtěch Bednář

Počítačové predikce pohybu buněk

Katedra aplikované matematiky

Vedoucí diplomové práce: Doc. RNDr. Zdeněk Hedrlín, CSc.

Studijní program: Informatika

Děkuji Doc. Zdeňkovi Hedrlínovi za vedení seminářů, které mi byly velkou inspirací nejen pro tuto práci, i za rady a konzultace, které mi ochotně a trpělivě poskytl.

Prohlašuji, že jsem svou diplomovou práci napsal samostatně a výhradně s použitím citovaných pramenů. Souhlasím se zapůjčováním práce.

V Praze dne 22. dubna 2001

Vojtěch Bednář

# Obsah

<b>1 Úvod</b>	<b>7</b>
1.1 Nejen o smyslu této práce.....	7
1.2 Úloha simulačního programu.....	8
1.3 Konkrétní biologická motivace.....	8
<b>2 Modelový svět</b>	<b>10</b>
2.1 Prostředí.....	10
2.1.1 Chemické látky.....	10
2.1.2 Chemické reakce.....	10
2.1.3 Políčko.....	10
2.1.4 Mřížka.....	11
2.2 Pevné objekty.....	11
2.2.1 Statické těleso.....	11
2.2.2 Pohyblivé těleso.....	11
2.2.3 Buňka a její membrána.....	11
2.3 Dynamika prostředí.....	13
2.3.1 Šíření látek.....	13
2.3.2 Chemické reakce v prostředí.....	13
2.3.3 Ubývání látek.....	14
2.3.4 Obtékání.....	14
<b>3 Buňka</b>	<b>20</b>
3.1 Biologická motivace.....	20
3.2 Model buňky.....	20
3.2.1 Tvar a membrána.....	20
3.2.2 Kumulační stav.....	21
3.2.3 Gradient kumulačního stavu.....	21
3.2.4 Zygotický graf.....	22
3.2.5 Pohlcování okolních látek.....	23
3.2.6 Dělení.....	23
3.2.7 Úprava množství receptorů.....	26
3.2.8 Pohyb.....	27
3.2.9 Shrnutí .....	27
<b>4 Difúze</b>	<b>29</b>
4.1 Fyzikální definice difúze.....	29
4.2 Náš model difúze.....	29
<b>5 Chemické reakce</b>	<b>32</b>
5.1 Zavedení chemických reakcí.....	32
5.2 Náš model chemických reakcí.....	33
<b>6 Další směřování práce</b>	<b>36</b>
<b>7 Diferenciace homogenní tkáně</b>	<b>37</b>
7.1 Homogenní tkáň.....	37
7.2 Programový model diferenciace.....	37
7.2.1 Jednorozměrný případ s jedním ostrovem.....	37
7.2.2 Jednorozměrný případ s více ostrovy.....	39

7.2.3 Dvojměrný případ.....	40
<b>8 Model růstu vlásečnic</b>	<b>44</b>
8.1 Autonomní buňka.....	44
8.1.1 Motivační příklad.....	44
8.1.2 Rozšíření pojmu typu buňky.....	44
8.1.3 Modelování autonomní buňky.....	45
8.1.4 Konkrétní příklad autonomní buňky.....	46
8.2 Vlasečnice.....	49
8.2.1 Modelová situace vzniku sítě vlásečnic.....	49
<b>9 Závěr</b>	<b>55</b>
9.1 Tématický závěr.....	55
9.2 Metodický závěr.....	55
9.3 Možná rozšíření.....	56
<b>Dodatek A – Seznam parametrů</b>	<b>57</b>
<b>Dodatek B - Uživatelská dokumentace</b>	<b>60</b>
B.1 Příprava pokusu.....	60
B.1.1 Tvar těles.....	60
B.1.2 Obtékání těles.....	60
B.1.3 Látkový vektor.....	61
B.1.4 Vkládání objektů.....	61
B.1.5 O uživatelském rozhraní.....	62
B.1.6 Látky.....	62
B.1.7 Receptory.....	62
B.1.8 Zygotický graf.....	63
B.1.9 Parametry.....	63
B.1.10 Spuštění pokusu.....	63
B.2 Sledování / modifikace pokusu.....	64
B.2.1 Vizualizace sledování.....	64
B.2.2 Sledování hodnot.....	65
B.3 Řízení simulace.....	66
B.4 Zásobárny.....	67
B.5 Části programu.....	67
B.5.1 Hlavní okno.....	67
B.5.2 Okno Editor látkového vektoru.....	70
B.5.3 Okno Zygotický graf.....	71
B.5.4 Okno Editor těles.....	72
B.5.5 Okno Obtékání tělesa.....	73
B.5.6 Okno Samovolné ubývání látek.....	75
B.6 Jak urychlit běh programu.....	75
B.6.1 Vykreslování situace.....	75
B.6.2 Zobrazování aktualizovaných údajů během simulace.....	76
B.6.3 Start a ukončení programu.....	76
<b>Dodatek C - Programátorské poznámky</b>	<b>77</b>
C.1 Přehled logických částí programu.....	77
C.1.1 SplitTree.....	77

C.1.2 Area.....	77
C.1.3 AreaObject.....	78
C.1.4 FileIO.....	78
C.1.5 FormExecutives.....	78
C.1.6 Forms.....	78
C.1.7 Inspector.....	78
C.1.8 Lv.....	78
C.1.9 Managers.....	79
C.1.10 Rendering.....	79
C.1.11 Repository.....	79
C.1.12 Sim.....	79
C.1.13 Support.....	79
C.1.14 UiExtension.....	79
C.1.15 ZygGraph.....	79
C.2 Popis tříd.....	79
<b>Literatura</b>	<b>80</b>
Odborná literatura.....	80
Diplomové práce.....	80

Název práce: Počítačové predikce pohybu buněk

Autor: Vojtěch Bednář

Katedra (ústav): Katedra aplikované matematiky

Vedoucí diplomové práce: Doc. RNDr. Zdeněk Hedrlín, CSc.

e-mail vedoucího: zdenek@kam.mff.cuni.cz

Abstrakt: Práce se zabývá počítačovým modelováním činnosti buněk s důrazem na simulaci dynamiky vytváření buněčných struktur. Základem našeho modelu je zygotický graf, určující chování buňky v závislosti na okolním látkovém prostředí, který je rozšířen o definici pohybu buňky ve směru nejvyšší koncentrace nejvíce viděné látky. Je modelově zkoumána diferenciace homogenních tkání a růst vlásečnic. Součástí práce je program koncipovaný jako virtuální laboratoř. Zkušenost pramenící z virtuálních pokusů poskytuje uživateli vhodný rámec pro porozumění danému tématu. Práce mimo jiné ukazuje, že množství diferencovaných buněk a potažmo tkání vzniklých z jedné primární je regulovatelné kvantitativní změnou jednoho parametru zygotického grafu. Dále se ukazuje, že dvojrozměrné uspořádání buněk v homogenní tkáni je určitým způsobem přirozenější pro diferenciaci, než jednorozměrné. Nakonec je pozorováno, že pohlcování látek buňkami z prostředí má alespoň v modelu důležitý vliv na správný vývoj sítě vlásečnic.

Klíčová slova: buňka, simulace, zygotický graf, virtuální laboratoř

Title: Computer predictions of cell movement

Author: Vojtěch Bednář

Department: Department of Applied Mathematics

Supervisor: Doc. RNDr. Zdeněk Hedrlín, CSc.

Supervisor's e-mail address: zdenek@kam.mff.cuni.cz

Abstract: This work is focused on computer modelling of cells' behavior. Emphasis is laid on dynamics of cell structures creation. Our model is based on so called zygotic graph, which defines production of ligands by the cell in response to its chemical environment. This graph is widened with cell movement definition in direction of the highest concentration of the most seen substance. There are two topics modelled. Differentiation of homogenous tissues and angiogenesis. Part of the work is a simulation program. Experiences with performing of virtual experiments in it gives to the user a suitable frame of understanding of given topic. We show, that the number of isolated differentiated cells could be regulated by one quantitative parameter. Next, it seems, that a two – dimensional layout of a homogenous tissue could be in particular sense more suitable for differentiation, than in one – dimensional case. The work also allows us to see that consumption of environmental chemical substances by cells could have an important influence on correct micro – vessel network buildup.

Keywords: cell, simulation, virtual laboratory, zygotic graph

# 1 Úvod

## 1.1 Nejen o smyslu této práce

Současná biologie nám dává mnoho informací o živých buňkách. Pokud bychom chtěli buňky studovat od základu, nabízí se začít u chemických látek, kterými jsou tvořeny. Dále bychom mohli studovat jádro, jednotlivé organely, strukturu DNA a funkci různých proteinů. Naše strategie by mohla spočívat v tom, že co nejpřeněji popíšeme jednotlivé části živých buněk, funkce těchto částí, a tím získáme popis, který nám dá celkovou představu. My se pokusíme o něco jiného. Zvolíme si určitou úroveň abstrakce. Vezmeme souhrn některých vlastností a chování buňky, který pro nás bude hrát úlohu axiomů. Volba těchto základních kamenů bude alespoň z části motivována klasickým biologickým poznáním, ale nebudeme lpět na tom, aby z něj musela vycházet.

Tato poslední věc se na první pohled může zdát nerozumná. Uvědomme si však následující věci: Dnešní biologické poznání není úplné a sami biologové jej nepovažují za úplně správné. Pokud začnete navštěvovat úvodní přednášku z buněčné biologie, je dost pravděpodobné, že uslyšíte něco jako " ... polovina z toho, co vám tady říkáme, není pravda, ale nikdo neví, která ... ". Je také dost možné, že některé pozorované jevy jsou důsledkem více základnějších zákonitostí. Pokud bychom striktně požadovali konzistenci našich axiomů s těmito jevy, mohlo by nás to svést falšným směrem. Jako příklad si uveďme gravitační zákon. Pokud považujeme gravitační pole země pro oblast, kde provádíme pokusy, za homogenní, budou v ní podle něj padat všechna tělesa stejně rychle. Pokud však v atmosféře spustíme ze stejné výšky železnou kouli a pírko, dopadne koule dříve. Do hry vstupuje ještě tření, které jsme neuvážili. Kdybychom si představili sami sebe jako badatele v době, kdy o gravitačním zákoně ani o tření nebylo nic známo, a my bychom zkoumali, jak se tělesa přitahují, mohlo by nás lpění na konzistenci našich teorií či hypotéz s oním pozorováním odvést z cesty vedoucí ke správným výsledkům.

Na základě zvolených základních kamenů si vytvoříme modelovou buňku. V modelu buňky nám nepůjde o axomatické zachycení co nejvíce aspektů. Tímto způsobem se pokusíme postihnout spíše málo vlastností, které ale budeme považovat za základní. Modelová buňka pro nás nebude cílem, ale základem, na kterém budeme stavět. Půjde nám předně o zachycení dynamiky vytváření a chování soustav buněk. Budeme konstruovat modelové situace a sledovat, jak se v nich naše buňky chovají. Výsledné chování či buněčné soustavy pro nás budou něčím, co budeme potenciálně moci konfrontovat s reálnými situacemi, v podstatě predikcí. Pokud se predikce shoduje s "realitou", můžeme náš model považovat za relevantní a chápat jeho dynamiku jako osvětlení dané situace, či její (alespoň částečné) vystižení v řeči našeho modelu. Pokud ne, může být srovnání výpovědí o tom, že jsme danou situaci v modelové nesprávně nebo nedostatečně zachytili, nebo může být stimulem pro znovuzvážení volby našich buněčných axiomů, či jejich jiný výběr. Důležité pro nás

<sup>1</sup> Tato poznámka nechce být v žádném případě zaměřena proti biologickému zkoumání, jde nám pouze o odůvodnění našeho přístupu.

bude, že výsledky budou v naší teorii konzistentní. Budeme tedy vytvářet jakýsi myšlenkový ostrov, který se svou povahou bude blížit způsobu uvažování, na který jsme zvyklí v matematice, nebo spíše v matematické informatice. Pokud by naše výsledky dávaly ve větší míře uspokojivá vysvětlení pro biologické situace, můžeme budování onoho myšlenkového ostrova chápat jako pokus nebo krůček k přenosu "matematického myšlení" do popisu biologických témat, do biologie, a formulování v něm základních zákonitostí. Zmiňme zde ještě, že důležitou roli pro nás hraje simulační program. Jednak pro samotný výpočet chování modelových buněk, jednak pro prezentaci (a interpretaci) tohoto výpočtu a jeho výsledků a jednak jako podpůrný nástroj pro vytváření představ, pomocí kterých o modelových (či "reálných") jevech přemýšlíme.

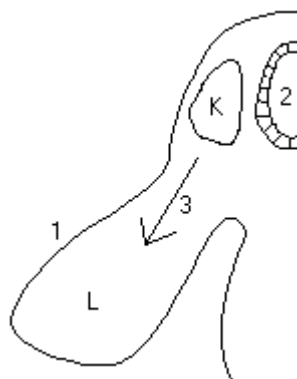
## 1.2 Úloha simulačního programu

Základním přístupem k modelování buněk pro nás bude tzv. lokální zákon. Jako metaforu si můžeme představit diferenciální rovnice, které můžeme chápat jako vyjádření vztahu mezi "lokálním" a "globálním". V našem případě definujeme chování jedné buňky v "malém" časovém okamžiku a ptáme se, jak bude vypadat soustava buněk v průběhu dlouhého času. Vzhledem k charakteru modelu (který bude popsán později) se jeví jako schůdné řešit úlohy v něm zadané pomocí diskrétní simulace. Na druhou stranu je dobře možné (a přínosné) o některých velmi jednoduchých situacích činit pozorování. Máme tedy dva druhy výsledků. Jednak "virtuálně experimentální", které nám vygeneroval program, přičemž obecnější zdůvodnění, proč simulace probíhala tak či onak, nám může být více nebo méně jasné, a jednak ty, které jsme přímo nahlédli. Program nám tedy slouží jednak jako pomoc pro činění (nahlížení) pozorování, dále pro nás může být představou o dané buňce (či soustavě buněk) zkušenost, kterou jsme získali sledování průběhu simulace případně její modifikací, obecně virtuálními experimenty.

## 1.3 Konkrétní biologická motivace

Konkrétní biologickou motivací pro nás bude migrace určitých buněk v embryu kuřete do končetinového pupene. Je pozorováno, že v jisté fázi vývoje se v končetinovém pupeni embrya objeví látka (řikejme jí  $L$ ), která zasáhne až do hlavové oblasti, kde je poblíž neurální trubice "ostrov" tkáň (označme ji  $K$ ), jejíž buňky drží pevně pohromadě. Pokud tyto buňky zjistí přítomnost látky  $L$ , začnou se okraje tkáně rozvolňovat a uvolnivší se buňky putují do končetinového pupene ve směru největšího gradientu látky  $L$ . Některé tam doputují, jiné ne, vytváří se pruh putujících buněk. Během putování se mohou některé buňky rozdělit. Tkáň, kterou buňky putují, je řídká a jsou v ní mezibuněčné prostory. Z namigrovaných buněk se později vytvářejí svaly.





*Obrázek 1: Schéma migrace buněk do končetinového pupene*

*1 Končetinový pupen*

*2 Neurální trubice*

*3 Směr putování buněk*

Je známo, který gen je zodpovědný za produkci látky L. Dále se ví, jaký gen kóduje receptor, kterým buňky tkáně K látku L vidí. Je experimentálně ověřeno, že pokud se alespoň jeden z obou genů vypne, k migraci nedochází.

V našem modelu (programu) by mělo být možno vytvořit situaci odpovídající výše uvedenému popisu a zkoumat pohyb buněk.

## 2 Modelový svět

Náš modelový svět můžeme rozdělit na dvě základní části: prostředí a pevné objekty, které se v prostředí mohou nacházet. Prostředí nám udává modelový prostor a jeho uspořádání, zejména základní rámec pro reprezentaci polohy a tvaru buněk, dále pak chemické látky, reakce mezi nimi a jejich koncentrace v prostoru. Pevné objekty mají svůj tvar a v každém čase nějakou pozici v prostředí. Jde především o modelové buňky.

### 2.1 Prostředí

#### 2.1.1 Chemické látky

Co se týče modelových chemických látek, půjde nám pouze o to, abychom je uměli vzájemně rozlišit a měli mezi nimi definované reakce. Označme si  $\mathbf{A}$ ,  $\mathbf{V}$  dvě množiny látek (konečné, neprázdné),  $|\mathbf{A}| = |\mathbf{V}|$ . Necht'  $\mathbf{A}$  a  $\mathbf{V}$  mají prázdný průnik, jejich sjednocení si označme  $\mathbf{L}_v$ . Jednotlivé látky z  $\mathbf{L}_v$  si očíslovme indexy  $1, 2, \dots, n$ ,  $|\mathbf{L}_v| = n$ . Pro každou látku  $\mathbf{l} \in \mathbf{L}_v$  necht' existuje právě jedna látka  $\mathbf{l}'$  taková, že  $\mathbf{l}$  a  $\mathbf{l}'$  spolu reagují,  $\mathbf{l}' \in \mathbf{L}_v$ ,  $\mathbf{l}' \neq \mathbf{l}$ . Je-li  $\mathbf{l} \in \mathbf{A}$  pak  $\mathbf{l}' \in \mathbf{V}$ , jinak je  $\mathbf{l}' \in \mathbf{A}$ . Označme si  $\mathbf{x}$  produkt reakce mezi  $\mathbf{l}$  a  $\mathbf{l}'$ . Pro něj bude platit, že  $\mathbf{x} \notin \mathbf{L}_v$ . Bud'  $\mathbf{X}$  množina všech produktů reakcí látek z  $\mathbf{L}_v$ . Bud'  $\mathbf{y} \in \mathbf{X}$ . Budeme uvažovat, že  $\mathbf{y}$  nereaguje s žádnou látkou z  $\mathbf{X}$ . O látkách  $\mathbf{l}$  a  $\mathbf{l}'$  budeme říkat, že spolu tvoří vax, a také že jsou navzájem komplementární.

Předmětem našeho zájmu budou téměř výhradně látky z  $\mathbf{L}_v$ . Každá z těchto látek má v programu jednoznačné jméno a nějakou barvu. Množině  $\mathbf{L}_v$  v něm říkáme *látkový vektor*. Mohutnost množiny  $\mathbf{A}$  (respektive  $\mathbf{V}$ ) je parametrem modelové situace, který značíme *Num vaxes*.

#### 2.1.2 Chemické reakce

Chemické reakce budou detailně popsány ve zvláštní kapitole. V programu se míra reakce látek ve vaxech udává parametrem *Afinita* pro každý vax. *Afinita* odpovídá konstantě  $K$  z kapitoly o chemických reakcích.

#### 2.1.3 Políčko

Políčko je nejmenší samostatná část prostředí. Reprezentuje jednotku prostoru. Koncentrace libovolné látky je v políčku konstantní. Políčko ví, jaké látky a v jakých koncentracích se v něm nacházejí (vektor koncentrací), dále nese informaci, zda je obsazeno nějakým pevným objektem, a kterým; políčko může být obsazeno

nejvýše jedním pevným objektem<sup>2</sup>. Říkejme, že políčko je obsazené právě když je obsazeno nějakým pevným objektem. Políčko si může držet informaci o koncentracích látek i po dobu co je obsazené, ale během této doby to nemá žádný praktický význam<sup>3</sup>.

### 2.1.4 Mřížka

Mřížku chápeme obecně jako množinu políček, přičemž je definováno, která políčka spolu sousedí. Naše mřížka má tvar obdélníkové "matice" čtvercových políček stejné velikosti. Délku hrany čtverce políčka považujeme za rovnou jedné. Pokud se políčko nenachází v prvním nebo posledním řádku, respektive sloupci, má čtyři sousedy: Levého, pravého, horního a dolního. Jinak mu odpovídající soused či sousedé chybí. Každé políčko má v mřížce své souřadnice. Levé horní políčko má souřadnice (0, 0). Má-li mřížka m řádků a n sloupců, pak pravé horní políčko má souřadnice (n-1, 0), levé dolní (0, m-1), a pravé dolní (n-1, m-1).

## 2.2 Pevné objekty

Pevný objekt reprezentuje těleso, které se nachází někde v prostředí. Zabírá neprázdnou množinu políček, která je dána jeho tvarem<sup>4</sup> a pozicí (souřadnicemi) v mřížce. Pevné objekty mohou být tři druhů: Statické těleso, pohyblivé těleso a buňka.

### 2.2.1 Statické těleso

Statické těleso nemůže v čase měnit svou pozici.

### 2.2.2 Pohyblivé těleso

Pohyblivé těleso má definovanou rychlost, se kterou se pohybuje. Pokud narazí na překážku, uplatní se v jeho pohybu jen ta složka rychlosti, v jejímž směru překážka není. Rychlost značí, o kolik políček v každém směru se těleso pohne v jednom časovém okamžiku (kroku simulace programu).

### 2.2.3 Buňka a její membrána

Buňka modeluje biologickou buňku. Je detailně popsána ve zvláštní kapitole, zde se zaměříme na model membrány z hlediska prostředí a na transport látek z a do buňky.

Biologické buňky jsou obklopeny membránou. Ta je pro některé látky neprostupná. Buňka však obecně umí ve svém okolí takovou neprostupující látku rozpoznat, nebo ji absorbovat. Děje se tak pomocí speciálních molekul, které jsou v

---

2 Výjimku tvoří interní simulace dělení, která odpovídá představě, že z pozice původní buňky se "rozestoupí" dvě dceřinné. To se však může dít jen v rámci jednoho kroku simulace a na jeho konci je zase "vše v pořádku".

3 Na tyto údaje o koncentraci se však vztahuje ubývání látek, které je popsáno níže.

4 Tvar, jakož i vztah mezi souřadnicemi a množinou obsazených políček, jsou definovány pro buňku v odstavci Model buňky kapitoly Buňka. Pro ostatní druhy pevných objektů chápeme tvar i souřadnice stejně.

membráně ukotveny, říkáme jim souhrnně receptory. Část receptoru je vně buňky, část uvnitř. Na vnější část se může navázat nějaká molekula (nebo molekuly) v okolí. Receptor změní svůj tvar. To například způsobí změnu chemických vlastností vnitřní části receptoru. Buňka je tak informována o přítomnosti molekuly, respektive látky, v okolním prostředí. Některé receptory mohou vypadat jako kanálky, které se umějí otvírat, zavírat, nebo aktivně transportovat molekuly skrz membránu. Na určitý druh receptorů se mohou vázat jen určité látky. Má-li buňka nějaké receptory rozptýleny po celém povrchu je (potenciálně) schopna pomocí nich vnímat koncentrační gradient. Může také rozpoznat, že se jí dotýká nějaká jiná buňka, respektive buňka, jejíž receptory se mohou vázat na receptory naší buňky.

