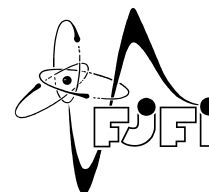




ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE
Fakulta jaderná a fyzikálně inženýrská



Matematické modelování transportu kontrastní látky v cévním řečišti s přestupem do okolní tkáně v problematice perfuze myokardu ve 3D

Mathematical modeling of contrast agent transport in vascular bed with transfer to surrounding tissue in myocardial perfusion problems in 3D

Diplomová práce

Autor: **Bc. Lenka Horvátová**
Vedoucí práce: **doc. Ing. Radek Fučík, Ph.D.**
Konzultant práce: **Ing. Jakub Solovský, Ph.D.**
Akademický rok: 2023/2024

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Horvátová** Jméno: **Lenka** Osobní číslo: **494951**
Fakulta/ústav: **Fakulta jaderná a fyzikálně inženýrská**
Zadávající katedra/ústav: **Katedra matematiky**
Studijní program: **Matematické inženýrství**

II. ÚDAJE K DIPLOMOVÉ PRÁCI

Název diplomové práce:

Matematické modelování transportu kontrastní látky v cévním řečišti s přestupem do okolní tkáně v problematice perfuze myokardu ve 3D

Název diplomové práce anglicky:

Mathematical modeling of contrast agent transport in vascular bed with transfer to surrounding tissue in myocardial perfusion problems in 3D

Pokyny pro vypracování:

1. Formulujte matematický model proudění a transportu kontrastní látky (KL) v cévním řečišti s přestupem do okolní tkáně ve 3D, přičemž cévní řečiště aproximujte pomocí systému propojených jednorozměrných výpočetních oblastí (cév) s variabilní propustností (světlostí).
2. Navrhněte vhodné numerické schéma pro řešení transportu a přestupu KL v cévním řečišti a difuzního procesu KL v okolní tkáni. V práci stručně popište jednotlivé části numerického schématu a jejich vzájemné propojení.
3. Implementujte numerické schéma v C++.
4. Otestujte implementaci numerického schématu na vhodně zvolené testovací úloze a zkoumejte vlastnosti daného modelu.

Seznam doporučené literatury:

1. A. N. Cookson, et al., A spatially-distributed computational model to quantify behaviour of contrast agents in MR perfusion imaging. *Medical Image Analysis* 18.7, 2014, 1200-1216.
2. Ch. S. Peskin, The immersed boundary method. *Acta Numerica* 11, 2002, 479-517.
3. E. Vitásek, Numerické metody. SNTL-Nakladatelství technické literatury, 1987.
4. R. J. LeVeque: Finite Difference Methods for Differential Equations. *AMath* 585-6, 1998.
5. R. J. LeVeque: Finite Volume Methods for Hyperbolic Problems. Cambridge University Press, 2004.

Jméno a pracoviště vedoucí(ho) diplomové práce:

doc. Ing. Radek Fučík, Ph.D. katedra matematiky FJFI

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) diplomové práce:

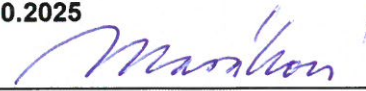
Ing. Jakub Solovský, Ph.D. katedra softwarového inženýrství FJFI

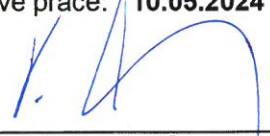
Datum zadání diplomové práce: **31.10.2023**

Termín odevzdání diplomové práce: **10.05.2024**

Platnost zadání diplomové práce: **31.10.2025**


doc. Ing. Radek Fučík, Ph.D.
podpis vedoucí(ho) práce



prof. Ing. Zuzana Masáková, Ph.D.
podpis vedoucí(ho) ústavu/katedry


doc. Ing. Václav Čuba, Ph.D.
podpis děkana(ky)

III. PŘEVZETÍ ZADÁNÍ

Diplomantka bere na vědomí, že je povinna vypracovat diplomovou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací. Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v diplomové práci.

2.11.2023
Datum převzetí zadání


Podpis studentky

Poděkování:

Chtěla bych zde poděkovat především svému školiteli doc. Ing. Radku Fučíkovi, Ph.D. a konzultantovi Ing. Jakubu Solovskému, Ph.D. za pečlivost, ochotu, vstřícnost, cenné rady a odborné i lidské zázemí při práci na mé diplomové práci. Mé poděkování patří také mé rodině a příteli za jejich trpělivost a dosavadní podporu při studiu. Dále bych chtěla poděkovat za podporu grantům č. SGS20/184/OHK4/3T/14, SGS23/188/OHK4/3T/14 a nadacím: Nadace pro rozvoj vzdělání [27], Nadace "Nadání Josefa, Marie a Zdeňky Hlávkových"[28] a Nadace Český literární fond [29].

Čestné prohlášení:

Prohlašuji, že jsem tuto práci vypracovala samostatně a uvedla jsem všechnu použitou literaturu.

V Praze dne 9. května 2024

Bc. Lenka Horvátová

Název práce:

Matematické modelování transportu kontrastní látky v cévním řečišti s přestupem do okolní tkáně v problematice perfuze myokardu ve 3D

Autor: Bc. Lenka Horvátová

Program: Matematické inženýrství

Druh práce: Diplomová práce

Vedoucí práce: doc. Ing. Radek Fučík, Ph.D., Katedra matematiky, Fakulta jaderná a fyzikálně inženýrská, České vysoké učení technické v Praze, Trojanova 13, 120 00, Praha 2

Konzultant práce: Ing. Jakub Solovský, Ph.D., Katedra softwarového inženýrství, Fakulta jaderná a fyzikálně inženýrská, České vysoké učení technické v Praze, Trojanova 13, 120 00, Praha 2

Abstrakt: Práce se zabývá matematickým modelováním situací vznikajících při vyšetření perfuze myokardu pomocí vpuštění kontrastní látky (KL). Popis transportu a přestupu KL z cévního řečiště do mimocévního prostředí je rozdělen na dvě části. Nejdříve na základě tlaků je spočítána rychlost v cévním řečišti, poté je do cévního řečiště vsříknuta KL s danou koncentrací. Přestup KL z cévního řečiště do mimocévního prostředí je modelován pomocí konvoluce s Diracovou delta funkcí. V dalším kroku je spočítána koncentrace v obou prostředích. Pro tento matematický model uvažujeme nestlačitelnou newtonovskou kapalinu, na kterou nepůsobí žádné vnější síly. Mimocévní prostředí je uvažováno jako porézní a rigidní. Hlavním cílem této práce je řešení problému transportu a přestupu KL v cévním řečišti zdravého a nezdravého myokardu. Transport KL v cévním řečišti je řešen metodou konečných objemů a v mimocévním prostředí metodou konečných diferencí.

Klíčová slova: advektivně-difuzní rovnice, Diracova delta funkce, cévní řečiště, metoda konečných diferencí, metoda konečných objemů, perfuze myokardu, porézní prostředí

Title:

Mathematical modeling of contrast agent transport in vascular bed with transfer to surrounding tissue in myocardial perfusion problems in 3D

Author: Bc. Lenka Horvátová

Abstract: This research project deals with mathematical modeling of problems arising during myocardial perfusion using the contrast agent. The description of the transport and transfer of the contrast agent from vascular bed to extravascular medium is divided into two tasks. First, the velocity in the vascular system is computed based on pressures. Then, a contrast agent with a given concentration is injected into the vascular system. The transfer of the contrast agent from the vascular system to the extravascular system is modeled using convolution with the Dirac delta function. In the second step, the concentration in both media is calculated. For this mathematical model, we consider an incompressible Newtonian fluid that is not subject to any external forces. The extravascular environment is considered to be porous and rigid. The main goal of this project is to solve the problem of transport and transfer of contrast agent in the vascular system of healthy and unhealthy myocardium using the finite volume method, and in the extravascular medium using the finite difference method.

Key words: advection-diffusion equation, Dirac's delta function, vascular bed, finite difference method, finite volume method, myocardial perfusion, porous media

Obsah

1	Matematický model	10
1.1	Perfuze myokardu	10
1.2	Mimocévní prostředí	11
1.2.1	Proudění tekutin v mimocévním prostředí	11
1.2.2	Transport KL v mimocévním prostředí	11
1.3	Cévní řečiště	12
1.3.1	Struktura cévního řečiště	12
1.3.2	Proudění tekutin v cévním řečišti	13
1.3.3	Výpočet rychlosti na křižovatkách	15
1.3.4	Transport KL v cévním řečišti	16
1.4	Propojení cévního a mimocévního prostředí	16
1.4.1	Metoda vnořené hranice	16
1.4.2	Přestup kontrastní látky mezi cévním a mimocévním prostředím	17
1.5	Převod veličin do bezrozměrného tvaru	18
1.6	Formulace úlohy	19
2	Numerický model	20
2.1	Mimocévní prostředí	20
2.1.1	Diskretizace mimocévního prostředí	20
2.1.2	Diskretizace diferenciálních výrazů v mimocévním prostředí	21
2.1.3	Množství látky v mimocévním prostředí	22
2.2	Cévní řečiště	22
2.2.1	Diskretizace cévního řečiště	22
2.2.2	Diskretizace diferenciálních výrazů v cévním řečišti	23
2.2.3	Množství látky v cévním řečišti	25
2.3	Přestup KL	25
2.3.1	Aproximace Diracovy delta funkce	26
2.3.2	Diskretizace zdrojových členů	27
2.4	Generování řečiště	27
2.5	Algoritmus	28
2.6	Více cévních řečišť	30
3	Implementace numerického schematu	31
3.1	Simulation.cpp	33
3.2	VascularEnvironment.h	38
3.3	ExtravascularEnvironment.h	39

3.4	Interface.h	40
4	Výsledky	42
4.1	Cévní řečiště	42
4.1.1	Vliv prostorové diskretizace	46
4.1.2	Vliv koeficientu přestupu na celkové množství KL v systému	48
4.1.3	Statistická analýza vlivu náhodnosti generování řečiště	52
4.2	Zahuštění oblasti - více cévních řečišť	54
4.2.1	Zájmová oblast	56
4.2.2	Nezdravý myokard	59

Úvod

Tato práce se zabývá matematickým modelováním transportu kontrastní látky (dále KL) s přestupem do porézního prostředí s aplikací v perfuzi myokardu ve spolupráci s Institutem klinické a experimentální medicíny v Praze (IKEM) a Children's Medical Center, University of Texas Southwestern v Dallasu (UTSW). Myokard v této práci bude značen symbolem Ω a bude složen z cévního řečiště Ω_c a mimocévního prostředí Ω_m . V rámci myokardu budeme uvažovat mimocévní prostředí jako porézní prostředí. Cévní řečiště bude složeno z konečného počtu úseček, které v praxi představují arterioly o průměru mezi $10\mu\text{m}$ až $100\mu\text{m}$ [23]. Všechny cévy menšího průměru než arterioly (tj. s průměrem menším než $10\mu\text{m}$) nerozlišujeme a zahrnujeme do mimocévního prostředí Ω_m . V této práci budeme cévní řečiště Ω_c reprezentovat grafem Γ složeným z vrcholů a hran. Hrany v grafu Γ reprezentují úsečky cévního řečiště Ω_c a vrcholy v grafu Γ reprezentují krajní body těchto úseček.

Předmětem práce bude modelovat situaci, která v praxi vzniká při vyšetření perfuze myokardu. Do cévy je vstříknuta KL, která je unášena cévním řečištěm Ω_c a přestupuje do mimocévního prostředí Ω_m . Pomocí magnetické rezonance lze sledovat vývoj koncentrace KL a na základě tzv. perfuzních křivek stanovit zásobení tkáně (myokardu) okysličenou krví (= perfuze). Následně KL přestupuje z mimocévního prostředí zpět do cévního řečiště, kterým opouští myokard.

Výsledky magnetické rezonance při vyšetření perfuze myokardu u zdravých a nemocných pacientů vykazují rozdílný časový vývoj celkového množství KL, proto důležitou veličinou zkoumanou v této práci bude celkové množství KL v myokardu. V této práci bude myokard nemocného pacienta obsahovat defektní část, což je oblast, ve které je omezený průtok krve a tím pádem omezený přestup KL.

Přestup KL mezi cévním řečištěm Ω_c a mimocévním prostředím Ω_m je modelován konvolucí s Diracovou delta funkcí. Podobným přístupem se zabývá T. Secomb, který vydal v roce 2015 článek [14] využívající Greenovu funkci pro modelování přestupu látky mezi cévami a mimocévním prostředím při perfuzi mozku [15]. Tématem matematického modelování perfuze myokardu pomocí mřížkové Boltzmannovy metody se zabývá článek [26].

Pro úlohu proudění a transportu KL s přestupem budeme předpokládat následující zjednodušení:

- na systém nepůsobí žádné vnější síly,
- cévní prostředí je rigidní (zanedbáváme biomechanické vlastnosti cévního systému a srdce),
- mimocévní prostředí je porézní a rigidní,
- tekutina je nestlačitelná a newtonovská,
- přítomnost KL neovlivňuje hustotu kapaliny (koncentrace KL jsou malé).

Na obrázku 1 je znázorněn řez myokardem, kde je vyznačena symbolem Ω_c jedna z cév (arteriol) cévního řečiště. Okolí cév představuje mimocévní prostředí Ω_m . Šipky naznačují směr proudění krve a transportu KL s přestupem skrz cévní stěnu cévy.



Obrázek 1: Histologie myokardu myši (převzato z [16]) se zvýrazněnou arteriolou.

Práce je rozdělena do 4 kapitol. V první kapitole definujeme základní pojmy týkající se vyšetření perfuze myokardu. Následně popíšeme model transportu KL v cévním řečišti Ω_c a v mimocévním prostředí Ω_m . Dále se budeme zabývat popisem přestupu KL mezi cévním a mimocévním prostředím. Na závěr této kapitoly formulujeme nastavení úlohy perfuze myokardu, která bude zkoumána v poslední kapitole.

Druhá kapitola je věnována numerickým metodám, které jsou použity k řešení úlohy matematického modelu. Zavedeme diskretizaci cévního řečiště a mimocévního prostředí, poté diskretizujeme rovnice transportu KL v obou prostředích. Konkrétněji se jedná o metodu konečných objemů řešenou v cévním řečišti (VCFVM - Vertex-Centered Finite Volume Method) a v mimocévním prostředí se bude jednat o metodu konečných diferencí. Dále popíšeme algoritmus generování cévního řečiště a na závěr popíšeme numerické schéma.

Ve třetí kapitole se budeme zabývat implementačními kroky. Celý kód je implementován v programovacím jazyce C++ za využití knihovny UMFPACK [17] a dostupný na githubu [24]. V této kapitole představíme stěžejní části implementovaného kódu.

Poslední kapitola bude zaměřena na rozbor numerických výsledků. Nejvíce zkoumanou veličinou bude již zmíněné celkové množství KL v myokardu. Ve zdravém myokardu budeme zkoumat vliv diskretizace cévního a mimocévního prostředí pro zvolenou numerickou metodu. Hlavním výstupem této kapitoly bude zkoumání časového průběhu celkového množství KL pro případ myokardu s defektem (se zhoršeným průtokem krve) v porovnání se zdravým myokardem.

Kapitola 1

Matematický model

V této kapitole nejprve stručně popíšeme pojmy související s perfuzí myokardu. Budeme se zabývat prouděním tekutin a transportu KL ve volném a v porézním prostředí a upřesníme význam předpokladů, které klademe na tekutinu a prostředí. Transport KL podrobně popíšeme pro situaci na úsečce (jedné cévě) a na křižovatce (rozvětvení cévy). Dále se budeme zabývat přestupem KL mezi cévním a mimocévním prostředím. Na závěr zformulujeme výslednou úlohu, která zahrnuje všechny výše popsané procesy.

Předpokládejme, že $\Omega_m \subset \mathbb{R}^3$ je oblast, do které je vnořeno cévní řečiště $\Omega_c \subset \mathbb{R}^3$, které je složeno z konečného počtu propojených úseček. Cévní řečiště budeme v této práci reprezentovat grafem Γ , který je složen z vrcholů v a hran e . Hrana e v cévním řečišti reprezentuje úsečku v_e a vrchol $v_{in,e}$, resp. $v_{out,e}$ reprezentuje bod úsečky v_e , kterým KL vstupuje do, resp. vystupuje z dané úsečky.

Úlohu budeme řešit v časovém intervalu $I = (0, T_{max})$, kde T_{max} je finální čas.

1.1 Perfuze myokardu

Srdce je složeno ze tří vrstev – perikard, myokard a endokard. Perikard se nachází na povrchu srdce. Naopak vnitřní stěna srdce se nazývá endokard. Srdeční sval, neboli myokard, je jeden z nejdůležitějších svalů v lidském těle [18], zodpovědný za kontrakci srdce a zajištění krevního oběhu.

Perfuzí myokardu je proces, při kterém se okysličená krev transportuje věnčitými tepnami do myokardu. Tento proces umožňuje okysličování myokardu. Mohou nastat situace, kdy je cévní řečiště, kterým okysličená krev putuje, z nějakého důvodu blokováno. Jedním z důvodů může být arteriální skleróza. Poté dochází k nedostatečnému okysličování myokardu, což může způsobit srdeční infarkt [19].

Stanovení perfuze myokardu se obvykle provádí pomocí vyšetření na magnetické rezonanci (MRI). Při vyšetření je do žíly pacienta vstříknuta KL, která zvyšuje kontrast na obrazech MRI. Tímto umožňuje lékařům pozorovat, jak krev prochází cévním řečištěm a jak se KL šíří v tkáni (mimocévním prostředí) [20].

V této práci se zabýváme zjednodušeným modelem situace, která nastává při vyšetření perfuze myokardu s využitím KL. Zkoumáním transportu KL cévním řečištěm s následným přestupem do myokardu lze ukázat, jak krev protéká tímto srdečním svalem. Pokud se do některé části myokardu KL nedostane, může se jednat o poškozenou oblast nebo o nedostatečný krevní tok touto oblastí. Poškozená místa budeme nazývat defektními místy.

1.2 Mimocévní prostředí

Tato sekce bude věnována matematickému popisu proudění tekutin v mimocévním prostředí a transportu KL tímto prostředím.

1.2.1 Proudění tekutin v mimocévním prostředí

Proces proudění tekutin v porézním prostředí Ω_m je v makroskopickém měřítku popsán rovnicí kontinuity a Darcyho zákonem [2]

$$\frac{\partial(\phi\rho)}{\partial t} + \nabla \cdot (\rho\mathbf{u}_m) = 0 \quad \text{v } \Omega_m \times I, \quad (1.1a)$$

$$\mu\mathbf{u}_m + \mathbb{K}\nabla p_m = 0 \quad \text{v } \Omega_m \times I, \quad (1.1b)$$

kde veličiny vystupující v těchto rovnicích jsou funkcemi polohy \mathbf{x} [m] a času t [s] s následujícím významem:

ρ	[kg m ⁻³]	hustota tekutiny,
\mathbf{u}_m	[m s ⁻¹]	Darcyho rychlost tekutiny,
μ	[kg m ⁻¹ s ⁻¹]	dynamická viskozita tekutiny,
\mathbb{K}	[m ²]	tenzor propustnosti,
ϕ	[-]	porozita,
p_m	[Pa]	tlak tekutiny.

Za předpokladu, že modelujeme proudění newtonovské nestlačitelné ($\rho = konst$) kapaliny v rigidním ($\phi_m = konst$), izotropním ($\mathbb{K}_m = K_m\mathbb{I}$, kde K_m [m²] se nazývá koeficient propustnosti) porézním prostředí, tak rovnice (1.1) přejdou do tvaru

$$\nabla \cdot \mathbf{u}_m = 0 \quad \text{v } \Omega_m \times I, \quad (1.2a)$$

$$\mu\mathbf{u}_m + K_m\nabla p_m = 0 \quad \text{v } \Omega_m \times I. \quad (1.2b)$$

V této práci předpokládáme nulovou rychlost \mathbf{u}_m v mimocévním prostředí. Dosazení nulové rychlosti do rovnice (1.2b) implikuje nulový gradient tlaku, tj. tlak v mimocévním prostředí je konstantní.

1.2.2 Transport KL v mimocévním prostředí

Rychlost proudění v mimocévním prostředí je nulová, proto je transport KL v tomto prostředí popsán difuzní rovnicí

$$\phi_m \frac{\partial c_m}{\partial t} = \tilde{D}_m \Delta c_m + \phi_m g_m \quad \text{v } \Omega_m \times I, \quad (1.3)$$

kde c_m [kg m⁻³] je koncentrace KL a dolní index m označuje, že uvedené veličiny jsou vztaženy k mimocévnímu prostředí Ω_m . Difuzní koeficient \tilde{D}_m [m² s⁻¹] je konstantní. Člen g_m [kg m⁻³ s⁻¹] reprezentuje přestup KL mezi cévním a mimocévním prostředím.

Rovnici (1.3) vydělíme porozitou ϕ_m a dostaneme

$$\frac{\partial c_m}{\partial t} = D_m \Delta c_m + g_m \quad \text{v } \Omega_m \times I, \quad (1.4)$$

kde

$$D_m = \frac{\tilde{D}_m}{\phi_m}. \quad (1.5)$$

Tímto jsme porozitu ϕ_m zahrnuli do difuzního koeficientu a ve zbytku práce ji nebudeme explicitně zmiňovat.

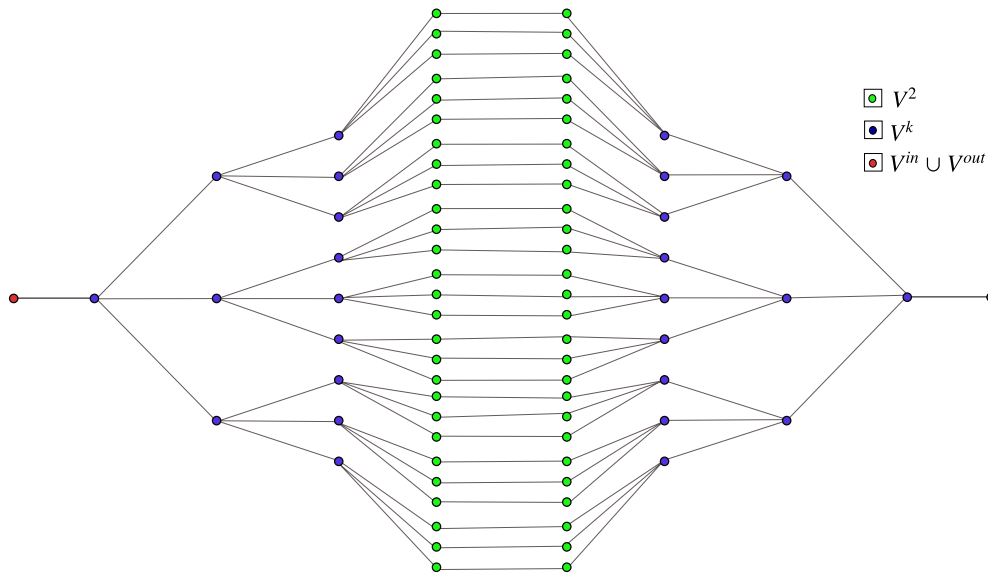
1.3 Cévní řečiště

Nejdříve popíšeme strukturu cévního řečiště, které reprezentujeme grafem. Dále se budeme zabývat prouděním tekutiny a transportu KL v cévním řečišti.

1.3.1 Struktura cévního řečiště

Cévní řečiště Ω_c je reprezentováno grafem Γ , který je složen z vrcholů a hran, tj. $\Gamma = (V, E)$, kde V je konečná množina vrcholů grafu Γ a $E \subseteq \{\{v, w\} \mid v \in V \wedge w \in V\}$ je množina hran grafu Γ .

Řekneme, že vrchol $v \in V$ je sousedem vrcholu $w \in V$ v grafu Γ , pokud jsou tyto dva vrcholy spojené hranou, tj. $\{v, w\} \in E$. Množinu sousedů vrcholu v budeme značit $\mathcal{N}(v) = \{w \in V \mid w \text{ je souseď } v\}$. Nyní již můžeme zavést pojem stupeň vrcholu $v \in V$ v grafu Γ jako $\deg_{\Gamma}(v) = |\mathcal{N}(v)|$, tj. počet sousedů vrcholu v v grafu Γ .



Obrázek 1.1: Ilustrace grafu Γ . Červenou barvou jsou vyznačeny okrajové vrcholy, modrou barvou značíme křižovatky a zeleně vnitřní vrcholy se 2 sousedy.

Mějme graf na obrázku 1.1, kde vrcholy tohoto grafu jsou znázorněny tečkami a úsečky mezi jednotlivými vrcholy reprezentují hrany grafu.

V této práci budeme rozlišovat tři typy vrcholů:

- (a) Krajní vrcholy označené na obrázku 1.1 červeně. Budeme rozlišovat vstupní a výstupní krajní vrcholy. Množinu vstupních krajních vrcholů značíme V^{in} a množinu výstupních krajních vrcholů značíme V^{out} . Rozlišení typu krajních vrcholů bude mít vliv na volbu okrajové podmínky. Množinou krajních vrcholů budeme rozumět $V^{in} \cup V^{out}$. Pro krajní vrcholy platí

$$\deg_{\Gamma}(v) = 1 \quad \forall v \in V^{in} \cup V^{out}. \quad (1.6)$$

- (b) Vnitřní vrcholy, které mají právě 2 sousedy, jsou na obrázku 1.1 označené zeleně. Transport KL přes tyto vrcholy je podobný problému zkoumanému v bakalářské práci [10]. Tyto vrcholy značíme V^2 a splňují

$$\deg_{\Gamma}(v) = 2 \quad \forall v \in V^2. \quad (1.7)$$

- (c) Křižovatka, tj. vrcholy označené modrou barvou na obrázku 1.1. Přestup KL přes tyto vrcholy bude podrobně zkoumán v této práci a množinu křižovatek budeme značit V^k . Pro stupeň těchto vrcholů platí vztah

$$\deg_{\Gamma}(v) \geq 3 \quad \forall v \in V^k. \quad (1.8)$$

Sjednocení množin V^2 a V^k značíme V^{int} a nazýváme vnitřní vrcholy.

