VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

DIPLOMOVÁ PRÁCE

Brno, 2022 Bc. Tomáš Votýpka

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

VYUŽITÍ EVOLUČNÍCH ALGORITMŮ PRO REGISTRACI DYNAMICKÝCH ULTRAZVUKOVÝCH SEKVENCÍ

EVOLUTIONARY ALGORITHMS FOR IMAGE REGISTRATION OF DYNAMIC ULTRASOUND SEQUENCES

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. Tomáš Votýpka

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Martin Mézl, Ph.D.

BRNO 2022

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

Diplomová práce

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

Ústav biomedicínského inženýrství

Student: Bc. Tomáš Votýpka ***ID:*** 203691

Ročník: 2 ***Akademický rok:*** 2021/22

NÁZEV TÉMATU:

Využití evolučních algoritmů pro registraci dynamických ultrazvukových sekvencí

POKYNY PRO VYPRACOVÁNÍ:

1) Seznamte se ze základy ultrazvukového zobrazování s využitím kontrastních látek. Proveďte rešerši metod registrace obrazu takovýchto sekvencí. 2) Seznamte se s evolučními algoritmy pro matematickou optimalizaci. Proveďte rešerši využití těchto metod pro registraci obrazu. 3) Navrhněte postup pro registraci dynamických ultrazvukových sekvencí. Implementujte vybranou kriteriální funkci a alespoň jednu optimalizační metodu. Výslednou implementaci otestujte na testovacích sekvencích. 4) Práci rozšiřte o implementaci alespoň dvou evolučních algoritmů. 5) Navržené algoritmy otestujte na dynamických ultrazvukových sekvencích, které budou poskytnuty vedoucím práce. 6) Proveďte vyhodnocení úspěšnosti registrace pomocí jednotlivých metod ve srovnání s manuální korekcí pohybu nebo některým z automatických přístupů.

DOPORUČENÁ LITERATURA:

[1] DAMAS, S., O. CORDÓN a J. SANTAMARÍA. Medical Image Registration Using Evolutionary Computation: An Experimental Survey. IEEE Computational Intelligence Magazine. 2011, 6(4): 26-42. ISSN: 1556-603X.

[2] HAUPT, Randy L a Witold PEDRYCZ. Practical genetic algorithms: a comprehensive framework of computational intelligence. 2nd ed. Hoboken, N.J.: Wiley-Interscience, 2004, xvii, 253 s. ISBN 978-0-471-45565-3.

Termín zadání: 7.2.2022 ***Termín odevzdání:*** 20.5.2022

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

Diplomová práce se zabývá registrací dynamických ultrazvukových sekvencí s využitím evolučních algoritmů. Tato práce teoreticky popisuje ultrazvukové zobrazování, proces registrace obrazů a optimalizaci pomocí optimalizačních a evolučních algoritmů. V praktické části práce je popsána realizace několika optimalizačních metod, které byly implementované v programovém prostředí MATLAB.

KLÍČOVÁ SLOVA

Ultrazvukové zobrazování, registrace obrazů, rigidní transformace obrazu, optimalizační algoritmy, evoluční algoritmy, genetické algoritmy, rojové algoritmy, MATLAB

ABSTRACT

Diploma thesis deals with the registration of of dynamic ultrasound sequences using evolutionary algorithms. This work theoretically describes ultrasound imaging, the process of image registration and optimization using optimization and evolutionary algorithms. The practical part of the work describes the implementation of several optimization methods that were implemented in the MATLAB software environment.

KEYWORDS

Ultrasound imaging, image registration, rigid image transformation, optimization algorithms, evolutionary algorithms, genetic algorithms, swarm algorithms, MATLAB

Vysázeno pomocí balíčku `thesis` verze 4.07; `http://latex.feec.vutbr.cz`

ROZŠÍŘENÝ ABSTRAKT

Výtah ze směrnice rektora 72/2017:

Bakalářská a diplomová práce předložená v angličtině musí obsahovat rozšířený abstrakt v češtině nebo slovenštině (čl. 15). To se netýká studentů, kteří studují studijní program akreditovaný v angličtině. (čl. 3, par. 7)

Nebude-li vnitřní normou stanoveno jinak, doporučuje se rozšířený abstrakt o rozsahu přibližně 3 normostrany, který bude obsahovat úvod, popis řešení a shrnutí a zhodnocení výsledků. (čl. 15, par. 5)

VOTÝPKA, Tomáš. *Využití evolučních algoritmů pro registraci dynamických ultrazvukových sekvencí*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav biomedicínského inženýrství, 2022, 95 s. Diplomová práce. Vedoucí práce: Ing. Martin Mézl, Ph.D.

Prohlášení autora o původnosti díla

Jméno a příjmení autora:	Bc. Tomáš Votýpka
VUT ID autora:	203691
Typ práce:	Diplomová práce
Akademický rok:	2021/22
Téma závěrečné práce:	Využití evolučních algoritmů pro registraci dynamických ultrazvukových sekvencí

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/nebo majetkových a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno ....

podpis autora*

*Autor podepisuje pouze v tištěné verzi.

PODĚKOVÁNÍ

Rád bych poděkoval vedoucímu diplomové práce panu Ing. Martinu Mézlovi, Ph.D. za odborné vedení, konzultace, trpělivost, ochotu a podnětné návrhy k práci.

Obsah

Úvod **15**

1 Ultrazvukové zobrazování **16**
1.1 Ultrazvuk v lékařství 16
1.2 Fyzika ultrazvuku 17
1.2.1 Vznik ultrazvuku 17
1.3 Módy zobrazení 18
1.4 Zobrazení s kontrastními látkami 18

2 Registrace obrazu **20**
2.1 Lícování („registrace“) obrazů 20
2.1.1 Podmínky transformací při lícování 21
2.1.2 Optimalizační proces lícování 21
2.2 Geometrické transformace obrazů 22
2.2.1 Tuhé (rigidní) transformace 23
2.2.2 Pružné (flexibilní) transformace obrazu 24
2.2.3 Transformace obrazu po částech 28
2.2.4 Problémy při užití geometrických transformací 29
2.3 Stanovení parametrů transformace 30
2.3.1 Analytické odvození 30
2.3.2 Zvolení typu funkce a vypočtení parametrů 30
2.4 Kritéria podobnosti obrazů 31
2.4.1 Kritérium podobnosti založené vzdálenosti 31
2.4.2 Kritéria podobnosti založená na intenzitě 31
2.4.3 Kritéria založená na informaci v obraze 33
2.4.4 Kritéria slícování podle význačných bodů 33

3 Evoluční algoritmy pro registraci obrazů **35**
3.1 Problém globální optimalizace 35
3.1.1 Formulace problému globální optimalizace 35
3.1.2 Deterministické a stochastické algoritmy 36
3.2 Optimalizace vycházející z matematické analýzy 37
3.2.1 Metoda nejstrmějšího sestupu 37
3.2.2 Newtonova metoda 38
3.3 Optimalizace bez derivací 39
3.3.1 Náhodné prohledávání 39
3.3.2 Nelder-Meadův algoritmus 39

3.3.3 Simulované žíhání . . . 41
3.4 Evoluční algoritmy . . . 42
3.4.1 Genetické algoritmy . . . 42
3.5 Rojové algoritmy . . . 44
3.5.1 Particle Swarm Optimalizace (PSO) . . . 44
3.5.2 Včelí algoritmus (BA) . . . 45
3.5.3 Algoritmus umělé včelí kolonie (ABC) . . . 46
3.5.4 Světluškový algoritmus (FA) . . . 47

4 Registrace sekvence simulované scény 49
4.1 Základní informace o sekvenci . . . 49
4.2 Předzpracování a transformace obrazu . . . 51
4.2.1 Transformace obrazu . . . 51
4.2.2 Vytvoření nové sekvence snímků . . . 51
4.3 Výběr registračního přístupu . . . 52
4.3.1 Výběr optimálních parametrů a vyhodnocení sekvence . . . 53
4.4 Volba kriteriální funkce . . . 54
4.5 Registrace s optimalizací spojitého genetického algoritmu (GA) . . . 55
4.5.1 Stanovení parametrů algoritmu spojitého GA . . . 55
4.6 Registrace s optimalizací PSO . . . 59
4.6.1 Stanovení parametrů algoritmu PSO . . . 60
4.7 Registrace s optimalizací včelího algoritmu (BA) . . . 63
4.7.1 Stanovení parametrů algoritmu BA . . . 64
4.8 Registrace s optimalizací umělé včelí kolonie (ABC) . . . 67
4.8.1 Stanovení parametrů algoritmu ABC . . . 67
4.9 Registrace s optimalizací světluškového algoritmu (FA) . . . 70
4.9.1 Stanovení parametrů algoritmu FA . . . 71
4.10 Srovnání optimalizačních metod na umělé sekvenci . . . 73

5 Registrace klinické sekvence nádoru pankreatu 79
5.1 Informace o kontrastu a převzorkování sekvence . . . 79
5.2 Srovnání optimalizačních metod na první klinické sekvenci . . . 80

6 Registrace klinické sekvence tenkého střeva 84
6.1 Informace o kontrastu a převzorkování sekvence . . . 84
6.2 Srovnání optimalizačních metod na druhé klinické sekvenci . . . 85

Závěr 87

Literatura 89

Seznam příloh **94**

A Obsah CD **95**

Seznam obrázků

2.1 Rigidní transformace: původní obrázek, posunutý a orotovaný obrázek, [7] . . . 24
2.2 Změna měřítka: vodorovné a vertikální zvětšení obrázku, [7] . . . 25
2.3 Rotace, [7] . . . 26
2.4 Příklad afinní transformace, [7] . . . 27
2.5 Geometrická transformace v trojúhelníkové síti: korespondující sady kontrolních bodů, [7] . . . 29
3.1 Simplexy v dvourozměrném a třírozměrném prostoru, [17] . . . 40
3.2 Vývojový diagram GA, [18] . . . 43
4.1 Snímek ze sekvence s výřezem oblasti zájmu . . . 49
4.2 Srovnání množství kontrastní látky v obdélníkovém výřezu prvního a patnáctého snímku . . . 50
4.3 Průběh koncentrace kontrastní látky napříč sekvencí snímků, ekvivalentní k průměrnému jasu . . . 51
4.4 Původní a transformovaný obrázek ze sekvence vložené do matice nul 52
4.5 Srovnání průměrné hodnoty poklesu nejlepšího jedince ze všech snímků zprůměrované ze 3 běhů . . . 57
4.6 Srovnání rozptylů průměrné MSE spojitého GA pro výchozí nastavení z tabulky 4.1 a nově stanoveného nejlepšího nastavení z tabulky 4.3 . 59
4.7 Srovnání průměrné hodnoty poklesu nejlepšího jedince ze všech snímků zprůměrovano ze 3 běhů . . . 61
4.8 Srovnání rozptylů průměrné MSE algoritmu PSO pro výchozí nastavení z tabulky 4.4 a nově stanoveného nejlepšího nastavení z tabulky 4.6 . . . 63
4.9 Srovnání průměrné hodnoty poklesu nejlepší včely ze všech snímků zprůměrované ze 3 běhů . . . 65
4.10 Srovnání průměrné hodnoty poklesu nejlepší včely ze všech snímků zprůměrované ze 3 běhů . . . 69
4.11 Srovnání průměrné hodnoty poklesu nejlepší světlušky ze všech snímků zprůměrované ze 3 běhů . . . 72
4.12 Průběh průměrných MSE na umělé sekvenci . . . 75
4.13 Uměle přidaný pohyb do 2. snímku zobrazeného na první referenční, registrace 2. snímku na první algoritmem PSO, registrace 2. snímku na první Nelder-Meadovým algoritmem . . . 78
4.14 Uměle přidaný pohyb do 13. snímku zobrazeného na první referenční, registrace 13. snímku na první algoritmem PSO, registrace 13. snímku na první Nelder-Meadovým algoritmem . . . 78

5.1 Snímek z klinické sekvence nádoru pankreatu s vyznačenou oblastí zájmu 79
5.2 Průběh kontrastu v první klinické sekvenci 80
5.3 Průběh průměrných MSE na první klinické sekvenci 82
5.4 Průběh průměrných MSE na první klinické sekvenci 83
6.1 Snímek z klinické abdominální sekvence s vyznačenou oblastí zájmu . 84
6.2 Průběh kontrastu v druhé klinické sekvenci 85
6.3 Průběh průměrných MSE na druhé klinické sekvenci 86

Seznam tabulek

4.1 Výchozí nastavení parametrů spojitého GA 56
4.2 Srovnání změny průměrné střední kvadratické chyby (MSE) a její směrodatné odchylky při vychýlení vždy jednoho parametru z výchozího nastavení ze tří běhů algoritmu spojitého GA 58
4.3 Srovnání nastavených parametrů, střední hodnoty a směrodatné odchylky MSE a výpočetní náročnosti pro výchozí a nově zvolené nastavení spojitého GA 58
4.4 Výchozí nastavení parametrů PSO 60
4.5 Srovnání změny průměrné střední kvadratické chyby (MSE) a její směrodatné odchylky při vychýlení vždy jednoho parametru z výchozího nastavení ze tří běhů algoritmu PSO 62
4.6 Srovnání nastavených parametrů, střední hodnoty a směrodatné odchylky MSE a výpočetní náročnosti pro výchozí a nově zvolené nastavení PSO 62
4.7 Výchozí nastavení parametrů BA 64
4.8 Srovnání změny průměrné střední kvadratické chyby (MSE) a její směrodatné odchylky při vychýlení vždy jednoho parametru z výchozího nastavení ze tří běhů algoritmu BA 66
4.9 Srovnání nastavených parametrů, střední hodnoty a směrodatné odchylky MSE a výpočetní náročnosti pro výchozí a nově zvolené nastavení BA 67
4.10 Výchozí nastavení parametrů ABC 68
4.11 Srovnání změny průměrné střední kvadratické chyby (MSE) a její směrodatné odchylky při vychýlení vždy jednoho parametru z výchozího nastavení zprůměrované ze tří běhů algoritmu ABC 69
4.12 Srovnání nastavených parametrů, střední hodnoty a směrodatné odchylky MSE a výpočetní náročnosti pro výchozí a nově zvolené nastavení ABC 70
4.13 Výchozí nastavení parametrů FA 71
4.14 Srovnání změny průměrné střední kvadratické chyby (MSE) a její směrodatné odchylky při vychýlení vždy jednoho parametru z výchozího nastavení zprůměrované ze tří běhů algoritmu FA 73
4.15 Srovnání nastavených parametrů, střední hodnoty a směrodatné odchylky MSE a výpočetní náročnosti pro výchozí a nově zvolené nastavení FA 73

4.16 Srovnání průměrných MSE dosažených středních a minimálních hodnot na jednotlivých snímcích a průměrných časů pro jednotlivá nejlepší nastavení optimalizačních metod na umělé sekvenci 74
4.17 Srovnání průměrných odchylek mezi skutečnými a algoritmy nalezenými parametry transformace a průměrné MSE středních hodnot u vybraných dvojic snímků na umělé sekvenci 76
5.1 Srovnání průměrných MSE dosažených minimálních hodnot na jednotlivých snímcích a průměrných časů pro jednotlivé optimalizační metody na první klinické sekvenci 81
5.2 Srovnání nalezených posunů a natočení jednotlivými algoritmy u vybraných dvojic snímků 1-15 a 1-28 83
6.1 Srovnání průměrných MSE dosažených minimálních hodnot na jednotlivých snímcích a průměrných časů pro jednotlivé optimalizační metody na druhé klinické sekvenci 86

Úvod

Ultrazvukové zobrazování je neinvazivní metodou a levnější alternativou k jiným lékařským zobrazovacím metodám, které nevystavuje pacienta ionizujícímu záření. Poskytuje nepřetržitý proud obrazů, které mohou sloužit například ke klinickému hodnocení celého orgánu. Pro zvýšení podílu informace v takto získané sekvenci ultrazvukových snímků bývá nutné obrazy na sebe zarovnat procesem nazývaným registrace. Takové registraci se říká monomodální a může být využita například pro diagnózu ultrazvukových snímků, které byly pořízeny měsice nebo roky od sebe. Multimodální registrace umožňuje kombinovat data z více modalit jako je ultrazvuk, CT, MRI, SPECT nebo PET a získat tak kompletní informace o pacientovi a zvýšit kvalitu zobrazování. Takto zarovnané sady obrazových dat jsou popsány transformacemi a předmětem registrace je poté najít parametry těchto transformací. Pro nalezení parametrů, aby došlo k co nejlepšímu zarovnání, se ukazuje jako dobrý způsob, využít optimalizační algoritmy a zejména pak evoluční algoritmy, které mají schopnost nalezení dobrého výsledku při hledání globálního optima, kterým jsou v tomto případě nejlepší hodnoty parametrů transformace.

Cílem první kapitoly je seznámení se s ultrazvukovým zobrazováním. Je zde vysvětleno, co ultrazvuk je, které formy ultrazvuku se v lékařství používají, jak ultrazvuk vzniká a které základní módy zobrazení se využívají, kdy je zejména kladen důraz na ultrazvukové zobrazení s kontrastními látkami.

Druhá kapitola teoreticky popisuje proces registrace obrazů. Ve svých podkapitolách dále do větší hloubky rozebírá pojem geometrická transformace obrazů a kritérium podobnosti obrazů.

Třetí kapitola se zabývá obecně optimalizačními algoritmy. Nejprve zde definuje problém globální optimalizace a dále se zabývá výčtem optimalizačních metod, mezi které patří tzv. neevoluční algoritmy, evoluční algoritmy a algoritmy rojové inteligence.

Čtvrtá až šestá kapitola se věnuje praktické části práci. Je zde popsán optimalizační proces hledání parametrů transformace v programovém prostředí MATLAB od nadefinování transformace obrazu, přes určení kriteriální funkce, po implementaci, konkrétní nastavení a vyhodnocení optimalizačních algoritmů, kterými byl spojitý genetický algoritmus (GA), Particle Swarm Optimalizace (PSO), včelí algoritmus (BA), algoritmus umělé včelí kolonie (ABC) a světluškový algoritmus (FA). Nejprve byly na simulované sekvenci určeny optimální parametry jednotlivých algoritmů a konkrétní nejlepší nastavení byly poté spuštěny na dvou reálných sekvencích.

1 Ultrazvukové zobrazování

Ultrasonografie je v současné medicíně nejrozšířenější diagnostická zobrazovací metoda. Důvodem jejího rozsáhlého využití v řadě lékařských oborů by se dalo připisovat vlastnostem, kterými nedisponují jiné zobrazovací modality. Vyšetření ultrazvukem je totiž neinvazivní [1], bezbolestné, bez negativních dopadů na lidský organismus a v neposlední řadě méně nákladné co se týče pořizovací ceny [1].

Ultrazvuková diagnostická technika vychází ze dvou samostatně se vyvíjejících odvětvích a její počátky sahají do 50. let minulého století. První linií, využívající ultrazvuk, byla průmyslová defektoskopie. Jedná se o metodu zkoumající vady materiálů, která funguje na principu detekce akustických rozhraní o různých akustických impedancích. Druhá linie byla založena na Dopplerově jevu, která má za úlohu detekovat pohyb tkání a měření rychlosti toku krve. V následujících desetiletích došlo k propojení obou skupin metod, a z toho důvodu se současné ultrazvukové diagnostické přístroje vyrábějí se zobrazovacími i dopplerovskými moduly [1].

1.1 Ultrazvuk v lékařství

Ultrazvuk je forma mechanického vlnění hmotného prostředí o frekvenci vyšší než je hranice rozsahu frekvencí vnímaných zdravým lidským uchem (16 Hz až 16 kHz), tj. vyšší než 20 kHz. Ultrazvuk v lékařství se často rozděluje podle frekvence kmitů na: [1, 2, 3]

- **nízkofrekvenční ultrazvuk** (od 20 kHz do 100 kHz) – využíván především v ultrazvukové chirurgii a k čištění materiálů a nástrojů
- **vysokofrekvenční ultrazvuk** (od 100 kHz) – uplatňuje se v:
  - *ultrazvukové terapii* (od 100 kHz do 3 MHz) k ohřevu tkání či drcení pevných útvarů v orgánech pomocí rázových vln a v zubním lékařství pro odstranění zubního kamene, a zejména pak v:
  - *ultrazvukové diagnostice* (od 2 MHz do 40 MHz) jako ultrazvukové zobrazovací systémy a v ultrazvukové mikroskopii, kde je využíváno až 80 MHz [2, 3].

Vlnění v oblastech frekvencí nižších než 16 Hz se označuje jako infrazvuk a vyšších než 1 GHz se nazývá hyperzvuk, které taktéž nelze lidským uchem zaznamenat [2, 3].

[1] Lékařský zásah prováděný bez zásahu do těla pacienta.

1.2 Fyzika ultrazvuku

Každé prostředí (např. tkáň) je z pohledu akustiky charakterizováno určitými parametry, mezi kterými jsou pro tuto úlohu nejvýznamější: rychlost šíření ultrazvuku danou tkání (fázová rychlost), akustická impedance a útlum. [1]

Na rozhraní dvou tkání dochází při projití ultrazvukového signálu k odrazu určitého množství akustické energie vlivem akustického rozhraní, vytvořeného rozdílnými akustickými impedancemi těchto tkání a frekvence ultrazvukových kmitů, udávající útlum. [1].

V medicínských aplikacích je element, generující a detekující ultrazvuk, obvykle stejný, bývá tento element pojmenovaný obecně jako ultrazvukový měnič, jelikož mění elektrický signál na akustický při generování ultrazvuku a akustický signál na elektrický při jeho detekci. [2]

1.2.1 Vznik ultrazvuku

V lékařství se energie ultrazvukové vlny z mechanické na elektrickou a naopak převádí zejména pomocí piezoelektrických a magnetostrinkčních měničů [3].

Piezoelektrické vlastnosti vykazují látky, jejíž ionty jsou uspořádány do krystalické mřížky, která při deformaci vytváří elektrický náboj a je deformována, je-li na ni přivedeno elektrické napětí [3].

Přímý piezoelektrický jev

Využívá se při detekci ultrazvukových vln. Pokud dojde k deformaci krystalu, kladné a záporné ionty se v krystalické mřížce přemístí, čímž se vytvoří měřitelný elektrický náboj, který lze detekovat [2, 3].

Nepřímý piezoelektrický jev

Využívá se u piezoelektrických generátorů jako zdroj ultrazvukových vln. Pokud je elektrické napětí přivedeno na povrch krystalu, kladné a záporné ionty se v krystalické mřížce přemístí, čímž dojde k deformaci krystalu, které je přímo úměrné velikosti přiloženého napětí [2, 3].

Magnetostrinkční měniče

Magnetostrinkční měniče fungují na principu magnetostrinkce, což je jev, který má za následek deformaci magnetostrinkčního materiálu, kterým nejčastěji bývají feromagnetické látky. Tyto látky jsou uloženy do střídavého magnetického pole a změnou

jejich objemu vlivem okolního magnetického pole dochází ke generování ultrazvukových vln. [2, 3].

1.3 Módy zobrazení

Materiály používané k výrobě měničů bývají obvykle krystaly oxidu křemičitého či keramické materiály, které jsou uměle připravené. Poskládají-li se poté měniče do většího celku, mohou pak dohromady tvořit ultrazvukovou sondu [2].

Mezi nejčasteji používané metody ultrazvukového měření v medicínských aplikacích patří v dnešní době *metoda průchodu* ultrazvukových vln vyšetřovaných prostředím, kdy jednou sondou je ultrazvuk vysílán a druhou přijímán, a *metoda odrazu* ultrazvukového vlnění, které je sondou vysláno do vyšetřovaného prostředí a následně po odrazu od rozhraní se vrací zpět do sondy [3].

Mezi nejvýznamější módy zobrazení v ultrazvukové diagnostice patří:

- **Zobrazení A** – nejjednodušší typ ultrazvukového zobrazení, které je jednorozměrné, kdy je využíván pouze jeden měnič [1, 3],
- **Zobrazení B** – dvojrozměrné zobrazení; vzniká dvourozměrný obraz, jehož světelné body (pixely) představují jas, který odpovídají velikosti odrazů ultrazvukových vln [3],
- **Zobrazení TM** – jednorozměrné dynamické zobrazení realizováno z B módu vybráním jedné jeho linie, která je zobrazena pomocí šedotónového signálu, který se vyvíjí v čase [3],
- **Zobrazení 3D** – redukuje obecný nedostatek zobrazovacích metod, kterým je ztráta informace z objemu daného objektu; snímání 3D dat je umožněno maticovými sondami nebo lineárním posunem, nakláněním a rotací sondy během snímání [1, 2]

1.4 Zobrazení s kontrastními látkami

Mezi ultrazvukové kontrastní látky patří mikrobubliny naplněné plynem v kapalině, která je pacientovi vstříknuta do krevního řečiště. Bublinky slouží jako odražeče, které ultrazvuk lépe odráží. Proto kontrastní látky v ultrazvukovém zobrazování slouží zejména ke zvýšení kontrastu v prokrvených částech objektu, kdy krev za normálního stavu pouze slabě odráží ultrazvukový impulz [2, 4]. Toto kontrastní ultrazvukové vyšetření se nazývá CEUS (Contrast-enhanced ultrasound).

Kontrastní látky pro ultrazvukové zobrazování by měly mít v ideálním případě tyto vlastnosti: být netoxické; mít schopnost procházet plicním, srdečním a kapilárním oběhem a být injektované intravenózně; mít obdobnou velikost jako mají

červené krvinky (do 10 µm, většinou však nepřesahují velikost 8 µm) , aby neucpávaly cévy a umožnily tak průchod bublinám do oblasti zájmu; a měly by obecně zvyšovat kontrast obrazu krve [5].

Mikrobubliny kontrastních látek jsou ohraničeny skořápkou, která má tloušťku v řádu desítek nm. Uvnitř mikrobublin se nachází plyn s nízkou rozpustností v krevním řečišti, který je po pár minutách intravenózního aplikování kontrastní látky vydechován plícemi z těla ven [6].