My budeme modelovat chytání látek na receptory pomocí vaxů. Na membráně modelové buňky může být ukotvena libovolná látka z  $L_v$ . Bude nám tedy jedno, jestli je nějaká látka receptor, (nebo spíše jeho část vně buňky,) nebo jestli se pohybuje volně v prostoru. Budeme ji vnímat jako jednu látku. Na modelový receptor  $r$  se tedy podle definice  $L_v$  může vázat právě jedna látka  $m \in L_v$ , přičemž  $r \neq m$ .

Zaměříme se nyní na modelovou membránu. Budeme ji vnímat jako množinu "menších" úseků, říkáme jim membránové elementy. Membránový element buňky se nachází všude tam, kde políčko jí zabrané sousedí s políčkem, které tato buňka nezabírá, nebo s okrajem mřížky. V membránovém elementu jsou receptory každého druhu rovnoměrně rozloženy. Pro každý druh receptoru, tedy látku z  $L_v$ , si pamatujeme její množství v membránovém elementu. Může vyvstat otázka, zda spolu budou reagovat látky přítomné jako receptory na jednom membránovém elementu, které jsou spolu ve vaxu. Za těchto podmínek budeme uvažovat, že reakce neprobíhá. Co se týče množství receptorů, je v každém membránovém elementu jedné buňky v každém okamžiku stejné. Membránu buňky považujeme tedy za homogenní.

V programu se pro větší názornost, pokud není řečeno jinak, látky na membráně označují jménem a barvou komplementární látky, tedy látky, která se na receptory váže.

Modelová buňka může skrze receptory na membránových elementech látky v okolí nejen rozpoznávat, ale také absorbovat. (Ovšem jen ty, které nejsou ukotveny na membráně jiné buňky.) Nějakou část množství látek, které se nachytaly na receptory, může buňka pohltnout. To je udáno parametrem *Pohlčení nachytaných látek*. Ten nabývá hodnot mezi 0 a 1 a udává podíl absorbovaného množství látky vůči nachytanému množství látky. Parametr *Pohlčení nachytaných látek* je pro všechny látky stejný.

Buňka může také vylévat látky do prostředí. Pokud chce vylít nějaký vektor množství látek do svého okolí, rozdělí ho rovnoměrně mezi jednotlivé membránové elementy, skrze které pak dojde k vlastnímu vylití. Pokud nasedá membránový element na nějaké políčko a toto není obsazeno žádným pevným objektem, vyleje do něj svoje látky. V opačném případě se nic nestane.

## 2.3 Dynamika prostředí

### 2.3.1 Šíření látek

Látky se mezi jednotlivými políčky šíří difúzí<sup>5</sup>. Ta probíhá mezi každými dvěma sousedními políčky a to jen tehdy, pokud žádné z nich není obsazeno pevným objektem. Výjimku tvoří pouze krajní políčka mřížky v případě, že je parametr *Pevné okraje situace* nastaven na *false*. Potom se vzhledem k difúzi předpokládá, že krajní políčka mají všechny sousedy, jako vnitřní políčka, přičemž fiktivní sousedé nejsou obsazeni pevným objektem a koncentrace všech látek v nich je v každém okamžiku nulová. Model difúze a její výpočet pro jednu látku je popsán v kapitole Difúze. Rychlost difúze se udává parametrem *Rychlost difúze*. Ten může nabývat hodnoty mezi 0 a 1. Z něj se spočítá parametr  $D$  z výše zmíněné kapitoly:  $D = \text{Rychlost difúze} * 0,2$ . 0,2 je převrácená hodnota největšího možného počtu sousedů políčka zvýšeného o 1. Rychlost probíhající difúze jedné látky mezi dvěma políčky závisí pouze na rozdílu koncentrací té látky v oněch políčkách a na parametru *rychlost difúze* (resp.  $D$ ). Látky se tedy v této věci vzájemně neovlivňují.

### 2.3.2 Chemické reakce v prostředí

Reakce probíhají mezi látkami v jednom neobsazeném políčku, dále mezi látkami v neobsazeném políčku a receptory membránových elementů, které na políčko nasedají, a nakonec mezi receptory dvou vzájemně nasedajících membránových elementů. O reakcích mezi látkami v jednom políčku uvažujeme, že vzhledem k daným koncentracím reaktantů, produktu a *Afinitě* proběhnou úplně v jednom časovém okamžiku (respektive v jednom kroku simulace programu), tedy že bude platit vztah (5.11) z kapitoly o chemických reakcích. Chemické reakce v políčku obklopeném membránovými elementy se modelují následujícím způsobem: Pokud nás zajímají koncentrace látek po proběhnutí reakce pouze v políčku, uvažujeme, jako by receptory na membránových elementech do reakce vůbec nevstupovaly. Pokud nás zajímá, kolik kterých látek se nachytalo receptory:

- Nejprve se spočte reakce mezi látkami z množiny **A** v políčku a na receptorech membránových elementů, a z množiny **V** v políčku. (Jako kdyby byly všechny jmenované látky v políčku.)
- Na základě znalosti stavu před reakcí (koncentrací, respektive množství látek z **A** v políčku a množství látek z **A** jako receptorů na jednotlivých membránových elementech) a dále na základě znalosti množství produktu reakce se určí, kolik které látky z **V** je navázáno na receptorech jednotlivých membránových elementů. (Celkové množství produktu se rozdělí v odpovídajících poměrech.)

---

<sup>5</sup> Látky se také mohou pohybovat mezi políčky při obtékání pohybujících se těles, to zmíníme později.

- Pak se výpočet analogicky opakuje, přičemž nyní bereme v úvahu látky z množiny  $V$  v políčku a na receptorech a látky z množiny  $A$  v políčku. Přitom neuvažujeme předchozí reakci, vstupní koncentrace všech látek i receptorů pro výpočet obou reakcí jsou stejné.

Reakci mezi dvěma nasedajícími membránovými elementy modelujeme jako dvě reakce "jakoby v políčku" - látky z  $A$  v prvním elementu s látkami z  $V$  ve druhém elementu a látky z  $V$  v prvním elementu s látkami z  $A$  ve druhém.

### 2.3.3 Ubývání látek

U chemických látek, které nejsou ve formě receptorů na nějakém membránovém elementu, budeme předpokládat jejich samovolný úbytek v čase. (Motivací může být například samovolný chemický rozpad, nebo obecně degradace pomocí látek a reakcí, které již nebudeme modelovat.) Ubývání budeme definovat pro každou látku  $I \in L_v \cup X$ . Rychlost ubývání pro jednotlivé látky udává parametr *Ubývání* – Vektor dimenze  $3|A|$ . Jeho složky mohou nabývat hodnot z intervalu  $(0, 1)$ . Bud'  $C_j^0$  koncentrace  $j$  – té látky z  $L_v \cup X$  v čase  $t_0$  v políčku  $P$ ,  $C_j^1$  koncentrace této látky v  $P$  v čase  $t_1 = t_0 + 1$ . Pokud nebudeme uvažovat jiné vlivy na změnu koncentrace této látky v  $P$  než ubývání, potom

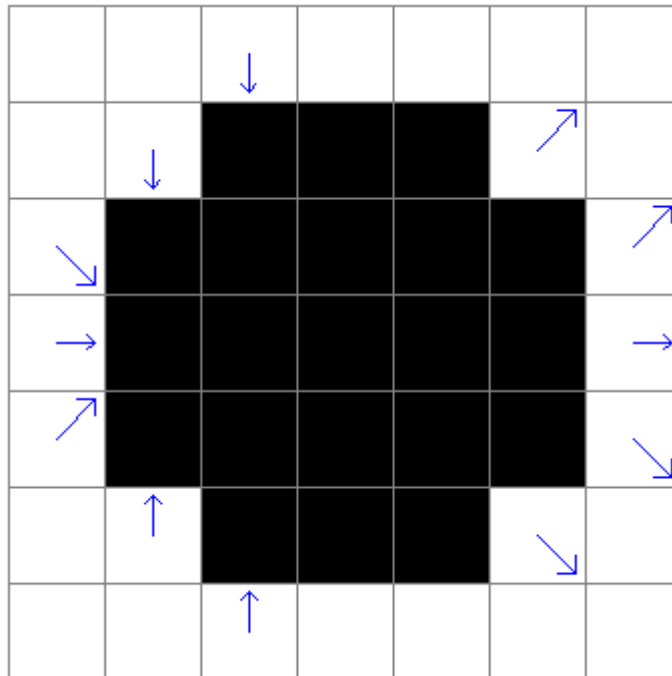
$$C_j^1 = C_j^0 \cdot (1 - \text{Ubývání}_j) . \quad (2.1)$$

### 2.3.4 Obtékání

Pokud se nějaký pevný objekt v prostředí pohybuje, vyvstává otázka, jak to ovlivní koncentrace jednotlivých látek v neobsazených políčkách. Kdybychom chtěli na tuto otázku hledat přesnou odpověď, dostali bychom se k proudění kapalin. (Předpokládáme, že buňky, které modelujeme, se nacházejí v kapalném nebo gelovém prostředí). Fyzikálně věrné modelování by bylo nad rámec této práce. Obtékání pevných objektů, respektive změny koncentrací látek v neobsazených políčkách díky pohybu objektu, budeme simulovat jen přibližně. Náš model se nebude opírat o hlubší teoretické podklady, ale bude se spíše odvíjet od intuitivních představ. Budeme také chtít, aby nebyl příliš výpočetně náročný, aby moc nespomaloval celkovou simulaci.

Nejprve zvolíme okolí objektu, na kterém budeme proudění uvažovat. To bude mít tvar obdélníku, přičemž dané pevné těleso musí být unvitř. Pak nadefinujeme, kterým směrem a s jakou intezitou se pohne obsah jednotlivých neobsazených políček v okolí, pokud se těleso přemístí o jedno políčko doprava, doleva, nahoru či dolů. Říkejme takovému přesunu tělesa jednotkový pohyb. V každém neobsazeném políčku okolí (před jednotkovým pohybem pevného objektu) budeme mít pro každý směr jednotkového pohybu dvourozměrný vektor, který nám udává tok látek (ve smyslu našeho modelu) z tohoto políčka do jeho sousedů.

Označme si nyní  $\mathbf{M}_p$  množinu políček našeho okolí. Pro  $\mathbf{P} \in \mathbf{M}_p$  si příslušný vektor označíme  $\mathbf{F}_{0p}$ . Vektory pro jednotlivá políčka jsou parametry modelu, v programu jsou definovány uživatelem.



Obrázek 2: Příklad obtékání pevného objektu pro jeden směr jeho jednotkového pohybu

Pro vlastní výpočet obtékání budeme požadovat, aby absolutní hodnoty obou složek  $\mathbf{F}_{0p}$  pro všechna  $\mathbf{P} \in \mathbf{M}_p$  nebyly větší než jedna. Pokud tento požadavek není splněn, budeme obtékání počítat postupně v  $n$  krocích. (V programu se všechny vykonávají v rámci jednoho kroku simulace.) V jednom kroku použijeme místo  $\mathbf{F}_{0p}$  vektor  $\mathbf{F}_p$ , přičemž  $\mathbf{F}_p = \mathbf{F}_{0p} / n$ ,  $n$  budeme volit takové, aby požadavek již splněn byl.

Co se týče okolí, bude mít pevnou velikost nezávisle na pozici objektu v mřížce. Může se tedy stát, že se jeho část octne mimo mřížku, což budeme potřebovat ošetřit. Při definici obtékání uvažujeme, jako by se v okolí nenacházel jiný pevný objekt. Pokud se tam bude nacházet, budeme chtít, aby do takto obsazených políček nic nepřitékalo, ani z nich nic nevytékalo. Dále se může stát, že z políčka na okraji okolí vytéká něco z okolí ven, čímž se mohou objevit políčka, která nejsou v okolí a měla by být ve výpočtu zahrnuta. Ve světle těchto skutečností použijeme pro výpočet obtékání rozšířené okolí, které si zavedeme takto: Bude mít výšku a šířku o dvě políčka větší, než původní, které se bude nacházet uprostřed. Políčka, která jsou v novém okolí a nejsou ve starém, budeme nazývat přidaný pás. Vektory všech políček přidaného pásu budou nulové. Pro každé políčko si budeme navíc pamatovat, zda do něho mohou přitékat látky z jiných políček (řekneme, že je přijímající), a zda z něj látky mohou do okolních políček vytékat (řekneme, že je emitující).

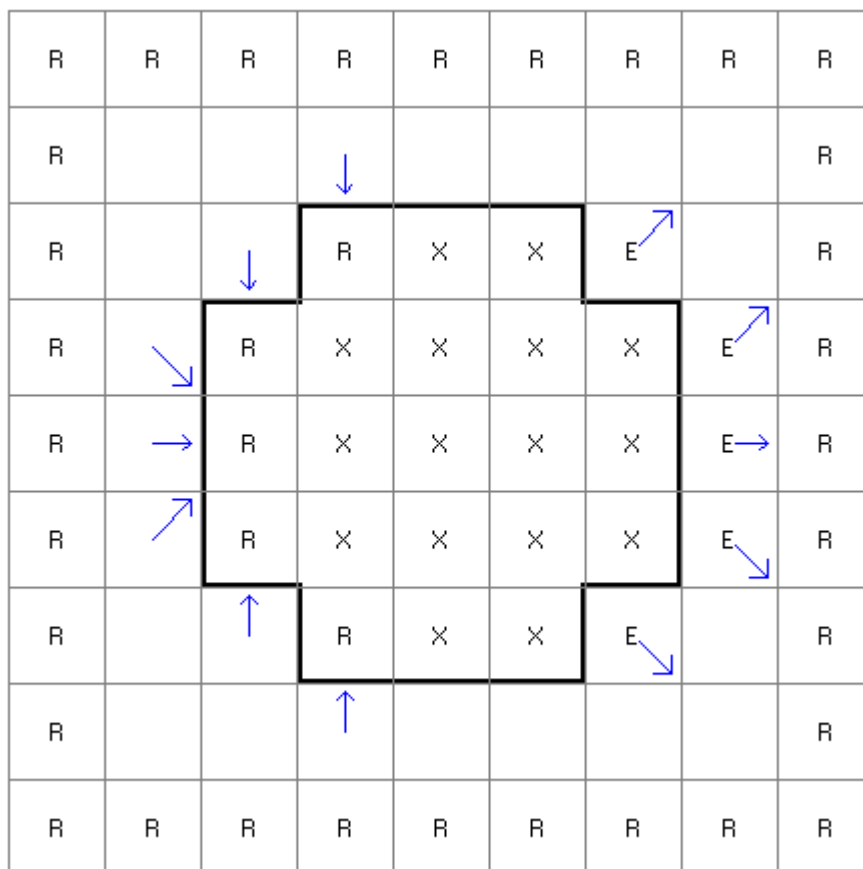
Jako emitující označíme ta políčka, která:

- Před proběhnutím daného jednotkového pohybu nejsou obsazena.
- Nacházejí se v mřížce.
- Nejsou v přidaném pásu.

Jako přijímající políčka označíme ta, pro která platí:

- Po proběhnutím daného jednotkového pohybu nejsou obsazena.
- Nacházejí se v mřížce.





Obrázek 3: Rozšířené okolí obtékání z obrázku 1. Předpokládáme, že se objekt pohne jednotkově doprava a že okolí je celé v mřížce.

*R: Přijímající políčko*

*E: Emitující políčko*

*X: Políčko není ani přijímající ani emitující*

*Jinak: Políčko je přijímající i emitující*

Nechť je  $P$  nějaké políčko z rozšířeného okolí. Označme  $C_0(P)$  vektor koncentrací látek v  $P$  před proběhnutím jednoho kroku výpočtu obtékání,  $C_1(P)$  po proběhnutí tohoto kroku. Sousedění políček budeme nyní chápat pouze pro políčka v okolí, a to osmisměrově (tedy i diagonální sousedy). Množinu emitujících sousedů  $P$  si označme  $Em(P)$ , množinu přijímajících sousedů  $Pr(P)$ . Výsledné koncentrace se spočtou jako:

$$C_1(P) = \sum_{S \in Em(P)} w_S C_0(S) + w_P C_0(P) , \quad (2.2)$$

přičemž

$$w_S = \frac{Flux(S, P)}{\sum_{M \in Em(P)} Flux(M, P) + Flux(P, P)} , \quad (2.3)$$

$$w_P = \frac{Flux(P, P)}{\sum_{M \in Em(P)} Flux(M, P) + Flux(P, P)} . \quad (2.4)$$

Pro políčko Q právě uvolněné díky jednotkovému pohybu je  $Flux(Q, Q) = 0$  . Jinak

$$Flux(X, Y) = u_Y r_X , \quad (2.5)$$

přičemž

$$u_Y = \frac{HardFlux(X, Y)}{\sum_{M' \in Pr(X)} HardFlux(X, M')} , \quad (2.6)$$

$$r_X = 1 - HardFlux(X, X) . \quad (2.7)$$

Pokud by byl jmenovatel některého z výrazů (2.3), (2.4) a (2.6) roven nule, pak položíme levou stranu příslušného výrazu také rovnou nule. Nakonec

$$HardFlux(A, B) = AreaSize(Intersect(Rect(A) + F_A, Rect(B))) . \quad (2.8)$$

$Rect(Z)$  značí čtverec zabraný políčkem  $Z$ . Je – li  $R$  čtverec a  $V$  dvojrozměrný vektor, potom zápis  $R + V$  znamená čtverec  $R$  posunutý o  $V$ .  $Intersect(O_1, O_2)$  značí průnik obdélníků  $O_1$  a  $O_2$ ,  $AreaSize(O)$  velikost plochy obdélníku  $O$ .

Vektory  $F_0$  respektive  $F$  pro jednotlivá políčka slouží tedy jako podklad pro výpočet prvního přiblížení modelového toku mezi dvěma políčky, který značíme  $HardFlux$ . Vidíme, že jeho hodnota se může pohybovat mezi 0 a 1. Z (2.8) je také patrné, proč jsme požadovali omezené velikosti složek vektorů  $F_0$ . Druhé přiblížení modelového toku,  $Flux$ , již bere v úvahu, že né všechna políčka mohou přijímat látky z okolních políček. Do přijímajících sousedů políčka zde rozdělíme vše, co z něj mělo vytéct podle  $HardFlux$  (2.5), (2.6), (2.7). Do políčka, které bylo právě uvolněno pohnuvším se objektem, mohou látky jen přitéct, žádné tam nemohou "zůstat z

minulosti". Nové koncentrace látek v políčku se nakonec spočtou jako vážený průměr původních koncentrací v něm a jeho sousedech. Váhy určíme z hodnot *Flux* z jeho sousedů do něj a z jeho vlastního příspěvku (2.2), (2.3), (2.4).

# 3 Buňka

## 3.1 Biologická motivace

Stojíme před úkolem vytvořit formální (respektive počítačový) model buňky. Jistě se nebudeme pokoušet do něj zahrnout všechny možné poznatky, které by nám nabízela současná biologie. Budeme chtít, aby byl dostatečně obecný, aby nebyl jen pro některé určité typy buněk (například buňky té či oné tkáně). Půjde nám o zachycení základních zákonitostí ohledně interakce buňky se svým okolím, a to předně pomocí chemických látek. Na buňce si především budeme všimnout její membrány a jádra, respektive genetického kódu. Na membráně pro nás bude důležité, že skrze ní nemohou volně prostupovat některé látky (ty osatní nebudeme pro náš model uvažovat), dále že na ní buňka může umísťovat receptory, na které se mohou vázat látky vně buňky a pomocí kterých je rovněž možné tyto látky absorbovat. Také pro nás bude důležité, že buňka má mechanismy, jak transportovat membránou neprostupné látky ven do svého okolí (například pomocí vesikul). Co se týče vnitřku buňky, bude pro nás důležité, že buňka umí rozpoznat, že se na její receptory něco navázalo a na základě toho změnit množinu exprimovaných genů, obecně modifikovat množinu receptorů na své membráně a měnit množinu látek, které se budou vylévat do okolního prostředí. Dále si všimneme takových buněk, které pomocí svých receptorů umějí detekovat koncentrační gradient a na základě této informace se v jeho směru (nebo v opačném) pohybovat. Ohledně genetického kódu vezmeme v úvahu takové buňky, u kterých se po dobu života buňky (přesněji času mezi dvěma děleními, nebo posledním dělením a smrtí) tento nemění. U dceřinných buněk vzniklých dělením jedné buňky budeme předpokládat stejný genetický kód jako u mateřské. Jiné způsoby dělení nebudeme uvažovat.

## 3.2 Model buňky

### 3.2.1 Tvar a membrána

O buňce a její membráně jsme se již zmínili v odstavci Pevné objekty kapitoly Modelový svět. Zopakujme, že buňka je pevné těleso. Její tvar budeme definovat jako množinu dvojrozměrných vektorů. Po složkách těchto vektorů požadujeme, aby byly nezáporné a celočíselné. Označme si tuto množinu  $\mathbf{Tv}$ . V každém časovém okamžiku se buňka nachází na nějaké pozici v mřížce (dvojrozměrný vektor s nezáp. celými složkami), označme ji  $\mathbf{Pos}$ . Zabírá v něm množinu políček  $\mathbf{Zab}$ ,  $\mathbf{Zab} = \{ P \mid \text{Coords}(P) = Z', Z' = Z + \mathbf{Pos}, Z \in \mathbf{Tv} \}$ , přičemž  $\text{Coords}(P)$  značí souřadnice políčka  $P$  v mřížce. Množina  $\mathbf{Tv}$  je pro nás parametrem buňky, v programu je nutné tvar zadat. Různé buňky v modelové situaci mohou mít obecně různé tvary. Pozice  $\mathbf{Pos}$  se určí na základě reálné pozice buňky  $\mathbf{RPos}$  (dvojrozměrný vektor), jako její dolní celá část. Hodnota  $\mathbf{RPos}$  je pro každou buňku počátečním parametrem modelové situace. V čase (v programu v jednotlivých krocích simulace) se mění na základě rychlosti buňky (Ozn.  $\mathbf{Rych}$ ), o jejímž výpočtu se zmíníme později.