Jednotlivé cévy budou modelovány jako válcové trubky s danou délkou a průřezem. Poznamenejme, že model představený v této sekci lze také chápat v cévním řečišti jako model puklinového proudění [6].

1.3.2 Proudění tekutin v cévním řečišti

Popis proudění tekutiny vyžaduje znalost rychlosti tekutiny na každé úsečce cévního řečiště $\Omega_c \subset \mathbb{R}^3$. Proudění popíšeme na hraně $e \in \Gamma$.

Proudění nestlačitelné tekutiny bez působení vnějších sil v cévním řečišti je popsáno rovnicí kontinuity a Navierovými-Stokesovými rovnicemi [21].

$$\nabla \cdot \mathbf{u}_c = 0 \quad \text{v } \Omega_c \times I, \quad (1.9)$$

$$\rho \frac{D\mathbf{u}_c}{Dt} = -\nabla p_c + \mu \Delta \mathbf{u}_c \quad \text{v } \Omega_c \times I, \quad (1.10)$$

kde veličiny vystupující v těchto rovnicích jsou funkcemi polohy \mathbf{x} [m] a času t [s] s následujícím významem:

\mathbf{u}_c	[m s ⁻¹]	rychlost tekutiny,
μ	[kg m ⁻¹ s ⁻¹]	dynamická viskozita tekutiny,
p_c	[Pa]	tlak tekutiny.

V rovnici (1.10) je Laplaceův operátor aplikován na \mathbf{u}_c po složkách a na levé straně rovnice (1.10) se vyskytuje operátor materiálové derivace

$$\frac{D}{Dt} = \frac{\partial}{\partial t} + \mathbf{u}_c \cdot \nabla. \quad (1.11)$$

Pro rychlost \mathbf{u}_c v rovnici (1.10) platí následující vztah

$$\mathbf{u}_c = u_{c,e} \mathbf{s}, \quad (1.12)$$

kde \mathbf{s} je směrový vektor úsečky, která je reprezentována hranou e . Dále budeme popisovat proudění tekutiny na hraně $e \in E(\Gamma)$.

Řešení rovnic (1.9) a (1.10) pro laminární proudění v trubce (reprezentována hranou e) s kruhovým průřezem s poloměrem r_e je Hagenův-Poiseuillov tok Q_e [kg s^{-1}] ve tvaru [25]

$$Q_e = \pi r_e^4 \frac{p_{out,e} - p_{in,e}}{8\mu L_e}, \quad (1.13)$$

kde $p_{in,e}$ [Pa], resp. $p_{out,e}$ [Pa] je tlak ve vstupním, resp. výstupním vrcholu hrany e a L_e [m] je vzdálenost mezi těmito vrcholy, viz obrázek 1.2a.

Pro zajímavost popíšeme proudění tekutiny v úsečce jako proudění v porézním prostředí. Tento přístup následně porovnáme s Hagenovým-Poiseuillovým laminárním prouděním.

Na obrázku 1.2 je znázorněn přístup Poiseuilleova laminárního proudění 1.2b a proudění tekutiny v porézním prostředí 1.2c.

Proudění tekutiny v porézním prostředí v jedné dimenzi je určeno Darcyho zákonem vztahem [2]

$$u_{c,e}^{(D)} = -\frac{K_e}{\mu} \frac{p_{out,e} - p_{in,e}}{L_e}. \quad (1.14)$$

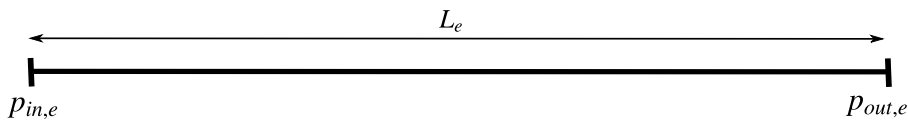
Celkový objemový tok Q_e skrz hranu e je dán

$$Q_e = A_e u_{c,e}^{(D)}, \quad (1.15)$$

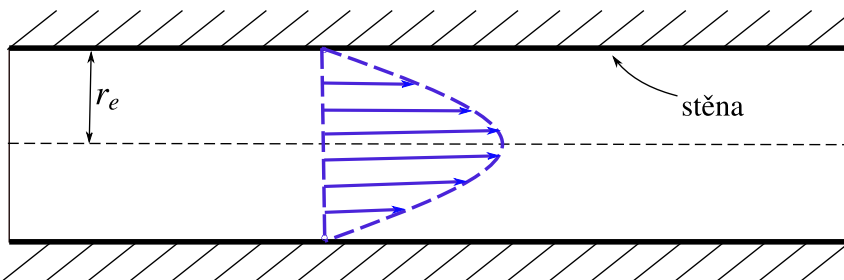
kde A_e [m^2] je průřez trubky, kterou hrana e reprezentuje.

Pro matematický model v této práci se tyto přístupy dají považovat za ekvivalentní. Porovnáním puklinového proudění a laminárního proudění dostaneme vztah pro koeficient propustnosti K_e ve tvaru

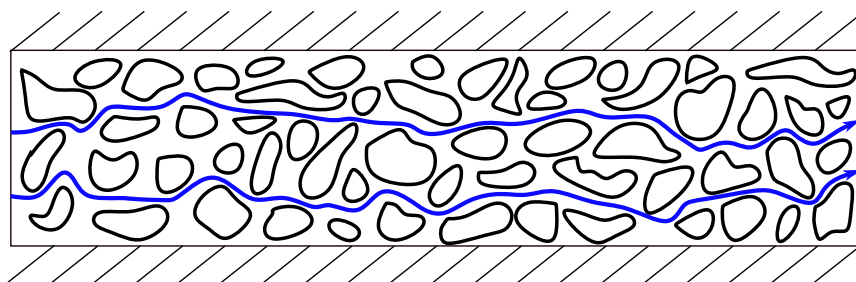
$$K_e = \frac{r_e^2}{8}. \quad (1.16)$$



(a) 1D reprezentace hrany e v grafu Γ . Tlak $p_{in,e}$, resp. $p_{out,e}$ je vstupní, resp. výstupní tlak.



(b) Poiseuilleovo laminární proudění v trubce o poloměru r_e . Zde je předepsána nulová rychlost na zdi, která vede na parabolický profil rychlosti tekutiny.



(c) Proudění tekutiny v porézním prostředí.

Obrázek 1.2: Dva možné přístupy výpočtu rychlosti v trubce.

Tímto způsobem jsme schopni spočítat rychlost tekutiny na dané hraně při znalosti tlaků ve vstupním a výstupním vrcholu hrany.

1.3.3 Výpočet rychlosti na křižovatkách

V grafu Γ známe pouze hodnoty tlaku v krajních vrcholech $v \in V^{in} \cup V^{out}$. Pro určení tlaku v ostatních vrcholech musíme doplnit další podmínky. V cévním řečišti předpokládáme, že se hmota zachovává, tj. množství tekutiny, které do daných vrcholů přiteče, musí zároveň i odtéct.

Předpokládejme, že máme vrchol $v \in V$, který je krajním vrcholem pro hrany e_1, \dots, e_n pro $n \in \mathbb{N}$. Pak pro vrchol v platí

$$\sum_{i=1}^n Q_{v,e_i} = 0, \quad (1.17)$$

kde Q_{v,e_i} označuje celkový objemový tok přes vrchol v skrz hranu e_i , $\forall i \in \hat{n}$.

Složením rovnic (1.14), (1.15) a (1.17) dostaneme soustavu pro výpočet neznámých hodnot tlaku ve všech vrcholech, tj. formálně dostaneme soustavu lineárních algebraických rovnic s řídkou maticí \mathbb{A} .

$$\mathbb{A}\mathbf{p} = \mathbf{b}. \quad (1.18)$$

Vyřešením rovnice (1.18) získáme hodnotu tlaku v každém vrcholu a dosazením tlaků do rovnice (1.14) spočítáme rychlost na každé hraně e grafu Γ .

1.3.4 Transport KL v cévním řečišti

Transport KL v cévním řečišti popíšeme pro hranu $e \in E(\Gamma)$. Jednotlivé hrany grafu Γ jsou propojeny křížovatkami, ve kterých uvažujeme platnost zachování hmoty (součet toků je nulový).

S použitím rychlosti $u_{c,e}$ získané z (1.14) můžeme popsat transport KL v cévním řečišti Ω_c advekčně-difuzní rovnicí

$$\frac{\partial c_c}{\partial t} + \nabla \cdot (c_c u_{c,e}) = \nabla \cdot (D_c \nabla c_c) + g_c \quad \text{v } \Gamma \times I, \quad (1.19)$$

kde c_c [kg m⁻³] je koncentrace KL a dolní index c označuje veličiny příslušné k cévnímu řečišti. D_c [m² s⁻¹] je difuzní koeficient, který bude v této práci konstantní. Zdrojový člen g_c [kg m⁻³s⁻¹] použijeme pro modelování přestupu KL mezi cévou a mimocévním prostředím.

1.4 Propojení cévního a mimocévního prostředí

V této sekci nejdříve popíšeme metodu vnořené hranice a následně ji aplikujeme na výpočet zdrojových členů.

1.4.1 Metoda vnořené hranice

V této práci je mimocévní prostředí popsáno eulerovskou prostorovou souřadnicí $\mathbf{x} \in \mathbb{R}^3$ a cévní řečiště je popsáno lagrangeovským bodem $\mathbf{X} \in \Omega_c$.

Mějme hranu e v grafu Γ . Tato hrana reprezentuje množinu $\{\mathbf{X}_e(s) \mid s \in S_e\}$, kde $S_e = \langle 0, L_e \rangle$ je parametrická množina souřadnic popisující hranu e .

Lagrangeovský bod $\mathbf{X}_e(s)$ nacházející se na hraně $e \in E(\Gamma)$ lze vyjádřit vztahem

$$\mathbf{X}_e(s) = \mathbf{v}_{e,1} + \frac{s}{L_e}(\mathbf{v}_{e,2} - \mathbf{v}_{e,1}) \quad s \in S_e, \quad (1.20)$$

kde $\mathbf{v}_{e,1}$ a $\mathbf{v}_{e,2}$ jsou vektory souřadnic okrajových vrcholů hrany e . Přestup KL bude popsán pro hranu $e \in E(\Gamma)$.

Pro popis přestupu KL mezi cévním a mimocévním prostředím budeme potřebovat definovat převodní vztahy veličin mezi těmito souřadnými systémy. Pro tyto účely využijeme metody vnořené hranice.

Necht' ω^E a ω^L jsou po řadě funkce popsány eulerovsky a lagrangeovsky. Převodní vztahy mezi těmito funkcemi jsou dány následovně [3]

$$\omega^E(\mathbf{x}, t) = \int_{S_e} \omega^L(\mathbf{X}_e(s), t) \delta(\mathbf{x} - \mathbf{X}_e(s)) ds \quad \forall \mathbf{x} \in \Omega_m, \forall t \in I, \quad (1.21a)$$

$$\omega^L(\mathbf{X}_e(s), t) = \int_{\Omega_m} \omega^E(\mathbf{x}, t) \delta(\mathbf{X}_e(s) - \mathbf{x}) d\mathbf{x} \quad \forall s \in S_e, \forall t \in I, \quad (1.21b)$$

kde δ je Diracova delta funkce. Formálně první integrál reprezentuje konvoluci funkce δ se spojitou funkcí ω^L s kompaktním nosičem na Ω_m a druhý integrál představuje jednoduchou vrstvu v Ω_c [12]. Poznamenejme, že se jedná o podobný konceptuální přístup jako pomocí Greenovy funkce [14].

1.4.2 Přestup kontrastní látky mezi cévním a mimocévním prostředím

Zdrojové členy g_c, g_m , které vystupují v rovnicích (1.19), (1.3), jsou funkcemi koncentrací c_c a c_m . Tyto zdrojové členy jsou uvažovány ve tvaru [7]

$$g_c = k_c(c_m - c_c), \quad (1.22a)$$

$$g_m = k_m(c_c - c_m), \quad (1.22b)$$

kde koeficienty přestupu k_c, k_m jsou dány

$$k_c(c_c, c_m) = \begin{cases} k_c^{(1)} & \text{pro } c_c \geq c_m, \\ k_c^{(2)} & \text{pro } c_c < c_m, \end{cases} \quad (1.23a)$$

$$k_m(c_c, c_m) = \begin{cases} k_m^{(1)} & \text{pro } c_c \geq c_m, \\ k_m^{(2)} & \text{pro } c_c < c_m, \end{cases} \quad (1.23b)$$

kde $k_c^{(p)}$ [s^{-1}] a $k_m^{(p)}$ [s^{-1}] pro $p \in \{1, 2\}$ jsou koeficienty přestupu popisující propustnost cévní stěny ve směru z cévního řečiště do mimocévního prostředí a z mimocévního prostředí do cévního řečiště. Horní index u koeficientů přestupu určuje zda kontrastní látka odchází z cévního řečiště do mimocévního prostředí (index 1) a horní index 2 reprezentuje přestup z mimocévního prostředí do cévního. Rozdílnost koeficientů $k_c^{(1)}$ a $k_c^{(2)}$, resp. $k_m^{(1)}$ a $k_m^{(2)}$ představuje obecně různou propustnost cévní stěny z vnitřní a vnější strany. V této práci bude propustnost cév z vnitřní a vnější strany stejná, tj. $k_c^{(1)} = k_c^{(2)} \equiv k_c$ a $k_m^{(1)} = k_m^{(2)} \equiv k_m$.

Pomocí vztahu (1.22) lze rovnice (1.19) a (1.3) přepsat do tvaru

$$\frac{\partial c_c}{\partial t} + \nabla \cdot (c_c \mathbf{u}_c) = \nabla \cdot (D_c \nabla c_c) + k_c(c_m - c_c), \quad (1.24a)$$

$$\frac{\partial c_m}{\partial t} = \nabla \cdot (D_m \nabla c_m) + k_m(c_c - c_m). \quad (1.24b)$$

Odvodíme vztah mezi koeficienty přestupu k_c a k_m . Množství KL, které odchází z cévního řečiště musí být stejné jako to, které přichází do mimocévního prostředí (a naopak), tj. bude platit

$$g_c = - (g_m)^L, \quad (1.25)$$

resp.

$$g_m = - (g_c)^E, \quad (1.26)$$

kde horní index L , resp. E je vyjádření dané veličiny v langrangeovských, resp. eulerovských souřadnicích. Dále budeme pracovat jen s rovnicí (1.25), pro rovnici (1.26) je postup analogický.

Aplikováním převodního vztahu (1.21) na (1.25) dostaneme $\forall e \in E(\Gamma), \forall s \in S_e, \forall t \in I$

$$g_c(\mathbf{X}_e(s), t) = \int_{\Omega_m} g_c^E(\mathbf{x}, t) \delta(\mathbf{X}_e(s) - \mathbf{x}) d\mathbf{x}. \quad (1.27)$$

Dosazením následujících vztahů

$$g_c(\mathbf{X}_e(s), t) = k_c(c_m^L(\mathbf{X}_e(s), t) - c_c(\mathbf{X}_e(s), t)) \quad \forall e \in E(\Gamma), \forall s \in S_e, \forall t \in I, \quad (1.28)$$

$$g_m(\mathbf{x}, t) = k_m(c_c^E(\mathbf{x}, t) - c_m(\mathbf{x}, t)) \quad \forall \mathbf{x} \in \Omega_m, \forall t \in I, \quad (1.29)$$

do rovnice (1.25) získáme vztah pro $\forall e \in E(\Gamma), \forall s \in S_e, \forall t \in I$

$$\begin{aligned} k_c(c_m^L(\mathbf{X}_e(s), t) - c_c(\mathbf{X}_e(s), t)) &= - \int_{\Omega_m} k_m(c_c^E(\mathbf{x}, t) - c_m(\mathbf{x}, t)) \delta(\mathbf{X}_e(s) - \mathbf{x}) \, d\mathbf{x} \\ &= -k_m \int_{\Omega_m} c_c^E(\mathbf{x}, t) \delta(\mathbf{X}_e(s) - \mathbf{x}) \, d\mathbf{x} + k_m \int_{\Omega_m} c_m(\mathbf{x}, t) \delta(\mathbf{X}_e(s) - \mathbf{x}) \, d\mathbf{x} \\ &= k_m(c_m^L(\mathbf{X}_e(s), t) - c_c(\mathbf{X}_e(s), t)), \end{aligned} \quad (1.30)$$

kde jsme použili následující vztahy

$$\int_{\Omega_m} c_c^E(\mathbf{x}, t) \delta(\mathbf{X}_e(s) - \mathbf{x}) \, d\mathbf{x} = c_c(\mathbf{X}_e(s), t) \quad \forall e \in E(\Gamma), \forall s \in S_e, \forall t \in I, \quad (1.31)$$

$$\int_{\Omega_m} c_m(\mathbf{x}, t) \delta(\mathbf{X}_e(s) - \mathbf{x}) \, d\mathbf{x} = c_m^L(\mathbf{X}_e(s), t) \quad \forall e \in E(\Gamma), s \in S_e, \forall t \in I. \quad (1.32)$$

Aby byla splněna rovnost v (1.30), musí platit

$$k_c = k_m \equiv k, \quad (1.33)$$

čímž zavedeme jeden přestupní koeficient k .

1.5 Převod veličin do bezrozměrného tvaru

Pro praktické využití matematického modelu je výhodné převést veličiny, které jsme doposud definovali, do bezrozměrného tvaru následujícími vztahy

$$c_c = c_0 c_c^*, \quad (1.34a)$$

$$c_m = c_0 c_m^*, \quad (1.34b)$$

$$p_c = p_0 p_c^*, \quad (1.34c)$$

$$\mathbf{u}_c = u_0 \mathbf{u}_c^*, \quad (1.34d)$$

$$\mathbf{x} = x_0 \mathbf{x}^*, \quad (1.34e)$$

$$\mathbf{X} = x_0 \mathbf{X}^*, \quad (1.34f)$$

$$t = t_0 t^*, \quad (1.34g)$$

$$k_\alpha = \frac{1}{t_0} k_\alpha^* \quad \forall \alpha \in \{c, m\}, \quad (1.34h)$$

kde c_0 [kg m⁻³] je charakteristická koncentrace KL, p_0 [Pa] je charakteristický tlak, u_0 [m s⁻¹] je charakteristická rychlost, x_0 [m] je charakteristická délka a t_0 [s] je charakteristický čas. Veličiny označené v (1.34) s horním indexem * jsou bezrozměrné. Dále budeme používat pouze bezrozměrné veličiny, kde pro lepší přehlednost budeme vynechávat horní index "*". Přechod k výše uvedeným bezrozměrným veličinám zjednoduší analýzu řešení rovnic (1.24).

1.6 Formulace úlohy

Nechť $\Omega_m = (0, L_{x_1}) \times (0, L_{x_2}) \times (0, L_{x_3})$ je třírozměrná oblast s hranicí $\partial\Omega_m$, kde L_{x_1} , resp. L_{x_2} , resp. L_{x_3} je délka stran Ω_m na ose x_1 , resp. x_2 , resp. x_3 . Dále necht' graf popisující cévní řečiště Ω_c představuje sjednocení konečného počtu hran. Hrana $e \in E$ reprezentuje úsečku s rychlostí $u_{c,e}$ o délce L_e . Mějme časový interval $I = (0, T_{max})$, kde T_{max} je finální čas.

Mějme prostorovou souřadnici $\mathbf{x} = (x_1, x_2, x_3)^T$ určující pozici v oblasti Ω_m a necht' hrana e v grafu Γ reprezentuje množinu $\{\mathbf{X}_e(s) \mid s \in S_e\} = \nu_e$.

Transportní rovnice (1.24) pro $c_c = c_c(\mathbf{X}_e(s), t)$ a $c_m = c_m(\mathbf{x}, t)$ jsou formulovány následovně

$$\frac{\partial c_c}{\partial t} + u_{c,e} \frac{\partial c_c}{\partial s} = D_c \frac{\partial^2 c_c}{\partial s^2} + g_c \quad \text{v } S_e \times I, \forall e \in E(\Gamma) \quad (1.35a)$$

$$\frac{\partial c_m}{\partial t} = D_m \left(\frac{\partial^2 c_m}{\partial x_1^2} + \frac{\partial^2 c_m}{\partial x_2^2} + \frac{\partial^2 c_m}{\partial x_3^2} \right) + g_m \quad \text{v } \Omega_m \times I, \quad (1.35b)$$

kde $u_{c,e}$ je konstantní rychlost na hraně e .

Předpokládáme, že do cévního řečiště je vsříknuta KL s časově proměnnou koncentrací $c = c(t)$ danou vztahem

$$c(t) = e^{-K_c(t-t_c)^2} \quad \forall t \in I, \quad (1.36)$$

kde K_c [-] a t_c [-] jsou bezrozměrné konstanty určující časové škálování a posun v čase. Počáteční a okrajové podmínky (o.p.) této úlohy jsou

- Pro cévní řečiště:

$$\text{Počáteční podmínka:} \quad c_c(\mathbf{X}, 0) = c_c^{ini}(\mathbf{X}) \quad \forall \mathbf{X} \in \overline{\Omega_c}, \quad (1.37a)$$

$$\text{Dirichletova o.p.:} \quad c_c(\mathbf{X}, t) = c(t) \quad \forall (\mathbf{X}, t) \in \partial\Omega_c^{in} \times I, \quad (1.37b)$$

$$\text{Neumannova o.p.:} \quad \frac{\partial c_c}{\partial s}(\mathbf{X}, t) = 0 \quad \forall (\mathbf{X}, t) \in \partial\Omega_c^{out} \times I, \quad (1.37c)$$

kde $\partial\Omega_c^{in}$, resp. $\partial\Omega_c^{out}$ je množina bodů, které jsou reprezentovány v grafu Γ množinou vrcholů V^{in} , resp. V^{out} .

- Pro mimocévní prostředí:

$$\text{Počáteční podmínka:} \quad c_m(\mathbf{x}, 0) = c_m^{ini}(\mathbf{x}) \quad \forall \mathbf{x} \in \overline{\Omega_m}, \quad (1.38a)$$

$$\text{Dirichletova o.p.:} \quad c_m(\mathbf{x}, t) = c_m^{Dir} \quad \forall (\mathbf{x}, t) \in \partial\Omega_m^{Dir} \times I, \quad (1.38b)$$

$$\text{Neumannova o.p.:} \quad \nabla c_m(\mathbf{x}, t) \cdot \mathbf{n}(\mathbf{x}) = 0 \quad \forall (\mathbf{x}, t) \in \partial\Omega_m^{Neu} \times I, \quad (1.38c)$$

kde $\mathbf{n}(\mathbf{x})$ je jednotkový vektor vnější normály hranice $\partial\Omega_m$ v bodě \mathbf{x} . Hranice oblasti Ω_m splňují

$$\partial\Omega_m^{Dir} \cap \partial\Omega_m^{Neu} = \emptyset, \quad (1.39a)$$

$$\overline{\partial\Omega_m^{Dir} \cup \partial\Omega_m^{Neu}} = \partial\Omega_m. \quad (1.39b)$$

Zdrojové členy g_c a g_m jsou dány vztahy ze sekce 1.4.2

$$g_c(\mathbf{X}, t) = k_c(c_m^L(\mathbf{X}, t) - c_c(\mathbf{X}, t)) \quad \forall (\mathbf{X}, t) \in \Omega_c \times I, \quad (1.40a)$$

$$g_m(\mathbf{x}, t) = k_m(c_c^E(\mathbf{x}, t) - c_m(\mathbf{x}, t)) \quad \forall (\mathbf{x}, t) \in \Omega_m \times I, \quad (1.40b)$$

kde koncentrace c_m^L a c_c^E jsou určeny převodními vztahy (1.21) ze sekce 1.4.1.

Kapitola 2

Numerický model

Popis numerické metody pro řešení úlohy (1.35) rozdělíme na několik sekcí: v prvních dvou se budeme zabývat transportem kontrastní látky, další bude věnována přestupu KL mezi cévním a mimocévním prostředím a na závěr popíšeme metodu pro generování cévního řečiště a samotný algoritmus.

V první části se budeme zabývat metodou konečných diferencí pro difuzní rovnici (1.3) v mimocévním prostředí a v části druhé metodou konečných diferencí pro advekčně-difuzní rovnici (1.19) v cévním řečišti. Postup pro obě metody bude obdobný, nejdříve diskretizujeme příslušnou oblast, poté aproximujeme derivace diferenciálními výrazy a aproximujeme zdrojové členy. Diskretizované oblasti a veličiny budou značeny stříškou ("^") nad daným symbolem.

Budeme se zabývat časově implicitními schémata s výjimkou výpočtu zdrojových členů.

Pro obě části zavedeme diskretizaci časového intervalu \bar{I} a I , kterou označíme $\hat{\bar{I}}$ a \hat{I} danou vztahem

$$\hat{\bar{I}} = \{t_i = i\Delta t \mid i \in \{0, 1, \dots, N_t - 1\}\}, \quad (2.1)$$

$$\hat{I} = \{t_i = i\Delta t \mid i \in \{1, \dots, N_t - 2\}\}, \quad (2.2)$$

kde Δt je časový krok, N_t představuje počet časových kroků a platí

$$\Delta t = \frac{T_{max}}{N_t - 1}. \quad (2.3)$$

2.1 Mimocévní prostředí

V této části se zaměříme na popis úlohy (1.35b) v sekci 1.6 pro mimocévní prostředí metodou konečných diferencí.