2 Registrace obrazu

Fúze obrazů je způsob, kdy je získávána informace z většího počtu obrazů, která by nebyla získatelná z každého jednotlivého snímku. Je zapotřebí obrazy nějakým způsobem dát dohromady zpravidla tzv. slícováním (z anglického jazyka *registrace*) [7], [8].

Fúze obrazů může být:

- *monomodální* – využívá se skupina snímků jediným zobrazovacím způsobem, např. vývoj v časové řadě snímků (sledování pohybu nebo vývoje léze v průběhu léčení apod.), stereoanalýza (sejmutí scény z různých pozorovacích míst pro zrekonstruování třírozměrného obrazu scény),
- *multimodální* – snímky jsou získány různými modalitami, např. kombinace CT, MRI, PET (SPECT) a ultrazvukového snímku [1] [7], [9].

2.1 Lícování („registrace“) obrazů

Lícování obrazů je předpokladem nebo předběžnou fází pro vlastní fúzi. K fúzi obrazů dochází, až jak jsou obrazy slícované a záleží na tom, co je cílem fúze, čili jaká informace je požadována, aby byla získána [7].

Během fúze dochází k několika postupným fázím, kdy nejprve je zapotřebí obrazy slícovat, čili položit je na sebe tak, aby souhlasila důležitá místa popř. vybrané body nebo zejména u medicínských snímků, kde nelze snadno nacházet význačné body, kdy snímky z vyšetření, které zjišťují rozdělení radiofarmak, jsou méně detailní než snímky např. z CT nebo MRI, je zapotřebí najít jiná kritéria podobnosti mezi obrazy a slícovat je tak, aby podobnost byla co největší [7] [8].

Proces lícování je transformace jednoho ze dvou obrazů, který se geometricky převede tak, aby odpovídající si body dané dvojice obrazů na sebe co nejlépe nalehly. Jde tedy o to, aby lícovaný obraz A se skutečně dobře posadil na referenční „bázový“ obraz B, který je nezdeformovaný, jednou provždy dán a na bodové množině X_B

$$X_A = \{\mathbf{x}_A\}, X_B = \{\mathbf{x}_B\}; \quad X_A \mapsto \{A(\mathbf{x}_A)\}, X_B \mapsto \{B(\mathbf{x}_B)\}. \tag{2.1}$$

Podpora X_A popř. X_B obrazu A popř. B je tvořena množinou bodů x_A popř. x_B a tyto podpory jsou zobrazeny na množiny jasů $A(\mathbf{x}_A)$ popř. $B(\mathbf{x}_B)$, což jsou obrázky A popř. B, které jsou na bodech $\mathbf{x}_A$ popř. $\mathbf{x}_B$ [7] [10].

[1] u ultrazvukových snímků bývá propojení obtížnější vzhledem k horší kvalitě snímků a větší šumové úrovni

Přilícovávaných obrazů A může být větši množství, kdy bude k dispozici stálý bázový obraz a k němu budou postupně přilícovávány více dílčích obrazů A_i, přičemž každý obraz bude přilícován samostatně [7] [8].

2.1.1 Podmínky transformací při lícování

Lícování je podmíněno nalezením geometrické transformace (podrobněji viz. kapitola 2.2), která obrazy optimálně přilícuje tak

$$T_\alpha : \quad \mathbf{x}_{A'} = T_\alpha\left(\mathbf{x}_A\right), \quad A'\left(\mathbf{x}_{A'}\right) = A\left(\mathbf{x}_A\right), \quad \mathbf{x}_{A'} = \mathbf{x}_B, \tag{2.2}$$

že body $\mathbf{x}_{A'}$, které budou podkladem obrazu A', budou geometricky transformované body z původního obrazu $\mathbf{x}_A$. Transformace má parametry charakterizující určitou transformaci, které jsou obsaženy ve vektoru α. Transformační předpis T_α se obvykle volí předem podle charakteru úlohy a vektor parametrů je nalezen příslušným optimalizačním postupem. Po provedení transformace bodů, hodnota jasu v transformovaném bodě transformovaného obrazu $A'\left(\mathbf{x}_{A'}\right)$ musí být stejná jako hodnota, která byla v původním bodě původního obrazu $A\left(\mathbf{x}_A\right)$ [7], [10].

Porovnávání obrazů se může uskutečnit pouze na jejich společné oblasti. Zpravidla se však transformovaný obraz zcela nepřekrývá s obrazem netransformovaným a jejich rozsahy se liší. Z toho důvodu je nutné kritérium podobnosti (podrobněji viz. 2.4) aplikovat pouze na část obrazu, která je společná a vytvořit průnik podpor transformovaného $X_{A'}$ a netransformovaného X_B obrazu

$$\Omega_\alpha = X_{A'} \cap X_B, \quad X_{A'} = \left\{T_\alpha\left(\mathbf{x}_A\right)\right\}, \quad X_B = \left\{\mathbf{x}_B\right\}, \tag{2.3}$$

kde Ω_α, symbol průniku dvou oblastí, je závislá na aktuální hodnotě vektoru α, jelikož množina transformovaných bodů se bude měnit podle toho, jaká transformace se zvolí. Pokud bude α hledáno iteračně, bude postupně nabývat různých hodnot a v každém kroku iterace bude jiný průnik [7] [8].

2.1.2 Optimalizační proces lícování

Optimalizace podle zvoleného podobnostního kritéria c (podrobněji viz. 2.4) lze zapsat jediným výrazem

$$\begin{aligned} \alpha_0 &= \arg\ \max_\alpha c\left(\left\{B\left(\mathbf{x}_B\right)\right\}, \left\{A'\left(T_\alpha\left(\mathbf{x}_A\right)\right)\right\}\right), \quad \mathbf{x}_B,\ T_\alpha\left(\mathbf{x}_A\right)\ \epsilon\ \Omega_\alpha \\ \alpha_0 &= \arg\ \min_\alpha c\left(\left\{B\left(\mathbf{x}_B\right)\right\}, \left\{A'\left(T_\alpha\left(\mathbf{x}_A\right)\right)\right\}\right), \quad \mathbf{x}_B,\ T_\alpha\left(\mathbf{x}_A\right)\ \epsilon\ \Omega_\alpha \end{aligned} \tag{2.4}$$

podle toho, jestli se jedná o minimalizaci nebo maximalizaci. Při optimalizaci je hledán argument α, který maximulizuje nebo minimalizuje kritérium c. Při zdárném

najití tohoto maxima nebo minima je nalezen hledaný optimální parametrický vektor α_0, který určuje transformaci. Podobnostní kritérium c, jež může být např. některé ze zmíněných kritérií v kapitole 2.4, hledá podobnost mezi množinou bodů $B(\mathbf{x}_B)$ tvořící obraz B a množinou přetransformovaných bodů $A'(T_\alpha(\mathbf{x}_A))$ obrazu A'. Dané kritérium se bude uplatňovat pouze na těch bodech $\mathbf{x}_B$ a $T_\alpha(\mathbf{x}_A)$, které spadají do společného prostoru Ω_α, který odpovídá konkrétnímu α, které se v dané transformaci použilo [7].

Existuje celá řada optimalizačních strategií, které jsou vyjádřeny příslušnými algoritmy. Obecně se však jedná o to, že v prostoru parametrů α, který je vícerozměrný a jeho dimenze záleží na tom, kolik paramterů určuje transformaci, je hledán takový bod, který bude maximalizovat nebo minimalizovat kritérium [7].

2.2 Geometrické transformace obrazů

Ke slícování obrazů je především potřeba geometricky transformovat obrazy. Zpravidla je postup takový, že jeden obraz je vybrán za standardní, s kterým se dále již nehýbe. Všechny ostatní obrazy, které budou podléhat fúzování, se budou geometricky přizpůsobovat tak, aby zde došlo k co nejlepšímu prostorovému souhlasu [7] [8].

Geometrické transformace obrazu jsou zobrazením původní roviny $\mathbf{r} = (x, y)$ do roviny $\mathbf{r}' = (x', y')$, do které bude vkládán geometricky transformovaný obraz. Zobrazení znamená, že je zadaný bod $\mathbf{r}$ v originální obrazové rovině, který je přenesen ve výstupní rovině do bodu $\mathbf{r}'$, jehož pozice bude dána transformační funkcí T [7]

$$\mathbf{r}' = T(\mathbf{r}). \tag{2.5}$$

Transformační funkce T musí být stanovena tak, aby ke každému vstupnímu vektoru $\mathbf{r}$ bylo možné nalézt výstupní vektor $\mathbf{r}'$. Dále musí být definována obecně nějakým vztahem (nejčastěji matematickou rovnicí nebo matematickým výrazem), ve kterém vystupují parametry, které určují, jak konkrétně bude funkce vypadat a určují transformaci [7].

Při geometrických transformacích je bod včetně svého jasu, který byl v $\mathbf{r}$, přenesen T do jiné pozice $\mathbf{r}'$, avšak jeho jas zůstane stejný. Tento poznatek vyjadřuje následující podmínka 2.6 [7]

$$f(\mathbf{r}') = f(T(\mathbf{r})) = f(\mathbf{r}). \tag{2.6}$$

Geometrické transformace se mohou aplikovat na celý obraz jako globální transformace anebo po částech, čímž se rozumí, že se obraz bude rozdělovat na několik dílčích částí a v každé části budou aplikovány samostatné geometrické transformace,

které se poté spojí dohromady za předpokladu splnění podmínek, např. že na hranicích oblastí musí na sebe navazovat [7].

2.2.1 Tuhé (rigidní) transformace

Jedná se o nejzákladnější geometrické transformace, které jsou lineární [9].

Posun

Nejjednodušší transformací tohoto typu je posun 2.7 [7] [8]

$$\mathbf{r}' = \mathbf{r} + \Delta\mathbf{r}, \tag{2.7}$$

kdy výstupní vektor $\mathbf{r}'$ je vůči vstupnímu vektoru $\mathbf{r}$ posunutý o $\Delta\mathbf{r}$. Je tedy možné se kterýmkoliv směrem v obrázku posunout, avšak jiným způsobem není měněn [7], [10].

Rotace

Rotace obrázku je další rigidní transformací, která je charakterizována transformační maticí $\mathbf{B}$ [7] [8]

$$\mathbf{r}' = \mathbf{B}\mathbf{r}. \tag{2.8}$$

Rotovat je možné kolem kteréhokoliv bodu v obraze. Pokud bude rotace podle počátku souřadnic, pak transformační matice mezi vektory $\mathbf{r}$ a $\mathbf{r}'$ bude mít tvar [7]

$$\mathbf{B} = \begin{pmatrix} cos\theta & -sin\theta \\ sin\theta & cos\theta \end{pmatrix}. \tag{2.9}$$

Obecná rigidní transformace

Spojí-li se obě již výše zmíněne transformace dohromady, vytvoří se nejobecnější tuhá transformace, jejíž vektorový zápis bude [7]

$$\mathbf{r}' = \mathbf{B}\mathbf{r} + \Delta\mathbf{r}. \tag{2.10}$$

Pro maticový zápis vztahu 2.10 pomocí transformační matice nelze maticí 2×2 vyjádřit přičítání. Proto byly zavedeny tzv. homogenní souřadnice $\mathbf{x} = [x, y, 1]$, kdy ke skutečným souřadnicím x a y byla přidána jedna nevyužívaná souřadnice, která je zpravidla rovna 1 [7].

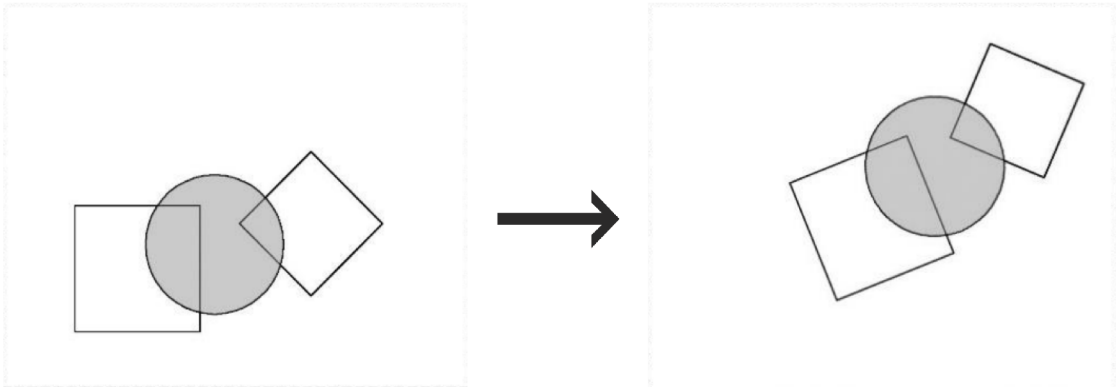
Vztah 2.10 je tedy možný zapsat pomocí jediné translačně-rotační matice $\mathbf{R}$ [7]

$$\mathbf{r}' = \mathbf{R}\mathbf{r} \tag{2.11}$$

a jeho 2D interpretace

$$\mathbf{r}' = \begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} cos\theta & -sin\theta & \Delta x \\ sin\theta & cos\theta & \Delta y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}. \tag{2.12}$$

Na obrázku 2.1 je vyobrazena celá tuhá transformace, kde původní obraz v nákresně vlevo je pootočený a posunutý na transformovaný obraz vpravo [7].



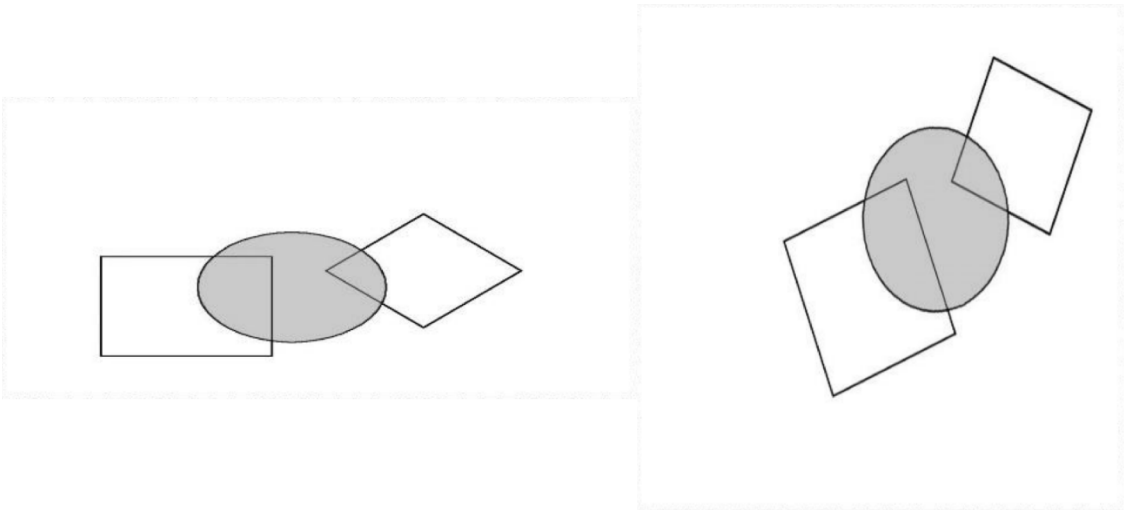
Obr. 2.1: Rigidní transformace: původní obrázek (vlevo), posunutý a orotovaný obrázek (vpravo), převzato z [7]

2.2.2 Pružné (flexibilní) transformace obrazu

Dalším typem geometrických transformací jsou transformace pružné, které mohou být lineární i nelineární. Fyzikální představa flexibilních transformací je taková, že je-li obraz nakreslen na pružné bláně, která je různým způsobem deformována (natahována v různých směrech), lze tímto způsobem dostat možnost především obecného zvětšení [7] [9].

Obecné zvětšení (změna měřítka)

Obecné zvětšení spočívá v tom, že pružná blána obrazu je táhnutá vodorovně, vertikálně nebo v obou dvou směrech, viz. obrázek 2.2 [7].



Obr. 2.2: Změna měřítka obrázku 2.1: vodorovné zvětšení (vlevo) a vodorovné i vertikální zvětšení (vpravo), převzato z [7]

Obecný předpis této transformace je definovaný jako [7] [10]

$$\mathbf{r}' = \mathbf{S}\mathbf{r} \tag{2.13}$$

a o kolik je obraz ve směru osy x a osy y zvětšen popř. zmenšen, je vystiženo koeficienty s_x (vodorovné zvětšení) a s_y (vertikální zvětšení) [7] [8]

$$\mathbf{S} = \begin{pmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{pmatrix}. \tag{2.14}$$

Pokud by zvětšení podle os bylo stejné, byly by zachovány úhly obrázku. Pokud však zvětšení podle os stejné není (obrázek 2.2), dochází ke změnám tvarů a úhlů obrázku. Dalším poznatkem je, že zůstává zachovaná rovnoběžnost rovnoběžných přímek [7].

Zkosení

Zkosení, jakožto další druh pružné transformace, je možné vyjádřit pomocí vztahu [7]

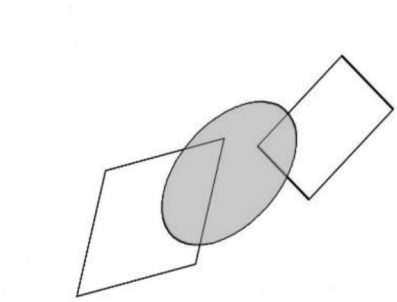
$$\mathbf{r}' = \mathbf{G}\mathbf{r}. \tag{2.15}$$

V tomto případě lze dílčí matice v dvourozměrném prostoru zapsat jako [7]

$$\mathbf{G} = \mathbf{G_x}\mathbf{G_y},\ \mathbf{G_x} = \begin{pmatrix} 1 & g_{xy} & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix},\ \mathbf{G_y} = \begin{pmatrix} 1 & 0 & 0 \\ g_{yx} & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \tag{2.16}$$

kde matice $\mathbf{G_x}$ představuje zkosení podél osy x a matice $\mathbf{G_y}$ zkosení podél osy y. Nenulové koeficienty g_{xy} a g_{yx} v maticích vyjadřují, jak se bude měnit x-ová popř. y-ová souřadnice, když se bude měnit y-ová popř. x-ová souřadnice, tedy x-ová souřadnice bude také záviset na y-ové souřanici a naopak. V případě 2.16 nenastává žádné zvětšení, jelikož na hlavní diagonále se vyskytují pouze jedičky a dochází tak pouze k zešikmení. Výsledná matice $\mathbf{G}$ je poté součin dvou dílčích matic a představuje zešikmení podél obou os [7].

Na obrázku 2.3 se původně pravoúhlému obrazu zkosením změnila nákresna, která má již jiné úhly. Přesto však toto zkosení zachovává rovnoběžnost rovnoběžných přímek [7].



Obr. 2.3: Rotace: transformován obsah i rám obrazu, převzato z [7]

Afinní transformace

Nejobecnější lineární transformace obrazu je tzv. afinní transformace, která je kombinací všech tří dílčích transformací výše popsaných - rotace a posunutí (obecná rigidní transformace popsána v kapitole 2.2.1), změma měřítka a zkosení

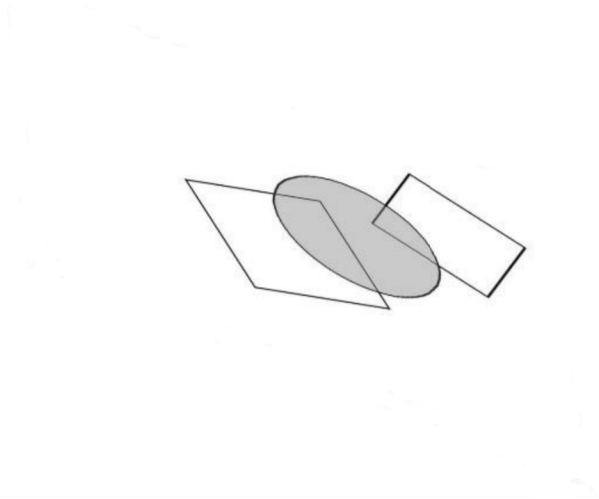
$$\mathbf{r}' = \mathbf{GSRr} = \mathbf{Ar} \tag{2.17}$$

a obráceně lze říct, že každá z těchto tří transformací je zvláštním případem afinní transformace. Afinní transformace je obecně vyjádřena maticí **A**, která v dvourozměrném prostoru bude vypadat takto

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} & t_1 \\ a_{21} & a_{22} & t_2 \\ 0 & 0 & 1 \end{pmatrix}, \tag{2.18}$$

význam koeficientů v této matici je dán kombinací všech dílčích transformací [2] [7], [9].

Příklad výsledku afinní transformace je vyobrazen na obrázku 2.4, který zachovává rovnoběžnost, ale nikoli úhly, tvary a vzájemné velikosti, které se mění se změnou tvaru [7].



Obr. 2.4: Příklad afinní transformace, převzato z [7]

Afinní transformace je pravděpodobně nejpoužívanější metodou pro jednoduché transformace obrazu, protože umožňuje přizpůsobit velikost a do značné míry se přizpůsobit tomu, že různé modality mají nepřesně pravoúhlost nákresny. Je však omezena pouze na tři výše uvedené transformace. Složitější transformace již musejí být nelineární [7].

Nelineární (a pružné) geometrické transformace

Projektivní transformace je afinní transformace, která je upravena tím, že pro každý bod obrazu se jeho pozice změní v daném určitém poměru. Uplatňuje se

[2] v obecné afinní transformaci již nelze mluvit o tom, že daný koeficient je např. jenom kosínus úhlu, ale stává se součástí těchto koeficientů, když je součástí afinní transformace a také rotace

zejména tehdy, pokud jde o zpracování optické transformace s perspektivním zkreslením. Umožňuje zpracovávat, rekonstruovat nebo odstraňovat perspektivní zkreslení [3] [7].

Afinní transformace, popsána v kapitole 2.2.2, je **polynomiální transformace** 1. stupně. Stupeň polynomů však může být jakkoliv vysoký. Výstup této transformace pro danou souřadnici je vyjádřený všemi možnými mocninani a kombinacemi v součinech mocnin daných veličin vstupního bodu, který vstupuje do transformace. Např. polynomiální transformace 2. stupně bude definována celkem dvanácti parametry [7].

2.2.3 Transformace obrazu po částech

Lokální transformace se zpravidla používá při transformacích obrazu po částech, které ovšem na sebe musí navazovat. Především je tedy nutné obraz rozdělit. Existují dvě možnosti, kterými dochází k rozdělení obrazu.

- *Pravidelné rozdělení* - rozdělení obrazu předem definovaným způsobem, apriorně se využívá mřížka, která bývá pravoúhlá.
- *Nepravidelné rozdělení* - obraz je dělen nepravidelně, rozdělení je většinou dáno kontrolními body (lícovacími body), které jsou k dispozici, jež obraz vhodným způsobem rozdělí [7].

Uvnitř každé sekce obrazu bude tatéž transformace, pokud jde o její typ (rovnice bude pořád stejná), ale lišit se bude parametry [4]. V každé části obrazu je tedy nutné znovu spočítat parametry, protože se v jednotlivých částech mohou lišit. Pro stanovení parametrů lokálních transformací je zapotřebí zajistit, aby jednotlivé části na sebe navazovaly, tzn. že na hranicích obraz musí geometricky navazovat a to také znamená, že zleva i zprava musí být na každé hranici stanoveny tytéž hodnoty jasu. V náročnějších případech lze klást podmínku, aby jasová funkce obrazu nejenom navazovala (aby zleva i zprava byly stejné hodnoty), ale aby dokonce souhlasily i tečny, tedy aby parciální derivace na hraně byly shodné zleva i zprava [7].

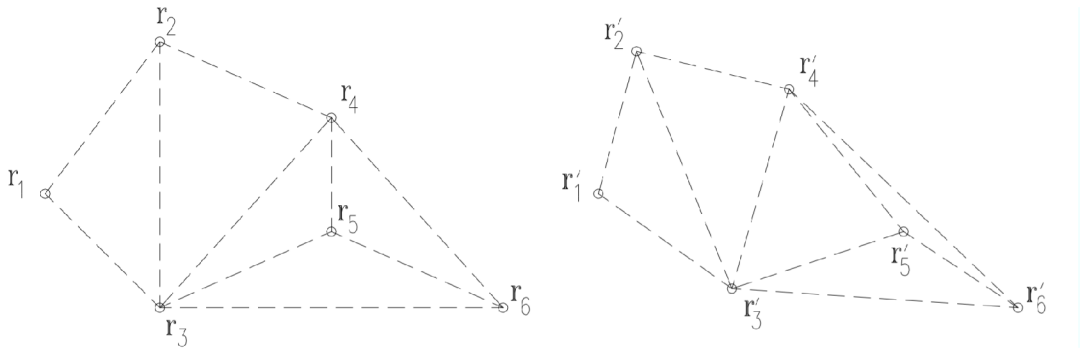
Geometrické transformace v trojúhelníkové síti

Ukázka rozdělení obrazu využitím dvojic k sobě patřících bodů (detailněji popsáno v kapitole 2.3.2) je vyobrazena na obrázku 2.5, kde výstupní obraz není známý (nejsou známy transformační vztahy), ale jsou známy pozice některých bodů. Pokud je vstupní obraz pomocí bodů, které jsou k dispozici, rozdělen na trojúhelníky, získávají se jednotlivé oblasti, přičemž např. oblasti $\mathbf{r_1}$, $\mathbf{r_2}$ a $\mathbf{r_3}$ vstupního obrazu

[3] např. odstranění sbíhavosti linií u objektů

[4] případ, že by se použily v různých oblastech při po částech řešené transformaci různé transformační vztahy se v praxi často neprovádí

odpovídá jinak tvarovaná oblast $\mathbf{r'_1}$, $\mathbf{r'_2}$ a $\mathbf{r'_3}$ ve výstupním obraze. Dochází tak k deformování každé oblasti (trojúhelníku) separátně, přičemž budou souhlasit odpovídající si body [7].



