

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

DIPLOMOVÁ PRÁCE

Brno, 2021

Bc. Zuzana Hemzalová



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV BIOMEDICÍNSKÉHO INŽENÝRSTVÍ

DEPARTMENT OF BIOMEDICAL ENGINEERING

EVOLUČNÍ ALGORITMY PRO ULTRAZVUKOVOU PERFÚZNÍ ANALÝZU

EVOLUTION ALGORITHMS FOR ULTRASOUND PERFUSION ANALYSIS

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. Zuzana Hemzalová

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Martin Mézl, Ph.D.

BRNO 2021



Diplomová práce

magisterský navazující studijní program **Biomedicínské inženýrství a bioinformatika**

Ústav biomedicínského inženýrství

Studentka: Bc. Zuzana Hemzalová

ID: 186657

Ročník: 2

Akademický rok: 2020/21

NÁZEV TÉMATU:

Evoluční algoritmy pro ultrazvukovou perfúzní analýzu

POKYNY PRO VYPRACOVÁNÍ:

1) Seznamte se s principy ultrazvukové perfúzní analýzy a jednotlivými metodami pro stanovení perfúzních parametrů. 2) Proveďte rešerši evolučních optimalizačních algoritmů a vyberte vhodné algoritmy pro další práci. Pro práci je možné využití Optimization a Global Optimization Toolboxu v Matlabu. 3) Implementujte metody pro perfúzní analýzu založené na proložení křivky vhodným modelem. Proveďte základní testování a srovnání konvenčních algoritmů a vybraného evolučního algoritmu. 4) Nastudujte problematiku metody bolus & burst a navrhňte vhodné využití evolučních algoritmů pro tuto metodu. 5) Práci rozšiřte o implementaci dalších evolučních algoritmů pro globální optimalizaci. 6) Otestujte a srovnajte jednotlivé algoritmy na simulovaných a preklinických datech. Proveďte diskuzi získaných výsledků.

DOPORUČENÁ LITERATURA:

[1] HAUPT, Randy L a S. E HAUPT. Practical genetic algorithms. 2nd ed. Hoboken, N.J.: John Wiley, c2004. ISBN 9780471455653.

[2] JIRIK, Radovan, Kim NYLUND, Odd Helge GILJA, Martin MEZL, Vratislav HARABIS, Radim KOLAR, Michal STANDARA a Torfinn TAXT. Ultrasound perfusion analysis combining bolus-tracking and burst-replenishment. IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control. 2013, 60(2), 310-319. DOI: 10.1109/TUFFC.2013.2567. ISSN 0885-3010. Dostupné z: <http://ieeexplore.ieee.org/document/6416486/>

Termín zadání: 8.2.2021

Termín odevzdání: 21.5.2021

Vedoucí práce: Ing. Martin Mézl, Ph.D.

prof. Ing. Ivo Provazník, Ph.D.
předseda rady studijního programu

UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

Abstrakt

Tato diplomová práce se zabývá principy ultrazvukové perfuzní analýzy a metodami stanovení perfuzních parametrů. Zkoumá Evoluční algoritmy a jejich schopnost optimalizace při aproximaci dilučních křivek z ultrazvukového snímání tkáně. Porovnává výkon optimalizace tří evolučních algoritmu. Spojitého genetického algoritmu GA, algoritmu SOMA a PSO. Metody hodnotí na simulovaných a klinických datech.

Klíčová slova

Ultrazvukové kontrastní zobrazení, hemodynamické parametry, křivka časové intenzity, R kvadrát, difuzní driftové modely, Evoluční algoritmy, Rojové algoritmy, optimalizace, Farmakokinetické modely.

Abstract

This thesis deals with the principles of ultrasonic perfusion analysis and methods for determining perfusion parameters. It examines Evolutionary algorithms and their ability to optimize the approximation of dilution curves from ultrasound tissue scanning. It compares the optimization performance of three evolutionary algorithms. Continuous genetic algorithm GA, algorithm SOMA and PSO. Methods are evaluated on simulated and clinical data.

Keywords

Ultrasound contrast imaging, hemodynamic parameters, time intensity curve, R square, diffusion drift models, Evolutionary algorithms, Particle Swarm algorithms, optimization, Pharmacokinetic models.

Bibliografická citace

HEMZALOVÁ, Zuzana. *Evoluční algoritmy pro ultrazvukovou perfúzní analýzu* [online]. Brno, 2021 [cit. 2021-05-21]. Dostupné z: <https://www.vutbr.cz/studenti/zav-prace/detail/134472>. Diplomová práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav biomedicínského inženýrství. Vedoucí práce Martin Mézl.

Prohlášení autora o původnosti díla

Jméno a příjmení studenta: Zuzana Hemzalová
VUT ID studenta: 186657
Typ práce: Diplomová práce
Akademický rok: 2020/21
Téma závěrečné práce: Evoluční algoritmy pro ultrazvukovou perfúzní analýzu

Prohlašuji, že svou závěrečnou práci jsem vypracoval samostatně pod vedením vedoucí/ho závěrečné práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené závěrečné práce dále prohlašuji, že v souvislosti s vytvořením této závěrečné práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

V Brně dne: 21.5.2021

podpis autora

Poděkování

Děkuji vedoucímu diplomové práce panu Ing. Martinu Mězlovi, Ph.D. za účinnou metodickou, pedagogickou a odbornou pomoc a další cenné rady při zpracování mé diplomové práce.

V Brně dne: 21.5.2021

podpis autora

Obsah

SEZNAM OBRÁZKŮ	9
SEZNAM TABULEK.....	11
ÚVOD	12
1. DILUČNÍ TECHNIKY	14
1.1 MODELOVÁNÍ DILUČNÍHO SYSTÉMU.....	15
1.2 STOCHASTICKÁ REPREZENTACE MODELU.....	15
1.3 KONTRASTNÍ LÁTKY	16
1.3.1 Aplikace kontrastní látky.....	16
1.4 KŘIVKY ČASOVÉ INTENZITY	17
2. MATEMATICKÉ MODELY	18
2.1 LOGNORMÁLNÍ FUNKCE	19
2.2 ERLANGOVA FUNKCE	19
2.3 GAMA VARIÁČNÍ FUNKCE.....	20
2.4 DIFUZNÍ DRIFTOVÉ MODELY	21
2.4.1 LDRW model.....	22
2.4.2 FPT model.....	22
3. METODA BOLUST&BURST	24
3.1 METODA BURST-REPLENISHMENT	24
3.2 METODA BOLUS TRACKING	25
3.3 METODA BOLUS & BURST	25
3.4 FARMAKOKINETICKÝ MODEL	26
3.5 SLEPÁ DEKONVOLUCE.....	27
4. NEEVOLUČNÍ ALGORITMY	29
4.1 OPTIMALIZACE.....	29
4.2 NÁHODNÉ PROHLEDÁVÁNÍ.....	29
4.3 NELDER-MEAD SIMPLEXNÍ ALGORITMUS	29
4.3.1 Algoritmus Nelder-Mead.....	30
5. EVOLUČNÍ ALGORITMY	32
5.1 GENETICKÝ ALGORITMUS.....	32
5.1.1 Binární reprezentace populace	33
5.1.2 Spojitá reprezentace populace	33
5.1.3 Selektce	33
5.1.4 Křížení.....	34
5.1.5 Reprodukce	34
5.1.6 Mutace	35
5.1.7 Základní algoritmus jednoduchého GA	36
6. ROJOVÉ ALGORITMY.....	37
6.1 OPTIMALIZACE KOLONIÍ MRAVENCŮ (ACO).....	37
6.2 OPTIMALIZACE ROJEM ČÁSTIC (PSO).....	37

6.2.1	<i>Porovnání PSO a GA</i>	38
6.2.2	<i>Inicializace PSO</i>	39
6.3	OPTIMALIZACE SOMA	42
6.3.1	<i>Strategie migrace SOMA</i>	42
6.3.2	<i>Inicializace algoritmu a parametry SOMA</i>	43
6.3.3	<i>Základní operátory v SOMA</i>	45
7.	PRAKTICKÁ ČÁST	46
7.1	PRVNÍ ČÁST	46
7.1.1	<i>Farmakokinetické modely</i>	47
7.1.2	<i>Optimization toolbox Matlab</i>	49
7.1.3	<i>Spojité genetický algoritmus</i>	49
7.1.4	<i>První praktická část – A</i>	50
7.1.5	<i>První praktická část – B</i>	52
7.2	DRUHÁ PRAKTICKÁ ČÁST – SIMULOVANÁ DATA	54
7.2.1	<i>Spojité genetický algoritmus GA</i>	57
7.2.2	<i>Algoritmus PSO</i>	58
7.2.3	<i>Algoritmus SOMA</i>	59
7.2.4	<i>Vyhodnocení</i>	60
7.2.5	<i>Vyhodnocení MSE pro PSO a SOMA</i>	63
7.2.6	<i>Srovnání časové náročnosti GA, SOMA a PSO</i>	65
7.3	DRUHÁ ČÁST – KLINICKÁ DATA	68
8.	ZÁVĚR	76
	LITERATURA	77
	SEZNAM SYMBOLŮ A ZKRATEK	80
	SEZNAM PŘÍLOH	84

SEZNAM OBRÁZKŮ

Obrázek 1: Schéma modelu zkumavky rozdělené na více sektorů, pro zjednodušené znázornění diluční dynamiky. Vzdálenost mezi místem vpichu indikátoru x_i a měřicím místem x_m je rozdělena do $k - 1$ sektorů neboli kompartmentů stejné délky x . Převzato z [8]	15
Obrázek 2: Ukázka reflexe a expanze u simplexu se třemi vrcholy. Na Obrázku vlevo je původní simplex znázorněn přerušovanou čarou, souvislou čarou je simplex po kroku reflexe. Symbol x je těžiště a symbol xr je bod reflexe. Obrázek vpravo znázorňuje expanzi. Bod xe je bodem expanze. Převzato z [26].	31
Obrázek 3: Ukázka vnější, vnitřní kontrakce a zmenšení. Původní simplex je znázorněn přerušovanou čarou. Obrázek vlevo je simplex po vnější kontrakci, bod xc je bodem vnější kontrakce. Obrázek uprostřed znázorňuje vnitřní kontrakci. Bod xcc je bodem vnitřní kontrakce. Obrázek vpravo je ukázka zmenšení simplexu. Bod $x1$ je bodem s nejlepší funkční hodnotou. Převzato z [26].	31
Obrázek 4: ukázka křížení v jednom náhodně zvoleném bodě. Převzato z [15]	34
Obrázek 5: Princip SOMA. Obrázek převzatý z [20]. Jednotlivý jedinci jsou zobrazeny modrými kolečky a vůdce je červeným kolečkem. Trajektorie jedinců jsou zobrazeny pomocí šipek. Nový vůdci jsou zobrazeny kosočtvercem. Potomstva žlutým kolečkem.	45
Obrázek 6: Zjednodušený tříkompartmentový model použitý k vytvoření koncentrační křivky	47
Obrázek 7: Simulovaný průběh vstupu indikátoru do žíly, výstup z prvního kompartmentu M_1	48
Obrázek 8: Simulovaný průběh průchodu vstupu indikátoru do srdce, výstup z druhého kompartmentu M_2	48
Obrázek 9: Simulovaný průběh průchodu indikátoru tkání, výstup ze třetího indikátoru M_3	49
Obrázek 10: Boxplot pro diluční křivku rušenou šumem s hodnotou SNR= 19,77 dB.	52
Obrázek 11: boxplot pro diluční křivku rušenou šumem s hodnotou SNR=26,11 dB.	52
Obrázek 12: boxplot pro diluční křivku rušenou šumem s hodnotou SNR=35,43 dB	53
Obrázek 13: simulovaný průběh AIF	56
Obrázek 14: simulovaný průběh $C(t)$	56
Obrázek 15: Simulovaný průběh $C(t)$ se SNR=30 dB	57
Obrázek 16: boxplot spojitý GA, MSE pro SNR= 10–50 dB	60
Obrázek 17: boxplot spojitý GA, MSE pro SNR= 10–40 dB	61
Obrázek 18: boxplot PSO, MSE pro SNR= 10–40 dB	62
Obrázek 19: boxplot PSO, MSE pro SNR= 10–30 dB	62
Obrázek 20: boxplot SOMA, MSE pro SNR= 10–50 dB	63
Obrázek 21: MSE PSO vs SOMA se SNR=10 dB	63
Obrázek 22: MSE PSO vs SOMA se SNR=20 dB	64
Obrázek 23: MSE PSO vs SOMA se SNR=30 dB	64
Obrázek 24: MSE PSO vs SOMA se SNR=40 dB	65
Obrázek 25: MSE PSO vs SOMA se SNR=50 dB	65
Obrázek 26: boxplot Spojitý GA, čas pro SNR= 10–50 dB	66
Obrázek 27: boxplot Spojitý PSO, čas pro SNR= 10–50 dB	66
Obrázek 28: boxplot SOMA, čas pro SNR= 10–50 dB	67
Obrázek 29: Snímek nádoru, který byl podkožně implantován myši.	68
Obrázek 30: PSO exp1. Nejlepší proložení PSO pro 8 signálu experimentu 1	69
Obrázek 31: SOMA exp1. Nejlepší proložení SOMA pro 8 signálu experimentu 1.	69
Obrázek 32: Srovnání hodnot MSE pro PSO a SOMA v proložení 8 křivek u experimentu 1-3.	70
Obrázek 33: hodnoty MSE exp1-3 pro SOMA	71
Obrázek 34: hodnoty MSE exp1-3 pro PSO	71
Obrázek 35: boxploty vektorů hodnot časů aproximace u metody PSO pro exp1 – exp3	72

Obrázek 36: boxploty vektorů hodnot časů aproximace u metody SOMA pro exp1 –exp3.....	72
Obrázek 37: Srovnání metody fmincon a PSO u aproximace 3 křivky exp1, hodnota MSE=1,25 · 10-4 .	73
Obrázek 38: Srovnání metody fmincon a SOMA u aproximace 3 křivky exp1, hodnota MSE=2,0008 · 10-4.....	74
Obrázek 39: Srovnání metody fmincon a PSO u aproximace 5 křivky exp1, hodnota MSE=3,197 · 10-4	74
Obrázek 40: Srovnání metody fmincon a SOMA u aproximace 5 křivky exp1, hodnota MSE=1,4430 · 10-4.....	75

SEZNAM TABULEK

Tabulka 1: strategie dynamické změny interní váhy ω	41
Tabulka 2: Parametry optimalizace Gama funkce, Lognormální funkce, FPT a LDRW	46
Tabulka 3: Parametry Spojitého GA.	50
Tabulka 4: Hodnoty Parametrů po proložení diluční křivky 1. metodou nejmenších čtverců.	50
Tabulka 5: Hodnoty parametrů po proložení diluční křivky 1. metodou spojitého genetického algoritmu.	50
Tabulka 6: Hodnoty parametrů po proložení diluční křivky 2. metodou nejmenších čtverců.	51
Tabulka 7: Hodnoty parametrů po proložení diluční křivky 2. metodou spojitého genetického algoritmu.	51
Tabulka 8: Hodnoty maxima, minima, mediánu a rozptylu 100 hodnot R kvadrátu pro diluční křivku rušenou šumem s hodnotou SNR=19,77 dB u spojitého GA.	53
Tabulka 9: Hodnoty maxima, minima, mediánu a rozptylu 100 hodnot R kvadrátu pro diluční křivku rušenou šumem s hodnotou SNR= 26,11 dB u spojitého GA.	53
Tabulka 10: Hodnoty maxima, minima, průměru a rozptylu 100 hodnot R kvadrátu pro diluční křivku rušenou šumem s hodnotou SNR=35,43 dB u spojitého GA.	53
Tabulka 11: Hodnoty R kvadrátu vypočítaného metodou nejmenších čtverců pro tři realizace šumu a čtyři teoretické funkce Gama, lognormální, LDRW, FPT.	54
Tabulka 12: hodnoty parametrů GA	57
Tabulka 13: Hodnoty parametrů PSO.....	58
Tabulka 14: Hodnoty Parametrů SOMA	59

ÚVOD

Zobrazovací techniky používané v současné době pro cévní diagnostiku, jako například rentgenová angiografie, ultrazvuk a magnetická rezonance, jsou schopny rozlišit relativně velké struktury ve vaskulárním systému. Pro nás ovšem zůstává velkou výzvou odvození hemodynamiky z cév o průměru menších než cca 1 mm. Tyto cévy jsou důležité z fyziologického hlediska proto, že zajišťují integritu tkáně na úrovni perfuze. Jejich průměr je menší než $50\mu\text{m}$ a rychlost průtoku krve cévou je menší než $1\text{ mm}\cdot\text{s}^{-1}$. V současné době dokážeme detekovat jejich hromadný účinek na tkáňový obraz. Příkladem je tzv. „červenání tkáně“ v angiografii, změny času T2 v magnetické rezonanci, heterogenita ve fázovém obraze a posuny frekvence v nízkofrekvenčním Doppleru, detekovaných pomocí ultrazvukových kontrastních látek. Tyto techniky nám pomáhají k detekci změn objemového průtoku krve tkání, které doprovází např. ischemii myokardu [5].

Existují také choroby, jejichž dopad na mikrocirkulaci se odráží spíše v její morfologické struktuře než v hemodynamických parametrech. Takovou nemocí je třeba rakovina. Mnoho nádorů existuje v in situ po značně dlouhé období (měsíce až roky) v avaskulárním klidném stavu. Takové nádory nerostou nad objem určený velikostí difuze kyslíku ve tkáni. Maligní růst je možný pouze po vytvoření cévního zásobení, to je dosaženo procesem známým jako angiogeneze [5]. Jedná se o proces novotvorby cév, který nachází uplatnění jak za fyziologických okolností, tak v patogenezi různých chorobných stavů. Angiogeneze je vícestupňový proces, který bývá označován jako angiogenní kaskáda. Proces je regulován souborem humorálních působků, zejména cytokinů, integrinů, adhezních molekul, proteolytických enzymů a nízkomolekulárních látek. Za fyziologických okolností jsou faktory stimulující novotvorbu cév v rovnováze s faktory antiangiogenními. Ztráta rovnováhy mezi stimulatory a inhibitory angiogeneze se projeví poruchou, jejíž charakter závisí na tom, která ze skupin regulátorů nabyla vrcholu. Zvýšené koncentraci angiogenních faktorů s následným zvýšením novotvorby cév je připisovaná významná úloha v patogenezi nádorového procesu a metastazování [6]. Nové kapiláry proniknou do nádorové hmoty a zajišťují přívod kyslíku a živin pro nádor a také cestu k metastazování do vzdálených míst. Vaskulární síť nádoru je chaotická a zamotaná na rozdíl od hierarchického větvení cév v normálním orgánu. Maligní transformace je důležitá pro pochopení progresu nádoru [5].

V důsledku toho existuje potřeba neinvazivních zobrazovacích metod posoudit jejich účinek. Mnoho zobrazovacích metod využívají indikátory, jejichž průchod může být detekován v krevním systému. Tyto indikátory se obvykle používají k poskytování informací o hemodynamice, např. intravaskulární objem a střední průtok. Předpokládáme tedy, že abnormální vaskulární struktura nádorů může být odhalena ve

formě křivek časové intenzity průchodu indikátoru krevním řečištěm [5]. Tato práce se věnuje evolučním a rojovým algoritmům pro optimalizaci dilučních křivek. Pomocí proložení diluční křivky vhodným matematickým modelem lze metodami optimalizace odhadnout perfuzní parametry diluční křivky.

1. DILUČNÍ TECHNIKY

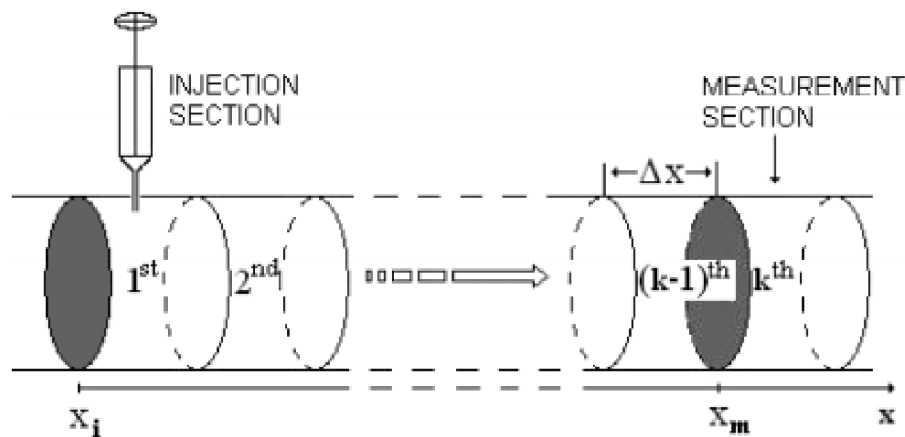
Diluční techniky se používají v kvantifikaci makro a mikrovaskulárního toku krve v mnoha klinických aplikacích. Umožňují odhadovat parametry hemodynamiky orgánu i nádorů a díky tomu můžeme dostat užitečné informace z oblasti diagnostiky a terapie. Diluční metody jsou široce využívány v radiologických aplikacích, jako je ultrazvuk, scintigrafie a zobrazování magnetickou rezonancí a počítačová tomografie [1].

Bylo navrženo několik teoretických formulací vysvětlujících tvar diluční křivky indikátoru. Jeden matematický přístup využívá integrály konvoluce, ve které je koncentrace indikátoru považována za neznámou nespécifikovanou funkci času $C(t)$. Rozsáhlé použití této metody je však omezené. Pokud aplikujeme tyto matematické formulace na křivku získanou *in vivo*, dochází k vysoké výpočetní a časové náročnosti. Hledáme proto specifickou funkci, která by vhodně vyjádřila koncentraci indikátoru. Po nalezení této funkce bychom mohli přesněji charakterizovat normální a abnormální diluční křivky indikátoru a získat přehled o některých faktorech ovlivňující tvar křivky [7].

Hemodynamická kvantifikace pomocí indikátoru injektovaného do krevního řečiště má dlouhou historii. Indikátor bolusu je vstříkován ve směru toku a měří se křivka koncentrace indikátoru v závislosti na čase, také označovaná jako diluční křivka indikátoru IDC. Diluční křivka indikátoru se měří proti proudu. Měření diluce indikátoru má v klinické diagnostice několik aplikací. Zpočátku pro kardiovaskulární kvantifikaci byly použity indikátory, které byly zaměřeny především na hodnocení objemu krve a srdečního výdeje. V současnosti se také používají na detekce myokardní ischemie, mozkové ischemie a angiogenetických procesů související s růstem rakoviny. Výpočet objemu krve a perfuze je založen na odhadu průměrné doby průchodu indikátoru a střední doby přepravy MTT. Přesný odhad tohoto parametru má zásadní klinický význam. V důsledku recirkulace indikátoru v oběhovém systému a nízkému poměru signál šum (SNR) měřených dilučních křivek, hledáme modely, které by vhodně aproximovali diluční křivky. Z modelů pak můžeme spočítat hemodynamické parametry systému [8].

Modely, které se běžně využívají v klinickém zařízení, jsou lognormální a gama modely. Tyto modely jsou empiricky vybrány kvůli jejich podobnosti s měřenými IDC, bez fyzikální interpretace ředícího procesu. Na rozdíl od modelu LDRW, který poskytuje fyzikální interpretaci ředícího procesu a je založen na řešení difuze s driftovou rovnicí. Objem krve lze vypočítat z MTT vynásobením průtokem Q dle následující rovnice [8]:

$$V = MTT \cdot Q. \quad (1.1)$$



Obrázek 1: Schéma modelu zkumavky rozdělené na více sektorů, pro zjednodušené znázornění diluční dynamiky. Vzdálenost mezi místem vpichu indikátoru x_i a měřícím místem x_m je rozdělena do $k - 1$ sektorů neboli kompartmentů stejné délky x . Převzato z [8]

1.1 Modelování dilučního systému

Pro účely modelování diluce může být hemodynamický model znázorněn jako nekonečná trubice podél osy x , viz. Obrázek 1. Trubicí teče nosná tekutina a přenáší indikátor z místa injekce x_i do místa měření x_m . Pro zjednodušení a bez ztráty obecnosti můžeme předpokládat $x_i = 0$. Trubicí uvažujeme jako multi-kompartmentový model, jednotlivé řezy dělí kompartmenty o stejné délce x a objemu:

$$V_c = \Delta x \cdot A, \quad (1.2)$$

kde A je plocha průřezu trubice. Použití tohoto modelu nám redukuje popis procesu diluce na reprezentaci jednorozměrné dynamiky. Transport a šíření indikátoru podél osy trubice, definované prostorovou souřadnicí x . Uvažujeme tedy model jako kaskádu stejných oddílů (části trubek) s konstantním průtokem. Každý oddíl lze považovat za směšovací komoru s jednosměrným vymýváním indikátoru [8].

1.2 Stochastická reprezentace modelu

Uvažujeme náhodnou proměnou jako pozici částice indikátoru podél trubice. K popisu koncentrace indikátoru jako hustoty pravděpodobnosti přechodu částic (vymývání) z jedné sekce (kompartmentu) do následující. Přechod indikační částice je reprezentován binomickým rozdělením. Pohyb částice indikátoru je reprezentován jako výsledek $n\Delta t$ pokusu, kde n je počet pokusů za časový interval Δt . Každému pokusu přidělujeme pravděpodobnost p , že dojde k přechodu částice do následujícího kompartmentu a pravděpodobnost $1 - p$, kdy částice zůstává v kompartmentu. Předpokládáme tedy jednosměrný pohyb. Pro $n \gg np \gg p$, tedy pro velké n a malé p je Poissonovo rozdělení dobrou aproximací binomického rozdělení s konstantou λ , kde:

$$\lambda = n \cdot p. \quad (1.3)$$

Pravděpodobnost k přechodů za čas t můžeme aproximovat Poissonovým rozdělením:

$$f_c(t) = \frac{e^{-\lambda t} (\lambda t)^k}{k!}. \quad (1.4)$$

Parametr λ označuje rychlost vymytí indikátoru [8].

1.3 kontrastní látky

K detekci mikrocirkulace lze použít ultrazvukové kontrastní látky. Dopplerovské ultrazvukové systémy pomocí kontrastních látek a nelineárního zobrazování dokážou detekovat odrazy z průtoku krve v cévách až do průměru 40 μm [4].

Mezi vlastnosti ideální kontrastní látky využívané při ultrazvukovém zobrazení patří:

- netoxicity
- injektované intravenózně
- schopné pocházet plicním, srdečním a kapilárním oběhem [4]
- mají srovnatelnou velikost s červenými krvinkami (průměr menší než $\sim 10 \mu\text{m}$) nedifundují ani neucpávají cévy
- chovají se jako čisté krevní stopovače
- zvyšují kontrast obrazu krve [1][3].

Kontrastní látky nám zajistí zvýšení diagnostické schopnosti v celé řadě normálních a abnormálních cév a orgánů celého těla [4]. Typické maligní nádory bez schopnosti nádorové novotvorby vazivového stromatu mají rychlý vzestup koncentrace kontrastní látky, následovaný více či méně vyjádřeným zachováním stejné úrovně koncentrace (plateau) a postupným vymýváním kontrastní látky (wash-out). Tkáně benigní povahy mají naopak pomalejší lineární či exponenciální vzestup koncentrace látky, eventuálně velmi rychlý vzestup a rychlý pokles u tkání, kterou kontrastní látka jen prochází [11].

1.3.1 Aplikace kontrastní látky

Standardní techniky jako termodiluční techniky nebo diluční techniky barvivovým indikátorem, které se uplatňují pro odhad srdeční parametrů jsou velmi invazivní z důvodu katetrizace. Zavedení stabilních ultrazvukových kontrastních látek (zapouzdřené bubliny od 1 μm do 10 μm) nám umožňují nahrazení standardních technik novějšími minimálně invazivními technikami [2]. Kontrastní látka může být zavedena do těla buď jako bolusová injekce nebo jako konstantní infuze. Při bolusové injekci se obvykle používá nízké MI a zobrazování v reálném čase. Mezi výhody bolusového podání kontrastní látky je to, že nám postačí menší množství kontrastní látky než u infuze. Dále nemusíme kontrolovat průtok, rychlost a množství kontrastní látky jako u infuze [1].