2.1.1 Diskretizace mimocévního prostředí

Zavedeme diskretizaci oblasti $\bar{\Omega}_m$ značenou symbolem $\hat{\bar{\Omega}}_m$ a bude dána předpisem

$$\bar{\hat{\Omega}}_m = \left\{ \mathbf{x}_{j,k,l} = \begin{pmatrix} j\Delta x_1 \\ k\Delta x_2 \\ l\Delta x_3 \end{pmatrix} \mid j \in \{0, 1, \dots, N_{x_1} - 1\}, k \in \{0, 1, \dots, N_{x_2} - 1\}, l \in \{0, 1, \dots, N_{x_3} - 1\} \right\}, \quad (2.4)$$

$$\hat{\Omega}_m = \left\{ \mathbf{x}_{j,k,l} = \begin{pmatrix} j\Delta x_1 \\ k\Delta x_2 \\ l\Delta x_3 \end{pmatrix} \mid j \in \{1, 2, \dots, N_{x_1} - 2\}, k \in \{1, 2, \dots, N_{x_2} - 2\}, l \in \{1, 2, \dots, N_{x_3} - 2\} \right\}, \quad (2.5)$$

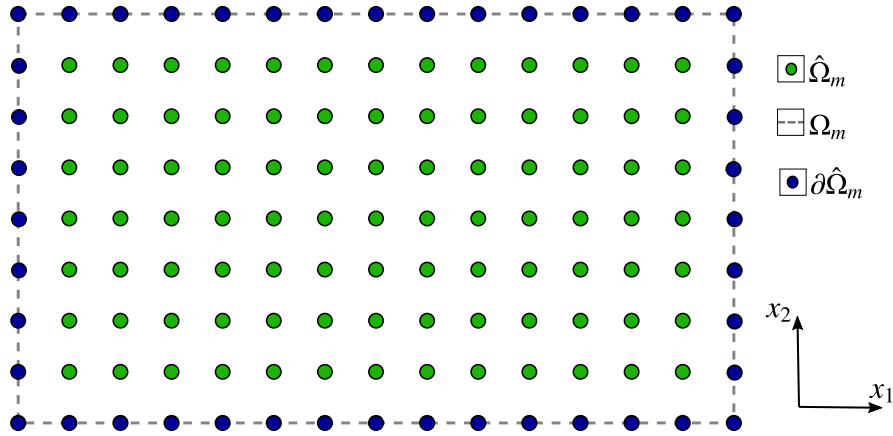
$$\partial\hat{\Omega}_m = \bar{\hat{\Omega}}_m \setminus \hat{\Omega}_m, \quad (2.6)$$

kde $\Delta x_1, \Delta x_2$ a Δx_3 je po řadě prostorový krok na ose x_1, x_2 a x_3 . Dále N_{x_1}, N_{x_2} a N_{x_3} představují počty uzlů na příslušných osách a splňují vztahy

$$\Delta x_1 = \frac{L_{x_1}}{N_{x_1} - 1}, \quad (2.7)$$

$$\Delta x_2 = \frac{L_{x_2}}{N_{x_2} - 1}, \quad (2.8)$$

$$\Delta x_3 = \frac{L_{x_3}}{N_{x_3} - 1} \quad (2.9)$$



Obrázek 2.1: Ilustrace diskretizace oblasti Ω_m pro řez $x_3 \in (0, L_{x_3})$.

2.1.2 Diskretizace diferenciálních výrazů v mimocévním prostředí

Diferenciální výrazy z rovnice (1.35b) nahradíme konečnými diferencemi pro koncentraci $c_m(\mathbf{x}_{j,k,l}, t_i) = c_{m,j,k,l}^i$ [8]

$$\frac{\partial c_m}{\partial t}(\mathbf{x}_{j,k,l}, t_i) \approx \frac{c_{m,j,k,l}^{i+1} - c_{m,j,k,l}^i}{\Delta t}, \quad (2.10a)$$

$$\Delta c_m(\mathbf{x}_{j,k,l}, t_i) \approx \frac{c_{m,j+1,k,l}^{i+1} - 2c_{m,j,k,l}^{i+1} + c_{m,j-1,k,l}^{i+1}}{\Delta x_1^2} + \frac{c_{m,j,k+1,l}^{i+1} - 2c_{m,j,k,l}^{i+1} + c_{m,j,k-1,l}^{i+1}}{\Delta x_2^2} \quad (2.10b)$$

$$+ \frac{c_{m,j,k,l+1}^{i+1} - 2c_{m,j,k,l}^{i+1} + c_{m,j,k,l-1}^{i+1}}{\Delta x_3^2}, \quad (2.10c)$$

kde časová derivace je nahrazena zpětnou časovou diferencí, druhá prostorová derivace je nahrazena prostorovou centrální diferencí 2. řádu a $j \in \{1, 2, \dots, N_{x_1} - 2\}$, $k \in \{1, 2, \dots, N_{x_2} - 2\}$, $l \in \{1, 2, \dots, N_{x_3} - 2\}$ a $i \in \{0, 1, 2, \dots, N_t - 2\}$.

Aplikací aproximací (2.10) na rovnici (1.3) dostaneme následující tvar numerického schématu pro mimocévní prostředí

$$\begin{aligned} \frac{c_{m,j,k,l}^{i+1} - c_{m,j,k,l}^i}{\Delta t} = & D_m \frac{c_{m,j+1,k,l}^{i+1} - 2c_{m,j,k,l}^{i+1} + c_{m,j-1,k,l}^{i+1}}{\Delta x_1^2} + D_m \frac{c_{m,j,k+1,l}^{i+1} - 2c_{m,j,k,l}^{i+1} + c_{m,j,k-1,l}^{i+1}}{\Delta x_2^2} \\ & + D_m \frac{c_{m,j,k,l+1}^{i+1} - 2c_{m,j,k,l}^{i+1} + c_{m,j,k,l-1}^{i+1}}{\Delta x_3^2} + g_{m,j,k,l}^i, \end{aligned} \quad (2.11)$$

kde $j \in \{1, 2, \dots, N_{x_1} - 2\}$, $k \in \{1, 2, \dots, N_{x_2} - 2\}$, $l \in \{1, 2, \dots, N_{x_3} - 2\}$, $i \in \{0, 1, \dots, N_t - 2\}$ a $g_{m,j,k,l}^i = g_m(\mathbf{x}_{j,k,l}, t_i)$ je zdrojový člen, jehož aproximace bude podrobně popsána v 2.3.2.

Rovnice (2.11) se dá vektorově zapsat do tvaru

$$\mathbb{B}_m \mathbf{c}_m^{i+1} = \mathbf{c}_m^i + \Delta t \mathbf{g}_m^i \quad i \in \{0, 1, \dots, N_t - 2\}, \quad (2.12)$$

kde matice \mathbb{B}_m je nezávislá na čase t a faktorizace této matice bude podrobněji popsána v kapitole 3.

2.1.3 Množství látky v mimocévním prostředí

Nyní definujeme celkové množství látky M_m v mimocévním prostředí Ω_m v čase $t \in \hat{I}$ vztahem, viz [11]

$$M_m(t) = \int_{\Omega_m} c_m(\mathbf{x}, t) d\mathbf{x} \approx \sum_{j=0}^{N_{x_1}-1} \sum_{k=0}^{N_{x_2}-1} \sum_{l=0}^{N_{x_3}-1} c_m(\mathbf{x}_{j,k,l}, t) \Delta x_1 \Delta x_2 \Delta x_3. \quad (2.13)$$

2.2 Cévní řečiště

V této sekci popíšeme metodu konečných objemů pro úlohu (1.35a) popsanou v sekci 1.6.

2.2.1 Diskretizace cévního řečiště

Nechť $e \in E$ je hrana v grafu Γ , nyní zavedeme diskretizaci $\hat{\Gamma}$ grafu Γ . Nejdříve diskretizujeme parametrickou množinu S_e pro hranu $e \in \Gamma$ následovně

$$\hat{S}_e = \{s_{\ell,e} = \ell \Delta S_e \mid \ell \in \{0, 1, \dots, N_{c,e} - 1\}\} \subset \langle 0, L_e \rangle, \quad (2.14)$$

$$\hat{S}_e = \{s_{\ell,e} = \ell \Delta S_e \mid \ell \in \{1, 2, \dots, N_{c,e} - 2\}\} \subset (0, L_e), \quad (2.15)$$

kde $N_{c,e}$ je počet uzlů na hraně $e \in E(\Gamma)$ a ΔS_e je prostorový krok na hraně e dán vztahem

$$\Delta S_e = \frac{L_e}{N_{c,e} - 1}. \quad (2.16)$$

Poznamenejme, že pro hranu $e \in E(\hat{\Gamma})$ ve tvaru $e = \{w_1, w_2\}$, kde w_1, w_2 jsou její okrajové vrcholy, představuje ΔS_e vzdálenost vrcholů w_1 a w_2 .

Cévní řečiště jsme v první kapitole reprezentovali grafem Γ , nyní diskretizované cévní řečiště značené $\hat{\Omega}_c$ budeme reprezentovat grafem $\hat{\Gamma}$, viz obrázek 2.2 pro hodnotu $N_{c,e} = 4, \forall e \in E(\Gamma)$. Pro lepší přehlednost budeme od teď explicitně psát, z jakého grafu daný vrchol nebo hranu uvažujeme, tj. pro vrchol v grafu Γ , resp. $\hat{\Gamma}$ bude platit $v \in V(\Gamma)$, resp. $v \in V(\hat{\Gamma})$. Obdobný vztah bude platit pro hrany grafů Γ a $\hat{\Gamma}$. Diskretizace cévního řečiště Ω_c je dána

$$\hat{\Omega}_c = (V(\hat{\Gamma}), E(\hat{\Gamma})), \quad (2.17)$$

$$V(\hat{\Gamma}) = V(\Gamma) \cup \{\mathbf{X}_e(s) \mid s \in \hat{S}_e, e \in E(\Gamma)\}, \quad (2.18)$$

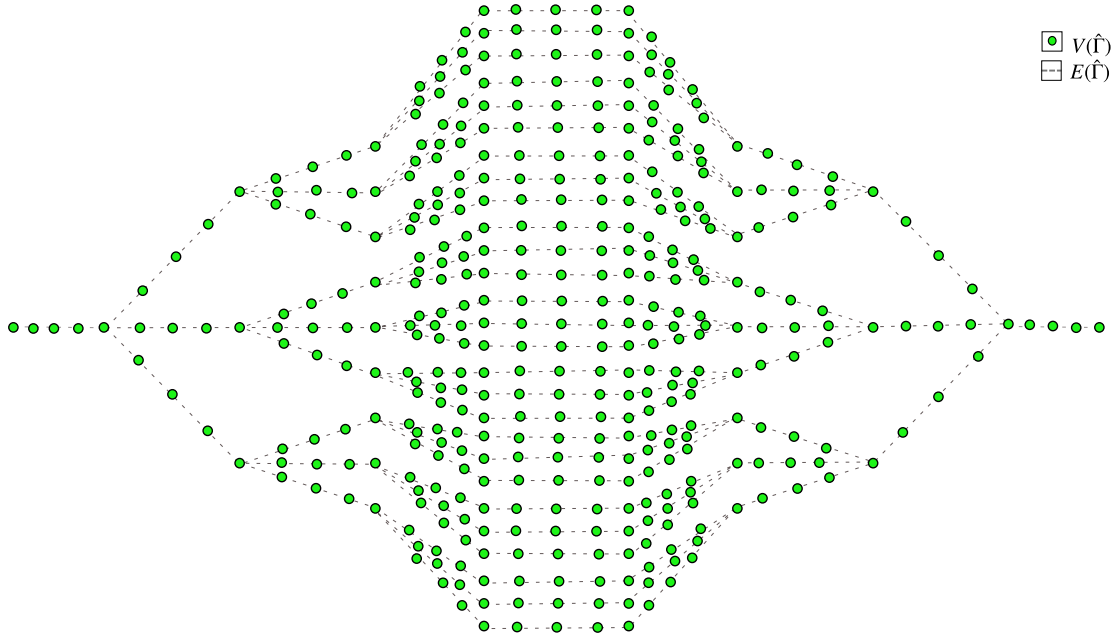
$$E(\hat{\Gamma}) = \{\{\mathbf{X}_e(s_{\ell,e}), \mathbf{X}_e(s_{\ell+1,e})\} \mid s_{\ell,e}, s_{\ell+1,e} \in \hat{S}_e, \ell \in \{0, 1, \dots, N_{c,e} - 2\}, e \in E(\Gamma)\}, \quad (2.19)$$

$$\partial\hat{\Omega}_c = \{v \in V(\hat{\Gamma}) \mid \deg_{\hat{\Gamma}}(v) = 1\}. \quad (2.20)$$

Dále definujeme celkový počet vrcholů N_c v diskretizovaném cévním řečišti $\hat{\Omega}_c$ vztahem

$$N_c = \#\{v \mid v \in V(\hat{\Gamma})\}, \quad (2.21)$$

kde symbol $\#$ označuje počet prvků množiny.



Obrázek 2.2: Diskretizace oblasti ($\#E(\Gamma) = 107, \#V(\Gamma) = 138, \#E(\hat{\Gamma}) = 428, \#V(\hat{\Gamma}) = 459$) Ω_c (reprezentována grafem $\hat{\Gamma}$) pro $N_{c,e} = 4 \forall e \in E(\Gamma)$. Poznamenejme, že zeleně jsou znázorněny vrcholy grafu $\hat{\Gamma}$ a čárkovanou čarou jsou znázorněny hrany grafu $\hat{\Gamma}$.

2.2.2 Diskretizace diferenciálních výrazů v cévním řečišti

Mějme hranu $e \in E(\hat{\Gamma})$, která reprezentuje trubku o délce L_e a průřezu A_e . V metodě konečných objemů budeme uvažovat vždy objem centrováný kolem nějakého vrcholu $v \in V(\hat{\Gamma})$ (VCFVM - Vertex-Centered Finite Volume Method) a tento objem budeme značit V_v .

Nejdříve advektivně-difuzní rovnici (1.19) zintegrujeme přes objem V_v a časový interval $\langle t_i, t_{i+1} \rangle$, a poté zavedeme aproximaci každého členu. Integrací dostaneme $\forall i \in \{0, 1, \dots, N_t - 2\}$

$$\int_{t_i}^{t_{i+1}} \int_{V_v} \frac{\partial c_c}{\partial t} dV dt + \int_{t_i}^{t_{i+1}} \int_{V_v} \nabla \cdot (c_c \mathbf{u}_c) dV dt = \int_{t_i}^{t_{i+1}} \int_{V_v} \nabla \cdot (D_c \nabla c_c) dV dt + \int_{t_i}^{t_{i+1}} \int_{V_v} g_c dV dt. \quad (2.22)$$

Pomocí Greenovy formule [1] převedeme rovnici (2.22) na následující tvar

$$\int_{t_i}^{t_{i+1}} \int_{V_v} \frac{\partial c_c}{\partial t} dV dt + \int_{t_i}^{t_{i+1}} \int_{\partial V_v} c_c \mathbf{u}_c \cdot \mathbf{n} dA dt = \int_{t_i}^{t_{i+1}} \int_{\partial V_v} D_c \nabla c_c \cdot \mathbf{n} dA dt + \int_{t_i}^{t_{i+1}} \int_{V_v} g_c dV dt, \quad (2.23)$$

kde \mathbf{n} je vnější normála na hranici objemu V_v a ∂V_v je hranice objemu V_v .

Aproximace integrálů v rovnici (2.23) pro $c_c(\mathbf{X}(v), t_i) = c_{c,v}^i$ ve vrcholu $v \in V(\hat{\Gamma})$ jsou dány

$$\int_{t_i}^{t_{i+1}} \int_{V_v} \frac{\partial c_c}{\partial t}(v, t_i) dV \approx |V_v|(\tilde{c}_{c,v}^{i+1} - \tilde{c}_{c,v}^i), \quad (2.24a)$$

$$\int_{t_i}^{t_{i+1}} \int_{\partial V_v} c_c \mathbf{u}_c \cdot \mathbf{n} dA \approx \Delta t \sum_{w \in \mathcal{N}(v)} A_{\{v,w\}} u_{c,\{v,w\}} \tilde{C}_{c,\{v,w\}}^{(upw),i+1}, \quad (2.24b)$$

$$\int_{t_i}^{t_{i+1}} \int_{\partial V_v} D_c \nabla c_c \cdot \mathbf{n} dA \approx \Delta t \sum_{w \in \mathcal{N}(v)} D_c A_{\{v,w\}} \frac{\tilde{c}_{c,w}^{i+1} - \tilde{c}_{c,v}^{i+1}}{\Delta S_{\{v,w\}}}, \quad (2.24c)$$

$$\int_{t_i}^{t_{i+1}} \int_{V_v} g_c dV dt \approx g_{c,v}^i \Delta t |V_v|, \quad (2.24d)$$

kde $\tilde{c}_{c,v}^i$ je střední hodnota koncentrace v i -tém časovém kroku v objemu V_v . V rovnicích (2.24) byly diferenciální výrazy aproximovány diferenčními [1]. Dále v rovnici (2.24b) po aproximaci integrálu přes povrch válcové trubky uvažujeme pouze člen reprezentující tok přes průřez $A_{\{v,w\}}$ trubky. Tok přes plášť (stěnu) je řešen zdrojovým členem g_c v podobě přestupu KL do mimocévného prostředí.

Funkce $\tilde{C}_{c,\{v,w\}}^{(upw),i+1}$ představuje upwindovou (proti směru) stabilizaci a je dána

$$\tilde{C}_{c,\{v,w\}}^{(upw),i+1} = \begin{cases} \tilde{c}_{c,w}^{i+1} & \text{pro } u_{c,\{v,w\}} < 0, \\ \tilde{c}_{c,v}^{i+1} & \text{pro } u_{c,\{v,w\}} \geq 0, \end{cases} \quad (2.25)$$

Po nahrazení diferenciálních členů v rovnici (2.23) aproximacemi (2.24) a vydělením Δt dostaneme následující rovnici

$$|V_v| \frac{\tilde{c}_{c,v}^{i+1} - \tilde{c}_{c,v}^i}{\Delta t} = \sum_{w \in \mathcal{N}(v)} A_{\{v,w\}} u_{c,\{v,w\}} \tilde{C}_{c,\{v,w\}}^{(upw),i+1} + \sum_{w \in \mathcal{N}(v)} D_c A_{\{v,w\}} \frac{\tilde{c}_{c,w}^{i+1} - \tilde{c}_{c,v}^{i+1}}{\Delta S_{\{v,w\}}} + |V_v| g_{c,v}^i, \quad (2.26)$$

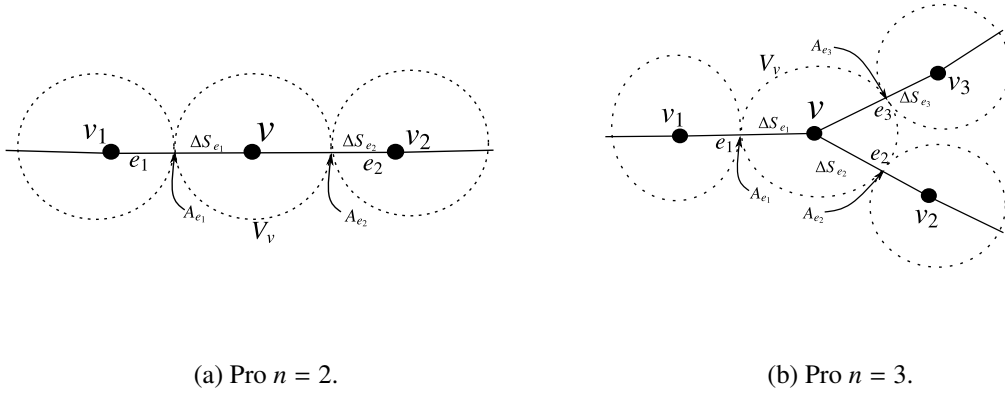
Člen $g_{c,v}^i = g_c(v, t_i)$ v (2.26) představuje aproximaci zdrojového členu, který popíšeme v sekci 2.3.2. Rovnice (2.26) se dá maticově zapsat ve tvaru

$$\mathbb{B}_c \mathbf{c}_c^{i+1} = \mathbf{c}_c^i + \Delta t \mathbf{g}_c^i \quad i \in \{0, 1, \dots, N_t - 2\}. \quad (2.27)$$

Poznanejme, že matice \mathbb{B}_c v rovnici (2.26) je v této práci konstantní, tudíž matici \mathbb{B}_c sestavíme pouze jednou. V kapitole 3 podrobněji popíšeme LU faktorizaci matice \mathbb{B}_c pomocí knihovny UMFPACK [17].

V rovnici (2.26) se vyskytuje velikost objemu centrovaného okolo daného vrcholu. Velikost objemu $|V_v|$ ve vrcholu v je uvažována následovně: necht' vrchol $v \in V$ je součástí hran e_1, e_2, \dots, e_n , kde $n \in \mathbb{N}$, pak velikost objemu $|V_v|$ ve vrcholu v je dána vztahem

$$|V_v| = \sum_{i=1}^n \frac{A_{e_i} \Delta S_{e_i}}{2}. \quad (2.28)$$



Obrázek 2.3: Ilustrace prostorové diskretizace metodou konečných objemů.

Na obrázku 2.3 jsou vyznačeny potřebné veličiny pro dosažení do rovnice (2.28) pro dva případy (V^2 na obrázku 2.3a a V^k na obrázku 2.3b). Pro oba případy je vrchol v takový vrchol, ve kterém uvažujeme objem V_v . Vrcholy v_1, v_2 na obrázku 2.3a, resp. v_1, v_2, v_3 na obrázku 2.3b jsou sousedními vrcholy vrcholu v . ΔS_{e_i} je délka hrany mezi vrcholem v a v_i a A_{e_i} je průřez hrany e_i , $i \in \{1, 2, 3\}$.

2.2.3 Množství látky v cévním řečišti

Celkové množství látky M_c v cévním řečišti Ω_c v čase $t \in \hat{I}$ aproximujeme vztahem

$$M_c(t) \approx \sum_{e \in E(\hat{\Gamma})} \frac{c_c(X_e(0), t) + c_c(X_e(L_e), t)}{2} A_e \Delta S_e, \quad (2.29)$$

kde $X_e(0), X_e(L_e)$ jsou okrajové vrcholy hrany $e \in E(\hat{\Gamma})$.

2.3 Přestup KL

Přestup KL mezi cévním a mimocévním prostředím je dán zdrojovými členy. V této části popíšeme nejdříve, kdy dochází k přestupu KL. Dále zavedeme aproximaci Diracovy delta funkce a na závěr diskretizujeme zdrojové členy. V této sekci bude přestup KL popsán pro hranu $e \in E(\Gamma)$.

K přestupu KL dochází mezi eulerovským a lagrangeovským bodem, které jsou dostatečně blízko sebe. Jak blízko od sebe se potřebují nacházet, aby docházelo k přestupu KL, závisí na volbě aproximace Diracovy delta funkce popsané později v (2.34). Pro zdrojové členy g_c , resp. g_m budou platit následující vztahy z (1.21)

$$g_m(\mathbf{x}, t) = \int_{S_e} k_m (c_c(\mathbf{X}_e(s), t) - c_m(\mathbf{x}, t)) \delta(\mathbf{x} - \mathbf{X}_e(s)) ds \quad \forall \mathbf{x} \in \Omega_m, \forall t \in I, \quad (2.30)$$

resp.

$$g_c(\mathbf{X}_e(s), t) = \int_{\Omega_m} k_c (c_m(\mathbf{x}, t) - c_c(\mathbf{X}_e(s), t)) \delta(\mathbf{X}_e(s) - \mathbf{x}) d\mathbf{x} \quad \forall s \in S_e, \forall t \in I. \quad (2.31)$$

Rovnice (2.30) a (2.31) představují konvoluci a jednoduchou vrstvu.

2.3.1 Aproximace Diracovy delta funkce

Konvoluci, resp. jednoduchou vrstvu v (2.30), resp. (2.31) aproximujeme následovně

$$g_m(\mathbf{x}, t) \approx k_m \int_{S_e} (c_c(\mathbf{X}_e(s), t) - c_m(\mathbf{x}, t)) \Psi(\mathbf{x} - \mathbf{X}_e(s)) ds \quad \forall (\mathbf{x}, t) \in \Omega_m \times I, \quad (2.32a)$$

$$g_c(\mathbf{X}_e(s), t) \approx k_c \int_{\Omega_m} (c_m(\mathbf{x}, t) - c_c(\mathbf{X}_e(s), t)) \Psi(\mathbf{X}_e(s) - \mathbf{x}) d\mathbf{x} \quad \forall (s, t) \in S_e \times I, \quad (2.32b)$$

kde Ψ je spojitá aproximace Diracovy delta funkce [5] daná vztahem

$$\Psi(\mathbf{z}) = \prod_{d=1}^3 \frac{1}{\Delta x_d} \varphi\left(\frac{z_d}{\Delta x_d}\right), \quad (2.33)$$

kde z_d je d -tá složka vektoru \mathbf{z} , Δx_d je prostorový krok na eulerovské ose x_d . Dále φ je spojitá jedno-rozměrná aproximace Diracovy delta funkce.