Obr. 2.5: Geometrická transformace v trojúhelníkové síti: korespondující sady kontrolních bodů, vstupní obraz (vlevo) a výstupní obraz (vpravo), převzato z [7]

Jedná se tedy o po částech lineární afinní transformaci, při N počtu trojúhelníků pro jednotlivé obrázky. Pro n-tý trojúhelník existují tři dvojice bodů. Rovnice pro tři body v trojúhelníku, které jsou označeny indexem k bude mít tvar [7]

$$\mathbf{r'}_{n,k} = T(\mathbf{r}_{n,k}, \mathbf{p}_n) = \mathbf{A}_n \mathbf{r}_{n,k}, \quad k = 1, 2, 3, \quad n = 1, \ldots, N. \tag{2.19}$$

Pro každý dílčí n-tý trojúhelník (trojici bodů) platí systém 6 rovnic pro 6 parametrů [7].

Geometrické transformace v apriorní (čtvercové) síti

Již se nejedná o linární transformaci, ale o transformaci po částech, kde v každé části se využívá polynomu 3. řádu. Výchozí síť bývá ve dvojrozměrném prostoru pravidelně čtvercová a bude souhlasit se vzorkovací sítí. Dále budou známy polohy bodů odpovídající uzlovým bodům sítě v přetransformovaném obraze. V referenčním obraze, na který se bude transformovat, se vyhledají pozice jednotlivých bodů, které jsou určeny uzlovými body vstupního obrazu. Na základě toho jsou navrženy příslušné polynomy, kdy nejčasteji se používá splinová transformace 3. stupně a spliny jsou navrženy tak, aby na sebe navazovaly na hranách i derivacemi [7].

2.2.4 Problémy při užití geometrických transformací

Při použití geometrických transformací dochází k problémům, kterým lze předejít nebo je vyřešit. Především se jedná o to, že transformovaný obraz může částečně

přesahovat rámec výstupního obrazu. Jestliže výstupní obraz je výjádřen stejně velkou maticí jako vstupní obraz, tak transformace může vést k tomu, že některé body se octnou mimo matici a jsou pro daný obraz ztraceny [7].

Další problém nastává, když část výstupního obrazu zůstane nepokrytá, protože se do něj nepřenesou žádné body transformací ze vstupního obrazu [7].

Navíc je zde vážný problém, že v diskrétním prostředí transformované pozice vzorků nespadají do uzlů vzorkovací mřížky. Zřejmě je tedy zapotřebí body ve vzorkovací mřížce výstupního obrazu přibližně zaplnit, což je prováděno zrekonstruováním (interpolací) hodnot jasu v bodech mřížky výstupního obrazu [7].

2.3 Stanovení parametrů transformace

Transformace jsou definovány parametry, které je zapotřebí nějakým způsobem určit, aby byly transformace proveditelné. K parametrům transformace lze dojít následujícími dvěma způsoby.

2.3.1 Analytické odvození

Parametry transformace je možné odvodit analyticky, pokud je předem známo, jaké má být geometrické zkreslení obrazu. V oblasti biomedicínského inženýrství jde např. o konverzi vějířového B-skenu v polárních souřadnicích v ultrasonografii do pravoúhlých kartézských souřadnic, jelikož monitory, na kterých se snímek vyhodnocuje, mají pravoúhlý charakter. Při transformaci z polárních do kartézských souřadnic je známý přesný transformační vztah a dají se předem vypočítat koeficienty transformace [7].

2.3.2 Zvolení typu funkce a vypočtení parametrů

U druhého způsobu stanovení parametrů transformace je známa pouze informace, že obraz se má nějakým způsobem přeměnit na nějaký jiný. Parametry transformace $\mathbf{p}$ lze vypočíst na základě množiny K dvojic bodů (které se často označují jako lícovací nebo registrační dvojice), které jsou definovány souřadnicemi bodů ve zpracovávaném obraze (x_k, y_k) a odpovídajícími souřadnicemi bodů ve výsledku (x'_k, y'_k). Pro každou dvojici bodů musí být splněna rovnice, kde je známý funkční předpis, který byl zvolen (např. polynom). Dále jsou známy pozice $\mathbf{r}'$ a $\mathbf{r}$ a parametry funkce $\mathbf{p}$ jsou neznámé. Z jedné rovnice [7]

$$\mathbf{r}' = T(\mathbf{r}, \mathbf{p}), \tag{2.20}$$

nelze parametry $\mathbf{p}$ určit. Avšak jedna dvojice bodů již určuje rovnice dvě a zřejmě, bude-li v rovnici $2K$ parametrů, tak je zapotřebí K dvojic a tzn. že v obraze se musí najít korespondujících dvojic tolik, aby se dostalo nejméně $2K$ rovnic, které jsou zapotřebí pro určení $2K$ parametrů funkce [7].

Složitost řešení rovnic bude záviset na tom, jakého typu jsou rovnice. Pokud budou rovnice nelineární, čili funkce T bude nelineární, tak v tom případě se může stát, že přímé řešení výsledného systému rovnic nebude snadné. Avšak v případě nejčastějších nelineárních transformací, což jsou transformace polynomiální, vztahy, které definují transformaci, jsou vzhledem ke koeficientům lineární [7].

2.4 Kritéria podobnosti obrazů

Podobnost dvou obrazů je vyjádřena kritérii podobností a může se týkat celých obrazů anebo jejich vymezených oblastí, přičemž porovnávané obrazy musejí mít stejnou velikost. Vyskytuje se velké množství skupin kritérií, které se používají pro hodnocení podobnosti obrazů. Tyto skupiny kriterií mohou záviset např. na intenzitě jasu, informaci v obraze nebo nalezených význačných bodech ve dvou obrazech [7].

Pro jednoduchost budou brány v potaz šedotónové obrazy, jelikož s ohledem na aplikaci v medicíně, převážná část medicínských obrazů je stále ještě šedotónová.

2.4.1 Kritérium podobnosti založené vzdálenosti

Podobnost dvou obrazů lze vyjádřit pomocí metriky střední kvadratické odchylky (MSE), která je hojně využívanou metrikou podobnosti při registraci ultrazvukových sekvencí. Tato metrika vyhodnocuje zarovnání výpočtem vzdáleností mezi polohou bodu v referenčním obraze x_i a transformovaném obraze $T(x_i)$ následujícím způsobem [11], [12]

$$MSE = \frac{1}{r} \sum_{i=1}^{r} \|x_i - c\,(T(x_i))\|^2\,, \tag{2.21}$$

kdy každý bod v referenčním obraze je přiřazen k nejbližšímu bodu v transformovaném bez ohledu na to, zda byl tento bod již přiřazen k jinému bodu referenčního obrazu [11].

2.4.2 Kritéria podobnosti založená na intenzitě

Kritéria založené na intenzitě vycházejí pouze z jasových hodnot. Jedná o monomodální fúzi, jelikož se fúzují obrázky, které pocházejí ze stejného typu zdroje [7].

Korelační kritéria

Prvním kritériem je tzv. **korelace** mezi obrazy, kdy je vytvářen součet součinů odpovídajících si hodnot ve dvou obrazech. Výpočet tohoto kritéria je jednoduchý, ale v mnoha úlohách je většinou nepoužitelné, protože je citlivé na lokální střední hodnoty, které mohou kritérium zcela zkreslit. Lepším kritériem, které se zbavuje středních hodnot obou obrazů, je tzv. **kovariance**, avšak je již výpočetně složitější. O něco spolehlivější co do skutečného zjištění podobnosti je tzv. **korelační koeficient**, ale taktéž se jedná o výpočetně poměrně náročnou úlohu [7], [11].

Vektorová kritéria

Vycházejí z představy obrazu jako vektoru. Tento vektor má N počet složek a nachází se v N-rozměrném prostoru. Každý porovnávaný obraz je v tomto N-rozměrném prostoru reprezentován jedním vektorem. Podobnost obrazů poté může být hodnocena **Euklidovskou vzdáleností** na základě absolutní hodnoty rozdílů vektorů. Toto kritérium však má velmi omezenou použitelnost, a to na obrazy, které mají shodné stupnice jasu [7].

Z tohoto hlediska je výrazně lepší použít **kosínové kritérium**, které se zabývá úhlem mezi danými vektory, a tedy čím větší bude úhel, tím si budou obrazy méně podobné. Úhel δ, který svírají vektory, může být vyjádřen pomocí vektorů jako součin jejich absolutních velikostí, který je násoben kosínem sevřeného úhlu jako $\mathbf{a}\,\mathbf{b} = |\mathbf{a}|\,|\mathbf{b}|\,cos(\delta)$. Tím pádem je možné toto kritérium (kosínus) vypočítat pomocí vzorce [7]

$$C_A\left(\mathbf{a},\mathbf{b}\right) = \frac{\mathbf{ab}}{|\mathbf{a}|\,|\mathbf{b}|} = \frac{\sum_i a_i b_i}{\sqrt{\sum_i a_i^2}\sqrt{\sum_i b_i^2}}, \tag{2.22}$$

kde v čitateli se vyskytuje skalární součin dvou vektorů a ve jmenovateli součin dvou absolutních velikostí vektoru. Dané kritérium je zcela necitlivé na lineární transformace jednotlivých obrazů, pokud jde o jas a málo citlivé, pokud jsou transformace monotónní. Pro případy kdy se jedná o monomodální porovnávání obrazů, kdy při lícování bude k dispozici jeden pevný obraz a budou se zjišťovat různé varianty druhého obrazu, bude stačit počítat kritérium tak, že se nebude stále určovat absolutní hodnota druhého obrazu a lze jí ze vzorce 2.22 vynechat. Tím pádem se daný výraz zjednoduší [7]

$$C'_A\left(\mathbf{a},\mathbf{b}\right) = \frac{\mathbf{ab}}{|\mathbf{a}|}. \tag{2.23}$$

Je tedy zřejmé, že aby byly obrazy co nejpodobnější, bude zapotřebí kosínus úhlu maximalizovat, aby úhel klesal co možná k nule [7].

2.4.3 Kritéria založená na informaci v obraze

Další skupinou kritérií jsou tzv. informační kritéria podobnosti, která se využívají zejména tam, kde se stupnice jasu nebo barev u dílčích obrazů liší, jako je tomu u multimodální fúze. Koncept této skupiny kritérií je najít a kvantifikovat vztah mezi třídami jasů v porovnávaných obrazech. Četnosti jasů, získány z histogramů obrazů, mohou být použity pro odhady pravděpodobností výskytu jednotlivých jasových tříd podílem absolutní četnosti celkovým počtem pixelů. Pravděpodobnost jasu l v prvním obraze A bude dána podílem příslušného sloupečku v histogramu daného obrazu, který odpovídá jasu l a celkovým počtem pixelů N obrazu. Zcela obdobný postup nastává i u druhého obrazu B. Pro společné pravděpodobnosti jasů obou těchto obrazů se bude vycházet z absolutních četností konkrétních kombinací dvou jasů l a m, které jsou opět poděleny celkovým počtem pixelů obrazů [5] [7] [9]

$$p_A\{l\} \approx \frac{1}{N} h_l^A, \quad p_B\{l\} \approx \frac{1}{N} h_l^B, \quad p_{AB}\{l,m\} \approx \frac{1}{N} h_{l,m}^{AB}. \tag{2.24}$$

Jakmile jsou deklarovány vztahy, je zapotřebí je kvantifikovat tak, aby byla vystižena vazba mezi obrazy jedinou hodnotou. Informační kritérium **vzájemné informace** I_{AB} vychází z teorie informace

$$I_{AB} = H_A + H_B - H_{A,B}, \tag{2.25}$$

kde H_A a H_B jsou entropie náležící obrazu A popř. B, které vycházejí z pravděpodobností jednotlivých jasových tříd daného obrazu. H_A a H_B tedy charakterizují informaci, kterou nese konkrétní obraz. $H_{A,B}$ ve vzorci 2.25 vyjadřuje průměrnou informaci, kterou nese jedna dvojice jasů dvou obrazů a charakterizuje informaci, která je obsažena v této sdružené dvojici obrazů, kdy se již v tomto případě jedná o tzv. vektorový obraz. Kritérium vzájemné informace je citlivé na podobnost obsahu obrazu a nezávisí na tom, jaké jsou konkrétní stupnice jasu (nebo barev) v jednom a druhém obrazu. Následkem toho je toto kritérium vhodné pro porovnávání obrazů v multimodálním případě a je tedy velmi často používáno v medicínské oblasti, jelikož se zde obvykle porovnávají a zejména lícují obrázky, které pocházejí z různých modalit [7], [8], [8], [9], [11].

2.4.4 Kritéria slícování podle význačných bodů

Mezi nejjednodušší skupinu kriterií podobností patří kritéria podle lícovacích bodů, kdy se předem určí, které body k sobě patří a které se tedy budou na sebe přetransformovávat (nalícovávat) a podobnost se poté může hodnotit tím, jak dobře se tyto

[5] při tomto odhadu se předpokládá, že obrazy jsou realizacemi náhodných polí A' a B' a dále pak, že pravděpodobnosti jsou odhadovány z jediné realizace těchto polí

sobě odpovídající body prostorově ztotožnily. Tedy na základě předem vybraných dvojic bodů korespondujících v obou obrazech bude zhodnoceno, jestli jsou obrázky dobře slícované. V medicínském případě se jedná např. o větvení a křížení cév, což jsou případy, kde je možné nalézt relativně snadno body, které si odpovídají, aniž by byla velká pravděpodobnost, že došlo k chybnému nalezení těchto bodů. Pro jednotlivou dvojici bodů, pokud jsou obrazy nějakým způsobem formulovány, např. transformovány při lícování, jde o porovnání pozic význačných bodů, kdy odchylka pro i-tou dvojici bodů $\mathbf{e}_i$ je vzdálenost dvou pozičních vektorů mezi dvěma obrazy. Pro n dvojic bodů lze vypočítat **průměrnou odchylku** a **průměrnou kvadratickou odchylku**

$$E_{abs} = \frac{1}{n}\sum_i w_i \left|\mathbf{e}_i\right|, \quad E_{sq} = \frac{1}{n}\sum_i w_i \left|\mathbf{e}_i\right|^2, \quad \sum_i w_i = n \tag{2.26}$$

které budou podléhat minimalizaci např. v rámci lícování, kdy dochází k postupným úpravám parametrů geometrické transformace tak, aby se dva obrazy co nejvíce podobaly. Soubor vah w_i je možné použít k tomu, aby byla vystižena důležitost některých dvojic, kterým bude přiřazena větší váha než méně důležitým dvojicím, u kterých lze připustit, že odchylka bude větší. Součet všech vah musí být roven počtu dvojic n [7].

3 Evoluční algoritmy pro registraci obrazů

Evoluční algoritmy jsou součástí širší problematiky, které se říká optimalizace. Evoluční algoritmy totiž nachází uplatnění především v optimalizaci.

3.1 Problém globální optimalizace

Optimalizace je proces, při kterém je hledáno nejvhodnější (optimální) řešení nějakého problému [1]. Nejlepší řešení, pokud není známo žádné lepší, se nazývá řešení optimální.

Při optimalizaci jsou měněny tzv. stavové proměnné optimalizovaného systému a zároveň je sledován vliv nastavení těchto proměnných na výsledné parametry tohoto systému. Kvalita řešení, čili závislost aktuálních parametrů daného systému na nastavených stavových proměnných, musí být popsána funkcí, která je v textech označována jako kriteriální (nebo také účelová či chybová). Optimalizaci lze poté chápat jako hledání minima nebo maxima této kriteriální funkce změnou hodnot stavových proměnných, které jsou nastaveny právě tak, aby bylo nalezeno její minimum nebo maximum [8], [13], [14]. V této práci je v praktické části uvažována minimalizace kriteriální funkce.

3.1.1 Formulace problému globální optimalizace

Problém globální optimalizace spočívá v tom, že musí být vymezena nějaká oblast D, kde je dané řešení hledáno [13]

$$D = \langle a_1, b_1\rangle \times \langle a_2, b_2\rangle \times \ldots \times \langle a_d, b_d\rangle = \prod_{i=1}^{d} \langle a_i, b_i\rangle, \quad a_i < b_i,\ i = 1, 2, \ldots d. \tag{3.1}$$

Tato oblast D je obecně vymezena jako d-rozměrný kvádr. Dále musí být k dispozici již zmíněna účelová funkce $f(x)$, která umožní vyhodnotit kvalitu každého bodu v d-rozměrném kvádru. Kriteriální funkce tedy musí být také d-rozměrná [13]

$$f : D \to R, D \subseteq R^d. \tag{3.2}$$

V bodě $\mathbf{x}^* = arg\ min_{x\epsilon D} f(\mathbf{x})$ je poté nalezeno globální minimum, což je jeden nebo více bodů v prostoru, ve kterém účelová funkce dosahuje minimální hodnoty. Formálně je globální minimum charakterizováno jako 3.3 [13]

$$\mathbf{x}^* = arg\ min_{x\epsilon D} f(\mathbf{x}) = \{\mathbf{x}^*\epsilon D : f(\mathbf{x}^*) \leq f(\mathbf{x}) \text{ pro } \forall \mathbf{x}\epsilon D\}. \tag{3.3}$$

[1] např. nastavení nějakého systému nebo nalezení správných transformačních parametrů

Pokud by daná úloha spočívala v hledání globálního maxima, tak by řešila stejným způsobem, akorát by se při hledání globálního minima funkce změnilo znaménko $g(x) = -f(x)$.

3.1.2 Deterministické a stochastické algoritmy

Pro optimalizaci jako již zmíněný proces hledání globálního minima nebo maxima kriteriální funkce změnou stavových proměnných, lze použít několik postupů. V zásadě jsou tyto přístupy rozděleny na dvě skupiny, a sice na tzv. deterministický a stochastický přístup.

Deterministický přístup

Tento postup využívá základní matematické analýzy, kdy nachází extrémy funkcí, u kterých existuje první a druhá derivace. Jelikož však účelová funkce nemusí mít první a druhou derivaci nebo v některých oblastech ji nemá, případně se mohou vyskytovat další nepříjemnosti, které ztíží deterministický postup, byly vyvíjeny další metody [13].

Stochastický přístup

Stochastické algoritmy využívají nějakou náhodnost. V zásadě se od deterministických algoritmů liší tím, že sice nezajišťují nalezení řešení v konečném počtu kroků, ale často pomohou nalézt v přijatelném čase takové řešení, které zajistí, že se dosáhne alespoň blízkosti optimality [13].

Stochastické algoritmy pro optimalizaci prohledávají daný prostor D, a to heuristicky. Heuristika je postup, který nemá formální matematický popis, využívá však náhodu, intuici, analogii a zkušenost. Dále tyto algoritmy obyčejně obsahují nějaký proces učení [13].

Evoluční algoritmy

Stochastické algoritmy evolučního typu jsou odvozeny ze znalostí přírodních nebo sociálních procesů, které se s poměrným úspěchem v posledních desetiletích hojně využívají pro hledání globálního minima funkcí. Tzv. evoluční algoritmy jsou algoritmy, které byly inspirovány Darwinovou evoluční teorií vývoje populací [13].

Princip těchto algoritmů je, že na začátku se vždy náhodně vygeneruje celá populace jedinců, kterou bychom měli postupně zkvalitňovat. Jedinci představují jednotlivá řešení daného problému a mají různou kvalitu, čili jedincům odpovídají různé hodnoty dané kriteriální funkce. Práce s populací následně spočívá v tom, že probíhá iteračně, kdy se vždy generuje nová populace s tím, že každá nová populace

by měla být kvalitnější, než ta předchozí. K tomu se využívají tzv. evoluční operátory, mezi které patří tyto tři základní [13]:

- *selekce* – spočívá v tom, že nejsilnější jedinci z populace mají větší pravděpodobnost přežití a předání svých vlastností do další generace potomků,
- *křížení* (rekombinace) – při křížení je o tom, že se dva nebo někdy i více jedinců (rodičů) z populace si vymění informace a vzniknou tak noví jedinci (potomci), kteří kombinují vlastnosti rodičů,
- *mutace* – spočívá v tom, že informace zakódovaná v náhodně vybraném jedinci může být nějakým způsobem náhodně modifikována [2] [13].

S podobnými pojmy pracuje i genetika.

Rojová inteligence

Zvláštní skupinu tvoří algoritmy tzv. rojové inteligence, které se začaly vyvíjet v posledních dvou desetiletích. Jedná se o optimalizační systémy, které jsou inspirovány inteligentním chováním společenstev některých živočichů (hmyzu - včely, netopýrů, vlků, ptáků, ryb atd.).

Roj zde představuje určitou populaci jedinců (někdy se používá pojem agentů, což jsou jednotlivá možná řešení) a tito jedinci spolu nějakým způsobem komunikují a jejich vzájemná interakce přispívá k tomu, aby se inteligentně chovalo celé společenství.

3.2 Optimalizace vycházející z matematické analýzy

Tyto metody hledají extrémy funkcí, u kterých existuje 1. a 2. derivace. Pokud má účelová funkce více lokálních minim, tak tento druh metod nalezne pouze jedno z těchto lokálních minim.

3.2.1 Metoda nejstrmějšího sestupu

Mezi nejjednodušší metody využívající numerických derivací patří metoda nejstrmějšího sestupu. Metoda je určena k hledání jediného minima. Při řešení problému optimalizace je zapotřebí mít k dispozici analytické vyjádření účelové funkce, což ovšem nebývá častý případ, a proto lze postupně aktualizovat odhad optima a využít numerický výpočet derivací [10], [15].

[2] Na první pohled by se mohlo zdát, že tento krok je celkem zbytečný, ale operátory selekce a křížení mohou někdy vést k tomu, že řešení spěje do lokálního extrému a nikoli do globálního. Je tedy žádoucí během řešení schválně nechat určité procento jedinců tzv. zmutovat (zmodifikovat) a tím je možnost se elegantně vymanit z nebezpečí dojítí do lokálního extrému

Postupná aktualizace odhadu optima je řešena iteračně. Prvním krokem tohoto algoritmu je test konvergence, který testuje, jestli aktuální odhad $\bar{x}_k$ optima už je přijatelný. Pokud je přijatelný, algoritmus končí a pokud není, tak pokračuje dalšími kroky, kdy je vypočten směr $\bar{p}_k$ hledání, kterým je posunut aktuální odhad optima a vypočtena délka kroku a_k, která je přičtena k aktuálnímu odhadu. Délka kroku nesmí být příliš malá, jelikož by výpočet trval dlouho, než by se dospělo do minima. Na druhé straně, pokud by délka kroku byla příliš velká, mohlo by dojít k přeskočení minima. Podmínkou je, aby funkční hodnota charakteristické rovnice byla menší než v původním bodě. Původní odhad optima je nakonec nahrazen novým, pokud dojde ke splnění této podmínky [15].

Minimum funkce je hledáno v opačném směru než gradient účelové funkce, jelikož gradient představuje směr nejstrmějšího růstu. Jinými slovy, při hledání minima je úmyslem co nejstrměji klesat a z toho důvodu je směr hledání volen jako záporná hodnota gradientu [15]

$$\overline{p}_k = -\overline{\nabla}_k. \tag{3.4}$$

Nový odhad optima $\bar{x}_{k+1}$ lze poté z původního odhadu získat následujícím způsobem, kdy výsledný algoritmus bude mít tuto podobu [15]

$$\overline{x}_{k+1} = \overline{x}_k - a_k\overline{\nabla}_k, \quad a_k \leq 1, \tag{3.5}$$

kdy hodnota délky kroku a_k se zpravidla na začátku algoritmu volí větší a postupně se snižuje [15].

3.2.2 Newtonova metoda

Newtonova metoda je založena na tom, že účelovou funkci aproximuje v okolí výchozího bodu kvadratickou funkcí a z této kvadratické aproximace spočítá její minimum, které je bráno další zpřesněný odhad optima. Kvadratická aproximace účelové funkce vychází z rozvoje funkce pomocí Taylorovy řady, kdy jsou brány v potaz pouze její první tři členy, kde figurují 1. a 2. derivace v daném bodě. Pokud se Taylorova řada přepíše do problému hledání minima, tak 1. derivace je nahrazena gradientem $\overline{\nabla}_k^T$ (parciálními derivacemi podle jednotlivých proměnných, pokud se jedná o vícerozměrný problém) a místo 2. derivace bude figurovat Hessova matice $\overline{\overline{G}}_k$ [15].

Je-li poté tato kvadratická aproximace, přepsána do problému hledání minima, derivována podle směru posunu $\bar{p}_k$, musí být 1. derivace nulová, aby byla zajištěna podmínka optimality 1. řádu, která říká, že v následujícím bodě $\bar{x}_{k+1}$ je minimum, maximum nebo sedlový bod. Hledaný směr posunu potom zřejmě bude mít tvar [15]

$$\overline{p}_k = -\overline{\overline{G}}_k^{-1}\overline{\nabla}_k, \tag{3.6}$$

z čehož vyplývá výsledná podoba Newtonova algoritmu [15]

$$\overline{x}_{k+1} = \overline{x}_k + \overline{p}_k = \overline{x}_k - \overline{\overline{G}}_k^{-1}\overline{\nabla}_k. \tag{3.7}$$

3.3 Optimalizace bez derivací

Jedná se o metody, které ke své realizaci nepoužívají derivace (či gradienty) minimalizovaných funkcí ani jejich aproximace. Obecně lze problémy řešené těmito metodami rozdělit na úlohy jednorozměrné a vícerozměrné optimalizace. Výhoda těchto metod je bezesporu jejich snadná implementace a schopnost najití minima při minimalizaci nediferencovatelných či nespojitých funkcí.

3.3.1 Náhodné prohledávání

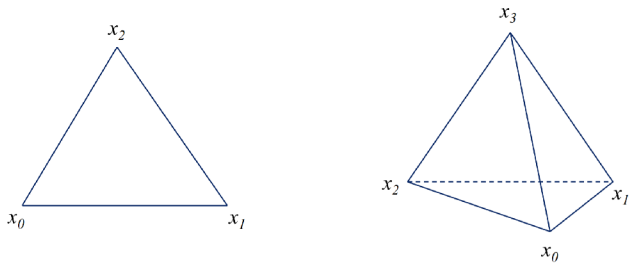
Metoda náhodného prohledávání, někdy označována jako „slepý algoritmus“, je vhodná pouze pro odhad optimální hodnoty. Spočívá v tom, že na vymezeném prostoru, kde se má nacházet extrém, který je hledán, se náhodně generují body. Spočítá se funkční hodnota náhodné polohy vygenerovaného bodu a pokud je menší než doposud nejmenší funkční hodnota, tak je zapamatována místo ní. Tato metoda je zejména používána v případě, kdy prohledávaný prostor je příliš veliký, aby byl detailně prohledáván celý. Algoritmus je ukončen tak, že se předepíše počet iterací [16].