1.4 Křivky časové intenzity

Křivka intenzity obrazu jako funkce času nám kvantifikují průtok krve v oblasti zájmu (ROI). Často jsou při snímání křivky rušené šumem. Při mírných koncentracích mikrobublin je akustické stínění zanedbatelné a intenzita I je přímo úměrná koncentraci. Proto je možné použít teoretické modely založené na technikách ředění indikátoru pro ultrazvukové kontrastní látky, které data přizpůsobí křivce, aby potlačily šum a izolovaly primární průchod. Tento přístup také umožňuje analytické stanovení důležitých parametrů související s hemodynamikou, jako je plocha pod křivkou časové intenzity (AUC), střední doba přepravy (MTT) a čas do dosažení maximální intenzity (t_p). Tyto parametry souvisejí s průtokem krve a objemem krve v ROI. Tento přístup vede k přesnějším výsledkům než jednoduché vyhlazovací techniky, které se často používají k vyloučení odlehlých hodnot z dat, ale nelze je použít k odstranění účinku recirkulace [1].

2. MATEMATICKÉ MODELY

Teorie diluce indikátoru se často používá pro měření klinických parametrů. Mnohé diluční techniky vyžadují katetrizaci a následnou hospitalizaci k určení klinických parametrů. Ultrazvukové kontrastní látky umožňují neinvazivní měření u ambulantních specialistů. Koncentrace kontrastní látky lineárně souvisí s akustickým zpětným rozptylem. Nevýhodou ultrazvukových obrazů je nízký poměr signálu k šumu, jejich diluční křivky jsou znehodnocené šumem. Důsledek toho je, že proložení těchto křivek je složitější. Proto se využívají matematické modely, které zlepšují interpretaci signálu [2]. Teoretické modely počítají s tím, že částice indikátoru po okamžité bolusové injekci procházejí ROI v různých časech. Indikátor je rozptýlen Brownovým pohybem, laminárním prouděním nebo turbulencí. V teoretickém modelu uvažujeme bolus indikátoru jako funkci mající tvar krátkého obdélníkového impulsu, blížící se svým tvarem funkci delta. Ve skutečnosti se indikátor do určité míry rozptýlí, proto uvažujeme obdélníkový tvar počátečního impulsu jako aproximaci skutečnosti. Diluční křivka je funkce hustoty pravděpodobnosti doby průchodu indikátoru přes systém. Zabýváme se množstvím indikátoru, který projde oblastí zájmu za určitou dobu. Pokud uvažujeme množství indikátoru v injekci m a koncentraci jako funkci času naší oblasti zájmu. Pak můžeme vypočítat objemový průtok krve F a objem krve V pomocí AUC a MTT [1]:

$$F = m \cdot (AUC)^{-1} \quad (2.1)$$

$$V = F \cdot MTT. \quad (2.2)$$

Tyto vztahy jsou známé jako Stewart-Hamiltonovi a nedávají nám žádné informace o tvaru dilučních křivek. U velkého průtoku krve se mikrobublinky rychle přesunou do oblasti zájmu, proto je doba dosažení maximální intenzity tp kratší. Proto ve vysoce vaskularizovaných zhoubných nádorech očekáváme rychlejší nárůst křivky maximální intenzity než u běžného parenchymu [1].

Steward- Hamiltonovi vztahy jsou platné za těchto předpokladů:

1. Platí zákon zachování hmotnosti.

Ve skutečnosti však zákon neplatí, protože se velká část indikátoru v těle ztratí dříve, než dosáhne oblasti zájmu. Například přirozeným poškozením bublin a filtračním účinkem přes transpulmonální oběh.

2. Představa systému je kompartment s jedním vstupem a výstupem.
3. Okamžité vstříknutí indikátoru lze aproximovat delta impulzem.

Tento předpoklad se více blíží skutečnosti, pokud střední doba přepravy je mnohem větší než doba trvání injekce.

4. Průtok krve je konstantní.

Tento předpoklad platí jen v některých případech. Skutečnosti se blíží, pokud čas měření je dostatečně malý.

5. Systém je stacionární.

6. Není narušen průtok krve injekcí indikátoru.

Platí pro většinu případů, protože množství injektovaného indikátoru je velmi malý ve srovnání s celkovým objemem krve.

7. Cévy mají neměnnou velikost a krev je nestlačitelná tekutina [1].

2.1 Lognormální funkce

Náhodná veličina t má lognormální rozdělení s parametry μ a σ^2 , pokud náhodná veličina $\ln(t)$ má normální rozdělení $N(\mu, \sigma^2)$ [39]. Lognormální distribuční funkce se zpožděním t_0 je dána vztahem:

$$I(t) = \frac{AUC}{\sqrt{2\pi}\sigma(t-t_0)} e^{\frac{(\ln(t-t_0)-\mu)^2}{2\sigma^2}} + I_0, \quad t > t_0. \quad (2.3)$$

Kde $I(t)$ je intenzita signálu, která je úměrná koncentraci indikátoru. V grafu $I(t)$ vidíme jako funkci závislou na čase. Proměnná μ je průměr a σ je standardní odchylka normálního rozdělení logaritmu nezávislé proměnné t . Kde σ je parametrem tvaru a μ je parametr umístění. Při změně μ dochází ke změně polohy křivky ve směru osy x a při změně σ dochází ke změně tvaru křivky. Při zvyšování σ se zkosení křivky zvětšuje. Počáteční intenzita I_0 má vykompenzovat přítomnou hladinu hluku v obraze. I_0 není součástí původního vzorce lognormální funkce vycházejícího ze statistických modelů. Proměnná slouží k lepší aproximaci pro výstupy z ultrazvuku. Počáteční hodnotu bolusu nastavíme na nulu. Parametr t_0 je zapotřebí kvůli obtížnosti přesné identifikace času prvního příchodu kontrastu v důsledku malého poměru signál-šum před příchodem bublin do ROI. Kde MTT a t_p je dáno vztahem [1]:

$$MTT = e^{\mu+\sigma^2/2}, \quad (2.4)$$

$$t_p = e^{\mu-\sigma^2}. \quad (2.5)$$

MTT definujeme jako první moment pravděpodobnostní hustotní funkce $(I(t) - I_0)$ minus čas příchodu bolusu t_0 a t_p [1].

2.2 Erlangova funkce

U Erlangova rozdělení předpokládáme, že konstantní průtok krve lze modelovat jako řadu n homogenních kompartmentů stejného objemu V_c , přičemž každý kompartment má jen jeden vstup a jeden výstup. Po okamžité injekci indikačního bolusu je rychlost změny koncentrace indikátoru $C_i(t)$ v kompartmentu i ($i = 1, \dots, n$) dána těmito diferenčními rovnicemi:

$$\frac{dC_1}{dt} = \frac{m}{V_c} \delta(t) - \frac{F}{V_c} C_1. \quad (2.6)$$

$$\frac{dC_i}{dt} = \frac{F}{V_c} [C_{i-1}(t) - C_i(t)], \quad i = 2, \dots, n. \quad (2.7)$$

Kde m je množství indikátoru a V_c je objem. Parametr $\delta(t)$ je Diracova delta funkce. Rovnice platí za předpokladu, že dochází k úplnému promíchání částic indikátoru s krví v každém kompartmentu. Pohyb indikátoru je z jednoho kompartmentu do druhého jednosměrný, proces difuze je tedy zanedbán. Pokud máme pouze jeden kompartment, jedná se o exponenciální rozpad. Po vyřešení diferenciálních rovnic dostáváme $C_n(t)$, které je úměrné $I(t)$:

$$I(t) = \frac{AUC}{\beta^n (n-1)!} (t - t_0)^{n-1} e^{-(t-t_0)/\beta} + I_0. \quad (2.8)$$

Funkce $(I(t) - I_0)$ je známá jako Erlangova pravděpodobnostní funkce hustoty se zpožděním t_0 . Parametr β se nazývá parametr rychlosti, protože je rovnocenný časové konstantě reakce jednoho oddílu na impulsní bolus:

$$\beta = V_c/F. \quad (2.9)$$

Počet oddílů n je také známý jako parametr tvaru, protože šikmost distribuce je rovna $2/\sqrt{n}$, $I(t)$ se stává symetričtější s přibývajícím počtem oddílů [1].

2.3 Gama variační funkce

Gama variační model pro proložení diluční křivky ředění indikátoru lze definovat jako:

$$C_\Gamma(t) = A_\Gamma \cdot t^\alpha \cdot e^{-\frac{t}{\beta}}, \quad (2.10)$$

kde α je parametrem tvaru a β je parametrem měřítka. Parametr A_Γ definuje amplitudu diluční křivky. Na tento model lze nahlížet jako na formální zjednodušení funkce hustoty pravděpodobnosti gama, která je daná vztahem:

$$f_\Gamma(t; \lambda, k) = \frac{\lambda e^{-\lambda t} (\lambda t)^{k-1}}{\Gamma(k)}. \quad (2.11)$$

Kde λ je měřítkový a k je tvarový faktor. Operátor $\Gamma(\cdot)$ reprezentuje operátor Gamma daný vztahem:

$$\Gamma(x) = \int_0^\infty y^{x-1} e^{-y} dy \quad x \text{ a } y \in \mathbb{R}. \quad (2.12)$$

Rovnice je zobecněním faktoriálu pro neceločíselná čísla. Gama variační model $C_\Gamma(t)$ lze snadno odvodit z operátoru Gamma $\Gamma(x)$ pro:

$$k - 1 = \alpha \quad (2.13)$$

$$\lambda = \beta^{-1} \quad (2.14)$$

$$A_\Gamma = \lambda^k \Gamma(k)^{-1} \quad (2.15)$$

Vztah mezi gama funkcí a faktoriálem je:

$$\Gamma(k) = (k - 1)!. \quad (2.16)$$

Distribuce gama je zobecnění Erlangovy distribuce pro neceločíselná čísla [8].

Po dosazení do Erlangovy funkce dostáváme vztah pro Gama funkci:

$$I(t) = \frac{AUC}{\beta^{\alpha+1}\Gamma(\alpha+1)!} (t - t_0)^\alpha e^{-(t-t_0)/\beta} + I_0, \quad (2.17)$$

kde α je parametrem tvaru a β je parametrem rychlosti. Jmenovatel $\beta^{\alpha+1}\Gamma(\alpha+1)$ normalizuje gama funkci. To znamená, že funkce $(I(t) - I_0)$ je pravděpodobnostní funkce, jejíž integrace se blíží jedné, když se AUC rovná jedné [1].

Vztah po střední dobu přepravy MTT a maximální intenzitu t_p je dán rovnicemi:

$$MTT = \beta(\alpha + 1), \quad (2.18)$$

$$t_p = \alpha \cdot \beta. \quad (2.19)$$

Při zvyšování parametru α , za stejné hodnoty β , se křivky stávají symetričtější (méně zkosené). Se zvyšující se hodnotou rychlostního parametru β se snižuje čas růstu a poklesu křivky. Gama funkce je na rozdíl od Erlangovy funkce obecnější, protože umožňuje neceločíselné hodnoty parametru α . Funkce gama může být rozšířena tak, aby vyhovovala oddílům s nesterjnými objemy. V takovém případě je výsledek součet n jednotlivých exponenciál, jejíž součet závisí na různých objemech kompartmentů. Nicméně se zjistilo, že s rostoucím počtem parametrů gama funkce nevede ke zlepšení aproximace reálné diluční křivky [1].

2.4 Difuzní driftové modely

Popisují vstřikování indikátoru do přímé nekonečně dlouhé trubice, kde tekutina (nosič) proudí konstantní rychlostí u . Předpokladem tohoto modelu je bolus indikátoru (Diracův impuls) a Brownův pohyb indikátoru. Částice interagují čistou elastickou kolizí. Bez ztráty obecnosti předpokládáme čas t_0 a pozici $x(t_0)$ jako nulové. Pokud se zaměříme na jednu částici lze její pozici $X(nt)$ a čas nt popsat náhodným procesem:

$$X(nt) = \sum_{i=1}^n S(iT). \quad (2.20)$$

S je náhodná proměnná, která představuje vzdálenost, kterou urazila částice za interval T (jeden krok). V důsledku hypotézy Brownova pohybu je každý krok $S(iT)$ nezávislý na předchozím a $X(nT)$ je řešením Markova procesu. Proto při zvětšení $n(T)$ platí centrální limitní teorém. Rovnice vyjadřuje funkci hustoty pravděpodobnosti náhodné proměnné X v čase nt s průměrem μ a střední směrodatnou odchylkou σ .

$$I(x, nT) = \frac{AUC}{\sqrt{2\pi n\sigma^2}} e^{-\frac{(x-n\mu)^2}{2n\sigma^2}}. \quad (2.21)$$

Pokud máme spojitý čas s nekonečně malým T :

$$t = n \cdot T, \quad (2.22)$$

lze rovnici vyjádřit Wienerovým procesem:

$$I(x, t) = \frac{AUC}{\sqrt{2\pi t\alpha}} e^{-\frac{(x-t\mu)^2}{2t\alpha}}. \quad (2.23)$$

Kde platí následující vztahy:

$$\alpha = \frac{\sigma^2}{T}, \quad (2.24)$$

$$u = \frac{\mu}{T}. \quad (2.25)$$

Koncentrace indikátoru $C(x, t)$ je určena m/A , kde m je hmotnost injektovaného indikátoru a A je sekce trubice. Tak jak vidíme na obrázku 1 je $C(x, t)$ popsáno normálním rozložením, které se pohybuje podél trubice se stejnou rychlostí jako tok krve. S průměrnou hodnotou rychlosti tu a rozptylem, který je lineární funkcí času (rozptyl se rovná $t\alpha$). Pokud uvažujeme $\alpha = 2D$, kde D reprezentuje difuzní koeficient. $C(x, t)$ je řešením jednorozměrné difuze s driftovou rovnicí. Difuzní driftové modely předpokládají pohyb částice indikátoru přes oběhový systém jako podélnou difuzi [32]. Rovnice, která popisuje tento jev je dána vztahem:

$$\frac{\partial C(x, t)}{\partial t} = D \frac{\partial^2 C(x, t)}{\partial x^2} - v \frac{\partial C(x, t)}{\partial x}, \quad (2.26)$$

kde $C(x, t)$ je koncentrace indikátoru v poloze x v čase t . D je koeficient podélné difuze a v je rychlost toku krve. Ve složitých systémech např. průtok krve v mikrocirkulaci nádoru je D kolektivní parametr, který představuje příspěvky různých fyzikálních mechanismů. Jako je například turbulentní proudění, Brownův pohyb a transport na mikrovaskulární úrovni s difuzní architekturou. Mezi difuzní modely patří model LDRW a FPT model [1].

2.4.1 LDRW model

Funkce LDRW patří mezi difúzně driftové modely. Název LDRW je zkratkou anglického názvu *local density random walk*. Jak už bylo zmíněno v teorii difuzních driftovéch modelů, model předpokládá, že částice indikátoru vykonává náhodný pohyb, který vede k postupnému šíření částic ve směru centrální osy x . Vztah pro intenzitu signálu $I(t)$, střední dobu přepravy MTT a maximální intenzitu t_p je dán těmito vztahy:

$$I(t) = AUC \cdot \left(\frac{e^\lambda}{\mu}\right) \sqrt{\frac{\mu}{(t-t_0)} \frac{\lambda}{2\pi}} \cdot e^{\left[-\frac{1}{2}\lambda\left(\frac{\mu}{(t-t_0)} + \frac{(t-t_0)}{\mu}\right)\right]} + I_0, \quad (2.27)$$

$$MTT = \mu, \quad (2.28)$$

$$t_p = \left(\frac{\mu}{2\lambda}\right) \left(\sqrt{1 + 4\lambda^2} - 1\right), \quad (2.29)$$

$$\mu = x_0/v. \quad (2.30)$$

Parametr x_0 je vzdálenost mezi počátkem a koncem křivky ROI. Parametr μ je střední doba potřebná k přepravě indikátoru na vzdálenost x_0 . Parametr λ je definován vztahem:

$$\lambda = \mu v^2 / 2D. \quad (2.31)$$

Parametr D reprezentuje difuzní koeficient a λ^{-1} je koeficient šikmosti křivky [1].

2.4.2 FPT model

Také jako LDRW model představuje IDC jako výsledek průchodu injektovaného indikátorového bolusu (rychlá injekce popsána Diracovou funkcí), která proudí v

dynamickém systému toku (nekonečně dlouhá trubice), přes detekční sekci [1]. Jediný rozdíl mezi FPT a LDRW se týká průchodu indikátoru přes místo detekce. Model FPT umožňuje pouze jediný průchod částic indikátoru, zatímco model LDRW uvažuje více průchodů. Model FPT je odvozen od modelu LDRW, kdy je uvažován pouze první průchod detekčním místem [32]. Vztah pro intenzitu signálu $I(t)$, střední dobu přepravy MTT a maximální intenzitu t_p je dán těmito vztahy [1]:

$$I(t) = AUC \cdot \left(\frac{e^\lambda}{\mu}\right) \sqrt{\frac{\lambda}{2\pi}} \cdot \left(\frac{\mu}{(t-t_0)}\right)^{\frac{3}{2}} \cdot e^{\left[-\frac{1}{2}\lambda\left(\frac{\mu}{(t-t_0)} + \frac{(t-t_0)}{\mu}\right)\right]} + I_0, \quad (2.32)$$

$$MTT = \mu, \quad (2.33)$$

$$t_p = \left(\frac{\mu}{2\lambda}\right) \left(\sqrt{9 + 4\lambda^2} - 3\right). \quad (2.34)$$

3. METODA BOLUST&BURST

Jedná se o parametrickou ultrazvukovou perfuzní analýzu kombinující metody Bolus tracking a Burst Replenishment. Metody perfuzní analýzy se většinou používají v sledování diagnostiky a terapie u ischemických a onkologických onemocnění [40]. Metoda je také schopna rozlišit mezi nádorovou a zánětlivou tkání. Tato metoda má kapacitu nám určit absolutní hodnoty průtoku, objemu a průměrné hodnoty průchodu krve a průměrnou dobu přepravy krve. Dále nám umožňuje určit odhad lokální arteriální vstupní funkce, která charakterizuje arteriální strom. Díky tomu jsme schopni určit čas příchodu bolusu. Mezi přínosy Bolust&Burst, jako u jiných metod ultrazvukového zobrazování, patří práce v reálném čase a nízké ceny. Mezi nevýhody lze zařadit artefakty (např. útlum) a špatná kvalita v ultrazvukových záznamech. Tyto nevýhody spějí k horší kvantitativní perfuzní analýze [42].

V perfuzním zobrazování zaznamenáváme časovou závislost změny koncentrace kontrastní látky ve sledované tkáni po intravaskulárním podání kontrastní látky. Mezi ultrazvukové kontrastní látky patří mikrobubliny naplněné plynem, které nevstupují do extravaskulárního prostoru. Díky této metodě můžeme odhadnout průtok F_b a objem krve V_b a průměrnou dobu přepravy kontrastní látky MTT . K modelování fáze Bolusu a fáze doplňování se využívá farmakokinetický model založený na konceptu arteriální vstupní funkce a tkáňové reziduální funkce [42].

3.1 Metoda Burst-Replenishment

U metody Burst-Replenishment (reperfuze) se kontrastní látka aplikuje jako infuze. U zobrazování tkáně se nejprve použijí ultrazvukové pulzy s vysokým mechanickým indexem MI , které zničí mikrobubliny kontrastní látky v analyzované oblasti zájmu ROI . Index MI se pohybuje obvykle kolem hodnoty 1. Poté jsou aplikované nízkoenergetické ultrazvukové pulzy, s nízkou hodnotou MI , které jsou citlivé na mikrobubliny např. pulzní inverze. Časový vývoj koncentrace kontrastního činidla ve fázi doplňování (replenishment), reperfuze, získanou po destrukci bublin modelujeme jako exponenciální funkci nebo komplexnější funkci. Lze modelovat jako standardní monokompartmentový model nebo jako více komplexní model [42].

U jednoduše kompartmentových modelů je maximum křivky A rovno objemu krve V_b analyzované frakce tkáně. Pomocí metody Burst-Replenishment můžeme určit objem krve V_b a krevní průtok na objem tkáně F_b . Pomocí odhadu časové konstanty exponenciální funkce β , je množství $A\beta$ úměrné F_b [40][42]. Mezi nevýhody tohoto přístupu patří to, že veličiny β a $A\beta$ jsou úměrné pouze perfuzním parametrům a konstanty úměrnosti nejsou známy a závisí na mnoha faktorech. Mezi tyto faktory například patří: dávkování a rychlost perfuze, clearance kontrastní látky, nastavení ultrazvukového skeneru a útlum. Hodnoty těchto faktorů se liší v závislosti na druhu vyšetření a u útlumu také na pozici v obraze [42].

3.2 Metoda Bolus Tracking

Metoda bolus tracking je založena na bolusovém podání kontrastní látky. Během průchodu stopovače se měří křivka koncentrace. Analýzou koncentrace kontrastní látky a analýzou křivek intenzity obrazu (průměrná intenzita obrazu v ROI tkáně versus čas) dostaneme odhad několika semikvantitativních a kvantitativních parametrů perfuze. Mezi semikvantitativní parametry patří plocha pod křivkou, časový interval k dosažení peaku, sklon, intenzita peaku atd. Kvantitativní parametry souvisejí se skutečnými parametry fyzické perfuze, mezi ně patří např. místní parametry související s difúzí a průměrný tranzitní čas. Nevýhodou této metody je, že nebere v potaz zpoždění a tvar křivky koncentrace činidla v lokálním arteriálním vstupu tkáně (arteriální vstupní funkce AIF). A vzhledem k tomu, že se při odhadu perfuzních parametrů mísí účinky AIF a parametry perfuze, dochází k nejedinečné interpretaci těchto parametrů. Také to vede k omezení při reprodukovatelnosti metody, protože AIF je stanovený podáváním bolusu a je závislý na specifickém cévním stromě pacienta. AIF se měří ve velké tepně zásobující analyzovanou tkáň. Funkce koncentrace tkáně se modeluje jako konvoluce AIF a tkáňové residuální funkce TRF. Odhad TRF pomocí dekonvoluce funkce koncentrace tkáně s AIF vede k absolutní kvantifikaci V_b a F_b . Měření AIF v krvi je obtížné kvůli útlumu, závislosti rychlosti krve na zpětně rozptýleném signálu a nízkému prostorovému rozlišení ultrazvukových snímků [40] [42].

3.3 Metoda Bolus & Burst

Jedná se o kombinaci Bolus Tracking and Burst Replenishment. Jako u metody Bolus Tracking se využívá jednodušší aplikace kontrastní látky pomocí bolusového podání. Stejně jako u metody burst – replenishment kvantifikuje parametry fyzické perfuze ve fázi doplňování dávek. Navíc s využitím informace z bolus tracking a burst replenishment poskytne potencionálně robustnější odhady MTT než standardní metoda bolus replenishment [42].

Nejdříve dojde k bolusovému podání kontrastní látky, k záznamu bolusové části se používají impulsy s nízkou hodnotou MI . V pozdější vymývající (wash-out) fázi bolusu, kdy koncentrace stopovače pomalu klesá, se aplikuje ultrazvukové pulzy s vysokou intenzitou, které zničí kontrastní látku v zobrazované ploše. Následuje (replenishment) doplňující fáze, která se zaznamenává pomocí zobrazovacích pulzů s nízkým MI po dobu asi 30 s. Během této fáze předpokládáme ustálený stav podobný infuzi. V konečné fázi koncentrace kontrastní látky klesá. Abychom zachovali předpoklad z fáze replenishment, kdy předpokládáme ustálený stav podobný infuzi, klesání koncentrace kontrastní látky z poslední fáze nebereme v potaz [42]. Bolusová fáze i fáze replenishment koncentrační křivky tkáně $C(t)$ je modelovaná pomocí AIF a TRF. Abychom nemuseli měřit AIF v krvi, použijeme místo toho slepou dekonvoluci. Pomocí slepé dekonvoluce lze odhadnout jak AIF, tak TRF. Metoda pracuje s předpokladem konstantní AIF ve fázi

replenishment. Koncentrační křivka $C(t)$ je modelovaná jako konvoluce AIF a TRF podle farmakokinetického modelu známého z jiných zobrazovacích modalit (PET, SPECT, MRI atd):

$$C(t) = F_b \int_0^{\infty} AIF(\tau)R(t - \tau)d\tau . \quad (3.1)$$

Kde F_b je průtok krve, $R(t)$ je bezrozměrná veličina tkáňového zbytku, tj. zbývající podíl kontrastní látky v analyzované tkáni ROI v čase t po okamžitém bolusu kontrastní látky do ROI (tj. Diracova delta funkce jako $AIF(t)$).

Pro jednokompartmentový model platí:

$$R(t) = \exp(-t/MTT), \quad (3.2)$$

kde MTT je střední doba přepravy. Jedná se o průměrnou dobu potřebnou pro průchod částice kontrastní látky v ROI. MTT můžeme vypočítat podle vztahu:

$$MTT = V_b/F_b \quad (3.3)$$

Bolusová aplikace koncentrační látky je modelovaná jako nulová počáteční podmínka v okamžiku zahájení replenishment fáze $C_{(tb)} = 0$ [41].

3.4 Farmakokinetický model

Aby bylo možné odhadnout parametry fyzické perfuze (objem krve, průtok krve a MTT) je nutné přeformulovat farmakokinetické modely obou metod. Bolusová část je modelovaná jako při perfuzní analýze jiných modalit. Jedná se například o pozitronovou emisní tomografii (PET), počítačovou tomografii (CT) a další. Tyto metody pracující s křivkou AIF jako s populační křivkou. Farmakokinetický model pro bolusovou část, kterou známe například z PET, SPECT, CT a MRI je

$$C_{bolus}(t) = F_b AIF(t) * R(t). \quad (3.4)$$

Kde $C_{bolus}(t)$ je koncentrace kontrastní látky v ROI, konvoluce je znázorněna znakem *, F_b je průtok krve na jednotku objemu tkáně (v mililitrech za minutu na mililitr tkáně), $AIF(t)$ je koncentrace kontrastní látky v arteriálním vstupu ROI a $R(t)$ je bezrozměrná veličina tkáňového zbytku, tj. zbývající podíl kontrastní látky v analyzované tkáni ROI v čase t po okamžitém bolusu kontrastní látky do ROI (tj. Diracova delta funkce jako $AIF(t)$). Funkce $R(t)$ je monotónně klesající pozitivní funkce kde $R(0) = 1$. Pro jednokompartmentový model ROI platí:

$$R(t) = e^{-(F_b/V_b) \cdot t}, \quad (3.5)$$

kde V_b je objem krve na jednotku objemu tkáně (v mililitrech na 100 ml tkáně)[42]. Nová definice farmakokinetického modelu pro replenishment fázi je založena na předpokladu, že AIF má charakter krokové funkce. To znamená, že funkce má hodnotu nula před replenishment fázi, tj. všechny mikrobubliny kontrastní látky byly zničeny. Během replenishment fáze je konstantní hodnota koncentrace, tato část reprezentuje podávání infuze. Kroková funkce AIF vede k upravené verzi rovnice:

$$C_{repl}(t) = F_b C_0 H(t) * R(t), \quad (3.6)$$

kde $C_{repl}(t)$ je koncentrace kontrastní látky v ROI, C_0 je koncentrace kontrastní látky v arteriálním vstupu ROI na plató, tj. před sekvencí vysokoenergetických pulzů. $H(t)$ je Heavisideova (jednotková) skoková funkce, funkce má nulovou hodnotu před replenishment fází a hodnotu jedna během fáze replenishment [42]. Pro jednokompartmentový model ROI, platí u replenishment fáze stejný vztah pro $R(t)$ jako u bolusové části kde:

$$R(t) = \exp(-(F_b/V_b)t). \quad (3.7)$$

Se monoexponenciální model pro replenishment fázi získá:

$$C_{repl}(t) = A(1 - e^{-\beta t}), \quad (3.8)$$

kde platí

$$A = V_b C_0, \quad (3.9)$$

$$\beta = F_b/V_b. \quad (3.10)$$

Ze vztahu můžeme odvodit, že A je úměrné objemu krve (nebo průřezové ploše krve) a $A\beta$ je úměrné průtoku krve [42]. K odhadu koncentrační křivky kontrastní látky pro ROI je zapotřebí rozdělit koncentraci stopovače na bolusovou část a replenishment část [41].