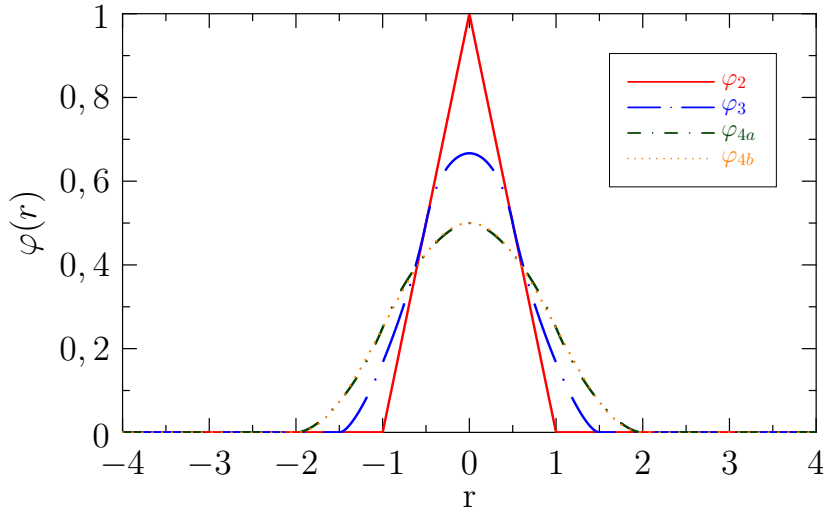
V bakalářské práci [10] jsme použili 4 tvary aproximace Diracovy delta funkce definované v rovnicích (2.34) pro $r = \frac{z_d}{\Delta x_d}$ [3]. Průběh spojitých aproximací Diracovy delta funkce je vyobrazen na obrázku 2.4. Zjistili jsme, že všechny aproximace Diracovy delta funkce dávají na dostatečně jemné síti stejné výsledky, proto v této práci budeme pro jednoduchost pro aproximaci Diracovy delta funkce používat pouze funkci φ_{4b} (viz (2.34d)).

$$\varphi_2(r) = \begin{cases} 1 - |r| \\ 0 \end{cases} \quad \begin{array}{l} \text{pro } |r| \leq 1, \\ \text{pro } |r| \geq 1, \end{array} \quad (2.34a)$$

$$\varphi_3(r) = \begin{cases} \frac{1}{3} (1 + \sqrt{1 - 3r^2}) \\ \frac{1}{6} (5 - 3|r| - \sqrt{-2 + 6|r| - 3r^2}) \\ 0 \end{cases} \quad \begin{array}{l} \text{pro } |r| \leq \frac{1}{2}, \\ \text{pro } |r| \in \left(\frac{1}{2}, \frac{3}{2}\right), \\ \text{pro } |r| \geq \frac{3}{2}, \end{array} \quad (2.34b)$$

$$\varphi_{4a}(r) = \begin{cases} \frac{1}{8} (3 - 2|r| + \sqrt{1 + 4|r| - 4r^2}) \\ \frac{1}{8} (5 - 2|r| - \sqrt{-7 + 12|r| - 4r^2}) \\ 0 \end{cases} \quad \begin{array}{l} \text{pro } |r| \leq 1, \\ \text{pro } |r| \in (1, 2), \\ \text{pro } |r| \geq 2, \end{array} \quad (2.34c)$$

$$\varphi_{4b}(r) = \begin{cases} \frac{1}{4} (1 + \cos \frac{\pi r}{2}) \\ 0 \end{cases} \quad \begin{array}{l} \text{pro } |r| \leq 2, \\ \text{pro } |r| \geq 2. \end{array} \quad (2.34d)$$



Obrázek 2.4: Spojité aproximace Diracovy delta funkce (2.34).

2.3.2 Diskretizace zdrojových členů

V této části aproximujeme zdrojové členy $g_{c,\ell}^i = g_c(\mathbf{X}_e(s_\ell), t_i)$ a $g_{m,j,k,l}^i = g_m(\mathbf{x}_{j,k,l}, t_i)$ z rovnic (2.26), resp. (2.11) vztahem (2.32) následovně

$$g_{c,\ell}^i \approx k_c \sum_{j=0}^{N_{x_1}-1} \sum_{k=0}^{N_{x_2}-1} \sum_{l=0}^{N_{x_3}-1} (c_{m,j,k,l}^i - c_{c,\ell}^i) \Psi(\mathbf{X}_e(s_\ell) - \mathbf{x}_{j,k,l}) \Delta x_1 \Delta x_2 \Delta x_3, \quad (2.35a)$$

resp.

$$g_{m,j,k,l}^i \approx k_m \sum_{\ell=0}^{N_c-1} (c_{c,\ell}^i - c_{m,j,k,l}^i) \Psi(\mathbf{x}_{j,k,l} - \mathbf{X}_e(s_\ell)) |V_{s_\ell}|, \quad (2.35b)$$

kde $c_{c,\ell}^i = c_c(\mathbf{X}_e(s_\ell), t_i)$, $c_{m,j,k,l}^i = c_m(\mathbf{x}_{j,k,l}, t_i)$, $e \in E(\hat{\Gamma})$, $\ell \in \{0, 1, \dots, N_c - 1\}$, $j \in \{0, 1, \dots, N_{x_1} - 1\}$, $k \in \{0, 1, \dots, N_{x_2} - 1\}$, $l \in \{0, 1, \dots, N_{x_3} - 1\}$ a $i \in \{0, 1, 2, \dots, N_t - 1\}$.

2.4 Generování řečiště

Dalším cílem této práce bylo navrhnout způsob generování cévního řečiště. Algoritmus generování řečiště je založen na několika krocích. Důležitým parametrem je *level* (úroveň) generování grafu Γ . *Level* určuje počet kroků (ozn. *step*) algoritmu generování grafu Γ .

1. V nultém kroku (ozn. *step* = 0) přidáme vrchol $v_{in} \in V^{in}$. Vrcholu v_{in} přiřadíme souřadnice $(x_1^{(in)}, x_2^{(in)}, x_3^{(in)})^T$ a Dirichletovu okrajovou podmínku viz (1.37b). Dále přidáme vrchol $v_1 \in V^{int}$ se souřadnicemi $(x_1^{(1)}, x_2^{(1)}, x_3^{(1)})^T$ a tento vrchol spojíme hranou e_1 s vrcholem v_{in} . Tento krok zajistí, že stupeň krajního vrcholu v_{in} v grafu Γ bude roven 1. Hraně e_1 přiřadíme průřez $A_{e_1} = 1$ (normalizovaný). Pomocí souřadnic vrcholů v_{in} a v_1 spočítáme délku hrany

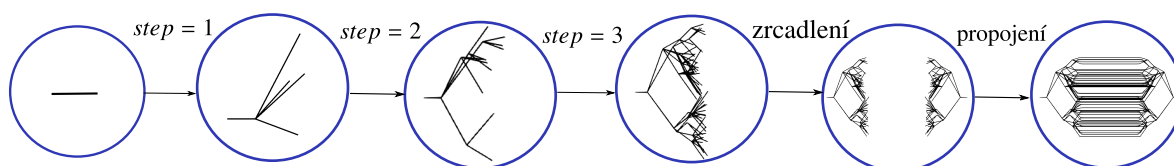
$$L_{e_1} = \sqrt{(x_1^{(in)} - x_1^{(1)})^2 + (x_2^{(in)} - x_2^{(1)})^2 + (x_3^{(in)} - x_3^{(1)})^2}. \quad (2.36)$$

2. Pro $step = 1$ se zvolí náhodné číslo $p \in \mathbb{N}, p < 5$ a vytvoří se p vrcholů v_2, \dots, v_{p+1} . Dále se přidá p hran e_2, \dots, e_{p+1} , kde hrana e_i spojuje vrchol v_1 a v_i pro $i \in \{2, \dots, p+1\}$. Obdobným způsobem jako v rovnici (2.36) se spočítá délka nově vytvořených hran. Průřez hrany e_i pro $i \in \{2, \dots, p+1\}$ je dán vztahem

$$A_{e_i} = \frac{A_{e_1}}{p}. \quad (2.37)$$

3. Další krok se provádí obdobně pro vrcholy $v_{p+2}, \dots, v_{p+2+r}$, $r \in \mathbb{N}$ jako krok 2. Tímto způsobem vygenerujeme nové vrcholy a hrany. Tento postup pokračuje, dokud je $step$ menší než $level$.
4. Doposud jsme vygenerovali téměř polovinu grafu Γ . V dalším kroku $step = level + 1$ se již vytvořený graf zrcadlově překlopí podle nadroviny $x_1 = 0$ viz obrázek 2.5, této fázi říkáme zrcadlení. V posledním kroku se původní graf a zrcadlený graf propojí hranami a tím získáváme finální graf Γ . Tento poslední krok nazýváme propojení. Na závěr se krajnímu vrcholu v_{out} vytvořenému procesem zrcadlení nastaví Neumannova okrajová podmínka (1.37c).

Příklad postupu generování řečiště pro $level = 2$ je vyobrazen na obrázku 2.5.



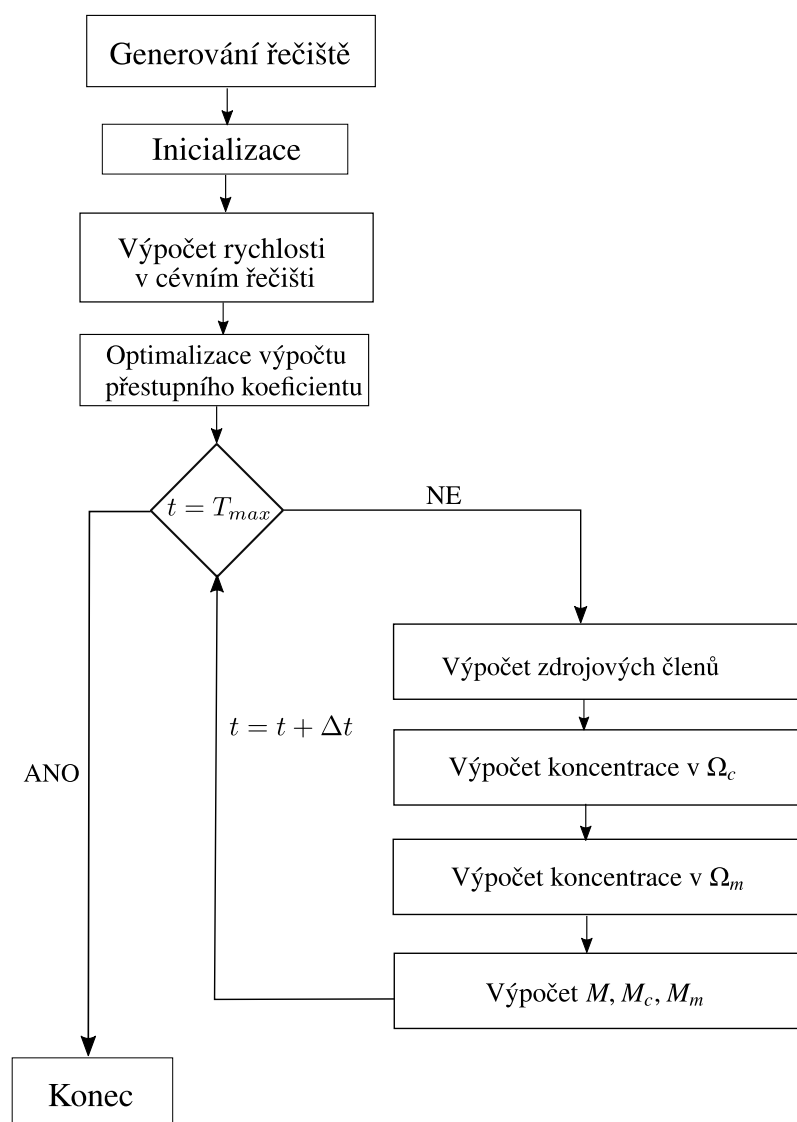
Obrázek 2.5: Ilustrace generování řečiště pro $level = 2$.

2.5 Algoritmus

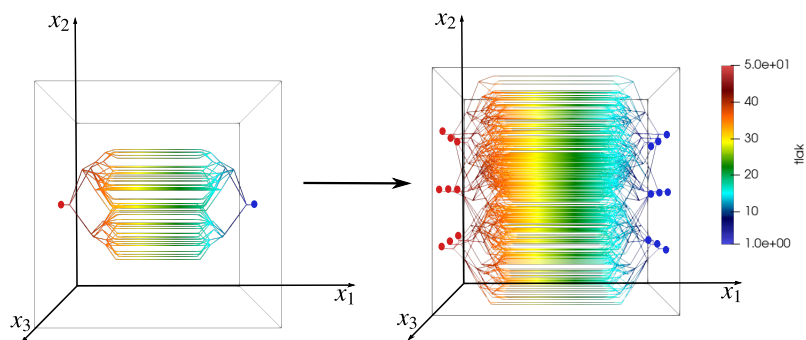
V této části popíšeme kompletní algoritmus numerického schématu pro výpočet koncentrací v cévním a mimocévním prostředí. Schéma celého algoritmu je vyobrazené na obrázku 2.6 a jeho jednotlivé kroky jsou následující:

1. **Generování řečiště.** V prvním kroku algoritmu se vygeneruje cévní řečiště postupem popsaným v sekci 2.4.
2. **Inicializace.** V tomto kroku algoritmu se nastaví počáteční a okrajové podmínky metody konečných objemů v cévním řečišti a konečných diferencí v mimocévním prostředí.
3. **Výpočet rychlostí v cévním řečišti.** Dále se vypočítají rychlosti na hranách grafu Γ , který reprezentuje cévní řečiště. Výpočet rychlostí v grafu Γ je popsán v sekci 1.3.2.
4. **Optimalizace výpočtu zdrojových členů.** V tomto kroku se ke každému vrcholu $v \in V(\hat{\Gamma})$ přiřadí body z mimocévního prostředí Ω_m , které jsou v jeho blízkosti podle nosiče zvolené Diracovy delta funkce (tzv. blízké vrcholy). Stejný postup se provede pro body z mimocévního prostředí. Při výpočtu přestupního koeficientu, který se napočítává v každém čase t , se bude procházet okolí pouze přes blízké vrcholy, viz podrobnější popis v kapitole 3.

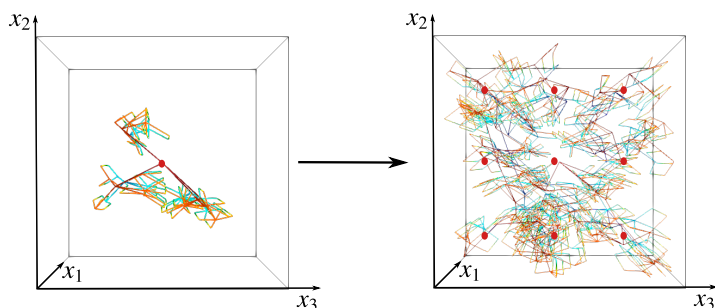
5. **Výpočet koncentrací a zdrojových členů.** V tomto kroku se nejprve pomocí metody vnořené hranice 1.4.1 vypočítají zdrojové členy z hodnot koncentrací z předešlé časové vrstvy. Poté se napočítají hodnoty koncentrací v cévním řečišti a v mimocévním prostředí na základě numerického schématu metody konečných objemů (2.26) a metody konečných diferencí (2.11).
6. **Výpočet celkového množství látky.** Pomocí vztahu (2.29) se napočítá celkové množství látky v cévním řečišti Ω_c . Pomocí (2.13) se napočítá celkové množství KL v mimocévním prostředí. Kroky 5 a 6 algoritmu se opakují, dokud není dosaženo finálního času T_{max} , jak je znázorněno na obrázku 2.6.



Obrázek 2.6: Schéma algoritmu pro výpočet koncentrací v cévní řečišti Ω_c a v mimocévním prostředí Ω_m .



(a) Pohled ve směru os x_1 a x_2 .



(b) Pohled ve směru os x_2 a x_3 .

Obrázek 2.7: Ilustrace přechodu od jednoho cévního řečiště k 9 cévním řečištím. Červeně jsou označeny vstupní vrcholy s bezrozměrným tlakem $p_A = 50$ a modře výstupní s tlakem $p_B = 1$. Barevná škála odpovídá velikosti tlaku.

2.6 Více cévních řečišť

Doposud jsme uvažovali v myokardu Ω pouze jedno cévní řečiště. Pro větší zahuštění myokardu budeme uvažovat více řečišť generovaných algoritmem 2.4.

Myokard Ω v této práci představuje krychli. Vrchol v_{in} z algoritmu 2.4 se nachází na stěně krychle, kterou budeme nazývat vstupní stěnou s $x_1 = x_1^{(in)}$. Na protější (výstupní) stěně krychle se nachází vrchol v_{out} .

Vstupní stěnu krychle rozdělíme na 9 stejných čtverců. Do středu každého nově vytvořeného čtverce umístíme krajní vrchol v_{in}^i , kde $i \in \{1, 2, \dots, 9\}$. Tímto jsme provedli nultý krok algoritmu generování grafu Γ . Pro každý vrchol v_{in}^i vygenerujeme graf Γ_i pro $i \in \{1, 2, \dots, 9\}$.

Na obrázku 2.7 je znázorněn přechod od jednoho cévního řečiště k 9 cévním řečištím. Při volbě více cévních řečišť je mimocévní prostředí hustěji zaplněné cévami.

Kapitola 3

Implementace numerického schematu

V této kapitole popíšeme stěžejní části implementovaného kódu. Kód lze stáhnout zde [24].

Jedná se o objektový kód psaný v programovacím jazyce C++ s využitím externí knihovny UMFPACK pro řešení řídkých lineárních soustav pomocí LU rozkladu matice. Numerické schéma implementovaného kódu je popsáno v sekci 2.5.

Nejdříve popíšeme soubory v příloženém kódu. Dále definujeme struktury použité při implementaci. Představíme základní strukturu funkce `main()`. Dále popíšeme jednotlivé kroky algoritmu generování cévního řečiště. Poté představíme, jak se plní konstantní matice \mathbb{B}_c z rovnice (2.27) a \mathbb{B}_m z (2.12). Následně popíšeme implementaci zdrojových členů g_c, g_m včetně optimalizace této části.

Implementový kód je složen z následujících souborů:

- `runfile`: soubor, který provádí kompilaci a spouští celý program.
- `Simulation.cpp`: v tomto souboru jsou definovány funkce pro generování cévního řečiště `graph1()` a hlavní funkce `main()`.
- `NodesEdges.h`: obsahuje strukturu pro vrchol `Node` a hranu `Edge` cévního řečiště.
- `VascularEnvironment.h`: obsahuje strukturu pro cévní řečiště VE.
- `ExtravascularEnvironment.h`: obsahuje strukturu pro mimocévní prostředí EE.
- `SparseMatrixUMFPACK.h`: soubor definuje strukturu `SparseMatrix` potřebnou pro řešení lineárních rovnic pomocí UMFPACKu.
- `Interface.h`: definuje strukturu `IE`, ve které najdeme funkce pro výpočet zdrojových členů g_c, g_m .
- `Convolution.h`: součástí tohoto souboru je struktura `Convolution`, ve které je zadefinována Diracova delta funkce a její spojitě aproximace.

Pro větší přehlednost vypíšeme souhrn používaných struktur a jejich využití:

- `Node`: třída `Node` je základní datovou strukturou pro reprezentaci vrcholů v cévním řečišti. Každý vrchol má následující atributy:
 - **Souřadnice** (x_1, x_2, x_3): Reprezentují polohu vrcholu v trojrozměrném prostoru.
 - **Tlak** (pressure): Skalární veličina reprezentující tlak v daném vrcholu.

- **Koncentrace** (concentration): Skalární veličina reprezentující koncentraci KL v daném vrcholu.
- **Objem** (volume): Objem centrováný kolem daného vrcholu v cévním řečišti.
- **Vektor sousedních vrcholů** (adjacentNodes): Obsahuje indexy sousedních vrcholů v cévním řečišti.
- **Vektor sousedních hran** (adjacentEdges): Obsahuje indexy hran, pro které je daný vrchol jejich okrajovým vrcholem.
- **Vektor sousedních vrcholů v mimocévním prostředí** (adjacent_EE_Nodes): Vektor obsahující indexy sousedních vrcholů nacházejících se v mimocévním prostředí.

Typy vrcholů (boundary_condition): Každý vrchol může být klasifikován jako:

- 0: Vnitřní vrchol bez speciálních okrajových podmínek.
- 1: Vrchol s Dirichletovou okrajovou podmínkou pro koncentraci.
- 2: Vrchol s Neumannovou okrajovou podmínkou pro koncentraci.

Struktura Node je definována v hlavičkovém souboru `VascularEnvironment.h` a je užívána prostřednictvím `NodeV`, který reprezentuje vektor těchto vrcholů, označený jako `V`.

- **Edge**: základní datová struktura pro reprezentaci hran v cévním řečišti. Každá hrana má následující atributy:
 - **Indexy okrajových vrcholů** (v_1, v_2): Indexy okrajových vrcholů dané hrany.
 - **Průřez hrany** (A): Průřez hrany.
 - **Délka hrany** (L): Délka hrany.
 - **Rychlost** (u): Rychlost na hraně.

Struktura Edge je definována v hlavičkovém souboru `VascularEnvironment.h` a je využívána prostřednictvím `EdgeE`, který reprezentuje vektor těchto hran, označený jako `E`.

- **SparseMatrix**: třída `SparseMatrix` je navržena pro efektivní práci s řídkými maticemi a umožňuje řešení lineárních soustav rovnic s využitím knihovny `UMFPACK`. Struktura má následující klíčové atributy a funkce:
 - **Velikost matice** (N): Celkový počet řádků a sloupců v matici.
 - **Počet nenulových prvků** (NCC): Počet nenulových prvků uložených v matici.
 - **Symbolická a numerická faktorizace** (Symbolic a Numeric): Ukazatelé na struktury používané pro efektivní faktorizaci a řešení matice.
 - **Indexy řádků nenulových prvků** (A_i): Pole indexů řádků pro každý nenulový prvek.
 - **Indexy sloupců nenulových prvků** (A_p): Pole, které označuje začátek každého sloupce v seznamu nenulových prvků.
 - **Hodnoty nenulových prvků** (A_x): Pole obsahující všechny nenulové hodnoty matice.

Mezi základní metody třídy patří:

- `solve(T *b, T *x)`: Metoda pro řešení systému lineárních rovnic $Ax = b$.

- `allocate()`: Alokuje paměť pro nenulové prvky matice.
- `get(int i, int j)`: Vrací hodnotu prvku na pozici (i, j) .
- `printMatrix()`: Vypíše matici ve standardní formě.

Struktura `SparseMatrix` je využívána pro výpočet adjacenční matice, pro řešení systémů tlaků v cévním řečišti a pro výpočet koncentrací v cévním řečišti (`VE_SparseMatrix`) a mimocévním prostředí (`EE_SparseMatrix`), vše definováno v souborech `VascularEnvironment.h` a `ExtravascularEnvironment.h`.

- **VE**: jedná se o strukturu obsahující prvky z cévního řečiště. Je zde sestavena adjacenční matice, matice pro tlaky a koncentrace. Dále je zde definována metoda pro výpočet tlaků a koncentrací v cévním řečišti.

Tato struktura je v kódu využita ve dvou instancích. První instancí je `ve1` pro vygenerování cévního řečiště a následného dopočítání tlaku v každém vrcholu. Dále při diskretizaci cévního řečiště vytváříme instanci `ve2`, která obsahuje všechny vrcholy instance `ve1` a nově vytvořené vrcholy a hrany z diskretizace. V rámci matematického a numerického modelu instance `ve1` odpovídá grafu Γ a instance `ve2` odpovídá grafu $\hat{\Gamma}$. Obě dvě instance `ve1` a `ve2` jsou definovány v `Interface.h`.

- **EE**: má podobnou funkcionalitu jako struktura **VE**. Definuje funkce pro naplnění matice pro koncentrace a její řešení, indexovací funkci pro každý bod v mimocévním prostředí. V kódu je definována ve značení `ee` v souboru `Interface.h`.
- **IE**: struktura `ie` je deklarována v souboru `Interface.h` a definována v `Simulation.cpp`. Hlavní prvky této struktury zahrnují funkce pro výpočet zdrojových členů g_c a g_m .
- **Convolution**: struktura `convolution`, definovaná v souboru `Interface.h`, obsahuje funkci `diracDelta()` pro výpočet aproximace Diracovy delta funkce. Dále zahrnuje funkci `approximationFunction()`, která definuje spojitě aproximace Diracovy delta funkce, jak je ukázáno na obrázku 2.4.

3.1 Simulation.cpp

V této části popíšeme hlavní funkci `main()`, funkci pro vygenerování cévního řečiště `graph1()` a funkci `setup_VE()` ze souboru `Simulation.cpp`. Ostatní funkce definované v tomto souboru lze najít v příloženém kódu [24].

Numerické schéma funkce `main()` je znázorněno na obrázku 2.6. Nejdříve z kompilovacího souboru `runfile` se převezmou parametry pro spuštění programu, poté dochází k nastavení prostorových kroků a časového kroku, počtu bodů diskretizace na jednotlivých osách. Dále pomocí funkce `setup_VE()` dochází k vygenerování cévního řečiště, spočítání tlau a rychlostí a naplnění konstantní matice z (2.27) a pomocí funkce `ee.solve()` se nastaví mimocévní prostředí společně s naplněním matice z (2.12). Následně se alokují pole pro zdrojové členy g_c, g_m a zvolí se typ spojitě aproximace Diracovy delta funkce (2.34d). V další části kódu napočítáme počáteční hodnoty celkových množství KL M, M_c, M_m .

Funkce `ie.neighbours_in_VE()`, resp. `ie.neighbours_in_EE()` každému bodu z mimocévního prostředí, resp. cévního řečiště přiřadí body v cévním řečišti, resp. mimocévním prostředí, které

jsou v blízkosti daného bodu. Definice těchto funkcí a pojmu blízkost bodu od jiného bodu, bude popsána níže.