3.3.2 Nelder-Meadův algoritmus

Nelder-Meadova metoda je modifikací základní simplexové metody. Simplexová metoda spočívá ve vygenerování simplexu S [17]

$$S = \{x_1, x_2 \dots x_{d+1}\}, \tag{3.8}$$

tvořeného množinou nekomplanárních $d + 1$ bodů v D-rozměrném prostoru. Jedná-li se o dvojrozměrný prostor, je simplex trojúhelník a v případě třírozměrného prostoru čtyřstěn, viz. obrázek 3.1.



Obr. 3.1: Simplexy v dvourozměrném (vlevo) a třírozměrném (vpravo) prostoru, převzato z [17]

V simplexu jsou nalezeny body s nejvyšší x_H a nejnižší x_L funkční hodnotou [17]

$$x_H = \arg \max_{x \epsilon S} f(x), \tag{3.9}$$

$$x_L = \arg \min_{x \epsilon S} f(x), \tag{3.10}$$

přičemž bod x_H s nejvyšší funkční hodnotou je nejméně kvalitní, jelikož je hledáno minimum a bude přepsán na lepší hodnotu. V Nelder-Meadově algoritmu je oproti klasické simplexové metodě nalezen ještě druhý nejhorší bod x_S.Těžiště g jako průměr d bodů, jež zůstanou po odstranění x_H, se spočítá jako

$$g = \frac{1}{d}\left(\sum_{x \epsilon S} x - x_H\right). \tag{3.11}$$

Iteračně je poté hledán nový bod y, který nahradí dosud nejhorší bod x_H simplexu, pokud je splněna podmínka $f(y) < f(x_H)$, tzv. **reflexí**

$$x_R = g + (g - x_H), \tag{3.12}$$

což je překlopení nejméně kvalitního bodu x_H přes těžiště g, přičemž jestliže jeho funkční hodnota $f_R = f(x_R)$ leží mezi funkčními hodnotami x_L a x_S, pak je tento bod akceptován. V opačném případě se počítá **expanze**

$$x_E = g + \gamma (x_R - g), \tag{3.13}$$

kdy se předpokládá, že $f_R < f_L$, čili bod x_H se překlopí do dvojnásobného intervalu od těžiště g do bodu x_R při standardní volbě expanzního faktoru γ určující míru expanze. Nový bod x_E má funkční hodnotu $f_E = f(x_E)$, která je porovnávána s f_R. Pokud je $f_E < f_R$, je akceptováno x_E, v opačném případě se akceptuje x_R, který je dále porovnáván s x_S, pokud má funkční hodnotu $f_R \geq f_S$. Nastává **kontrakce**

x_C jako jedna z následujících dvou možností. Pokud $f_S \leq f_R < f_H$, je počítán bod kontrakce x_{C1}

$$x_{C1} = g + \beta\left(x_R - g\right), \tag{3.14}$$

kdy s bodem x_H se nedojde až do x_R, ale pouze do poloviny vzdálenosti mezi těžištěm g a x_R při standardní volbě kontrakčního faktoru β. Nový bod má funkční hodnotu $f_{C1} = f(x_{C1})$, kdy pokud $f_{C1} \leq f_R$, je akceptován x_{C1}, jinak dochází k redukci. Druhá možnost při kontrakci je, že $f_R \geq f_H$, kdy je počítán bod kontrakce x_{C2}

$$x_{C2} = g + \beta\left(x_H - g\right), \tag{3.15}$$

kde $f_{C2} = f(x_{C2})$. Pokud $f_{C2} \leq f_H$, bod je akceptován, jinak dochází k redukci. Během **redukce** je spočítáno d nových vrcholů x_j

$$x_j = x_L + \delta\left(x_j - x_L\right), \tag{3.16}$$

pro $j = 0, 1, \ldots d$, když $j \neq L$, čili dojde k posunu všech bodů směrem k bodu x_L, což bude mít za následek zmenšení simplexu [17].

Nejhorší bod x_H je poté iteračně transformován do splnění ukončovací podmínky, kterou může být konvergence vrcholů simplexu k jednomu bodu, konvergence funkčních hodnot vrcholů simplexu nebo překročení počtu iterací nad stanovenou mez.

3.3.3 Simulované žíhání

Inspirace této metody je založena na procesu žíhání, kdy materiál, který má krystalickou mřížku, např. železo, se zahřeje na vysokou teplotu. Následně je teplota snižována, krystal se ochlazuje a probíhá tzv. žíhání, které je výsledkem stabilnější struktury krystalové mřížky s minimem defektů. Při simulováněm žíhání je zahřívaný krystal reprezentován vektorem $\mathbf{x}$, ke kterému se přiřadí „energie krystalu“ jako funkční hodnota $f(\mathbf{x})$, která je v daném bodě. Během žíhání se poté minimalizuje energie krystalu, čili minimalizuje se funkce. Stávající vektor $\mathbf{x}$ se s jistým působením náhodnosti přemění na nový vektor $\mathbf{x}'$ z okolí původního vektoru s pravděpodobností danou tzv. Metropolisovským vzorcem [10], [16]

$$P\left(x' \to x\right) = \begin{cases} 1 & f\left(x'\right) \leq f\left(x\right) \\ e^{-\frac{f(x')-f(x)}{T}} & f\left(x'\right) > f\left(x\right) \end{cases} \tag{3.17}$$

kde P symbolizuje pravděpodobnost, že nový bod x' nahradí předchozí x. Pravděpodobnost je rovna 1 za předpokladu, že funkční hodnota v novém bodě je menší nebo rovna funkční hodnotě v x a značí, že bylo dosaženo kvalitnějšího bodu a bod

je automaticky akceptován do dalšího procesu žíhání. V případě, že funkční hodnota v novém bodě je větší než funkční hodnota ve starém bodě, je zde nenulová pravěpodobnost akceptování nového bodu dána exponenciálním výrazem, kde konstanta T představuje analogii teploty, která se postupně snižuje. V tomto případě i nové horší řešení má nějakou šanci se prosadit a je zde tak možnost uniknout z lokálních minim. Konstanta T se vyskytuje ve jmenovateli exponentu, a proto když je na začátku procesu její hodnota nastavena poměrně velká, má za následek vyšší hodnotu pravděpodobnosti akceptování nového horšího řešení. V průběhu procesu je hodnota konstanty T pomale snižována, čímž je snižována i pravděpodobnost, že budou akceptována nová horší řešení [16, 12].

3.4 Evoluční algoritmy

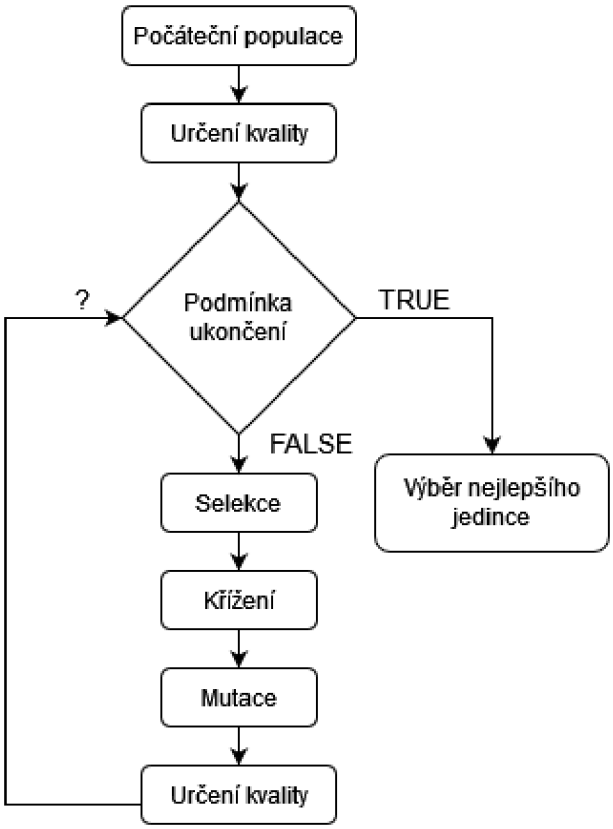
Princip této skupiny algoritmů byl také popsán v kapitole 3.1.2. Evoluční algoritmy (dále EA) jsou stochastické algoritmy, které modelují genetickou dědičnost, tzn. používají tři základní evoluční operátory (selekce, křížení a mutace). Pracují s populací řešení, kde jedinci v populaci představují jedno řešení dané úlohy. EA jsou iterační algoritmy, takže v každé iteraci bude vytvářená nová generace jedinců.

3.4.1 Genetické algoritmy

Jedná se o heuristické stochastické algoritmy, které využívají stochasticitu, ale zároveň modelují Darwinovy děje. Genetické algoritmy (dále GA) využívají paralely s evolucí, že znalost adaptace vůči vnějšímu prostředí je uchovávána v genetické informaci v chromozomech každého jedince. Tzn. v populaci řešení (jedinců) je každý jedinec reprezentován jedním chromozomem a pomocí genetických operací jsou předávány informace dalším jedincům (potomkům) v další generaci. Využíváno je také toho, že v přírodě přežívají silnější jedinci, kteří mají větší šanci předat svoji genetickou informaci potomkům. Cílem GA je tedy vytvářet v populaci stále silnější jedince, čili převedením na využití pro algoritmy, řešící optimalizační problémy, jsou hledány silnější jedinci z pohledu lepší hodnoty účelové funkce [18], [19], [20], [21].

GA mají hodnoty parametrů zakódovány v chromozomech, přičemž chromozom představuje jedno řešení (jedince) v prohledávaném prostoru. Chromozom je tvořen geny, což jsou jednotlivé kódované parametry a každý prvek genu je jedna alela. Matice jedinců s rozměry: počet jedinců $\times$ počet proměnných poté tvoří jednu generaci jedinců. V první generaci je sestavena výchozí populace náhodně. Během optimalizačního procesu je v každé další generaci pro každého jedince vyčíslena kvalita, tzn. hodnota účelové funkce, která vyjadřuje kvalitu řešení a je definována podmínka ukončení, což může být počet iterací nebo stagnace řešení. Následně v samotném

GA jsou prováděny genetické operace. Selekcí jsou vybráni zpravidla nejlepší jedinci dle jejich kvality, avšak mohou být také vybráni náhodně či použitím metod, kterými jsou např. metoda váhované rulety nebo turnajové selekce. Při křížení si zpravidla dva jedinci, které představují rodiče, vymění svoji informaci tak, že se zpravidla určí bod křížení a před nebo za bodem křížení jsou prohozeny dílčí řetězce, čímž vzniknou dva noví jedinci představující potomky. Mutací je v řetězci náhodně změněno určité procento alel. Nakonec je určena kvalita a zpětně testována podmínka, jestli už došlo ke splnění parametrů pro ukončení. Nakonec je vybrán nejlepší jedinec za předpokladu, že jedinec, který bude poskytovat nejlepší řešení, bude při minimalizaci ten, který má nejnižší funkční hodnotu účelové funkce. Postup GA je znázorněn ve vývojovém diagramu na obrázku 3.2 a stručném pseudokódu 1 [18], [19], [20], [21].



Obr. 3.2: Vývojový diagram GA, převzato z [18]

Algorithm 1 Zobecněný pseudokód GA [18]

```
inicializace populace
vyčíslení funkce kvality
while není splněno kritérium ukončení do
    výběr rodičů
    křížení s pravděpodobností
    mutace s pravděpodobností
    dekódování a výpočet funkce kvality
    výběr přeživších jedinců
    najítí nejlepšího jedince
end while
return nejlepší jedinec
```

GA mohou být v základu rozděleny na binární GA, kdy hodnoty parametrů účelové funkce jsou zakódovány binárně a spojité GA, kdy hodnoty parametrů představují reálná čísla.

3.5 Rojové algoritmy

Rojové algoritmy jsou základní ideou podobné genetickému algoritmu, ale postup jejich činnosti, tzn. jak se bude pracovat s jedinci a popř. vytvářet noví jedinci, jak bude daná optimalizace, je nepatrně odlišný. Dané optimalizační metody svůj předobraz berou v tzv. rojové inteligenci, tzn. snaží se napodobit chování nějakého zvířecího druhu, které je typické tím, že je v něm definována nějaká stádová inteligence.

3.5.1 Particle Swarm Optimalizace (PSO)

Metoda PSO, na rozdíl od GA, již nevyužívá evoluční operátory a neprovádí tvorbu nových potomků, ale uvažuje pohyb napříč prohledávaným prostorem a snahou je, aby se alespoň jeden jedinec dostal do globálního minima. Inspirací této metody se zpočátku stalo chování hejna ptáků, kdy hejno se pohybuje prostorem a hledá nejvyšší vrcholek, čili hejno svoji výšku nastavuje podle tohoto detekovaného nejvyššího vrcholku, tzn. je schopno svoji výšku upravit [8], [10], [22].

Na začátku algoritmu je náhodně umístěna populace jedinců (označováni také jako částice nebo agenti), kdy jejich pohyb, který je pro každou částici uchován ve vygenerovaném vektoru rychlosti $\bar{v}_j$, bude v iteracích náhodný. Stejně jako u GA je zapotřebí si spočítat kvalitu každého jedince, tedy v závislosti na jeho pozici $\bar{x}_j$ je spočítána funkční hodnotu účelové funkce. Pro nejkvalitnější částici, tzn. nejlepšího

jedince z počáteční populace, se uloží její pozice x_{LP} a funkční hodnota $f(x_{LP})$ tak, aby se s touto částicí mohlo dále pracovat. Zároveň jsou počáteční hodnoty uloženy do matice LJ, ve které pro každou částici bude její doposud nejlepší zapsaná pozice. V dalším kroku algoritmu je základní pohyb definován následovně [8], [10], [22], [23]

$$\bar{v}_j\left(t+1\right) = \bar{v}_j\left(t\right) + c_1\ rand\ \left(\bar{x}_{LP} - \bar{x}_J\left(t\right)\right) + c_2\ rand\ \left(\bar{x}_{LJ} - \bar{x}_J\left(t\right)\right), \tag{3.18}$$

$$\bar{x}_j\left(t+1\right) = \bar{x}_j\left(t\right) + \bar{v}_j\left(t+1\right), \tag{3.19}$$

kde rychlost jedince v další iteraci $\bar{v}_j\left(t+1\right)$ se skládá ze tří složek. Paměťová (individuální) složka $\bar{v}_j\left(t\right)$, kdy si částice pamatuje, jakou cestou se vydávala v předchozí iteraci. Další dvě složky jsou náhodné a složky rychlosti směřují podle směrového vektoru v závorkách v případě prvního směrového vektoru k nejlepší pozici $\bar{x}_{LP}$ v rámci celé populace (elitnímu jedinci ve všech iteracích - přizpůsobivá složka) a v případě druhého směrového vektoru k lokální pozici daného jedince $\bar{x}_{LJ}$ (konzervativní složka). Konstanty c_1 a c_2 jsou tzv. učící faktory a mohou ovlivňovat preferenci dané složky pohybu. Výsledná souřadnice každého jedince $\bar{x}_j\left(t+1\right)$ je dána součtem předchozí souřadnice $\bar{x}_j\left(t\right)$ a rychlosti $\bar{v}_j\left(t+1\right)$ [10], [22], [23].

Rychlost jedince má v průběhu běhu algoritmu tendenci se zvyšovat. Je záhodno jeho rychlost omezit, aby se částice nevzdalovaly od nejlepšího řešení. Zavádí se omezení v_{max}, aby rychlost nebyla povolena v rámci prohledávaného prostoru na příliš vysokou tak, aby nedošlo k opuštení částic z prostoru [23].

V dalších iteracích algoritmu tak dochází k postupnému shlukování částic v oblasti řešení, kdy částice, které ve shluku původně nejsou, jsou postupně ke shluku přitahovány. V závislosti na nastavení parametrů, tzn. váhování převažujících druhů pohybu, se PSO jeví jako efektivní algoritmus, který prohledávání prostoru a nalezení optima bude zvládat [23].

3.5.2 Včelí algoritmus (BA)

Algoritmus, který se inspiroval chováním včel, využívá rozdělení populace jedinců do menších skupin dílčích jedinců, které mají svoji funkci a při optimalizaci hrají určitou roli. Jak je známo z přírodních věd, u včel se uplatňuje specifické chování, kdy hierarchie v rámci včelího společenství je poměrně přísně organizována. Včelí algoritmus (dále BA) patří do skupiny metod inspirované sháněním potravy a včely v tomto algoritmu jsou vyčleněny na skupiny průzkumnic a vyčkávajících včel. Pro jednotlivé dvě skupiny jsou definovány základní vzorce chování, podle kterých se v každé iteraci provádí úprava souřadnic a posun k požadovanému řešení [24], [25], [26], [27].

Algoritmus BA se specificky inspiroval chováním včel v rámci opylování rostlin, které jsou v okolí definovaného úlu. Nejprve se z úlu vydají průzkumnice, které se v okolí úlu pohybují náhodně. Po jejich návratu do úlu, každá průzkumnice v závislosti na výsledku svého hledání provádí „taneční rituál“, kterým předává informaci o nalezišti, které našla, a o směru a vzdálenosti v jaké toto naleziště je. Takovým způsobem dochází k informování ostatních včel o nalezištích a ty se potom vydávají do takto otypovaných nalezišť a provádí sběr potravy. V rámci náhodného pohybu navíc provádí lokální prohledávání v okolí těchto otypovaných míst. V každé iteraci se vybere z celkového množství průzkumnic m nejlepších, které představují nejlepší řešení optimalizačního problému. U těchto m nejlepších včel se bude provádět lokální prohledávání a navíc je uplatňováno elitářství, kdy se vybere část elitních včel e a zbytek $m - e$ budou ostatní vybrané včely. Následně přichází na řadu vyčkávající včely, kdy každé z m nejlepších průzkumnic je přiřazen definovaný počet vyčkávajících včel a ty se začnou pohybovat v lokálním okolí dané průzkumnice ngh. Pokud dojde k vylepšení řešení, daná průzkumnice si svoje řešení aktualizuje. Včely, které nejsou vybrány jako nejlepší $n - m$, se náhodně doplní do počtu n jako nové průzkumnice a v další iteraci jsou z úlu vyslány a opět provádí počáteční odhad. Zároveň tedy dochází ke zlepšování stávajících řešení a generování nových náhodných bodů. Princip optimalizace spočívá v tom, že v definovaném lokálním okolí se zlepšuje pozice nejlepších řešení. Princip základního BA je znázorněn v pseudokódu 2 [24] [25], [26].

Algorithm 2 Pseudokód základního BA [24], [26],[27]

Inicializace počáteční populace n včel.
Výpočet fitness počáteční populace.
while není splněna podmínka **do**
    Výběr m nejkvalitnějších včel (lokalit).
    Určení velikosti okolí lokalit ngh.
    Náhodné umístění vyčkávajících včel do okolí a výpočet fitness.
    Výběr nejkvalitnějších jedinců z každého okolí.
    Doplnění $m - n$ náhodně rozložených průzkumnic a výpočet fitness.
    Nechej běžet algoritmus dokud nebudou splněna kritéria ukončení.
end while

3.5.3 Algoritmus umělé včelí kolonie (ABC)

Dalším algoritmem ze skupiny včelích algoritmů je algoritmus umělé včelí kolonie (dále ABC). V tomto případě se vyskytují tři skupiny včel, kterými jsou v rámci

hierarchie dělnice, vyčkávající včely a průzkumnice. Pro dělnice platí, že představují nějaké doposud stanovené řešení, které bude snaha v každém kroku zlepšit. Zlepšení bude jako u BA pomocí lokálního prohledávání. Pracující dělnice se budou snažit k sobě vábit vyčkávající včely, které budou svůj pohyb ke konkrétním dělnicím provádět na základě pravděpodobnostního přístupu vážené rulety. Tento výběr je tedy závislý na kvalitě včely, která se snaží vyčkávající včely navábit. Vyčkávající včela, jež si zvolí nějakou dělnici, se snaží zlepšit její řešení ještě pomocí lokálního prohledávání [8], [10], [28], [29].

Algoritmus ABC je tedy velmi podobný BA s tím rozdílem, že je proveden výběr počtu vyčkávajících včel na základě pravděpodobnostního přístupu a navábění k libovolné dělnici. Pokud dojde ke zlepšení pozice dané včely, dělnice si toto řešení zaktualizuje a popř. pokud ke zlepšení nedojde, setrvá na své pozici. Pokud nedojde u dané dělnice ke zlepšení pozice po několik iterací, tzn. bude se nacházet ve stabilním bodě, dělnice opustí svoji pozici a stane se průzkumnou včelou. V tom případě si stanoví náhodný bod a posune se do něj. Zároveň je zde udržována informace o nejlepší doposud nalezené pozici, s kterou se dále pracuje [8], [10], [28], [29].

3.5.4 Světluškový algoritmus (FA)

Světluškový algoritmus (dále FA), jak název napovídá, je inspirovaný pohybem světlušek, kdy dochází k hledání jednak potravy nebo nejbližšího partnera na základě vzájemného navádění jedinců (v tomto případě světlušek). Princip FA je takový, že je snaha namodelovat chování světlušek, kdy světlušky jsou schopny emitovat světlo a nastává u nich tzv. bioluminiscence. Na základě tohoto jevu dochází k definování pohybu a vzájemnému ohlivnění jednotlivých světlušek. Základní model chování je takový, že samičky, které mají více světelných orgánů, vydáváním světla lákají samečky, kteří si létáním okolo na základě svícení vybírají partnerky [31], [32], [33], [34].

V rámci optimalizačního algoritmu, který definuje pohyb jedinců, si lze zavést zjednodušení, že světlušky budou bezpohlavní, tzn. že všechny světlušky budou přitahovat všechny a jejich atraktivnost bude úměrná tomu, jaký vykazují jas, tzn. že světlušky, které svítí méně, budou přitahovány světluškami, které svítí více a bude definován pohyb k této jasnější světlušce. Dále hraje roli vzdálenost světlušek taková, že pokud v okolí konkrétní světlušky nebude žádná světluška, která svítí více než ona, tak pohyb bude v náhodné podobě. Jasnost světlušky, která je označována jako její kvalita, je úměrná funkční hodnotě pro optimalizovanou funkci na které je hledáno minimum nebo maximum [31], [32], [33], [34].

Popis FA podle [31] je varianta algoritmu All-to-All Adaptive, kdy jsou procházeny všechny světlušky každá s každou, přičemž vždycky jasnější světluška přitáhne

méně jasnou a jejich přitažlivost bude klesat s rostoucí vzdáleností mezi nimi. Mimo pohyb k jasnější světlušce bude definována i náhodná složka pohybu a její vliv bude klesat v závislosti na provedeném počtu iterací. Je tedy zjevné, že daná konkrétní světluška vykoná tolik pohybů, kolik jasnějších světlušek se nachází v jejím okolí. Jelikož se pracuje se seřazenou populací, tak dojde k tomu, že na konci každé iterace bude poslední pohyb směrem k nejlepšímu jedinci. Zároveň zde ale hraje důležitou roli vzdálenost, kdy pokud nejlepší světluška bude sice hodně jasná, ale bude velmi daleko od konkrétní světlušky, tak pohyb ve směru jasné světlušky bude velice malý a naopak preferovanější budou pohyby směrem ke světluškám, které jsou v blízkém okolí.

Pohyb světlušky i k jasnější světlušce j je popsán rovnicí

$$\bar{x}_i\left(t+1\right) = \bar{x}_i\left(t\right) + \beta_0 e^{-\gamma r_{ij}^2}\left(\bar{x}_j\left(t\right) - \bar{x}_i\left(t\right)\right) + \alpha\left(t\right)\bar{\varepsilon}\left(t\right), \tag{3.20}$$

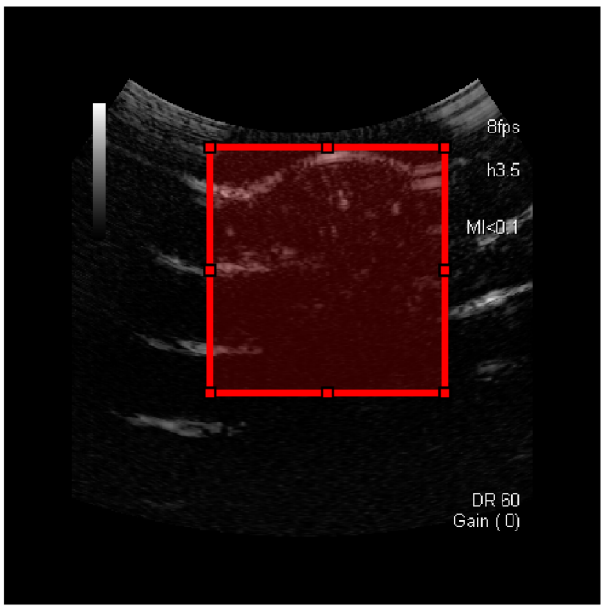
a lze jej rozdělit do tří složek. Pozice dané světlušky $\bar{x}_i$ v kroku $(t+1)$ vychází z pozice v předchozím kroku (t) a samotný pohyb je popsán dvěma složkami. První složka pohybu je pohyb ke světlušce $\bar{x}_j$, která je jasnější v jejím okolí. Tento pohyb je dán ve směru směrového vektoru, popsaného pozičními souřadnicemi těchto dvou světlušek, který je váhován konstantou β_0 a členem e, ve kterém se vyskytuje vliv vzdálenosti r_{ij}^2, kdy nejjednodušeji se uvažuje euklidovská a čím bude tato vzdálenost větší, tím menší hodnotu bude mít váha pohybu. Druhá složka pohybu představuje náhodný pohyb, který je definován na základě váhy náhodného pohybu α, která postupně s počtem iterací klesá a samotného náhodného pohybu ε, který si lze představit jako náhodnou proměnnou v obou směrech, která ke souřadnici přičte nebo odečte nějakou definovanou náhodnou hodnotu [31].