3.5 Slepá dekonvoluce

Modely odpovídající přizpůsobení bolusovým a replenishment signálům jsou konvoluční, a proto lze přizpůsobení křivky definovat jako dekonvoluci. Algoritmus slepé dekonvoluce je formulován jako minimalizace rozdílů nejmenších středních čtverců mezi naměřeným a modelovaným diskretním signálem [41]. K odhadu $R(t)$ a $AIF(t)$ nám slouží slepá vícekanálová dekonvoluce. Tato metoda používá k odhadu bolusovou i replenishment fázi, obě fáze jsou považovány za dvě nezávislé proměnné se stejným neznámým konvolučním jádrem, $R(t)$ a různými arteriálními komponentami konvoluce. Mezi apriorní informace této metody patří pozitivita a plynulost AIF [42].

Slepá vícekanálová dekonvoluce je definovaná jako optimalizační problém s následujícími rovnicemi:

$$R(p), F_b k AIF(n) = \arg\{\min(J_{bolus} + J_{repl} + J_{prior})\}, \quad (3.11)$$

$$J_{bolus} = \sum_{n=1}^N [k C_{bolus}(n) - F_b k AIF(n) * R(n)]^2, \quad (3.12)$$

$$J_{repl} = \sum_{m=1}^M [k C_{repl}(m) - F_b k C_0 H(m) * R(m)]^2, \quad (3.13)$$

$$J_{prior} = \lambda \sum_{n=1}^N [k AIF(n) * L(n)]^2. \quad (3.14)$$

Vzhledem k diskrétnímu charakteru signálu je časová proměnná t nahrazena indexy n , m a p . J_{bolus} a J_{repl} jsou členy s maximální věrohodnosti popisují věrnost měřených signálu $kC_{bolus}(n)$ a $kC_{repl}(m)$ k modelu. Třetí člen J_{prior} je parametr prioru a penalizuje vysokofrekvenční složky AIF(n). Proměnná N značí délku bolusové části signálu a proměnná M délku replenishment fáze signálu. $R(p)$ je tkáňová reziduální funkce a platí $R(p)$ pro $p = 1 \dots P$, kde P je maximum N a M . Parametr λ je váhový faktor ovlivňující plynulost výsledného odhadu AIF. $L(n)$ je Laplacianův operátor (horní propust) definován jako posloupnost $[-1, 2, -1]$, také známá jako Tikhonova regularizace. Abychom zabránili nejednoznačnému výsledku optimalizace použijeme omezení

$R(0) = 1$. TRF je definován jako:

$$R(n) = \begin{cases} 0 & \text{když } n < 0 \\ e^{-\frac{T_s}{MTT}n} & \text{když } n \geq 0, \end{cases} \quad (3.15)$$

kde T_s je vzorkovací perioda a MTT je střední doba přepravy. AIF je modelován jako suma aktuálních a zpožděných lognormálních funkcí, které představují první a druhý příchod bolusu. Matematická formulace lognormální funkce je [41][42]:

$$L(n) = \frac{k}{(n+1)T_s\sigma\sqrt{2\pi}} e^{-\frac{\{\log[(n+1)T_s]-\mu\}^2}{\sigma^2}}. \quad (3.16)$$

4. NEEVOLUČNÍ ALGORITMY

4.1 Optimalizace

U optimalizace se snažíme v rámci možností o nalezení co nejlepšího výsledku. Pomocí funkce definované obecně vztahem $f: D \Rightarrow \mathbb{R}$ na množině D prvků vhodného vektorového prostoru. Tato práce se zabývá optimalizací při hledání minima funkce $\min_{x \in R^n} f(x)$ [38].

4.2 Náhodné prohledávání

Nejjednodušší stochastický algoritmus pro hledání globálního minima. U náhodného prohledávání používáme jemné vzorkování funkce k nalezení globálního minima. U náhodného prohledávání opakovaně generujeme náhodné řešení. Pamatujeme si vždy nejlepší řešení. V každém kroku se ptáme, jestli aktuální řešení je lepší než nejlepší řešení, pokud ano, nahradíme nejlepší řešení aktuálním řešením. Největší předností této metody je její jednoduchost. Náhodným řešením může být například bod x v souvislé oblasti D .

$$D = \prod_{i=1}^d (a_i - b_i) \quad (4.1)$$

Díky dostatečně jemnému vzorkování, se nezastaví v lokálním minimu [24].

Náhodné prohledávání pracuje jak se spojitými, tak s diskrétními proměnnými. Největší nevýhodou je jeho časová náročnost, abychom snížili dobu algoritmu, můžeme zvětšit vzorkovací krok. To může mít však za příčinu přeskočení globálního minima. Algoritmus je proto vhodný zejména pro menší množství proměnných v omezeném prohledávacím prostoru. Další možností je použít ze začátku hrubé vzorkování a poté postupně volit jemnější vzorkování v místě potencionálního minima. Tento postup urychluje konvergenci, umožňuje nám pracovat s více proměnnými a snižuje pravděpodobnost překročení globálního minima [18].

4.3 Nelder-Mead simplexní algoritmus

Nelder-Mead algoritmus byl poprvé publikovaný v roce 1965. Jedná se o jednu z nejrozšířenějších metod nelineární neomezené optimalizace. Tato metoda spadá do obecné třídy metod přímého vyhledávání, k její realizaci nepotřebujeme znát derivace funkce. Pracujeme se simplexem, geometrickým obrazcem v n rozměrném prostoru. V každé iteraci vyhledávání začínáme simplexem, který se skládá z $n+1$ vrcholů a jemu odpovídajícími funkčními hodnotami. V jednotlivých iteracích se mění simplex a nové funkční hodnoty vrcholů splňují určité kritérium sestupnosti [26].

Nelder-Mead algoritmus slouží k minimalizaci reálné hodnoty $f(x)$ pro $x \in R^n$. Na začátku si určíme skalární parametry, jedná se o koeficient reflexe α , expanze β , kontrakce γ a zmenšení δ . Tyto parametry by měli splňovat určité podmínky:

$$\alpha > 0, \beta > 1, \beta > \alpha, 0 < \gamma < 1, 0 < \delta < 1 \quad (4.2)$$

Nejčastěji používané hodnoty parametrů jsou $\alpha = 1$, $\beta = 2$, $\gamma = 0,5$ a $\delta = 0,5$ [26]. Pokud uvažujeme funkce dvou proměnných, Nelder-Meadova metoda pracuje se simplexem v R^2 , který je specifikován třemi vrcholy [27]. Pokud předpokládáme, že hledáme minimum funkce $f(x)$ platí $\min_{x \in R^2} f(x)$. Označíme si vrcholy simplexu jako x_1, x_2, \dots, x_{n+1} . Nelder-Meadova metoda iterativně generuje posloupnost jednoduchých aproximací vztahu. Při každé iteraci jsou vrcholy simplexu seřazeny podle funkční hodnoty podle:

$$f(x_1) \leq f(x_2) \leq \dots \leq f(x_{n+1}). \quad (4.3)$$

Kde x_1 je nejlepší vrchol (podle funkční hodnoty $f(x_1)$) a x_{n+1} je nejhorší vrchol. Jak už jsme naznačili při volbě parametrů, algoritmus používá čtyři možné operace. Operace jsou reflexe, expanze, kontrakce a zmenšení. Určíme si m jako těžiště n nejlepších vrcholů. Pak platí následujících vztah [27]:

$$m = \frac{1}{n} \sum_{i=1}^n x_i. \quad (4.4)$$

4.3.1 Algoritmus Nelder-Mead

1. **Setřídění:** Seřadíme vrcholy podle jejich funkční hodnoty.
2. **Reflexe:** vypočítáme reflexní bod

$$x_r = m + \alpha(m - x_{n+1}). \quad (4.5)$$

Zjistíme funkční hodnotu $f_r = f(x_r)$. Pokud $f_1 \leq f_r < f_n$, nahradíme x_{n+1} hodnotou x_r .

3. **Expanze:** Pokud $f_r < f_1$ potom vypočítáme bod expanze dle vztahu:

$$x_e = m + \beta(x_r - m). \quad (4.6)$$

Vypočteme funkční hodnotu $f_e = f(x_e)$. Pokud platí, že $f_e < f_r$, nahradíme x_{n+1} hodnotou x_e , pokud ne, nahradíme x_{n+1} hodnotou x_r .

4. **Vnější kontrakce:** Pokud $f_n \leq f_r < f_{n+1}$ pak vypočítáme bod x_{OC} jako vnější kontrakci

$$x_{OC} = m + \gamma(x_r - m). \quad (4.7)$$

Pokud $f_{OC} \leq f_r$, nahradíme x_{n+1} za x_{OC} , pokud ne, jdeme na krok 6.

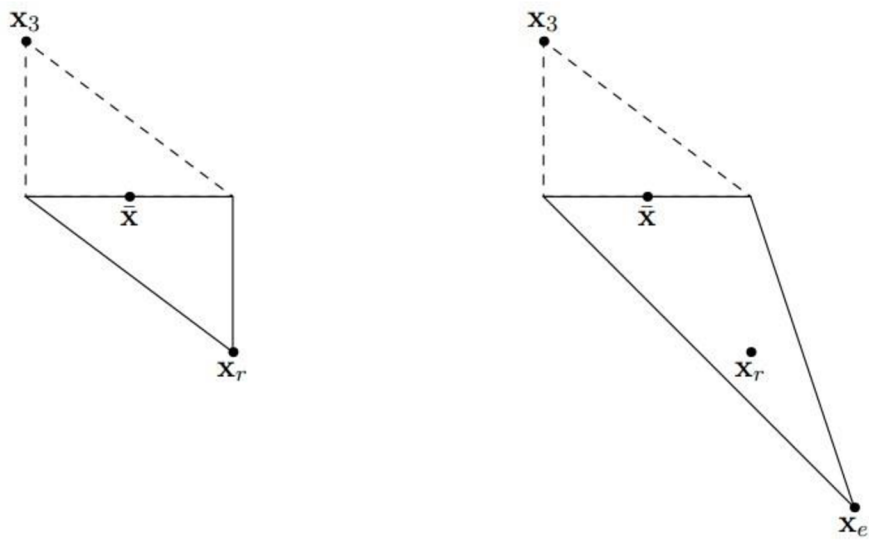
5. **Vnitřní kontrakce:** Pokud $f_r < f_{n+1}$, pak vypočítáme vnitřní kontrakci x_{iC} ,

$$x_{iC} = m - \gamma(x_r - m). \quad (4.8)$$

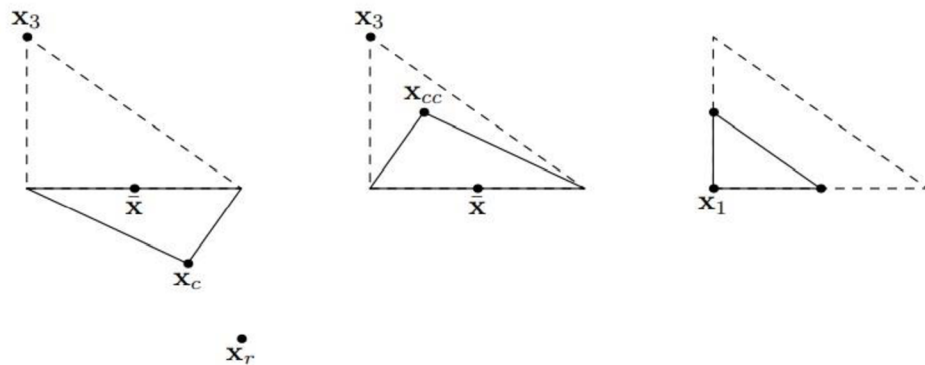
Pokud $f_{iC} < f_{n+1}$, pak nahradíme x_{n+1} za x_{iC} , pokud ne, jdu na krok 6.

6. **Zmenšení:** Pro $2 \leq i \leq n + 1$ definujeme [27]:

$$x_i = x_1 - \delta(x_i - x_1). \quad (4.9)$$



Obrázek 2: Ukázka reflexe a expanze u simplexu se třemi vrcholy. Na Obrázku vlevo je původní simplex znázorněn přerušovanou čarou, souvislou čarou je simplex po kroku reflexe. Symbol \bar{x} je těžiště a symbol x_r je bod reflexe. Obrázek vpravo znázorňuje expanzi. Bod x_e je bodem expanze. Převzato z [26].



Obrázek 3: Ukázka vnější, vnitřní kontrakce a zmenšení. Původní simplex je znázorněn přerušovanou čarou. Obrázek vlevo je simplex po vnější kontrakci, bod x_c je bodem vnější kontrakce. Obrázek uprostřed znázorňuje vnitřní kontrakci. Bod x_{cc} je bodem vnitřní kontrakce. Obrázek vpravo je ukázka zmenšení simplexu. Bod x_1 je bodem s nejlepší funkční hodnotou. Převzato z [26].

5. EVOLUČNÍ ALGORITMY

Evoluční algoritmy napodobují přirozené evoluční procesy a využívají představy evoluce živé hmoty pro optimalizaci hodnot. Simulují jednoduchý model Darwinovy evoluční teorie o vývoji populací. Populaci chápeme jako soubor jedinců, populace podléhá postupnému vývoji, jedinci spolu kooperují. Úspěšní jedinci přežívají a množí se, neúspěšní vymírají [25]. Hybnou silou změn je křížení a mutace. Evoluční algoritmus je stochastická vyhledávací technika pro řešení globálních problémů s optimalizací. Hlavní výhodou těchto algoritmů je jejich schopnost řešit velké optimalizační problémy, pokud máme funkci s více lokálními minimy, je zde menší pravděpodobnost uvíznutí v těchto bodech než u tradičních gradientních metod [14].

Evoluční algoritmus je optimalizační algoritmus, který aktualizuje řešení v každé iteraci a používá deterministická pravidla k přechodu. Nejlepší řešení se stává novým výsledkem a výše uvedený postup se několikrát opakuje [9].

Všechny evoluční algoritmy zahrnují tři hlavní kroky. V prvním kroku je inicializace, kdy dochází ke generování počáteční populace jednotlivců. V dalším kroku vypočítáme kvalitu jedinců. Ve třetím je vytvoření nové populace. Tyto kroky se opakují, dokud není splněna ukončující podmínka [10].

V každém cyklu je generována nová populace a proces skončí, když je splněno ukončující kritérium. Ukončující podmínka může být statická nebo dynamická. Mezi statické kritérium patří počet iterací. Kdy si na začátku stanovíme maximální počet iterací. Příkladem dynamického kritéria je opakování procesu, dokud výsledek nedosáhne určité optimální hodnoty. V některých příkladech se používá několik kritérií pro zastavení algoritmu [10].

Stanovení vhodné velikosti populace a maximálního počtu řešení má velký vliv na kvalitu řešení a časovou náročnost. V praxi jsou tyto hodnoty téměř vždy určovány empiricky pomocí pilotního běhu [10].

5.1 Genetický algoritmus

Jedince reprezentuje binární řetězec konstantní délky [29]. V genetickém algoritmu nazýváme jednotlivá řešení (jedince) chromozomy. Chromozom je vektor parametrů, jednotlivé parametry nazýváme geny a hodnoty parametrů alely [37]. Každému chromozomu je přiřazena určitá hodnota kvality. Genetický algoritmus používá tři hlavní operátory: selekce, křížení a mutace. Při selekci jsou chromozomy s lepší kvalitou vybírány s větší pravděpodobností než chromozomy s nižší kvalitou. Křížení kombinuje

chromozomy rodičů a vytváří nové potomky. Vzhledem k tomu, že při křížení jsou vybírány silnější (kvalitnější) jedinci častěji, může docházet k tomu, že nové generace se začínou velmi podobat. Rozmanitost populace může klesnout, a to by mohlo vést ke stagnaci populace. Proto jako další operátor používáme mutaci, která přináší do nové generace rozmanitost a zabraňuje stagnaci [10].

5.1.1 Binární reprezentace populace

Aplikace skutečných hodnot parametrů způsobuje u genetických algoritmů problémy, proto dochází ke kódování hodnot parametrů. Nejčastěji se používá binární reprezentace dat, a to standardní binární kódování a Grayovo kódování. Hodnota každého parametru je zakódována do binárního řetězce. Jednotlivé binární řetězce se zřetězí a vytvoří jeden společný řetězec (chromozom), který představuje celý vektor parametrů. Bitová pozice odpovídá genu a hodnota bitu alele. Vektor parametrů označíme jako x a jemu odpovídající bitový řetězec jako X . Dána reprezentace je tedy převedena na kombinatorický problém, kdy body prohledávacího prostoru jsou vrcholy vícerozměrné krychle. Grayův kód má tu vlastnost, že při zvýšení hodnoty parametru o jeden krok odpovídá změně jednoho bitu v grayovu kódu [37].

5.1.2 Spojitá reprezentace populace

Spojitou reprezentaci dat používáme, pokud pracujeme se spojitými hodnotami proměnných a potřebujeme znát jejich velmi přesné hodnoty. Pokud bychom použili binární kódování, potřebovali bychom mnoho bitů k jejich reprezentaci. U spojitých hodnot používáme k reprezentaci hodnot pohyblivou řádovou čárkou. Tato reprezentace dat je méně náročná na paměť algoritmu než bitová reprezentace, protože hodnoty proměnných jsou uloženy jako jedna hodnota na rozdíl od bitové reprezentace hodnot, kdy proměnou představuje více hodnot, binární řetězec. Spojitá reprezentace je také rychlejší, protože nedochází k dekodování hodnot [18].

5.1.3 Selektce

Selektci vybíráme jedince, které zkřížíme. Mezi nejoblíbenější metody selektce jedinců patří *turnaj* a *ruleta* [10]. Při ruletě přiřadíme každému jednotlivci pravděpodobnost výběru P_j na základě jeho kvality. Pravděpodobnost každého jedince je dána vztahem:

$$P_i = \frac{F_i}{\sum_{j=1}^N F_j}, \quad (5.1)$$

kde F_i je individuální kvalita, N je velikost populace. Kvalita jedince je tedy dána jeho kvalitou dělenou sumou kvalit všech jedinců. Vytvoříme hodnoty kumulativní pravděpodobnosti:

$$C_i = \sum_{j=1}^i P_j. \quad (5.2)$$

Vygenerujeme náhodné číslo U z rozsahu $(0,1)$ a hledáme do kterého intervalu kumulativní pravděpodobnosti patří, tedy

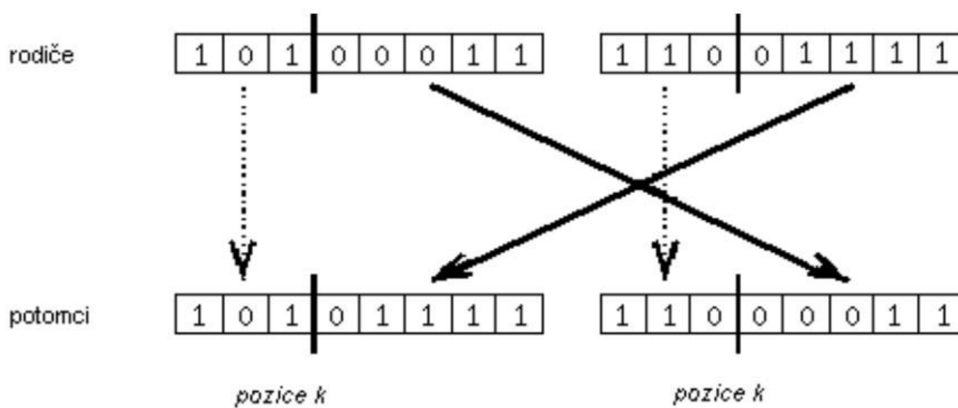
$$C_{i-1} < U \leq C_i. \quad (5.3)$$

Do výstupní populace vybereme jedince, kterému náleží hodnota C_i [12].

U turnajové selekce je vybraná určitá skupina jedinců, kdy do další generace postupují jen ti nejlepší z hlediska kvality. Úpravou parametrů, které charakterizují velikost vybrané skupinky a počet nejlepších jedinců, které pošleme do další generace, můžeme vykonávat určitou kontrolu nad velikostí selekčního tlaku a tím měnit rychlost konvergence. Pokud vybereme nejmenší velikost turnaje, kdy vybíráme pouze dva jedince, dochází k nejmenší konvergenci [33]. Výhodou turnajové selekce je, že nemusíme třídít celou populaci. U velkých populací se třídění stává velmi časově náročné. Proto je turnajová selekce vhodná pro velké populace [18].

5.1.4 Křížení

Křížení je nejvýznamnějším operátorem GA. Křížením vznikají noví jedinci (potomci) rekombinací rodičovského genetického materiálu. Počet rodičů při křížení jsou obvykle dva. Dva jedinci vybraní selekcí vytvoří dva potomky, kteří jdou do další generace. Jedním typem křížení je křížení v jednom bodě. Kdy si stanovíme náhodně dělicí bod v chromozomu např. g , nový dva jedinci jsou vytvořeni z původních dvou chromozomů tak, že první potomek je tvořen geny na pozici 1 až g prvního rodiče a geny $g+1$ druhého rodiče. Druhý potomek to má naopak, tedy z prvního rodiče má geny $g+1$ a z druhého 1 až g . Do nové generace postupují zcela nové chromozomy (potomci) [15].



Obrázek 4: ukázka křížení v jednom náhodně zvoleném bodě. Převzato z [15].

5.1.5 Reprodukce

Reprodukce je křížení, které se uplatňuje převážně u spojitého genetického algoritmu. Existuje několik realizací reprodukce, v této kapitole jsou některé metody pro vytvoření dvou či tří potomků dvou rodičů [18].

U této metody křížení (reprodukce) si označíme prvního rodiče jako O_1 a druhého jako O_2 , c je náhodné číslo z intervalu $(0,1)$, p_c je pravděpodobnost křížení, O'_1 a O'_2 jsou potomci. Pokud $c \geq p_c$, a kvalita O_1 je menší než kvalita O_2 pak platí, že

$$O'_1 = O_1 + r(O_1 - O_2) \quad (5.4)$$

$$O'_2 = O_2 + r(O_1 - O_2). \quad (5.5)$$

Pokud je kvalita O_2 menší než kvalita O_1 pak platí

$$O'_1 = O_1 + r(O_2 - O_1) \quad (5.6)$$

$$O'_2 = O_2 + r(O_2 - O_1). \quad (5.7)$$

Kde r je náhodná hodnota z intervalu $(0,1)$ určující stupeň křížení. Pokud je $c < p_c$, pak ke křížení nedochází [36].

Metodu můžeme rozšířit o přidání dalšího rodiče. Kvalita rodiče O_1 je nejmenší kvalita ze tří vybraných jedinců. Tedy kvalita jedince O_1 je menší než kvalita jedince O_2 a O_3 . Rovnice pro výpočet potomků jsou následující:

$$O'_1 = O_1 + r(2O_1 - O_2 - O_3), \quad (5.8)$$

$$O'_2 = O_2 + r(2O_1 - O_2 - O_3), \quad (5.9)$$

$$O'_3 = O_3 + r(2O_1 - O_2 - O_3). \quad (5.10)$$

Ostatní postup je stejný jako u křížení dvou jedinců [36].

Další metodou je metoda blending (míchání). Hodnoty potomků jsou v rozsahu mezi hodnotami obou rodičů. Kde platí následující rovnice:

$$O'_1 = r \cdot O_1 + O_2 \cdot (1 - r), \quad (5.11)$$

$$O'_2 = (1 - r) \cdot O_1 + O_2 \cdot r. \quad (5.12)$$

Další metodou pro vytvoření tří potomků je metoda nazývaná Extrapoláční metoda. U této metody se můžeme dostat za rozsah hodnot mezi hodnotami obou rodičů [18]:

$$O'_1 = 0,5 \cdot O_1 + O_2 \cdot 0,5, \quad (5.13)$$

$$O'_2 = 1,5 \cdot O_1 - O_2 \cdot 0,5, \quad (5.14)$$

$$O'_3 = -0,5 \cdot O_1 + O_2 \cdot 0,5. \quad (5.15)$$

5.1.6 Mutace

Mutací nazýváme náhodnou změnu jedné nebo více hodnot proměnných (prvků chromozomu), která ovlivní řešení ať už kladně nebo záporně. S jakou pravděpodobností dojde ke změně, nám určuje pravděpodobnost mutace. Pravděpodobnost mutace si zvolíme na počátku cyklu. Mutace nám umožňuje zamezit přílišné specializaci. Mutace diverzifikuje vyhledávací směry a vyhybá se konvergencím k místním optimům. Zavádí

do chromozomů nové informace. Pokud bychom použili u GA pouze mutaci, výsledek by se nelišil od metody náhodného prohledávání [25][34][35]. U jednobodové mutace se mění 1 na 0 a naopak. Body mutace jsou vybrány náhodně z $N_{pop} \times N_{bity}$, tedy z celkového počtu bitů z populační matice. K mutaci nedochází v poslední iteraci. Většinou nedochází k mutaci u jedinců s nejlepším řešením. Taková řešení označujeme jako elitní řešení. Pokud elitářství je rovno jedné, počet mutací je dán vztahem:

$$počet\ mutací = \left(\frac{\mu}{100}\right) \times (N_{pop} - 1) \times N_{bity}, \quad (5.16)$$

kde μ je procento mutovaných bitů [18].

5.1.7 Základní algoritmus jednoduchého GA

1. Definujeme základní problém, který má řešit GA.
2. Vytvoříme počáteční populaci N jedinců.
3. Vytvoříme vhodnou fitness funkci (funkci kvality) pro získání kvality každého jedince.
4. Provedeme selekci jedinců a vytvoříme potomky pomocí křížení.
5. Provedeme náhodnou mutaci.
6. Vypočítáme kvalitu každého jedince.
7. Vybereme N nejlepších jedinců podle kvality.
8. Pokud není splněno ukončující kritérium vracíme se ke kroku 4, pokud je splněno, ukončíme cyklus [25] [34].

6. ROJOVÉ ALGORITMY

Rojové algoritmy patří mezi populační algoritmy inspirované přírodou. Rojové algoritmy jsou inspirované kolektivním chováním zvířat jako jsou ptáci, ryby, mravenci, včely, termiti. Metoda se zabývá multi-agentově distribuovanou systémovou inteligencí. Roj je populace homogenních jednoduchých agentů, kteří vykonávají spíše elementární úkoly a interagují lokálně mezi sebou a svým prostředím bez centrální kontroly. Kolektivní chování se objevuje jako důsledek sebeorganizace a místních, částečně stochastických interakcí. I když jsou tyto rojové jedinci relativně jednodušší a s limitujícími vlastními schopnostmi, je jejich schopnost dosáhnout optimálního výsledku uzpůsobena jejich behaviorálním chováním a vzájemnými interakcemi prostřednictvím sdílení informací. Díky tomu jsou populační algoritmy schopné poskytnout robustní řešení komplexních úloh v relativně krátké době. Mezi nepoužívanější rojové algoritmy patří optimalizace kolonií mravenců (ACO) a optimalizace rojem částic. Mezi další algoritmy patřící do této skupiny patří například metoda SOMA, včelí algoritmus, hledání kukaček, netopýří algoritmus a algoritmus světlušek [16] [17].

6.1 Optimalizace kolonií mravenců (ACO)

Pravděpodobnostní optimalizační metoda, která se uplatňuje v metodách, kde lze úkol vyjádřit jako hledání nejlepší cesty podél grafu. Inspirací u mravenců je jejich putování od své kolonie ke zdroji potravy a zpět. Umělý mravec iterativně vytváří řešení daného problému přechodem z kandidátského „státu“ do jiného „státu“. Postup jednotlivých mravenců je ovlivněn kombinací dvou faktorů. Prvním faktorem je atraktivita tahu, což je metaheuristický parametr apriorní vhodnosti přechodu, obvykle nepřímo souvisí se vzdáleností k cílovému bodu. Druhým faktorem je „feromonová cesta“, která nepřímo zajišťuje sociální interakci mezi agenty. Jedná se o analogii chování skutečných mravenců, kteří při svém putování při hledání potravy vylučují feromony. Členové kolonie mravenců budou volit cesty s vyšší pravděpodobností nalezení optima, to znamená cesty s výraznou koncentrací feromonů [17].