Následuje časová smyčka, ve které se nejdříve vypočítají zdrojové členy g_c, g_m na základě hodnot koncentrace z předchozí časové vrstvy. Poté dochází k výpočtu koncentrací c_c a c_m . Na závěr se vypočítají celková množství KL a aktualizuje se čas.

```

1  int main(int argc, char **argv) {
2      // Checking if the required number of parameters is provided
3      const int params = 3;
4      if (argc <= params) {
5          printf("error: required %d parameters: \n%s\n", params, argv[0]);
6          return 1;
7      }
8      // Retrieving parameters from command line arguments
9      int N = atoi(argv[1]);          // Number of discretization points
10     double k = atof(argv[2]);      // Transfer coefficient
11     double d = atof(argv[3]);      // Defect coefficient
12     // Setting up space steps, diffusion, number of nodes in each axis
13     initialization(N, k);
14     // Setting up the vascular bed and setting up constant matrix B_c
15     setup_VE(N, dt, diffusion_c, d);
16     // Setting up the extravascular environment and setting up constant matrix B_m
17     ee.solve();
18     // Allocating memory for source terms g_c and g_m
19     ie.allocate_g();
20     // Convolution approximation - choose of type psi_4b
21     convolution.n = 4;
22     // Initial mass calculation
23     double Mc = 0;
24     double Mm = 0;
25     double M = 0;
26     Mc = calculate_mass_VE();
27     Mm = ee.calculate_mass_EE(ee.c_m);
28     M = Mc + Mm;
29     // Initial time t_0 plus time step dt
30     double time = t_0 + dt;
31     // Setting up neighbours of each vertex in both environments
32     ie.neighbours_in_VE();
33     ie.neighbours_in_EE();
34     // Time loop for concentration calculation
35     while (time <= t_N + dt) {
36         // Calculating source terms
37         ie.function_g_c(time);
38         ie.function_g_m(time);
39         // Solving concentration
40         ve2.solveConcentration(dt, time, ie.g_c);
41         ee.solve_concentration_m(ie.g_m);
42         // Calculating mass quantities
43         Mc = calculate_mass_VE();
44         Mm = ee.calculate_mass_EE(ee.c_m);
45         M = Mc + Mm;
46         // Setting up new time
47         time = time + dt;
48     }
49     return 0;
50 }

```

Následující kód definujeme funkce `setup_VE()`, která nejdříve pomocí funkce `graph1()` zkonstruuje cévní řečiště reprezentované grafem Γ a vypočítá tlaky. Podrobněji bude tato funkce popsána níže. Dále funkce `graph2()` diskretizací grafu Γ zkonstruuje graf $\hat{\Gamma}$. Diskretizace probíhá tak, že každou hranu grafu Γ zdiskretizuje N vrcholy, které následně propojí hranami. Detaily této funkce lze najít v příloženém kódu [24]. Dále dojde k nastavení počáteční podmínky ve všech vrcholech struktury `ve2`. Funkce `ve2.buildConcentrationMatrix()` alokuje prvky struktury `VE_SparseMatrix` pro matici \mathbb{B}_c a funkce `ve2.fillConcentrationMatrix()` tuto matici naplní. Poslední zmíněná funkce bude podrobně popsána v další části této kapitoly. Přesnou definici ostatních funkcí naleznete v [24].

```

1 void setup_VE(int N, double dt, double D_c, double d) {
2     // Creating the graph and discretizing it
3     graph1(d, 2); // Corresponds to structure ve1
4     graph2(N); // Corresponds to structure ve2 - discretized ve1
5     // Initializing the concentration values and constructing the concentration matrix
6     ve2.initialConditions(); // Setting up initial value of concentration to 0
7     ve2.buildConcentrationMatrix(); // Setting up structure of VE_SparseMatrix
8     // Filling the concentration constant matrix
9     ve2.fillConcentrationMatrix(D_c, dt);
10 }

```

Následující blok kód popisuje funkci pro vygenerování cévního řečiště. Rozvětvení cévního řečiště využívá funkci `rand()`. Abychom byly schopni dělat simulace pokaždé na stejném typu cévního řečiště, tak využíváme funkci `srand()`, která fixuje daný typ cévního řečiště pomocí jejího argumentu.

V sekci 2.6 byla popsána motivace pro vygenerování více cévních řečišť. V následující ukázce kódu zvolíme počet cévních řečišť `number`, které se mají vygenerovat následovně

$$\text{number} = \text{number_x2} * \text{number_x3}, \quad (3.1)$$

kde `number_x2` a `number_x3` určují počet cévních řečišť po řadě ve směru osy x_2 a x_3 .

Dále se nastaví souřadnice $x_1^{(in)}$ vrcholu v_{in} ze sekce 1.4.2. Souřadnice $x_2^{(in)}$, $x_3^{(in)}$ jsou určeny v polích `EN_x2[]`, `EN_x3[]`. Na řádce 23 přidáme vrchol v_{in} a vrchol v_1 . Souřadnice $x_2^{(1)}$ a $x_3^{(1)}$ vrcholu v_1 jsou stejné jako souřadnice vrcholu v_{in} , souřadnice $x_1^{(1)}$ je zadána manuálně pomocí `x1_1`. Spojením vrcholů v_{in} a v_1 vytvoříme pomocnou hranu e_1 s průřezem A_{e_1} a délkou danou vztahem (2.36).

Funkce `graph1_branch()` provede kroky algoritmu z 2.4 dokud $step = level + 1$. Více bude tato funkce popsána v dalším kódovém bloku.

Další krok této funkce je krok zrcadlení. Nejdříve si vytvoříme adjacenční matici pro již přidané vrcholy a hrany, poté si uložíme počet vrcholů a hran. Na řádce 41 přidáme `mNodes` vrcholů zrcadlově překlopených a tyto vrcholy spojíme hranou ve stejném uskupení jako do $step = level + 1$.

Krok propojení začíná na řádce 47, kdy projdeme všechny vrcholy v vytvořené před krokem zrcadlení. První podmínkou v `i f` funkci na řádce 49 zajistíme, že se nebude jednat o vrchol v_{in} , druhá podmínka říká, že se musí jednat o vrchol s pouze jedním sousedem. Následně projdeme všechny hrany e vytvořené před krokem zrcadlení. Pokud nastane situace, kdy daný vrchol v bude okrajový vrchol hrany e , tak vrchol v propojíme hranou s protějším vrcholem vzniklým krokem zrcadlení.

Další část této funkce prochází všechny doposud vytvořené vrcholy. Vrcholům, které mají jen jednoho souseda a nejsou vstupními vrcholy, přiřadí Neumannovu okrajovou podmínku.

Poslední část funkce se zabývá nastavením defektní části myokardu. Pomocí velikosti mimocévního prostředí zvolíme oblast, kde upravíme průřez hran. Tím zajistíme to, že transport KL bude v hranách, nacházejících se v defektní části, omezen.

Na závěr vytvoříme adjacenci matrici pro tento graf, spočítáme tlaky v každém vrcholu a následně dopočítáme rychlost na každé hraně grafu.

```

1 void graph1(double d, int level, double reflection_point = 7.) {
2     // Number of step
3     int step = 0;
4     // Seed the random number generator
5     srand(11);
6     // Number of entry nodes in x2 and x3 axis
7     int number_x2 = 3;
8     int number_x3 = 3;
9     // Set x1 coordinates for vertices x1_in and x1_1 of edge e1
10    double x1_in = 0;
11    double x1_1 = 0.5;
12    // Initial area of edge e1
13    double A_e1 = 1;
14    // Coordinates x2 and x3 of entry nodes(EN)
15    double EN_x2[3] = {2.5, 5, 7.5};
16    double EN_x3[3] = {2.5, 5, 7.5};
17    // Integer for indexing entry and help nodes
18    int count = 0;
19    // Creating vascular bed until step = level + 1
20    for (int k = 0; k < number_x2; k++) {
21        for (int l = 0; l < number_x3; l++) {
22            // Add entry and help nodes to the graph
23            ve1.EntryNode[count] = ve1.addNode(x1_in, EN_x2[k], EN_x3[l],0,0,Dirichlet);
24            ve1.HelpNode[count] = ve1.addNode(x1_1,EN_x2[k],EN_x3[l],0,0,0);
25            step = 1;
26            // Add adge e1
27            ve1.addEdge(ve1.EntryNode[count],ve1.HelpNode[count],A_e1);
28            // Build the graph starting from the help node with level
29            graph1_branch(step,level,ve1.HelpNode[count],A_e1);
30            count++;
31        }
32    }
33    // Step - mirroring
34    // Build matrix of adjacent matrix
35    ve1.buildAdjacent();
36    // Current number of vertices and edges
37    int mNodes = ve1.V.size();
38    int mEdges = ve1.E.size();
39    // Add reflected nodes and edges to the graph
40    for (int i = 0; i < mNodes; i++)
41        ve1.addNode(1.5*reflection_point-ve1.V[i].x1,ve1.V[i].x2,ve1.V[i].x3);
42
43    for (int i = 0; i < mEdges; i++)
44        int ei = ve1.addEdge(ve1.E[i].v1+mNodes,ve1.E[i].v2+mNodes,ve1.E[i].A);
45
46    // Step - connecting
47    for (int i = 0; i < mNodes; i++){
48        // 1 neighbour and it can not be v1_1
49        if (ve1.V[i].x1 > x1_1 / 2.0 && ve1.V[i].adjacentNodes.size() < 2) {
50            int j;
51            for (int k = 0; k < mEdges; k++)
52                if (i == ve1.E[k].v1 || i == ve1.E[k].v2) {
53                    j = k;
54                    break;
55                }

```

```

56     ve1.addEdge(i, i+mNodes, ve1.E[j].A);
57 }
58 }
59 // Set Neumann boundary conditions
60 count = 0;
61 for (int i = 0; i < ve1.V.size(); i++) {
62     // Only for nodes with one neighbour and not the entry node
63     if (ve1.V[i].x1 >= ee.x1_N / 2.0 && ve1.V[i].adjacentNodes.size() < 2) {
64         ve1.V[i].boundary_condition = Neumann;
65         // Index of final node
66         ve1.FinalNode[count] = i;
67         count++;
68     }
69 }
70 // Defect: Modification of edge cross-sections A based on a defect factor d
71 // Center point of the defect area - cube with r radius
72 double x1_D = (ee.x1_N + ee.x1_0) / 2.0;
73 double x2_D = (ee.x2_N + ee.x2_0) / 2.0;
74 double x3_D = (ee.x3_N + ee.x3_0) / 2.0;
75 double r = 2;
76 mNodes = ve1.V.size();
77 mEdges = ve1.E.size();
78 for (int e = 0; e < mEdges; e++) {
79     int v1 = ve1.E[e].v1;
80     int v2 = ve1.E[e].v2;
81     if (fabs((ve1.V[v1].x1 + ve1.V[v2].x1)/2.0-x1_D) < r &&
82         fabs((ve1.V[v1].x2 + ve1.V[v2].x2)/2.0-x2_D) < r &&
83         fabs((ve1.V[v1].x3 + ve1.V[v2].x3)/2.0-x3_D) < r) {
84         ve1.E[e].A *= d;
85     }
86 }
87 // Rebuild adjacency matrix
88 ve1.buildAdjacent();
89 // Solve pressure
90 ve1.solvePressure();
91 }

```

Následující funkce zajišťuje rozvětvení cévního řečiště. Nejdříve si uložíme souřadnice vrcholu $V[\text{parent_index}]$, ve kterém dochází k větvení. Následně pomocí funkce `rand()` určíme počet vrcholů, které se mají vytvořit. Dále dopočítáme průřez budoucích hran a souřadnice nového vrcholu. Zkontrolujeme, jestli se souřadnice nového vrcholu nachází v oblasti mimocévního prostředí. Pokud ano, tak vytvoříme nový vrchol $V[v_i]$ a tento vrchol spojíme s vrcholem $V[\text{parent_index}]$, tím vytvoříme novou hranu. Pro vrchol $V[v_i]$ opakujeme stejný postup jako pro $V[\text{parent_index}]$. Podmínka k zastavení rozvětřování je uvedena na začátku funkce, tj. $\text{step} = \text{level} + 1$.

```

1 void graph1_branch(int step, int level, int parent_index, double A) {
2     // Base case: If the current level exceeds the maximum level, return
3     if (step > level+1) return;
4     // Extract coordinates of the parent node
5     double x1 = ve1.V[parent_index].x1;
6     double x2 = ve1.V[parent_index].x2;
7     double x3 = ve1.V[parent_index].x3;
8     // Generate a random number of child nodes to create
9     int number = rand() % (step + 3) + 2;
10    // Loop to create child nodes
11    for (int i = 0; i < number; i++) {
12        // Calculate the cross-section of new edge ei

```

```

13     double A_ei = A / number;
14     // Generate random coordinates for the new node
15     double x1_new = x1 + 0.5 + 0.5 * epsilon() / (level + 1);
16     double x2_new = x2 + randomPlusMinusOne() * 4 * epsilon() / (level + 2);
17     double x3_new = x3 + randomPlusMinusOne() * 2 * epsilon() / (level + 1);
18     // Adjust x3 coordinate for a specific case
19     if (i == 1) {
20         x3_new = x3 + 1 * epsilon() / (level + 5);
21     }
22     // New node is within the defined bounds of extravascular environment
23     if (x1_new >= ee.x1_0 && x1_new <= ee.x1_N &&
24         x2_new >= ee.x2_0 && x2_new <= ee.x2_N &&
25         x3_new >= ee.x3_0 && x3_new <= ee.x3_N) {
26         // Add the new node to the graph
27         int v_i= ve1.addNode(x1_new, x2_new, x3_new, 0, 0, 0);
28         // Add edge e_i connecting v_i and parent_node
29         ve1.addEdge(parent_index, v_i, A_ei);
30         // Recursively build the graph from the new node
31         graph1_branch(step+1, level, v_i, A_ei);
32     }
33 }
34 }

```

3.2 VacularEnvironment.h

Následující funkce plní konstantní matici z rovnice (2.27). Ve funkci je cyklus přes všechny vrcholy cévního řečiště. V prvním kroku se na diagonálu matice \mathbb{B}_c nastaví hodnota 1. Další krok se provádí pouze pro vrcholy, které mají alespoň dva sousedy (kromě okrajových vrcholů).

Na řádcích 10 – 13 projdeme všechny sousední hrany daného vrcholu a spočítáme objem daného vrcholu.

V dalším kroku opět projdeme sousední hrany daného vrcholu a pro každou hranu si do proměnné `ve1` uložíme rychlost na dané hraně. Přidání advektivního členu bude záviset na znaménku této proměnné. Na diagonálu přidáme příspěvek difuzního členu, obdobně to bude na místo sousedního vrcholu v matici \mathbb{B}_c .

Za podmínky, že `ve1 < 0` se podobně jako difuzní část uloží do matice příspěvek za advektivní část. Podmínka `ve1 < 0` představuje upwindovou stabilizaci 1. řádu.

Následně pro vrchol s Neumannovou okrajovou podmínkou se uloží do matice hodnota `-1` na pozici vrcholu sousedícího s vrcholem s Neumannovou o.p.

```

1 void fillConcentrationMatrix(double diffusion, double dt) {
2     int edge = 0;
3     // Loop through each node
4     for (int r = 0; r < VE.N; r++) {
5         // Set diagonal matrix to 1
6         VE_SparseMatrix(r,r) = 1.0;
7         // If node has more neighbours
8         if (V[r].adjacentEdges.size() > 1) {
9             // Calculate the total volume of adjacent edges
10            for (int s = 0; s < V[r].adjacentEdges.size(); s++) {
11                edge = V[r].adjacentEdges[s];
12                V[r].volume += E[edge].A*E[edge].L/2.0;
13            }
14            // Loop through the neighbors of each node

```

```

15     for (int s = 0; s < V[r].adjacentEdges.size(); s++) {
16         edge = V[r].adjacentEdges[s];
17         // Velocity direction for upwind scheme
18         double vel = velocity(r,V[r].adjacentNodes[s],edge);
19         // Diffusion part
20         VE_SparseMatrix(r,r) += E[edge].A*dt*diffusion/(E[edge].L*V[r].volume);
21         VE_SparseMatrix(r,V[r].adjacentNodes[s]) += -E[edge].A*diffusion*
22                                                     dt/(E[edge].L*V[r].volume);
23         // Advection part
24         if (vel < 0) {
25             VE_SparseMatrix(r,r) += -E[edge].A*vel*dt/V[r].volume;
26             VE_SparseMatrix(r,V[r].adjacentNodes[s]) += E[edge].A*vel*dt/V[r].volume;
27         }
28     }
29 }
30 else {
31     // Neumann boundary condition
32     if (V[r].boundary_condition == 2) {
33         VE_SparseMatrix(r,V[r].adjacentNodes[0]) = -1;
34     }
35 }
36 }
37 }

```

3.3 ExtravacularEnvironment.h

Naplnění matice \mathbb{B}_m z rovnice (2.12) je podobné naplňování matice \mathbb{B}_c . Projdeme všechny body mimocévního prostředí. Uspořádání bodů v mimocévním prostředí je pravidelné, proto naplnění matice je výrazně jednodušší a pro každý bod se uloží difuzní příspěvek ze všech směrů na diagonálu \mathbb{B}_m a na místa sousedních bodů se uloží difuzní příspěvek z daného směru.

```

1 void fill_matrix_M() {
2     // Loop through all grid points
3     for (int k = 0; k < Nx3; k++) {
4         for (int j = 0; j < Nx2; j++) {
5             for (int i = 0; i < Nx1; i++) {
6                 int r = index(i,j,k); // Current grid point index
7                 // Fill the matrix entries with coefficients for the diffusion equation
8                 EE_SparseMatrix(r,index(i,j,k)) = 1+2*dt*diffusion_m/(dx1*dx1)+
9                                                     2*dt*diffusion_m/(dx2*dx2)+
10                                                     2*dt*diffusion_m/(dx3*dx3);
11                 EE_SparseMatrix(r,index(i-1,j,k)) = -dt*diffusion_m/(dx1*dx1);
12                 EE_SparseMatrix(r,index(i+1,j,k)) = -dt*diffusion_m/(dx1*dx1);
13                 EE_SparseMatrix(r,index(i,j-1,k)) = -dt*diffusion_m/(dx2*dx2);
14                 EE_SparseMatrix(r,index(i,j+1,k)) = -dt*diffusion_m/(dx2*dx2);
15                 EE_SparseMatrix(r,index(i,j,k-1)) = -dt*diffusion_m/(dx3*dx3);
16                 EE_SparseMatrix(r,index(i,j,k+1)) = -dt*diffusion_m/(dx3*dx3);
17             }
18         }
19     }
20 }

```

3.4 Interface.h

V této části nejdříve popíšeme výpočet blízkosti bodu cévního řečiště v mimocévním prostředí a naopak. Tento krok provádíme z důvodu urychlení celého programu. V každém časovém kroku potřebujeme spočítat zdrojové členy g_c, g_m dané rovnicí (2.32). Definice vyžaduje počítání integrálů přes celou oblast, ale nenulový příspěvek má pouze několik členů, které jsou dostatečně blízko v závislosti na použité aproximaci Diracovy delta funkce. Tato vzdálenost je po celou dobu stejná. Po nastavení cévního řečiště a mimocévního prostředí si můžeme pro každý bod z cévního řečiště uložit body z mimocévního prostředí, které jsou blízko bodu z cévního řečiště a naopak. Blížkost je určována pomocí nenulového příspěvku Diracovy delta funkce.

Následující kódový blok určuje sousední body cévního řečiště pro bod z mimocévního prostředí. Pro každý bod mimocévního prostředí projdeme celé cévní řečiště. Pokud je Diracova delta funkce po složkách x_1, x_2, x_3 nenulová, pak tento bod cévního řečiště uložíme k danému bodu z mimocévního prostředí.

```
1 void neighbours_in_VE() {
2     // Iterate over the discretized space of the extravascular environment
3     for (int i = 0; i < ee.Nx1; i++) {
4         for (int j = 0; j < ee.Nx2; j++) {
5             for (int k = 0; k < ee.Nx3; k++) {
6                 // Iterate over all vertices in the vascular environment
7                 for (int l = 0; l < ve2.V.size(); l++) {
8                     // Check if the vascular node is close to extravascular node
9                     if ((convolution.diracDelta(ve2.V[l].x1 - ee.x1[i], ee.dx1) > 0) &&
10                        (convolution.diracDelta(ve2.V[l].x2 - ee.x2[j], ee.dx2) > 0) &&
11                        (convolution.diracDelta(ve2.V[l].x3 - ee.x3[k], ee.dx3) > 0)) {
12                         // Save the index of the adjacent vascular node
13                         ee.vertices[ee.index(i, j, k)].adjacent_VE_Nodes.push_back(l);
14                     }
15                 }
16             }
17         }
18     }
19 }
```

Podobně funguje následující část kódu. Pro každý bod z cévního řečiště projdeme mimocévní prostředí a body z mimocévního prostředí, pro které je Diracova delta funkce nenulová ve všech složkách x_1, x_2, x_3 , uložíme k danému bodu z cévního řečiště.

```
1 void neighbours_in_EE() {
2     // Iterate over all vertices in the vascular environment
3     for (int l = 0; l < ve2.V.size(); l++) {
4         // Iterate over the discretized space of the extravascular environment
5         for (int k = 0; k < ee.Nx3; k++) {
6             for (int j = 0; j < ee.Nx2; j++) {
7                 for (int i = 0; i < ee.Nx1; i++) {
8                     // Check if the extravascular node is close to vascular node
9                     if ((convolution.diracDelta(ee.x1[i] - ve2.V[l].x1, ee.dx1) > 0) &&
10                        (convolution.diracDelta(ee.x2[j] - ve2.V[l].x2, ee.dx2) > 0) &&
11                        (convolution.diracDelta(ee.x3[k] - ve2.V[l].x3, ee.dx3) > 0)) {
12                         // Save the index of the adjacent extravascular node
13                         ve2.V[l].adjacent_EE_Nodes.push_back(ee.index(i, j, k));
14                     }
15                 }
16             }
17         }
18     }
19 }
```



```

17     }
18   }
19 }

```

Následující funkce definují výpočet zdrojových členů g_m, g_c . Popíšeme průběh funkce `function_g_m()`, pro funkci `function_g_c()` je popis obdobný.

Procházíme všechny body mimocévního prostředí a pro každý bod napočítáváme zdrojový člen g_m . Zdrojový člen pro daný bod mimocévního prostředí sčítá příspěvky rozdílu koncentrace mimocévního bodu a jeho sousedního bodu v cévním řečišti. Tyto příspěvky jsou vynásobeny přestupovým koeficientem, velikostí objemu bodu cévního řečiště a součinem aproximací Diracovy delta funkce, které má v argumentu rozdíl prostorových souřadnic bodu z mimocévního prostředí a jeho sousedního bodu z cévního řečiště ve směrech eulerovských os.

```

1 void function_g_m(double t) {
2   int X;
3   // Iterate over all vertices in the extravascular environment
4   for ( int x = 0; x < ee.Nx1*ee.Nx2*ee.Nx3; x++ ) {
5     g_m[x] = 0; // Initialize the source function value for the current vertex
6     // Iterate over adjacent vascular nodes for the current extravascular vertex
7     for (int s = 0; s < ee.vertices[x].adjacent_VE_Nodes.size() ; s++ ){
8       X= ee.vertices[x].adjacent_VE_Nodes[s];
9       g_m[x] += k*(ve2.V[X].concentration-ee.vertices[x].vertex_c)*
10        convolution.diracDelta(ee.vertices[x].vertex_x1 - ve2.V[X].x1, ee.dx1)*
11        convolution.diracDelta(ee.vertices[x].vertex_x2 - ve2.V[X].x2, ee.dx2)*
12        convolution.diracDelta(ee.vertices[x].vertex_x3 - ve2.V[X].x3, ee.dx3)*
13        ve2.V[X].volume;
14     }
15   }
16   return;
17 }

```

```

1 void function_g_c(double t) {
2   int x;
3   // Iterate over all vertices in the vascular environment
4   for (int X = 0; X < ve2.V.size(); X++) {
5     g_c[X] = 0; // Initialize the source function value for the current vertex
6     // Iterate over adjacent extravascular nodes for the current vascular vertex
7     for (int ijk = 0; ijk < ve2.V[X].adjacent_EE_Nodes.size(); ijk++) {
8       x = ve2.V[X].adjacent_EE_Nodes[ijk];
9       g_c[X] += k*(ee.vertices[x].vertex_c-ve2.V[X].concentration) *
10        convolution.diracDelta(ee.vertices[x].vertex_x1-ve2.V[X].x1, ee.dx1) *
11        convolution.diracDelta(ee.vertices[x].vertex_x2-ve2.V[X].x2, ee.dx2) *
12        convolution.diracDelta(ee.vertices[x].vertex_x3-ve2.V[X].x2, ee.dx3) *
13        ee.dx1*ee.dx2*ee.dx3;
14     }
15   }
16   return;
17 }

```

Kapitola 4

Výsledky

V této kapitole se budeme zabývat nastavením numerických simulací a výsledky těchto simulací. Máme dvě úlohy, které budou dále rozděleny na podúlohy, kde každá bude zkoumat jinou vlastnost.

V sekci 4.1 popíšeme nastavení první úlohy, zabývající se transportem KL v jednom cévním řečišti s přestupem do mimocévního prostředí. V první části se budeme věnovat numerické konvergenci této úlohy. Budeme zkoumat vliv prostorové diskretizace na celkové množství KL bez přestupu. Dále budeme zkoumat vliv volby koeficientu přestupu na celkové množství KL v systému. Obě dvě části budou zkoumány na jednom konkrétním cévním řečišti. Závěr této sekce bude věnován různým realizacím náhodně generovaného cévního řečiště. Vygenerujeme si $K \in \mathbb{N}$ podobných cévních řečišť a budeme zkoumat vliv různorodosti cévních řečišť na celkové množství KL v systému.