4 Registrace sekvence simulované scény

Následující kapitoly pojednávají a představují realizace registrace celých sad ultrazvukových snímků. Implementace vybrané kriteriální funkce, optimalizačních metod a následné vyhodnocení probíhalo v programovém prostředí MATLAB (konkrétní verze byla MATLAB R2020b). Z hlediska metod optimalizace, které tato práce porovnává, byl zvolen spojitý genetický algoritmus (GA), Particle Swarm Optimalizace (PSO), včelí algoritmus (BA), algoritmus umělé včelí kolonie (ABC) a světluškový algoritmus (FA). V závěru této kapitoly jsou zmíněné algoritmy inspirované přírodou porovnány s neevolučním heuristickým Nelder-Meadovým algoritmem.

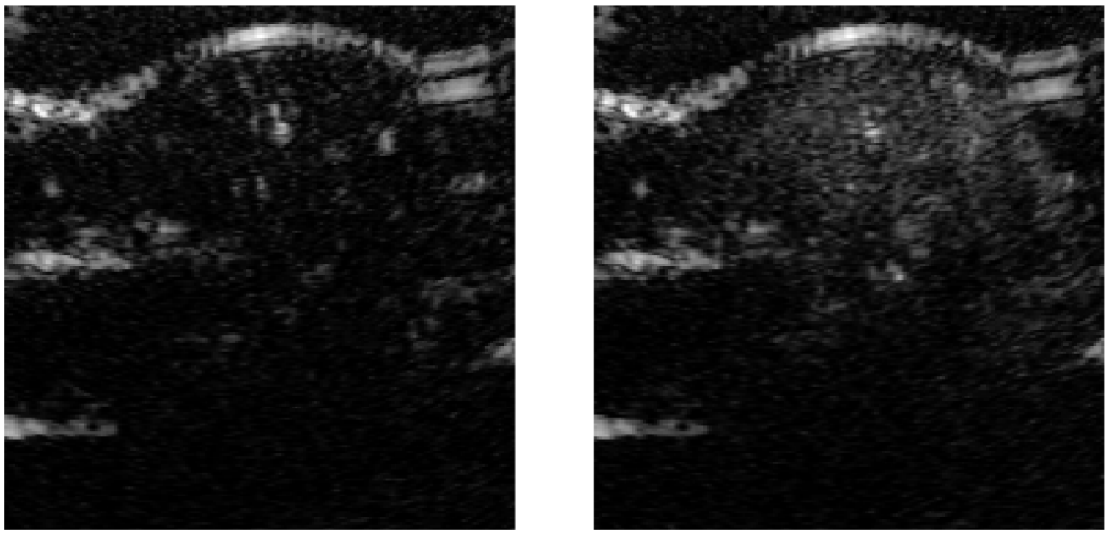
4.1 Základní informace o sekvenci

Ultrazvuková data, kterými se tato kapitola zabývá, jsou tvořena sekvencí šedesáti snímků, byla nasnímaná zvířecím ultrazvukem VEVO 2100 a byla vytvořena pro ověření snímací metody, která využívá kontrastní látku. Jedná se o snímky simulované scény, konkrétně uzavřeného obvodu tvořeného hadičkami a pumpou, jejíž centrální částí, a také oblast zájmu, která byla registrována, byl dialyzační filtr. Předem určenou oblast zájmu pro tuto sekvenci snímků tvořilo manuálně zvolené obdélníkové okno, jež je možné vidět na obrázku 4.1.



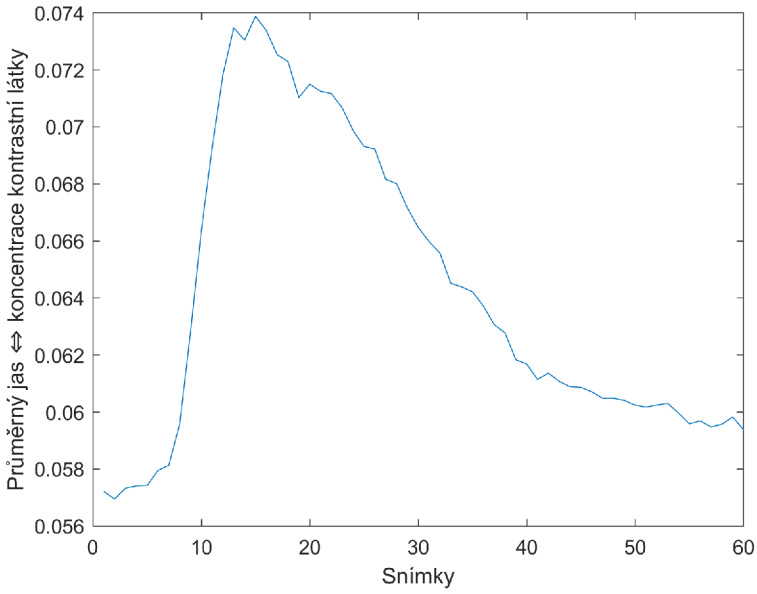
Obr. 4.1: Snímek ze sekvence s výřezem oblasti zájmu

To, co bylo snímáno, byl řez dialyzační patrony, a jelikož jí během experimentu proudila voda s kontrastní látkou, došlo v jeho průběhu k jejímu roztažení. Na obrázku 4.2 je poté možné zahlédnout, že v průběhu měření došlo k naplnění patrony kontrastní látkou, pokud je srovnán první snímek sekvence, kdy bylo zahájeno měření se snímkem, kde je patrona už naplněna kontrastní látkou. Snímek vpravo v oblasti řezu patrony se jeví jasnější, což značí zvýšené množství protékající kontrastní látky. Na snímky v tomto obrázku je již pro lepší přehlednost oblasti zájmu aplikován obdélníkový výřez.



Obr. 4.2: Srovnání množství kontrastní látky v obdélníkovém výřezu prvního (vpravo) a patnáctého (vlevo) snímku

Data byla nasnímána staticky, tzn. že sonda byla chycena do držáku, což zapříčinilo, že se scéna neměnila. Jediné, co podléhalo pohybu a v průběhu sekvence se měnilo, byl kontrast z kontrastní látky, tedy pohyb mikrobublin. Změnu její koncentrace napříč sekvencí snímků nabízí graf 4.3, který zobrazuje měnící se průměrnou jasovou hodnotu každého snímku z definovaného obdélníkového výřezu oblasti zájmu. Jak je z grafu patrné, koncentrace protékající kontrastní látky, tedy průměrný jas ve snímcích, nejdříve narůstá, maximální hodnoty dosahuje okolo 15. snímku a následně už pouze klesá. Za zmínku stojí i fakt, že změnu kontrastu v sekvenci snímků nemá za následek pouze množství protékající kontrastní látky, ale také pro ultrazvukové snímky typický šum speckle.



Obr. 4.3: Průběh koncentrace kontrastní látky napříč dynamické ultrazvukové sekvence snímků, ekvivalentní k průměrnému jasu

4.2 Předzpracování a transformace obrazu

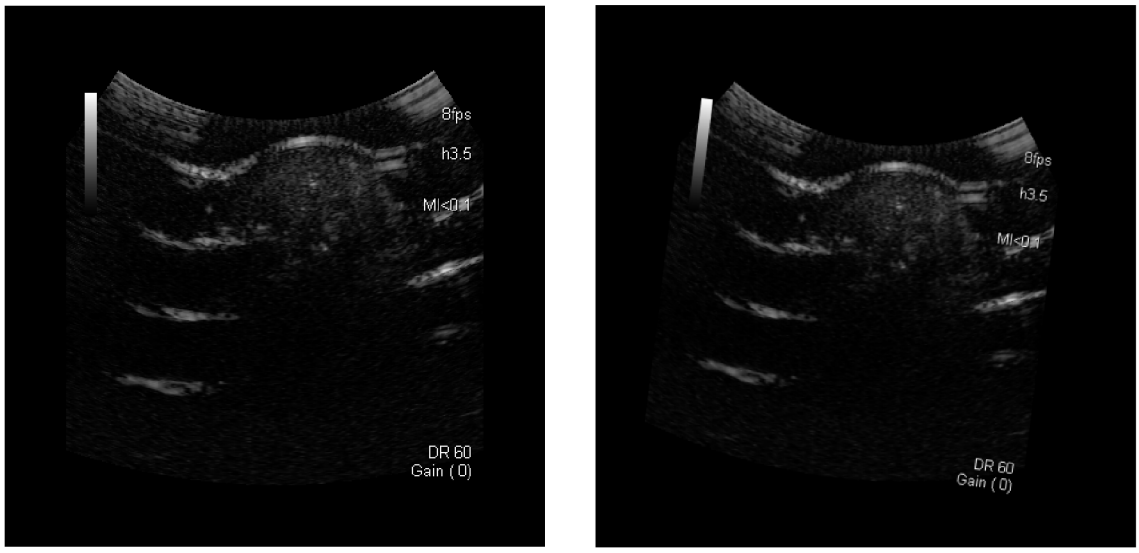
4.2.1 Transformace obrazu

Stěžejní úlohu při procesu registrace obrazů má výběr vhodné geometrické transformace pro slícování dvou obrazů. Nejjednodušší způsob takové transformace je posun a rotace. Další způsoby geometrické transformace byly zmíněny v kapitole 2.2. Jelikož data byla sama o sobě postižena typickým šumem speckle, byly z pohledu transformace obrazů pro tuto práci uvažovány právě již zmíněné posuny a rotace, což jsou tři parametry - posun v osách x, y a rotace, které byly detailněji popsány v kapitole 2.2.1, zabývající se rigidními transformacemi.

4.2.2 Vytvoření nové sekvence snímků

Pro účely registrace této sekvence snímků byla vytvořena funkce, která do původní sady obrazů naimplementovala umělý posun a otočení pomocí matlabovských příkazů `imtranslate` a `imrotate`. Z toho důvodu byl nejprve každý z 60 snímků převeden do formátu double, aby při vykonání zmíněných standardizovaných funkcí nedocházelo k možným chybám způsobených interpolací hodnot při vnášení obrázků do nové mřížky. Následně došlo k vložení obrázků do matice nul, která zajistila, aby nedošlo ke ztrátě důležitých informací. Pohyb, tedy posun a rotace, byl pro

každý snímek v sekvenci kromě prvního náhodně vygenerován v mezích, které byly nastaveny v rozmezí 20 pixelů pro posun v obou osách (minimální hodnota -10 a maximální +10) a úhel otočení pro rotaci v rozmezí 20° (minimální úhel −10° a maximální +10°). Následně bylo definováno obdélníkové okno a obrazy byly tímto oknem ořízнuty (viz. obrázek 4.1), aby bylo později možné snímky porovnat se snímky s nezaneseným pohybem v oblasti zájmu, tedy tam kde se vyskytoval pohyb kontrastní látky. Odstraněny tak byly části obrazů, které se v průběhu sekvence neměnily, jako např. colorbar vlevo nahoře, text v pravé části obrazů nebo vícenásobné odrazy směrem do hloubky. Pro možné potřeby vyhodnocení procesu registrace byly náhodně vygenerované hodnoty pohybu uloženy jako 3 vektory na výstupu funkce (2 vektory posunu a 1 vektor rotace) společně s oříznutými obrazy s oblastí zájmu, do kterých byl vnesen náhodně vygenerovaný pohyb. Podobu původního snímku v sekvenci a téhož, avšak transformovaného snímku je možné vidět na obrázku 4.4.



Obr. 4.4: Původní (vlevo) a transformovaný (vpravo) obrázek ze sekvence vložené do matice nul

4.3 Výběr registračního přístupu

Registraci podléhala vždy dvojice snímků ze sekvence. Nabízí se otázka, jakým způsobem na sebe snímky zaregistrovat. V úvahu připadají 3 základní registrační přístupy, které však sužuje řada problémů.

První nabízenou možností bylo registrovat příslušné snímky z umělé sekvence, do kterých byl vnesen vygenerovaný náhodný pohyb, na totožné avšak neposunuté snímky z původní sekvence. Jedná se však spíše o teoretický postup a v praktických

aplikacích se nevyužívá, jelikož jej nelze provést bez známé reference. Z toho důvodu přichází v úvahu další dva přístupy, které jsou již v praxi využívány.

Druhým uvažovaným přístupem byla registrace na předchozí snímek v sekvenci, čili jakmile byl zaregistrován aktuální obraz, další snímek v sekvenci byl registrován na tento poslední registrovaný snímek. Značné úskalí tohoto přístupu však představuje skutečnost, že pokud daný optimalizační algoritmus selže na některém ze snímků sekvence, tzn. daný snímek nebude přesně lícovat na předchozí, chyba způsobená na tomto snímku se promítne dále i do dalších snímků.

Jako pravděpodobně nejlepší možnost se jeví postupně registrovat všechny snímky na první, referenční obraz v sekvenci. Z grafu 4.3 je však patrné, že v sekvenci se v závislosti na množství protékající kontrastní látky mění jas v obrazech. Pokud tedy bude srovnán referenční první snímek, který je bez kontrastu s těmi, kde se kontrast taktéž příliš neprojeví (obrázky na začátku a konci sekvence), registrace by měla při optimálně zvolených parametrech proběhnout v pořádku. Avšak při jeho srovnání se snímky s výraznějším jasem, může dojít k selhání registrace, jelikož bude registrován tmavý objekt na světlý. Tento fakt představuje značnou komplikaci při optimalizačním procesu pro nalezení hledaných optimálních parametrů, nicméně v kontextu této práce je zapotřebí s ním počítat.

Tento způsob registrace byl v práci využit, navíc pro tuto sekvenci s tím rozdílem, že do všech snímků kromě prvního byl zanesen náhodný pohyb v podobě posunu v osách x, y a rotace (viz. kapitola 4.2.2). Takto posunutá sekvence byla po proběhnutí optimalizačních algoritmů s tímto typem registrace ve vybrané oblasti zájmu srovnána s původní sekvencí a byla vyhodnocena úspěšnost registrace pro každé nastavení parametrů na celkem třech bězích.

4.3.1 Výběr optimálních parametrů a vyhodnocení sekvence

Výběr parametrů pro použité algoritmy optimalizace v této práci probíhal v tom duchu, že od základní pozice, které byly získány z odborných publikací, literatury nebo doporučení, byl vždy jeden z parametrů měněn [1] a ostatní zůstaly fixované. Daný měněný parametr byl nejprve zvýšen o určitý procentní bod, následně naopak snížen a bylo pozorováno, zda v některém z nastavení došlo ke zlepšení či nikoliv. Následně na základě tohoto prvotního testování pokud došlo ke zlepšení, a zároveň došlo ke zlepšení i u jiného parametru, který byl měněn od své základní pozice, bylo odzkoušeno toto nastavení parametrů, zda v kombinaci dohromady dávají ještě lepší výsledek než předchozí nastavení. Jelikož jsou využité optimalizační algoritmy stochastické, každé takovéto nastavení parametrů bylo spuštěno na třech bězích.

[1] v případě výběru počtu jedinců a iterací byly tyto parametry měněny zároveň a byl sledován rozdíl

Co se týče vyhodnocení a srovnání použitých algoritmů v této práci, nabízí se výpočet odchylky skutečných hodnot od těch, které vyšly optimalizačními algoritmy. Avšak tento způsob vyhodnocení má své úskalí, že musí být známa reference (tedy skutečné hodnoty), což nelze zajistit u sekvencí s reálnými daty, kterými se zabývají další kapitoly. Jako pravděpodobně nejlepší možnost srovnání pro tento daný typ úlohy nabízí výpočet střední kvadratické chyby (MSE), kterou lze využít i pro sekvence, kde nelze zajistit známost referenčních hodnot. Vyhodnocení tedy v tomto případě pro tuto sekvenci probíhalo tak, že náhodně posunuté snímky, zaregistrované na referenční první snímek, které byly posunuty a natočeny o nalezené parametry transformace, byly porovnávány s prvním referenčním obrázkem v původní neposunuté sekvenci ve vybrané oblasti zájmu.

4.4 Volba kriteriální funkce

Nezbytnou součástní každého optimalizačního procesu a taktéž registrace je zvolit vhodné kritérium podobnosti, které popisuje kvalitu zarovnání a vyjádří, že dané dva obrazy na sebe lícují a jsou stejné. Kritérium vychýleného obrazu bylo počítáno k referenčnímu obrazu. Při implementaci algoritmu bylo rozhodnuto obsáhnout funkci, která popisuje transformaci obrazu (posun a rotaci) a funkci, která popisuje kritérium, jak jsou si obrazy podobné, do jedné funkce.

Jako kritérium podobnosti obrazů byla dle článku [11] a [12] využita metrika střední kvadratické odchylky (MSE) (viz. kapitola 2.4.1, vzorec 2.21), která počítala podobnost mezi prvním referenčním obrázkem a snímky z náhodně posunuté sekvence, vytvořené v kapitole 4.2.2 a pouze v oblasti zájmu. Jelikož byla cílem optimalizace minimalizace zvoleného kritéria, byla metrika MSE během optimalizační části algoritmu minimalizována.

Dalšími kriteriálními funkcemi používané u registrace ultrazvukových snímků jsou kritéria založená na korelaci a přesněji kosínové kritérium, které bylo popsané v kapitole 2.4.2 a pravděpodobně nejpokročilejší je kritérium vzájemná informace, vycházející z definice entropií, bylo popsáno v kapitole 2.4.3, které k registraci snímků použila také studie [11] anebo [9]. První z těchto dalších zmíněných kritérií bylo odzkoušeno v předcházející práci při registraci jednoho snímku na tentýž, ale náhodně vychýlený, avšak dosahovalo horších výsledků, co se týče nalezených parametrů optimalizace. Zároveň čas, který algoritmu s touto kriteriální funkcí trvalo než dosáhlo stejného počtu iterací nebo vzorkovacích kroků (v případě úplného prohledávání), byl téměř dvojnásobný oproti stejnému nastavení s kriteriem MSE.

4.5 Registrace s optimalizací spojitého genetického algoritmu (GA)

Genetické algoritmy simulují Darwinovu teorii o vývoji druhů. Zejména je kladen důraz na to, aby do další generace postupovali pouze kvalitnější jedinci, tzn. v rámci iteračního procesu se jednotlivé generace zkvalitňují. Jsou prováděny genetické operace - selekce, křížení a mutace (podrobnější popis viz. kapitoly 3.1.2 a 3.4.1) a iterační algoritmus běží do splnění podmínky (většinou se udává maximální počet iterací, který byl také při realizaci použit). Matice jedinců je v případě spojitých GA tvořena reálnými čísly.

4.5.1 Stanovení parametrů algoritmu spojitého GA

U spojitých GA se obvykle genetické operace selekce a křížení spojují do jednoho bloku, který se nazývá reprodukce. Selekce v rámci tohoto bloku probíhala pomocí vážené rulety, která byla řízena parametrem β vycházejícího z Boltzmannova rozdělení, který kontroloval pravděpodobnost výběru rodičů v populaci na základě jejich hodnot kriteriální funkce a zajistil, že lepší řešení mají větší šanci být vybrány jako rodiče [35]. Křížení bylo uniformní, čili probíhalo na základě náhodně vygenerované masky, kdy byly vždy náhodně vybráni dva jedinci, kteří byli reprodukování na potomky. Parametr p_C udával procento potomků vytvořených při procesu překřížení z rodičovské populace a parametr γ váhoval jednotlivé dvojice jedinců při uniformním křížení a dopomohl průzkumu algoritmu. Mutace závisí na velikosti populace. Proto se pravděpodobnost mutace μ udává v procentech a testuje, jestli k mutaci došlo nebo ne tak, že je srovnávána s náhodně vygenerovaným číslem. V případě, že došlo k mutaci jedince, parametr δ váhoval náhodnou změnu této pozice jedince. Prostor, ve kterém byly hledány parametry, byl omezen při generování počáteční populace ve stejných mezích, jako byly generovány náhodné posuny a rotace při vytváření nové sady snímků v kapitole 4.2.2. Výchozí doporučené nastavení hodnot parametrů vychází z nejčastěji se objevujících nastaveních konkrétního parametru ve studiích [14], [21], [36] a zejména [37], zabývajícími se aplikací GA algoritmu. Výchozí nastavení parametrů je vypsáno v tabulce 4.1.

Tab. 4.1: Výchozí nastavení parametrů spojitého GA

Parametr GA	**Výchozí hodnota**
Velikost populace	50
Maximální počet iterací	30
β	1
p_C	0,95
γ	0,1
μ	0,05
δ	0,1

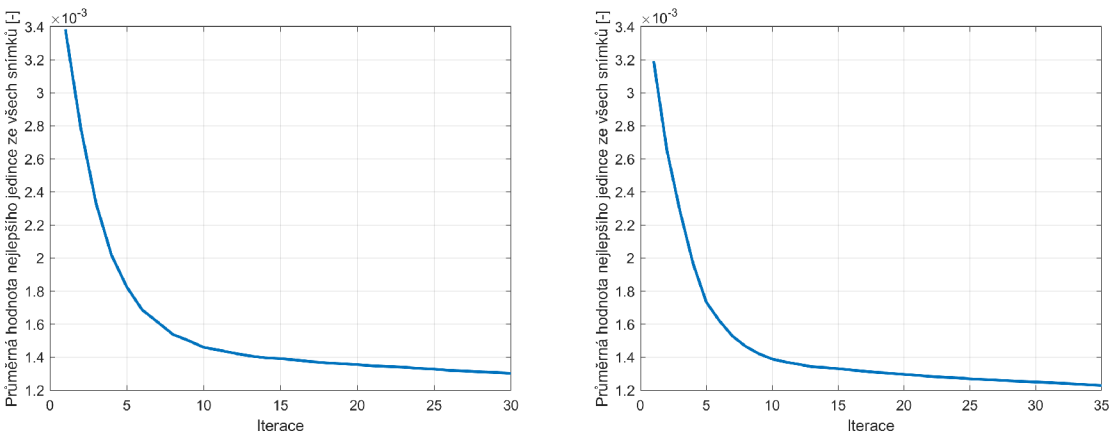
Studie [37] uvádí, že pro GA je obecně doporučeno mít menší velikost populace, ale zato větší počet iterací. Při prvotním odzkoušení parametrů však bylo zjištěno, že lepší výsledky z hlediska poklesu kriteriální funkce a MSE dává pro tento optimalizační problém a tyto data pravý opak.

Nejprve byly měněny ty parametry, kde je předpoklad, že pokud se bude zvyšovat jejich hodnota, dojde k nalezení lepšího řešení. Zvýšení velikosti populace nepředstavuje pro GA tak značné změny ve výpočetní náročnosti, jako je tomu u dalších použitých algoritmů. Pokud se velikost populace zvýší na dvojnásobnou hodnotu, měla by výpočetní náročnost být také dvojnásobná. Dále z pohledu kriteriální funkce, jejíž hodnota by měla během optimalizačního procesu klesat, již nemá smysl z hlediska výpočetního času brát v úvahu takový počet iterací, který bude mít za následek pouze její pomalý pokles nebo dokonce stagnaci.

Další kvantifikace probíhala tak, že pokud nedošlo při společném navýšení počtu jedinců a iterací o zhruba 20 %[2] z jejich výchozích hodnot k poklesu kriteriální funkce o více jak 20 %, bylo prozatím bráno v potaz základní nastavení těchto parametrů v tabulce 4.1. Vyhodnocení probíhalo na průměrné hodnotě hodnoty poklesu nejlepšího jedince ze všech snímků ze tří běhů algoritmu.

Jak je z obrázku 4.5 partné, při zvýšení počtu jedinců a iterací zároveň došlo pouze k minimálnímu dodatečnému poklesu kriteriální funkce. Nejnižší průměrná hodnota nejlepšího jedince v základním nastavení dosahuje po 30 iteracích ze tří běhů hodnoty $1{,}302 \cdot 10^{-3}$, zatímco při navýšení počtu 60 jedinců a 35 iterací $1{,}229 \cdot 10^{-3}$, což nevyhovuje požadovanému snížení o 20 %, které by mělo činit zhruba $1{,}042 \cdot 10^{-3}$. Navíc průměrný výpočetní čas byl pro základní nastavení 368 sekund, kdežto pro zmíněné dva zvýšené parametry 491 sekund, což činí v průměru více jak 2 minutový rozdíl ve výpočetním čase, kterým nelze kompenzovat tak malou změnou poklesu účelové funkce.

[2] zvýšení hodnot parametrů počtu jedinců a iterací o 20 % bylo pouze přibližné a zaokrouhleno na číslo, které bylo možné dělit 5



Obr. 4.5: Srovnání průměrné hodnoty poklesu nejlepšího jedince ze všech snímků zprůměrované ze 3 běhů pro výchozí nastavení 50 jedinců a 30 iterací (vlevo) a zvýšenou populaci 60 jedinců a 35 iterací (vpravo)

Nyní bylo na řadě měnění ostatních parametrů, u kterých dávala smysl jejich obměna a sledovat, zda dojde k dosažení lepšího výsledku. Předmětem sledování již nyní byla střední kvadratická chyba (MSE) mezi prvním a registrovaným snímkem na tento referenční, konkrétně průměrná minimální dosažená hodnota na každém z šedesáti snímků a směrodatná odchylka, u kterých bylo porovnáváno, zda při změně parametrů z výchozího nastavení dojde k jejich snížení. V případě spojitého GA bylo dále rozhodnuto z výchozího nastavení měnit hodnotu parametru p_C pro uniformní křížení a parametr μ pro mutaci. Hodnota těchto parametrů byla z jejich základních pozic zvýšena a snížena o 20 %, pokud to bylo možné. V přídadě, že se jednalo o procenta a hodnota již byla kolem 100 %, došlo ke zvýšení hodnoty pouze na 100 % a v případě některých celočíselných parametrů či velmi nízkých hodnot byla změna modulována jinak. Pokud dával lepší výsledek víc jak jeden měněný parametr, tak na základě tohoto prvotního testování byly vychýleny oba zároveň a bylo znovu sledováno, zda došlo k dalšímu zlepšení.

Ze zapsaných hodnot v tabulce 4.2 lze konstatovat, že při změně pouze jednoho z parametrů vybraného pro vychýlení ze svého základního nastavení dosahovaly menší průměrné hodnoty MSE a směrodatné odchylky změny parametrů $p_C = 1$ a $\mu = 0{,}1$.