Buňka je dále obklopena membránou **Membrane**, množinou membránových elementů. Membránový element **Me** definujeme jako trojici  $\langle T, V_p, Recept \rangle$ , přičemž:

- $V_p \in \{\text{vlevo, vpravo, nahoře, dole}\}$ .
- $T \in \mathbf{Tv}$  a platí alespoň jedna z následujících podmínek:
  - $T + (1, 0) \notin \mathbf{Tv}$ , (potom smí být  $V_p = \text{vpravo}$ , jinak ne),
  - $T + (-1, 0) \notin \mathbf{Tv}$ , (potom smí být  $V_p = \text{vlevo}$ , jinak ne),
  - $T + (0, 1) \notin \mathbf{Tv}$ , (potom smí být  $V_p = \text{dole}$ , jinak ne),
  - $T + (0, -1) \notin \mathbf{Tv}$ , (potom smí být  $V_p = \text{nahoře}$ , jinak ne).
- *Recept* je vektor množství receptorů pro jednotlivé látky z **Lv**.

O membránových elementech uvažujeme, že nasedají na okrajová políčka buňky na daných stranách (všude, kde mohou existovat) a účastní se chemických reakcí v prostředí, jak je popsáno v již zmíněné kapitole. Množství receptorů na jednotlivé látky ve všech membránových elementech buňky jsou stejná.

### 3.2.2 Kumulační stav

Kumulačním stavem budeme chtít zachytit vnitřní stav modelové buňky, který určuje její chování v daném časovém okamžiku. Bud'  $\mathbf{Ks} \in \mathbf{Lv}$ , budeme říkat, že buňka je v kumulačním stavu na látku **Ks**. V každém časovém okamžiku (kroku simulace programu) je buňka v kumulačním stavu na právě jednu látku z **Lv**. Ten se určí následujícím způsobem: Bud'

$$NachytanoCelkem = \sum_{Me \in Membrane} Nachytano(Me) , \quad (3.1)$$

kde  $Nachytano(Me)$  značí vektor množství jednotlivých látek, která se nachytala na receptory v membránovém elementu  $Me$ . Kumulační stav bude odpovídat indexu největší složky vektoru  $NachytanoCelkem$ . Pokud by takových složek bylo více:

- Je – li mezi nimi ta, která odpovídá **Ks** v předcházejícím čase (před započítáním tohoto kroku simulace programu), pak se **Ks** nezmění.
- Jinak bude **Ks** odpovídat té s nejmenším indexem.

Buňka je tedy v kumulačním stavu na tu látku, které "vidí" ve svém okolí nejvíce.

### 3.2.3 Gradient kumulačního stavu

Buňka si v každém časovém okamžiku zjišťuje gradient látky, na kterou je v kumulačním stavu. Ten se určí:

$$Grad_{Ks} = \frac{1}{|Membrane|} \sum_{Me \in Membrane} Normal(Me) \cdot Nachytano(Me) \quad (3.2)$$

$Normal(Me)$  označuje jednotkový vektor kolmý na hranu čtverce políčka, na kterou  $Me$  nasedá. Směr tohoto vektoru je ven z políčka.  $Nachytano(Me)_{Ks}$  označuje množství látky  $Ks$  nachytané na receptory  $Me$ .

### 3.2.4 Zygotický graf

Zygotický graf hraje úlohu "genetického kódu" modelové buňky a říká, které "geny" se v daném kumulačním stavu budou číst, respektive co v něm má buňka dělat. Bude určovat následující činnosti:

- Vylévání látek do okolí.
- Přidávání receptorů na membránu.
- Rychlost pohybu buňky ve směru (popřípadě proti směru) gradientu látky kumulačního stavu.

Zavedme si zygotický graf  $Zg$  jako trojici  $\langle G_P, G_R, G_M \rangle$ , přičemž:

- $G_P$  je orientovaný graf na množině  $Lv$  jehož hrany jsou ohodnoceny nezápornými vahami. Hrana z vrcholu A do vrcholu B znamená, že je-li buňka v nějakém okamžiku v kumulačním stavu na látku A, pak se v tomto okamžiku do okolí vyleje látka B, a to v množství, které udává váha hrany, přesněji, toto množství se rozdělí rovným dílem mezi membránové elementy, do jejichž vnějších sousedních neobsazených políček (pokud existují) se tato množství vylejí. Grafu  $G_P$  budeme říkat produkční graf.
- $G_R$  je orientovaný graf na množině  $Lv$  jehož hrany jsou ohodnoceny nezápornými vahami. Hrana z vrcholu A do vrcholu B znamená, že je-li buňka v nějakém okamžiku v kumulačním stavu na látku A, pak se v tomto okamžiku přidá na její membránu množství receptorů na látku B (tedy komplementární látky k látce B), které udává váha hrany. Toto množství se rozdělí rovným dílem mezi membránové elementy. (Díl se přičte se k vektoru *Recept* každého membránového elementu.) Grafu  $G_R$  budeme říkat receptorový graf. V programu je možno volit způsob zadávání receptorového grafu - zda se myslí přidávání receptorů určité látky, nebo receptorů na danou látku.
- $G_M$  je zobrazení z  $Lv$  do  $\mathbb{R}$  které udává, jakou rychlostí se bude buňka v daném časovém okamžiku pohybovat ve směru gradientu látky, na kterou je v kumulačním stavu, pokud je velikost tohoto gradientu větší, než parametr *MinGradient*. Říkejme zobrazení  $G_M$  pohybový graf.

Zygotický graf je pro všechny buňky v modelové situaci stejný. Naše buňky jsou v ní tedy buňkami "jednoho organismu", přesněji "jednoho genetického kódu".

### 3.2.5 Pohlcování okolních látek

Buňka může pohlcovat látky, které se navázaly na její receptory v jednotlivých membránových elementech a které nejsou zrovna receptory jiné nasedající buňky. Výpočet úbytku látek v daném sousedním políčku je popsán v odstavci Pevné objekty kapitoly Modelový svět. To, v jakém množství buňka látky pohltní, na ni nemá žádný vliv.

### 3.2.6 Dělení

Nejprve si zodpovězme otázku, kdy se buňka dělí. Buď

$$CellRecept = \sum_{Me \in Membrane} AllRecept(Me) , \quad (3.3)$$

kde  $AllRecept(Me)$  značí množství všech receptorů na membránovém elementu  $Me$ . Buňka se chce dělit, pokud  $CellRecept \geq Treshold$ .  $Treshold$  je parametr modelové situace, je pro všechny buňky stejný nezávisle na počtu membránových elementů jednotlivých buněk. Musí být kladný. Pokud se buňka v daném časovém okamžiku chce dělit, není dceřinnou buňkou vyšlou z nějakého dělení, které proběhlo v tomto okamžiku, a v mířce je dost místa pro dceřinné buňky, buňka se rozdělí. V jednom dělení vzniknou z jedné buňky dvě. Dělení probíhá v jednom časovém okamžiku (kroku simulace programu). Dceřinné buňky mají stejný tvar, jako původní buňka, odpovídající si membránové elementy obsahují poloviční množství receptorů na každou látku oproti původnímu. Defnice obtékání okolím při pohybu se rovněž dědí.

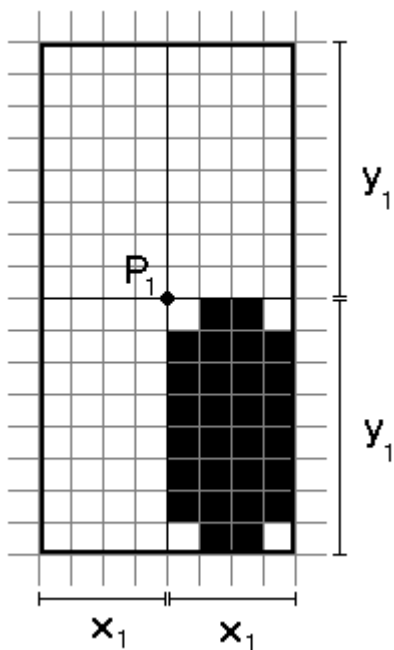
Nyní se zabýváme pozicí dceřinných buněk po rozdělení mateřské. Buňka si může vybrat směr, ve kterém se chce dělit. S ohledem na volné místo v jejím okolí se pak snaží, aby jím byla co nejvíce ovlivněna vzájemná poloha dceřinných buněk. Tento usilovaný směr je určen parametrem *Směr Dělení*, který může nabývat tří hodnot:

- Dle gradientu: Buňka se chce dělit ve směru gradientu látky, na který je v kumulačním stavu, pokud je dost velký. (Pokud dosáhne hodnoty parametru *MinGradient*.) Jinak je nenulový preferovaný směr vybrán náhodně.
- Kolmo na gradient: Stejně jako v minulém případě až na to, že preferovaný směr je tentokrát kolmý na daný gradient.
- Náhodně: Preferovaný směr je vybrán náhodně, nenulový.

Dále budeme chtít, aby dceřinné buňky byly "blízko" mateřské, (a tedy) i sobě navzájem.

Buď  $x_1 = \max \{ x \mid (x, y) \in \mathbf{Tv} \}$ ,  $y_1 = \max \{ y \mid (x, y) \in \mathbf{Tv} \}$ . Označme si nyní mateřskou buňku  $B_0$ , necht' je na pozici  $P_0$ . Dceřinné buňky označíme  $B_1$ , a  $B_2$ .

Je-li  $B_1$  na pozici  $P_1$ , pak se pozice  $P_2$  buňky  $B_2$  musí nacházet na některé straně obdélníku s vrcholy  $P_1 - (x_1, y_1)$ ,  $P_1 + (x_1, -y_1)$ ,  $P_1 + (x_1, y_1)$ ,  $P_1 + (-x_1, y_1)$ . (Označme si tuto podmínku podmínka 1)

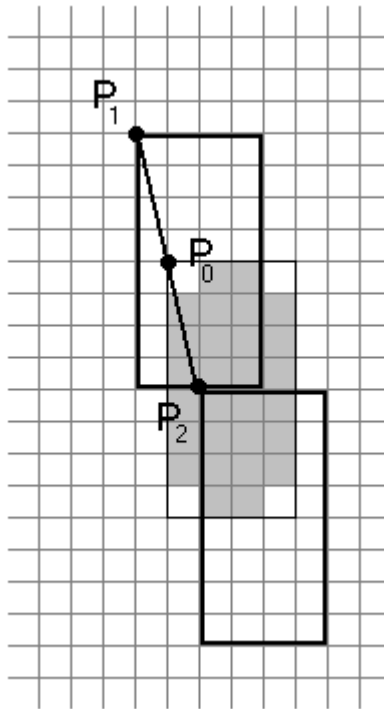


Obrázek 4: Vzájemná pozice dceřinných buněk

Pozice druhé dceřinné buňky musí být na některé straně zvýrazněného obdélníku.

Dále musí platit, že  $P_0$  leží na úsečce  $P_1, P_2$ . (Podmínka 2)





Obrázek 5: Pozice dceřinných buněk vůči mateřské

Dceřinné buňky budou ve zvýrazněných obdélnících.

Nakonec pro pozice  $P_1$  a  $P_2$  musí platit, že dceřinné buňky budou celé v mřížce a na políčkách, která obsadí, se nenachází žádné pevné těleso, kromě mateřské buňky před dělením (Podmínka 3).

Při hledání optimální pozice budeme postupovat následovně: Nechť je  $Q$  nějaká pozice v mřížce,  $S = \{ R \mid \text{dceřinné buňky umístěné na pozice } Q \text{ a } R \text{ by splňovaly podmínku 1} \}$ . Nebude nám vadit, pokud by  $R$  byla mimo mřížku, půjde nám nyní jen o vektory  $(R - Q)$ . Pro každou  $R \in S$  si budeme pamatovat váhu, kterou spočítáme jako

$$w_R = 1 - \frac{|(R - Q) \cdot \text{PrefSmer}|}{\|R - Q\| \cdot \|\text{PrefSmer}\|}, \quad (3.4)$$

kde  $\text{PrefSmer}$  je nenulový vektor udávající preferovaný směr dělení. Váha  $w_R$  nám udává "blízkost" směru  $(R - Q)$  a  $\text{PrefSmer}$ . Nyní budeme postupně brát jednotlivé  $R \in S$  podle neklesajících  $w_R$ , pro dané  $R$ :  $D_{V_R} = \{ (P_1, P_2) \mid P_2 = P_1 + (R - Q), P_1 \text{ a } P_2 \text{ splňují podmínku 2} \}$ . Pro každé  $D = (P_1, P_2)$ ,  $D \in D_{V_R}$  si spočteme váhu

$$w'_D = \max \{ \|P_0 - P_1\|, \|P_2 - P_0\| \} . \quad (3.5)$$

Budeme brát jednotlivé  $D \in D_{VR}$  postupně podle neklesajících  $w'_D$ . První dvojice pozic  $D = (P_1, P_2)$ , která splňuje podmínku 3, bude naše hledaná optimální pozice. Pokud pro žádnou  $R \in S$  nenajdeme v jí odpovídající množině  $D_{VR}$  dvojici pozic, která by splňovala podmínku 3, buňka se nerozdělí. Říkejme tomuto případu, že se buňka nemůže rozdělit.

Při hledání pozic pro dceřinné buňky tedy v první řadě vybíráme dvojice, které se blíží preferovanému směru dělení. Dále se snažíme, aby původní pozice mateřské buňky byla co nejvíce uprostřed pozic dceřinných buněk (3.5). Výběr pozic je možno dále modifikovat některými parametry. Pokud má parametr *Preferováno nasedání* hodnotu *true*, je v rámci daného směru dělení vybírána taková dvojice dceřinných pozic, aby se buňky dotýkaly. Z takových dvojic jsou vybírány ty, kde střed dvojice je co nejbližší středu mateřské buňky. Pokud je *Požadováno nasedání* rovno *true*, pak jsou vybírány takové dvojice pozic, aby počet membránových elementů, kterými se dceřinné buňky dotýkají, byl mezi *Minimum společných elementů* a *Maximum společných elementů* i za cenu odchýlení se od směru dělení. Pokud není nalezena dvojice splňující tyto podmínky, bere se, že se buňka nemůže rozdělit.

Když se buňka chce a může dělit, máme určeny pozice dceřinných buněk a samozřejmě známe pozici mateřské. Z hlediska samotných buněk bychom mohli mateřskou prostě vyjmout z mřížky, umístit ji na jednu dceřinnou pozici (a samozřejmě správně nastavit její atributy) a na zbývající pozici umístit novou odpovídající buňku. My však budeme chtít modelovat dělení buňky také vzhledem ke koncentracím látek v políčkách v její blízkosti. Učiníme to následujícím způsobem: Nejprve budou obě dceřinné buňky "na sobě" na mateřské pozici, budou zabírat stejnou množinu políček. Poté se začnou pohybovat každá ke své cílové pozici, obě urazí svoji dráhu za stejný čas. Během tohoto pohybu se počítá obtékání obou buněk okolními látkami. Obě buňky mohou v této situaci "prostupovat" pevnými tělesy, která by jim stála v cestě. Jde nám jen o koncentrace látek v právě volných políčkách. Nakonec buňky dorazí do svých cílových pozic, žádná pevná tělesa se již nepřekrývají. Celý tento pohyb při dělení je v programu počítán v jednom kroku simulace.

### 3.2.7 Úprava množství receptorů

V každém časovém okamžiku si buňka přidá množství jednotlivých druhů receptorů na svoji membránu podle receptorového grafu. Pokud má na membráně příliš mnoho receptorů, chce se dělit. Je ale možné, že se buňka dělit nemůže. Postupně by tedy množství jejích receptorů mohlo růst nade všechny meze. Největší celkové množství receptorů buňky omezíme hodnotou parametru *Kapacita membrány*. Budeme požadovat, aby *Kapacita membrány*  $\geq$  *Threshold*. Pokud se buňka v nějakém časovém okamžiku chce a nemůže dělit, upravíme její receptory následujícím způsobem:

```

void UpravReceptory ()
    f
        AllRecept = 0;
        for ( Me ∈ Membrane )
            AllRecept += Me.Receptory.Sum ();
        if (AllRecept > KapacitaMembrány)
            f
                Denom = (AllRecept * |Membrane|);
                for (Me ∈ Membrane)
                    Me.Receptory *= KapacitaMembrány /
                                    Denom;
            f
    f

```

Parametr *Kapacita membrány* je stejný provšechny buňky v modelové situaci.

### 3.2.8 Pohyb

Označme si koncentrační gradient látky, na kterou je buňka v kumulačním stavu,  $Gradient(Ks)$ . Pokud  $\| Gradient(Ks) \| > MinGradient$  ( $MinGradient$  je nezáporné číslo, parametr modelové situace), určí se vektor rychlosti pohybu buňky v daném okamžiku jako

$$Rych = G_M(Ks) \cdot \frac{Gradient(Ks)}{\| Gradient(Ks) \|}, \quad (3.6)$$

kde  $G_M$  je pohybový graf. Pokud  $\| Gradient(Ks) \| \leq MinGradient$ , bude rychlost nulová. Pro pohyb buňky na základě její rychlosti platí stejná pravidla, jako pro pohyb pohyblivého tělesa.

### 3.2.9 Shrnutí

Buňka:

- V každém okamžiku zabírá neprázdnou množinu políček mřížky, která je dána jejím tvarem a pozicí.
- Vnější okraje těchto políček tvoří membránu, pomocí které buňka komunikuje se svým okolím.

Ve svém okolí buňka vnímá:

- Látky, na které má receptory, respektive jejich koncentrace.
- Koncentrační gradient látky, na kterou je v kumulačním stavu.

Buňka vykonává (může vykonávat) následující činnosti:

- Určení kumulačního stavu
- Pohlcování látek nachytaných na receptory
- Vylévání látek do okolního prostředí
- Přidávání receptorů na membránu
- Úprava množství receptorů
- Pohyb
- Dělení

Život modelové buňky v jednom časovém okamžiku můžeme popsat následujícím algoritmem:

```
void CellSimStep()  
    f  
        Ks = GetMostSeenLigand();  
        Membrane.ConsumeSeenLigands (  
                                PohlčeníNachytanýchLátek);  
        ProduceLigands (ZygGraph.GetProduction (Ks));  
        AddReceptors (ZygGraph.GetReceptAdd (Ks));  
        Gradient = GetGradient (Ks);  
        if (Gradient.Norm() > MinGradient)  
            f  
                Speed = ZygGraph.GetPohyb (Ks);  
                Move (Speed);  
            f  
        Receptors = GetAllReceptors ();  
        if (Receptors > Treshold)  
            if (!TryToDivide ())  
                UpravReceptory ();  
    f
```

# 4 Difúze

## 4.1 Fyzikální definice difúze

Difúze je proces, který vede k vyrovnávání koncentrací látek v rámci jednoho skupenství. Zákony difúze dávají do souvislosti velikost toku difundující látky a koncentrační gradient, který je za tento tok zodpovědný. Definujme difúzní tok  $J$  látky ve směsi s jinými látkami jako množství této látky, které projde kolmo na referenční rovinu jednotkové velikosti za jednotku času. Jednotka  $J$  je: množství látky přes  $\text{cm}^2$  za sekundu. Na volbě jednotky pro množství látky nezáleží, může to být například hmotnost, nebo počet molekul.

Je-li  $x$  souřadnice zvolená kolmo na referenční rovinu a  $c$  koncentrace difundující látky daná jako množství látky na centimetr krychlový, potom Fickův první zákon difúze můžeme napsat ve tvaru

$$J = -D \frac{dc}{dx} , \quad (4.1)$$

kde  $D$  je koeficient difúze pro danou látku. Vztah (4.1) můžeme formulovat nezávisle na systému souřadnic vektorově

$$\mathbf{J} = -D \nabla c . \quad (4.2)$$

(4.2) říká, že vektor  $\mathbf{J}$  difúzního toku má opačný směr, než koncentrační gradient, který ho způsobuje, a je přímo úměrný velikosti tohoto gradientu, přičemž referenční rovina je na něj kolmá.

## 4.2 Náš model difúze

Považujme nyní koncentraci  $c$  za funkci jedné proměnné, která má všude derivaci. Můžeme psát

$$c(x+h) - c(x) = c'(x)h + \eta(h)h . \quad (4.3)$$

Položíme-li  $h = 1$  a zanedbáme-li hodnotu  $\eta(h)$ , můžeme (4.1) nahradit vztahem

$$J = -D(c(x+1) - c(x)) . \quad (4.4)$$

Při modelování difúze vyjdeme ze vztahu (4.4). Prostředí budeme mít rozdělené na políčka. Sousedí-li spolu 2 políčka, pak je to jednotkovou plochou a uvažujeme o nich, že jsou od sebe vzdálena jednotku vzdálenosti. Za jednotku času budem považovat 1 krok simulace. Koncentraci látky v políčku považujem v každém kroku za konstantní. Uvažme dvě sousední políčka  $P_x$  a  $P_{x+1}$ , která mají koncentrace difundující látky  $C_x$  a  $C_{x+1}$ . Máme

$$J = -D(C_{x+1} - C_x) . \quad (4.5)$$

V rámci jednoho kroku se  $C_x$  sníží o  $J$  jednotek množství látky a  $C_{x+1}$  se zvýší o  $J$ . Pro dva následující kroky můžeme koncentraci v políčku  $P_x$  vyjádřit

$$C_x^{t+1} = C_x^t - D(C_x^t - C_{x+1}^t) . \quad (4.6)$$

Uvažujme nyní, že políčko  $P_x$  má  $n$  sousedů  $Q_1, \dots, Q_n$ , se kterými probíhá difúze, s koncentracemi  $S_1, \dots, S_n$ . Koncentraci v  $P_x$  můžeme vyjádřit

$$C_x^{t+1} = C_x^t - \sum_{i=1}^n D(C_x^t - S_i^t) . \quad (4.7)$$

Předpokládáme, že  $D$  je se všemi sousedy stejný. Upravme (4.7) na

$$C_x^{t+1} = (1 - nD)C_x^t + D \sum_{i=1}^n S_i^t . \quad (4.8)$$

Jaké hodnoty můžeme pro  $D$  volit? Potřebujeme, aby byla splněna podmínka, že difuzní tok má opačný směr než koncentrační gradient, který ho způsobuje. (V našem případě, že tok probíhá z políčka s vyšší koncentrací do políčka s nižší koncentrací.) Pro její splnění musí určitě být  $D \geq 0$ . (Viz (4.5).)

Pro hledání horní meze  $D$  použijeme naši podmínku jako motivaci pro následující požadavek: Buď  $Q_m$  takový soused políčka  $P_x$ , že  $S_m \leq S_i$   $i=1, \dots, n$ . Uvažujme pouze difúzi mezi  $P_x$  a jeho sousedy. Je-li  $C_x \geq S_m$ , potom budeme vyžadovat, aby také po provedení kroku platilo  $C_x \geq S_m$ . (Koncentrace se "nanejvýš" vyrovnají, ale nestane se, že by koncentrace ve "zdrojovém" políčku byla nižší, než v onom "cílovém".)