Druhá úloha v sekci 4.2 bude zaměřená na zahuštění cévního řečiště. Oproti první úloze bude do druhé úlohy přidáno celkem 9 rovnoměrně rozprostřených cévních řečišť. Nejdříve budeme zkoumat vliv prostorové diskretizace na celkové množství KL v systému. Dále definujeme nové veličiny pro celkové množství KL v daném systému, původní veličiny M_c , M_m a M zůstanou jen na vnitřní oblast celého systému Ω , volba vnitřní oblasti systému bude popsána při definici těchto nových veličin. V druhé části této sekce zavedeme pojem defekt cévního řečiště. Bude se jednat o nezdravou část cévního řečiště, kterou budeme modelovat formou sníženého průřezu cév. Tím docílíme toho, že KL bude postiženou oblastí protékat méně a budeme zkoumat vliv působení defektu na celkové množství KL v systému.

4.1 Cévní řečiště

V této sekci se nejdříve budeme zabývat transportem KL v cévním řečišti bez přestupu do mimocévního prostředí, tj. $k = 0$. Poté se zaměříme na závislost celkového množství KL na volbě koeficientu přestupu k . Závěr této sekce bude věnován různým volbám cévního řečiště.

Obecně v této sekci bude ve vstupním, resp. výstupním vrcholu předepsán tlak p_A , resp. p_B . Na základě znalosti tlaků v okrajových vrcholech a pomocí postupu popsaného v sekci 1.3.2 vypočítáme rychlost KL v cévním řečišti Ω_c . Poté z vypočítané rychlosti a z hodnot uvedených v zadání úlohy níže vypočteme hodnotu koncentrace KL v daném čase.

Obecné nastavení úlohy je popsáno níže. Při generování cévního řečiště je vstupní průřez A_{in} jednotkový. Průřez obecné hrany v cévním řečišti záleží na způsobu generování řečiště, který je popsán 2.4. Pro tuto úlohu byl vybrán jeden konkrétní typ řečiště nastavením inicializace pseudonáhodného generátoru pomocí funkce `srand` s hodnotou (seed) 1.

Úloha 1: Úloha perfuze myokardu bez defektní tkáň

Uvažujme úlohu (1.35) s následujícím nastavením:

- $T_{max} = 100$,
- $k \in \{0; 0,01; 0,1; 1\}$,
- $L_{x_1} = L_{x_2} = L_{x_3} = 10$,
- $D_m = 0,003$,
- $D_c = 0,0025$,
- $N_{x_1} = N_{x_2} = N_{x_3} = N_{c,e} \equiv N \in \{5, 10, 20, 40\}$, $e \in E(\Gamma)$,
- Náhodně generovaná realizace cévního řečiště: $\text{srand}(1)$,
- $g_m(\mathbf{x}, t) = k(c_c^E(\mathbf{x}, t) - c_m(\mathbf{x}, t)) \quad \forall (\mathbf{x}, t) \in \hat{\Omega}_m \times \hat{I}$,
- $g_c(\mathbf{X}_e(s), t) = k(c_m^L(\mathbf{X}_e(s), t) - c_c(\mathbf{X}_e(s), t)) \quad \forall (\mathbf{X}_e(s), t) \in \hat{\Omega}_c \times \hat{I}$,
- $p_A = 50, p_B = 1$,
- $K_c = 0,5, t_c = 5$,
- $\Delta t = 0,01$,
- Aproximace Diracovy delta funkce: φ_{4b} ,

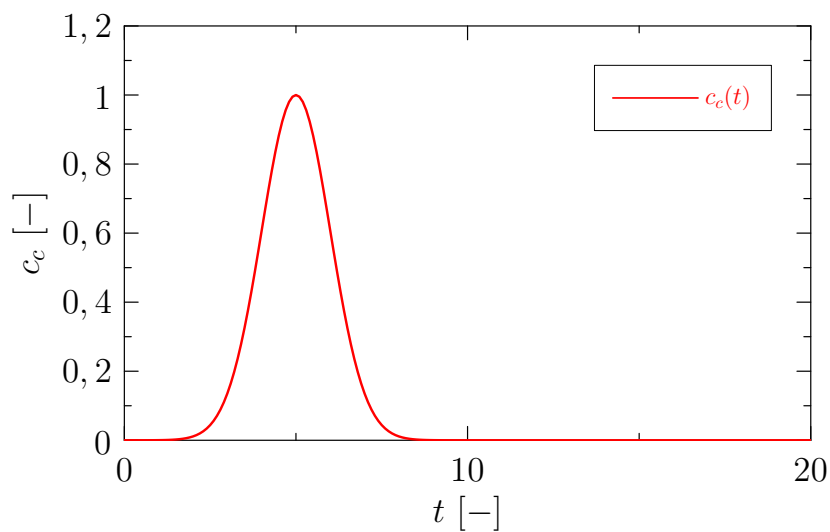
s počáteční a okrajovými podmínkami:

$$\begin{aligned}
 c_c^{ini}(\mathbf{X}_e(s), 0) &= 0 & \forall \mathbf{X}_e \in \hat{\Omega}_c, \\
 c_c(\mathbf{X}_e(s_{in}), t) &= \exp[-K_c(t - t_c)^2] & \forall t \in \hat{I}, \text{ viz(1.36)}, \\
 \frac{\partial c_c}{\partial s}(\mathbf{X}_e(s_{out}), t) &= 0 & \forall t \in \hat{I}, \\
 c_m^{ini}(\mathbf{x}, 0) &= 0 & \forall \mathbf{x} \in \hat{\Omega}_m, \\
 c_m(x_1, x_2, 0, t) &= c_m(x_1, x_2, L_{x_3}, t) & \forall x_1 \in (0, L_{x_1}), \forall x_2 \in (0, L_{x_2}), \\
 c_m(0, x_2, x_3, t) &= c_m(L_{x_1}, x_2, x_3, t) & \forall x_2 \in (0, L_{x_2}), \forall x_3 \in (0, L_{x_3}), \\
 c_m(x_1, 0, x_3, t) &= c_m(x_1, L_{x_2}, x_3, t) & \forall x_1 \in (0, L_{x_1}), \forall x_3 \in (0, L_{x_3}).
 \end{aligned}$$

Poznamenejme, že číslo $N_{c,e}$ určuje počet bodů diskretizace hrany $e \in E(\Gamma)$ nezávisle na délce dané hrany. Dále v mimocévním prostředí bude prostorový krok ve všech směrech stejný, tj. $N_{x_1} = N_{x_2} = N_{x_3}$. Pro jednoduchost grafové interpretace bude platit

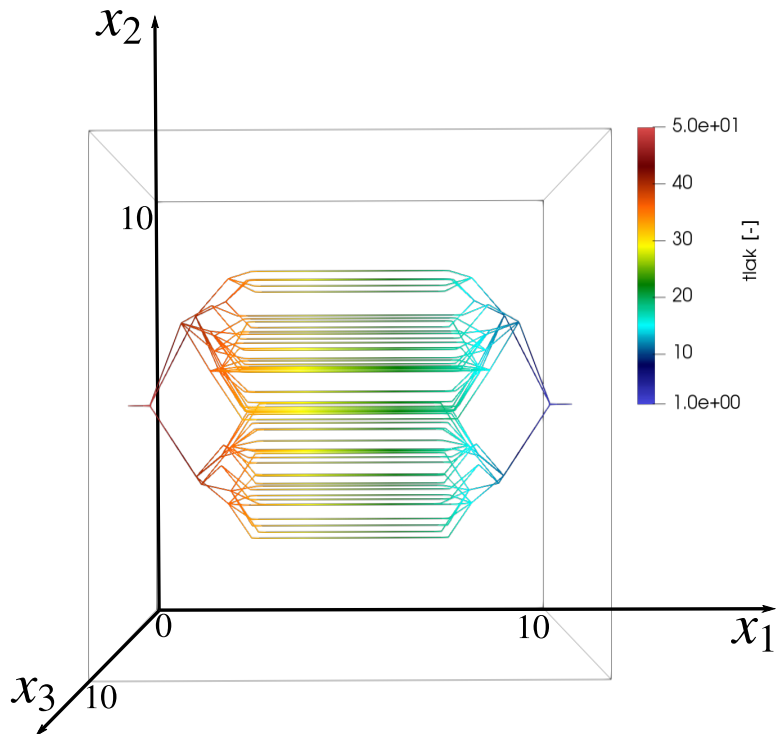
$$N_{x_1} = N_{x_2} = N_{x_3} = N_{c,e} \equiv N. \quad (4.1)$$

Ve vstupním vrcholu je do cévního řečiště vpouštěna KL o koncentraci dané (1.36), časový průběh této koncentrace je znázorněn na obrázku 4.1.



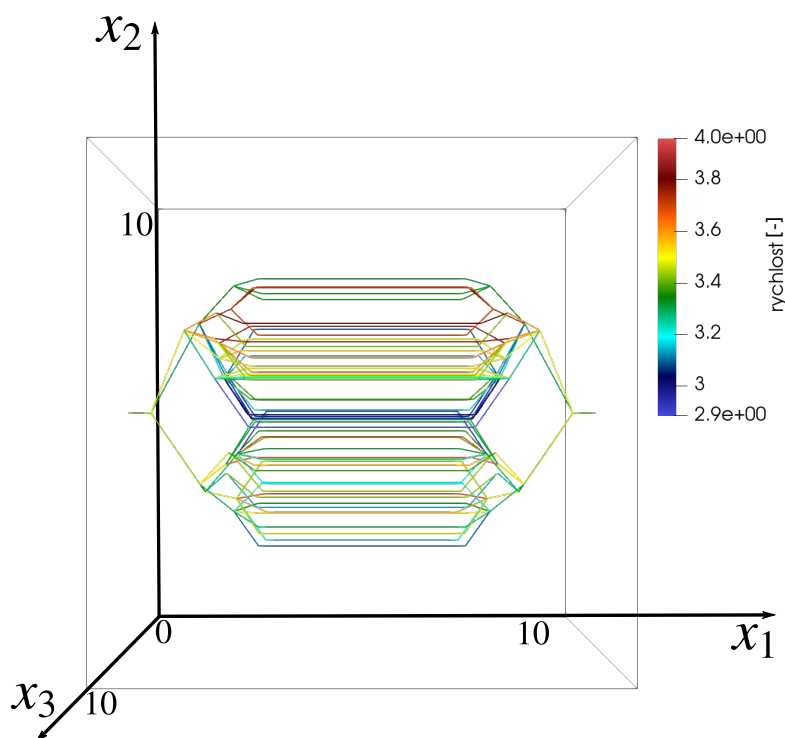
Obrázek 4.1: Časový průběh koncentrace vstříknuté KL.

Na obrázku 4.2 je graficky znázorněn tlak v cévním řečišti Ω_c . Lze pozorovat prostorovou závislost hodnoty tlaku v cévním řečišti Ω_c .



Obrázek 4.2: Tlak v cévním řečišti.

Na obrázku 4.3 je znázorněna vypočtená rychlost v cévním řečišti Ω_c . Rychlost závisí na tlaku na vstupu a na výstupu a na konfiguraci cévního řečiště jako řešení (1.18).



Obrázek 4.3: Graficky znázorněná rychlost v cévním řečišti.

V celém cévním řečišti Ω_c uvažujeme konstantní difuzní koeficient. Rychlost je konstantní pouze na dané úsečce.

Nyní se budeme zabývat tokem KL v čase v závislosti na různém počtu bodů na hraně. Nejdříve si zdefinujeme veličiny pro tok na hranici. Q_{in} je tok KL, která do cévního řečiště přiteče a Q_{out} představuje tok KL, která z cévního řečiště odtéká, tj. $\forall t \in \langle 0, T_{max} \rangle, \forall e \in E(\Gamma)$

$$Q_{in}(t) = - \int_{\partial\Omega_c(in)} c_c(\mathbf{X}_e(s), t) \mathbf{v} \cdot \mathbf{n} \, ds, \quad (4.2)$$

$$Q_{out}(t) = - \int_{\partial\Omega_c(out)} c_c(\mathbf{X}_e(s), t) \mathbf{v} \cdot \mathbf{n} \, ds, \quad (4.3)$$

kde $\partial\Omega_c(in)$, resp. $\partial\Omega_c(out)$ je hranice cévního řečiště Ω_c s hodnotou tlaku p_A , resp. p_B . Vektor \mathbf{n} v rovnici (4.2) je jednotkový vektor vnější normály hranice $\partial\Omega_c(in) \cup \partial\Omega_c(out)$ v bodě $\mathbf{X}_e(s)$.

Dále definujeme kumulativní množství KL M_{in}, M_{out}, M_{bil} vztahy $\forall t \in \langle 0, T_{max} \rangle$

$$M_{in}(t) = \int_0^t Q_{in}(\tau) \, d\tau, \quad (4.4)$$

$$M_{out}(t) = \int_0^t Q_{out}(\tau) \, d\tau, \quad (4.5)$$

$$M_{bil}(t) = M_{in}(t) - M_{out}(t), \quad (4.6)$$

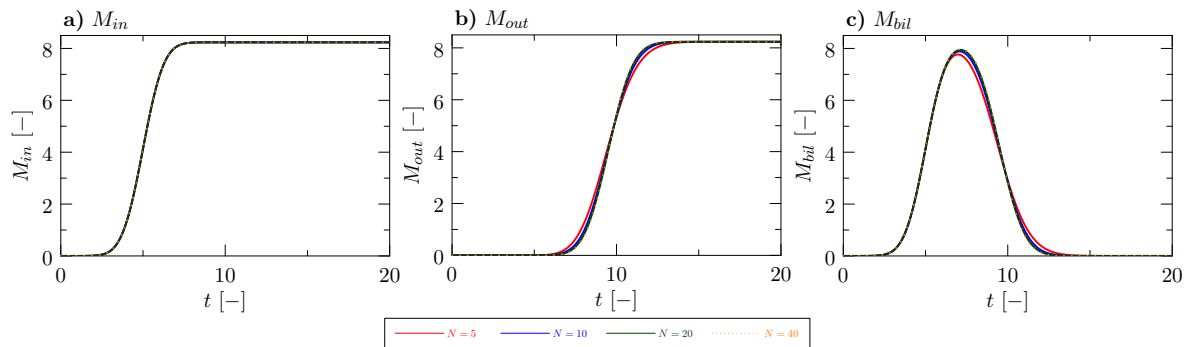
kde M_{in} , resp. M_{out} je celkové množství KL, která přitekla hranicí $\partial\Omega_c(in)$, resp. odtekla hranicí $\partial\Omega_c(out)$ od začátku výpočtu do času t . Veličina $M_{bil}(t)$ představuje množství KL dané rozdílem množství $M_{in}(t)$ a $M_{out}(t)$ v čase t . Definujeme celkové množství KL v systému Ω vztahem

$$M(t) = M_c(t) + M_m(t) \quad \forall t \in \langle 0, T_{max} \rangle. \quad (4.7)$$

4.1.1 Vliv prostorové diskretizace

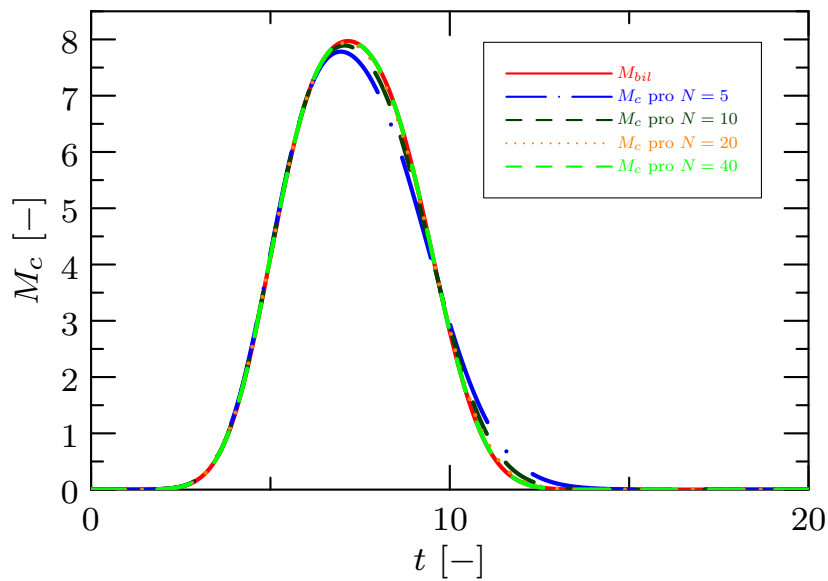
V této části budeme zkoumat vliv prostorové diskretizace na celková množství KL M_{in} , M_{out} , M_{bil} a M .

Na obrázku 4.4 a) pozorujeme, že pro celkové množství KL M_{in} nemá volba diskretizace žádný vliv, což bylo i předpokládáno, jelikož v bodech hranice $\partial\Omega_c(in)$ předepisujeme Dirichletovu okrajovou podmínku. Z pohledu numerické konvergence jsou zajímavější obrázky 4.4 b), c). Na těchto grafech již pozorujeme vliv diskretizace na celkové množství KL. Pro volbu $N = 20$ a $N = 40$ již dostáváme téměř totožný průběh celkového množství KL.



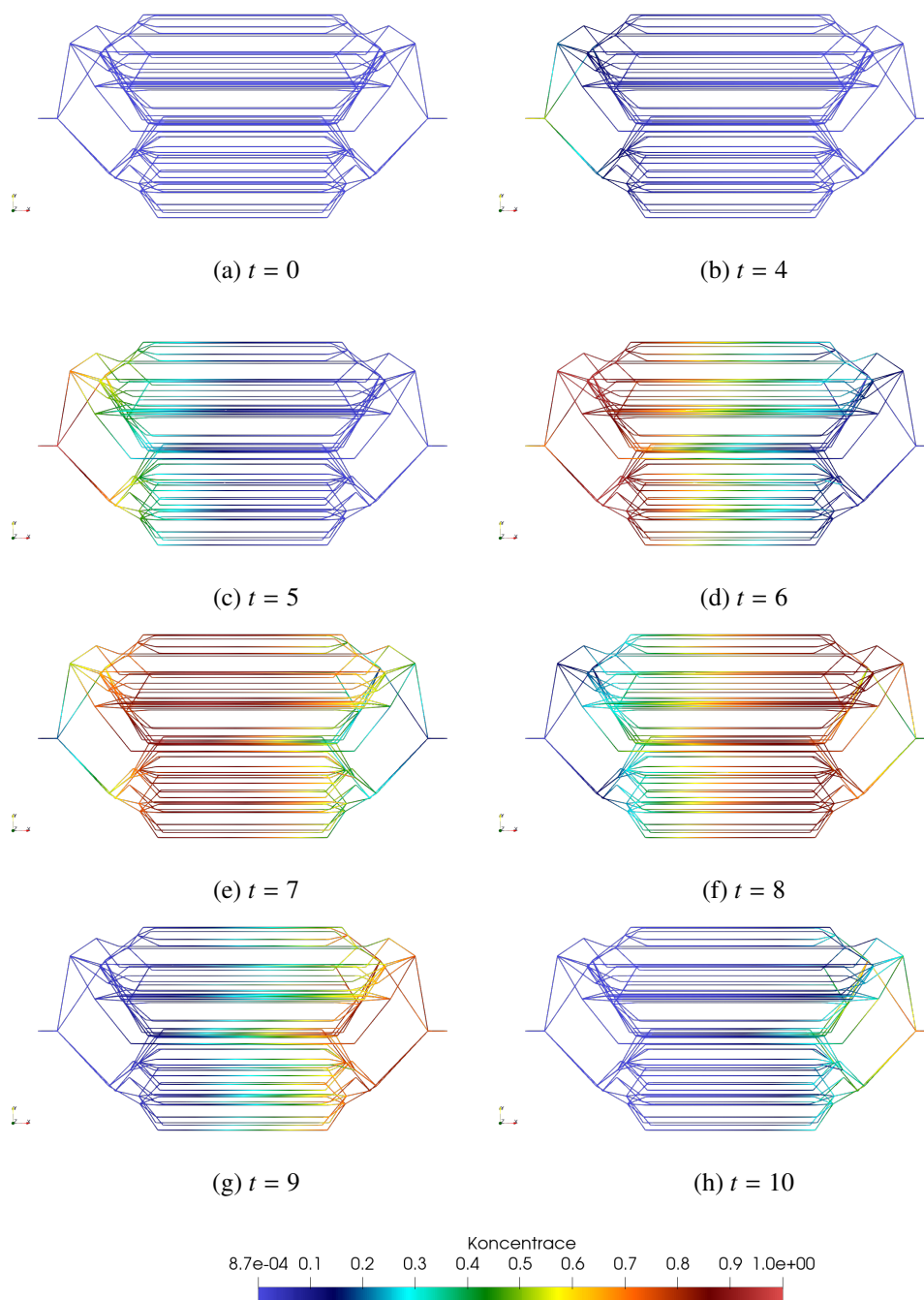
Obrázek 4.4: Vliv volby prostorového kroku N na časový průběh celkového množství KL M_{in} , M_{out} , M_{bil} .

Obrázek 4.5 porovnává celkové množství KL M_c v závislosti na volbě N s celkovým množstvím KL M_{bil} . Opět pozorujeme situaci, kdy pro větší počet N (viz rovnice (4.1)) grafy M_c a M_{bil} jsou téměř totožné. Proto v dalších sekcích této práce budeme volit $N = 40$ a považovat tuto volbu za dostatečně přesnou prostorovou aproximaci.



Obrázek 4.5: Vliv volby prostorového kroku N na časový průběh celkového množství KL M_c .

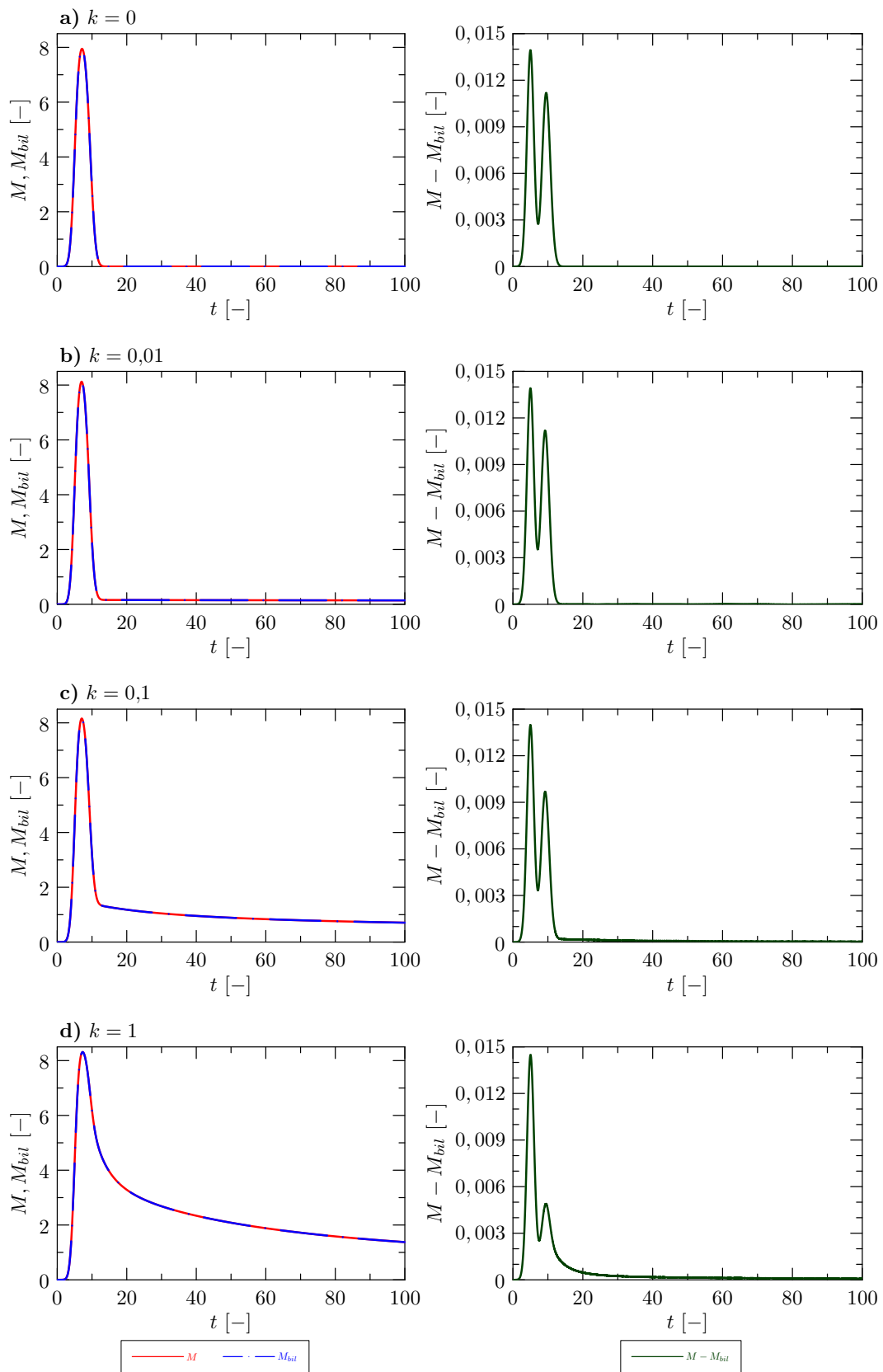
Na obrázku 4.6 je znázorněn časový vývoj koncentrace v cévním řečišti Ω_c . Koncentrace KL se dostává do cévního řečiště s nejvyšší hodnotou v čase $t = 5$. Poté se cévním řečištěm transportuje a následně z něj odtéká pryč.