Tab. 4.2: Srovnání změny průměrné střední kvadratické chyby (MSE) a její směrodatné odchylky při vychýlení vždy jednoho parametru z výchozího nastavení ze tří běhů algoritmu spojitého GA

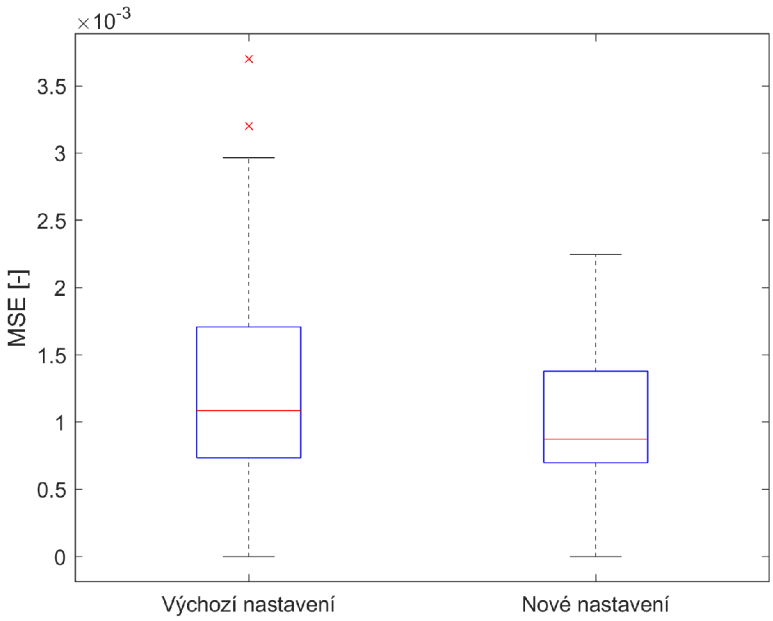
Měněný parametr	MSE [10^{-3}]	Směrodatná odchylka
Žádný	1,0741	0,00053
$p_C = 1$	1,0451	0,00051
$p_C = 0{,}75$	1,0838	0,00056
$\mu = 0{,}1$	1,0549	0,00050
$\mu = 0{,}01$	1,0622	0,00053

Z toho důvodu byly následně p_C a μ vychýleny ze základního nastavení současně, kdy v tabulce 4.3 byla zaznamenána nejníže dosažená průměrná hodnota MSE a směrodatné odchylky pro dané nastavení ze tří běhů tohoto algoritmu.

Tab. 4.3: Srovnání nastavených parametrů, střední hodnoty a směrodatné odchylky MSE a výpočetní náročnosti pro výchozí a nově zvolené nastavení spojitého GA

Nastavené parametry	Výchozí nastavení	Nové nastavení
Velikost populace	50	50
Maximální počet iterací	30	30
β	1	1
p_C	0,95	1
γ	0,1	0,1
μ	0,05	0,1
δ	0,1	0,1
MSE [10^{-3}]	1,0741	1,0403
Směrodatná odchylka	0,00053	0,00049
Průměrný výpočetní čas [s]	368	386

Na obrázku 4.6 je boxplotem vyjádřen rozptyl hodnot MSE pro výchozí a nově stanovené nejlepší nastavení, které se liší jen minimálně. Pokud je předmětem srovnání časová výpočetní náročnost, vychází lépe výchozí nastavení o 18 sekund, kdy novému nastavení trvala optimalizace v průměru 386 sekund. Tato změna se však dala očekávat vlivem změn obou parametrů, kdy větší procento jedinců v každé iteraci podléhá zároveň procesu křížení a mutaci. Tato změna však není nijak zásadní a je možné toto nové nastavení považovat za nejlepší dosažené, pokud je předmětem zájmu pro dosažení optimálního výsledku co nejkratší výpočetní čas.



Obr. 4.6: Srovnání rozptylů průměrné MSE spojitého GA pro výchozí nastavení z tabulky 4.1 (vlevo) a nově stanoveného nejlepšího nastavení z tabulky 4.3 (vpravo)

4.6 Registrace s optimalizací PSO

Algoritmus PSO provádí řešení optimalizačního problému pomocí roje částic, kdy jsou definovány předpisy pro pohyb jednotlivých částic a v jednotlivých iteracích dochází k úpravě celé populace jedinců. Narozdíl od GA nerozlišuje vývoj mezi rodiči a potomky, ale spíše se dívá na populaci jedinců nějakého živočišného nebo rostlinného druhu, která se v rámci prohledávaného prostoru definovaným způsobem pohybuje a z pohledu kvality řešení kriteriální funkce, představuje nejlepší jedinec řešení.

Pro každého jedince je uložena jeho konkrétní pozice v dané iteraci s jeho vektorem rychlosti, což představuje směrový vektor, kam se jedinec posune a s hodnotou nejlepší pozice, kterou každý jedinec doposud měl. Zároveň se nese informace o globální nejlepší pozici, tzn. o elitním jedinci. Pohyb, resp. změna rychlosti pro aktuální iteraci, se skládá ze tří složek. Částečně zůstane pohyb ve směru předchozího pohybu s tím, že je váhovaný definovanou vahou. Dále se projeví dvě složky, kdy první je vůči pozici daného jedince, která byla nejlepší v předchozích iteracích a druhá složka pohybu je vůči doposud nejlepšímu jedinci (podrobněji viz. vzorec 3.18). Pro pozici jedince platí, že se ze svojí pozice posune ve směru směrového vektoru do nové pozice (viz. vzorec 3.19).

4.6.1 Stanovení parametrů algoritmu PSO

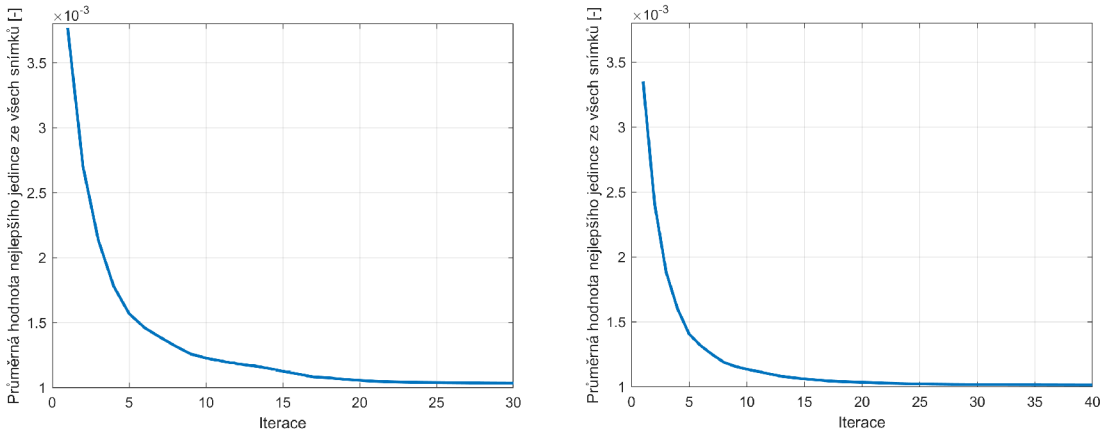
V definovaném počtu iterací je procházen každý jedinec, kdy na počátku je jejich pozice náhodná s uniformní distribucí. Pohyb každého jedince se poté skládá z dílčích složek, z nichž první je částečný pohyb ve směru původního, který je váhován konstantou w od 0 do 1 a jejím koeficientem utlumení w_{damp}, který se vzrůstajícím počtem iterací snižuje váhu této konstanty [38]. Dále jeho aktuální pozici ovlivňuje jeho nejlepší pozice v předchozích iteracích, jež je váhována konstantou c_1 a pozice nejlepšího jedince v celé populaci, která je váhována konstantou c_2. Jak c_1, tak c_2 mají náhodný charakter. Následně dochází k sumaci těchto tří složek, což dohromady dává výsledný pohyb daného jedince. Na základě toho je upravena velikost rychlosti a změna poziční informace. Společně s tím se vyhodnocuje, jestli dojde ke zlepšení poziční informace do bodu, který by představoval nové nejlepší nalezené řešení pro konkrétní pozici. Po projití všech jedinců se z těchto nejlepších řešení stanoví, jestli se zároveň zlepšila i hodnota nejlepšího celkového řešení. Hranice prohledávaného prostoru proměnných byla u algoritmu PSO nastavena ve shodných mezích, ve kterých byla generována nová sekvence snímků se synteticky přidaným pohybem v kapitole 4.2.2. Iniciální nastavení parametrů v tabulce 4.4 bylo vybráno na základě nejčastěji zastoupených hodnot konkrétních parametrů ze studií [8], [10], [37], zabývající se optimalizací PSO.

Tab. 4.4: Výchozí nastavení parametrů PSO

Parametr PSO	**Výchozí hodnota**
Počet jedinců	20
Maximální počet iterací	30
c_1	2
c_2	2
w	0,8
w_{damp}	0,99

PSO pracuje s tím, že bere populaci jedinců a v základní použité podobě ji porovnává k jednomu vůdci. Je tedy postačující pouze jednou za iteraci aktualizovat pozici vůdce, která je po celou dobu probíhání iterace konstatní, a k ní pouze připočítávat jedince s náhodnýma váhama. Iterace a celkově výpočetní část algoritmu se tedy takto oredukuje o další jeden cyklus. Tím pádem nepředstavuje pro PSO zvýšení velikosti populace tak značné dodatečné výpočetní nároky. Je však otázkou, zda zvýšení počtu částic povede u PSO k výraznému poklesu kriteriální funkce. To samé platí i u navýšení počtu iterací, které pokud nepovede k výraznějšímu zlepšení výsledku, nebude bráno v potaz.

Tak jako v případě spojitého GA, i zde nenastalo výrazné zlepšení výsledku v podobě poklesu hodnoty kriteriální funkce, při navýšení počtu částic a iterací přibližně o 20 % [3] oproti výchozích hodnotám, o čemž se lze přesvědčit z obrázku 4.7. Průměrná hodnota kriteriální funkce ze všech snímků po doběhnutí všech iterací ze 3 běhů činila $1{,}032 \cdot 10^{-3}$ pro výchozí nastavení a $1{,}015 \cdot 10^{-3}$ pro zvýšenou populaci 25 jedinců a 40 iterací, což je zhruba pouhé 1,5 % zlepšení, které když je brán v potaz průměrný výpočetní čas, který z původních 151 sekund stoupnul o více jak 1,5 minuty na 250 sekund, je pouze zlepšení zanedbatelné.



Obr. 4.7: Srovnání průměrné hodnoty poklesu nejlepšího jedince ze všech snímků zprůměrovano ze 3 běhů pro výchozí nastavení 20 jedinců a 30 iterací (vlevo) a zvýšenou populaci 25 jedinců a 40 iterací (vpravo)

Postup při hledání ostatních optimálních parametrů geometrické transformace byl tak jako v případě spojitého GA na bázi vychylování jednotlivých parametrů ze své základní pozice jeden po jednom a bylo sledováno, zda během tří běhů každého nastavení došlo ke zlepšení výsledku. Změna hodnot jednotlivých parametrů z jejich výchozích byla přibližně rovna 20 % výchylce anebo byly odzkoušeny jiné užité hodnoty ve výše zmíněných studiích, ze kterých bylo vybíráno původní nastavení. Na základě nižších dosažených průměrných hodnot minim a směrodatných odchylek MSE než výchozí hodnoty ze všech šedesáti snímků v tabulce 4.5 (jak je z výsledků patrné, v tomto případě byly brány v potaz pouze průměrné hodnoty MSE, jelikož směrodatné odchylky se lišily pouze minimálně), byly k dalšímu testování vybrány parametry $c_1 = 1{,}7$, $c_2 = 1{,}7$ a $w = 0{,}6$. Parametr w_{damp} na základě výsledků nebyl

[3] zvýšení hodnot parametrů počtu částic a iterací o 20 % bylo pouze přibližné a bráno takové nejblíže možné, které po navýšení o 20 % z výchozí hodnoty bylo možné dělit 5 a zároveň bylo vyšší než navýšení o 20 %

nijak dále měněn, čímž byla ponechána jeho značně malá váha tlumení konstanty w se vzrůstajícím počtem iterací.

Tab. 4.5: Srovnání změny průměrné střední kvadratické chyby (MSE) a její směrodatné odchylky při vychýlení vždy jednoho parametru z výchozího nastavení ze tří běhů algoritmu PSO

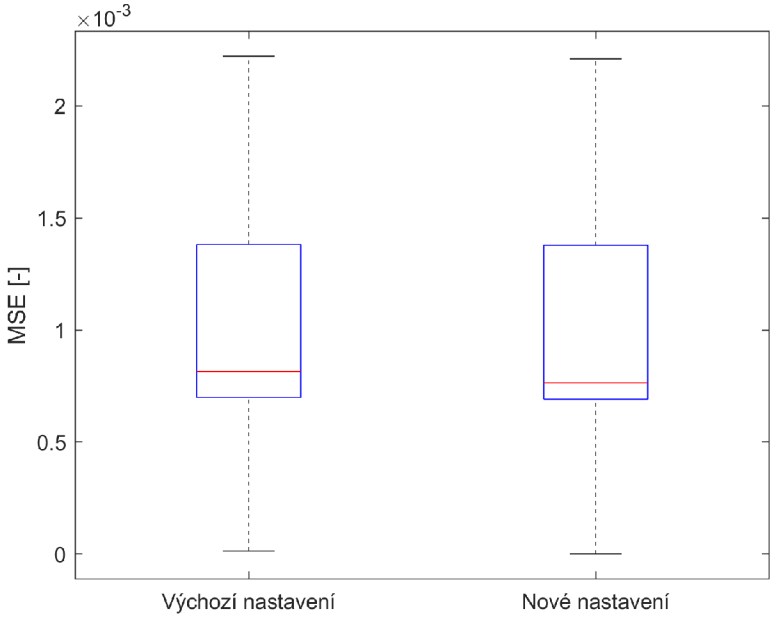
Měněný parametr	MSE [10^{-3}]	Směrodatná odchylka
Žádný	1,01028	0,000493
$c_1 = 2{,}5$	1,01365	0,000493
$c_1 = 1{,}7$	1,00965	0,000493
$c_2 = 2{,}5$	1,01316	0,000494
$c_2 = 1{,}7$	1,01010	0,000493
$w = 1$	1,02269	0,000492
$w = 0{,}6$	1,00889	0,000493
$w_{damp} = 1$	1,01651	0,000492
$w_{damp} = 0{,}9$	1,02081	0,000495

Jelikož současné vychýlení nově zvolených hodnot parametrů c_1, c_2 a w dohromady nepřinesly zlepšení v podobě průměrné hodnoty MSE a její směrodatné odchylky napříč všemi snímky, byly na sekvenci odzkoušeny další kombinace těchto parametrů, přičemž některá ze zmíněných nově nastavených hodnot zůstala zachována ve výchozím nastavení. Jednotlivá nastavení s nově odzkoušenými kombinacemi parametrů a jejich výsledky jsou uvedeny v tabulce 4.6. Nastavení "Nové 1" představuje vychýlení všech zvolených parametrů z výchozí pozice a ostatní nastavení jsou poté kombinace, kdy některý z parametrů byl ponechaný v základním nastavení.

Tab. 4.6: Srovnání nastavených parametrů, střední hodnoty a směrodatné odchylky MSE a výpočetní náročnosti pro výchozí a nově zvolené nastavení PSO

Nastavené parametry	Výchozí	Nové 1	Nové 2	Nové 3	Nové 4
Počet jedinců	20	20	20	20	20
Maximální počet iterací	30	30	30	30	30
c_1	2	1,7	2	1,7	1,7
c_2	2	1,7	1,7	2	1,7
w	0,8	0,6	0,6	0,6	0,8
w_{damp}	0,99	0,99	0,99	0,99	0,99
MSE [10^{-3}]	1,01028	1,00955	1,00880	1,00948	1,00957
Směrodatná odchylka	0,000493	0,000493	0,000493	0,000493	0,000493
Průměrný čas [s]	151	150	150	149	150

Z výsledků je patrné, že kombinace parametrů v nastavení "Nové 2" zaznamenala nejnižší dosaženou průměrnou hodnotu MSE ze třech realizací algoritmu PSO. Pro úplnost srovnání výchozího a nově zvoleného nastavení, byl vykreslen boxplot na obrázku 4.8, který stejně jako v případ spojitého GA vykazuje pouze minimální odchylky v rozptylu hodnot MSE. To může svědčit již o vhodně zvoleném iniciálním nastavení parametrů, které s ohledem na co nejkratší výpočetní čas registrace dosahovalo již téměř nejlépších možných výsledků, co se týče hodnoty poklesu kritéria a střední kvadratické chyby (MSE). Jedná se však o stochastický algoritmus, který na počátku generuje náhodné jedince a v průběhu iterací přidává další složku náhodnosti v podobě náhodných vah, takže je zde určitá pravdpodobnost, že nejlepšího možného řešení z hlediska času nebylo dosaženo vlivem omezeného počtu spuštění konkrétní realizace parametrů.



Obr. 4.8: Srovnání rozptylů průměrné MSE algoritmu PSO pro výchozí nastavení z tabulky 4.4 (vlevo) a nově stanoveného nejlepšího nastavení z tabulky 4.6 (vpravo)

4.7 Registrace s optimalizací včelího algoritmu (BA)

Včelí algoritmus (BA) pracuje v konečném okolí nejlepších včel. Inspirací se stalo vyhledání potravy v rámci včelího úlu, kdy se pracuje se stanoveným počtem n včel průzkumnic, z nichž je vybrána frakce m nejlepších včel, v jejímž okolí bude docházet k lokálnímu prohledávání. Z těchto nejlepších včel se vyberou elitní včely

e a pro každou včelu, která bude prohledávat, budou definované počty vyčkávacích včel, které budou prohledávat její okolí.

Na začátku každé iterace algoritmu je počáteční populace definována náhodně a pracuje se se seřazenou populací podle kvality, kdy se z celkového množství n včel uvažuje m nejlepších a provádí se lokální prohledávání ve stanoveném okolí, který je zapsán parametrem r a dává informaci o rozsahu lokálního prohledávání. Pokud dojde ke zlepšení ve stanoveném okolí, zlepšená pozice se pro m včel zapíše. Zbývající počet $n - m$ včel je poté generováno náhodně.

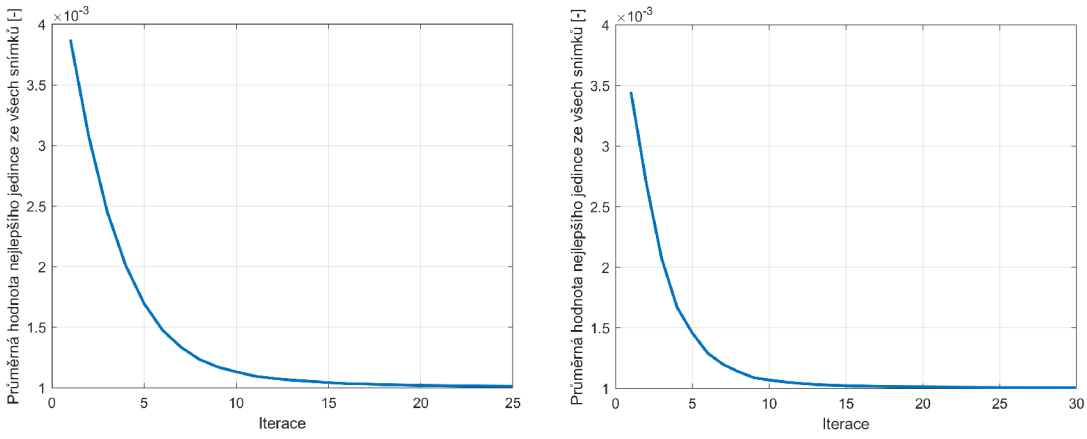
4.7.1 Stanovení parametrů algoritmu BA

Z pohledu nastavení parametrů BA byl tedy uvažován počet včel v jednotlivých skupinách, tzn. počty v n, m a e. Počet včel průzkumnic n byl nastaven na celé číslo a zbylé dvě skupiny včel m a e byly dány zlomkem vůči počtu průzkumnic, kdy byl uvažován stanovený počet ve skupině m a e, který byl zadán procenty z celkového množství. Počet vyčkávajících včel $n - m$ pro elitní průzkumnice se zpravidla volí dvojnásobný oproti neelitním jedincům. V rámci "tanečního rituálu" (viz. kapitola 3.5.2) parametr r udával rádius, ve kterém se jedna z náhodně vybraných hledaných pozic včel (posun v ose x, y nebo rotace) náhodně změnila v maximálně 10 % rozsahu z prohledávacího prostoru algoritmu BA, který byl omezen ve shodných mezích, ve kterých byl generován náhodný pohyb snímků sekvence v kapitole 4.2.2. Parametr r_{damp} poté s rostoucím počtem iterací utlumoval tento možný rádius změny některé z náhodně vybraných pozic včel, jelikož byl vysloven předpoklad, že prohledávání včel s rostoucími iteracemi bude směřovat do blízkosti globálního optima [39]. Na základě studií [25], [26] a [27] byly odvozeny výchozí hodnoty algoritmu BA v tabulce 4.7, kdy příkazem `round` bylo zaokrouhleno procentní množství z určité hodnoty na nejbližší celé číslo.

Tab. 4.7: Výchozí nastavení parametrů BA

Parametr BA	**Výchozí hodnota**
n	20
m	6 (30 % z n)
e	3 (50 % z m)
Vyčkávající včely pro m-e	4 (20 % z n)
Vyčkávající včely pro e	8
r	2
r_{damp}	0,95
Maximální počet iterací	25

Jakmile bylo určeno počáteční nastavení parametrů, došlo v závislosti na průměrné hodnotě kriteriální funkce ze všech snímků sekvence k hodnocení, zda zvýšením počtu iterací a včel ve stejném poměru jako výchozí nastavení, povede k jejímu výraznějšímu poklesu. Počet iterací byl z počátečních 25 navýšen o 20 % na 30 a množství průzkumnic vzrostlo přibližně o 20 % z výchozích 20 na 25. Z těchto 25 průzkumnic poté bylo ve stejném poměru jako v základním nastavení stanoveno množství zbylých včel - tedy 8 nejlepších včel m, 4 elitní včely e, 5 vyčkávajícch včel pro $m - e$ a dvojnásobný počet 10 vyčkávajících včel pro e. Výsledný graf průměrného poklesu kriteriální funkce na obrázku 4.9 byl získán jako průměrná hodnota ze tří běhů každé z 2 realizací. Je zjevné, že větší počet iterací a samovolné navýšení počtu jedinců (průzkumnic) a ostatních včel ve stejném poměru jako základní nastavení, nepřineslo žádné výrazné zlepšení v podobě poklesu kriteriální funkce, pokud nejsou pro další testování brány v potaz jiné poměry zbylých dvou skupin včel, které u BA figurují. Výchozí nastavení dosáhlo průměrného poklesu kriteriální funkce $1{,}012 \cdot 10^{-3}$ za 342 sekund, kdežto nastavení s navýšenou populací a počtem iterací $1{,}009 \cdot 10^{-3}$, což je, vezme-li se navíc v úvahu téměř 3,5 minutový rozdíl ve výpočetním čase, který činil 547 sekund, velice zanedbatelné zlepšení.



Obr. 4.9: Srovnání průměrné hodnoty poklesu nejlepší včely ze všech snímků zprůměrované ze 3 běhů pro výchozí nastavení 20 průzkumnic a 25 iterací (vlevo) a zvýšenou populaci 25 průzkumnic a 30 iterací (vpravo)

Pro zjištění optimálního poměru včel v jednotlivých skupinách v závislosti na pevně stanoveném počtu průzkumnic, byly odzkoušeny další procentní množství včel, vycházející ze článků, na základě kterých bylo určeno výchozí nastavení parametrů BA. V tabulce 4.8 jsou poté vypsány parametry, které se oproti základnímu nastavení měnily společně s průměrnou minimální hodnotou MSE a její směrodatnou odchylkou ze snímků sekvence získané ze tří běhů algoritmu, podle kterých bylo

určeno, které parametry budou podléhat dalšímu zkoumání. Při tomto testování byla ze své základní pozice vychylována pouze zapsaná skupina včel anebo v případě m nejlepších včel i e elitních, které byly určovány zlomkem vůči m. Jelikož je zpravidla počet vyčkávajících včel pro elitní jedince brán jako dvojnásobek neelitních, tento parametr nebyl nijak dál měněn. Taktéž bylo rozhodnuto ponechat r_{damp} v základním nastavení, jelikož byl vysloven předpoklad, že se vzrůstajícím počtem iterací se budou včely pohybovat stále blíže k optimálnímu řešení.

Tab. 4.8: Srovnání změny průměrné střední kvadratické chyby (MSE) a její směrodatné odchylky při vychýlení vždy jednoho parametru z výchozího nastavení ze tří běhů algoritmu BA

Měněný parametr	MSE [10^{-3}]	Směrodatná odchylka
Žádný	1,00894	0,000492
$m = 10$ (50 % z n)	1,00871	0,000492
$m = 4$ (20 % z n)	1,00955	0,000493
$e = 2$ (20 % z m)	1,00918	0,000492
Vyčkávající m-$e = 2$ (10 % z n)	1,01246	0,000493

Ve všech případech měněných parametrů se kromě prvního jednalo o zjednodušení výpočtu v podobě menšího počtu jedinců, kteří každý jiným specifickým způsobem prohledávají vymezený prostor parametrů. Je tak zapotřebí si uvědomit, že vlivem toho jsou na algoritmu kladeny nižší výpočetní nároky a s tím spojená časová naročnost, které vyusťují k horším výsledkům. Jako nové nejlepší nastavení parametrů byla nakonec zvolena kombinace 10 nejlepších včel m (50 % z n) a 5 elitních (50 % z m), jejíž srovnání s původním nastavením nabízí tabulka 4.9. Rozptyl hodnot nového nastavení byl ve srovnání s výchozím téměř identický, akorát průměrná hodnota dosažených minim MSE na jednotlivých snímcích sekvence se nepatrně lišila.