Mějme speciální případ. Necht'  $S_1 = S_2 = \dots = S_n$ , označme tuto koncentraci  $S$ . Dále necht'  $C_x \geq S$ . Uvažujme pouze difúzi mezi  $P_x$  a jednotlivými sousedy. Pro splnění požadavku musí tedy být

$$D \leq \frac{1}{n+1} . \quad (4.9)$$

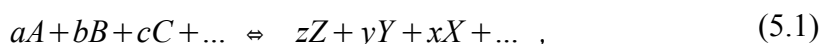
(Viz (4.7).) Pokud by pro některého souseda platilo, že  $S_i \geq S$ , potom by množství látky, které v jednom kroku přejde z  $P_x$  do  $Q_i$  bylo menší nebo rovné tomu, které přejde z  $P_x$  do jiného souseda. Tedy  $C_x$  po provedení kroku se oproti předchozímu případu nezmenší. Snadno již nahlédneme, že (4.9) je postačující pro splnění požadavku v obecném případě.

Odvodili jsme vztahy pro výpočet difúze. Pro programovou implementaci můžeme použít (4.8), popř. (4.7), pokud by konstanta  $D$  nebyla všude stejná. Parametrem  $D$  se dá nastavovat rychlost difúze, přičemž jsme spočítali jeho omezení.

# 5 Chemické reakce

## 5.1 Zavedení chemických reakcí

Zaměříme se na reakce, ve kterých zanikají a opět vznikají vazby, předně kovalentní. Takovou reakci můžeme zapsat chemickou rovnicí



kde velká písmena reprezentují konkrétní molekuly nebo atomy a malá písmena reprezentují stechiometrické koeficienty. Rychlost dopředné reakce  $R_{dop}$  je

$$R_{dop} = k_d [A]^a [B]^b [C]^c \dots \quad (5.2)$$

a rychlost zpětné reakce  $R_{zp}$  je

$$R_{zp} = k_z [Z]^z [Y]^y [X]^x \dots, \quad (5.3)$$

kde  $k_d$  je konstanta rychlosti dopředné reakce,  $k_z$  je konstanta rychlosti zpětné reakce a  $[L]$  znamená koncentraci látky  $L$ . Jestliže jsou rychlosti dopředné a zpětné reakce stejné, je soustava v rovnovážném stavu a koncentrace všech látek zůstávají nezměněny. V tom případě platí

$$k_d [A]^a [B]^b [C]^c \dots = k_z [Z]^z [Y]^y [X]^x \dots, \quad (5.4)$$

$$\frac{k_d}{k_z} = \frac{[Z]^z [Y]^y [X]^x \dots}{[A]^a [B]^b [C]^c \dots} \quad (5.5)$$

Označme

$$K_r = \frac{k_d}{k_z}, \quad (5.6)$$

$K_r$  je konstanta rovnovážného stavu. V případě, že koncentrace látek v soustavě nesplňují (5.5), probíhá dopředná reakce rychleji než zpětná, nebo naopak. Koncentrace jednotlivých látek se mění, až je dosaženo rovnovážného stavu<sup>6</sup>.

---

<sup>6</sup> To, zda reakce skutečně probíhá, záleží ještě na dalších vlivech, které pro jednoduchost neuvažujeme.



## 5.2 Náš model chemických reakcí

My budeme modelovat reakce, do kterých vstupují dva reaktanty a vystupuje z nich jeden produkt, všechny stechiometrické koeficienty budeme považovat za rovné jedné. Chemická rovnice tedy vypadá



a pro koncentrace v rovnovážném stavu platí

$$\frac{[Z]}{[P][Q]} = K_r \quad (5.8)$$

Nechť jsou A, V a X nějaké chemické látky, necht' spolu reagují podle chemické rovnice  $A + V \rightleftharpoons X$ . Necht' jsou  $[A_0]$ ,  $[V_0]$  a  $[X_0]$  libovolné koncentrace A, V a X před proběhnutím reakce. Ptáme se, jaké budou jejich koncentrace  $[A_1]$ ,  $[V_1]$  a  $[X_1]$  po proběhnutí reakce, tedy po dosažení rovnovážného stavu.

Označme si  $C_A$  celkovou koncentraci látky A ve volné i vázané formě, která vstupuje do reakce. Budeme tedy mít  $C_A = [A_0] + [X_0]$ .  $C_A$  se během reakce nezmění (zákon zachování hmoty), takže rovněž platí

$$C_A = [A_1] + [X_1] \quad (5.9)$$

Obdobně si označíme  $C_V$  pro látku V,  $[V_0] + [X_0] = C_V$ ,

$$C_V = [V_1] + [X_1] \quad (5.10)$$

Po proběhnutí reakce platí

$$\frac{[A_1][V_1]}{[X_1]} = K \quad (5.11)$$

Budeme předpokládat, že

$$C_A \geq 0, \quad C_V \geq 0, \quad K \geq 0 \quad (5.12)$$

a budeme požadovat, aby

$$[A_1] \geq 0, \quad [V_1] \geq 0, \quad [X_1] \geq 0 \quad (5.13)$$

Na základě (5.9), (5.10), (5.12) a (5.13) musí platit

$$[X_1] \leq C_A, \quad [X_1] \leq C_V \quad (5.14)$$

Vyjádřením  $[A_1]$  z (5.9),  $[V_1]$  z (5.10) a dosazením do (5.11) dostáváme

$$\frac{[C_A - X_1][C_V - X_1]}{[X_1]} = K, \quad (5.15)$$

což upravíme na

$$[X_1]^2 - [X_1](C_A + C_V + K) + C_A C_V = 0 \quad (5.16)$$

a kvadraticou rovnicí vyřešíme

$$[X_1] = \frac{C_A + C_V + K \pm \sqrt{(C_A + C_V + K)^2 - 4C_A C_V}}{2} \quad (5.17)$$

Snadno se dokáže, že pro libovolná  $a, b$  platí:

$$(a+b)^2 - 4ab \geq 0, \quad (a+b)^2 - 4ab > 0 \text{ pro } a \neq b \quad (5.18)$$

Tedy vidíme, že diskriminant rovnice (5.17) je nezáporný. Nyní ukážeme, že kořen

$$[X_1] = \frac{C_A + C_V + K + \sqrt{(C_A + C_V + K)^2 - 4C_A C_V}}{2}, \quad (5.19)$$

až na případ, kdy kořeny splývají, nesplňuje podmínky (5.14).

Pokud  $C_V \neq C_A$ , nemůže zároveň platit

$$C_A \geq \frac{C_A + C_V}{2}, \quad C_V \geq \frac{C_A + C_V}{2}. \quad (5.20)$$

Necht' třeba

$$C_A < \frac{C_A + C_V}{2} \quad (5.21)$$

Potom by bylo

$$\frac{C_A + C_V + K + \sqrt{(C_A + C_V + K)^2 - 4C_A C_V}}{2} < \frac{C_A + C_V}{2}, \quad (2.22)$$

což ovšem nemůže platit. Pro  $C_V = C_A$  buď kořeny splývají ( $K=0$ ) nebo pro  $K \neq 0$  vidíme z (5.19)

což odporuje (5.14). Jako řešení nám zbývá kořen

$$[X_1] = C_A + \frac{K}{2} + Z, \quad Z \geq 0 \quad (5.23)$$

$$[X_1] = \frac{C_A + C_V + K - \sqrt{(C_A + C_V + K)^2 - 4C_A C_V}}{2} . \quad (5.24)$$

To, že (5.24) splňuje (5.13), (5.14) již ukazovat nebudeme. Z (5.9) okamžitě dostáváme

$$[A_1] = C_A - [X_1] = \frac{C_A - C_V - K + \sqrt{(C_A + C_V + K)^2 - 4C_A C_V}}{2} \quad (5.25)$$

a z (5.10) dostáváme

$$[V_1] = C_V - [X_1] = \frac{C_V - C_A - K + \sqrt{(C_A + C_V + K)^2 - 4C_A C_V}}{2} . \quad (5.26)$$

## 6 Další směřování práce

Původní biologickou motivací pro naši práci bylo putování buněk v embryu kuřete. Můžeme si jí oživit pohledem na obrázek 1. (Viz také odstavec 1.3.) Během vytváření modelu této situace se ukázalo, že není příliš "zajímavý", že nepřináší nějaké další osvětlení dynamiky migrace samotné, jakož i výraznější osvětlení osatních pozorovaných aspektů situace. Náš model pohybu buněk definuje, že buňka se pohybuje ve směru gradientu látky, které vidí nejvíce, rychlostí, která je zadána jako parametr. Pohyb buněk v dané modelové situaci je přímým důsledkem tohoto "axiomu". Pokud mají buňky v cestě překážky, které je možno obejít, typicky, je postupem času obejdou a doputují do míst s největší koncentrací "navigující" látky. V modelu tedy vytvoříme poměrně přímočaré simulace, jejichž výsledky se alespoň v základních rysech podobají modelovaným situacím. Bylo by sice možné pokoušet se tuto podobnost "zvyšovat", avšak sledování tohoto cíle se nezdá příliš relevantní pro získávání netriviálních výsledků na základě modelové abstrakce buňky.

Obecně bychom mohli říci, že daná abstrakce v našem modelu se nejevila být taková, že na jejím základě vneseme do dané situace něco nového, ale ani její netriviální možné zdůvodnění, které by bylo "zajímavé" z hlediska modelu. Z opačného pohledu můžeme říct, že daná pozorování "reality" se nezdála být nepřímochaře odůvodnitelná modelem, tedy neskýtala možnost jeho aplikace, aby tato mohla přinést nové netriviální poznatky vyjádřené v jeho řeči.

Zároveň se však objevila jiná témata, která možná nejsou tak těsně spjata s konkrétními biologickými výzkumy, jako předcházející, avšak skýtají prostor pro aplikaci našeho modelu, která do nich může vnášet nové světlo, a tato osvětlení mohou vycházet z jeho netriviálního použití. V dalších odstavcích se budeme věnovat dvěma takovým.

# 7 Diferenciace homogenní tkáně

## 7.1 Homogenní tkáň

V živých organismech pozorujeme různé typy buněk, jako například svalové, nervové, kostní, nebo jaterní. Pokud je buněk jednoho typu více pohromadě, popřípadě jsou li nějak specificky uspořádány, chápeme je jako tkáň. Budeme-li předpokládat, že organismus vznikl z jedné buňky – zygoty a že všechny jeho (námi zkoumané) buňky mají stejný genetický kód, můžeme si položit otázku, kdy již je nějaká konkrétní buňka buňkou určitého typu (popřípadě tkáně) a kdy ještě ne.

V našem modelu si zavedeme typ buněk následovně: Budeme říkat, že buňky  $B_1$  a  $B_2$  jsou (silně) stejného typu právě když

$$Receptory(B_1, L) = Receptory(B_2, L), L \quad (7.1)$$

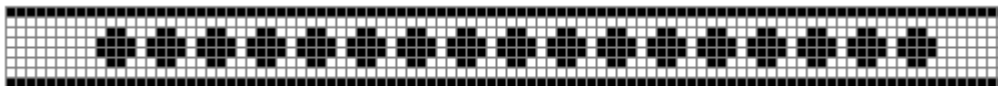
kde  $Receptory(B, L)$  značí celkové množství receptorů buňky  $B$  na látku  $L$ .

## 7.2 Programový model diferenciace

Vznik diferencovaných buněk, tzv. ostrovů, byl zkoumán v práci [4]. Zde navrhne jednodušší model opírající se o naši definici typu buněk.

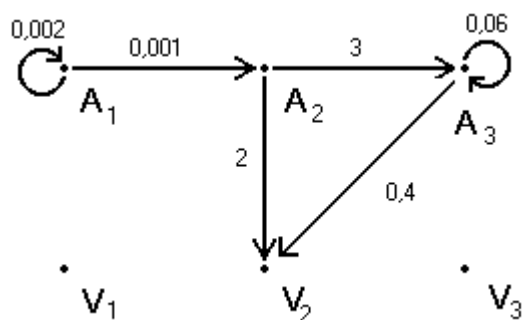
### 7.2.1 Jednorozměrný případ s jedním ostrovem

Vezměme si následující situaci:

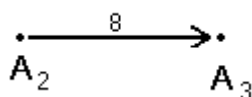


Obrázek 6: Uspořádání buněk pro jednorozměrný model diferenciace

Buňky jsou vedle sebe v řadě a kolem nich je místo pro difúzi. V okolí jsou na počátku koncentrace všech látek nulové. Produkční a receptorový graf vypadají následovně:



Obrázek 7: Produkční graf pro jednorozměrný model diferenciacie s jedním ostrovem



Obrázek 8: Receptorový graf pro jednorozměrný model diferenciacie

Na začátku jsou všechny buňky v kumulačním stavu na látku  $A_1$  a jsou stejného typu. Množství receptorů pro buňku na jednotlivé látky vypadá:

$A_1$	1	$A_2$	2	$A_3$	0,5
$V_1$	0	$V_2$	0	$V_3$	0

Tabulka 1: Počáteční množství receptorů buňky na jednotlivé látky pro jednorozměrný model diferenciacie

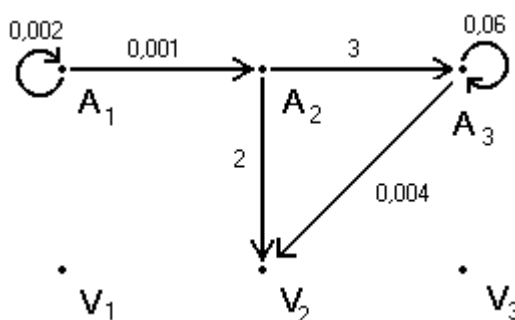
Buňky nejprve produkují látky  $A_1$  a  $A_2$ . Je důležité, že látky  $A_1$  se produkuje více než  $A_2$ . Buňky tedy vidí víc  $A_1$  a zůstávají v kumulačním stavu na ní. V prostředí roste koncentrace  $A_1$  a  $A_2$ , přičemž díky difúzi se po nějaké době vytvoří maximum koncentrací uprostřed naší situace. Po určitém čase je množství látky  $A_1$  kolem střední buňky (a později i kolem ostatních buněk) tak velké, že je za hranicí toho, co může buňka vnímat. (Buňka může vidět jen tolik látky, kolik na ní má receptorů. Je-li látky více, buňka stále vidí množství menší nebo rovné počtu jejích receptorů.) Zatímco se viděné množství  $A_1$  ustálilo na (nebo se blíží) hodnotě 1, koncentrace látky  $A_2$  stále roste. Buňky jsou schopny detekovat větší množství  $A_2$  než  $A_1$ . V momentě, když vnímaná koncentrace  $A_2$  přesáhne vnímanou koncentraci  $A_1$ , daná buňka se přepne do kumulačního stavu na  $A_2$ . V našem případě se jako první přepne střední

buňka. Nyní budeme chtít zajistit, aby ostatní buňky již nezměnily svůj kumulační stav a aby se naše buňka nepřepela zpátky do stavu na  $A_1$ . Tato jednak vyleje velké množství  $V_2$ , čímž sníží koncentraci  $A_2$  ve svém okolí a zabrání bezprostřednímu přepnutí okolních buněk. Dále vyleje dostatečně mnoho látky  $A_3$  a přidá si na ní receptory tak, aby jí v následujícím časovém okamžiku viděla více než ostatních látek a přepela se do kumulačního stavu na ní. Poněvadž ostatní buňky mají na  $A_3$  méně receptorů než na  $A_1$ , neovlivní libovolná koncentrace  $A_3$  jejich kumulační stav, pokud je  $A_1$  v jejich okolí dostatečné množství. V kumulačním stavu na  $A_3$  produkuje naše buňka jednak látku  $A_3$ , aby jí viděla dostatek a udržela se v něm, dále pak produkuje  $V_2$ , pomocí které snižuje koncentraci  $A_2$  a zamezuje tak přepnutí ostatních buněk.

Tato modelová situace je v souboru *Lineární diferenciace 1 ostrov.st.* Nešlo nám v ní o to, že ona prostřední buňka přestala být buňkou stejného typu, jako ostatní, ale že se byla schopna udržet v jiném kumulačním stavu, zatímco ostatní buňky ten svůj nezměnily. Nový přetrvávající kumulační stav můžeme chápat jako odrazový můstek pro nějakou další diferenciaci, popřípadě dělení. Je také zřejmé, že poté, co se buňka přepnula, přestala být "závislá" na ostatních a vydala se "svou cestou".

## 7.2.2 Jednorozměrný případ s více ostrovy

Uvažme následující produkční graf:



Obrázek 9: Produkční graf pro jednorozměrný model diferenciace s více ostrovy

Je stejný, jako v minulém případě až na to, že váha hrany z  $A_3$  do  $V_2$  se zmenšila. Ostatní (počáteční) parametry necht' jsou stejné, jako v předešlém případě. Výsledkem je, že se vytvoří více ostrovů – zde 7. Ostrovy se vytvářejí postupně. Nejprve se přepne buňka uprostřed. Zanedlouho se přepnou buňky ob jednu na každou stranu od středové. Potom přepínání pokračuje ke krajům, přičemž intervaly mezi přepnutými buňkami se zvětšují.

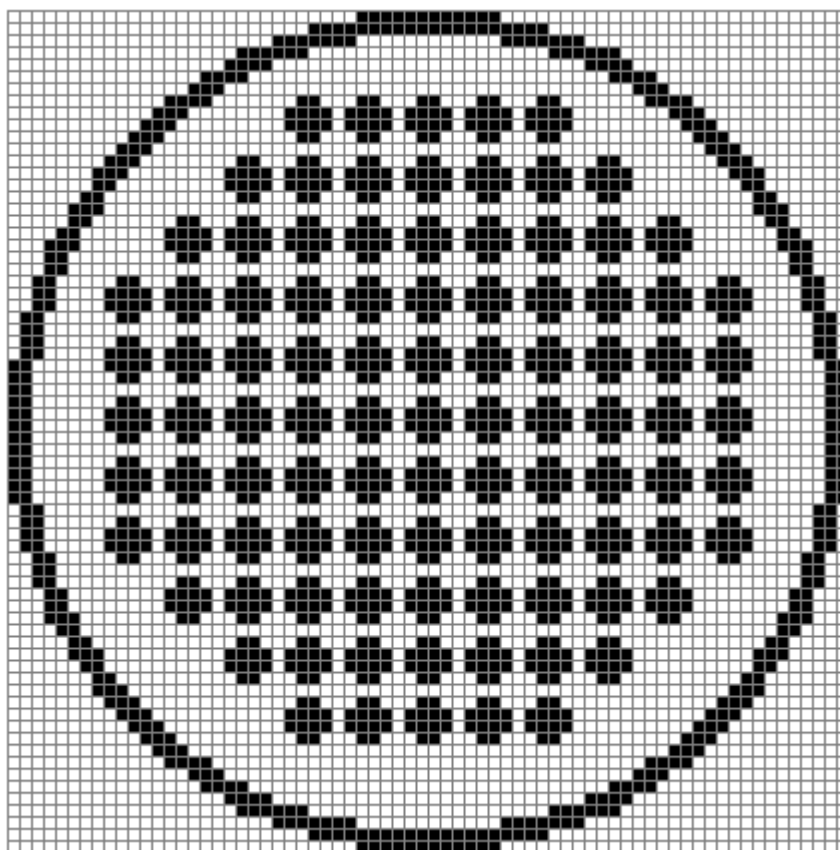
Tato modelová situace je v souboru *Lineární diferenciace ostrovy.st.* Ukazuje se, že jednorázové vylití "velkého" množství látky  $V_2$  v kumulačním stavu na  $A_2$  samo o sobě nebylo postačující na ovlivnění většího okolí dané buňky.

Pokud bude mít váha hrany z  $A_3$  do  $V_2$  hodnotu  $0,04$ , vytvoří se tři ostrovy – buňka uprostřed a její "sousedky" ob dvě na každou stranu. Vidíme, že jednou hodnotou v produkčním grafu můžeme měnit počet ostrovů. Kdybychom se na

ostrovy dívali jako na zárodky nějakých viditelných částí budoucího organismu, dostali jsme způsob, jak by změna kvantitativního parametru na úrovni produkce látek (obecně vnitřního chování buňky) mohla způsobit změnu kvality pozorovatelnou pouhým okem.