Obrázek 4.6: Časový průběh koncentrace v cévním řečišti Ω_c pro $k = 0$ a $N = 40$.

4.1.2 Vliv koeficientu přestupu na celkové množství KL v systému

V této části budeme zkoumat vliv volby koeficientu přestupu k na celkové množství KL v systému Ω . Cílem této sekce bude ukázat, že numerické schéma (včetně přestupu mezi Ω_c a Ω_m) je konzervativní, tj. celkové množství KL M v systému bude korespondovat s celkovým množstvím KL M_{bil} .



Obrázek 4.7: Porovnání celkového množství KL v systému M s celkovým množstvím KL M_{bit} pro různé koeficienty přestupu. Pravé obrázky ukazují chybu způsobenou numerickou diskretizací.

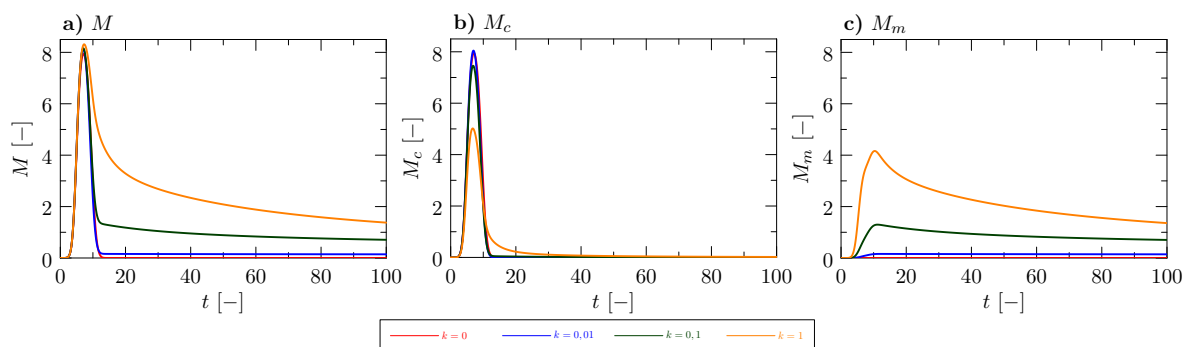
Obrázek 4.7 ukazuje, že při různé volbě koeficientu přestupu se celková množství KL M a M_{bil} téměř neliší. Dále pro různé koeficienty přestupu pozorujeme jiný profil grafu pro M . Pro koeficient přestupu $k = 0$ KL přiteče do cévního řečiště a následně odtéká. Na druhou stranu pro $k = 1$ KL přiteče do cévního řečiště, a poté část této KL rychleji přestupuje do mimocévního prostředí, kde difunduje. Když koncentrace KL v cévách klesne pod hodnotu koncentrace v mimocévním prostředí, tak KL přestupuje z mimocévního prostředí zpět do cévního řečiště, odkud odtéká z celého systému.

Na pravé straně obrázku 4.7 je znázorněn rozdíl M a M_{bil} . Pozorujeme, že k mírným rozdílům dochází v čase, kdy KL přitéká do oblasti, resp. odtéká z oblasti.

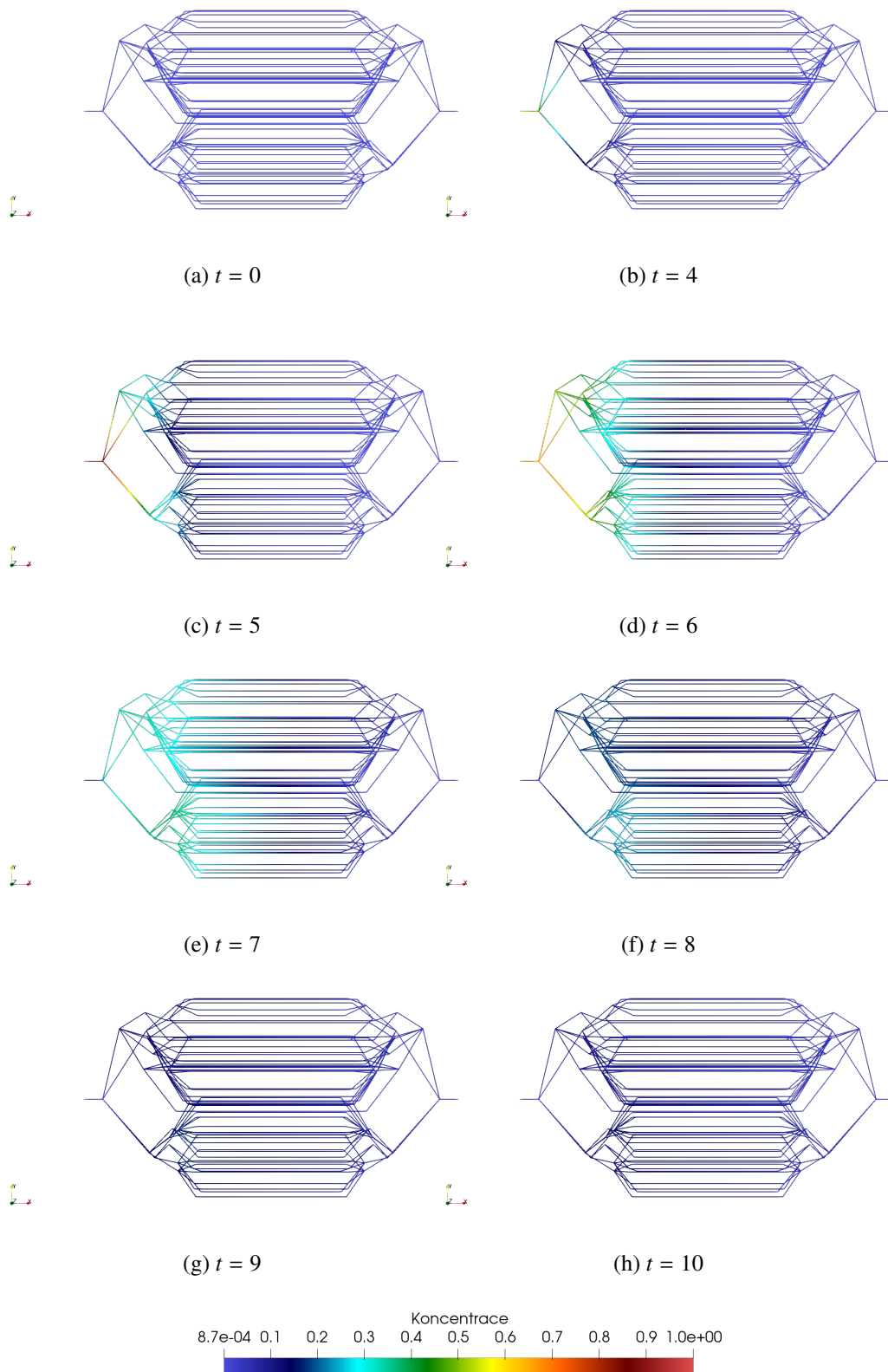
Na obrázku 4.8 je vyobrazen vliv koeficientu přestupu na jednotlivá celková množství M , M_c , M_m . Na obrázku 4.8 b) pozorujeme, že pro větší volbu koeficientu přestupu klesá maximální hodnota M_c , což odpovídá situaci pro M_m (obrázek 4.8 c)), kde se profil naopak zvětšuje - větší množství KL se rychleji dostává z cév do mimocévního prostředí.

Nyní se více zaměříme na obrázek 4.8 a). Můžeme si všimnout, že pro koeficient přestupu $k = 1$ je hodnota celkového množství KL M okolo času $t = 10$ vyšší než pro jiné koeficienty přestupu. Pro nižší hodnoty koeficientů přestupu menší množství KL přestoupí do mimocévního prostředí a KL, která zůstala v cévním řečišti, odtéká z celého systému. Naopak pro $k = 1$ přestoupí více KL do mimocévního prostředí a tím pádem i menší množství KL odeče z celého systému v čase t než pro jinou volbu k .

Pro koeficient přestupu $k = 0$ můžeme na obrázku 4.6 pozorovat časový průběh koncentrace v cévním řečišti. Vidíme, že již v čase $t = 5$ dochází k odtoku KL ze systému. Na obrázku 4.9 je znázorněn obdobný časový průběh koncentrace v cévním řečišti jako na obrázku 4.6 s koeficientem přestupu $k = 1$. Můžeme pozorovat, že KL odtéká cévním řečištěm později a navíc jsou hodnoty koncentrace KL v cévách výrazně menší než pro $k = 0$. Celkové množství KL M v systému bude v této fázi výpočtu větší.



Obrázek 4.8: Vliv koeficientu přestupu na celková množství KL M , M_c , M_m .



Obrázek 4.9: Časový průběh koncentrace v cévním řečišti Ω_c pro $k = 1$.

4.1.3 Statistická analýza vlivu náhodnosti generování řečiště

V předchozích sekcích jsme zkoumali vlastnosti dané úlohy pro jeden typ cévního řečiště s volbou $\text{srand}(1)$. V této části budeme zkoumat vliv typu cévního řečiště na celková množství KL M, M_c, M_m . Náhodně generovanou realizaci cévního řečiště bude určovat funkce $\text{srand}(\omega)$ pomocí jejího argumentu ω .

Pro $\omega \in \{1, 2, \dots, K\}$ vygenerujeme K cévních řečišť $\Omega_c(\omega)$. V každém cévním řečišti napočítáme celková množství KL $M(\omega), M_c(\omega), M_m(\omega)$. Tento postup provedeme pro všechny koeficienty přestupu ze zadání úlohy.

Zavedeme veličinu střední hodnota celkového množství KL M, M_c, M_m vztahy [22]

$$EM(t) = \frac{1}{K} \sum_{\omega=1}^K M(\omega), \quad (4.8)$$

$$EM_c(t) = \frac{1}{K} \sum_{\omega=1}^K M_c(\omega), \quad (4.9)$$

$$EM_m(t) = \frac{1}{K} \sum_{\omega=1}^K M_m(\omega). \quad (4.10)$$

Dále definujeme statistickou veličinu rozptyl M, M_c, M_m vztahy [22]

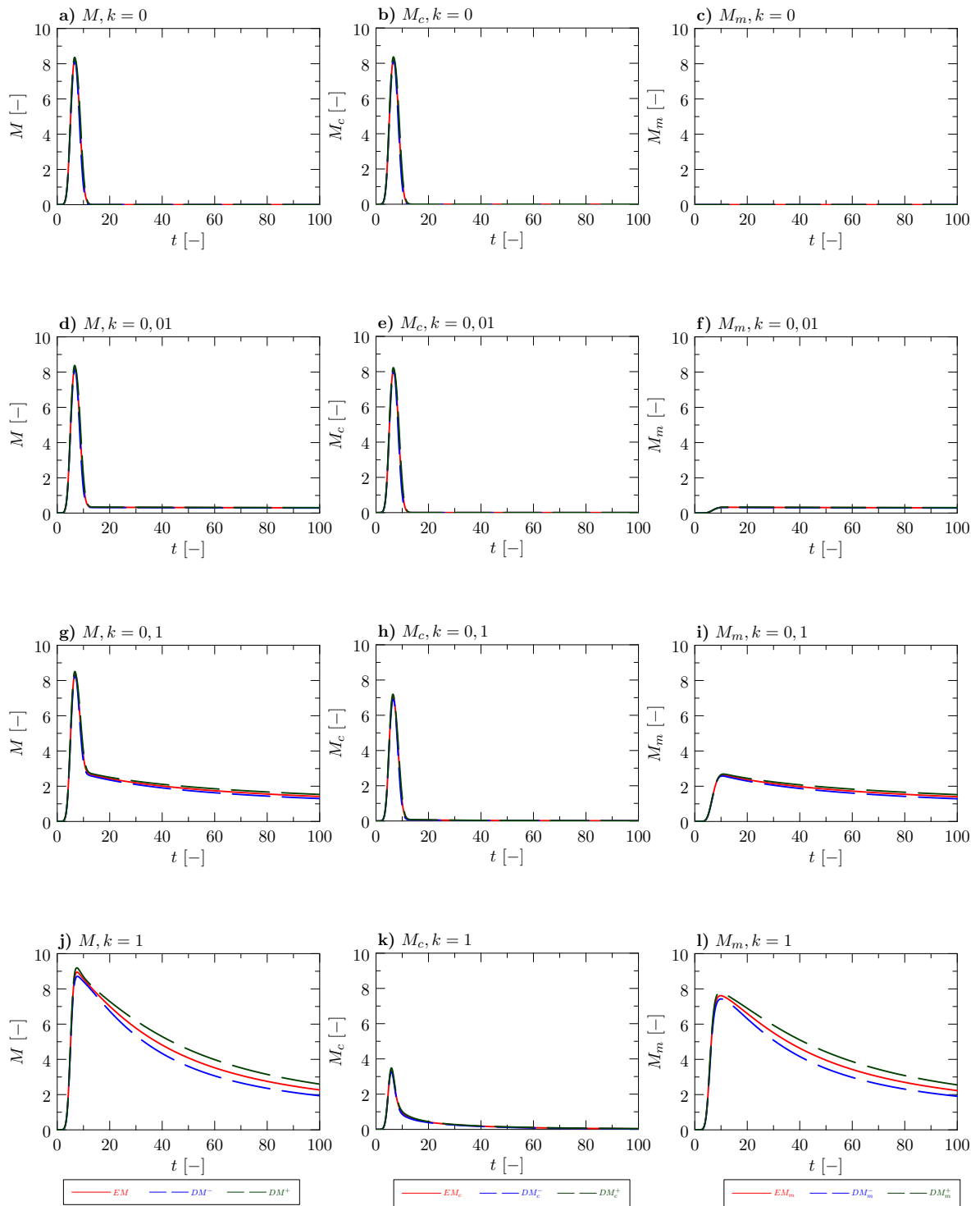
$$DM(t) = \frac{1}{K} \sum_{\omega=1}^K (M(\omega) - EM(t))^2, \quad (4.11)$$

$$DM_c(t) = \frac{1}{K} \sum_{\omega=1}^K (M_c(\omega) - EM_c(t))^2, \quad (4.12)$$

$$DM_m(t) = \frac{1}{K} \sum_{\omega=1}^K (M_m(\omega) - EM_m(t))^2. \quad (4.13)$$

Jednotlivé sloupce obrázku 4.10 představují celková množství pro M, M_c a M_m . Každý řádek obrázku 4.10 je věnován jinému koeficientu přestupu. V každém grafu jsou 3 křivky: $EM_\beta, DM_\beta^+, DM_\beta^-$, kde $\beta \in \{-, c, m\}$ rozlišuje celková množství KL M, M_c, M_m . Graf DM_β^+ , resp. DM_β^- představuje horní, resp. dolní rozptyl od střední hodnoty EM_β .

Na obrázku 4.10 lze vidět, že čím větší koeficient přestupu, tím větší rozptyl celkového množství KL. To je způsobeno různým rozmístěním jednotlivých úseček v různých cévních řečištích. Závěrem můžeme říct, že typ cévního řečiště (až na malé odchylky) nemá velký vliv na celkové množství KL, a tím pádem budeme nadále v této práci používat jeden typ cévního řečiště.

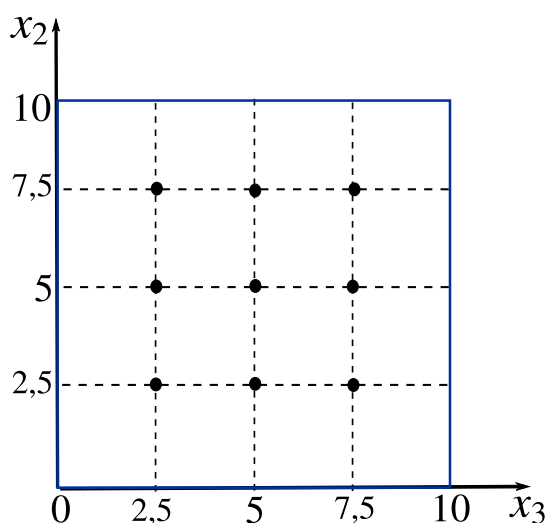


Obrázek 4.10: Střední hodnota a rozptyl celkového množství látky v systému, v cévním řečišti a v mimocévním prostředí pro $K = 100$.

4.2 Zahuštění oblasti - více cévních řečišť

V předchozí sekci jsme se zabývali transportem KL v jednom cévním řečišti s přestupem do mimocévního prostředí. Cílem této sekce bude více zaplnit mimocévní prostředí (myokard) jednorozměrnými úsečkami (cévy). Snažíme se více přiblížit anatomii srdce, kde je celý myokard velmi hustě protkán cévami různých průměrů.

Od jednoho cévního řečiště přejdeme k 9 cévním řečištím postupem popsaným 2.6. Generování 9 cévních řečišť bude prováděno tak, že vytvoříme 9 vstupních okrajových vrcholů, pro které proběhne algoritmus generování řečiště 2.4. Na obrázku 4.11 lze vidět rozmístění vstupních okrajových vrcholů.



Obrázek 4.11: Ilustrace rozmístění okrajových vstupních vrcholů jednotlivých cévních řečišť pro $x_1 = 0$.

Cévní řečiště bude nyní reprezentováno 9 grafy a mimocévní prostředí bude stále třírozměrná oblast. Podrobnější nastavení úlohy je definováno v Úloze 2. Dirichletova okrajová podmínka 1.37b je předepsána ve všech 9 okrajových vstupních vrcholech a to bude mít vliv na celkové množství KL, které bude oproti Úloze 1 9-krát větší.

Úloha bude rozdělena na dvě části. V první části se budeme věnovat celkovému množství KL definovaném na zúžené oblasti mimocévního prostředí. Zúženou oblast mimocévního prostředí budeme nazývat zájmová oblast (ozn. ROI z anglického překladu region of interest). Tuto oblast budeme značit symbolem Ω^{ROI} . Podrobněji bude popsáno v následující podsekci.

V druhé části této úlohy definujeme různé druhy defektů a následně budeme zkoumat vliv daného defektu a koeficientu přestupu na celkové množství KL.

Úloha 2: Úloha perfuze myokardu s defektní tkání

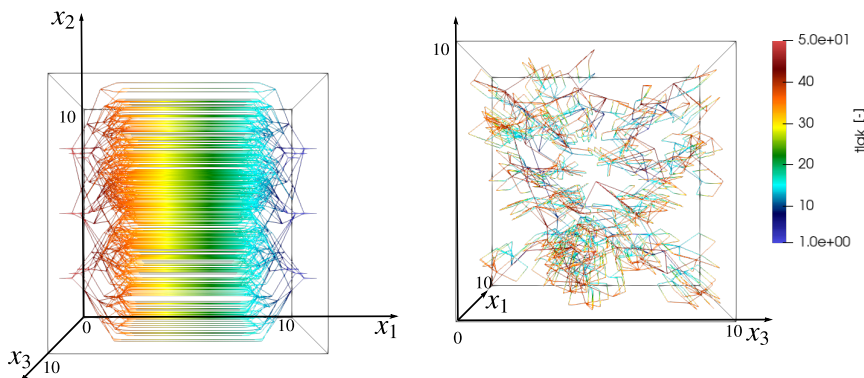
Uvažujme úlohu (1.35) s následujícím nastavením:

- $T_{max} = 100$,
- $k \in \{0; 0,01; 0,1; 1\}$,
- $L_{x_1} = L_{x_2} = L_{x_3} = 10$,
- $D_m = 0,003$,
- $D_c = 0,0025$,
- $N_{x_1} = N_{x_2} = N_{x_3} = N_{c,e} \equiv N = 40$,
- Náhodně generovaná realizace cévního řečiště: `srand(1)`,
- $g_m(\mathbf{x}, t) = k \left(c_c^E(\mathbf{x}, t) - c_m(\mathbf{x}, t) \right) \quad \forall (\mathbf{x}, t) \in \hat{\Omega}_m \times \hat{I}$,
- $g_c(\mathbf{X}_e(s), t) = k \left(c_m^L(\mathbf{X}_e(s), t) - c_c(\mathbf{X}_e(s), t) \right) \quad \forall (\mathbf{X}_e(s), t) \in \hat{\Omega}_c \times \hat{I}$,
- $p_A = 50, p_B = 1$,
- $K_c = 0,5, t_c = 5$,
- $\Delta t = 0,01$,
- Aproximace Diracovy delta funkce: φ_{4b} ,
- Počet cévních řečišť: 9,

s počáteční a okrajovými podmínkami:

$$\begin{aligned}
 c_c^{ini}(\mathbf{X}_e(s), 0) &= 0 & \forall \mathbf{X}_e \in \hat{\Omega}_c, \\
 c_c(\mathbf{X}_e(s_{in}), t) &= \exp[-K_c(t - t_c)^2] & \forall t \in \hat{I}, \text{ viz (1.36)}, \\
 \frac{\partial c_c}{\partial s}(\mathbf{X}_e(s_{out}), t) &= 0 & \forall t \in \hat{I}, \\
 c_m^{ini}(\mathbf{x}, 0) &= 0 & \forall \mathbf{x} \in \hat{\Omega}_m, \\
 c_m(x_1, x_2, 0, t) &= c_m(x_1, x_2, L_{x_3}, t) & \forall x_1 \in (0, L_{x_1}), \forall x_2 \in (0, L_{x_2}), \\
 c_m(0, x_2, x_3, t) &= c_m(L_{x_1}, x_2, x_3, t) & \forall x_2 \in (0, L_{x_2}), \forall x_3 \in (0, L_{x_3}), \\
 c_m(x_1, 0, x_3, t) &= c_m(x_1, L_{x_2}, x_3, t) & \forall x_1 \in (0, L_{x_1}), \forall x_3 \in (0, L_{x_3}).
 \end{aligned}$$

Na obrázku 4.12 je graficky znázorněno nastavení úlohy 2.



Obrázek 4.12: Ilustrace rozmístění 9 cévních řečišť ze dvou pohledů.

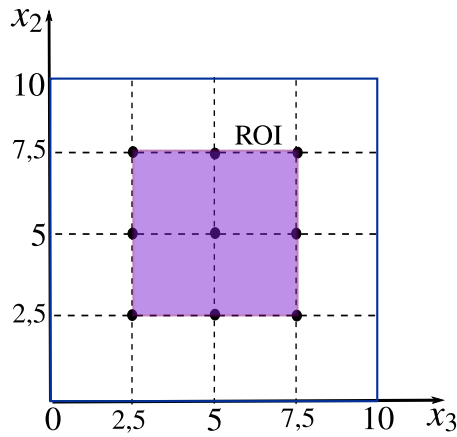
4.2.1 Zájmová oblast

Tato část bude zaměřena na porovnání průběhu celkového množství KL v celém systému Ω a v zájmové oblasti (značíme Ω^{ROI}). Ilustrace oblasti Ω^{ROI} je na obrázku 4.13 znázorněna fialovou barvou pro pevné $x_1 \in \langle 0, 10 \rangle$.

Dále definujeme oblast Ω_c^{ROI} , resp. Ω_m^{ROI} následujícími vztahy

$$\Omega_c^{ROI} = \Omega_c \cap \Omega^{ROI}, \quad (4.14)$$

$$\Omega_m^{ROI} = \Omega_m \cap \Omega^{ROI}. \quad (4.15)$$



Obrázek 4.13: Ilustrace výpočetní oblasti Ω^{ROI} pro pevné $x_1 \in \langle 0, 10 \rangle$.

V úloze 1 jsme pozorovali časový průběh celkového množství KL M . Při větším koeficientu přestupu většina KL přestoupila do mimocévního prostředí, kde difundovala i do okrajů mimocévního prostředí. Jelikož jsme nebyli schopni v úloze 1 zaplnit celé mimocévní prostředí cévami, tak kontrastní látka, nacházející se na okrajích mimocévního prostředí, buď difundovala do blízkosti některé z cév nebo zůstala v mimocévním prostředí k finálnímu času T_{max} .

Zahuštěním mimocévního prostředí 9 řečišti se snažíme této situaci zamezit. I tak je z obrázku 4.13 je patrné, že jsou místa v mimocévním prostředí, která nejsou v dostatečné blízkosti některé z cév. Proto jsme definovali oblast Ω^{ROI} , která taková místa neobsahuje.