Tab. 4.9: Srovnání nastavených parametrů, střední hodnoty a směrodatné odchylky MSE a výpočetní náročnosti pro výchozí a nově zvolené nastavení BA

Nastavené parametry	Výchozí nastavení	Nové nastavení
n	20	20
m	6 (30 % z n)	10 (50 % z n)
e	3 (50 % z m)	5 (50 % z m)
Vyčkávající včely pro m-e	4 (20 % z n)	4 (20 % z n)
Vyčkávající včely pro e	8	8
r	2	2
r_{damp}	0,95	0,95
MSE [10^{-3}]	1,00894	1,00871
Směrodatná odchylka	0,000492	0,000492
Průměrný výpočetní čas [s]	342	424

4.8 Registrace s optimalizací umělé včelí kolonie (ABC)

Algoritmus ABC lze rozdělit do celkem tří fází, v nichž v každé z nich figuruje jiný typ včely. V první fázi první typ včel, označované jako dělnice, vygenerují nové řešení v prohledávaném prostoru, spočítají funkční hodnoty kriteriální funkce a dochází k přepočtu funkčních hodnot na hodnoty pravděpodobností, na základě kterých je vybrána konkrétní včela. Optimalizace dělnic byla provedena lokálním prohledáváním, kdy původní pozice včely byla změněna o váhovaný náhodný krok ϕ ve směru nějaké jiné náhodně vybrané včely za předpokladu, že funkční hodnota kriteriální funkce v nové pozici byla lepší oproti původní. Dále byl pro každou včelu zaveden čitač, který zaznamenával, jestli se změnila její pozice a v případě že ano, byl tento čitač pro danou včelu vynulován. V druhé fázi byly ruletovým pravidlem generovány dělnice, ke kterým se vyčkávající včely přitáhy. Tento druhý typ včel vykonával náhodné prohledávání v okolí dělnic stejným způsobem jako jej vykonávaly dělnice v první fázi. Pokud vyčkávající včela nalezla lepší pozici, dělnice, která byla vybrána ruletovým pravidlem, zaktualizovala svoji pozici a pokud ke zlepšení nedošlo, čitač pro danou dělnici se inkrementoval o jedničku. Ve třetí fázi, pokud čitač přesáhnul tzv. mezní hodnotu pro zánik dělnice, daná včela zanikla a stala se z ní průzkumnice, která náhodně prohledávala prostředí.

4.8.1 Stanovení parametrů algoritmu ABC

V případě BA byl uvažován parametr velikosti kolonie, který byl roven počtu dělnic, vyčkávajících včel a průzkumnic. Dále připadala v úvahu mezní hodnota pro zánik

dělnice, která byla ve výchozím nastavení stanovena pomocí vzorce [40]

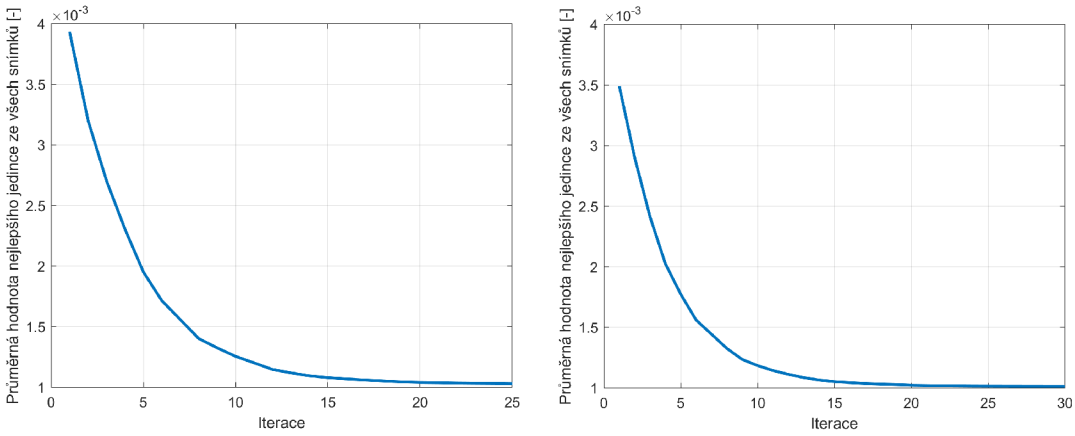
$$Mezní\ hodnota = SN \cdot D \cdot 0,6\ , \tag{4.1}$$

kde SN značí velikost roje a D dimenzi, tedy 3 hledané parametry (posun v ose x, y nebo rotace). Konstanta a sloužila jako horní hranice pro konstantu ϕ a udávala rozsah lokálního prohledávání včel a nakonec počet iterací sloužil jako ukončovací podmínka. Prostor prohledávání parametrů byl znovu omezen ve stejných mezích, ve kterých byl generován synteticky přidaný pohyb do snímků sekvence v kapitole 4.2.2. Dle studií [8], [10] a [29] zabývajícími se aplikací BA algoritmu, bylo určeno počáteční nastavení parametrů v tabulce 4.10.

Tab. 4.10: Výchozí nastavení parametrů ABC

Parametr ABC	**Výchozí hodnota**
Velikost kolonie	20
Maximální počet iterací	25
Mezní hodnota pro zánik dělnice	36
a	1

Nejprve byl tak jako u předešlých metod optimalizace sledován pokles kriteriální funkce v závislosti na velikosti populace (v případě BA to byla velikost kolonie) a počtu iterací. Iniciální velikost kolonie byla navýšena přibližně o 20 % na 25 a počet iterací na 30. Grafy na obrázku 4.10 znázorňují tento průměrný pokles ze třech realizací každého ze dvou nastavení ze všech snímků sekvence. Ve výchozím nastavení dosáhla kriteriální funkce ve 25. iteraci hodnoty $1{,}03 \cdot 10^{-3}$ v průměru během 246 sekund a při zvýšení počtu jedinců ve 30. iteraci hodnoty 1,014 za 377 sekund. Navýšení velikosti kolonie společně s počtem iterací má tak v závislosti na výchozím nastavení těchto parametrů velmi malý vliv na pokles kriteria s ohledem na výpočetní čas jednotlivých nastavení.



Obr. 4.10: Srovnání průměrné hodnoty poklesu nejlepší včely ze všech snímků zprůměrované ze 3 běhů pro výchozí nastavení velikosti kolonie 20 a 25 iterací (vlevo) a zvýšenou velikost kolonie 25 a 30 iterací (vpravo)

Následně byla jako u předchozích algoritmů testována nejlepší průměrná hodnota MSE a její směrodatná odchylka pro zbylé parametry, kdy napřed byl každý z parametrů vychylován zvlášť. Na základě studií byly odzkoušeny 2 další mezní hodnoty zániku dělnice dle rovnic 4.2 a 4.3

$$Mezní\ hodnota = SN \cdot D\ , \tag{4.2}$$

$$Mezní\ hodnota = SN \cdot D \cdot 0,5\ , \tag{4.3}$$

a dvě odlišné hodnoty konstanty a, které jsou společně s výsledky uvedeny v tabulce 4.11.

Tab. 4.11: Srovnání změny průměrné střední kvadratické chyby (MSE) a její směrodatné odchylky při vychýlení vždy jednoho parametru z výchozího nastavení zprůměrované ze tří běhů algoritmu ABC

Měněný parametr	MSE [10^{-3}]	Směrodatná odchylka
Žádný	1,01347	0,000493
Mezní hodnota podle 4.2	1,00998	0,000493
Mezní hodnota podle 4.3	1,01823	0,000492
$a = 2$	1,06221	0,000499
$a = 0{,}5$	1,02402	0,000490

Oproti základnímu nastavení vykazovala pouze mezní hodnota zániku dělnice dle 4.2 zlepšení v podobě nižší dosažené průměrné MSE. Konstanta a, váhující rozsah

lokální prohledávání, nezaznamenala v žádné z vychýlených pozic zlepšení a byla ponechána v základním nastavení. Porovnání testováním nejlepšího dosaženého a výchozího nastavení je možné vidět v tabulce 4.12, kde lze pozorovat, že průměrný vypočetní čas nové nejlepší realizace je dokonce o jednotky sekund kratší. Na základě výsledků směrodatných odchylek MSE lze usoudit, že rozptyly průměrných MSE obou realizací se v zásadě neliší.

Tab. 4.12: Srovnání nastavených parametrů, střední hodnoty a směrodatné odchylky MSE a výpočetní náročnosti pro výchozí a nově zvolené nastavení ABC

Nastavené parametry	Výchozí nastavení	Nové nastavení
Velikost kolonie	20	20
Maximální počet iterací	25	25
Mezní hodnota pro zánik dělnice	36 (podle 4.1)	60 (podle 4.2)
a	1	1
MSE [10^{-3}]	1,01347	1,00998
Směrodatná odchylka	0,000493	0,000493
Průměrný výpočetní čas [s]	246	240

4.9 Registrace s optimalizací světluškového algoritmu (FA)

Algoritmus FA byl inspirován chováním světlušek, které vábí ostatní světlušky pomocí tzv. bioluminiscence, kdy dochází k přeměně energie a emisi ve viditelné části spektra. Světlušky představují jednotlivá řešení problému a je hledána změna pozice světlušek. Motivace pro světluškový algoritmus je taková, že světluška, která je z pohledu řešení jeho lepší aproximací, přitáhne světlušky, které jsou horší. Světlo, které je generováno, je v základní podobě konstatntní a míra přitáhnutí závisí na nastavení parametrů, přičemž v tomto případě to ovlivní vzdálenost světlušek. Bude platit, že pokud jsou světlušky blízko k sobě, budou se přitahovat více než světlušky, které jsou od sebe dále. Algoritmus FA je v užité realizaci ze skupiny All-to-All Adaptive, čili iteračně jsou procházeny všechny světlušky každá s každou a výsledný pohyb, který světluška vykoná, bude superpozicí pohybů vůči všem jasnějším světluškám. Vlivem pracování na seřazené populaci jedinců jsou zanechány pouze interakce směrem k lepším světluškám. V každé iteraci je poté vyhodnocen nejlepší jedinec. Navíc je zde ještě definována forma náhodného pohybu, kdy s rostoucím počtem iterací bude klesat jeho vliv a bude převažovat směr od jedné světlušky k druhé.

4.9.1 Stanovení parametrů algoritmu FA

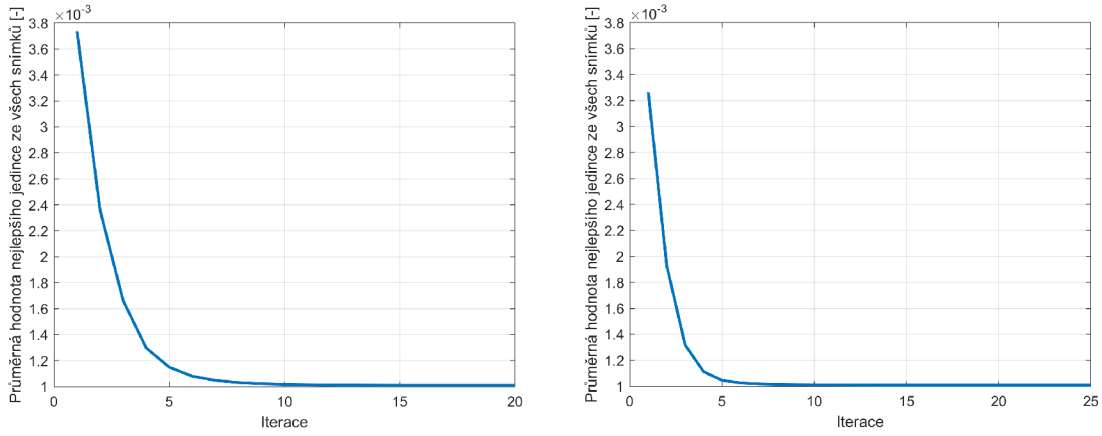
V rámci FA bylo pracováno s populací jedinců (světlušek), přičemž počáteční populace byla vygenerována formou náhodného rozložení, kdy byl stanoven její jas v podobě hodnoty kriteriální funkce. Poloha světlušky, jež byla přitahována z pohledu řešení lepší světluškou, byla dána složkami pohybu, podrobněji popsané v rovnici 3.20 v kapitole 3.5.4. Parametr α představoval váhu složky náhodného pohybu, jejíž vliv byl s rostoucím počtem iterací tlumen parametrem α_{damp}, čímž začla převahovat druhá složka pohybu [41]. Tento druhý člen pohybu byl dán směrovým vektorem, který směřoval do pozice jasnější, z pohledu řešení lepší, světlušky. Velikost směrového vektoru byla váhována členy β_0 a γ, kdy konkrétně γ představovala váhu jednotlivých vzdáleností světlušek, která byla v této realizaci uvažována jako euklidovská. Prohledávaný prostor parametrů byl shodně omezen jako u předchozích metod optimalizace a jako v případě přidávání náhodných posunů a rotací do snímků sekvence v kapitole 4.2.2, které podléhaly registraci na první referenční snímek. Výchozí parametry v tabulce 4.13 byly stanoveny na základě článků [33] a [34].

Tab. 4.13: Výchozí nastavení parametrů FA

Parametr FA	**Výchozí hodnota**
Počet světlušek	12
Maximální počet iterací	20
β_0	2
γ	1
α	0,2
α_{damp}	0,98

Problematický aspekt FA je jeho výpočetní náročnost pro nalezení optimálního řešení, které počítá v každé iteraci ve dvou cyklech v sobě, kdy vyhodnocuje světlušky vůči sobě, přesněji řečeno polohu každé světlušky vůči všem ostatním světluškám. To znamená, že okamžiku kdy se zdvojnásobí počet světlušek, se doba trvání k nalezení řešení nezdvojnásobí, ale teoreticky by měla být čtyřikrát větší. Na základě tohoto mínění byla populace světlušek a počet iterací zvýšeno pouze nepatrně, přibližně o 20 %, avšak rozdíl výpočetního času oproti výchozímu nastavení byl zásadní. V průměrné čase 325 sekund ze tří běhu algoritmu dosáhlo výchozí nastavení průměrného poklesu kriteriální funkce hodnoty $1{,}0091 \cdot 10^{-3}$ ze všech snímků sekvence, kdežto s navýšenými parametry světlušek a iteracemi to byl pokles na hodnotu $1{,}0089 \cdot 10^{-3}$ za téměr dvojnásobný průměrný čas 640 sekund. Tvrzení se tak potvrdilo, čímž další navyšování počtu jedinců a iterací nemělo význam a bylo k dalšímu

zkoumání ponecháno výchozí nastavení parametrů.



Obr. 4.11: Srovnání průměrné hodnoty poklesu nejlepší světlušky ze všech snímků zprůměrované ze 3 běhů pro výchozí nastavení 12 světlušek a 20 iterací (vlevo) a zvýšenou populaci 15 svtlušek a 25 iterací (vpravo)

Sledování průměrné směrodatné odchylky a jednotlivých dosažených minimálních hodnot MSE na snímcích sekvence bylo znovu provedeno vychylováním ostatních parametrů FA, u kterých bylo naznáno, že by významněji mohly ovlivnit výsledek pro optimální registraci na základě jiných navrhovaných hodnot parametrů ve zmíněných článcích. Například je-li váha náhodného pohybu zvýšena hodnotou parametru $\alpha = 0{,}6$; je větší pravděpodobnost, že z počátku iterací světlušky opustí svá řešení a budou hledat jiná, přičemž opačná situace nastane při snížení tohoto parametru. Změna parametru γ, který váhuje to, na jak velkou vzdálenost se světlušky ovlivní, vlivem jeho snížení na hodnotu $\gamma = 0.5$ by mělo dospět k tomu, že světlušky budou ovlivňovány i na velké vzdálenosti a s určitou pravděpodobností vytvoří jeden jednolitý velký shluk, který už potom bude provádět pouze lokální prohledávání. Parametr β_0 poté váhuje druhou složku pohybu, tedy směrový vektor směrem k lepším světluškám. V tabulce 4.14 jsou jednotlivé parametry s hodnotou, s kterou byly vychýleny společně s výsledky uvedeny.

Tab. 4.14: Srovnání změny průměrné střední kvadratické chyby (MSE) a její směrodatné odchylky při vychýlení vždy jednoho parametru z výchozího nastavení zprůměrované ze tří běhů algoritmu FA

Měněný parametr	MSE [10^{-3}]	Směrodatná odchylka
Žádný	1,00885	0,000493
$\beta_0 = 2{,}1$	1,00887	0,000493
$\beta_0 = 1{,}8$	1,00888	0,000493
$\gamma = 2$	1,00889	0,000493
$\gamma = 0{,}5$	1,00888	0,000493
$\alpha = 0{,}6$	1,00977	0,000493
$\alpha = 0{,}1$	1,00877	0,000493

Z výsledků je patrné, že při snížení parametru α na hodnotu 0,1; čili vliv náhodného pohybu světlušek byl co možno nejvíc potlačen, bylo dosáhnuto nejlepších výsledků z pohledu průměrné MSE. Celkové srovnání výchozího a nově nalezeného nejlepšího řešení je v tabulce 4.15. Průměrný výpočetní čas byl u nového nastavení kratší, avšak rozdíl oproti výchozímu je téměř zanedbatelný.

Tab. 4.15: Srovnání nastavených parametrů, střední hodnoty a směrodatné odchylky MSE a výpočetní náročnosti pro výchozí a nově zvolené nastavení FA

Nastavené parametry	Výchozí nastavení	Nové nastavení
Počet světlušek	12	12
Maximální počet iterací	20	20
β_0	2	2
γ	1	1
α	0,2	0,1
α_{damp}	0,98	0,98
MSE [10^{-3}]	1,00885	1,00877
Směrodatná odchylka	0,000493	0,000493
Průměrný výpočetní čas [s]	325	319

4.10 Srovnání optimalizačních metod na umělé sekvenci

Jak již bylo řečeno v úvodu této kapitoly, sekvence se sestávala z celkem šedesáti ultrazvukových snímků, do kterých byl kromě prvního uměle vnesen náhodný pohyb v podobě posunů v osách x a y a natočení (viz. kapitola 4.2.2). Cílem této kapitoly

bylo především zpětně naleznout tyto parametry geometrické transformace pomocí minimalizace vybrané kriteriální funkce MSE (viz. kapitola 4.4) popisující kvalitu zarovnání, která povede k optimální registraci za co nejkratší možný časový interval. K tomu bylo zapotřebí si zpočátku zadefinovat adekvátní registrační přístup (4.3). Ten byl zvolen s ohledem na jeho využívání v praktických aplikacích. Registrace tak probíhala mezi dvojicí snímků, konkrétně mezi prvním referenčním a vždy náhodně vychýleným snímkem ze sekvence, přičemž optimalizační proces přilícování jednoho snímku ke druhému probíhal pomocí 5 implementovaných optimalizačních metod. Na kvalitu hledané transformační funkce mají zejména vliv optimálně zvolená nastavení parametrů konkrétních pěti metod, kterými se zabývaly kapitoly 4.5 v případě spojitého genetického algoritmu (GA), 4.6 pro Particle Swarm Optimalizaci (PSO), 4.7 pro včelí algoritmus (BA), 4.8 pro algoritmus umělé včelí kolonie (ABC) a nakonec 4.9 pro světluškový algoritmus (FA). Parametry, které se prokázaly, že mají nezanedbatelný vliv na kvalitu řešení a hlavně také na výpočetní čas, byly velikost populace a počet iterací jejíchž optimální nastavení byly zkoumány ze všeho nejdřív vždy na počátku hledání vhodného nastavení daného algoritmu. Patrně nejlépe lze vystihnout kvalitu registrace pomocí střední kvadratické chyby (MSE), která při uvažování užitého registračního přístupu, byla taktéž počítána mezi referenčním prvním snímkem a něho registrovaným který, jak již bylo zmíněno, byl náhodně zašuměný.

Tab. 4.16: Srovnání průměrných MSE dosažených středních a minimálních hodnot na jednotlivých snímcích a průměrných časů pro jednotlivá nejlepší nastavení optimalizačních metod na umělé sekvenci

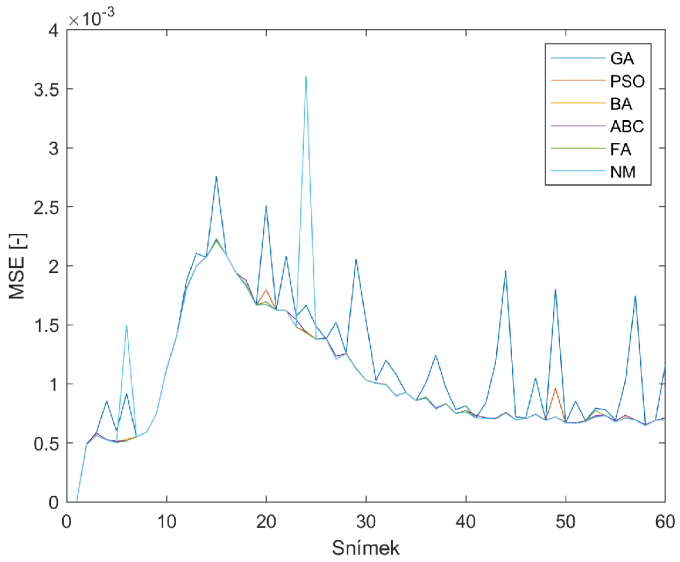
Algoritmus	**MSE** [10^{-3}]		**Čas** [s]
	Minimální	**Průměrná**	
Spojitý GA	1,04026	1,20377	386
PSO	1,00880	1,01528	150
BA	1,00871	1,01066	424
ABC	1,00998	1,01577	240
FA	1,00877	1,01023	319
Nelder-Mead	1,00792	1,06098	278

V souhrnné tabulce 4.16 jsou uvedeny průměrné MSE nejnižších dosažených hodnot MSE na snímích jednotlivých optimalizačních metod, které byly vyvozeny jako vhodné indikátory pro to, jakých nejlepších hodnot je daný algoritmus s daným nastavením schopen dosáhnout a z toho důvodu právě na nich byly určovány nejlepší parametry jednotlivých metod. Společně s nimi jsou také zaneseny průměrné MSE

průměrných dosažených hodnot MSE na snímcích, které narozdíl od minim dávají lepší přehled o variabilitě výsledků.

Pro srovnání použitých rojových optimalizačních metod a jedné evoluční byla v této práci zvolena Nelder-Meadova metoda optimalizace, kterou je možné v MATLABu implementovat pomocí funkce `fminsearch`. Jedná se o deterministický algoritmus, takže pokud je zavolán ze stejné pozice opakovaně, výsledek bude stále stejný. Úspěšnost této metody se tedy značně odvíjí od toho, odkud je zvolen počáteční odhad. Nabízí se generování pozic náhodně, které bylo provedeno tak, že jelikož byly hledány 3 parametry (dva posuny a natočení), byl každý z parametrů omezen znaménkem a bylo odzkoušeno 6 kombinací. Která z těchto kombinací pro danou dvojici snímků dávala nejlepší hodnotu MSE, tato kombinace byla vybrána.

Z minim průměrných hodnot MSE lze poznamenat, že dosažené výsledky se pro jednotlivé algoritmy lišily v řádu statisícin, výjma spojitého GA, který se od ostatních lišil již v řádu desetitisín. O něco zatelnější, avšak také velice nepatrné rozdíly byly zaznamenány u středních hodnot průměrných MSE, které dávají lepší přehled o variabilitě výsledků. Obecně lze říci, že toto jsou velmi malé rozdíly které znamenají, že žádný z algoritmů globálně při optimalizační úloze neselhal. Co ovšem není zanedbatelné je doba, která algoritmům trvala dosáhnout optimálních parametrů, která se v případě roje částic (PSO) a včelího algoritmu (BA) lišila téměř trojnásobně. Co se týče Nelder-Meadova algoritmu, který byl implementován za účelem srovnání evolučních technik s jinými optimalizačními algoritmy, dosáhl v případě průměrných minim MSE nejlepšího výsledku, avšak pokud jde o průměrné střední hodnoty, tak zde selhal nejvíce.



Obr. 4.12: Průběh průměrných MSE na umělé sekvenci

Na obrázku 4.12 je graficky znázorněn průběh průměrných hodnot MSE na všech snímcích sekvence pro všechny optimalizační algoritmy. Lze si povšimnout, že průměrné hodnoty MSE velmi přesně kopirují koncentraci kontrastní látky v jednotlivých snímcích, které bylo možné vidět na obrázku 4.3, díky čemuž je možné prohlásit, že výsledek registrace pro danou dvojici snímků byl značně ovlivněný nejen přítomným šumem speckle, uměle zanesenými pohyby, ale také množstvím zachycené kontrastní látky, která napříč sekvencí není konstantní.