### 7.2.3 Dvojměrný případ

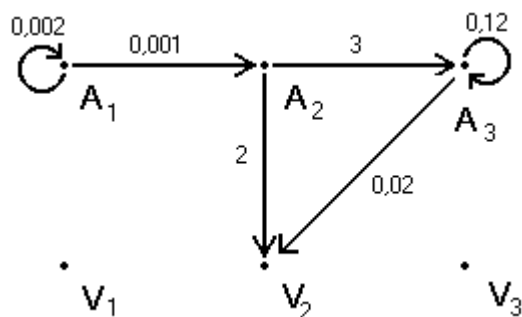
Nyní se zabýváme situací, kde jsou buňky uspořádány do dvojrozměrné mřížky, kolem jednotlivých buněk je opět místo pro difúzi. Situace bude vypadat následovně:



Obrázek 10: Uspořádání buněk pro dvojrozměrný model diferenciac

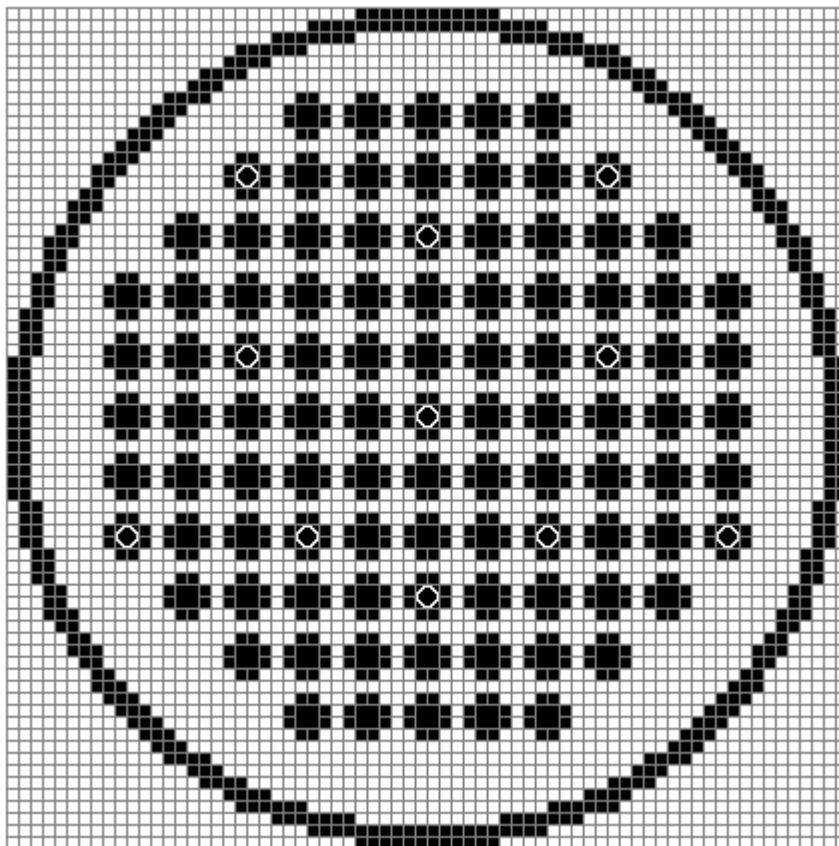
Produkční graf bude vypadat:





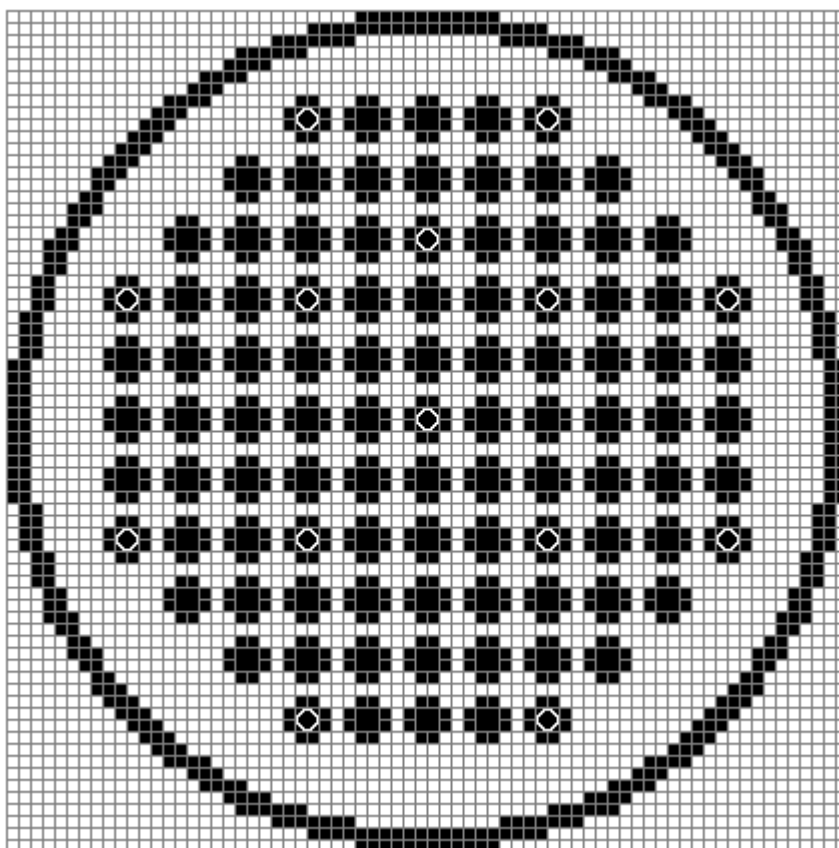
Obrázek 11: Produkční graf pro dvojrozměrný model diferenciacie

Od grafu pro jednorozměrný případ se liší hranami vedoucími z  $A_3$ , tedy v produkcích "přepnutých" buněk. Všechny ostatní parametry situace jsou stejné, jako v jednorozměrném případě. Průběh diferenciacie je obdobný, jako v jednodimenzionálním modelu vytvoření více ostrovů. Maxima koncentrací  $A_1$  a  $A_2$  je dosahováno uprostřed situace, kde se také přepne první buňka. Postupně se vytvářejí ostrovní buňky podle vzdálenosti od středu.



Obrázek 12: Přepnuté buňky (označené mezikružím) ve dvojrozměrném modelu diferenciaci

Situace je v souboru *Diferenciace 2D.st.* Pokud změníme váhu hrany z  $A_2$  do  $V_2$  na hodnotu  $0,02$ , tedy produkce látky  $V_2$  v kumulačním stavu na  $A_2$  bude stejná jako ve stavu na  $A_3$ , bude výsledek vypadat takto:



Obrázek 13: Přepnuté buňky ve dvojrozměrném modelu diferenciacce, váha hrany z  $A_2$  do  $V_2$  v produkčním grafu = 0,02 .

Vidíme, že se "příliš" neliší od předchozího. V jednorozměrném případě mělo vylití velkého množství látky  $V_2$  ve stavu  $A_2$  důležitý význam pro potlačení přepnutí sousedních buněk. Zde tomu tak není. Toto vylití sice oddálí "sousední" diferencované buňky, avšak není nutné pro inhibici přepnutí nejbližších sousedů. Obecně můžeme usuzovat, že dvojrozměrné uspořádání buněk by v reálných situacích mohlo být v určitém smyslu přirozenější pro tvorbu ostrovů, než jednorozměrné.

# 8 Model růstu vlásečnic

## 8.1 Autonomní buňka

### 8.1.1 Motivační příklad

Představme si živou buňku, která vykazuje následující chování: Pokud je sama v určitém látkovém prostředí, nic nedělá. Jestliže se objeví nějaká konkrétní látka, buňka se dělí, přičemž osa dělení je kolmá na koncentrační gradient této látky. Je-li našich buněk více pohromadě a jsou – li těsně vedle sebe, žádná buňka se nedělí i když je daná látka přítomná. Jestliže je však nějaká buňka na okraji skupiny tak, že s ní není příliš těsně spjata, způsobí ona látka u této buňky dělení.

Náš příklad je sice uměle vykonstruovaný, ale je popsán způsobem, který by mohl vycházet z pozorování pomocí mikroskopu a z hlediska biologie být relevantní informací o naší buňce. Tu bychom mohli pojmenovat, uvést příklady organismů a míst v nich, kde se tyto buňky vyskytují, nebo zkomat molekulární mechanismy buněčného rozpoznávání dané látky, například strukturu receptorů, kterými buňka látku vnímá, nebo hledat geny zodpovědné za syntézu těchto receptorů.

My popíšeme virtuální buňku těchto "pozorovatelných" vlastností v řeči našeho modelu a budeme sledovat její dynamiku.

### 8.1.2 Rozšíření pojmu typu buňky

V odstavci 7.1 jsme definovali, kdy jsou dvě buňky (silně) stejného typu. Nyní si definici rozšíříme. Budeme říkat, že buňky  $B_1$  a  $B_2$  jsou slabě stejného typu právě když

$$\frac{Receptory(B_1, L)}{\sum_{L' \in Lv} Receptory(B_1, L')} = \frac{Receptory(B_2, L)}{\sum_{L' \in Lv} Receptory(B_2, L')}, \quad L \in Lv \quad (8.1)$$

přičemž  $Receptory(B, L)$  značí celkové množství receptorů buňky  $B$  na látku  $L$ . Pokud by byl jmenovatel některého zlomku roven nule, buď hodnota celého zlomku nulová. Dále definujeme typ buňky jako stochastický vektor  $T = (t_1, \dots, t_n)$ ,  $n = |Lv|$ . Budeme říkat, že buňka  $B$  je typu  $T$  právě když

$$\frac{Receptory(B, L)}{\sum_{L' \in Lv} Receptory(B, L')} = t_{Index(L)}, \quad L \in Lv \quad (8.2)$$

Buď  $L \in Lv$ . O buňce  $B$  řekneme, že je slabě  $L$  – vitální právě když svými receptory může (potenciálně) vidět více látky  $L$  než libovolné jiné látky z  $Lv$ .

Buď  $L \in Lv$ . O buňce typu  $T$  s nenulovým celkovým množstvím receptorů na membráně řekneme, že je silně  $L$  – vitální právě když může vidět více látky  $L$  než kterékoli jiné látky z  $Lv$  nezávisle na počtu svých receptorů.

### 8.1.3 Modelování autonomní buňky

Předpokládejme nyní čtyři látky,  $A_1, A_2, V_1, V_2$ ,  $A$  a  $V$  se stejnými indexy jsou spolu ve vaxu. Autonomní buňka bude následujícího typu:

$A_1$	$m$	$A_2$	$c \cdot m$
$V_1$	0	$V_2$	$c \cdot m$

Tabulka 2: Typ autonomní buňky

Nechť  $c > 0$ , dále buďte  $c$  a  $m$  volena tak, aby vektor splňoval požadavky na typ buňky. Předpokládejme, že naše buňka má  $n$  membránových elementů, dále zvolme pevně nějaké  $p \in \mathbb{N}$  tak, aby

$$p \leq \frac{n}{2} . \quad (8.3)$$

Budeme požadovat, aby pokud  $p$  membránových elementů naší buňky nasedá na nějaké jiné membránové elementy, na kterých mohou být přichyceny látky v libovolných množstvích, byla naše buňka silně  $A_1$  – vitální. Dále budeme požadovat, aby pokud má buňka dva sousedy, všechny tři buňky jsou silně stejného typu a sousedé nasedají na buňku celkem na  $2p$  membránových elementech, buňka nebyla slabě  $A_1$  – vitální.

Uvažujme, že buňka má celkem  $k \cdot m$  receptorů na  $A_1$  a  $k \cdot c \cdot m$  receptorů na  $A_2$  ( $V_2$ ),  $k > 0$ . K prvnímu požadavku: Chceme, aby množství receptorů na  $A_1$  na  $n - p$  elementech bylo větší, než množství receptorů na  $A_2$  ( $V_2$ ) na  $p$  elementech. Tedy aby

$$k \cdot m \cdot \frac{(n - p)}{n} > k \cdot m \cdot c \cdot \frac{p}{n} , \quad (8.4)$$

tedy

$$p(c + 1) < n . \quad (8.5)$$

Vzhledem ke druhému požadavku dostaneme nerovnost

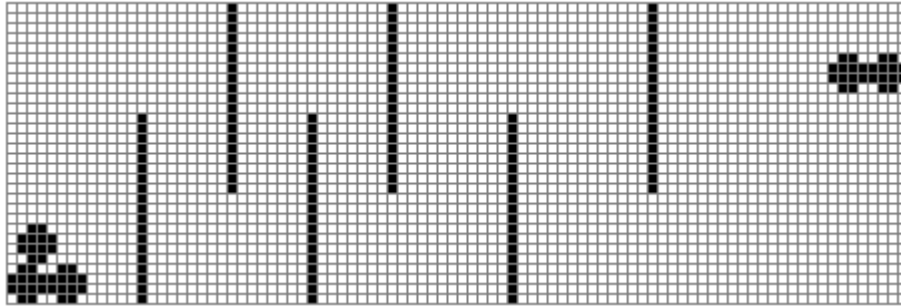
$$2p(c + 1) > n , \quad (8.6)$$

takže oba požadavky jsou splněny, platí – li

$$p(c+1) < n < 2p(c+1) . \quad (8.7)$$

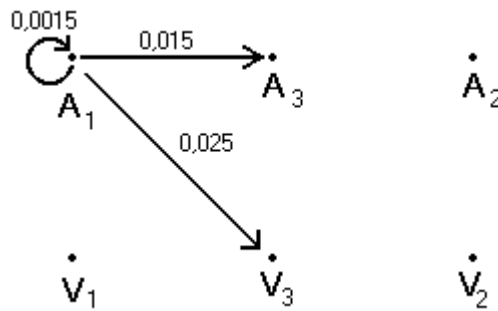
### 8.1.4 Konkrétní příklad autonomní buňky

Vezměme si následující modelovou situaci:

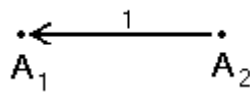


Obrázek 14: Modelová situace pro autonomní buňky

Receptorový a produkční graf vypadjí takto:



Obrázek 15: Receptorový graf pro autonomní buňky



Obrázek 16: Produkční graf v modelové situaci autonomních buněk

Dělení buněk probíhá ve směru gradientu látky kumulačního stavu, dále je nastaveno, aby na sebe dceřinné buňky po rozdělení nasedaly právě jedním membránovým elementem. Tři buňky v levém dolním rohu jsou naše autonomní buňky. Množství jejich receptorů jsou:

$A_1$	0,13	$A_2$	0	$A_3$	1,3
$V_1$	0	$V_2$	0	$V_3$	2,115

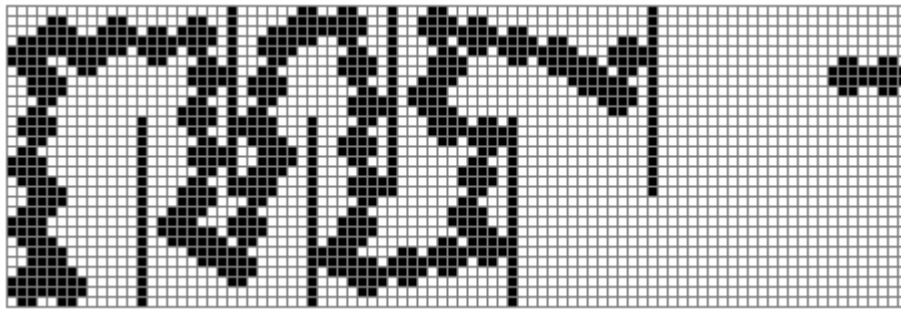
Tabulka 3: Počáteční množství receptorů autonomních buněk

Buňky vpravo nahoře zajišťují produkci látky  $A_1$ , jsou spíše "technického" charakteru, nepřepínají se do kumulačního stavu na látku  $A_1$ . Látky  $A_1$ ,  $A_3$ , a  $V_3$  odpovídají látkám  $A_1$ ,  $A_2$ , a  $V_2$  z minulého odstavce.  $V_3$  se sice přidává více, než látky  $A_3$ , ale to v případě sousedění buněk silně stejného typu nevadí. Pomocí receptorů na  $V_3$  vidí buňka vlastně receptory sousední buňky na  $A_3$  a těch je opět "málo". Z počátečního množství receptorů a zygotického grafu je vidět, že naše autonomní buňky jsou slabě stejného typu. Dosadíme-li do nerovností (8.7)  $n = 16$ ,  $c = 10$  a  $p = 1$  vidíme, že jsou pro naši situaci splněny. Buňky tedy splňují požadavky uvedené ve zmíněném odstavci. V programové simulaci je slabě stejný typ buněk zachován jen s určitou přesností, což ale v konečném důsledku nebude vadit. Pokud se tedy naše autonomní buňka dotýká jiné jedním membránovým elementem, je silně  $A_1$  – vitální. Je – li látky  $A_1$  v okolí dostatek, přepne se na ní do kumulačního stavu. Jestliže se buňka rozdělí, pak je to v kumulačním stavu na  $A_1$  a ve směru koncentračního gradientu této látky. (Směr dělení je stejný jako směr daného gradientu jen s přesností omezenou faktem, že dceřinné buňky se musejí dotýkat každá právě jedním membránovým elementem.) Pokud směr dělení nesvráá příliš velký úhel se spojnicí středu naší buňky a té, na kterou nasedá, má jedna z dceřinných buněk nyní dva sousedy. (Předpokládáme, že naše buňka se jinak ničeho nedotýká a v okolí je dost místa.) Jestliže měla sousedka naší buňky, než se tato naše rozdělila, celkové množství receptorů rovné polovině parametru *Threshold*, můžeme po rozdělení všechny tři buňky považovat za silně stejného typu, a tedy, jak plyne z odstavce 8.1.3, není prostřední buňka slabě  $A_1$  – vitální, což znamená, že se nemůže přepnout do kumulačního stavu na  $A_1$ . (Pokud se ona, nebo některá její sousedka neodsunou.) To také znamená, že již nezmění množství receptorů na své membráně a toto tedy bude rovno polovině parametru *Threshold*. Při dalším dělení koncové buňky se bude situace analogicky opakovat. Buňky budou postupným dělením koncové vytvářet řetízek, kde vnitřní buňky se již nebudou dělit. Řetízek se bude v každém místě vinout přibližně ve směru gradientu koncentrace látky  $A_1$ , jaký byl v okamžiku daného dělení. Někdy se může stát, že se právě vzniknuvší dvoubuněčný konec řetízku utrhne a všechny volné konce budou mít potenciál k dalšímu růstu. Také se může stát, že se konec řetízku k němu někde přichytí, vytvoří se "smyčka" a růst ustane. Je též dobré si uvědomit, že obsazené políčko vedle membránového elementu koncové buňky, kterým se "drží" zbytku řetízku, způsobuje nepřesnost v detekci koncentračního gradientu látky  $A_1$  touto buňkou.

Doplňme ještě, že levá ze dvou buněk v pravém horním rohu v naší situaci obsahuje receptory na  $A_3$  a  $V_3$ , takže pokud se jí koncová buňka řetízku dotkne, přestane být (slabě)  $A_1$  – vitální. V tom momentě se také tato produkující buňka přepne

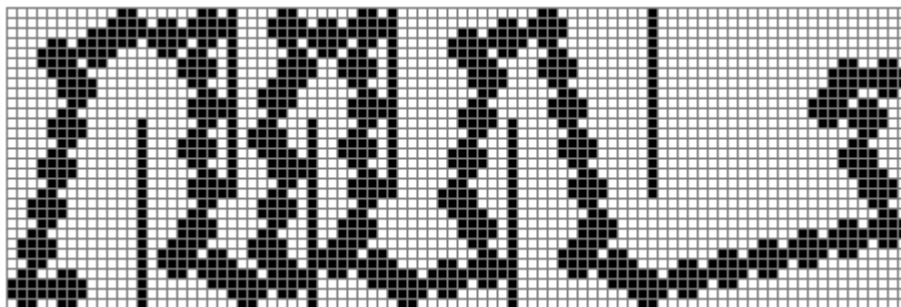
do stavu na  $A_3$  nebo  $V_3$  a produkce látky  $A_1$  ustane. Pravá z dvojice buněk má na své membráně látku  $A_2$ , kterou je levá schopna vnímat a přepnout se do jejího stavu (pokud není dosažena řetízkem).

Naše modelová situace je uložena v souboru *Cevy I.st.* Po nějaké době dodifunduje látka  $A_1$  až k autonomním buňkám. Koncová autonomní buňka se na ní přepne, ale hned se zase "vrátí zpět". (Přidá více receptorů na  $A_3$ , té je na sousední buňce "mnoho".) Po nějaké době tohoto střídání již buňka vidí pořád více látky  $A_1$ . (Nebo se rozdělí a dceřinná koncová buňka se octne v místě s větší koncentrací  $A_1$ .) Řetízek pak soustavně roste. Je zajímavé sledovat vliv parametru *Pohlčení nachytaných látek* na růst řetízku. Je – li roven nule, potom řetízek nedoroste ke zdroji látky  $A_1$ . Koncová buňka v jednom místě uhne do "slepé uličky", kde již nemá místo pro dělení. (Vzhledem k podmínce jednoho společného membránového elementu dceřinných buněk. ) Jestliže se však tento parametr rovná jedné, řetízek doroste k cíli. "Intenzivní" pohlcování látky  $A_1$  způsobí, že doba od jejího objevení se u "první" koncové buňky do započetí růstu řetízku je výrazně delší, než v předešlém případě. Obě situace ukazují následující obrázky:



Obrázek 17: Autonomní buňky; řetízek nedorazil ke zdroji navigační látky.

*Pohlčení nachytaných látek* = 0



Obrázek 18: Autonomní buňky; řetízek dorazil ke zdroji navigační látky.

*Pohlčení nachytaných látek* = 1



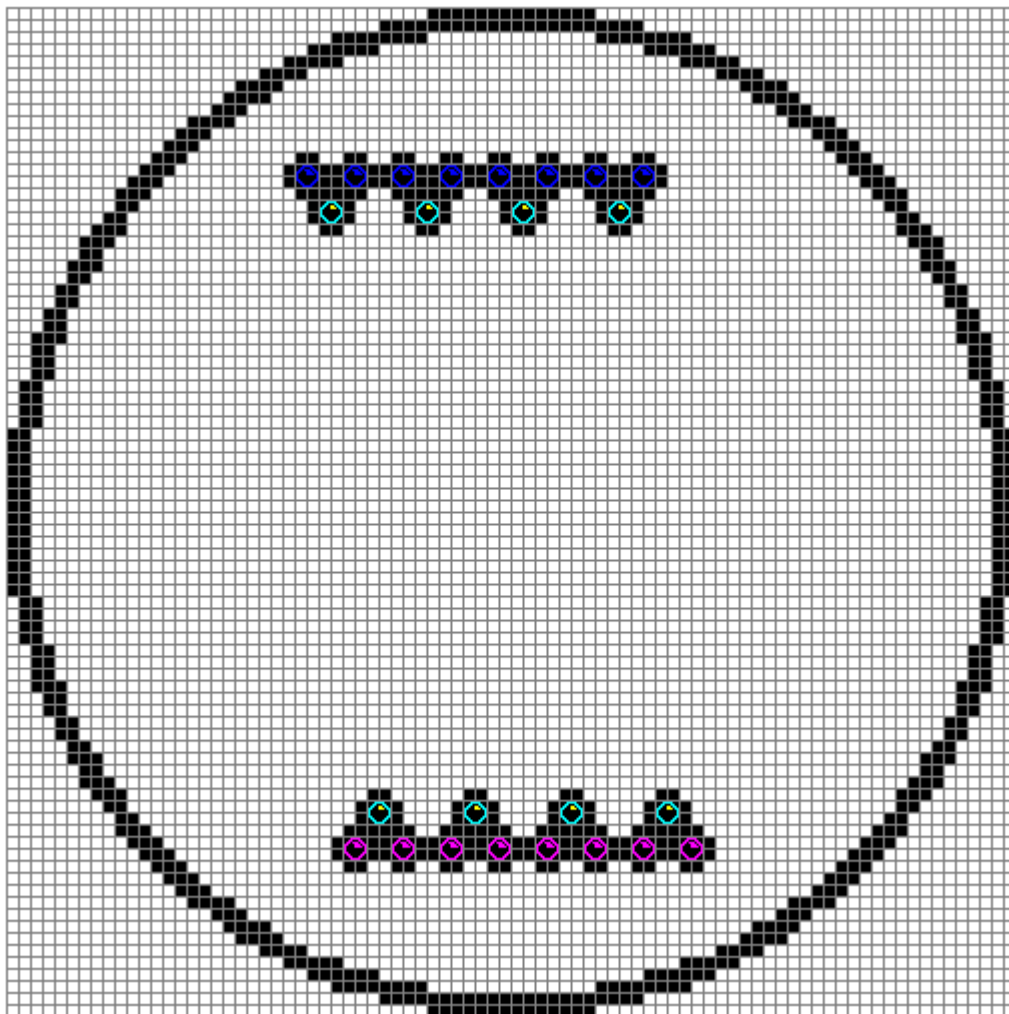
Můžeme usuzovat, že pohlcování látky, za kterou řetízek roste, způsobuje, že se tento lépe drží gradientu její koncentrace v "těsných" prostorách nebo v místech, kde se gradient rychleji mění. Relativně "vodorovný" úsek řetízku v pravé dolní části obrázku 17 může dávat tušit, že pohlcování zmenšuje přesnost rozpoznání gradientu, na "otevřených prostranstvích" nebo pokud se tento v závislosti na pozici příliš nemění.