6.2 Optimalizace rojem částic (PSO)

Stochastická evoluční technika založená na pohybu a inteligenci rojů. Počátek ideje byl nalézt algoritmus využívající jednoduché analogie sociálních chování zvířat. Příkladem jsou například hejna ptáků a ryby. Zabývá se kolováním informací a sociálním chováním pozorovaných u zvířat. Algoritmus více využívá analogii sociálních interakcí zvířat, nezabývá se natolik individuálních kognitivních schopností zvířat. Pokud využijeme tuto analogii u PSO, můžeme si hlavní princip metody představit jako hejno ptáků letící nad naší oblastí zájmu, každý pták letí určitou rychlostí, která je dynamicky upravována podle vlastních zkušeností s létáním a letové zkušenosti jeho společníků [16].

Algoritmus patří mezi globální optimalizační algoritmy a je zvláště vhodný pro řešení úloh kde je optimální řešení bod ve vícerozměrném parametrickém prostoru. V PSO je soubor kandidátských řešení optimalizačního problému definován jako roj částic, které mohou proudit parametrickým (vyhledávacím) prostorem definující trajektorie, které jsou poháněny jejich vlastními a sousedními nejlepšími výkony. Částice v PSO, které se pohybují ve vyhledávacím prostoru se vyznačují nejen svou pozicí, ale také rychlostí [17].

V PSO je řada jednoduchých entit – *částic* umístěných ve vyhledávacím prostoru. *Částic* je jeden z kandidátů na správné řešení a představuje bod v D – *dimenzionálním* prostoru. Parametr D udává počet proměnných, které mají být optimalizovány [17]. Každá částice ohodnotí svou pozici a určí svůj pohyb na základě kombinací určitých aspektů, mezi tyto aspekty patří: nejlepší lokace dané *částice*, její aktuální pozice, nejlepší pozice jednoho nebo více členů roje, náhodný vychylující parametr [13]. Každá *částice* je nositelkou dvou hlavních vlastností: polohou a rychlostí, částice se na novou pozici přesune určitou rychlostí. Jakmile se částice přesunou, dojde k aktualizaci nejlepší pozice každé částice a nejlepší polohy roje. Rychlost každé *částice* je po té aktualizovaná na základě zkušeností s ní. Proces opakujeme, dokud nejsou splněny ukončující podmínky [10]. *Částice* samotné nemají skoro téměř žádnou pravomoc vyřešit optimalizační problém, k pokroku dochází pouze když *částice* interagují. Jedná se o tzv. *populační fenomén*, podstatou tohoto fenoménu je individuální chování *částic* prostřednictvím jejich interakcí. Populace jsou organizované na základě určité komunikační struktury nebo topologie, nazývané sociální síť. Nejčastěji se topologie skládá z obousměrných hran spojující páry (*částice*). Každá *částice* komunikuje s ostatními *částicemi* a ovlivněna nejlepším bodem nalezeným kterýmkoliv členem jejího topologického sousedství [13].

6.2.1 Porovnání PSO a GA

Algoritmus PSO stejně jako u GA začíná s populační maticí náhodných čísel, řádky matice jsou nazývané částice (u GA jsou chromozomy) a populace se nazývá roj částic. U PSO není binární kódování řádků (*částic*) matice jako u binárního genetického algoritmu. Na rozdíl od GA nemá PSO evoluční operátory jako je křížení a mutace [18]. U PSO nedochází k procesu selekce, všechny částice PSO jsou během běhu PSO zachovány, pracujeme stále se stejnou populací. Běh u PSO je definován jako celkový počet generací, které proběhnou, než dojde k ukončení procesu. Algoritmus nepracuje s myšlenkou přežití nejschopnějších jedinců. V každé generaci částice mohou nabývat pouze omezených hodnot, tzv. mohou létat pouze v omezeném počtu směrů, kde se předpokládá, že budou dobrými oblastmi k letu podle vlastní zkušenosti a zkušenosti roje. Zatímco u GA je díky mutaci možno dosažení i jiných hodnot částic, mohou „létat“ v různých směrech [16]. U algoritmu PSO na rozdíl od GA je velkou výhodou, že nepožaduje seřazení kvalit jedinců po každé generaci. To je významná výpočetní výhoda,

a to především když pracujeme s velkou populací. Aktualizace rychlosti a polohy v PSO vyžaduje pouze jednoduché aritmetické operace s reálnými čísly [10].

6.2.2 Inicializace PSO

Podobně jako u GA je prvním procesem PSO inicializace počáteční populace, v našem případě roje částic. Pokud zvolíme větší populaci, dojde ke zvyšování rozmanitosti roje a jeho průzkumné schopnosti. Na druhé straně se zvýší výpočetní náročnost a může se zvýšit pravděpodobnost předčasné konvergence. Ve většině případů bylo prokázáno, že pokud je počet jedinců vyšší než 50, dojde k tomu, že při dalším zvyšování počtu jedinců se snižuje citlivost PSO na změnu velikosti populace [17]. Každá částice je inicializovaná náhodnou pozicí a rychlostí. Aby došlo k vylepšení konvergenčních vlastností je vhodné na začátku definovat hranice vyhledávacího prostoru podle vztahu:

$$x_{id}(0) \sim U(x_{d,\min}, x_{d,\max}), \quad (6.1)$$

kde $x_{d,\min}$ a $x_{d,\max}$ jsou dolní a horní hranice d -té dimenze vyhledávacího prostoru a U označuje uniformní rozdělení. Tento postup zajišťuje dobré počáteční pokrytí parametrického prostoru a podporuje průzkumnou schopnost algoritmu, což vede k relativně vysoké rychlosti konvergence [10].

Každá částice je ohodnocena tzv. kvalitou (fitness hodnota). Po každém výpočtu kvality se porovná předchozí nejlepší kvalita částice a předchozí nejlepší kvalita roje a podle potřeby dojde k aktualizaci lokální a globální nejlepší pozice. Pokud není splněna ukončující podmínka, dojde k aktualizaci rychlosti a polohy, a tedy vytvoření nového roje. Lokální, globální nejlepší pozice a stará rychlost se použije při výpočtu nové rychlosti. Dvě klíčové operace algoritmu PSO patří aktualizace rychlosti a polohy. Rychlost je aktualizovaná na základě tří složek: stará rychlost (setrvačnost nebo hybnost), zkušenosti ostatních částic (kognitivní, samoučení) a zkušenosti celého roje (sociální učení roje). S každou složkou je spojena váhová konstanta. U základního algoritmu PSO je počet požadovaných konstant tři. [10].

U PSO je každé kandidátské řešení nazývané „částice“ a představuje bod v D -dimenzionálním prostoru. Každá částice se pohybuje v D dimenzionálním prostoru určitou rychlostí. Jedinec si nese informaci tří D dimenzionálního vektorů, jedná se o aktuální pozici x_i , předchozí nejlepší polohu p_i , a rychlost v_i . Aktuální poloha x_i je definovaná sadou souřadnic popisující bod v prostoru. Při každé iteraci algoritmu je aktuální poloha ohodnocena, pokud je tato pozice lepší než tak, která byla dosud nalezena, tak se souřadnice dané polohy uloží do vektoru p_i . Hodnoty dosud nejlepších výsledků funkce (všechny p_i) jsou uloženy v proměnné $pbest_i$, pro srovnání v pozdějších iteracích [17].

Pokud D je počet parametrů, které mají být optimalizovány je poloha i -té částice reprezentovaná vektorem $X_i = (x_{i1}, x_{i2}, \dots, x_{iD})$. A populaci N kandidátských řešení tvoří roj $X = \{x_1, x_2, \dots, x_N\}$. Kde D je dimenze prostoru. Při hledání optimálního řešení problému definují částice trajektorie v parametrickém prostoru (tj. iteračně aktualizují své polohy) na základě následující pohybové rovnice [17]:

$$x_{i,d}(t+1) = x_{i,d}(t) + v_{i,d}(t+1), \quad (6.2)$$

kde t a $t+1$ označuje dvě po sobě jdoucí iterace algoritmu. Rychlost změny polohy částice, rychlost, je pro částici i vyjádřena vektorem $V_i = (v_{i1}, v_{i2}, \dots, v_{iD})$. Rychlostní vektory určují způsob pohybu částic v prohledávacím prostoru jsou tvořeny příspěvkem tří složek. První složka je definovaná setrvačností nebo hybností, slouží k zamezení význačné změně směru sledováním předchozího směru toku [16]. Druhý člen se nazývá kognitivní složka odpovídá tendenci částice vracet se do svých dříve nalezených nejlepších pozic. Poslední složka se nazývá sociální, zastupuje tendenci částice pohybovat se směrem k nejlepší poloze celého roje. Na základě těchto úvah je rychlost i -té částice definovaná jako:

$$v_{i,d}(t+1) = \omega \cdot v_{i,d}(t) + c_1 \cdot r_1 \cdot (p_{i,d} - x_{i,d}(t)) + c_2 \cdot r_2 \cdot (g_{i,d} - x_{i,d}). \quad (6.3)$$

Nejlepší předchozí pozice i -té částice (polohy, které dali nejlepší kvalitu) jsou reprezentované vektorem $P_i = (p_{i1}, p_{i2}, \dots, p_{iD})$. Zatímco $G_i = (g_{i1}, g_{i2}, \dots, g_{iD})$ je globální nejlepší řešení, tj. celkové nejlepší řešení získané rojem. Částice si aktualizují svou polohu a rychlost na základě nejlepší místního a globálního řešení [18]. Parametr $V_i = (v_{i1}, v_{i2}, \dots, v_{iD})$ je rychlost částice a $X_i = (x_{i1}, x_{i2}, \dots, x_{iD})$ je poloha částice v d dimenzi. Parametry c_1 a c_2 jsou konstanty zrychlení, jedná se o konstanty nabývající hodnot kladných reálných čísel. Konstanta c_1 se nazývá kognitivní koeficient a konstanta c_2 je sociální koeficient. Tyto konstanty regulují velikost kroků, které částice podnikne ve směru svého osobního maxima, respektive globálního maxima. Konstanty nabývají reálných hodnot obvykle z rozsahu $0 \leq c_1, c_2 \leq 4$. Obecně se ukázalo, že nastavení hodnoty na hodnotu 2 u obou koeficientů funguje dobře pro většinu aplikací [16][18]

Na druhé straně parametry r_1 a r_2 jsou nezávislá náhodná čísla z intervalu $[0,1]$, takže jak sociální, tak kognitivní složka má stochastický vliv na aktualizaci rychlosti podle rovnice (6.3). Vzhledem k tomu, že trajektorie částic se skládá z příspěvku systematické přitažlivosti k lokálnímu a globálnímu nejlepšímu řešení a stochastické váhy těchto obou složek, je trajektorie nakreslená částicemi částečně náhodné povahy. Parametr ω se nazývá interní váha a v rovnici řídí hybnost částice modulováním příspěvků rychlosti předchozí iterace. Velká hodnota parametru ω ($\omega > 1$) nám umožní globální vyhledávání, zatímco malá hodnota parametru ω ($\omega < 1$) lokální vyhledávání. Parametr ω tedy slouží k vyvažování mezi globálním a lokálním prohledáváním. U základního algoritmu PSO se využívala konstantní hodnota parametru ω [16] [18]:

$$\omega(t) = \omega = konst \quad (6.4)$$

Později se však ukázalo, že dynamická úprava její hodnoty výrazně zlepši konvergenční vlastnosti algoritmu. Lineárním snížení ω z relativně vysoké hodnoty na malou v průběhu běhu algoritmu PSO, dojde k větší schopnosti globálního vyhledávání v začátcích běhu algoritmu a ke konci k lepšímu lokálnímu vyhledávání. Rovnice lineárně klesající setrvačné hmotnosti je:

$$\omega(t) = \omega_{max} - \frac{\omega_{max} - \omega_{min}}{t_{max}} \cdot t, \quad (6.5)$$

kde ω_{max} je maximální a ω_{min} je minimální hodnota interní váhy. Parametr t je aktuální hodnota iterace a t_{max} je maximální hodnota iterací algoritmu. Ze zkušeností více autorů je vhodné zvolit k dosažení nejlepšího výkonu $\omega_{max} = 0,9$ a $\omega_{min} = 0,4$. V tabulce je ukázka dalších strategií dynamické změny interní váhy. Náhodná setrvačná váha je vhodnější u algoritmů, kde je požadována rychlejší konvergence. Zatímco za požití strategie chaotické náhodné setrvačné váhy při snižování inertní váhy je nejkonzistentnější. Metoda konstantní lineární snižování setrvačné interní váhy vede k celkově k nejnižším chybám.

nejlepší globální – lokální inertní váha	$\omega_{i,d}(t) = 1,1 - \frac{p_{i,d}(t)}{g_{i,d}(t)}$
Chaotické sestupná inertní váha	$\omega(t) = (\omega(0) - \omega(t_{max})) \left(\frac{t_{max}-t}{t_{max}} \right) + \omega(t_{max})z$ $z = 4r(1 - r) \quad r \sim U(0,1)$
Náhodná setrvačná váha	$\omega(t) = 0,5 + \frac{r}{2} \quad r \sim U(0,1)$
Chaotická náhodná setrvačná hmotnost	$\omega(t) = 0,5r_1 + 0,5z$ $z = 4r_2(1 - r_2) \quad r_1 r_2 \sim U(0,1)$

Tabulka 1: strategie dynamické změny interní váhy ω

Algoritmus PSO aktualizuje rychlostní vektor pro každou částici. Iterační proces popsany oběma rovnicemi se opakuje až do splnění ukončujících podmínek. Mezi tyto podmínky patří například předem stanovený počet iterací, maximální počet iterací od poslední aktualizace globálního nejlepšího řešení nebo předefinovaná cílová minimální hodnota kvality [16] [18].

6.3 Optimalizace SOMA

Metoda Soma (self-organizing migration) patří mezi populační algoritmy patřící do skupiny rojových algoritmů. SOMA je založena na samoorganizujícím se chováním skupin jednotlivců v „sociálním prostředí“, například stáda zvířat hledající potravu nebo smečka vlků řízena vůdcem. SOMU můžeme klasifikovat jako evoluční algoritmus, a to i navzdory skutečnosti, že během hledání nejsou vytvářeny žádné nové generace jednotlivců na rozdíl například u GA. V algoritmu dochází pouze ke změně polohy jednotlivců ve vyhledávacím prostoru během tzv. *migrační smyčky (ML)*. Soma je založena na kooperativním hledání (migraci) v prostoru parametrů [19]. Vzhledem k tomu, že se nevytvářejí žádní noví jedinci a v D – *dimenzionální* hyperrovině se pohybují pouze existující jedinci je tento algoritmus nazýván samoorganizující migrační algoritmus, zkráceně SOMA [21]. Velikost D – *dimenze* se rovná počtu parametrů. Hlavním principem algoritmu SOMA je migrace jednotlivců v populaci za vůdcem [40]. Můžeme si algoritmus SOMA představit jako jednoduchý model lovcí smečky. Jedinci smečky se pohybují (migrují) k vůdci, ohodnotí polohu a zapamatují si nejlepší pozici. Jedinci se pohybují po D *rozměrné* území, kdy se jedinci nemusí pohybovat ve všech *dimenzi prostoru* D , ale pouze v některých dimenzích [24]. Během *migrační smyčky ML* jednotlivci soutěží proti sobě tím, že se snaží najít nejlepší pozici, zatímco později spolupracují výměnou informací. Jedinec s nejlepší pozicí se stává novým vůdcem. Toto soutěživé kooperativní chování je typické pro některé druhy zvířat (např. vlci) a je reprodukován memetickými algoritmy [19].

6.3.1 Strategie migrace SOMA

Stejně jako ostatní evoluční a rojové algoritmy pracuje SOMA s populací P_x^{ML} jednotlivců x na migračním *cyklu ML*. Populace se skládá z daného počtu jednotlivců, značíme jako *velikost populace*. Populace je reprezentovaná D – *rozměrným* vektorem parametrů s reálnou hodnotou x_i^{ML} . Kde ML označuje smyčku algoritmu. Pro každý parametr jsou určeny maximální a minimální hodnoty, které vyjadřují horní hranici $x_j^{(hi)}$ a dolní hranici $x_j^{(lo)}$ vyhledávání. Počáteční populace se skládá z vektorů náhodných čísel v předepsaném rozsahu. Algoritmus SOMA má několik základních strategií migrace. Jednotlivé názvy označují počet a charakter vůdců [40].

Algoritmus SOMA má několik základních strategií migrace:

- All to one – základní verze, vůdcem je nejlepší jedinec v populaci. V každém migračním cyklu všichni jedinci putují za vůdcem.
- All to all – není vůdce, jednotlivci se v každém migračním cyklu postupně stávají vůdci, ostatní jedinci putují postupně za nimi. Nevýhodou této metody je, že dochází k mnoha nadbytečným pohybům. Kdy se jedinci pohybují mezi špatnými

jedinci nebo od dobrého ke špatnému jedinci. To vede časové náročnosti algoritmu [22].

- All to all adaptive – adaptivní varianta All to all [24].
- All to one random – vůdce je vybrán náhodně. Může být i více vůdců. Algoritmus si může vybrat pouze jednoho nebo více náhodně vybraných vůdců. Počet náhodně vybraných vůdců může být neměnné, tedy ve fázi inicializace určíme počet vůdců, nebo se může náhodně vybrat v každém migračním cyklu (adaptivní varianta) [19][40].

V neadaptivních variantách jsou jednotlivci přesunuti na lepší pozice na konci migračního kola ML. V adaptivních variantách dochází k přesunu jednotlivce okamžitě [40].

6.3.2 Inicializace algoritmu a parametry SOMA

SOMA začíná generováním populace obsahující daný počet jednotlivců, z nichž každý je kandidátským řešením úlohy. Jedinci jsou náhodně generováni v celém vyhledávacím prostoru podle rovnice:

$$P = x_j^{(lo)} + r(x_j^{(hi)} - x_j^{(lo)}), \quad (6.6)$$

kde P je počáteční populace algoritmu, $x_j^{(lo)}$ je minimální mezní hodnota a $x_j^{(hi)}$ je maximální mezní hodnota. Kde r je náhodné číslo z rozsahu $(0,1)$. Během inicializace určíme i hodnotu délky cesty (PL) a kroku. Tyto parametry jsou neměnné během migrační smyčky ML. *Délka cesty* nám určuje, v jaké vzdálenosti od vůdce se zastaví jedinec během průzkumu. Abychom zabránili uvíznutí jedince x_i^{ML} v lokálním minimu, musí být délka cesty větší než 1. Například pokud se *délka cesty* = 1, to znamená, že jedinci se zastaví na pozici vůdce. Pokud *délka cesty* = 2 to znamená, že jedinci předběhnou vůdce o $1 \cdot$ *délku cesty*. Optimální rozsah délky cesty je $[1,1;3]$. Doporučená hodnota je 3. Hodnoty menší než 1 vedou často k předčasné konvergenci [19][22][40].

Každý jedinec migruje za vůdce o určité velikosti kroků. Aby se zabránilo předčasné konvergenci, neměla by být délka cesty plně dělitelná krokem. Parametr počet kroků (ns) je počet pozic zkoumaných každou částicí v každé migrační smyčce. Nižší hodnoty ns prohledávají důkladněji prostor parametrů za cenu zpomalení algoritmu. Optimální rozsah počtu kroků je $[5;10]$, doporučená hodnota je 6 [19].

Po inicializaci dojde k migrační smyčce. V každé migrační smyčce je vybrán jedinec s nejlepší hodnotou kvality a stane se vůdcem. Ostatní jednotlivci cestují nezávisle a každý z nich se bude krok za krokem pohybovat směrem k vůdci, dokud nebude vzdálenost jeho aktuální polohy od počáteční polohy větší než *délka cesty*. Rovnice definující přesun jedinců je definována takto:

$$x_{i,d}^{ML+1} = x_{i,d}^{ML} + (x_{L,d}^{ML} - x_{i,d}^{ML}) \cdot t \cdot PRTVector_i. \quad (6.7)$$

Kde $x_{i,d}^{ML+1}$ je nová pozice v další migrační smyčce, $x_{i,d}^{ML}$ je pozice v aktuální migrační smyčce, $x_{i,d}^{ML}$ je pozice vůdce v současné migrační smyčce, t je skokový krok, tedy od jeho pozice s velikostí kroku do *délky cesty*. Mezi další parametr v SOMA patří vyhledávací vektor SV, který spojuje počáteční pozici jednotlivce s vůdcem. [22] [40]. Na konci migračního cyklu se jedinec přesune na nejlepší kandidátskou pozici. Kde $ML + 1$ označuje následující migrační cyklus. Proměnná t se zvyšuje na základě počtu kroků. V prvním cyklu je $t = krok$, ve druhém kroku $t = 2 \cdot krok$ atd. Parametr t se zvyšuje, když platí, že $t < délka cesty$. Pokud jsou všechny hodnoty vektoru $\overrightarrow{PRTVector}_i$ rovny nule, jedinec se nepohybuje. Pokud je hodnota kroku malá, bude počet kroků v jednom cyklu větší. Vícerozměrný prostor bude prohledáván podrobněji. Doporučuje se volit velikost parametru krok, tak aby $1/krok$ nebylo celé číslo, tak aby jedinec na své cestě za vůdcem nenavštívil místo obsazené vůdcem [24] [40].

Dalším parametrem je perturbace PRT (vychýlení), kdy je směr pohybu jedince \vec{x}_i^{ML} za vůdcem \vec{x}_L^{ML} dán perturbačním (vychylujícím) vektorem $\overrightarrow{PRTVector}_i$. Velikost během inicializace zvolíme z rozsahu $[0;1]$, pokud nastavíme $PRT=1$ vznikne nám deterministický algoritmus vhodný pro lokální prohledávání. Optimální rozsah je $[0,1;0,7]$, doporučená hodnota je 0,4. Aby jednotlivci prozkoumávali (skákali) v d -dimenzionální prostoru namísto k posunu (skoku) přímo k vůdci, je vytvořen vektor náhodných čísel $rnd_j \in [0,1]$ a porovnáván s PRT prahem. Pokud je menší než PRT, je $\overrightarrow{PRTVector}_i$ tohoto skoku nastaven na 1, pokud je to naopak, je nastaven na 0 [22] [40]. Pokud zvolíme menší hodnotu PRT , dojde k pomalejší konvergenci algoritmu. Vyšší hodnota PRT urychluje konvergenci algoritmu, nicméně může to mít za následek ztrátu rozmanitosti v několika migračních cyklech ML. Na začátku migračního kola pro každého jednotlivce \vec{x}_i^{ML} je $\overrightarrow{PRTVector}_i$ generován následovně: pro každý j -tý parametr je generované náhodné číslo $rnd_j \in [0,1]$ s uniformním rozdělením. Pokud je hodnota rnd_j menší než hodnota PRT, hodnota j -tého parametru v $\overrightarrow{PRTVector}_i$ bude nastavena na hodnotu jedna. V opačném případě bude nastavena na nulu. Tento proces lze formulovat takto [40][19]:

$$PRTVector_{i,j} = \begin{cases} 1 & \text{když } rnd_j < PRT \\ 0 & \text{jinak} \end{cases} \quad (6.8)$$

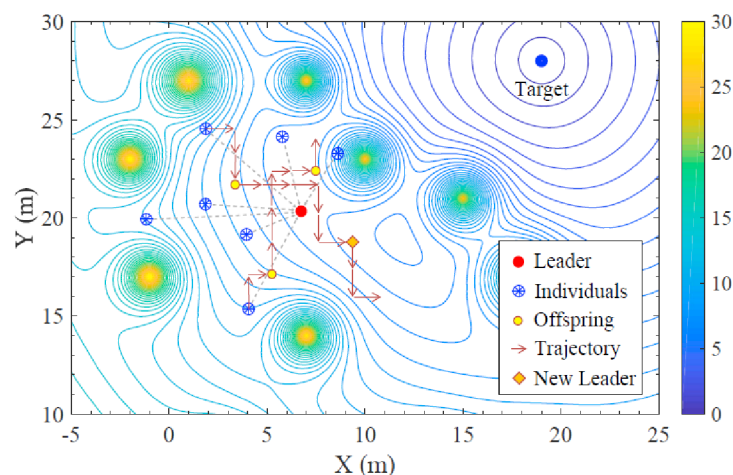
Nastavení kroku je závislé na typu úlohy a spolu s PRT významně ovlivňuje výkon SOMA. V SOMA jsou parametry velikost populace, délka cesty, krok a PRT inicializovány na začátku algoritmu jako konstanty [40]. Algoritmus poběží, dokud nebude splněno kritérium zastavení. Mezi zastavovací kritéria patří například počet iterací, minimální hodnota kvality atd. Výkonnost SOMA, podobně jako u většiny evolučních algoritmů, závisí na konkrétním nastavení některých řídicích parametrů (PS, PL, ns a PRT). U SOMY celkově platí vysoká citlivost na nastavení parametrů. Strategie

popsána výše je SOMA AllToIne. V této strategii se všichni jednotlivci až na vůdce pohybují směrem k vůdci [22].

6.3.3 Základní operátory v SOMA

Vývoj v SOMA se provádí pomocí sady stochastických evolučních operátorů, které manipulují s jednotlivci v populaci. Operátorům v klasických strategiích SOMA se říká perturbace a křížení. V evolučních algoritmech zajišťuje mutace rozmanitost populace a poskytuje prostředky k obnovení ztracených informací. U SOMA nelze použít mutaci, protože nejsou vytvářeni žádní noví jedinci. Stejného efektu můžeme ale dosáhnout pomocí perturbace (vychylování). Perturbace závisí na perturbačním vektoru (PV), který je generován podle parametru PRT. Nový PV vektor je vytvořený pro každého jedince předtím, než se začne pohybovat ve vyhledávacím prostoru [19]. PRTVector se mění při každém jednotlivém skoku jedince, aby se zajistila rozmanitost populace a zvýšila se účinnost algoritmu. Jednotlivé trajektorie jedinců nejsou proto přímé viz Obrázek 5: Na rozdíl od evolučních algoritmů se mutace, u SOMA perturbace, provádí před křížením [20].

Vzhledem k tomu, že SOMA nevytváří nové jednotlivce jako kombinace dvou nebo více jedinců, neexistuje žádný operátor křížení jako v evolučních algoritmech. Operátor podobný křížení je však dán pohybem jednotlivců ve vyhledávacím prostoru. V rámci každé migrační smyčky jednotlivci prozkoumají sadu pozic, zapamatují si tu nejlepší pozici nalezenou na své trajektorii a na konci této fáze se vrátí do této pozice [19]. Procesem křížení můžeme nazvat v SOMA proces kdy dojde přesunu (perturbativně) jedinců do nové pozice. PRTVector způsobí, že se jedinec posunem směrem k vůdci a následuje trajektorii vytvořenou dříve v procesu perturbace. Jedince na nové pozici nazýváme potomci [20].



Obrázek 5: Princip SOMA. Obrázek převzatý z [20]. Jednotliví jedinci jsou zobrazeni modrými kolečky a vůdce je červeným kolečkem. Trajektorie jedinců jsou zobrazeny pomocí šipek. Nový vůdce jsou zobrazeni kosoúhelníkem. Potomstva žlutým kolečkem.