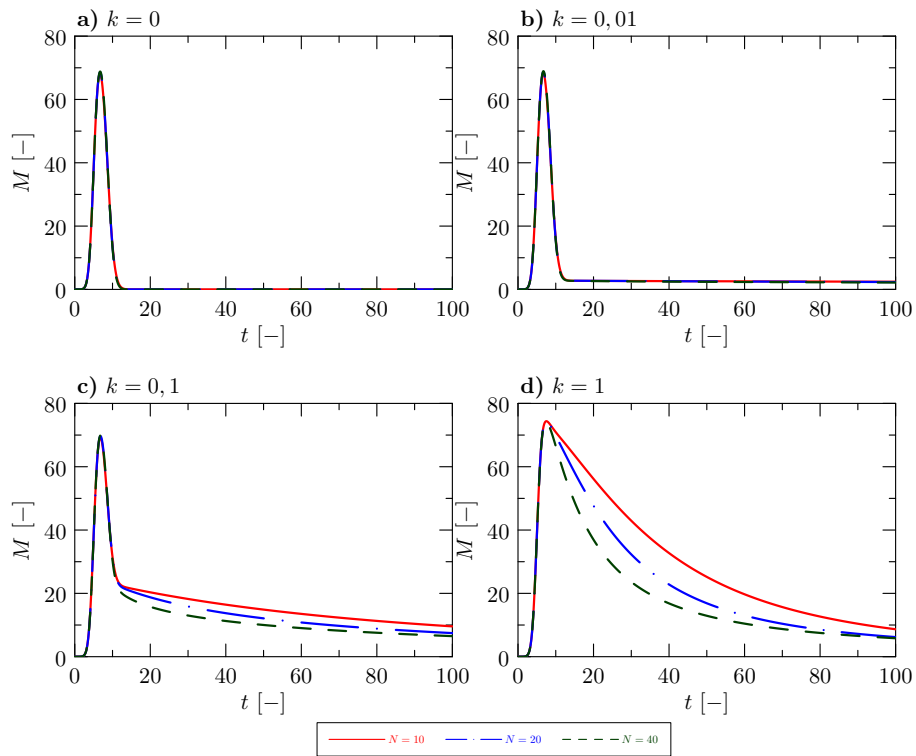
Definujeme celková množství KL M, M_c, M_m v oblasti Ω^{ROI} následovně $\forall t \in \langle 0, T_{max} \rangle$

$$M_c^{ROI}(t) = \int_{\Omega_c^{ROI}} c_c(\mathbf{X}_e(s), t) A(\mathbf{X}_e(s)) ds \quad \forall e \in E(\Gamma), \quad (4.16)$$

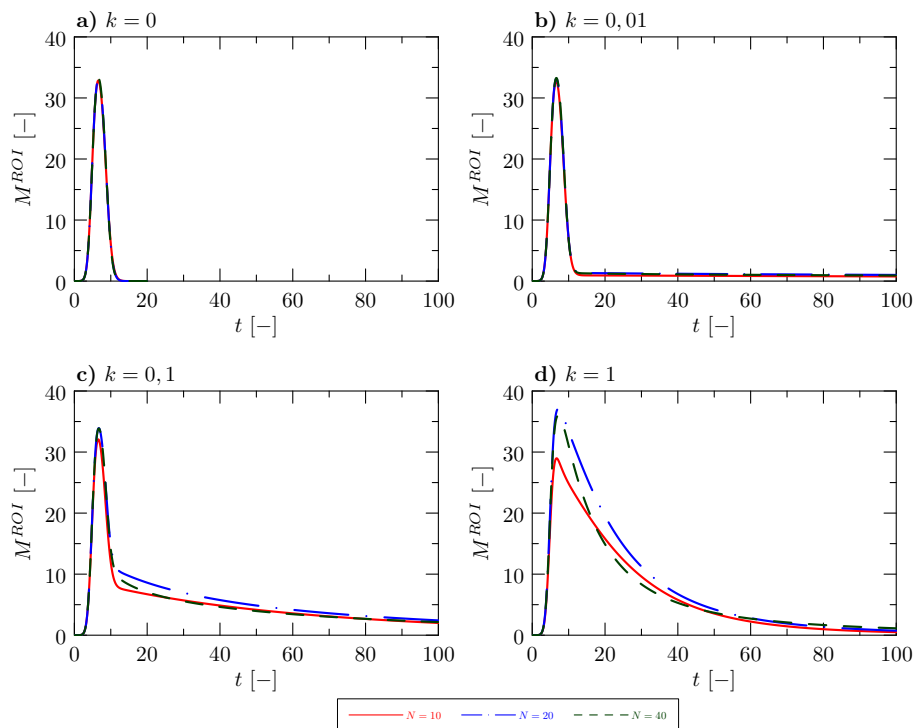
$$M_m^{ROI}(t) = \int_{\Omega_m^{ROI}} c_m(\mathbf{x}, t) d\mathbf{x}, \quad (4.17)$$

$$M^{ROI}(t) = M_c^{ROI}(t) + M_m^{ROI}(t). \quad (4.18)$$

Na obrázcích 4.14, resp. 4.15 je vyobrazeno celkové množství KL v systému Ω , resp. v Ω^{ROI} . Pozorujeme, že pro větší volbu koeficientu přestupu k v obou případech má volba diskretizace vliv na celkové množství KL.

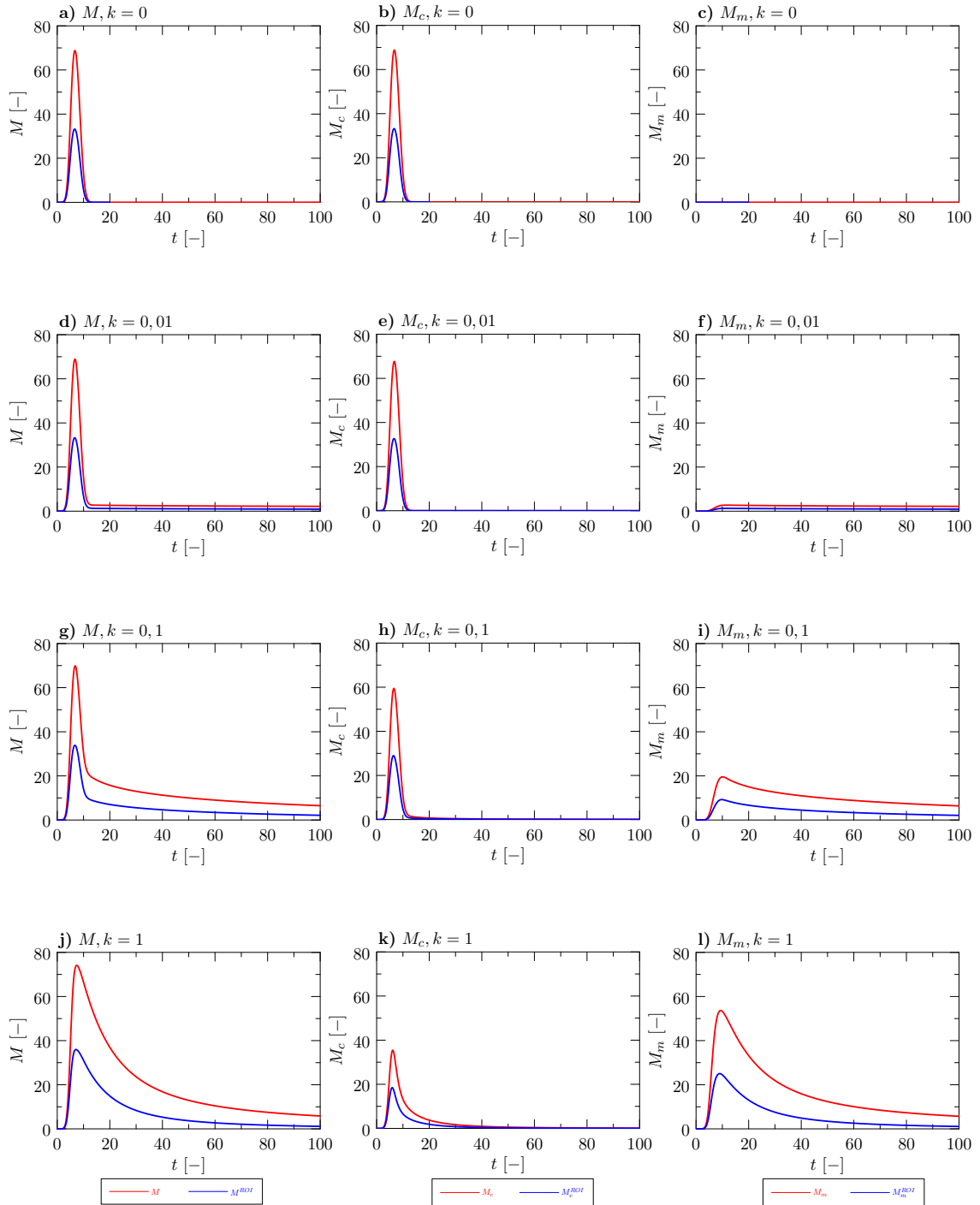


Obrázek 4.14: Vliv prostorové diskretizace na celkové množství KL M pro různé koeficienty přestupu.



Obrázek 4.15: Vliv prostorové diskretizace na celkové množství KL M^{ROI} pro různé koeficienty přestupu.

Na obrázku 4.16 lze vidět porovnání celkových množství KL definovaných v celém systému Ω a v systému Ω^{ROI} . V této úloze nás zajímalo, zda bude zůstat KL v mimocévním prostředí nebo rychleji opustí daný systém. Na obrázku 4.16 i) a l) vidíme, že KL stále zůstává v systému Ω^{ROI} podobně jako v systému Ω . V dalších částech této práce budeme pracovat pouze s veličinami definovanými v Ω .



Obrázek 4.16: Porovnání celkového množství KL M a M^{ROI} , M_c a M_c^{ROI} , M_m a M_m^{ROI} .

4.2.2 Nezdravý myokard

Doposud jsme zkoumali situace pro zdravé srdce, tj. všechny úsečky reprezentující v praxi cévy měly daný průřez na základě úrovně dané úsečky v cévním řečišti. Transport KL cévním řečištěm probíhal rovnoměrně stejně v závislosti na délce a průřezu dané úsečky.

Nyní budeme simulovat nezdravé srdce. Zvolíme si oblast myokardu (tzv. defektní oblast), kde uměle zmenšíme průřez úseček. Tím docílíme toho, že KL bude defektní oblastí protékat pomaleji nebo skoro vůbec. V této práci bude defektní oblast ve tvaru krychle o délce hrany 4 [-] umístěna do středu mimocévního prostředí viz obrázek 4.17.

Defektní oblast budeme značit Ω_d , kde dolní index d bude určovat zmenšení daného průřezu úsečky v_e reprezentované hranou e následovně

$$A_{d,e} = dA_e \quad \forall v_e \in \Omega_d, \quad (4.19)$$

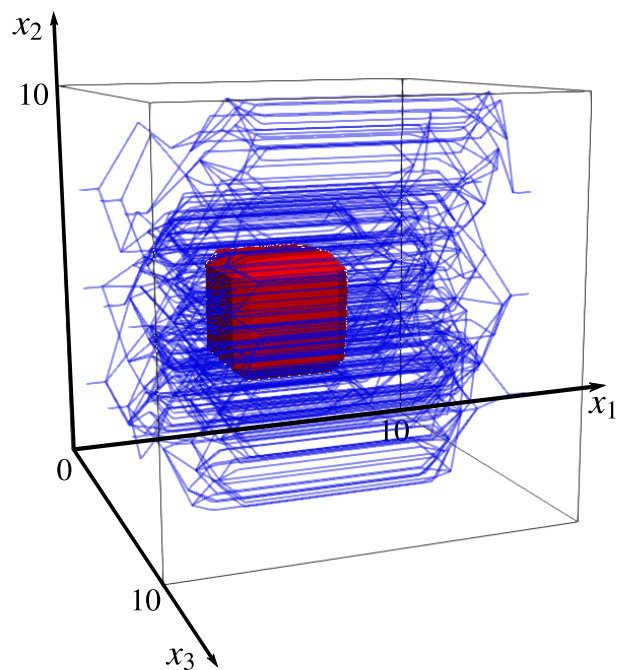
kde A_e je průřez hrany e ve zdravém srdci a $A_{d,e}$ je průřez hrany e v nezdravém srdci.

V této části se budeme zabývat 5 typy defektů:

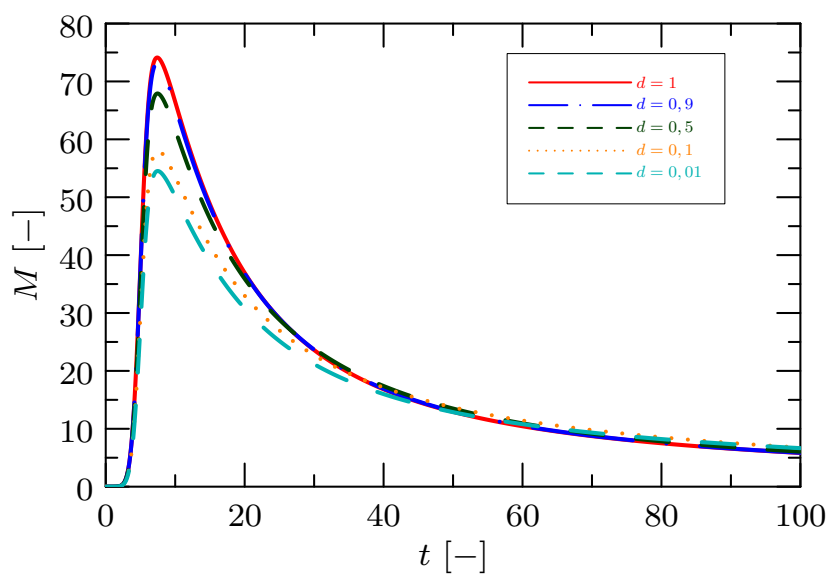
1. $d = 1$ (zdravé srdce),
2. $d = 0,9$,
3. $d = 0,5$,
4. $d = 0,1$,
5. $d = 0,01$.

Vliv volby d a koeficientu přestupu k na celková množství KL M , M_c , M_m je znázorněn na obrázku 4.19. Pozorujeme situaci, kdy při menší volbě d KL přestupuje do mimocévního prostředí s větším zpožděním. Podobným výzkumem se zabývala diplomová práce [13].

Obrázek 4.18 znázorňuje vliv typu defektní oblasti charakterizovanou hodnotou d na celkové množství KL M pro volbu $k = 1$.



Obrázek 4.17: Ilustrace defektní oblasti Ω_d , která je umístěna ve středu myokardu.

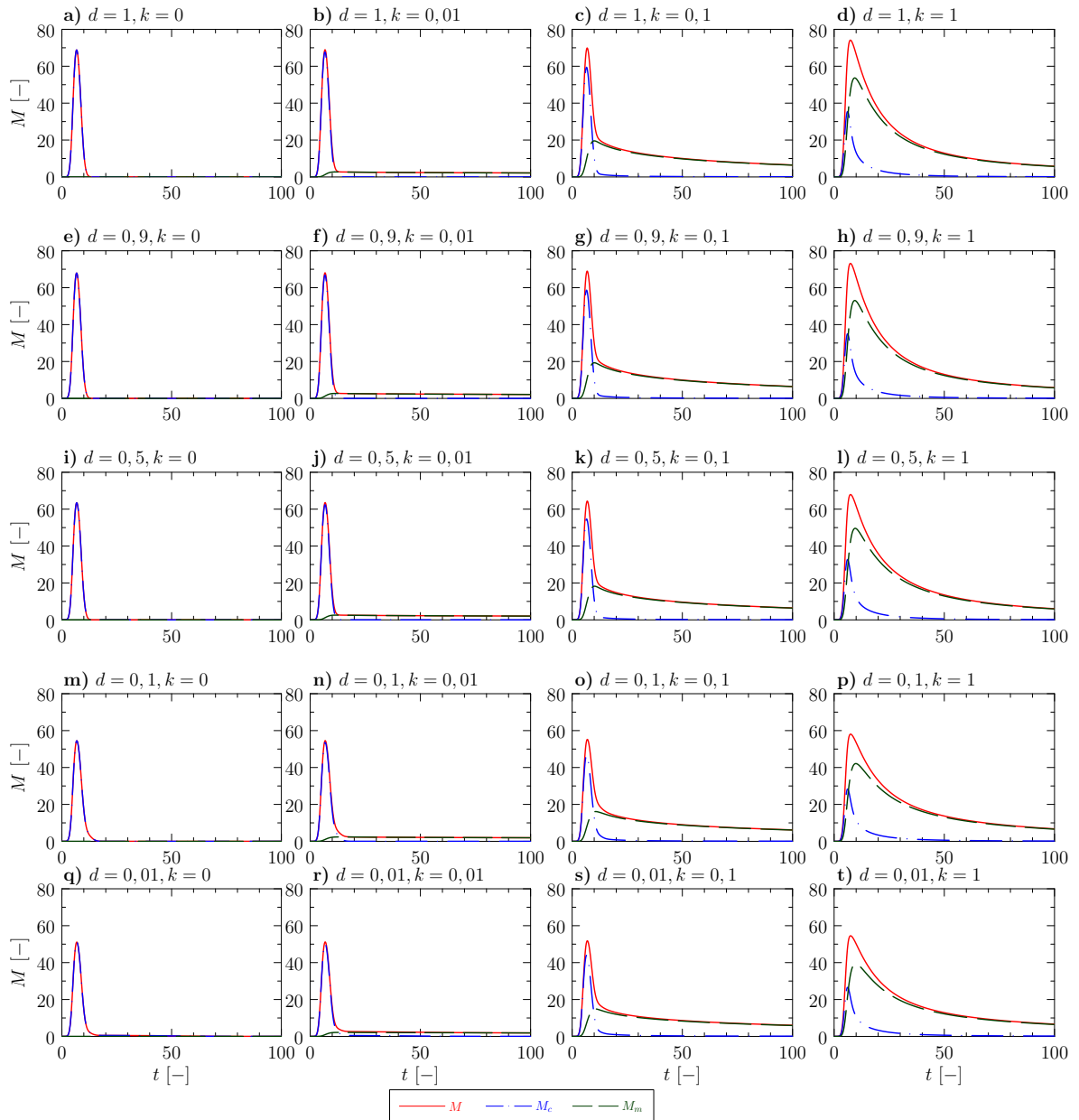


Obrázek 4.18: Průběh celkového množství KL M pro $k = 1$ a různé volby defektu d .

V části Odkazy na externí přílohy jsou přiloženy animace časového průběhu koncentrace KL

pro zdravý myokard ($d = 1$) a pro dva typy nezdravého myokardu (typ I: $d = 0,5$ a typ II: $d = 0,01$) s koeficientem přestupu $k = 0,1$. Pozorujeme rozdílné proudění KL ve zdravém a nezdravém myokardu. Pro volbu defektu $d = 0,01$ defektní oblastí protéká velmi omezené množství KL.

Poznamenejme, že animace jsou vyobrazeny ve dvou pohledech pro dané nastavení. Název souboru končící "xyz" zobrazuje řezy mimocévnního prostředí v osách x_1, x_2 a x_3 , zatímco soubor s koncovkou "xz" vyobrazuje řez mimocévním prostředím v osách x_1 a x_3 .



Obrázek 4.19: Průběh celkového množství KL M, M_c, M_m pro různou volbu koeficientu přestupu k a defektu d .

Závěr

Cílem této práce bylo matematické modelování transportu KL v cévním řečišti s přestupem do okolní tkáně a naopak. Cévní řečiště bylo reprezentováno náhodně generovaným grafem, ve kterém jsme určili tlak v každém vrcholu a následně jsme dopočítali rychlost na každé hraně grafu. Poté jsme řešili rovnice transportu KL v cévním a mimocévním prostředí. Dále jsme se zabývali přestupem KL mezi cévním a mimocévním prostředím. Na závěr jsme se zaměřili na nezdravý myokard, ve kterém jsme zkoumali časový vývoj celkového množství KL a vliv různých defektů.

V první kapitole jsme se zabývali rovnicemi proudění tekutin a transportu v porézním prostředí. Další část byla věnována struktuře cévního prostředí a rovnicím proudění a transportu KL v cévním prostředí. Následující část se zabývala propojením cévního řečiště a mimocévního prostředí pomocí metody vnořené hranice. Na závěr jsme zformulovali zkoumanou úlohu.

Druhá kapitola byla zaměřená na numerickou část této práce. Nejprve jsme diskretizovali cévní řečiště, mimocévní prostředí a dále jsme se věnovali aproximaci diferenciálních výrazů v rovnicích transportu KL. Poté jsme diskretizovali zdrojové členy a použili jsme spojitou aproximaci Diracovy delta funkce.

Implementační poznámky byly součástí třetí kapitoly, kde jsme vybrali stěžejní funkce implementovaného kódu. Jednou z těchto funkcí byla funkce generující cévní řečiště. Dále zde byly uvedeny funkce pro sestavení matic řešených rovnic a výpočet zdrojových členů.

V poslední kapitole jsme zkoumali numerické simulace na základě formulované úlohy z první kapitoly. První část této kapitoly byla rozdělena na 3 dílčí části. Nejdříve jsme zkoumali vliv prostoro-ového kroku na celkové množství KL v cévním řečišti bez přestupu do mimocévního prostředí. Dále jsme se zkoumali vliv volby koeficientu přestupu na celkové množství KL v systému. Obě dvě části byly provedeny pro jeden typ cévního řečiště. V poslední části jsme zkoumali vliv náhodně vygenerované realizace cévního řečiště na celkové množství KL. Ukázali jsme, že nedochází k úbytku/nárůstu KL v systému, tj. že dochází k zachování celkové hmoty. Dále jsme zjistili, že typ náhodně vygenerovaného cévního řečiště nemá velký vliv na časový vývoj celkového množství KL.

Další část čtvrté kapitoly byla věnována transportu KL v myokardu, který byl více zahuštěn cévami přidáním více identických cévních řečišť do oblasti. Následně jsme zkoumali časový vývoj celkového množství KL v celém systému. Na závěr jsme definovali úroveň poškození v defektní oblasti a porovnávali jaký je vliv úrovně defektu na celkové množství KL pro zdravý a nezdravý myokard.

V budoucnu bychom se mohli zaměřit na převedení numerických schémat na čistě implicitní včetně výpočtu zdrojových členů. Popřípadě bychom se mohli věnovat optimalizaci implementovaného programu. Dále bychom se mohli zabývat použitím představeného matematického modelu ke kvantifikaci perfuze myokardu pacientů na základě měření z magnetické rezonance ve spolupráci s Institutem klinické a experimentální medicíny v Praze (IKEM) a Children's Medical Center, University of Texas Southwestern v Dallasu (UTSW).

Odkazy na externí přílohy

Název	Odkaz	Sekce
Kód	https://github.com/horvalen/Diploma_thesis.git	3
Animace	https://github.com/horvalen/Diploma_thesis/tree/main/Video	4.2.2

Literatura

- [1] R. J. Leveque: *Finite Volume Methods for Hyperbolic Problems*. Cambridge University Press (2004).
- [2] Z. Chen, G. Huan, Y. Ma: *Computational methods for multiphase flows in porous media*. Society for Industrial and Applied Mathematics (2006).
- [3] R. Fučík, P. Eichler, R. Straka, P. Pauš, J. Klinkovský, T. Oberhuber: *On optimal node spacing for immersed boundary lattice Boltzmann method in 2D and 3D*. *Computer and Mathematics with Applications* 77 (2019), s. 1144-1162.
- [4] R. J. Leveque: *Finite Difference Methods for Differential Equations*. *AMath* 585-6 (1998).
- [5] C. S. Peskin: *The immersed boundary method*. *Acta Numerica* (2002), s. 479-517.
- [6] A. Cardona, T. Finkbeiner, J. C. Santamarina: *Natural Rock Fractures> From Aperture to Fluid Flow*. Springer (2021).
- [7] A. N. Cookson, J. Lee, C. Michler, R. Chabiniok, E. Hyde, D. Nordsletten, N. Smith: *Spatially Distributed Computational Model to Quantify Behaviour of Contrast Agent in MR Perfusion Imaging*. *Medical Image Analysis* 18 (2014), s. 1200-1216.
- [8] E. Vitásek: *Numerické metody*. SNTL-Nakladatelství technické literatury (1987).
- [9] P. M. Adler, J. F. Thovert: *Fractures and Fracture Networks*. Springer (1999).
- [10] L. Horvátová: *Matematické modelování transportu a přestupu kontrastní látky v problematice perfuze myokardu*. Bakalářská práce. České vysoké učení technické v Praze (2022).
- [11] T. Sauer, O. Runborg: *Numerical analysis*. Pearson (2011).
- [12] V. S. Vladimirov: *Equations of mathematical physics*. Marcel Dekker (1971).
- [13] J. Kovář: *Matematické modelování perfuze myokardu pomocí mřížkové Boltzmannovy metody*. Diplomová práce. České vysoké učení technické v Praze (2022).
- [14] T.W. Secomb: *A Green's function method for simulation of time-dependent solute transport and reaction in realistic microvascular geometries*. *Mathematical Medicine and Biology* (2015).
- [15] J.P. Alberging, T.W. Secomb: *Simulation of antiogenesis in three dimensions: Application to cerebral cortex*. *PLoS Computational Biolgy* (2021).
- [16] *Myokardinfarkt (rezidiviert)*. Heidelberg university. Navštíveno 31.3.2022. URL: <https://eliph.klinikum.uni-heidelberg.de/allg/57/myokardinfarkt-rezidiviert>.

- [17] T. Davis: *Algorithm 832: UMFPACK, an unsymmetric-pattern multifrontal method*. ACM Transactions on Mathematical Software (2004).
- [18] R. Ripa, T. George, K.R. Shumway, Y. Sattar: *Physiology, Cardiac Muscle*. National Library of Medicine (2023).
- [19] A. Zingaro, Ch. Vergara, L. Dede, F. Regazzoni, A. Quarteroni: *A comprehensive mathematical model for cardiac perfusion*. Scientific reports (2023).
- [20] P. Tyler, S. Butt: *Basic Principles of MRI*. Springer (2012).
- [21] J.D. Anderson, J. Wendt: *Computational fluid dynamics*. Sv. 206 Springer (1995).
- [22] L. Wasserman: *All of Statistics: a concise in statistical inference*. Springer (2005).
- [23] A.J. Pappano, W.G. Wier: *Cardiovascular physiology-e-book*. Elsevier Health Sciences (2018).
- [24] L. Horvátová: URL: https://github.com/horvalen/Diploma_thesis.git.
- [25] R. Fučík, P. Strachota: *Úvod do dynamiky kontinua*. Skripta (2023).
- [26] R. Fučík, J. Kovář, K. Škardová, O. Polívka, R. Chabiniok: *A lattice Boltzmann approach to mathematical modeling of myocardial perfusion*. Wiley (2024).
- [27] *Nadace pro rozvoj vzdělání*. Navštíveno 18.4.2024. URL: <https://www.nadaceprovzdelani.cz/>.
- [28] *Nadace "Nadání Josefa, Marie a Zdeňky Hlávkových"*. Navštíveno 18.4.2024. URL: <http://www.hlavkovanadace.cz/>.
- [29] *Nadace Český literární fond*. Navštíveno 18.4.2024. URL: <https://www.nclf.cz/>.