## 8.2 Vlasečnice

Autonomní buňku, kterou jsme zkonstruovali v minulých odstavcích, nyní mírně vylepšíme a použijeme k modelu růstu protichůdných řetízků, které se umějí vzájemně "napojit". Nepůjde nám tedy o věrné vystižení vnitřní struktury jednotlivé vlasečnice, ale o obecný princip navigace a spojování rostoucích vláken tvořených buňkami. Na základě našeho modelu se pokusíme poskytnout osvětlení dynamiky vytváření sítě vlasečnic.

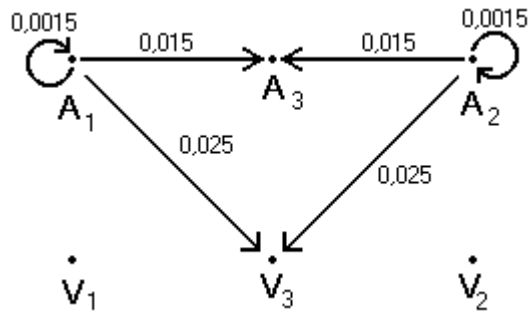
### 8.2.1 Modelová situace vzniku sítě vlasečnic

Vezměme si následující situaci:

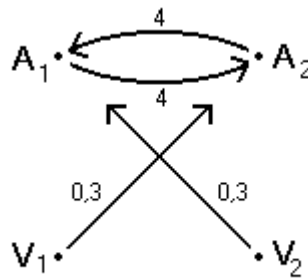


Obrázek 19: Počáteční uspořádání buněk pro model růstu vlásenic

Receptorový a produkčního graf budou vypadat:



Obrázek 20: Receptorový graf pro model růstu vlásčenic



Obrázek 21: Produkční graf pro model růstu vlásčenic

Podívejme se na skupinu buněk nahoře na obrázku 19. Ty, které jsou v řadě těsně vedle sebe, mají receptory pouze na látku  $V_1$ . Říkejme jim produkující. Do této řady jsou zespoda vsazeny čtyři autonomní buňky. Jejich (počáteční) množství receptorů jsou:

$A_1$	0,127	$A_2$	0	$A_3$	1,273
$V_1$	0	$V_2$	0	$V_3$	2,121

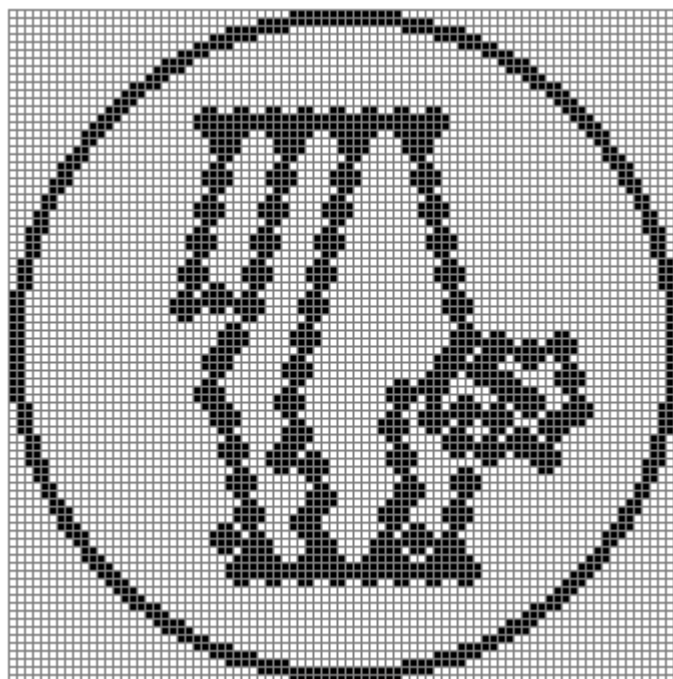
Tabulka 4: Počáteční rozložení receptorů pro první skupinu autonomních buněk pro model růstu vlásčenic

Protože autonomní buňky mají na své membráně receptory na  $A_1$  (tedy látku  $V_1$ ), produkující buňky vidí  $V_1$  a jsou v kumulačním stavu na ní, takže produkují látku  $A_2$ . Autonomní buňky zase vidí na svých sousedkách látku  $A_1$  a jsou na ní přepnuty. Nyní tedy všechny buňky v naší skupině produkují látku  $V_1$ , autonomní buňky si navíc přidávají receptory. Po krátké době se autonomní buňky rozdělí. Dceřinná buňka, která je

na pozici mateřské, nyní vidí více látky  $V_3$  ( $A_3$ ), než  $A_1$ . Z autonomních buněk se vytvoří dvojčlenné řetízky, přičemž produkující buňky stále produkují látku  $A_2$ . V dolní skupině buněk dole na obrázku je situace obdobná, přičemž látky  $A_1$  a  $V_1$  si vyměnily úlohy s látkami  $A_2$  a  $V_2$ . (Tedy produkující buňky mají receptory na  $V_2$  a autonomní na  $A_2$  místo na  $A_1$ , množství receptorů odpovídajících si látek jsou stejná.) Po nějaké době látka  $A_1$  produkovaná dolními produkujícími buňkami dodifunduje k horním autonomním buňkám a koncové buňky dvojčlenných řetízků postupně začnou růst. Dole je situace analogická. Konce řetízků ve stavu na  $A_1$  ( $A_2$ ) produkují "hodně"  $A_2$  ( $A_1$ ). Pokud se konce protichůdně rostoucích řetízků k sobě dostatečně přiblíží, začnou se vzájemně navigovat jeden na druhý. Jestliže se tyto koncové buňky dotknou (a zůstanou spojeny se zbytky svých původních řetízků), přestanou být slabě  $A_1$  ( $A_2$ ) – vitální, růst ustane a vznikne jedno vlákno.

Při napojování konců mohou vzniknout nové řetízky, popřípadě může dojít k rozvětvení díky tomu, že není dost místa pro dceřinné buňky dělivší se koncové. Směr dělení se potom výrazně odchýlí od směru gradientu dané látky, buňka (buňky) ztratí kontakt s původním vláknem a mohou vznikat nové konce s potenciálem růst. Vzniklé výběžky se typicky po nějaké době růstu připojí k "hlavnímu" vláknem. Tento jev můžeme chápat jako důsledek "nepřesnosti" našeho modelu. Je ale také možno se na něj dívat jako na jeho legitimní součást. Též bychom ho mohli brát jako model nějaké patologické situace, defektu v napojování vlasečnic.

Naše situace je v souboru *Vlasecnice 1.st.* Parametr *Pohlčení nachytných látek* je roven jedné. Po ustání růstů vypadá výsledek takto:

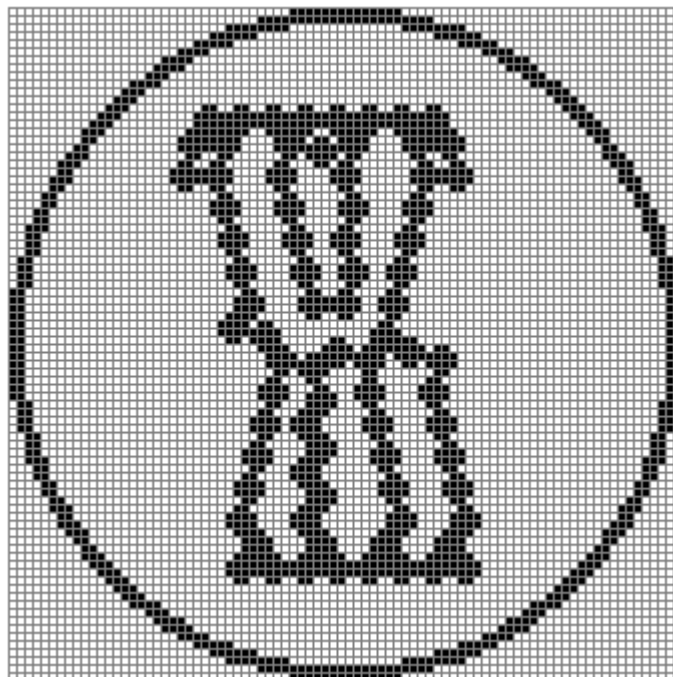


Obrázek 22: Modelové vlásečnice

*Pohlčení nachytaných látek = 1*

Pokud položíme *Pohlčení nachytaných látek* rovno nule, bude situace po ustání růstů vypadat tak, jak ukazuje obrázek 23.

Než pokročíme dál, podívejme se ještě na buňky na pozicích prvních autonomních. Tyto mají schopnost přepnout se na navigační látku produkovanou buňkami z "opačné" strany situace i když z nich už jedno vlákno vychází. Druhé vlákno vyrůstající z takovéto buňky se však téměř okamžitě spojí s prvním (nebo vedlejším). Tato věc je spíše "technickým" důsledkem realizace naší modelové situace, než něčím, co bychom chtěli zkoumat.



Obrázek 23: Model růstu vlásečnic

*Pohlčení nachytaných látek = 0*

V případě, že buňky nepohlcují látky ve svém okolí, nastává růst vláken mnohem dříve, než v předešlém, když jsou látky konzumovány. Vlákna také rostou všechna současně. Produkující buňky nemají takový význam pro produkci  $A_1$  a  $A_2$ . Tyto látky jsou převážně vytvářeny koncovými buňkami rostoucích vláken. Vlákna rostoucí z jedné strany směřují do středu (těžiště) skupiny protichůdných koncových buněk. Také dochází ke spojení některých vláken rostoucích stejným směrem. Naproti tomu však nepozorujeme "těžkosti" s napojováním proti sobě rostoucích vláken.

Obecně můžeme pozorovat, že pohlcování látek buňkami působí proti přibližování a spojování vláken rostoucích stejným směrem. Pokud jsou si vlákna (jejich rostoucí konce) blízko, je koncentrace navigační látky mezi nimi snižována a růst je směřován "od sebe". Pohlcování látek také prodlužuje dobu, než začnou vlákna růst a během které jsou navigační látky vytvářeny téměř pouze produkujícími buňkami, což přispívá ke zvýšení lokálnosti navigace zprostředkované vylučováním příslušných látek rostoucími konci vláken.

# 9 Závěr

## 9.1 Tématický závěr

Zkoumali jsme dva druhy situací. Vznik odlišných (diferencovaných) buněk v homogenní tkáni a růst vlásečnic.

V prvních modelech jsme mohli vidět iniciaci a lokalizaci diferenciaci jako důsledek difúze látky, která je produkována v homogenní tkáni. Dále jsme navrhli a teoreticky ověřili možný způsob, kterým si diferencované buňky udrží svoji odlišnost a zároveň zamezí diferenciaci jiných buněk. Ukazuje se, že počet vznikajících ostrovů se dá ovlivnit změnou hodnot jednoho parametru na úrovni produkce látek buňkou. Určitou kvalitu (změnu kvality) v živých organismech na úrovni pozorovatelné pouhým okem by mohlo být možné vysvětlit jako důsledek kvantitativních změn na úrovni chování buňky, konkrétně intenzity produkce nějaké látky. Dále jsme mohli pozorovat dvojrozměrné uspořádání buněk vůči jednorozměrnému v určitém smyslu jako "méně náročné" na diferenciaci, na udržení odlišnosti změnivších se buněk oproti ostatním.

Ve druhé skupině situací jsme postupně dospěli k modelu růstu vlásečnic. Vytvořili jsme autonomní buňku, určitého "agenta", který je ve vhodných podmínkách schopen vytvářet vlákna rostoucí ve směru největší koncentrace určité látky. Ukázalo se, že na navigaci vláken má nemalý vliv pohlcování okolních látek buňkami. Jsou-li látky pohlcovány, je růst vlákna přesnější v "těsných prostorech", nebo v místech, kde se výrazněji mění koncentrace navigující látky. V opačném případě zase buňky přesněji rostou na "otevřených prostranstvích" a tam, kde se koncentrace oné látky příliš nemění. Mírně modifikované autonomní buňky jsme použili k modelu růstu protichůdných vzájemně se napojujících vláken, simulujícího vznik sítě vlásečnic. Mohli jsme vidět, že pohlcování látek napomáhá zamezování přílišnému přibližování a srůstání souhlasně rostoucích vláken a ke zlepšení lokální navigace konců protichůdných vláken na sebe.

## 9.2 Metodický závěr

Z obecnějšího hlediska nám obě modelované skupiny situací (zvláště druhá) dávají tušit, že v určitém smyslu není velký principiální rozdíl mezi buňkou a buňkami (popřípadě jejich strukturou). Jedná se vlastně o tentýž "program" v různých fázích jeho běhu (běhu jeho částí) na základě různých počátečních podmínek. Jedna buňka v sobě nese celý program a výsledné pozorované situace jsou důsledkem běhu či průběhem samotným "částí programu" na určitých vstupních datech. Takovýto úsudek bychom jistě mohli učinit i na základě zkoumání buněk jen "klasickými" způsoby. Simulační program nám ovšem dává možnost virtualizace a prezentace tohoto principu v řeči našeho modelu. Ba co víc, dostáváme do rukou nástroj pro psaní a testování (ladění) vlastních programů implementujících možné chování buněk. Pokud jsme se na náš program dívali jako na virtuální laboratoř, ve které můžeme provádět experimenty, můžeme ho ještě navíc s trochou nadsázky chápat jako nástroj

pro podporu "reverse engineering" programu živých buněk. Vymyslíme a "odladíme" náš virtuální buněčný program a poté jeho běh a výsledky můžeme srovnávat s "reálnými" pozorováními.

Jedním z klasických přístupů k chápání biologické reality je snaha určit, co má ten který gen nebo protein na starosti, jeho funkci. Pokud bude onen gen vypnut, či protein nepřítomen, "výsledný" organismus bude mít ty a ty vlastnosti či defekty. Náš model nahlíží na situaci s širším kontextem a na jiné úrovni základních pojmů. V jeho pojetí je vyrábění určité látky, či látek obecně, jen jedním z parametrů. Důležitou roli hraje například difúze, tvary oblastí, ve kterých probíhá, a další věci. Výsledek tedy není dán pouze "biologickými" faktory, ale souhrnem těchto s více ostatními, které je třeba zahrnout a brát v každém okamžiku v úvahu jejich interakce. Metaforicky bychom mohli říci, že buněčný program sice je napsán biologickým jazykem, ale triky a metody, které používá, jsou časo ze zcela jiného soudku. Pro odhadování programu je tedy důležité zvážit, které "soudky" ještě přicházejí v úvahu a do modelu je s tím "biologickým" rovnoprávně zahrnout. Vhodnost, a vzhledem k dostupným technickým prostředkům dost možná i nezbytnost simulace, jako jednoho z nástrojů takového myšlení, je dosti zřejmá.

### **9.3 Možná rozšíření**

Co se týče modelů diferenciací, mohlo by být zajímavé detailněji analyzovat koncentraci látky způsobující prvotní přepnutí buněk v závislosti na poloze, konkrétně v čase kolem přepnutí první buňky; zkoumat "ostrost" maxima, které u ní způsobí změnu a pokusit se navrhnout způsoby "řízení polohy" ostatních diferencovaných buněk.

U modelu růstu vlasečnic je možno zkoumat vliv různých překážek na konečný tvar sítě i průběh jejího vytváření, což by mohlo přinést další osvětlení způsobu její tvorby. Jiným tématem by mohlo být modelování vláken, která mají za určitých podmínek potenciál "vypustit odnož", i když už jsou "hotová" a simulovat tak adaptivní rozšiřování sítě dle potřeby.



# Dodatek A – Seznam parametrů

*Num vaxes:*

$Num\ vaxes \in \mathbb{N}$ , udává počet vaxů, tedy velikost množiny **A** (**V**).

*Afinita:*

$Afinita \in \mathbb{R}^n$ ,  $n = |\mathbf{A}|$ ,  $Afinita_i \geq 0$ ,  $i = 1, \dots, n$ . Parametr udává afinitu chemických reakcí mezi dvojicemi látek z množin **A** a **V**, které jsou spolu ve vaxu.

*Kapacita membrány:*

$Kapacita\ membrány \in \mathbb{R}^+$ . Udává maximální množství všech receptorů, které se mohou nacházet na membráně buňky. Platí pro všechny buňky v modelové situaci.

*MinGradient:*

$MinGradient \in \mathbb{R}$ ,  $MinGradient \geq 0$ . Udává dolní mez pro vnímání gradientu látky buňkou v jejím okolí. Gradient látky kumulačního stavu pro určení rychlosti buňky, respektive směru jejího dělení je brán v potaz jen pokud jeho velikost překročí *MinGradient*. Parametr platí pro všechny buňky v modelové situaci.

*Odkrytá políčka:*

Tento parametr není v předchozím textu zmíněn. Udává způsob, kterým se určí koncentrace látek v políčku právě uvolněném po jednotkovém pohybu pevného objektu v případě, že je do něj definován přítok látek v obtékání, ale žádný z přítoků se nemůže uplatnit. (Zdrojová sousední políčka jsou obsazena, nebo mimo mřížku.) Může nabývat následujících hodnot:

- Vyprázdnit: Koncentrace všech látek v těchto políčkách budou nulové.
- Průměr okolí: Koncentrace každé látky se určí jako průměr koncentrací této látky v sousedních políčkách pevného objektu po provedení jednotkového pohybu, která jsou neobsazená a nebyla právě odkryta díky jednotkovému pohybu.
- Nechat: Koncentrace látek v těchto políčkách budou takové, jaké byly před (posledním) obsazením tohoto políčka naším pevným objektem v případě, že je pro ně ubývání nulové. Jinak je "množství látek na těchto zakrytých políčkách" snižováno ubýváním stejným způsobem, jako v neobsazených políčkách.

*Pevné okraje situace:*

*Pevné okraje situace*  $\in \{ true, false \}$ . Pokud má hodnotu *true*, nemohou látky unikat difúzí ven z mřížky přes její okraje. Jestliže má hodnotu *false*, pak se vzhledem k difúzi předpokládá, že krajní políčka mřížky mají všechny sousedy, jako vnitřní políčka, přičemž fiktivní sousedé nejsou obsazeni pevným objektem a koncentrace všech látek v nich je v každém okamžiku pokládána za nulovou. Látky tedy mohou difúzí unikat ven z mřížky.

*Pohlčení nachytaných látek:*

*Pohlčení nachytaných látek*  $\in \langle 0, 1 \rangle$ . Pro každou látku látkového vektoru udává její relativní množství, které buňka skonzumuje, vůči množství této látky, které se nachytalo na receptory buňky. (Nachytané látky se konzumují pouze v případě, že nejsou samy receptory jiné buňky.) Parametr je pro všechny vaxy i buňky v situaci společný.

*Rychlost difúze:*

*Rychlost difúze*  $\in \langle 0, 1 \rangle$ . Udává rychlost difúze, stejnou pro všechny látky v modelové situaci. Na základě tohoto parametru se určí parametr *D* z kapitoly Difúze:  $D = \text{Rychlost difúze} * 0,2$ . 0,2 je převrácená hodnota největšího možného počtu sousedů políčka zvýšeného o 1.

*Směr dělení:*

Udává způsob určení preferovaného směru pro dělení buňky. Může nabývat hodnot:

- Dle gradientu: Preferovaný směr dělení je stejný jako směr gradientu koncentrace látky kumulačního stavu, pokud jeho velikost přesáhne *MinGradient*. Jinak je nenulový preferovaný směr vybrán náhodně.
- Kolmo na gradient: Preferovaný směr dělení je kolmý na směr gradientu koncentrace látky kumulačního stavu, pokud jeho velikost přesáhne *MinGradient*. Jinak je nenulový preferovaný směr vybrán náhodně.
- Náhodně: Nenulový preferovaný směr dělení je vybrán náhodně.

*Threshold:*

*Threshold*  $\in \mathbb{R}^+$ . Udává množství všech receptorů, při jehož dosažení nebo přesažení se buňka chce dělit. Musí platit: *Threshold*  $\leq$  *Kapacita membrány*. Tento parametr je společný pro všechny buňky v modelové situaci.

*Ubývání*:

*Ubývání*  $\in \langle 0, 1 \rangle^n$ ,  $n = 3|A|$ . Pro každou látku  $z \mathbf{L} \cup \mathbf{X}$  udává relativní množství, které jí ubude (pokud není ve formě receptoru) po uplynutí jednoho časového okamžiku (kroku simulace).

*Preferováno nasedání*:

*Preferováno nasedání*  $\in \{ true, false \}$ . Pokud má hodnotu *true*, je v rámci daného směru dělení vybírána taková dvojice dceřinných pozic, aby se buňky dotýkaly. Z takových dvojic jsou vybírány ty, kde střed dvojice je co nejbližší středu mateřské buňky.

*Požadováno nasedání*:

Rovněž nabývá hodnot z  $\{ true, false \}$ . Pokud je *Požadováno nasedání* rovno *true*, pak jsou vybírány takové dvojice pozic, aby počet membránových elementů, kterými se dceřinné buňky dotýkají, byl mezi *Minimum společných elementů* a *Maximum společných elementů* i za cenu odchýlení se od směru dělení. Pokud není nalezena dvojice splňující tyto podmínky, buňka se nemůže rozdělit.

*Minimum společných elementů*:

Parametr je z  $\mathbb{N}$ . Udává nejmenší počet membránových elementů, kterými se musejí dotýkat dceřinné buňky po rozdělení mateřské, pokud má *Požadováno nasedání* hodnotu *true*.

*Maximum společných elementů*:

Parametr je z  $\mathbb{N}$ . Udává největší počet membránových elementů, kterými se musejí dotýkat dceřinné buňky po rozdělení mateřské, pokud má *Požadováno nasedání* hodnotu *true*.