7. PRAKTICKÁ ČÁST

7.1 První část

Pomocí nástroje Matlab Simulink byl vytvořen tříkompartmentový model. Byl simulován průběh diluční křivky indikátoru ve tkáni po intravenózním aplikaci bolusové injekce. Vytvořili se dvě křivky časové intenzity, jedna byla s kratším časem dosažení maximální intenzity t_p , tj. s rychlejším nárůstem a poklesem křivky. Křivky byly zarušeny generátorem náhodných čísel a dvěma simulovanými vlnami s různou fází. Provedlo se proložení diluční křivky pomocí nelineární metody nejmenších čtverců a pomocí spojitého genetického algoritmu. Kritérium pro hodnocení kvality odhadu byl součet kvadrátů odchylek mezi neznámou funkcí a simulovaným průběhem diluční křivky. U spojitého genetického algoritmu byla vybrána nejlepší hodnota z pěti realizací algoritmu. Dále byly použity k výpočtu křivky čtyři teoretické modely, LDR, FPT, lognormální a gama funkce. K hodnocení byl použit RMSE a R kvadrát. Optimalizovali se dva parametry, nazveme je a a b . V Tabulka 2 je vypsáno čemu se rovnají parametry optimalizace a a b u vybraných matematických funkcí.

parametr	a	b
Gama	α	β
Lognormální	μ	σ
FPT	λ	μ
LDRW	λ	μ

Tabulka 2: Parametry optimalizace Gama funkce, Lognormální funkce, FPT a LDRW

Vzhledem k tomu, že simulované diluční křivky dosahovaly vysokých intenzit, teoretické modely byly vynásobeny určitou hodnotou amplitudy. Každá výstupní hodnota modelu byla vynásobena stejnou hodnotou amplitudy, nedošlo proto ke změně tvaru křivky. Hodnoty na osy y se zvětšili podle:

$$y^{new} = y^{old} \cdot amplituda. \quad (7.1)$$

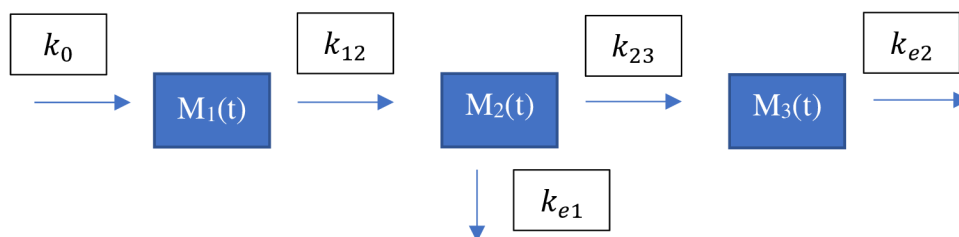
Hodnota amplitudy u každého modelu byla odhadnuta podle grafu a podle kvality optimalizace. Byla zobrazena diluční křivka modelu a teoretického modelu v jednom grafu, hodnoty amplitudy se zvyšovali nebo snižovali na základě nejlepšího přiblížení hodnotám intenzity simulované křivky a na základě kvality optimalizace.

Druhá fáze byla věnována vlivu intenzity šumu na výsledek aproximace. Vytvořili jsme třetí křivku nástrojem Matlab Simulink, která byla rušena třemi různými hodnotami šumu (generátorem náhodných čísel a dvěma simulovanými vlnami s různou fází) s různou hodnotou SNR (35,43dB, 26,11dB, 19,77dB). SNR určuje poměr signálu vůči šumu a uvádí se v dB. K výpočtu hodnoty SNR byla použita Matlab funkce *snr*. Opět jsme použili spojitý GA a metodu nejmenších čtverců. Opět jsme vybrali matematické

modely FPT, LDRW, lognormální a gama. Vzhledem k tomu, že spojitý genetický algoritmus je stochastický, tak jsme použili 100 realizací ke zhodnocení rozptylu R kvadrátu výsledných hodnot. Výsledné hodnoty R kvadrátu jsme zobrazili pomocí matlab boxplot. Matlab boxplot vykreslí vektor dat do tzv. boxu. Kdy vodorovná červená čára v boxu označuje medián. Spodní a horní hranice boxu označuje 25 a 75 percentil dat. Odlehlé hodnoty jsou znázorněny jako červené plusy.

7.1.1 Farmakokinetické modely

Pomocí nástroje Matlab Simulink byl vytvořen tříkompartmentový model. První kompartment modeloval vstup kontrastní látky do žilního oběhu, druhý modeloval difuzi činidla do levé komory a levé síně a třetí kompartment vstup do tkáně. V našem modelu jsme neuvažovali pronikání indikátoru z tkáně zpět do srdce. Na Obrázek 6 vidíme zjednodušený tříkompartmentový model, který byl použit k vytvoření diluční křivky pronikání indikátoru ve tkáni. Parametry k jsou rychlostní konstanty indikátoru a parametr M jsou kompartmenty. Parametr k_0 je rychlostní konstanta podávání indikátoru. Parametr k_e je rychlostní konstanta eliminace indikátoru. Rovnice jsou diferenciální rovnice pro množství indikátoru.

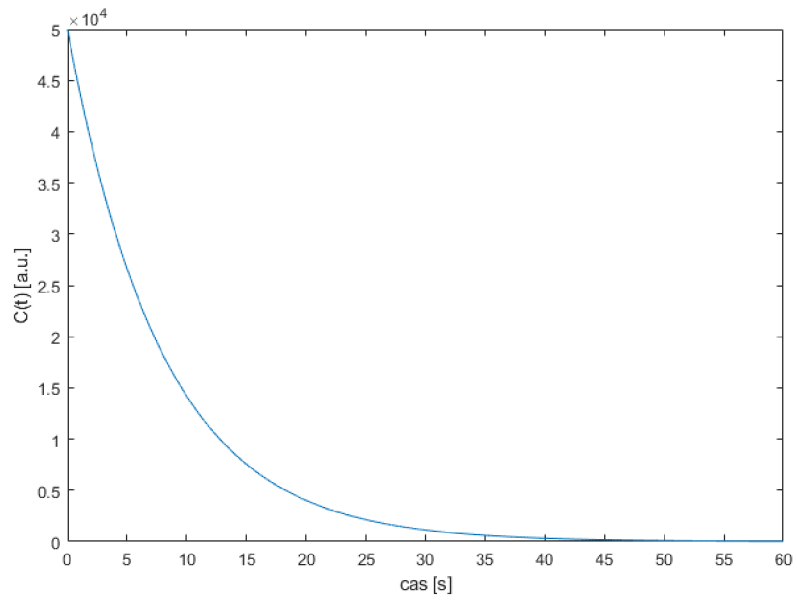


Obrázek 6: Zjednodušený tříkompartmentový model použitý k vytvoření koncentrační křivky

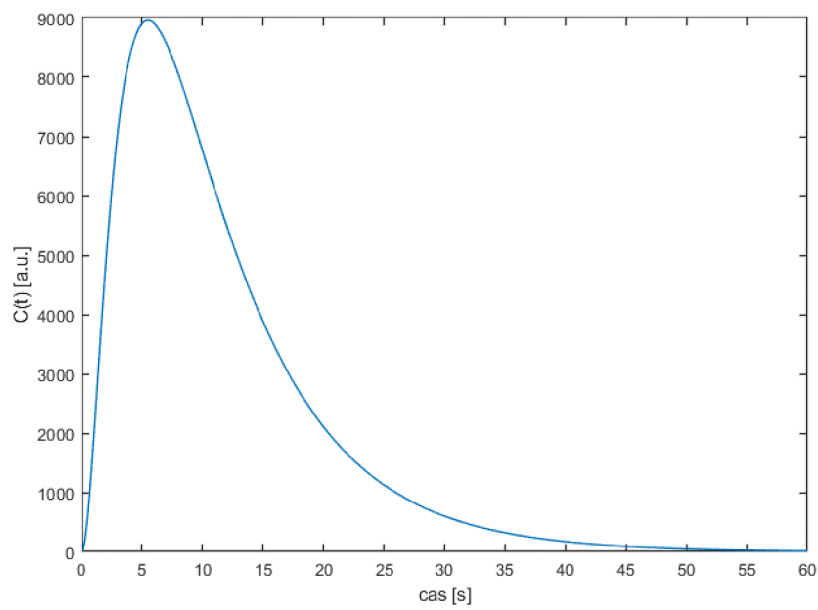
$$\frac{dM_1(t)}{dt} = k_0 - k_{12} \cdot M_1(t) \quad (7.2)$$

$$\frac{dM_2(t)}{dt} = k_{12} \cdot M_1 - k_{e1} \cdot M_2(t) - k_{23} \cdot M_2(t) \quad (7.3)$$

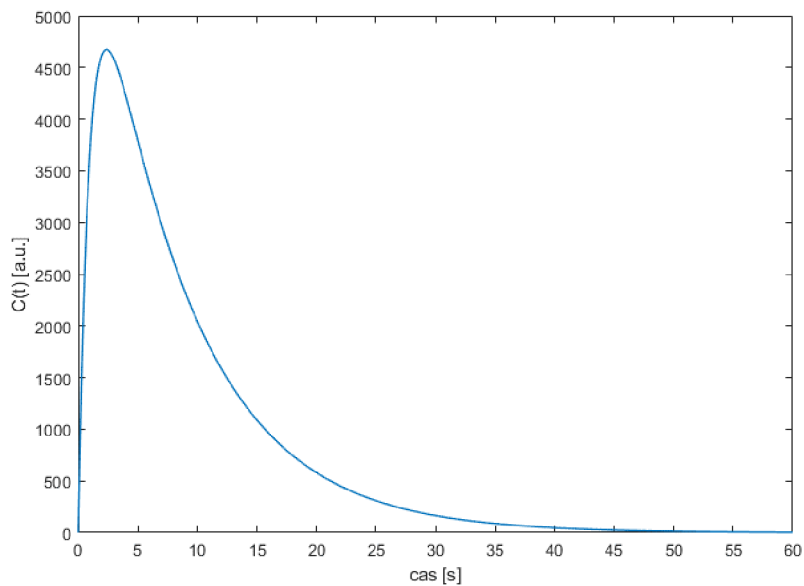
$$\frac{dM_3(t)}{dt} = k_{23} \cdot M_2 - k_{e2} \quad (7.3)$$



Obrázek 7: Simulovaný průběh vstupu indikátoru do žíly, výstup z prvního kompartmentu M_1 .



Obrázek 8: Simulovaný průběh průchodu vstupu indikátoru do srdce, výstup z druhého kompartmentu M_2 .



Obrázek 9: Simulovaný průběh průchodu indikátoru tkání, výstup ze třetího indikátoru M_3 .

7.1.2 Optimization toolbox Matlab

Pro proložení křivky pomocí metody nejmenších čtverců byla využita aplikace Curve Fitting z Optimization toolbox Matlab, která využívá Matlab funkci *fit*. Maximální počet iterací jsme zvolili 300 u každé funkce. Počáteční odhad parametrů se lišil podle použité funkce.

- Gama funkce: $\alpha = 0,7$ a $\beta = 2$
- Lognormální funkce: $\mu = 0,7$ a $\sigma = 2$
- FPT funkce: $\lambda = 0,7$ a $\mu = 2$
- LDRW funkce: $\lambda = 0,7$ a $\mu = 0,5$

7.1.3 Spojitý genetický algoritmus

Kritériem pro hodnocení kvality odhadu byl součet kvadrátů odchylek mezi simulovanou křivkou a výslednou křivkou optimalizace:

$$krit = \frac{1}{N} \sum (y_n - y_t)^2. \quad (7.4)$$

Kde y_t je výsledná funkce optimalizace spojitého GA vytvořená pomocí matematických modelů s nalezenými parametry (gama, LDRW, FPT, lognormální). N je počet vzorků signálu y_n a y_t . Parametr y_n je funkce vytvořená nástrojem Simulink, tj. teoretická diluční křivka průchodu indikátoru tkání ROI. Čím menší hodnota kritériální funkce tím je lepší aproximace funkce. Parametry Spojitého GA jsou v Tabulka 3. Hodnoty byly zvolené na základě vícečetného spuštění běhu algoritmu.

Maximální počet iterací	50
Počet jedinců	20
Pravděpodobnost mutace	0,4
Volba elitářství	1

Tabulka 3: Parametry Spojitého GA.

V bloku reprodukce se náhodně vybírala dvojice jedinců P_1 a P_2 , kteří se reprodukují na potomky. Potomci se vypočítali dle následujících rovnic, kde β je náhodně generované číslo z intervalu (0,1).

$$O'_1 = \beta P_1 + (1 - \beta)P_2 \quad (7.5)$$

$$O'_2 = (1 - \beta)P_1 + \beta P_2 \quad (7.6)$$

7.1.4 První praktická část – A

	FPT	LDRW	lognormal	gama
RMSE	3428,29	11134,37	4508,47	854,23
R ²	0,92	0,15	0,86	0,99
a	0,98	1,06	3,02	2,23
b	18,75	10,00	0,63	9,41
MTT	18,75	10,00	24,96	27,18
tp	5,38	7,04	13,85	24,57

Tabulka 4: Hodnoty Parametrů po proložení diluční křivky 1. metodou nejmenších čtverců.

	FPT	LDRW	lognormal	gama
RMSE	14622,93	8568,88	4702,49	3712,98
R ²	0,63	0,64	0,96	0,99
a	7,13	0,95	3,00	2,89
b	12,58	11,83	0,67	8,14
MTT	12,58	11,83	25,10	26,46
tp	9,93	6,36	12,77	23,57

Tabulka 5: Hodnoty parametrů po proložení diluční křivky 1. metodou spojitého genetického algoritmu.

Hodnoty MTT a tp byly vypočítané podle vztahů z teoretické části 2. Pomocí hodnot R kvadrátu a RMSE jsme zjistili, že nejhorší výsledek proložení diluční křivky 1. u metody nejmenších čtverců je u funkce LDRW a nejlepší u funkce gama. U proložení první diluční křivky pomocí metody spojitého genetického algoritmu nám vyšla také nejlépe gama funkce. Nejhorší výsledek proložení byl u FPT funkce. Gama funkce byla s

hodnotami $RMSE=854,23$ a $R^2=0,99$ lepší u metody nejmenší čtverců než u metody spojitého genetického algoritmu, kde hodnoty byly $RMSE=3712$ a $R^2=0,99$.

	FPT	LDRW	lognormal	gama
RMSE	524,57	405,65	432,19	199,97
R^2	0,96	0,98	0,98	0,99
a	0,91	1,65	2,18	2,28
b	7,12	6,82	0,72	4,59
MTT	7,12	6,82	11,45	12,75
tp	1,67	13,74	5,27	10,47

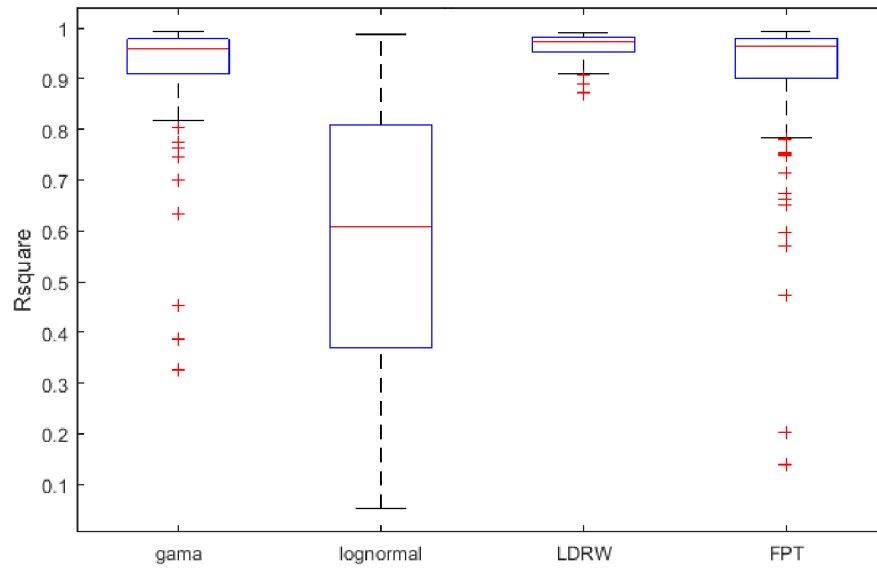
Tabulka 6: Hodnoty parametrů po proložení diluční křivky 2. metodou nejmenších čtverců.

	FPT	LDRW	lognormal	gama
RMSE	319,14	609,47	3130,14	569,92
R^2	0,99	0,97	0,61	0,99
a	6,35	9,40	2,32	2,19
b	6,76	6,45	1,12	4,91
MTT	6,76	6,45	19,21	12,95
tp	5,35	6,12	2,89	10,76

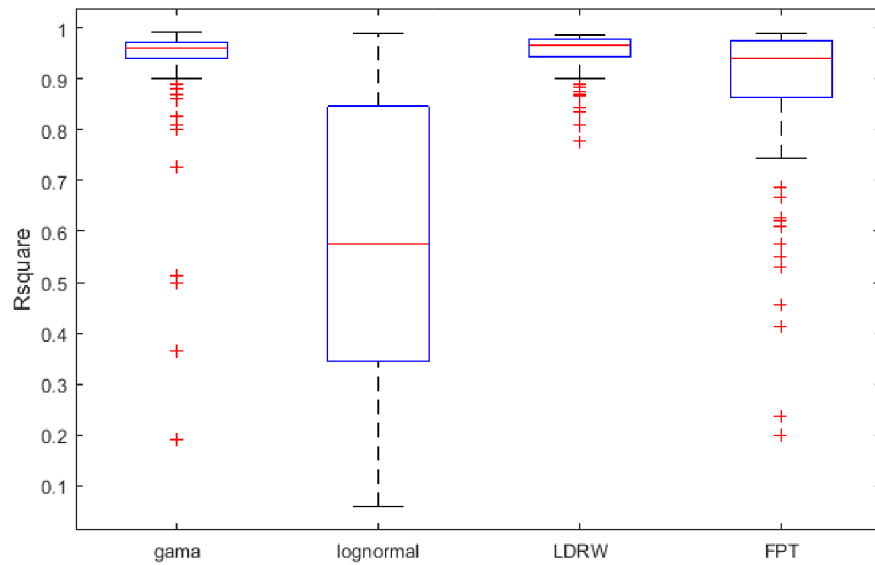
Tabulka 7: Hodnoty parametrů po proložení diluční křivky 2. metodou spojitého genetického algoritmu.

U druhé diluční křivky nám také u obou metod vyšlo nejlepší proložení pomocí gama funkce. U metody nejmenších čtverců nám nejhorší aproximace vyšla u funkce FPT a u metody spojitého genetického algoritmu bylo nejhorší proložení lognormální funkcí.

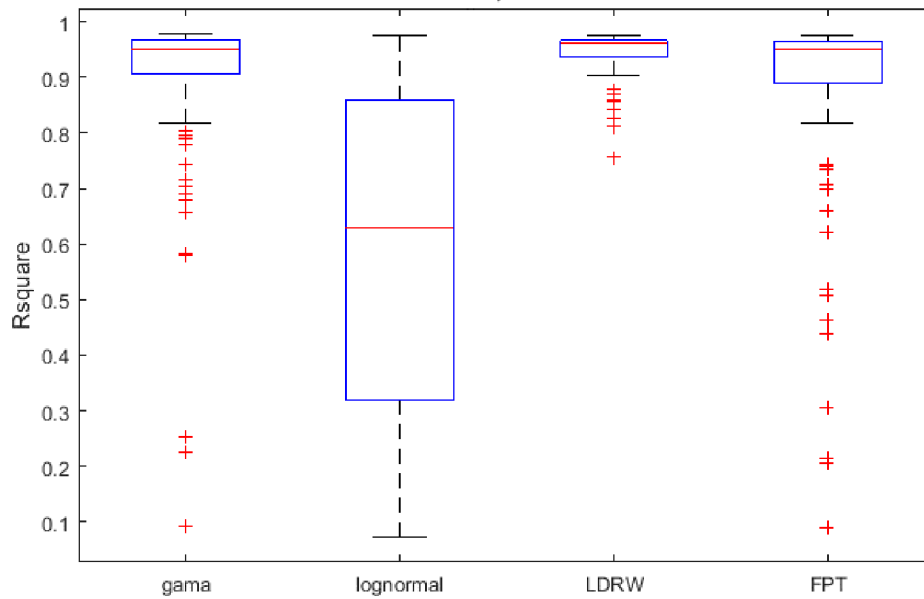
7.1.5 První praktická část – B



Obrázek 10: Boxplot pro diluční křivku rušenou šumem s hodnotou SNR= 19,77 dB.



Obrázek 11: boxplot pro diluční křivku rušenou šumem s hodnotou SNR=26,11 dB.



Obrázek 12: boxplot pro diluční křivku rušenou šumem s hodnotou SNR=35,43 dB

	Gama	lognormal	LDRW	FPT
minimum	0,326	0,053	0,871	0,139
maximum	0,993	0,988	0,990	0,993
rozptyl	0,013	0,069	0,001	0,021
medián	0,959	0,609	0,973	0,963

Tabulka 8: Hodnoty maxima, minima, mediánu a rozptylu 100 hodnot R kvadrátu pro diluční křivku rušenou šumem s hodnotou SNR=19,77 dB u spojitého GA.

	Gama	lognormal	LDRW	FPT
minimum	0,190	0,059	0,779	0,200
maximum	0,992	0,990	0,985	0,990
rozptyl	0,015	0,077	0,002	0,024
medián	0,960	0,574	0,965	0,941

Tabulka 9: Hodnoty maxima, minima, mediánu a rozptylu 100 hodnot R kvadrátu pro diluční křivku rušenou šumem s hodnotou SNR= 26,11 dB u spojitého GA.

	Gama	lognormal	LDRW	FPT
minimum	0,093	0,073	0,757	0,089
maximum	0,979	0,977	0,977	0,976
rozptyl	0,023	0,083	0,001	0,033
medián	0,951	0,629	0,961	0,950

Tabulka 10: Hodnoty maxima, minima, průměru a rozptylu 100 hodnot R kvadrátu pro diluční křivku rušenou šumem s hodnotou SNR=35,43 dB u spojitého GA.

	Gama	lognormal	LDRW	FPT
SNR [dB]	R kvadrát			
19,77	0,962	0,955	0,951	0,949
26,11	0,946	0,947	0,959	0,952
35,43	0,933	0,937	0,946	0,941

Tabulka 11: Hodnoty R kvadrátu vypočítaného metodou nejmenších čtverců pro tři realizace šumu a čtyři teoretické funkce Gama, lognormální, LDRW, FPT.

U metody spojitého genetického algoritmu vyšlo nejhůře proložení funkcí lognormální u všech realizací šumu. Hodnoty měly největší rozptyl hodnot a nejnižší hodnotu mediánu a 75 percentilu. Hodnoty mediánu byly u lognormálního rozdělení v rozsahu od 0,574 - 0,629 u všech realizací šumu, zatímco u ostatní teoretických funkcí nebyla hodnota mediánu nižší než 0,9. Dále je patrné, že s přibývajícím hodnotou SNR se zvyšuje rozptyl dat u všech teoretických funkcí, pokud počítáme i s extrémními hodnotami. Netýká se to pouze hodnoty LDRW u SNR= 35,43 dB, kde došlo poklesu rozptylu o 0,001 z druhé největší realizace šumu na největší. Vliv šumu můžeme vidět na poklesu maximálních hodnot R kvadrátu s přibývajícím hodnotou šumu u všech funkcí kromě lognormální funkce. Druhý největší rozptyl hodnot a druhou nejmenší hodnotu mediánu měla funkce FPT u všech realizací, a proto můžeme říct, že se jedná o druhou nejhorší funkci aproximace u spojitého GA.

U metody nejmenších čtverců vidíme pokles hodnoty R kvadrátu s přibývajícím hodnotou šumu. Neplatí to pro funkci LDRW a FPT, kdy druhá největší realizace šumu má větší hodnotu R kvadrátu než první realizace. Pokud metodu nejmenších čtverců porovnáme se spojitým genetickým algoritmem je patrné, že metoda nejmenších čtverců lépe aproximuje lognormální funkcí. Pokud porovnáme hodnoty mediánu u spojitého genetického algoritmu s hodnotami R kvadrátu u metody nejmenších čtverců jsou hodnoty u spojitého algoritmu ve většině případů nepatrně lepší, netýká se to lognormální funkce. Až na lognormální funkci nejsou velké rozdíly hodnot R kvadrátu u spojitého genetického algoritmu a u metody nejmenších čtverců. Všechny hodnoty jsou větší jak 0,9.

7.2 Druhá praktická část – simulovaná data

V této praktické části jsme se zabývali metodou bolust&burst. Pracovali jsme se souborem simulovaných dat založených na reálných měření, která simulovala průběh koncentrace kontrastního činidla ve tkáni ROI pomocí metody bolust&burst. Soubor simulovaných dat obsahoval pět úrovní Gaussova šumu odpovídající SNR 10, 20, 30, 40 a 50 dB. Pro každou úroveň šumu bylo vygenerováno 40 realizací náhodného šumu. Celkem jsme měli 200 realizací signálů.

V této optimalizační úloze jsme hledali 7 parametrů. Prvních 6 parametrů určovalo průběh AIF a sedmý parametr průběh TRF. Průběh AIF byl vytvořen pomocí parametrického vyjádření jako součet dvou lognormálních funkcí podle rovnice:

$$\text{AIF} = n_1 \cdot \text{LN}(t, n_2, n_3) + n_4 \cdot \text{LN}(t, n_5, n_6), \quad (7.7)$$

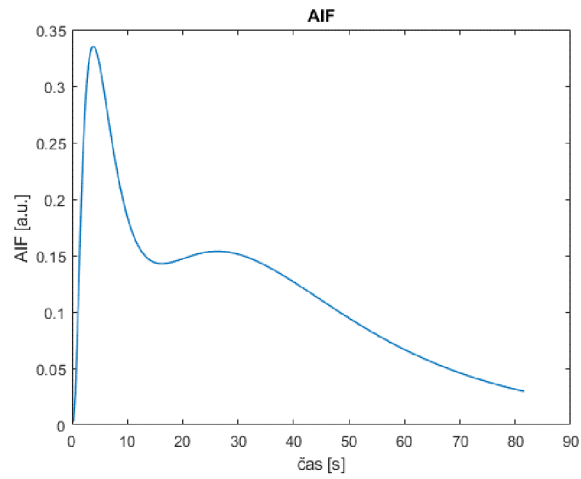
kde $(n_1 \dots n_6)$ jsou parametry, které optimalizujeme. Parametr t je vektor hodnot osy x . $\text{LN}(t, \mu, \sigma)$ je označení Log-normálního rozdělení. Dle kapitoly 2.1 platí, že μ je průměr a σ je standardní odchylka normálního rozdělení logaritmu nezávislé proměnné t . Kde σ je parametrem tvaru a μ je parametr umístění. Ukázková funkce AIF je na obrázku Obrázek 13.

Pracovalo se s předpokladem metody bolust&burst, že je časový průběh koncentrace kontrastní látky v oblasti zájmu (ROI) popsán konvolucí arteriální vstupní funkce (AIF) a reziduální funkce tkáně (TRF).

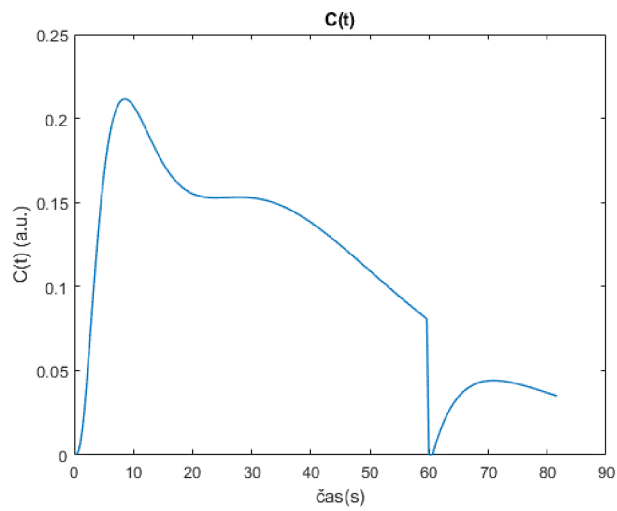
V optimalizaci jsme srovnávali tři typy algoritmů, spojitý genetický algoritmus GA, PCO a SOMA. Algoritmy optimalizovali 7 parametrů. V každém běhu algoritmu se poslalo na vstup funkce pro výpočet AIF prvních 6 parametrů. Výsledná křivka AIF a poslední optimalizační parametr (definující TRF) vstupovali do funkce pro výpočet konvoluce AIF a TRF. Výsledek konvoluce byl odhad $C(t)'$ simulované koncentrační křivky $C(t)$ rušenou Gaussovým šumem. Kritérium pro hodnocení kvality odhadu byl součet kvadrátu odchylek (MSE) mezi funkcí $C(t)'$ a vstupní simulovanou funkcí $C(t)$ o určité realizaci šumu.

Kritérium hodnocení kvality jednotlivých metod bylo MSE mezi výslednou křivkou jednotlivých algoritmů $C(t)'_{pso}$, $C(t)'_{soma}$, $C(t)'_{ga}$, a původní simulovanou křivkou $C(t)$ bez šumu. Vzhledem k tomu, že se jedná o stochastické algoritmy, jsme pro každou realizaci signálu uskutečnili 10 běhu algoritmu. Následně jsme vybrali pouze jeden nejlepší výsledek.