# Dodatek B - Uživatelská dokumentace

Na program se můžeme dívat jako na virtuální laboratoř, ve které se dají provádět pokusy. Nejprve je potřeba, abychom si pokus připravili. Potom ho provedeme, průběh simuluje počítač. My při tom můžeme pokus sledovat. Pokud chceme něco změnit, dá se průběh pokusu zastavit, modifikovat současnou situaci, a pak zase pokračovat. Naše laboratoř je vybavena "skříněmi", do kterých je možno ukládat si již dříve postavené, nebo právě rozestavěné "aparatury", přesněji pokusy v nějaké fázi jejich průběhu. Také si do nich můžeme uschovat námi vytvořené jednodušší elementy, které pak vystupují jako součásti našich experimentů.

V následujících odstavcích si nejprve stručně přiblížíme metodiku přípravy, řízení a sledování pokusů, jakož i používání "skříní" – v programu jim říkáme zásobárny. Pak bude následovat popis jednotlivých částí programu.

## B.1 Příprava pokusu

### B.1.1 Tvar těles

Nejprve je třeba definovat tvary pro buňky (statická tělesa, pohyblivá tělesa), které se budou ve virtuálním pokusu nacházet. K tomu slouží okno **Editor těles**. Otevře se z menu hlavního okna volbou **Okna | Editor těles**, pokud tato není zaškrtnutá. (Pokud je zaškrtnutá, okno bychom zavřeli.) Jestliže je okno otevřené, ale není vidět, vyzdvihneme jej na povrch volbou **Okna | Vyzvednout okno | Editor těles**. Do zobrazené mřížky kreslíme myší tvar budoucího pevného objektu. Pokud není mřížka vidět, můžeme její zobrazování zapnout volbou **Mřížka | Zobrazit mřížku** z menu tohoto okna. Levým tlačítkem se označí políčko, pravým se označení zruší. Na pozici tvaru v mřížce nezáleží. Nejlevější označené políčko (políčka) bude mít x-ovou souřadnici tvaru rovnou nule, obdobně nejhornější políčko (políčka) bude mít y-ovou souřadnici rovnou nule. Po zadání tvaru je ho třeba uložit do zásobárny pro pevné objekty. Toho dosáhneme výběrem **Zásobárna | Uložit do zásobárny...**. Objeví se dialog, do kterého se zadává jméno, případně popis.

### B.1.2 Obtékání těles

Pro tvar (tvary) pevných objektů je dále třeba zadat obtékání. K tomu slouží okno **Obtékání tělesa**. Otevřeme ho z menu hlavního okna pomocí **Okna | Obtékání tělesa**, nebo ho vyzdvihneme výběrem **Okna | Vyzvednout okno | Obtékání tělesa**, analogicky jako tomu bylo u **Editoru těles**. Nyní nahrajeme náš tvar zvolením **Zásobárna | Nahrát těleso**. Velikost okolí, na kterém budeme definovat obtékání, nastavíme pomocí **Okolí | Nastavit velikost**. V objevivším se dialogu zadáme šířky pruhů políček nad a pod, popřípadě napravo a nalevo od tvaru. Vhodnou

velikost zobrazování mřížky volíme pomocí **Mřížka | Zvětšit a Mřížka | Zmenšit**. Zelený rámeček nám označuje aktivní políčko. Aktivní políčko se mění výběrem myši. Vektor obtékání pro aktivní políčko můžeme editovat:

- Zadáváním jeho složek v kartézských ( $y$  – osa roste dolů) nebo polárních (úhel roste ve směru hodinových ručiček) souřadnicích v rámečku **Vektor proudění**. Vždy je třeba stisknout příslušné tlačítko **Použít**.
- Myši (toto je třeba zapnout volbou **Ovládání | Editovat vektor pravým tlačítkem**). Stisknutím a držením pravého tlačítka editujeme vektor tažením, uvolněním pravého tlačítka editaci ukončíme.

K urychlení editace je možno použít schránku vektorů políček, která je přístupná z kontextového menu (volba **Ovládání | Editovat vektor pravým tlačítkem** musí být vypnutá). Směr jednotkového pohybu objektu, pro který právě definujeme obtékání, udává stisknuté tlačítko s odpovídající zelenou šipkou vpravo nahoře v rámečku **Pohyb tělesa**. Stisknutím tlačítka s jinou šipkou můžeme daný směr jednotkového pohybu změnit. (Toto lze také z menu **Pohyb tělesa**). Pro urychlení editace obtékání pro jednotlivé směry jednotkového pohybu můžeme použít schránku mřížky, přístupnou z menu **Úpravy**. Pokud je obtékání tělesa pro některé směry jednotkového pohybu stejné, jen analogicky otočené, je ve spojení se schránkou možno užít **Úpravy | Otočit vektorovou mřížku**. Po jednom zvolení této položky se mřížka i vektory v ní (né však objekt – tvar) otočí o 90 stupňů ve směru pohybu hodinových ručiček. Pokud je nám milejší představovat si obtékání tak, že pevný objekt stojí a okolní kapalina se pohybuje kolem něj, můžeme před započítím editace pro daný směr zvolit **Úpravy | Odečíst jednotkový směr**. Od vektorů ve všech políčkách se odečte jednotkový vektor současného směru. Poté zadáváme vektory, jako by proudilo okolí. Na konci zvolíme **Úpravy | Přičíst jednotkový směr**. Když máme obtékání pro všechny směry zadané, je třeba daný tvar s tímto obtékáním uložit do zásobárny. K tomu slouží položka **Zásobárna | Uložit těleso**. Dialog nám nabídne původní jméno. Pokud chceme objekt s naším obtékáním uložit pod oním jménem, bude třeba původní objekt tohoto jména přepsat, na což budeme dotázáni.

### B.1.3 Látkový vektor

Ještě než budeme vytvářet modelovou situaci, je vhodné zvolit počet látek látkového vektoru, pokud se má lišit od současného. K tomu je třeba otevřít okno **Editor látkového vektoru** z menu **Okna** hlavního okna. Nový počet dvojic látek, které jsou spolu ve vaxu, zadáme pomocí **Látkový vektor | Nový**.

### B.1.4 Vkládání objektů

Nyní již můžeme vkládat do modelové situace v hlavním okně pevné objekty, které budou mít některý z námi definovaných tvarů a při pohybu budou obtékány, jak jsme to určili. Typ pevného objektu, který vznikne po vložení, (buňka, statické těleso, pohyblivé těleso) volíme v dialogu vyvolaném pomocí **Objekt | Typ**

**vkládáního objektu...** Samotné vkládání začneme volbou **Objekt | Nahrát ze zásobárny**. Objekt pak umístíme pomocí myši (levé tlačítko). Vkládání můžeme zrušit stisknutím Esc. Vložené těleso je možno opětovně zvednout levým tlačítkem (pokud neběží simulace, viz níže) a měnit jeho pozici jako při vkládání. (Esc – těleso se umístí na původní pozici, Del – těleso se vymaže.)

### B.1.5 O uživatelském rozhraní

Než budeme pokračovat, učiňme malou vsuvku ohledně hlavního okna. To může sestávat z různého počtu najednou viditelných a uživatelem uspořádaných panelů. Poskytuje nám podporu, abychom mohli alespoň do určité míry sledovat zvolenou množinu údajů najednou. Panely se přidávají dělením stávajících panelů, odstraňují se vypouštěním.

- Dělení panelu: V kontextovém menu pro daný panel zvolíme **Rozdělit**. Objeví se dialog, ve kterém vybereme požadovaný panel, určíme směr dělení a pozici pro původní panel.
- Spojování panelů: Z kontextového menu lišty oddělující panely, respektive bloky panelů, vybereme **Spojit | VypustitXXX**, kde XXX značí pozici panelu / bloku, který chceme vypustit.

Rozložení panelů můžeme dále upravovat tažením oddělujících lišt. (Lištu uchopíme a držíme levým tlačítkem.)

### B.1.6 Látky

Poté, co jsme vložili naše pevné objekty do situace, můžeme nastavit koncentrace jednotlivých látek látkového vektoru v políčkách. K tomu je třeba, aby se v hlavním okně zobrazoval panel **Látkový vektor oblastí**, dále je vhodné aby se zobrazoval panel **Zobrazování látek** a v něm bylo zaškrtnuto **Zobrazovat** pro zapnutí barevného zobrazování látek, které nás zajímají, v mřížce. Políčko, nebo obdélníkovou oblast políček, ve které chceme měnit koncentrace látek, vybereme myší - Ctrl + levé tlačítko. (Říkejme této oblasti vybraná oblast.) Poté klepneme v panelu **Látkový vektor oblastí** na místo vedle svislé čáry s barvou látky, jejíž koncentraci chceme měnit. Objeví se místní editor, do kterého zadáme požadovanou koncentraci. (Po zadání hodnoty je třeba stisknout Enter, zrušení editace – Esc.) Všechna políčka vybrané oblasti budou obsahovat danou látku v této koncentraci. Vybrání nějaké oblasti můžeme zrušit vybráním jiné, nebo pomocí Alt + levé tlačítko kdekoli v mřížce.

### B.1.7 Receptory

Dále je možné nastavit (počáteční) množství receptorů jednotlivých buněk. Je třeba, aby byl zobrazován panel **Receptory vybrané buňky**. Kýženou buňku musíme nejprve vybrat pomocí Shift + levé tlačítko. (Říkejme obecně objektu, který jsme takto vybrali, vybrané těleso.) Množství jejich receptorů na jednotlivé látky

nastavíme v panelu **Receptory vybrané buňky** analogickým způsobem, jako jsme nastavovali koncentrace látek ve vybrané oblasti. Látka v tomto panelu označuje tu, která se chytá na receptory, jejichž množství editujeme.

### B.1.8 Zygotický graf

Než spustíme simulaci našeho pokusu, je třeba zadat zygotický graf. K tomu potřebujeme otevřít okno **Zygotický graf** z menu **Okno** hlavního okna. V něm zadáváme produkční, respektive receptorový graf maticí sousednosti. Řádka značí látku kumulačního stavu (odkud hrana vychází), sloupec látku, která se produkuje / na níž se přidávají receptory<sup>7</sup> (kam hrana vede). Rychlosti pro pohybový graf se zadávají na diagonálu dané matice. Hodnoty zadané jinde se ignorují. Zygotický graf nepůjde editovat, pokud není vypnutá simulace (viz níže).

### B.1.9 Parametry

Co se týče parametrů modelové situace (pro jejich editaci musí být vypnutá simulace):

- *Afinita* se nastavuje v okně **Editor látkového vektoru** pod každou dvojicí komplementárních látek. (Editace – kliknout, Enter – potvrdit, Esc – zrušit.)
- *Odkrytá políčka, Pevné okraje situace, Pohlcení nachytných látek, Rychlost difúze, Směr dělení, Preferováno nasedání, Požadováno nasedání, Minimum společných elementů a Maximum společných elementů* se nastavují v dialogu vyvolaném z menu hlavního okna **Možnosti | Modelová situace...**
- *Kapacita membrány, MinGradient* a *Treshold* se nastavují v dialogu zobrazeném pomocí menu **Atributy** okna **Zygotický graf**.
- *Ubývání* se nastavuje v okně **Samovolné ubývání látek**, které se otevře z menu **Okna** z hlavního okna.

### B.1.10 Spuštění pokusu

Ke spuštění simulace je vhodné, aby byl v hlavním okně zobrazován panel **Simulace**. Náš virtuální pokus spustíme tlačítkem "Play". Opětovným stisknutím tohoto tlačítka simulaci pozastavíme. Pozastavenou simulaci můžeme vrátit do počátečního stavu tlačítkem "Rew", zastavíme ji tlačítkem "Stop". Pokud simulace běží, nebo je pozastavená, nelze měnit některá nastavení a parametry. Simulaci lze také ovládat z menu hlavního okna **Simulace**.

---

<sup>7</sup> Zda sloupec značí receptory na látku, nebo receptory typu látky, se volí v menu **Atributy | Receptorový graf**, viz níže.

## B.2 Sledování / modifikace pokusu

Nejprve ohledně modifikace pokusu. Pokud jednou spustíme simulaci, nelze situaci modifikovat v pravém slova smyslu. Je nutné simulaci (pokus) zastavit, pozměnit výsledek, a opět ji spustit z této nové výchozí situace.

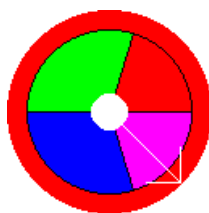
### B.2.1 Vizualní sledování

Průběh pokusu můžeme sledovat vizuálně v mřížce. Barva v neobsazeném políčku je výsledkem subtraktivního složení barev, které odpovídají koncentracím těch látek látkového vektoru, pro které je zapnuto zobrazování v panelu **Zobrazování látek**. Určení barev odpovídajících koncentracím se pro jednotlivé látky edituje v sekcích panelu **Zobrazování látek**. Barva se určí z koncentrace s použitím parametrů  $C0$  a  $C1$  následovně:

- $C0$  udává dolní okraj intervalu koncentrací "převáděného na barvy". Koncentrace  $c \leq C0$  odpovídá pro danou látku bílé barvě.
- $C1$  udává horní okraj intervalu koncentrací "převáděného na barvy". Koncentrace  $c \geq C1$  odpovídá pro danou látku její barvě (barvě levého pruhu dané sekce panelu).
- Koncentrace  $c \in (C0, C1)$  se převede na barvu  $b$  na úsečce mezi bílou (označme ji  $B0$ ) a barvou látky (označme  $B1$ ):

$$b = B0 + (B1 - B0) \cdot \frac{c - C0}{C1 - C0} . \quad (\text{B.1})$$

Buňky na sobě mohou mít koláčový graf.



Obrázek 24:  
Koláčový graf  
buňky

Označme si množství všech receptorů buňky  $AllRecept$ , množství všech jejích receptorů na  $i$  - tou látku  $Recept_i$ , celkové množství látek, které se nachytalo na všechny receptory  $AllSeen$  a množství  $i$  - té látky nachytané na všechny receptory  $Seen_i$ . Části koláčového grafu mají následující význam:

- Vnější mezikruží má barvu látky, na kterou je buňka v kumulačním stavu.
- Ve středu koláčového grafu se zobrazí bílý kruh právě když  $AllRecept \geq Threshold$ .



- Význam šipky a výšečí se zadává v dialogu **Prostředí** ve stránce **Buňky/Látky**, který se zobrazí výběrem **Možnosti | Prostředí...** z menu hlavního okna.

Označme si relativní velikost výšeče s barvou  $i$  - té látky  $V_i$ , relativní velikost černé výšeče  $Rest$  a předpokládejme, že žádná látka nemá černou barvu.

Pokud je vybráno **Zobrazovat receptory**:

Je-li  $AllRecept < Threshold$ , potom

$$V_i = \frac{Recept_i}{Threshold} , \quad (B.2)$$

$$Rest = 1 - \frac{AllRecept}{Threshold} . \quad (B.3)$$

Jestliže  $AllRecept \geq Threshold$ , pak

$$V_i = \frac{Recept_i}{AllRecept} . \quad (B.4)$$

Šipka označuje směr (né však velikost) gradientu látky, na kterou je buňka v kumulačním stavu.

Pokud je vybráno **Zobrazovat nachytáno**:

$$V_i = \frac{Seen_i}{AllSeen} . \quad (B.5)$$

Šipka značí směr (né však velikost) rychlosti buňky.

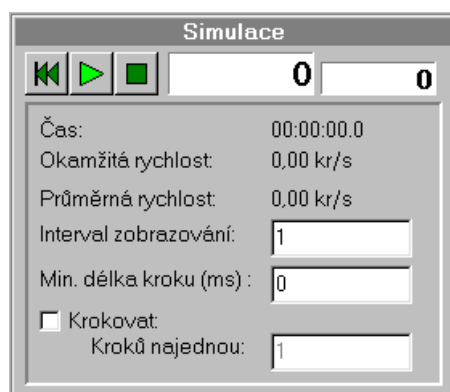
## B.2.2 Sledování hodnot

Vedle vizuálního pozorování pokusu můžeme sledovat údaje o vybraném tělese a vybrané oblasti. Průměrná koncentrace každé látky v políčkách vybrané oblasti je zobrazována v panelu **Látkový vektor oblastí**. Je-li vybraný objekt buňka, jsou množství jejích receptorů na jednotlivé látky látkového vektoru zobrazována v panelu **Receptory vybrané buňky (na)**. Množství jednotlivých látek nachytaná na receptory jsou zobrazována v panelu **Vybraná buňka vidí**, jsou vždy jen pro čtení, nelze je modifikovat. Údaje o vybraném pevném tělese jsou zobrazovány v panelu **Inspektor**. Vždy je zobrazen název typu tělesa, dále jeho souřadnice jako desetinná čísla. Pro pohyblivé těleso je navíc zobrazována rychlost (složky jako desetinná čísla),

kteřá se dá modifikovat. Pro buňku je zobrazována rychlost, kumulační stav a koncentrační gradient látky kumulačního stavu. Modifikovat se zde dá pouze kumulační stav.

## B.3 Řízení simulace

Simulace se ovládá z panelu **Simulace**. Povelý pro řízení jejího běhu je také možné zadávat z menu hlavního okna **Simulace**.



Obrázek 25: Panel Simulace

Základní ovládání již bylo popsáno výše. Větší počítadlo v panelu **Simulace** udává současný (právě počítaný) krok simulace. Na menším je krok simulace právě zobrazovaný v mřížce hlavního okna.

- **Čas** udává čas uběhlý od počátku simulace.
- **Okamžitá rychlost** udává okamžitou rychlost výpočtu simulace v krocích za sekundu.
- **Min. délka kroku** udává nejmenší povolený čas, který smí trvat jeden krok simulace.
- **Průměrná rychlost** udává průměrnou rychlost výpočtu simulace v krocích za sekundu.
- **Interval zobrazování** udává, po kolika krocích se vykreslí spočítaná situace. Pokud je roven 0, situace se nevykresluje, až při pozastavení nebo zastavení simulace.
- **Min. délka kroku** udává nejmenší povolený čas, který smí trvat jeden krok simulace.
- Volbou **Krokovat** se zapíná a vypíná krokování simulace.
- Položka **Kroků najednou** udává, po kolika krocích se simulace automaticky pozastaví. Položka se bere v úvahu jen když je zapnuto krokování.

Intervaly, ve kterých se jednotlivé údaje o průběhu simulace při jejím výpočtu aktualizují, se zadávají v dialogu **Prostředí** ve stránce **Simulace**. (Vyvolá se z hlavního okna výběrem **Možnosti | Prostředí...**.) Tyto intervaly je možné zadat v čase nebo v krocích simulace. Čas je však vzorkován s frekvencí jednoho kroku.

## B.4 Zásobárny

Zásobárny slouží k (persistentnímu) ukládání virtuálních pokusů a některých "prefabrikátů", s jejichž použitím se pokusy konstruují. V programu jsou následující zásobárny:

- **Zásobárna pevných objektů.** S ní jsme se již setkali v odstavci Příprava pokusu. Slouží k ukládání tvarů s obtékáním pro budoucí pevné objekty v mřížce. Je přístupná z **Editoru těles**, z okna **Obtékání tělesa** a z hlavního okna ( volba **Objekt | Nahrát ze zásobárny...** ).
- **Zásobárna situací.** Do ní je možno ukládat celé pokusy. (Vše, čím je pokus určen.) Je přístupná z menu **Situace** hlavního okna.
- **Zásobárna zygotických grafů.** Do ní se dá uložit samotný zygotický graf. Společně s ním se také uloží vše, co se zadává v dialogu **Atributy zygotického grafu** (zobrazí se pomocí volby **Atributy** z okna **Zygotický graf**). Tato zásobárna je přístupná z okna **Zygotický graf**.

Zásobárny můžeme administrovat z dialogu **Práce se zásobárnou: XXX**, kde XXX značí danou zásobárnu. Je možné mazat jejich prvky, persistentně je ukládat, nastavovat, zda se jejich obsah má nahrávat při spuštění programu a ukládat při jeho konci.

## B.5 Části programu

### B.5.1 Hlavní okno

Hlavní okno slouží zejména k editaci, provádění a sledování virtuálního pokusu. Sestává se z proměnného počtu uspořádatelných panelů.

#### Kontextové menu panelu:

- **Rozdělit:** Rozdělení daného panelu, objeví se dialog k upřesnění.

#### Kontextové menu dělicí lišty:

- **Spojit | Vypustit XXX:** Panel (skupina panelů) vedle lišty ve směru XXX bude vypuštěn.
- **Pevná velikost | Zafixovat XXX:** Velikost panelu(ů) vedle lišty ve směru XXX (ne)půjde ve směru kolmém na lištu měnit.

#### Panely:

- **Receptory vybrané buňky (na):** Slouží ke sledování a editaci množství receptorů vybrané buňky na jednotlivé látky látkového vektoru. Svislý pruh na levé straně sekce pro danou látku má její barvu. Číslo napravo od něj udává množství receptorů (editace – klepnutím), pole s číslem má barvu, která odpovídá "koncentraci" látky vyjádřené číslem.

Receptory vybrané buňky (na)	
3	1
A1	A2
2	4
V1	V2

Obrázek 26: Panel Receptory vybrané buňky (na)

- **Látkový vektor oblastí:** Zobrazuje průměrnou koncentraci látek ve vybrané oblasti. Umožňuje také nastavení koncentrací ve vybrané oblasti. Je analogický panelu **Receptory vybrané buňky (na)**.
- **Vybraná buňka vidí:** Zobrazuje množství jednotlivých látek nachytná na receptory buňky. Je analogický panelu **Receptory vybrané buňky (na)**, neumožňuje editaci.
- **Zobrazování látek:** Slouží k zapínání a vypínání zobrazování koncentrací látek v situaci (zaškrtnutím **zobrazovat**) a k nastavování parametrů  $C0$  a  $C1$  pro převod koncentrace jednotlivých látek na barvu. Pokud je zapnutá funkce  $CMin - CMax$  (přístupná ve stránce **Ostatní dialogu Prostředí**, který se zobrazí volbou **Možnosti | Prostředí**), zobrazují se v jeho sekcích také minimální a maximální koncentrace v situaci pro každou látku. Nastavování  $C0$  a  $C1$ :
  - Levý tahový ovladač slouží k nastavování hodnoty  $C0$ , pravý k nastavování  $C1$ .
  - $C0$  a  $C1$  lze také editovat přímo klepnutím na údaj s jejich hodnotou. (Konec editace - Enter/Esc).
  - Číslo nad tahovými ovladači udává hodnotu odpovídající jejich poloze nahoře, číslo pod tahovými ovladači udává udává hodnotu odpovídající jejich poloze dole (tuto lze editovat klepnutím). Při změně dolní hodnoty uživatelem nemusejí polohy ovladačů odpovídat hodnotám  $C0$  a  $C1$ .
  - Tlačítka se šipkami dole slouží ke změně hodnoty odpovídající dolní poloze tahových ovladačů na polovinu / dvojnásobek.



Obrázek 27: Sekce panelu Zobrazování látek

- **Inspektor:** Používá se k zobrazování a editaci údajů o vybraném tělese.

Inspektor	
Atribut	Hodnota
Typ	Buňka
Pozice X	30
Pozice Y	16
Rychlost X	0
Rychlost Y	0
Ks	 A1
Gradient	(0; 0)

Obrázek 28: Panel Inspektor

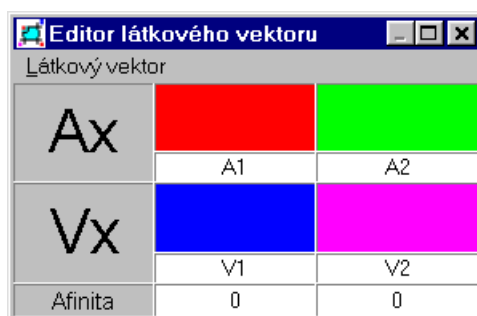
- **Simulace:** Slouží k řízení simulace. Byl popsán v odstavci Řízení simulace.
- **Nahrát objekt ze zásobárny, Nahrát situaci ze zásobárny, Uložit situaci do zásobárny:** Tyto panely jsou obdobou modálních dialogů pro zásobárnu situací a nahrávání objektu.

### Menu (stručný přehled):

- **Situace:** Vytváření nového pokusu (**Nová...**), ukládání (a nahrávání) pokusů do zásobárny a do souboru, administrace zásobárny pokusů (**Zásobárna situací...**).
- **Objekt:** Nahrávání pevného objektu ze zásobárny, určení jeho typu, extrakce tvaru a obtékání vybraného objektu do **Zásobárny pevných objektů**.
- **Úpravy:** Ovládání schránky pro pevné objekty.
- **Simulace:** Povelů pro základní ovládání simulace (Start, Pause, Stop, Rewind), duplicitní k tlačítkům panelu **Simulace**.
- **Pohled:**

- **Zvětšit / Zmenšit:** Velikost políček se zdvojnásobí / zmenší na polovinu. (Logická velikost políčka z pohledu modelu zůstává stále 1.)
- **Zobrazit mřížku:** Zapíná / vypíná kreslení čar mřížky.
- **Možnosti:**
  - **Modelová situace:** Nastavení parametrů modelu *Odkrytá políčka*, *Pevné okraje situace*, *Pohlčení nachytaných látek*, *Rychlost difúze* a *Směr dělení*.
  - **Prostředí:** Objeví se dialog **Prostředí**, ve kterém se nastavují různé volby programu (prostředí). Volby jsou popsány v nápovědě v elektronické formě.
- **Okna:** Zobrazování, skrývání a vyzvedávání ostatních oken programu.

## B.5.2 Okno Editor látkového vektoru

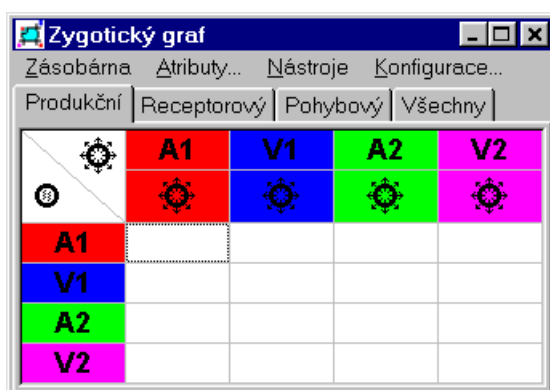


Obrázek 29: Editor látkového vektoru

Toto okno slouží k zadání počtu vaxů, jmen a barev látek a afinit látek v jednotlivých vaxech.

- Vybráním menu **Látkový vektor** | **Nový** se vytvoří nový látkový vektor. Objeví se dialog pro zadání počtu vaxů.
- Klepnutím na barevný obdélníček lze editovat barvu příslušné látky (ligandu).
- Klepnutím na okénko pod barevným obdélníčkem lze editovat jméno dané látky (Jména musejí být jednoznačná).
- Klepnutím na okénko v řádce Afinita lze editovat *Afinitu* látek vaxu v odpovídajícím sloupci.

### B.5.3 Okno Zygotický graf



Obrázek 30: Možný vzhled okna *Zygotický graf*

Okno **Zygotický graf** slouží k editaci produkčního, receptorového a pohybového grafu. Řádka reprezentuje látku kumulačního stavu, sloupec v produkčním grafu produkovanou látku. Sloupec v receptorovém grafu označuje buď látku, na kterou se přidávají receptory, nebo látku, jejíhož typu jsou přidávané receptory, podle toho, jak je to určeno v dialogu **Atributy zygotického grafu** (viz níže). Pro pohybový graf se hodnoty zadávají na diagonálu. (Údaje na jiných místech budou ignorovány.) Políčko se začne editovat klepnutím (pokud je vypnutá simulace).

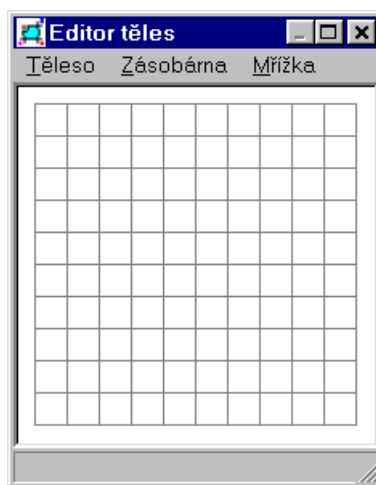
#### Menu (stručný přehled):

- **Zásobárna:** Slouží k nahrávání a ukládání zygotických grafů do **Zásobárny zygotických grafů** a k její administraci.
- **Atributy...:** Objeví se dialog **Atributy zygotického grafu**. V něm se:
  - Zadávají parametry modelu *Kapacita membrány*, *MinGradient* a *Treshold*.
  - V rámečku **Receptorový graf** definuje význam sloupce pro zadávání receptorového grafu – receptory na látku / receptory typu dané látky. Pokud je tento význam změněn, bude k dispozici volba **Provést transformaci**. Pokud je zaškrtnuta, zachová se význam původního grafu. V opačném případě se původním hodnotám v tabulce přiřadí nový význam, tabulka zůstane "opticky" stejná.
- **Nástroje:**
  - **Implicitní hodnoty:** Grafům v aktivní stránce budou přiřazeny implicitní hodnoty. Tj. pro produkční a pohybový graf všechny hodnoty rovny nule, pro receptorový graf všechny hodnoty

rovny nule až na diagonálu, kde budou nastaveny na **Implicitní hodnoty pro diagonálu**, která se zadává v dialogu **Prostředí**, stránce **Ostatní**, rámečku **Editor receptorvého grafu**.

- **Vyčistit:** Všechny hodnoty grafů aktivní stránky budou nastaveny na nulu.
- **Konfigurace...:** Objeví se dialog, ve kterém je možné přidávat a ubírat stránky v našem okně a určovat, které grafy budou na jaké stránce zobrazeny.

#### B.5.4 Okno **Editor těles**



Obrázek 31: Editor těles

Okno **Editor těles** se používá k editaci tvaru pro budoucí pevné objekty v modelové situaci. Nálezení políčka k tělesu se označí levým tlačítkem, zruší se pravým.

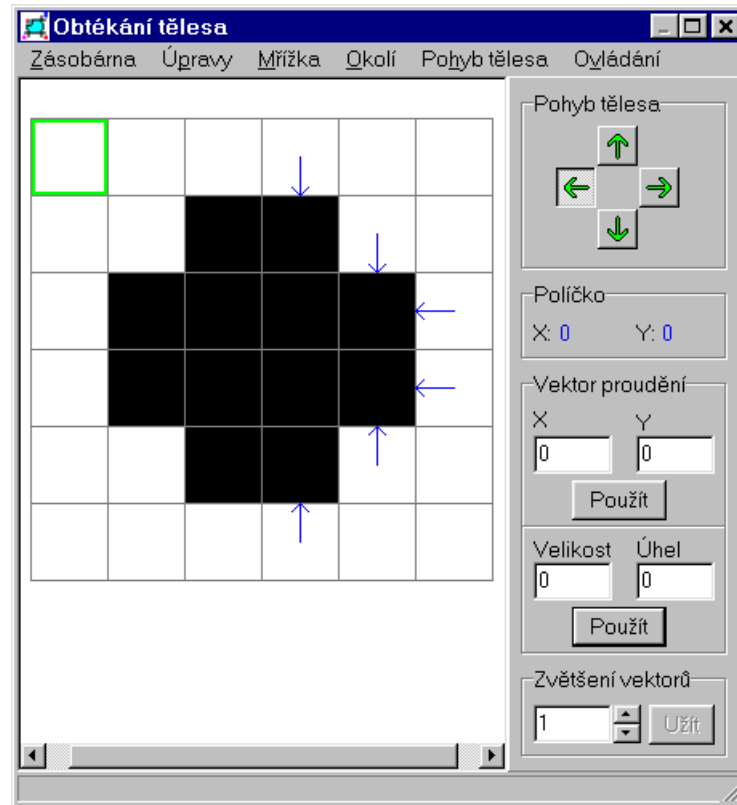
##### **Menu (stručný přehled):**

- **Těleso:**
  - **Nové:** Zruší označení všech políček v mřížce.
- **Zásobárna:** Nahrávání a ukládání tvarů do **zásobárny pevných objektů** a její administrace.
- **Mřížka:**
  - **Nová:** Vytvoření mřížky o nových rozměrech a velikosti zobrazování políček.
  - **Zvětšit / Zmenšit / Zobrazit mřížku:** Velikost zobrazování políček se zdvojnásobí / zmenší na polovinu, / zapínání a vypínání zobrazování linek mřížky.



- **Automatická velikost:** Maximální velikost okna se mění automaticky podle velikosti mřížky.

### B.5.5 Okno **Obtékání tělesa**



Obrázek 32: *Obtékání tělesa*

V tomto okně se definuje obtékání pohybujícího se pevného tělesa okolím pro jednotlivé směry jednotkového pohybu. Informace o obtékání je vždy uchovávána s tělesem. Pokud není pro nějaké těleso obtékání definováno, má toto implicitní obtékání. Okolí pro implicitní obtékání je nejmenší obdélník, který má svislé / vodorovné strany, do kterého se dané těleso vejde. Ve všech neobsazených políčkách pro každý směr je vektor obtékání nulový.

#### **Pravý panel okna:**

- **Pohyb tělesa:** Zde se zadává směr jednotkového pohybu tělesa, pro který právě editujeme obtékání. Tlačítko s odpovídající šipkou je stisknuté. Editovaný směr jednotkového pohybu je možno duplicitně volit z menu **Pohyb tělesa**.
- **Políčko:** Souřadnice vybraného políčka. Počátek je v levém horním rohu.

- **Vektor proudění:** V tomto rámečku se zadává vektor proudění pro vybrané políčko. Je ho možno zadávat v kartézských nebo polárních souřadnicích. Aby se hodnota zadaná do editačního okénka stala platnou, je třeba stisknout příslušné tlačítko **Použít**.
- **Zvětšení vektorů:** Udává zvětšení, se kterým jsou kresleny šipky reprezentující vektory pro daná políčka. Pokud je hodnota vepsána do okénka "ručně", je třeba stisknout tlačítko **Užít**.

### Menu (stručný přehled):

- **Zásobárna:**
  - **Nahrát těleso:** Nahraje těleso ze zásobárny.
  - **Uložit těleso:** Uloží těleso s obtékáním do zásobárny. Po skončení editace obtékání je nutno těleso uložit do zásobárny, nové obtékání bude jen u tohoto uloženého tělesa.
- **Úpravy:** První tři položky tohoto menu slouží k ovládní schránky pro naši mřížku.
  - **Otočit vektorovou mřížku:** Mřížka bude otočena o 90 stupňů ve směru pohybu hodinových ručiček a vektory v jednotlivých políčkách také.
  - **Přičíst jednotkový směr:** K vektoru v každém (neobsazeném) políčku bude přičten vektor velikosti jedna směru současného jednotkového pohybu.
  - **Odečíst jednotkový směr:** Od vektoru v každém (neobsazeném) políčku bude odečten vektor velikosti jedna směru současného jednotkového pohybu.
- **Mřížka:**
  - **Zvětšit / Zmenšit / Zobrazit mřížku:** Velikost zobrazování políček se zdvojnásobí / zmenší na polovinu, / zapínání a vypínání zobrazování linek mřížky.
- **Okolí:**
  - **Nastavit velikost:** Nastavení velikosti okolí pro definování obtékání.
- **Pohyb tělesa:** Určení směru jednotkového pohybu tělesa, pro který je právě editováno obtékání.
- **Ovládání:**
  - **Editovat vektor pravým tlačítkem:** Zapíná / vypíná vizuální editaci vektorů jednotlivých políček tažením myši (pravým tlačítkem).

## B.5.6 Okno **Samovolné ubývání látek**



Obrázek 33: Samovolné ubývání látek

Toto okno slouží k editaci parametru *Ubývání*. Tlačítkem **Použít** se stanou právě provedené změny platnými. Tlačítkem **Zpět** se do editačních okének načtou současně platné hodnoty. Zbývající tlačítka zavřou okno (**Ok** s uplatněním změn, **Zrušit** bez uplatnění změn). Každé jiné zavření okna nezpůsobí uplatnění změn. Pokud je okno otevřené a dojde ke spuštění simulace, provedené změny se automaticky stanou platnými. Pokud není vypnutá simulace, nelze hodnoty editovat.

## B.6 Jak urychlit běh programu

Hlavními faktory, které omezují rychlost výpočtu při běhu programu jsou:

- Vykreslování situace.
- Zobrazování aktualizovaných údajů během simulace.
- Množství buněk v situaci.
- Logická velikost mřížky.

Poslední dva faktory jsou samozřejmě dány vstupem - vlastní modelovou situací, kterou chceme simulovat. Simulace buňky je podstatně výpočetně náročnější, než simulace ostatních druhů pevných objektů. Je proto výhodné si s nimi, pokud je to možné, co nejvíce vypomáhat. Co se týče velikosti mřížky, můžeme si pomoci snížením počtu látek v látkovém vektoru.

### B.6.1 Vykreslování situace

Vykreslování situace je možno urychlit:

- Zapnutím používání rozhraní DirectX. (Z menu hlavního okna výběrem **Možnosti | Prostředí**, stránka **Ostatní**.)
- Zmenšením velikosti políčka v hlavním okně. Pokud je používáno DirectX, tento krok výrazně nepomůže (většina zvětšování mřížky je přenechávána grafickému akceleratoru), ale pokud jsme bez DirectX, je zlepšení znatelné.

- Zmenšením počtu zobrazovaných látek (v panelu **Zobrazování látek**).

Další variantou je zvětšit interval zobrazování (situace) v panelu **Simulace**, popřípadě jej nastavit na 0 – potom se aktuální stav vykreslí až při zastavení nebo pozastavení simulace.

### B.6.2 Zobrazování aktualizovaných údajů během simulace

Množství těchto údajů snížíme, pokud

- V situaci nebude vybraná oblast.
- V situaci nebude žádný objekt vybraný.
- Vypneme funkci CMin – CMax.

Dále je možno zvětšit intervaly překreslování v dialogu **Prostředí** na stránce **Simulace**. Přínosné je také vypnout volbu **Automatická změna velikosti okna** na stránce **Uživatelské rozhraní** v tom samém dialogu.

### B.6.3 Start a ukončení programu

Dobu startu a ukončování programu můžeme urychlit, pokud v zásobárnách, především v **Zásobárně situací** bude málo objektů a tyto budou malé. Méně často používané obsahy zásobáren je možno ukládat do separátních souborů.

# Dodatek C - Programátorské poznámky

Při psaní programu bylo hlavním záměrem vytvořit "virtuální laboratoř" pro "zkoumání" buněk vytvořených podle našeho modelu.

Po této "laboratoři" bylo požadováno:

- Názorné a nepříliš složité vytváření elementů modelu a modelových situací.
- Názornou vizualizaci průběhu simulovaného pokusu a klíčových modelových veličin.
- Možnost přesného sledování požadované množiny modelových veličin.
- Rychlý výpočet simulace a rychlé vykreslování.
- Možnost rychlé a snadné manipulace s často používanými daty.

Funkční jádro programu tvoří tvoří diskretní simulace implementovaná v logickém modulu **Sim**. Editace a řízení zobrazování modelové situace se provádí v části **Area**, která pro vlastní zobrazování používá část **Renderer**. Objekty reprezentující látkový vektor, zygotický graf a parametry pro zobrazování modelové situace mají své editory. Část **Managers** propůjčuje těmto objektům mechanismus, pomocí kterého se jejich "registrovaní" uživatelé informují o změnách jejich obsahu, případně znovuvytvoření.

Program je navržen jako samostatná aplikace běžící v jednom vlákně výpočtu. Je napsán v jazyce C++, konkrétně v prostředí Borland C++ Builder 1 pro platformu Win32. Pro rychlejší vykreslování je podporováno využití rozhraní Microsoft DirectX 7.0, jehož používání je však podporováno pouze na platformě Windows 95/98.

## C.1 Přehled logických částí programu

### C.1.1 SplitTree

V této součásti je naimplementováno uživatelské rozhraní umožňující rozdělovat okno na menší části, do kterých se umísťují uživatelem zvolené panely. Vše zastřešuje třída "TSplitTree". Soustava panelů je reprezentována stromem. Kořen stromu implementuje třída "TRootNPanel", uzly třída "TNPanel".

### C.1.2 Area

Zde je implementován základní koncept modelové situace, její zobrazování, editace

a interakce s ostatními částmi programu. Integrujícím prvkem je zde třída "TArea", která koordinuje reprezentaci modelové situace (třída "TAreaContent"), zobrazování (třídy "TDrawManager, TPixelGrid"), editaci (třída "TAreaEmbedding") a interakci s uživatelem (TAreaSurface).

### C.1.3 AreaObject

AreaObject obsahuje implementaci pevných objektů, které mohou být součástí modelové situace (třída "TAreaObject" a její potomci). Každý takový pevný objekt obsahuje samostatný "prefabrikát" ("TSolid"), ve kterém jsou shrnuty základní vlastnosti pevných objektů. Nejsložitějším pevným objektem je buňka (třída "TBioCell"; deklarována v TAreaObject.h). Membrána buňky, zprostředkávající její látkovou interakci s prostředím, je implementována ve třídě "TMembrane" (deklarované v TAOMembraneMap.h).

### C.1.4 FileIO

Zdejší objekty zprostředkávají pro zbytek programu soubor jako strom vnořených sekcí, jde v podstatě o zjednodušenou analogii k Windows strukturovaným souborům. (Třídy TSection, TBasicSaver.) Na tomto konceptu je implementováno komfortní prostředí pro ukládání základních datový typů (třída "TElemSaver"), nahrávání polymorfních objektů (třída "TObjectLoader") a ukládání větších celků včetně řízení práce se souborem. (Třída "TPersistenceManager".)

### C.1.5 FormExecutives

FormExecutives obsahuje implementaci samostatných funkčních celků realizujících část uživatelského rozhraní některých formulářů.

### C.1.6 Forms

Zde jsou sdruženy všechny formuláře uživatelského rozhraní (až na 1 výjimku, třídu "TSplitDialog"), jakožto i pomocné formuláře, používané pro tvoření větších celků za běhu programu.

### C.1.7 Inspector

Součástí **Inspector** obsahuje soustavu objektů implementujících zobrazování a editaci parametrů uživatelem vybraného pevného tělesa.

### C.1.8 Lv

Lv obsahuje objekty pro reprezentaci látkového vektoru (třída "TLatkovyVektor"), vektoru koncentrací látek (třída "TLvInst") a zobrazování vektoru koncentrací (třídy "TZobratovacLv", "TZobrParms").

### C.1.9 Managers

Managers zahrnuje objekty pro zprostředkování koordinace, komunikace, a některé objekty zajišťující persistenci parametrů větších celků v programu.

### C.1.10 Rendering

V části Rendering jsou objekty implementující rozhraní pro kreslení grafických primitiv, scén, a obecně pro grafický výstup na nižší úrovni. Vstupním místem je třída "TRenderManager". Kreslení je podporováno, dle volby, pomocí objektů nabízených knihovnou VCL, Windows API a pomocí knihovny DirectX.

### C.1.11 Repository

Repository realizuje úschovnu, vlastně volitelně persistentní množinu uživatelem vytvořených a spravovaných objektů v programu. Stěžejní je zde strom tříd zděděných od "TRepository".

### C.1.12 Sim

Zde je implementován základní koncept simulace (třída "TSimulace"), dále koncept simulace volající v každém kroku množinu výkonných modulů, kde každý provádí specifické akce na modelové situaci (třída TCompoundAreaSimulace; deklarována v TSimulace.h). Jednotlivé výkonné moduly jsou deklarovány v souboru TAreaSimExecutive.h. Zvlášť je jen simulátor difúze ("TDifuzeSimExecutive"). Modul simulující buněčné pochody je logickou analogií třídy TCompoundAreaSimulace, volá výkonné vrstvy realizující jednotlivé akce s buňkami (deklarované v TCellSimLayers.h). Pro výměnu dat ohledně simulovaných objektů mezi simulací a částí Area slouží "SimPacky" deklarované v TSimPack.h.

### C.1.13 Support

V této části jsou definovány konstanty, makra, rutiny a objekty pomocného a podpůrného charakteru, používané na různých místech programu.

### C.1.14 UiExtension

UiExtension sdružuje objekty drobně rozšiřující možnosti uživatelského rozhraní poskytovaného knihovnou VCL (vyjma SplitTree).

### C.1.15 ZygGraph

ZygGraph implementuje zygotický graf pro buňky.

## C.2 Popis tříd

Popis jednotlivých tříd, jejich metod i atributů je uveden v hlavičkových (\*.h) souborech. Zdrojové texty jsou dále v méně srozumitelných místech okomentovány.

# Literatura

## Odborná literatura

- [1] W. Jost (1970): Diffusion in solids, liquids, gases. Academic Press, New York
- [2] Lodish a kol. (2000): Molecular cell biology. W. H. Freeman and Company, New York

## Diplomové práce

- [3] Václav Bartáček (1999): Simulace činnosti buněk v závislosti na jimi produkovaných látkách. MFF UK, Praha, 1999
- [4] Jiří Bartoň: Simulace vzniku tvaru rostlin. MFF UK, Praha, 2001