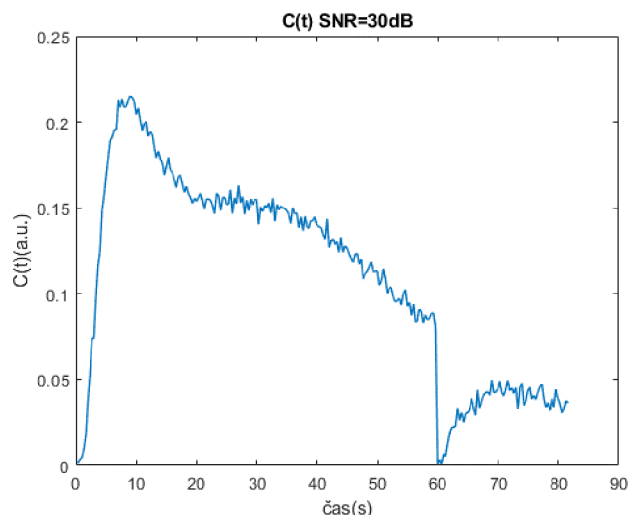
Funkce pro výpočet AIF, konvoluce, načítání sady signálů a simulované průběhy byly převzaty z [43].



Obrázek 13: simulovaný průběh AIF



Obrázek 14: simulovaný průběh $C(t)$



Obrázek 15: Simulovaný průběh $C(t)$ se SNR=30 dB

7.2.1 Spojitý genetický algoritmus GA

Nastavení hodnot jednotlivých parametrů spojitého genetického algoritmu je v Tabulka 12:.

Velikost populace	1 000
Maximální počet iterací	30 000
Pravděpodobnost mutace	0,4
Maximální hodnoty parametrů	[8 4 4 4 4 1]
Minimální hodnoty parametrů	[3 0 0 0 0 0]
Volba elitářství	1

Tabulka 12: hodnoty parametrů GA

Velikost populace byla zvolena na základě vícečetného spuštění běhu algoritmu, kdy byla vybrána největší hodnota velikosti populace, která ještě měla vliv na účinnost algoritmu. Tj. zvyšování populace nad 1000 už nemělo vliv na výsledek optimalizace. Maximální počet iterací byl zvolen také na základě vícečetného spuštění běhu algoritmu, byla vybrána maximální hodnota počtu iterací, která dostatečně zlepšovala účinnost algoritmu. Mezi kritérium pro ukončení algoritmu jsme zvolili maximální počet iterací a velikost stagnace. Hodnoty počtu iterací větší než 30 000 měli zanedbatelný výsledný účinek na optimalizaci. Vzhledem k tomu, že se s počtem iterací zvyšovala časová náročnost algoritmu, nebylo žádoucí volit větší hodnoty počtu iterací. Maximální velikost stagnace bylo zvolena 30. Hodnota stagnace před spuštěním algoritmu je 0. V každém cyklu algoritmu se ptáme, zda je rozdíl nejlepšího jedince od nejlepšího v minulém kroku menší než 10^{-6} . Pokud ano, dojde k navýšení hodnoty stagnace o jedna. Aby došlo k ukončení algoritmu musí platit obě ukončovací podmínky. To znamená počet iterací je větší, jak maximální počet iterací a hodnota stagnace je rovno nebo větší než 30.

V prvních spuštění algoritmu jsme náhodně zvolili maximální hodnoty všech parametrů 10 a minimální 0. Abychom snížili výpočetní čas algoritmu, jsme potom rozsahy parametrů omezili. Po více spuštění běhu výsledek parametru jedna dosahoval hodnot v intervalu (6 – 7,5), na základě těchto předpokladů byla heureristicky zvolena maximální hodnota 8 a minimální hodnota 3. Vzhledem k tomu, že parametry 2-5 nenabývali hodnot větších jak 4, jsme zvolili maximální hodnotu 4 a minimální 0. Poslední parametr byl zvolen v intervalu (0,1).

Ke spojitému algoritmu, na rozdíl od první praktické části, jsme přidali funkci *zrcadlení*, která zajišťuje, že při běhu algoritmu nedojde k dosažení hodnot parametrů mimo určený interval. Funkce „zrcadlení“ byla převzata z [24]. Tato funkce je běžnou součástí rojových algoritmů SOMA a PSO a běžně se nevyskytuje u spojitých genetických algoritmů. Funkce byla použita proto, že algoritmus měl tendenci během běhu opustit daný interval hodnot a nekonvergoval zpět na daný interval. To mělo za následek vysokou chybovost výsledných hodnot.

7.2.2 Algoritmus PSO

Nastavení hodnot jednotlivých parametrů algoritmu PSO jsou v Tabulka 13.

Velikost populace	300
Maximální počet iterací	800
c_1	2
c_2	2
Maximální hodnoty parametrů	[8 4 4 4 4 4 1]
Minimální hodnoty parametrů	[3 0 0 0 0 0 0]
ω_{max}	0,9
ω_{min}	0,4

Tabulka 13: Hodnoty parametrů PSO

Velikost populace jsme zvolili heureristicky na základě několika předpokladů. Vycházelo se z předpokladu dle článku [17], že ve většině případů bylo prokázáno, že pokud je počet jedinců vyšší než 50, dojde k tomu, že při dalším zvyšování počtu jedinců se snižuje citlivost PSO na změnu velikosti populace. Jako počáteční hodnota populace byla zvolena populace velikosti 50. Pustili jsme několik běhů algoritmu a postupně jsme navyšovaly velikost populace a vyhodnocovali výsledek PSO. Zjistilo se, že pokud zvolíme hodnotu populace větší než 300 nedochází ke změně výsledku PSO. Algoritmus přestal být citlivý na další navyšování velikosti populace. Jako ukončovací kritérium bylo zvoleno maximální počet iterací. Maximální počet iterací byl také zvolen na základě několika běhů algoritmů. Navyšování počtu iterací nad 800 měl na výsledek PSO zanedbatelný vliv. Hodnoty c_1 , c_2 , ω_{max} a ω_{min} byly zvoleny na základě doporučení viz kapitola 6.2.2 a odkazy na články [16][18]. PSO pracovalo s interní klesající setrvačnou váhou.

Abychom zachovali podobné podmínky pro všechny algoritmy, zvolili jsme maximálních a minimálních hodnoty parametrů stejné jako u GA. Abychom se s hodnotami parametrů během cyklu nedostali mimo interval, byla použita stejně jako u GA funkce *zrcadlení*. Funkce pro výpočet PSO byla převzata z [18].

7.2.3 Algoritmus SOMA

Nastavení hodnot jednotlivých parametrů algoritmu SOMA jsou v Tabulka 14.

Velikost populace	1000
mass – přeskočení vůdce	2,5
krok	1,4
PRT	0,4
Max_evals	15 000
My_eps	0,001
Maximální hodnoty parametrů	[8 4 4 4 4 4 1]
Minimální hodnoty parametrů	[3 0 0 0 0 0 0]

Tabulka 14: Hodnoty Parametrů SOMA

Algoritmus SOMA byl z převzatý ze skript Evoluční algoritmy od Josefa Tvrdíka viz odkaz [24]. Algoritmus SOMA je varianty All to One. Mezi parametry algoritmu patří tzv. mass, jedná se o relativní velikost přeskočení vůdce a je vypočítaný dle vztahu:

$$mass = \frac{\|x'_l - r_0\|}{\|x_l - r_0\|}, \quad (7.8)$$

kde r_0 je výchozí pozice jedince, x_l je pozice vůdce a x'_l je poloha jedince, kam se daný jedinec dostal při přeskočení vůdce. $\|x'_l - r_0\|$ je norma vektoru $(x'_l - r_0)$ a $\|x_l - r_0\|$ je norma vektoru $(x_l - r_0)$. Normy vektoru odpovídají délkám úseček s koncovými body r_0, x_l , resp. r_0, x'_l . Hodnoty parametru mass jsou doporučeny z intervalu [1.1;3] [24].

Velikost kroku je doporučeno v interval [0,11; mass] a zároveň platí, že 1/krok není celé číslo [24].

Na základě těchto předpokladů jsme po více spuštění běhu SOMA hledali hodnoty z daných intervalů, které dají nejlepší výsledky SOMA. Zvolili jsme mass=2,5 a krok=1,4. Hodnotu PRT jsme zvolili 0,4 na základě doporučení viz [19]. Maximální a minimální hodnoty parametrů byly zvolené na základě zachování podobných podmínek, tedy stejné jako u GA a SOMA. Algoritmus SOMA má kritéria ukončení algoritmu určena dvěma podmínkami, které zároveň musí platit současně. První je, že rozdíl nejhoršího jedince z populace z hlediska kvality a nejlepšího je menší než *my_eps*. Algoritmus v každém cyklu seřadí matici populace od nejlepšího po nejhoršího jedince. Jedná se tedy o rozdíl posledního a prvního člena populace jedinců. Hodnotu *my_eps* jsme nastavili na 0,001. Další ukončující podmínka je srovnatelná s počtem iterací. Máme parametr *počet kroků*, který se vypočítá podle vztahu:

$$\text{počet kroků} = \frac{\text{mass}}{\text{krok}}, \quad (7.9)$$

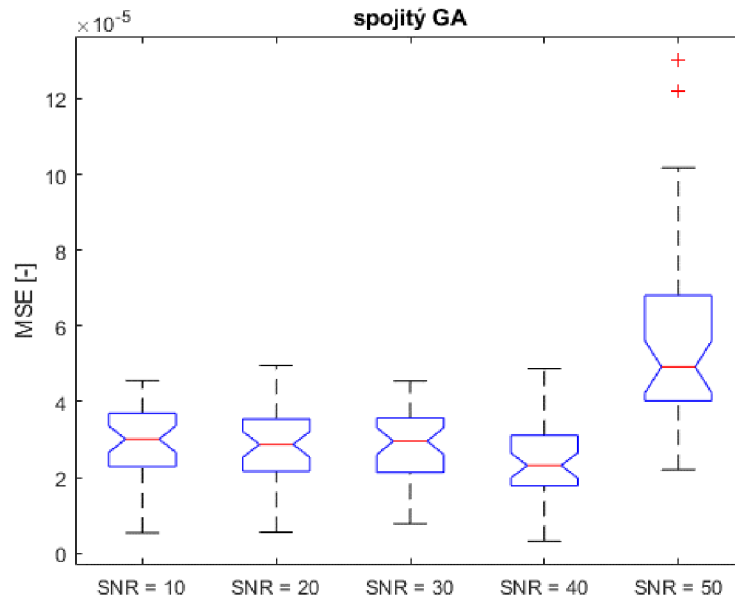
v programu se počet kroků zaokrouhlí pomocí matlabovské funkce *fix*. Tato funkce zaokrouhlí číslo na nejbližší celé číslo směrem k nule. V každém cyklu algoritmu dojde k navýšení proměnné nazvané *func_evals* o počet kroků. Počáteční hodnota *func_evals* je velikost populace. Další podmínka pro ukončení algoritmu je daná vztahem:

$$\text{func_evals} \geq d \cdot \text{max_evals}, \quad (7.10)$$

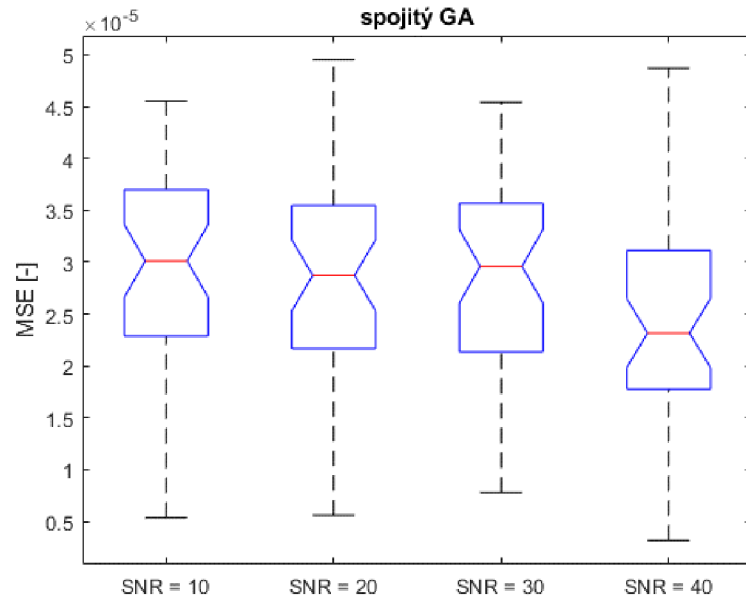
pokud platí následující rovnice, je splněna jedna z podmínek k ukončení algoritmu [24]. Parametr *d* je počet parametrů, v našem případě sedm a *max_evals* je proměnná, kterou jsme si zvolili na začátku ve fázi inicializace. Hodnotu *max_evals* jsme zvolili 15 000, hodnotu jsme vybrali na základě více spuštění algoritmu. Hodnotu *max_evals* jsme zvolili co největší a zároveň, aby její navyšování mělo ještě vliv na funkci SOMA a nezvětšovalo neúměrně výpočetní čas algoritmu. Maximální a minimální hodnoty parametrů jsme zvolili stejné jako u PSO a GA.

7.2.4 Vyhodnocení

Vyhodnocení bylo provedeno na základě MSE výsledné křivky vytvořené algoritmy spojitého GA, PSO a SOMA a simulované koncentrační křivky $C(t)$ bez Gaussova šumu. Dále byla hodnocena časová náročnost jednotlivých metod. K porovnání výkonu metod jsme použili Matlabovskou funkci *boxplot*. *Boxplot* zobrazí maximum, minimum, medián, 25 Percentil, 75 Percentil, rozsah hodnot mezi 25 a 75 percentilem a odlehlé hodnoty vstupní datové sady.



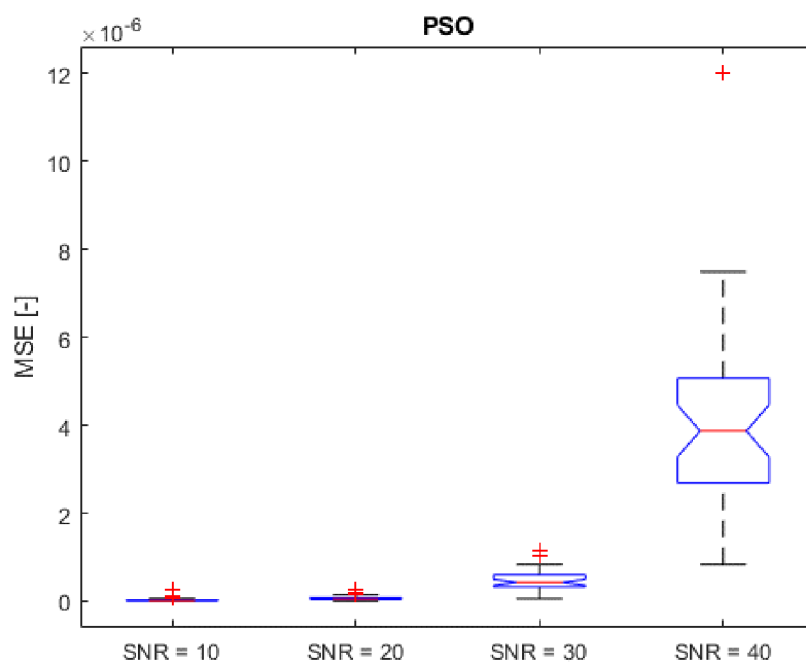
Obrázek 16: boxplot spojitý GA, MSE pro SNR= 10–50 dB



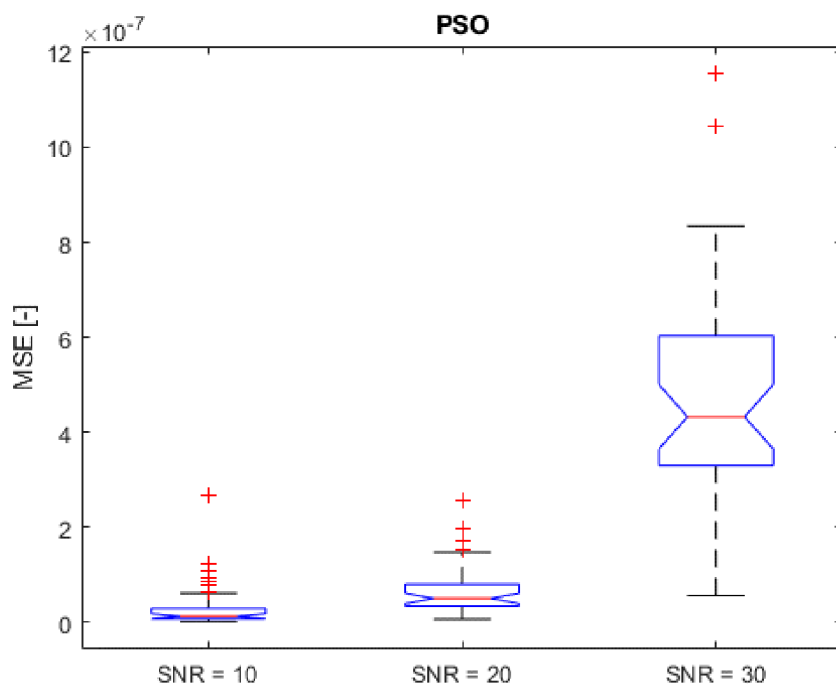
Obrázek 17: boxplot spojitý GA, MSE pro SNR= 10–40 dB

Hodnoty parametrů boxplotu jsou viz. Příloha A - U Spojitého GA z Obrázek 16 a Obrázek 17 je zřejmé, že pro signály s hodnotou SNR=50 dB algoritmus optimalizuje hůře než pro SNR= 10-40 dB. Maximum MSE u SNR=50 dB je více jak dvakrát větší než SNR=10-40 dB. Všechny hodnoty boxplotu: minimum, maximum, rozptyl, medián, 25 a 75 Percentil nabývají větších hodnot než u SNR=10-40 dB. Hodnoty mediánu u SNR=10-30 dB nabývají velmi podobných hodnot a to od $2,87 \cdot 10^{-5}$ do $3,01 \cdot 10^{-5}$. Hodnota mediánu $2,32 \cdot 10^{-5}$ u SNR=40 je menší než u SNR=10-30 dB. SNR=40 dB nabývá také menší hodnoty u 25-75 Percentilu. Hodnotu rozsahu mezi 25-75 Percentilem jsou si u SNR=10-30 dB velmi blízké s hodnotami $1,38 - 1,43 \cdot 10^{-5}$, u SNR=40 dB je hodnota ještě menší a to $1,34 \cdot 10^{-5}$. U SNR=50 dB s hodnotou $2,79 \cdot 10^{-5}$ se blížíme skoro dvojnásobku hodnot SNR=10-40 dB. Nejmenší minimální hodnotu nabývá SNR=40 dB s hodnotou $0,32 \cdot 10^{-5}$.

U metody PSO z Obrázek 18 je zřejmý rozdíl výkonu algoritmu v závislosti na šumu. S vyšší hodnotou šumu se zhoršuje výkon PSO. Metoda svým výkonem převyšuje spojitý algoritmus GA. Boxplot PSO pro SNR=10-50 dB je v příloze C.1. U SNR=50 dB to není tak zřejmé, i přes to všechny hodnoty parametrů jsou menší u PSO než u GA.



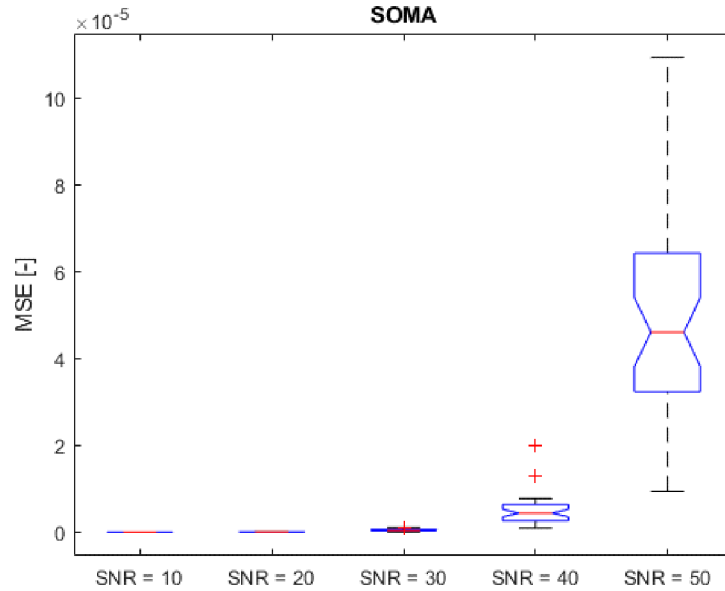
Obrázek 18: boxplot PSO, MSE pro SNR= 10–40 dB



Obrázek 19: boxplot PSO, MSE pro SNR= 10–30 dB

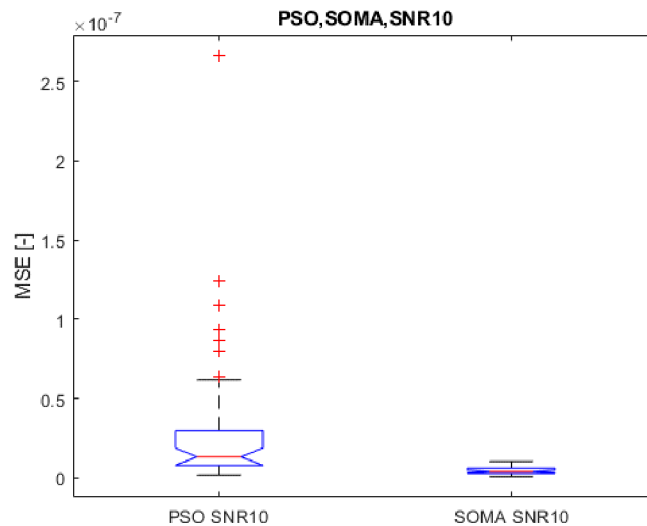
U PSO je vysoký rozdíl výkonu u SNR=50 dB a u SNR =10-40 dB viz příloha C.1., díky velkému maximum SNR=50 dB nelze vidět rozdíl odlišného výkonu u SNR=10-30 dB. Z Obrázek 18 lze vidět, že rozsah hodnot u SNR=10 dB s hodnotou $2,20 \cdot 10^{-8}$ je tak dvakrát menší než u SNR=20 dB s hodnotou $4,56 \cdot 10^{-8}$.

Například u SNR=20 dB a SNR=30 dB rozdíl rozsahu hodnot s hodnotami $4,56 \cdot 10^{-8}$ a $27,3 \cdot 10^{-8}$ je už velmi velký. Jak vidíme na Obrázek 20 u metody SOMA je také sestupná tendence výkonu se zvyšující hodnotou SNR. Optimalizace SOMA jako u PSO vykazuje lepší výkon optimalizace než u spojitého GA. V další části jsme se zabývali porovnáním výkonu PSO a SOMA.

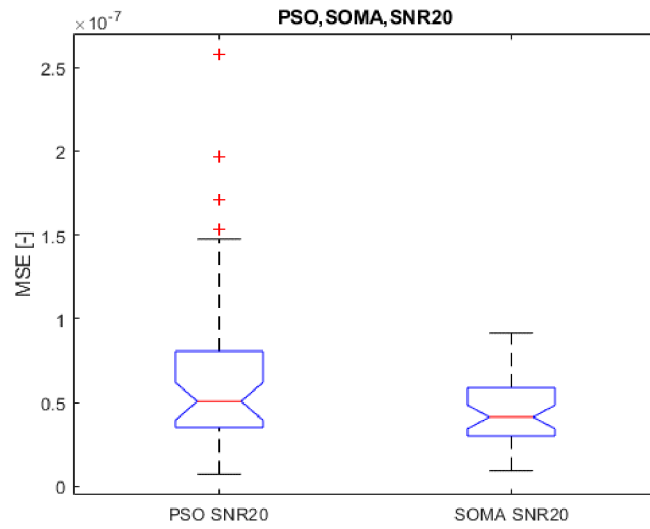


Obrázek 20: boxplot SOMA, MSE pro SNR= 10–50 dB

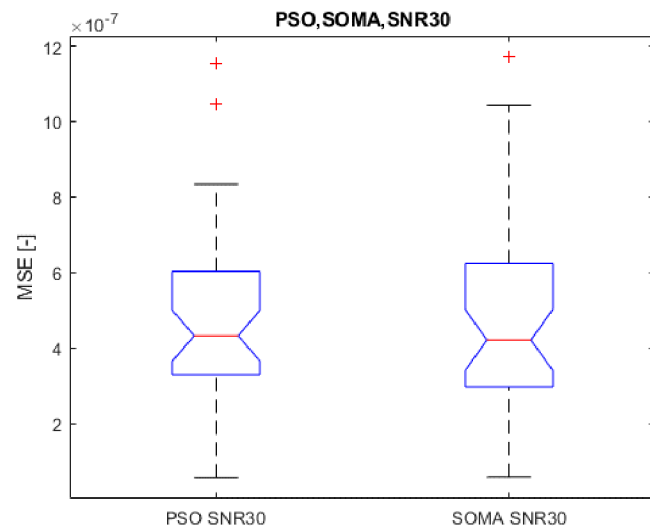
7.2.5 Vyhodnocení MSE pro PSO a SOMA



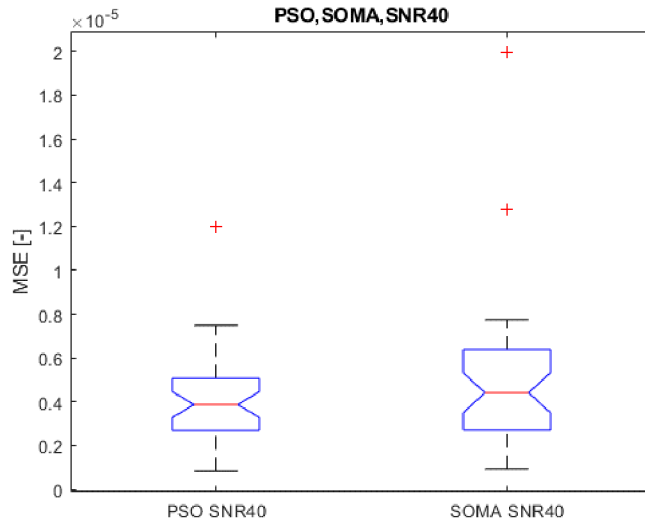
Obrázek 21: MSE PSO vs SOMA se SNR=10 dB



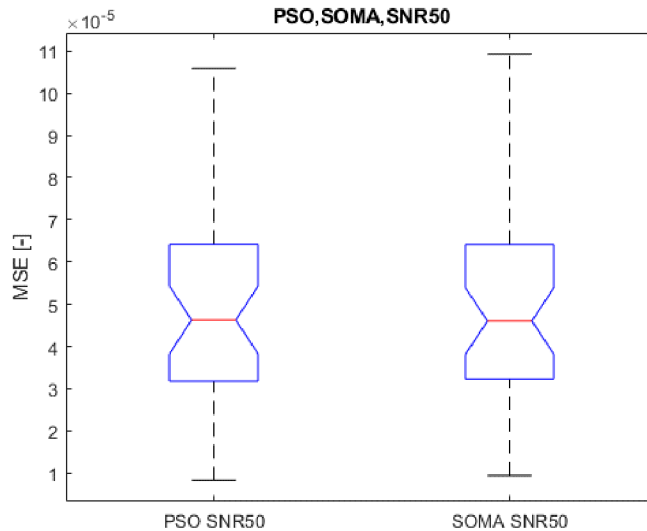
Obrázek 22: MSE PSO vs SOMA se SNR=20 dB



Obrázek 23: MSE PSO vs SOMA se SNR=30 dB



Obrázek 24: MSE PSO vs SOMA se SNR=40 dB

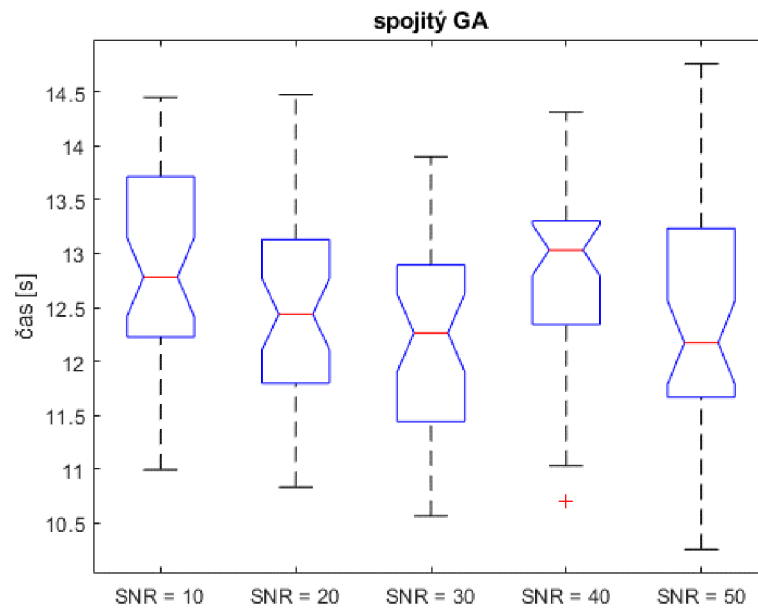


Obrázek 25: MSE PSO vs SOMA se SNR=50 dB

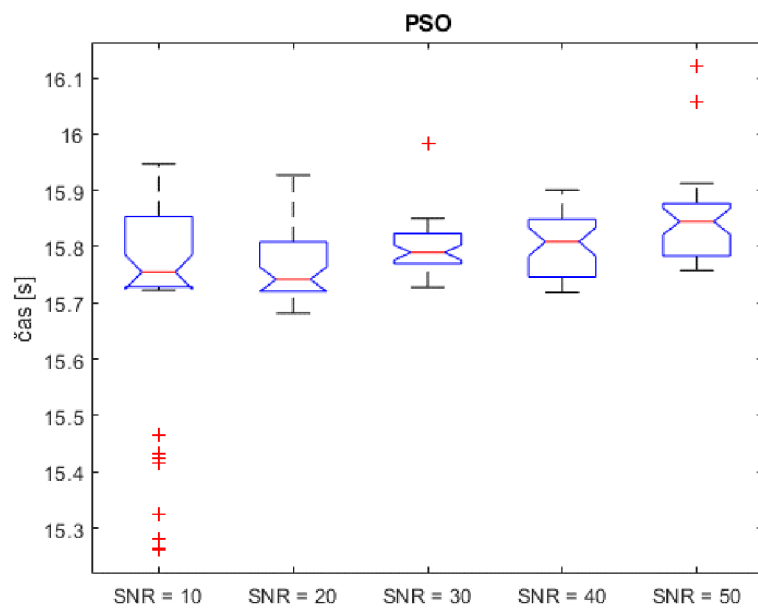
U SNR=10 a 20 dB je lepší výkonnost metody SOMA než PSO. U SNR= 30 dB je menší hodnota maxima, minim, rozsahu a 75 percentilu u PSO, hodnota mediánu a 25 Percentilu je nepatrně menší u SOMA. U SNR=40 dB je lepší ve všech parametrech metoda PSO. Výkonnost metod u SNR =50 dB je u PSO a SOMA téměř srovnatelná.

7.2.6 Srovnání časové náročnosti GA, SOMA a PSO

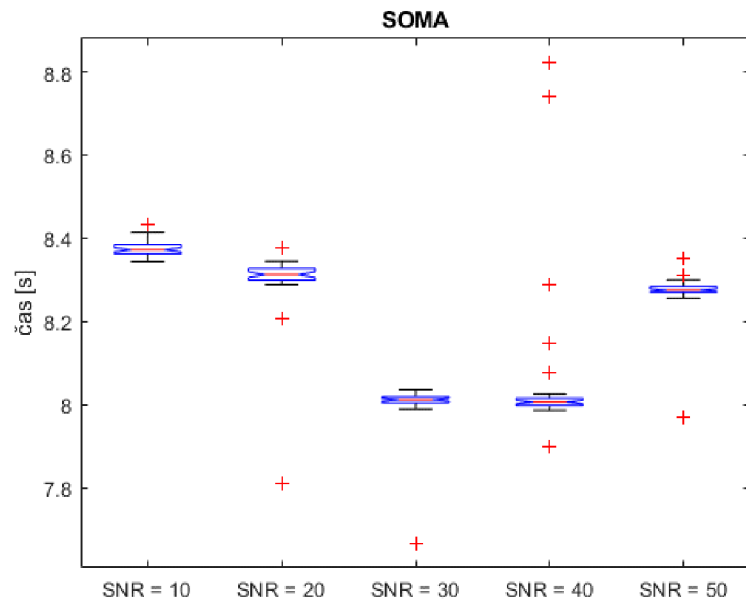
Na Obrázek 26, Obrázek 27 a Obrázek 28 je srovnání doby trvání optimalizace pro spojitý GA, PSO a SOMA pro SNR=10–50 dB. Nejrychlejší je algoritmus SOMA s hodnotami kolem 8 sekund. Nejpomalejší algoritmus je metoda PSO s hodnotami nad 15 sekund. Spojitý GA má největší rozsahy hodnot od 10-15 sekund. Rozsah hodnot mezi 25-75 percentilem je větší jak 1 sekunda.



Obrázek 26: boxplot Spojitý GA, čas pro SNR= 10–50 dB



Obrázek 27: boxplot Spojitý PSO, čas pro SNR= 10–50 dB

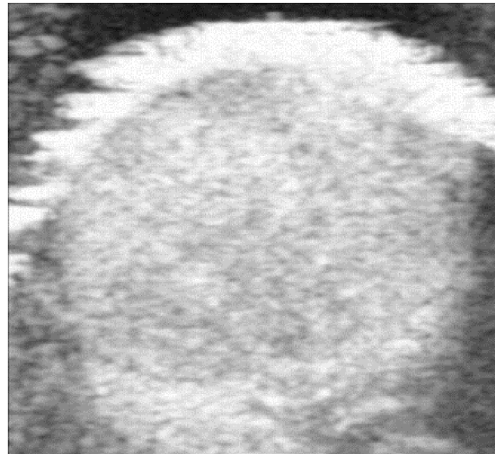


Obrázek 28: boxplot SOMA, čas pro SNR= 10–50 dB

7.3 Druhá část – klinická data

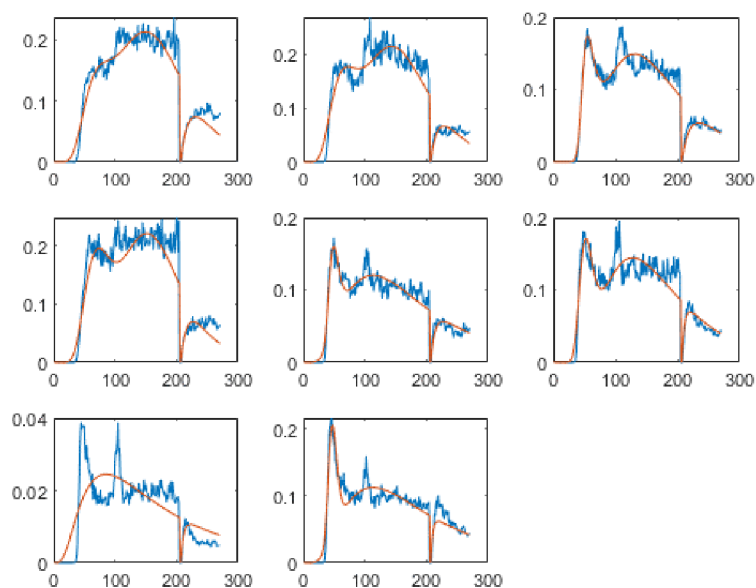
Tato část se zabývala optimalizací reálných klinický dat. V této části jsme použili k optimalizaci metody PSO a SOMA. Spojitý genetický GA v optimalizaci reálných klinických dat vykazoval vysokou chybovost. Pracovali jsme se třemi datovými sady ze tří různých experimentů. Každá sada obsahovala 8 signálů. Pro každý signál jsme spustili program 10 × a vybrali nejlepší realizaci. Kritériem pro hodnocení kvality optimalizace bylo MSE mezi reálným signálem a hledanou křivkou optimalizace. Počet parametrů zůstal zachovaný. U parametru PSO jsme navýšili velikost populace a maximální počet iterací na 1000. Ostatní parametry zůstali zachované. Pro SOMU jsme zvolili stejné parametry jako v předešlé kapitole.

Klinická data byla měřena na myši s podkožně implantovaným tumorem. Tento nádor byl snímán animálním ultrazvukovým systémem Vevo 2100 na pracovišti Ústavu přístrojové techniky AV ČR v Brně. Jako kontrastní látka byl použit roztok Vevo MicroMarker v koncentraci doporučené výrobcem. Pro snímání byla použita lineární sonda s frekvencí 20 MHz. Samotné snímání probíhalo v režimu Bolust&Burst, kdy čas Burstu byl volen experimentálně po ustálení koncentrace látky v krevním oběhu. Data byla dále zpracována v prostředí Matlab.



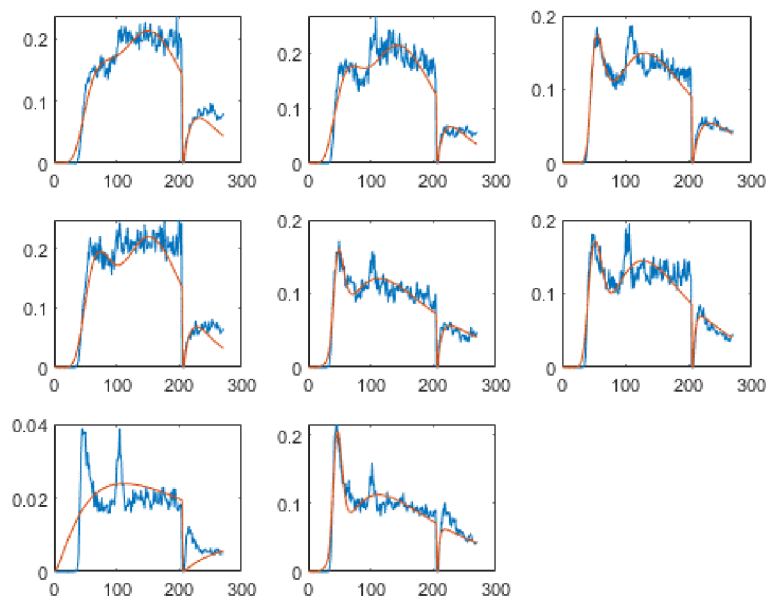
Obrázek 29: Snímek nádoru, který byl podkožně implantován myši.

Ukázky reálných dat pro první experiment a výsledné funkce vytvořené metodou PSO a SOMA jsou na Obrázek 30 a Obrázek 31. Ukázky pro experiment 2-3 jsou v příloze Příloha B -.

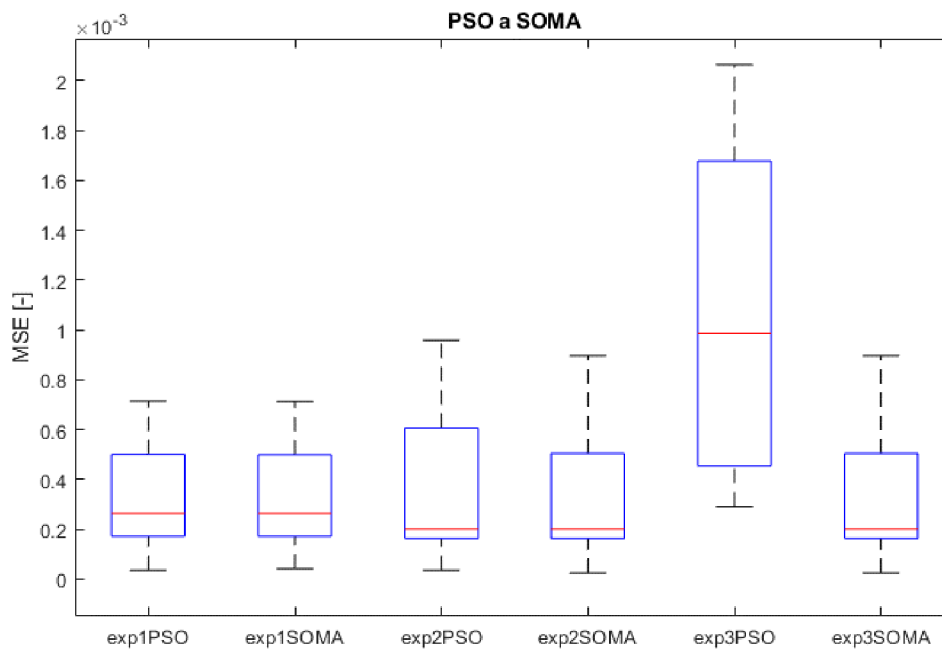


Obrázek 30: PSO exp1. Nejlepší proložení PSO pro 8 signálu experimentu 1.

Na Obrázek 30 vidíme proložení reálných dat z experimentu 1. Jedná se o nejlepší realizace z 10 spuštění běhu pro každou křivku z pohledu MSE. Pro metodu SOMA je na Obrázek 31. Lze vidět, že si obě metody neporadili se signálem 7, graf vlevo dole.

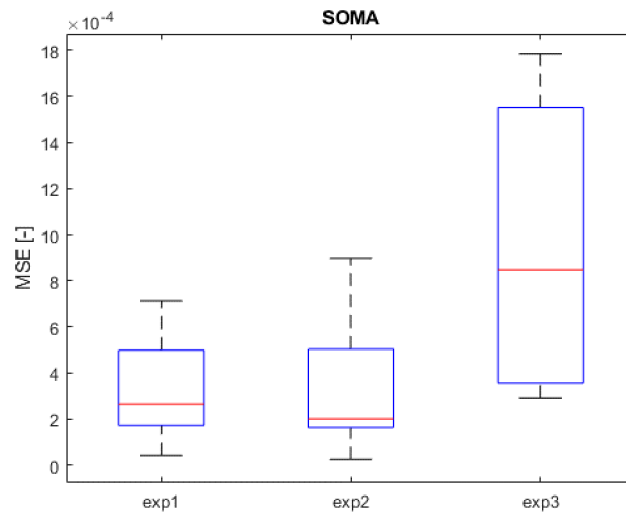


Obrázek 31: SOMA exp1. Nejlepší proložení SOMA pro 8 signálu experimentu 1.



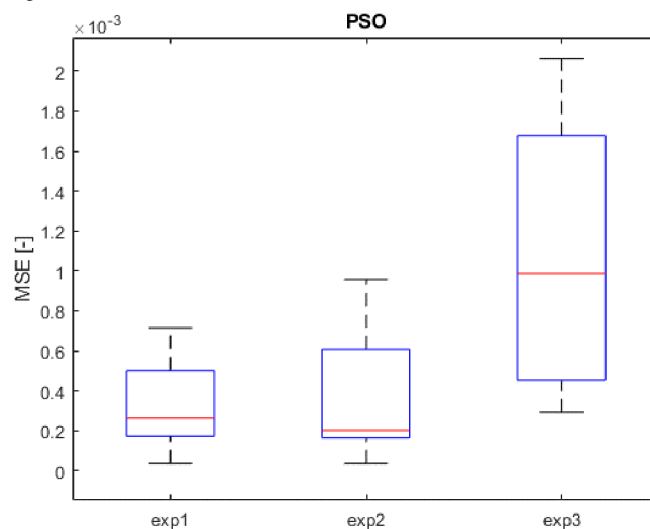
Obrázek 32: Srovnání hodnot MSE pro PSO a SOMA v proložení 8 křivek u experimentu 1-3.

Na Obrázek 32 vidíme srovnání MSE u metod PSO a SOMA u experimentu 1-3. Výkonnost obou metod v exp1 jsou téměř srovnatelné. Pro data exp2 je lepší metoda SOMA, která má menší rozptyl hodnot mezi 25-75 percentilem a hodnotu 75 percentilu než u PSO. U obou metod je hodnota mediánu u exp2 menší než u exp1, maximální hodnoty mají však větší u exp2. U exp3 ztelně převládá účinnost metody SOMA. U exp3 SOMA dosahuje menších hodnot minima a maxima a rozptylu hodnot, než u PSO také hodnota mediánu je u PSO řádově větší než u SOMA.

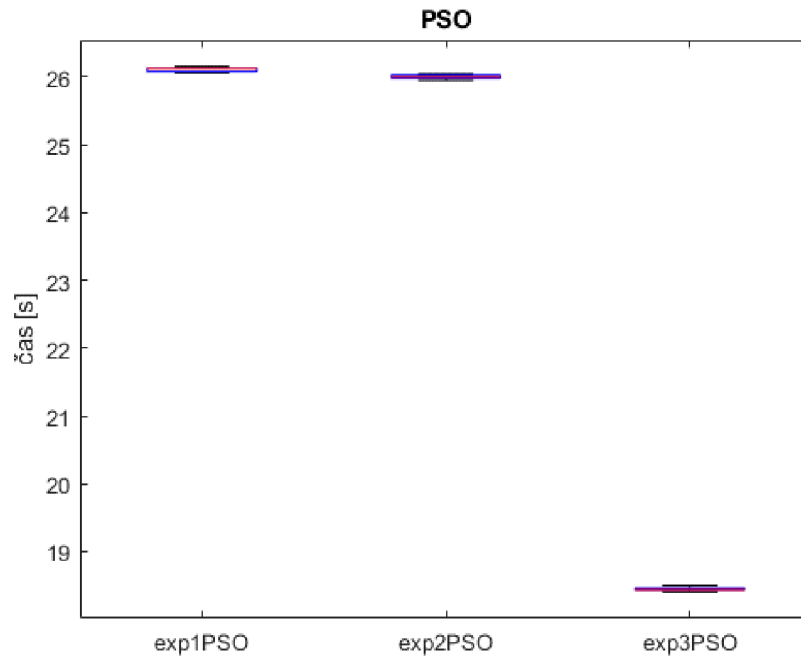


Obrázek 33: hodnoty MSE exp1-3 pro SOMA.

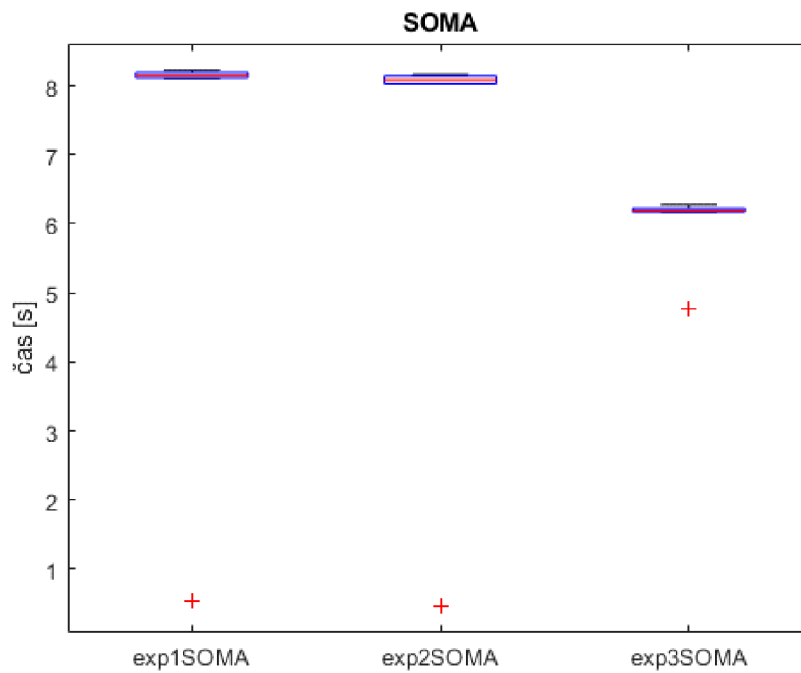
Na Obrázek 33 lze vidět hodnoty MSE pro exp 1-3 pouze pro SOMA. Hodnoty MSE pouze pro PSO lze vidět na Obrázek 34. Lze vidět, že hodnoty exp3 daleko převyšují hodnotou mediánu, rozptylu, maxima, minima, 25 a 75 percentilu než u exp. 1-2. Na Obrázek 35 a na Obrázek 36 lze vidět časovou náročnost výpočtu aproximace u obou metod a experimentů. Boxploty byly vypočítány z vektoru výpočetního času pro 8 křivek každého experimentu. Lze vidět, že PSO dosahuje hodnot kolem 26 sekund u exp1-2 a u exp3 dosahuje menších hodnot, kolem 18 sekund. Metoda SOMA konverguje daleko rychleji než PSO. Pro exp1 – exp2 dosahuje hodnot kolem 8 sekund. U exp3 je to kolem 6 sekund.



Obrázek 34: hodnoty MSE exp1-3 pro PSO.



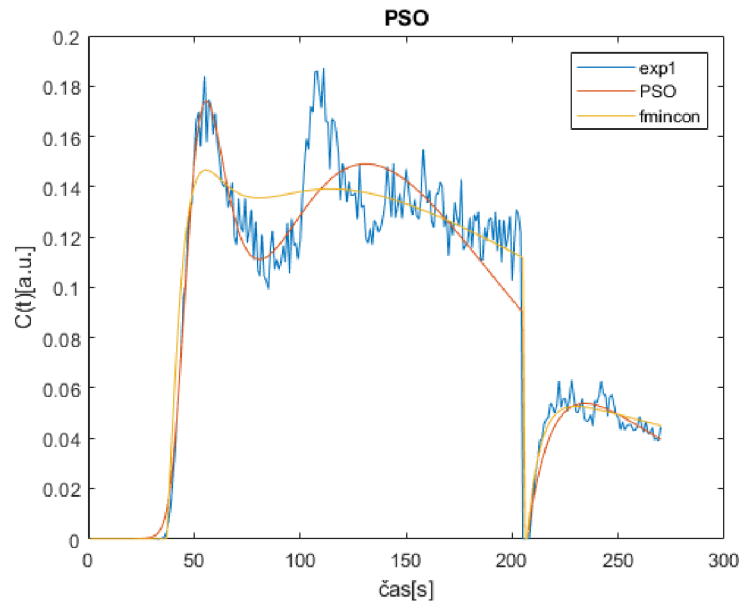
Obrázek 35: boxploty vektorů hodnot časů aproximace u metody PSO pro exp1 – exp3



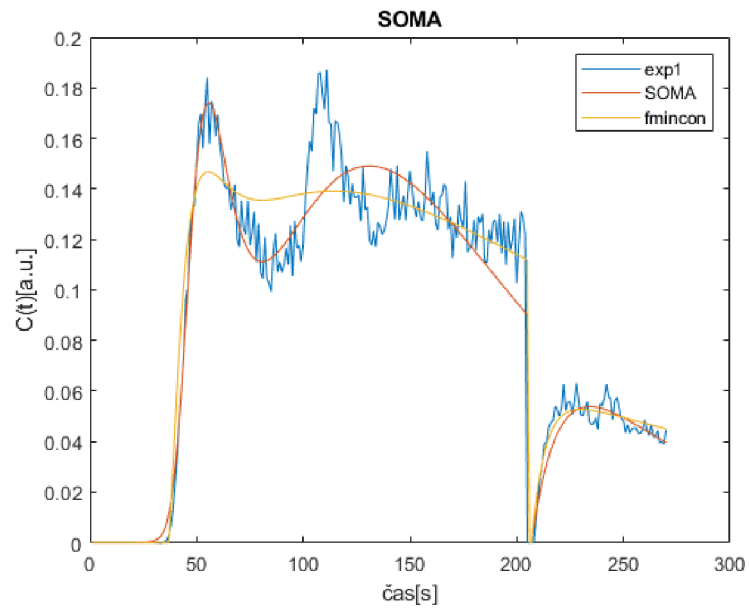
Obrázek 36: boxploty vektorů hodnot časů aproximace u metody SOMA pro exp1 – exp3

V poslední části jsme porovnali optimalizaci Matlab funkcí fmincon pro podmíněnou optimalizaci s PSO a SOMA. Vzhledem k tomu, že jsme data vzali z článku viz [43], kde tuto funkci použili k optimalizaci, vypočítali jsme hodnoty

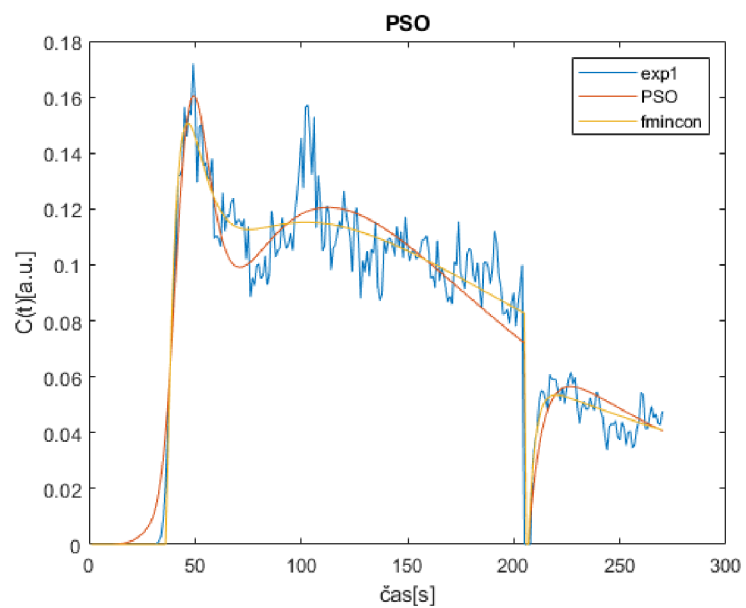
MSE mezi jejich vytvořenou funkcí pomocí fmincon a PSO a SOMA. Na obrázcích 37-40 jsou ukázky rozdílu aproximace a hodnota MSE u dvou křivek z experimentu 1. Pro PSO byla pro 3 křivku hodnota $MSE=1,25 \cdot 10^{-4}$ a pro 5 křivku hodnota $MSE=3,197 \cdot 10^{-4}$. Pro Metodu SOMA byla hodnota pro 3 křivku $MSE=2,0008 \cdot 10^{-4}$ a pro 5 křivku $MSE = 1,4430 \cdot 10^{-4}$.



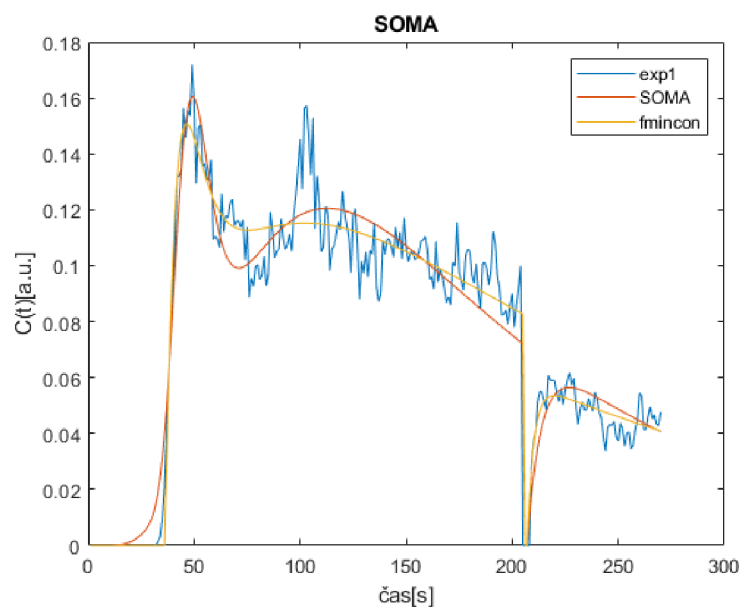
Obrázek 37: Srovnání metody fmincon a PSO u aproximace 3 křivky exp1, hodnota $MSE=1,25 \cdot 10^{-4}$



Obrázek 38: Srovnání metody fmincon a SOMA u aproximace 3 křivky exp1, hodnota $\text{MSE}=2,0008 \cdot 10^{-4}$



Obrázek 39: Srovnání metody fmincon a PSO u aproximace 5 křivky exp1, hodnota $\text{MSE}=3,197 \cdot 10^{-4}$



Obrázek 40: Srovnání metody fmincon a SOMA u aproximace 5 křivky exp1, hodnota $MSE=1,4430 \cdot 10^{-4}$

8. ZÁVĚR

Tato práce se zabývá principy ultrazvukové perfúzní analýzy a jednotlivými metodami pro stanovení perfúzních parametrů. V první kapitole byl popsán význam dilučních technik a jejich použití v kvantifikaci toku krve. Dále jsme nastínili význam dilučních křivek a možnost jejich aproximace. Následně byly popsány fyzikální a stochastické reprezentace hemodynamického modelu. Seznámili jsme se s vlastnostmi, chováním a významem ultrazvukových kontrastních látek.

Ve druhé kapitole jsme se zabývali teoretickými matematickými modely pro aproximaci dilučních křivek. Vysvětlily jsme si hlavní předpoklady Stewart-Hamiltonových vztahů pro výpočet průtoku krve a objemu pomocí AUC a MTT. Dále jsme popsali jednotlivé funkce sloužící k aproximaci dilučních křivek. Uvedli jsme princip a předpoklady difuzního driftového modelu.

Ve třetí kapitole jsme se zabývali metodou Bolust&Burst se kterou jsme dále pracovali ve druhé praktické části.

Kapitola čtyři se věnovala teoriím neevolučních technik optimalizace. Metodou náhodného prohledávání a Simplexovou metodou.

Pátá kapitola se zabývala teorií Evolučních algoritmu, konkrétně genetickým algoritmem.

Šestá kapitola uvedla rojové algoritmy, menší kapitolku věnovala optimalizaci kolonií mravenců (ACO). Více se však zabývala algoritmy SOMA a PSO se kterými jsme pracovali ve druhé praktické části.

Sedmá kapitola se věnovala praktické části. Praktická část byla rozdělena na dvě části. V první části jsme pomocí nástroje Matlab Simulink vytvořili simulaci průběhu tvaru diluční křivky po intravenózní aplikaci bolusové injekce. Vytvořili jsme dvě diluční křivky, kdy jedna křivka byla více strmá než druhá. Provedli jsme proložení křivky pomocí nástroje Curve fitting z Optimization toolbox v Matlabu, který využívá k proložení metodu nejmenších čtverců. Provedli jsme proložení simulované křivky metodou nejmenších čtverců a metodou spojitého genetického algoritmu čtyřmi funkcemi LDRW, FPT, gama a lognormální. Dále jsme se zabývali vlivem šumu na výkon obou metod. Simulovaný signál byl rušen třemi hodnotami šumu a byl zkoumán vliv na výkon algoritmů. Zvyšující se hodnoty SNR měly negativní vliv na optimalizaci u obou metod. Ve druhé praktické části jsme se zabývali metodou Bolust&Burst. Simulovaná data jsme optimalizovali pomocí metody spojitého genetického algoritmu, pomocí metody PSO a SOMA. Průběh AIF byl vytvořen pomocí parametrického vyjádření jako součet dvou lognormálních funkcí. Došli jsme k názoru, že metody SOMA a PSO převyšují svým výkonem metodu spojitého GA. Ve druhé části jsme aproximovali reálná data ze tří experimentů metodou SOMA a PSO. Nejlepší hodnoty výkonu metod jsme dostali v experimentu 1. Algoritmus SOMA byl rychlejší než PSO. Výkon PSO u experimentu 3 byl řádově horší než u SOMA a než u experimentů 1-2.

LITERATURA

- [1] STROUTHOS, Costas, et al. Indicator dilution models for the quantification of microvascular blood flow with bolus administration of ultrasound contrast agents. *IEEE transactions on ultrasonics, ferroelectrics, and frequency control*, 2010, 57.6: 1296-1310.
- [2] MISCHI, M.; KALKER, A. A. C. M.; KORSTEN, H. H. M. Moment method for the local density random walk model interpolation of ultrasound contrast agent dilution curves. *tC*, 2004, 2.2: 2.
- [3] Sirsi S, Borden M. Microbubble Compositions, Properties and Biomedical Applications. *Bubble Sci Eng Technol.* 2009;1(1-2):3–17. doi:10.1179/175889709X446507
- [4] GOLDBERG, Barry B.; LIU, Ji-Bin; FORSBERG, Flemming. Ultrasound contrast agents: a review. *Ultrasound in medicine & biology*, 1994, 20.4: 319333.
- [5] KARSHAFIAN, Raffi; BURNS, Peter N.; HENKELMAN, Mark R. Transit time kinetics in ordered and disordered vascular trees. *Physics in Medicine & Biology*, 2003, 48.19: 3225.
- [6] Prof. MUDr. Pavel Klener, DrSc. Angiogeneze a nádorová onemocnění. *REMEDIA* [online]. 1/2002 [cit. 2020-01-31]. Dostupné z: <http://www.remedia.cz/Clanky/Editorial/Angiogeneze-a-nadorovaonemocneni/6-M-fa.magarticle.aspx>
- [7] THOMPSON JR, Howard K., et al. Indicator transit time considered as a gamma variate. *Circulation Research*, 1964, 14.6: 502-515
- [8] MISCHI, M.; DEN BOER, J. A.; KORSTEN, H. H. M. On the physical and stochastic representation of an indicator dilution curve as a gamma variate. *Physiological measurement*, 2008, 29.3: 281.
- [9] DEB, Kalyanmoy. *Multi-objective optimization using evolutionary algorithms*. John Wiley & Sons, 2001.
- [10] KACHITVICHYANUKUL, Voratas. Comparison of three evolutionary algorithms: GA, PSO, and DE. *Industrial Engineering and Management Systems*, 2012, 11.3: 215-223.
- [11] KEPRT, Aleš a Václav SNÁŠEL. *Genetické algoritmy pro redukci dimenze a analýzu binárních dat* [online]. 708 33, Ostrava-Poruba, , 2015, 17. listopadu, , 1-8,
- [12] PENCHEVA, Tania; ATANASSOV, K.; SHANNON, A. Modelling of a roulette wheel selection operator in genetic algorithms using generalized nets. *Int. J. Bioautomation*, 2009, 13.4: 257-264.
- [13] POLI, Riccardo; KENNEDY, James; BLACKWELL, Tim. Particle swarm optimization. *Swarm intelligence*, 2007, 1.1: 33-57.
- [14] QIN, A. Kai; SUGANTHAN, Ponnuthurai N. Self-adaptive differential evolution algorithm for numerical optimization. In: 2005 IEEE congress on evolutionary computation. IEEE, 2005. p. 1785-1791.

- [15] VOLNÁ, Eva. Evoluční algoritmy a neuronové sítě. Ostrava: Ostravská univerzita v Ostravě. Dostupné z: http://physics.ujep.cz/~mmaly/vyuka/MPVT_II/Heuristiky/Evolucni_algoritmy_a_neuronove_siteSOMA+Genetika.pdf, 2012.
- [16] SHI, Yuhui; EBERHART, Russell C. Empirical study of particle swarm optimization. In: *Proceedings of the 1999 congress on evolutionary computation-CEC99 (Cat. No. 99TH8406)*. IEEE, 1999. p. 1945-1950.
- [17] MARINI, Federico; WALCZAK, Beata. Particle swarm optimization (PSO). A tutorial. *Chemometrics and Intelligent Laboratory Systems*, 2015, 149: 153-165.
- [18] HAUPT, Randy L a S. E HAUPT. Practical genetic algorithms. 2nd ed. Hoboken, N.J.: John Wiley, c2004. ISBN 9780471455653.
- [19] DOS SANTOS COELHO, Leandro; ALOTTO, Piergiorgio. Electromagnetic optimization using a cultural self-organizing migrating algorithm approach based on normative knowledge. *IEEE Transactions on Magnetics*, 2009, 45.3: 1446-1449.
- [20] BAO, Diep Quoc; ZELINKA, Ivan. Obstacle avoidance for swarm robot based on self-organizing migrating algorithm. *Procedia Computer Science*, 2019, 150: 425-432
- [21] ZELINKA, Ivan. SOMA—self-organizing migrating algorithm. In: *New optimization techniques in engineering*. Springer, Berlin, Heidelberg, 2004. p. 167-217.
- [22] Diep, Q., Zelinka, I. and Das, S. 2019. Self-Organizing Migrating Algorithm Pareto. *MENDEL*. 25, 1 (Jun. 2019), 111-120. DOI:<https://doi.org/10.13164/mendel.2019.1.111>.
- [23] ROBINSON, Jacob; RAHMAT-SAMII, Yahya. Particle swarm optimization in electromagnetics. *IEEE transactions on antennas and propagation*, 2004, 52.2: 397-407.
- [24] TVRDÍK, Josef. Evoluční algoritmy. Skripta, Přírodovědecká fakulta, Ostravská univerzita, 2004.
- [25] OSPÍCHAL, Jiří; KVASNIČKA, Vladimír. Genetické algoritmy: nic pro biology. *Pokroky matematiky, fyziky a astronomie*, 1995, 40.1: 1-10.
- [26] LAGARIAS, Jeffrey C., et al. Convergence properties of the Nelder--Mead simplex method in low dimensions. *SIAM Journal on optimization*, 1998, 9.1: 112-147.
- [27] GAO, Fuchang; HAN, Lixing. Implementing the Nelder-Mead simplex algorithm with adaptive parameters. *Computational Optimization and Applications*, 2012, 51.1: 259-277.
- [28] MCKINNON, Ken IM. Convergence of the Nelder--Mead Simplex Method to a Nonstationary Point. *SIAM Journal on optimization*, 1998, 9.1: 148-158.
- [29] KEPRT, Aleš a Václav SNÁŠEL. *Genetické algoritmy pro redukci dimenze a analýzu binárních dat* [online]. 708 33, Ostrava-Poruba, 2015, 17. listopadu, 1-8,

- [30] MECHL, M., V VÁLEK, A ŠPRLÁKOVÁ-PUKOVÁ, Š BOHATÁ a J FOUKAL. *Ultrazvukové vyšetření s kontrastní látkou – indikace a klinické zkušenosti: Přehledné sdělení* [online]. Radiologická klinika LF MU a FN Brno, 2012, 5-9 [cit. 2020-01-31].
- [31] FERDA, Jiří, et al. Diagnostické zobrazení mikrosvěta a makrosvěta nádorů. *Ces Radiol*, 2010, 64.Suppl 1: 7-23.
- [32] LEONDES, Cornelius T. *Biomechanical Systems Technology: Cardiovascular Systems*. World Scientific, 2007.
- [33] HORN, Jeffrey; NAFPLIOTIS, Nicholas; GOLDBERG, David E. A niched Pareto genetic algorithm for multiobjective optimization. In: *Proceedings of the first IEEE conference on evolutionary computation. IEEE world congress on computational intelligence*. Ieee, 1994. p. 82-87.
- [34] HONG, Tzung-Pei; WANG, Hong-Shung; CHEN, Wei-Chou. Simultaneously applying multiple mutation operators in genetic algorithms. *Journal of heuristics*, 2000, 6.4: 439-455.
- [35] WANG, Sun-Chong. Genetic algorithm. In: *Interdisciplinary Computing in Java Programming*. Springer, Boston, MA, 2003. p. 101-116.
- [36] CHANG, Wei-Der. Coefficient estimation of IIR filter by a multiple crossover genetic algorithm. *Computers & Mathematics with Applications*, 2006, 51.9-10: 1437-1444.
- [37] WRIGHT, Alden H. Genetic algorithms for real parameter optimization. In: *Foundations of genetic algorithms*. Elsevier, 1991. p. 205-218.
- [38] DOSTÁL, ZDENĚK; BERMELIJSKI, P. Metody optimalizace. VŠB-TU Ostrava a ZČU v Plzni. Dostupné z WWW: http://mi21.vsb.cz/sites/mi21.vsb.cz/files/unit/metody_optimalizace.pdf, 2012.
- [39] RNDr. Marie Budíková, Dr a kol. Logaritnicko normální rozdělení. *Statistika a pravděpodobnost: Přírodovědecká fakulta Masarykovy univerzity* [online]. [cit. 2020-01-31]. Dostupné z: <https://is.muni.cz/do/rect/el/estud/prif/ps15/statistika/web/pages/logaritnicko-normalni.html>
- [40] SKANDEROVA, Lenka; FABIAN, Tomas; ZELINKA, Ivan. Self-adapting self-organizing migrating algorithm. *Swarm and Evolutionary Computation*, 2019, 51: 100593.
- [41] JIŘÍK, Radovan, et al. Parametric ultrasound perfusion analysis combining bolus tracking and replenishment. In: *2012 IEEE International Ultrasonics Symposium*. IEEE, 2012. p. 1323-1326
- [42] JIRIK, Radovan, et al. Ultrasound perfusion analysis combining bolus-tracking and burst-replenishment. *IEEE transactions on ultrasonics, ferroelectrics, and frequency control*, 2013, 60.2: 310-319.
- [43] MÉZL, M a R JIŘÍK. *Srovnání optimalizačních algoritmů pro ultrazvukovou perfúzní analýzu metodou Bolust & Burst*. Ústav biomedicínského inženýrství, VUT v Brně, Ústav přístrojové techniky AV ČR, v.v.i., 5.

SEZNAM SYMBOLŮ A ZKRATEK

$C(t)$	funkce koncentrace indikátoru
IDC	Diluční křivka
MTT	Střední doba přepravy
SNR	Poměr signál vůči šumu
$LDRW$	teoretický model založený na řešení difuze s driftovou rovnicí
Q	průtok
V	objem
x_i	místo vpichu indikátoru
P	pravděpodobnost
x_m	místo měření koncentrace indikátoru
V_c	objem jednoho kompartmentu
A	plocha průřezu trubice
Δx	specifický úsek na ose x
Δt	časový interval
p	pravděpodobnost přechodu částice
$1-p$	pravděpodobnost toho, že částice zůstává v kompartmentu
λ	konstanta Poissonova rozdělení, kde $\lambda=np$
MR	magnetická rezonance
ROI	křivka intenzity obrazu jako funkce času
I	intenzita
AUC	plocha pod křivkou časové intenzity
t_p	čas dosažení maximální intenzity
F	průtok krve
m	množství indikátoru v injekci
μ	průměr
σ	standardní směrodatná odchylka
I_0	počáteční intenzita
t_0	čas příchodu bolusu
$\delta(t)$	Diracova delta funkce
β	parametr rychlosti
J_{bolus}	člen s maximální věrohodností bolusové části
J_{repl}	člen s maximální věrohodností replenishment části
$L(n)$	Laplacianův operátor
T_s	vzorkovací perioda
\mathbb{R}	realná čísla
α	reflexe
β	expanze
γ	kontrakce
δ	zmenšení
m	těžiště
\bar{x}	těžiště
x_r	bod reflexe
x_e	bod expanze

$\Gamma(x)$	funkce gama
!	faktoriál
S	náhodná proměnná
T	časový interval
$X(nt)$	pozice částice za čas nt
D	difuzní koeficient
v	rychlost toku krve
λ^{-1}	koeficient šikmosti
FPT	typ difuzního driftového modelu
x^*	globální minimum
x	souřadnice
n	počet pokusu, počet kompartmentů
$f_c(t)$	křivka aproximace Poissonovým rozdělením
$N(\mu, \sigma^2)$	normální rozdělení
$C_i(t)$	koncentrace indikátoru v kompartmentu i
$I(i=1, \dots, n)$	kompartiment
$2/\sqrt{n}$	šikmost distribuce
α	parametr tvaru
β	parametr měřítka
$A\Gamma$	amplituda diluční křivky
$\Gamma(\cdot)$	operátor Gamma
C_Γ	Gamma variační model
$\Gamma(x)$	operátor Gamma
!	faktoriál
A	sekce trubice
$C(x,t)$	koncentrační křivka
D	difuzní koeficient
v	rychlost toku krve
$t\alpha$	rozptyl
x_0	vzdálenost mezi počátkem a koncem křivky
λ^{-1}	koeficient šikmosti křivky
F_b	průtok krve
V_b	objem krve
MI	mechanický index
ROI	oblast zájmu
A	maximum křivky
β	časová konstanta
TRF	tkáňová reziduální funkce
$R(t)$	bezrozměrná veličina tkáňového zbytku
AIF(t)	arteriální vstupní funkce
PET	pozitronová emisní tomografie

SPECT	jednofotonová emisní tomografie
CT	počítačová tomografie
MRI	magnetická rezonance
$C_{\text{bolus}}(t)$	koncentrační křivka bolusové části
*	konvoluce
$C_{\text{repl}}(t)$	koncentrační křivka fáze replenishment
$H(t)$	Heavisideova (jednotková) funkce
x_{oc}	bod vnější kontrakce
x_{ic}	vnitřní kontrakce
X	bitový řetězec
P_j	pravděpodobnost výběru
F_i	individuální kvalita jedince
N	velikost populace
g	dělící bod
O_1	první rodič
O_2	druhý rodič
O'_1	první potomek
O'_2	druhý potomek
O'_3	třetí potomek
p_c	pravděpodobnost křížení
ACO	optimalizace kolonií mravenců
PSO	optimalizace rojem částic
D	počet parametrů, dimenzí prostoru
$x_{\text{d,min}}$	dolní hranice
$x_{\text{d,max}}$	horní hranice
U	uniformní rozdělení
v	rychlost
p_{best_i}	nejlepší pozice částice
t	iterace
V_i	vektor rychlostí
G_i	vektor globálních nejlepších řešení
c_1	konstanta zrychlení, kognitivní koeficient
c_2	konstanta zrychlení, sociální koeficient
$r_1 r_2$	náhodná čísla
ω	interní váha
ω_{max}	maximální hodnota interní váhy
ω_{min}	minimální hodnota interní váhy
GA	genetický algoritmus
$x_j^{(l_0)}$	dolní hranice

$x_j^{(hi)}$	horní hranice
PV	perturbační vektor
LN	lognormální rozdělení

SEZNAM PŘÍLOH

PŘÍLOHA A - MĚŘENÉ HODNOTY PARAMETRŮ	85
PŘÍLOHA B - NAMĚŘENÉ GRAFY	86
PŘÍLOHA C - BOXPLOTY	88
PŘÍLOHA D - CD.....	89

Příloha A - Měřené hodnoty parametrů

A.1 Tabulka: Hodnoty minima, maxima, mediánu, 25 percentilu, 75 percentilu a rozsah hodnot mezi 25 a 75 percentilem MSE pro GA, PSO a SOMA pro SNR 10-30 dB

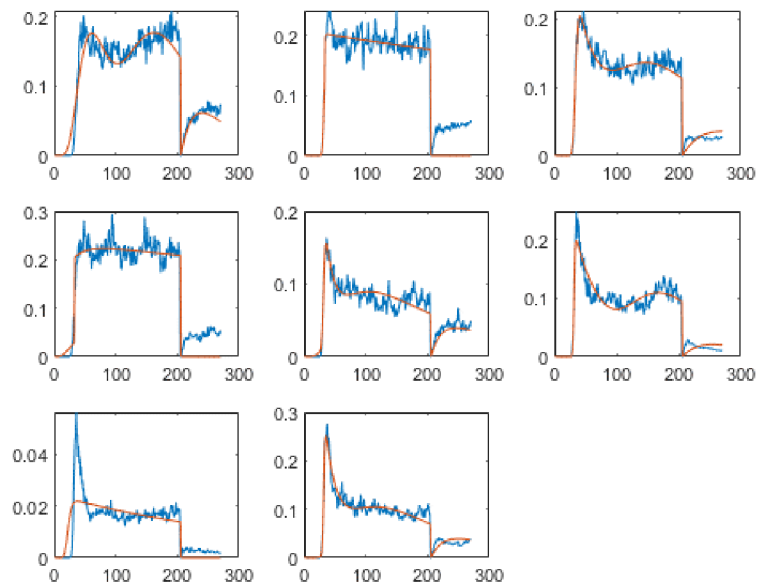
MSE	SNR=10 dB			SNR=20dB			SNR=30dB		
	GA	PSO	SOM	GA	PSO	SOM	GA	PSO	SOM
	[10 ⁻⁵]	[10 ⁻⁸]	[10 ⁻⁹]	[10 ⁻⁵]	[10 ⁻⁸]	[10 ⁻⁸]	[10 ⁻⁵]	[10 ⁻⁷]	[10 ⁻⁷]
min	0,54	0,18	0,81	0,56	0,72	0,94	0,78	0,57	0,59
max	4,55	26,61	10,26	4,95	25,77	9,14	4,54	11,55	11,71
Media	3,01	1,33	3,97	2,87	5,07	4,13	2,96	4,33	4,21
25per.	2,29	0,77	2,90	2,17	3,50	3,00	2,13	3,30	2,98
75per.	3,70	2,97	6,07	3,55	8,06	5,88	3,57	6,03	6,25
Rozs.	1,41	2,20	3,18	1,38	4,56	2,88	1,43	2,73	3,27

A.2 Hodnoty minima, maxima, mediánu, 25 percentilu, 75 percentilu a rozsah hodnot mezi 25 a 75 percentilem MSE pro GA, PSO a SOMA pro SNR 40-50 dB

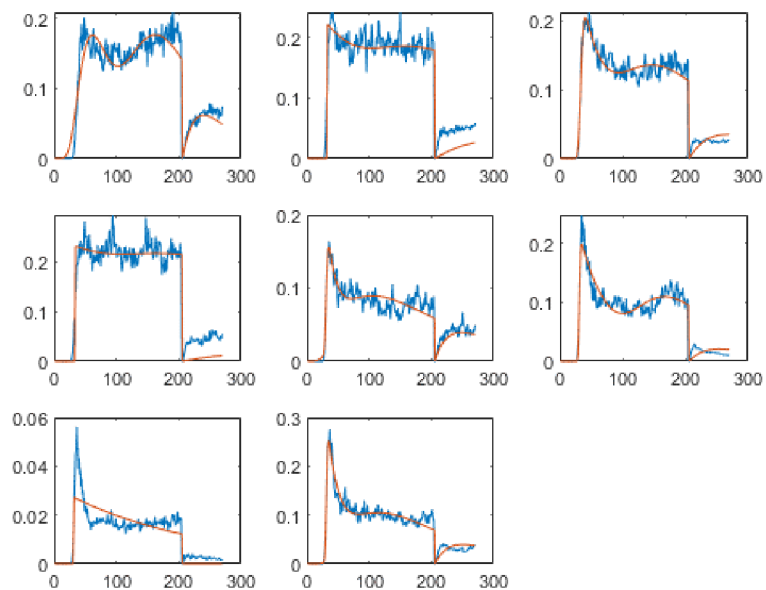
MSE	SNR=40 dB			SNR=50 dB		
	GA	PSO	SOMA	GA	PSO	SOMA
	[10 ⁻⁵]	[10 ⁻⁶]	[10 ⁻⁶]	[10 ⁻⁵]	[10 ⁻⁶]	[10 ⁻⁵]
minimum	0,32	0,83	0,93	2,20	8,32	0,94
maximum	4,87	12,01	19,95	13,0	105,80	10,93
medián	2,32	3,87	4,41	4,92	46,29	4,60
25 percen.	1,78	2,69	2,70	4,02	31,78	3,23
75 percen.	3,11	5,07	6,38	6,81	64,26	6,42
rozsah	1,34	2,38	3,67	2,79	32,48	3,20

Příloha B - Naměřené grafy

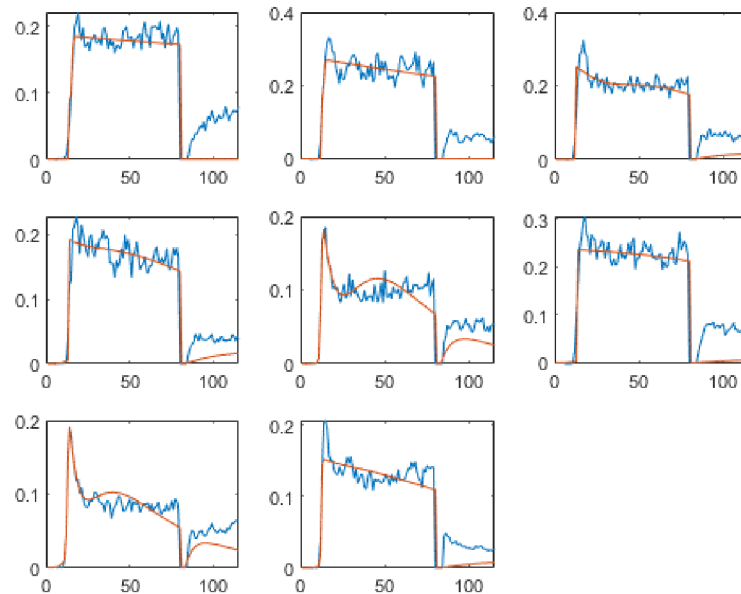
B.1 Klinická data z experimentu 2. a aproximace metodou PSO



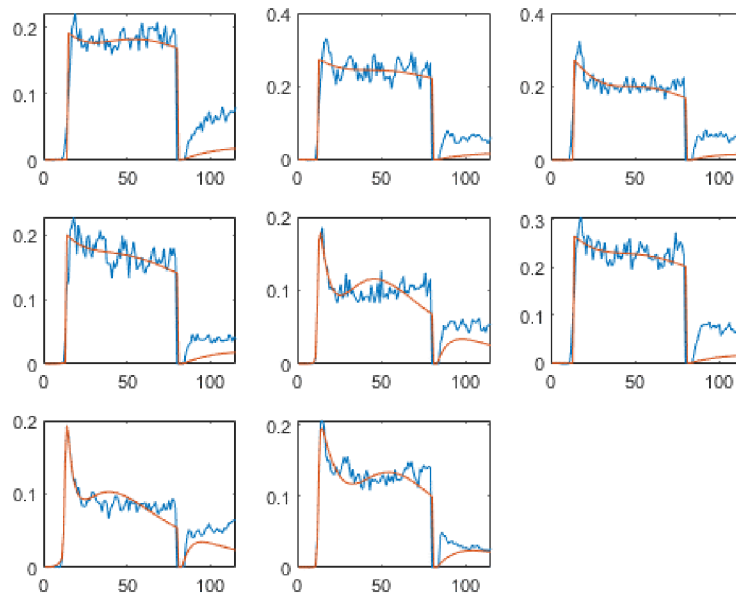
B.2 Klinická data z experimentu 2 a aproximace metodou SOMA



B.3 Klinická data z experimentu 3 aproximace metodou PSO

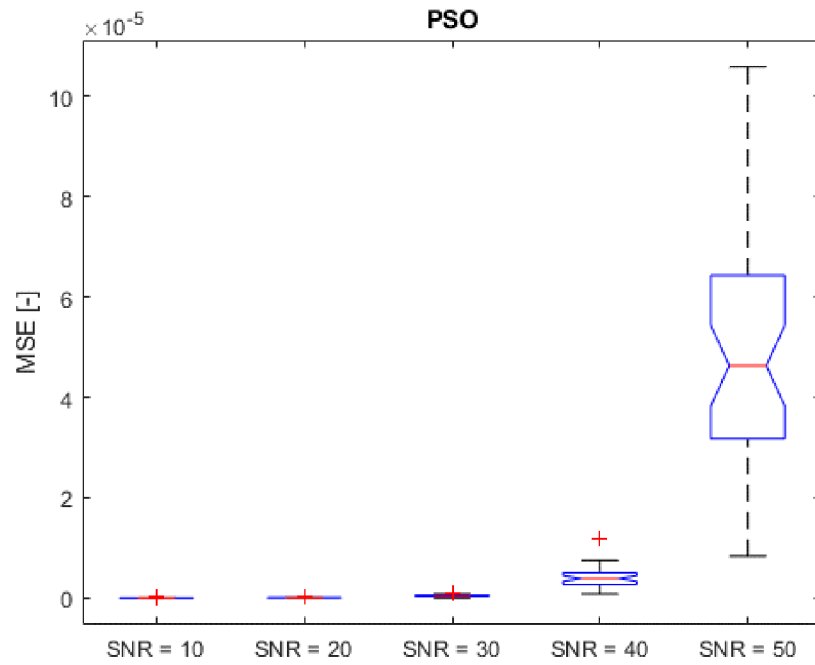


B.4 Klinická data z experimentu 3 aproximace metodou SOMA



Příloha C - Boxploty

C.1 Boxplot hodnoty PSO, MSE pro SNR= 10–50 dB pro Simulovaná data



Příloha D - CD

Příložené CD obsahuje:

- Složka – první část:
 - Simulace (Simulink Model)
 - repr (MATLAB Code)
 - mutace_s (MATLAB Code)
 - spojity GA (MATLAB Code)
 - funkceprolozeni (MATLAB Code)
 - funkce1.mat
 - funkce2.mat
- mutace_s (MATLAB Code)
- reprodukce (MATLAB Code)
- us_perfan_v031GA (MATLAB Code)
- us_perfan_v031pso (MATLAB Code)
- us_perfan_v031soma (MATLAB Code)