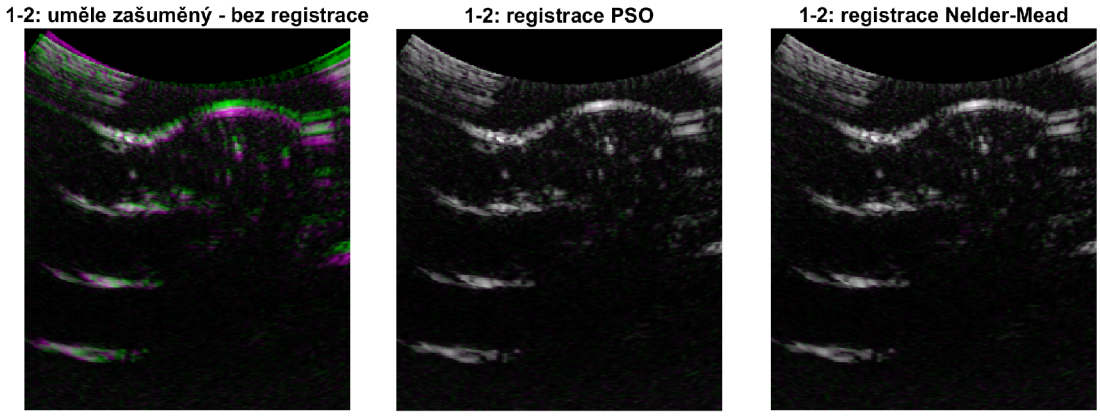
Tab. 4.17: Srovnání průměrných odchylek mezi skutečnými a algoritmy nalezenými parametry transformace a průměrné MSE středních hodnot u vybraných dvojic snímků na umělé sekvenci

Algoritmus	Posun v x [px]	Posun v y [px]	Rotace [°]	MSE [10^{-3}]
Dvojice snímků 1-2				
GA	0,2180	0,0008	0,0068	0,48437
PSO	0,2128	0,0031	0,0042	0,48451
BA	0,2516	0,0083	0,0022	0,48549
ABC	0,1762	0,0035	0,0112	0,49302
FA	0,2185	0,0039	0,0061	0,48450
Nelder-Mead	0.2232	0,0001	0.0043	0,48433
Dvojice snímků 1-13				
GA	0,6173	1,3511	0,0193	2,10556
PSO	0,3117	1,5904	0,0001	1,99828
BA	0,2824	1,6207	0,0114	2,00059
ABC	0,3097	1,5905	0,0001	1,99836
FA	0,3131	1,5916	0,0001	1,99829
Nelder-Mead	0,3509	1,5933	0,0250	1,99782
Dvojice snímků 1-24				
GA	1,2397	0,6347	0,2303	1,66346
PSO	0,6830	0,1417	0,1002	1,44398
BA	0,7193	0,2340	0,0616	1,43509
ABC	0,7256	0,2097	0,0777	1,43352
FA	0,7218	0,2108	0,0703	1,43304
Nelder-Mead	3,6467	2,8743	1,9325	3,60811

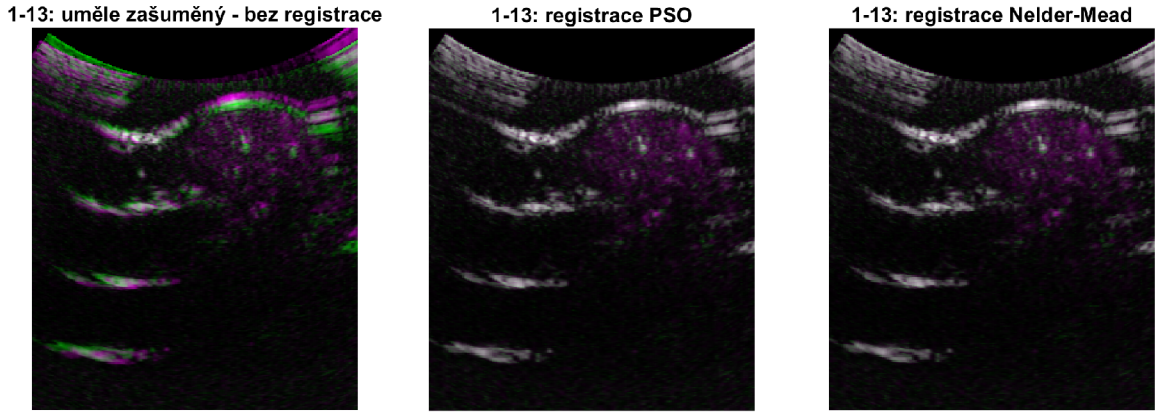
Jelikož byly snímky podléhající registraci na referenční snímek zkresleny známými hodnotami posunů a rotací, které byly ukládány ve formě vektorů (viz. kapitola 4.2.2), bylo možné správnost registrace ověřit pomocí odchylky mezi skutečnými (tedy náhodně vygenerovanými) a optimalizačními metodami nalezenými parame-

try transformace. V tabulce 4.17 jsou uvedeny průměry získaných odchylek ze třech realizací nejlepších nastavení daných algoritmů pro vybrané dvojice snímků. Ty byly vybrány na základě v nich rozdílně přítomného zastoupení množství kontrastu. U dvojice po sobě jdoucích snímků 1-2 byl kontrast podobný, naopak u dvojice 1-13, kde kolem 13. snímku byla předpokládáná nejvyšší hodnota kontrastu, byl rozdíl největší a nakonec u dvojice 1-24 se kontrast kolem 24. snímku začal vracet k počátečnímu množství, kdy se kontrastní látka začala vyplavovat. Dle předpokladu vycházejí pro dvojici snímků na začátku sekvence o poznání lépe odchylky translace a natočení společně s MSE, jelikož zde rozdíl v kontrastu nebyl nijak odlišný. Dvojice 1-24 byla taktéž vybrána na základě výsledku MSE Nelder-Meadova algoritmu, který na tomto snímku v 4.3 divergoval, kdy bylo zjištěno, že náhodně vygenerovaný rotační úhel byl ve všech třech případech spuštění maximálně možný (tedy +10 stupňů) a pravděpodobně tak Nelder-Mead uvíznul v nastavených hranicích algoritmu a nepodařilo se mu již uniknout z lokálního optima.

Na základě výsledků průměrné MSE minimálních dosažených hodnot na snímcích sekvence z tabulky 4.16 se globálně ukázaly jako nejúspěšnější metody PSO a Nelder-Meadova heuristika, byť pouze o nepatrný rozdíl. Na závěr této kapitoly je u dvou ze zvolených dvojic snímků přiložena ukázka grafického výstupu registrace těchto dvou algoritmů. V případě dvojice 1-2 na obrázku 4.13 vlevo byl druhý snímek postihnut posunem v ose x o -3 pixely, v ose y o 4 pixely a úhlem natočení $-3°$ a položen na první neposunutý. Uprostřed se nachází výsledek registrace druhého snímku na první referenční metodou PSO a vpravo Nelder-Meadovou. Výsledek potvrzuje tvrzení, že na začátku sekvence nebyl přítomen žádný dodatečný kontrast ve snímcích, neboť zaregistrované snímky vložené přes sebe se jeví jako stejné. Na obrázku 4.14 vlevo byl u dvojice 1-13 třináctý snímek o 5 pixelů posunut v ose x, o -3 pixely v ose y a orotován o $4°$ a vložen přes první. Ve výsledné registraci uprostřed metodou PSO a Nelder-Meadem vpravo je možné u zaregistrovaných snímků zahlédnout fialové zbarvení v oblasti zájmu, které je dáno změnou kontrastu, která ztěžovala proces registrace, avšak zvolené optimalizační metody se zvolenými parametry si s tímto problémem dokázaly poradit.



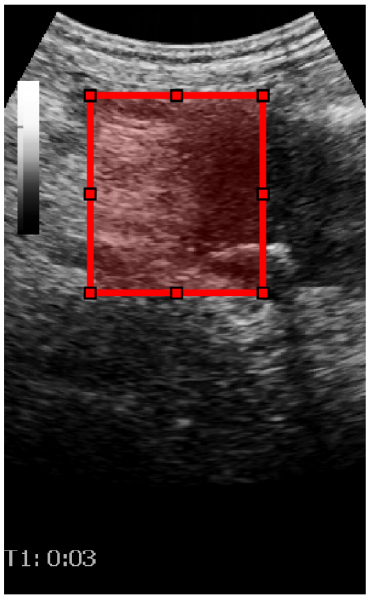
Obr. 4.13: Uměle přidaný pohyb do 2. snímku zobrazeného na první referenční (vlevo), registrace 2. snímku na první algoritmem PSO (uprostřed), registrace 2. snímku na první Nelder-Meadovým algoritmem (vpravo)



Obr. 4.14: Uměle přidaný pohyb do 13. snímku zobrazeného na první referenční (vlevo), registrace 13. snímku na první algoritmem PSO (uprostřed), registrace 13. snímku na první Nelder-Meadovým algoritmem (vpravo)

5 Registrace klinické sekvence nádoru pankreatu

V této kapitole byla realizována registrace klinické sekvence po podání kontrastní látky. Měřeným objektem byl v tomto případě pacient s nádorem pankreatu neboli slinivky břišní. Samotný nádor ve snímcích představuje hypoechogenní oblast (tmavě šedá barva), ve které v důsledku jeho prokrvení docházelo k nejvýraznějším změnám v kontrastu při aplikaci kontrastní látky. Pankreas je poté uložen vlevo od něj. Oblast zájmu, ve které registrace probíhala, byla zvolena tak, aby částečně zasahovala jak do nádoru, tak do oblasti výskytu pankreatu. Podobu jednoho snímku ze sekvence s vyznačeným oknem, kde probíhala registrace, je možné vidět na obrázku 5.1.



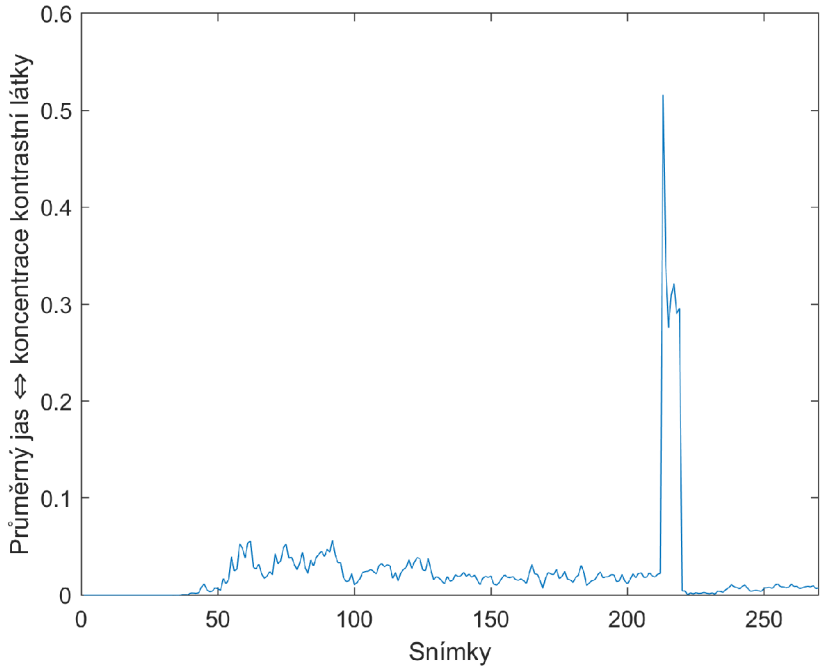
Obr. 5.1: Snímek z klinické sekvence nádoru pankreatu s vyznačenou oblastí zájmu

Data z tohoto snímku byly získány z první harmonické komponenty a nesou v sobě informaci o anatomické struktuře. V druhé harmonické komponentě jsou potom kontrastní data, kde je větší výskyt informace o kontrastní látce. Registrace probíhala na anatomnických datech z první harmonické komponenty, ve kterých nejsou změny kontrastu tak významné.

5.1 Informace o kontrastu a převzorkování sekvence

Jedná se o dynamickou sekvenci, čili v průběhu sekvence se měnil kontrast snímků v závislosti na množství kontrastní látky. V sekvenci je přítomno celkem 270 snímku

s obrazovou frekvencí 3 FPS (90 sekund záznamu). Průměrný jas v obrazech napříč celou sekvencí snímků ve zvolené oblasti zájmu je možné vidět v grafu 5.2, který byl získán z kontrastních dat druhé harmonické komponenty, kde je možné lépe sledovat změnu kontrastu.



Obr. 5.2: Průběh kontrastu v první klinické sekvenci

Z grafu bylo zjištěno, že od 40. snímku dochází k nárůstu kontrastu a na 213. snímku k projevu použité metody Bolust&Burst, která aplikuje vysokoenergetický pulz, jenž rozbije mikrobubliny v dané oblasti. Následně je v této oblasti touto technikou sledováno zaplnění kontrastní látkou. Jelikož byly v rámci registrace hledány optimální parametry geometrické transformace - posun a rotace, tak v okamžiku vyslání tohoto umělého energetického pulzu jsou snímky přesvícené, což značně ztěžuje proces registrace.

Pro účely práce byla sekvence převzorkována, tedy vybrán každý třetí obraz z intervalu 45. až 210. snímku, což ve výsledku dalo 56 snímků, které podléhaly registraci.

5.2 Srovnání optimalizačních metod na první klinické sekvenci

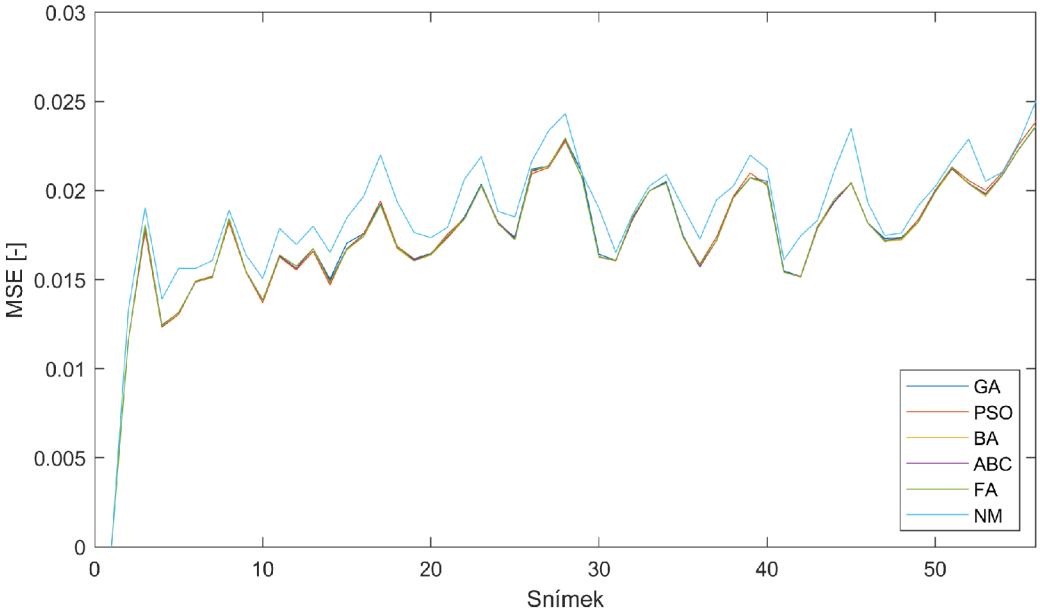
Na rozdíl od sekvence simulované scény, kde bylo možné díky známé referenci srovnávat úspěšnost registrace na základě skutečných transformačních hodnot posunů a

natočení, na sekvencích s reálnými daty bylo zapotřebí si pro vyhodnocení vystačit s dosaženými průměrnými hodnotami střední kvadratické chyby MSE jednotlivých optimalizačních metod a posléze srovnat nalezené parametry transformace těmito metodami mezi sebou. Registrace probíhala tak jako v kapitole 4 mezi prvním referenčním a na něho transformovaným obrazem, kterými bylo zbylých 55 snímků sekvence optimalizačními algoritmy spojitého GA, PSO, BA, ABC, FA a neevoluční Nelder-Meadovy metody, jež již byly využity pro hledání parametrů transformace simulované scény, kde bylo nalezeno jejich optimální nastavení s ohledem na nízkou výpočetní náročnost, které bylo využito i zde. Všechny algoritmy byly odzkoušeny na celkem třech bězích. Z toho důvodu dávalo větší význam pro jejich srovnání směrodatným kritériem střední kvadratické chyby použít minimální dosažené hodnoty na jednotlivých snímcích, které jsou lepšími indikátory toho, jakých nejlepších hodnot je daná metoda schopna dosáhnout. V tabulce 5.1 jsou tyto průměrné minimální hodnoty MSE vypsány společně s průměrným časem trvání pro nalezení těchto hodnot.

Tab. 5.1: Srovnání průměrných MSE dosažených minimálních hodnot na jednotlivých snímcích a průměrných časů pro jednotlivé optimalizační metody na první klinické sekvenci

Algoritmus	**MSE** [10^{-3}]	**Celkový čas** [s]
Spojitý GA	17,60061	202
PSO	17,49310	83
BA	17,58736	489
ABC	17,59701	136
FA	17,59011	175
Nelder-Mead	18,10206	123

Z těchto výsledků je patrné, že evoluční techniky zareagovaly na sadu snímků na jednotky setin stejně. Co se však lišilo, byl čas, který jednotlivým metodám zabralo nalezení tohoto optimálního řešení na celé sadě snímků. V případě včelího algoritmu BA to bylo přes 8 minut, zatímco roji částic PSO zabralo necelé 1,5 minuty nalezení ještě o tisícinu lepší řešení než včelám. Variabilitu výsledků lze lépe než v případě minimálních hodnot podobnostního kritéria MSE vystihnout průměrnými hodnotami MSE spočítané algoritmy na jednotlivých snímcích, které jsou vyobrazené na obrázku 5.3.



Obr. 5.3: Průběh průměrných MSE na první klinické sekvenci

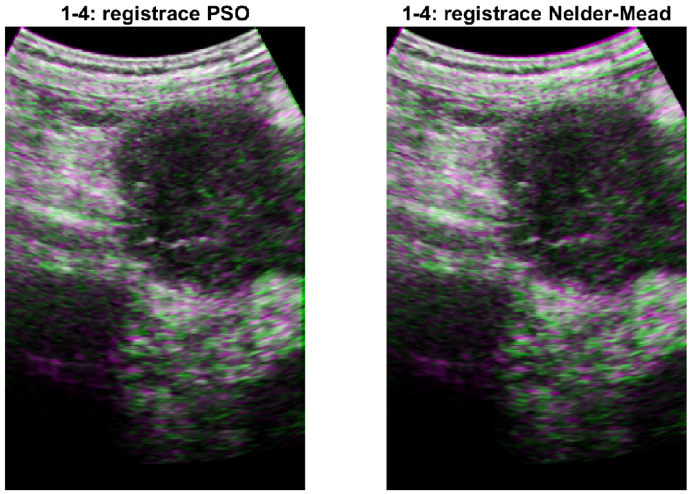
Průměrná hodnota MSE se odvíjí od množství kontrastu a koncentrace kontrastní látky, která v celé sekvenci není konstatní. Globálně však lze říct, že tak jako v případě minimálních MSE, i zde si evoluční a rojové algoritmy s optimalizací poradily lépe než Nelder-Meadův algoritmus, který v závislosti na větší změně konstrastu u registrace dvojic snímků 1-17, 1-23, 1-28, 1-34, 1-39 a 1-45 zde pravděpodobně divergoval do lokálního optima.

Nalezené parametry transformace se globálně mezi evolučními a rojovými algoritmy lišily v řádu desitin pixelů a úhlů pro posuny a natočení, čehož bylo možné usoudit již na základě zkoumání podobnostních kritérií MSE. V rámci dalšího pozorování byla vytvořena tabulka 5.2 pro dvě vybrané dvojice snímků. Na základě 5.3 bylo možné mezi 1-15 pozorovat pravděpodobně největší zaznamenanou změnu průměrné MSE v rámci metaheuristických algoritmů, a to u spojitého GA, který se lišil přibližně v řádu 2 jednotek pixelů v ose x od rojových algoritmů, načež jeho průměrná hodnota MSE je od nich rozdílná v řádech tisícin. Nelder-Meadův algoritmus pak selhával nejvíce v oblastech s vysokým kontrastem, kdy globálně nejvyšší zaznamenáné průměrné MSE napříč všemi algoritmy byly při registraci mezi dvojicí 1-28, výjma snímků ke konci sekvence, kdy Nelder-Meadova metoda selhala při nalezení posunu v ose y, který se od ostatních lišil v řádu 2 jednotek pixelů, což se projevilo na hodnotě průměrné MSE, která je zhruba o 2 tisíciny větší než u ostatních metod.

Tab. 5.2: Srovnání nalezených posunů a natočení jednotlivými algoritmy u vybraných dvojic snímků 1-15 a 1-28

Algoritmus	Posun v x [px]	Posun v y [px]	Rotace [°]	MSE [10^{-3}]
Dvojice snímků 1-15				
GA	7,8306	-0,1707	0,5802	17,04335
PSO	9,5321	-0.4745	1,2700	16,73542
BA	10,4838	-0,4629	1,3284	16,66987
ABC	9.5034	-0.4397	1,3084	16,73330
FA	9,5518	-0,4817	1,2707	16,70491
Nelder-Mead	1,2758	0,4757	-0,9862	18,47283
Dvojice snímků 1-28				
GA	-6.5199	0,4830	-1.8370	22,91747
PSO	-7,4954	0,1806	-2,0482	22,76683
BA	-8,4936	0,5035	-2,2880	22,85621
ABC	-7.8806	0,5183	-2.1233	22,88084
FA	-7,1656	0,8345	-1,1375	22,95059
Nelder-Mead	-5,1692	4,1165	0,4663	24,32064

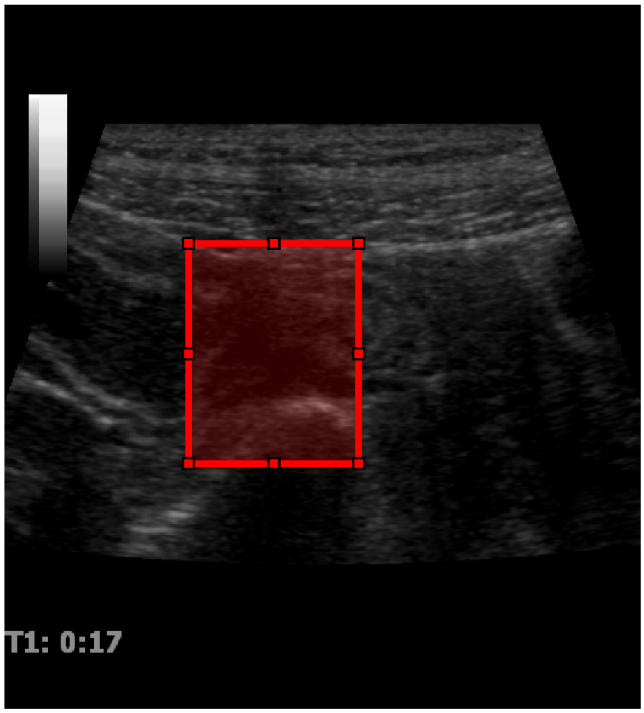
Mimo náběh kontrastní látky na počátku sekvence dosáhly všechny evoluční algoritmy nejlepšího výsledku dle 5.3 na dvojici snímků 1-4. Globálně nejúspěšnější se ukázal být dle tabulky 5.3 roj částic PSO, který byl srovnán s Nelder-Meadovým algoritmem, který selhával nejvíce. Soudě dle obrázku 5.4 výstupu registrace, je rozdíl téměř nepostřehnutelný.



Obr. 5.4: Grafický výstup registrace dvojice snímků 1-4 pro PSO (vlevo) a Nelder-Meadův algoritmus (vpravo)

6 Registrace klinické sekvence tenkého střeva

Poslední sekvencí, které je v praktické části práce věnována pozornost, byla klinická sekvence tenkého střeva. Konkrétně se jednalo o abdominální sekvenci pacienta s Crohnovou nemocí, u kterého byla sledována tkáň tenkého střeva v místě onemocnění. U pacienta proběhlo vyšetření klinickým ultrazvukovým systémem s využitím injekce kontrastní látky SonoVue. V sekvenci je kromě střevní tkáně také patrná střevní tepna. Na obrázku 6.1 je jeden ze snímků sekvence s vyznačenou oblastí zájmu pro registraci, kde docházelo k nejvýraznějším změnám kontrastu.



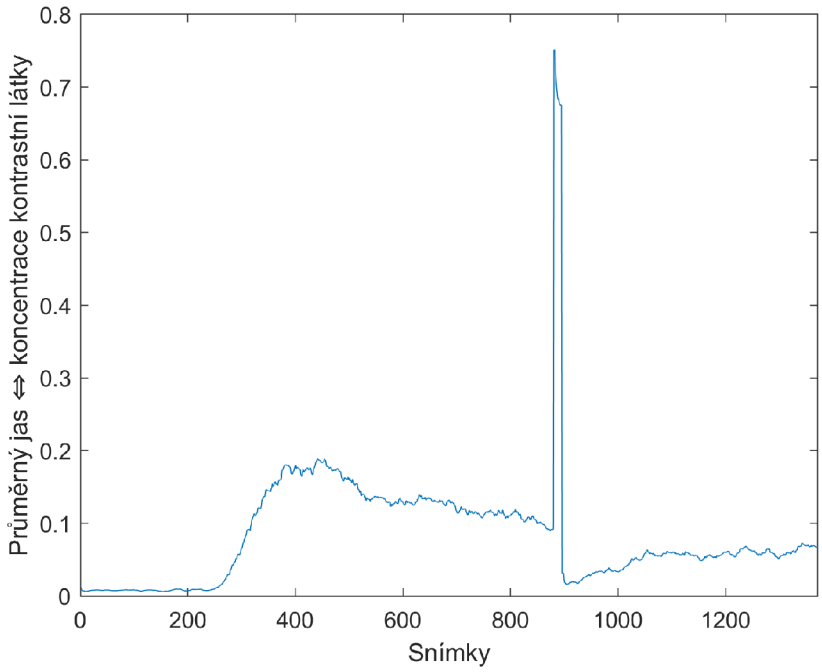
Obr. 6.1: Snímek z klinické abdominální sekvence s vyznačenou oblastí zájmu

Data pro registraci byly získány z první harmonické komponenty, která dává lepší přehled o jednotlivých strukturách tkání. Hodnotnější informaci o měnícím se kontrastu a koncentraci kontrastní látky v jednotlivých snímcích sekvence dává druhá harmonická komponenta.

6.1 Informace o kontrastu a převzorkování sekvence

Klinická sekvence měla původně 1376 snímků s obrazovou frekvencí 15 FPS (přibližně 90 sekund záznamu) a byla pro účely této práce upravena tak, aby zachycovala průběh mezi náběhem kontrastní látky a aplikací vysokofrekvenčního pulzu, což byl

přibližně čas od 10 do 40 sekundy, kdy z tohoto intervalu byl pro další testování brán každý pátý snímek. Registrace tak probíhala na klinických datech o délce 100 snímků. Průběh kontrastu v původní sekvenci, ze které byl pro registraci brán v úvahu pouze určitý interval, lze vidět na obrázku 6.2, který byl získán z oblasti zájmu druhé harmonické komponenty.



Obr. 6.2: Průběh kontrastu v druhé klinické sekvenci

6.2 Srovnání optimalizačních metod na druhé klinické sekvenci

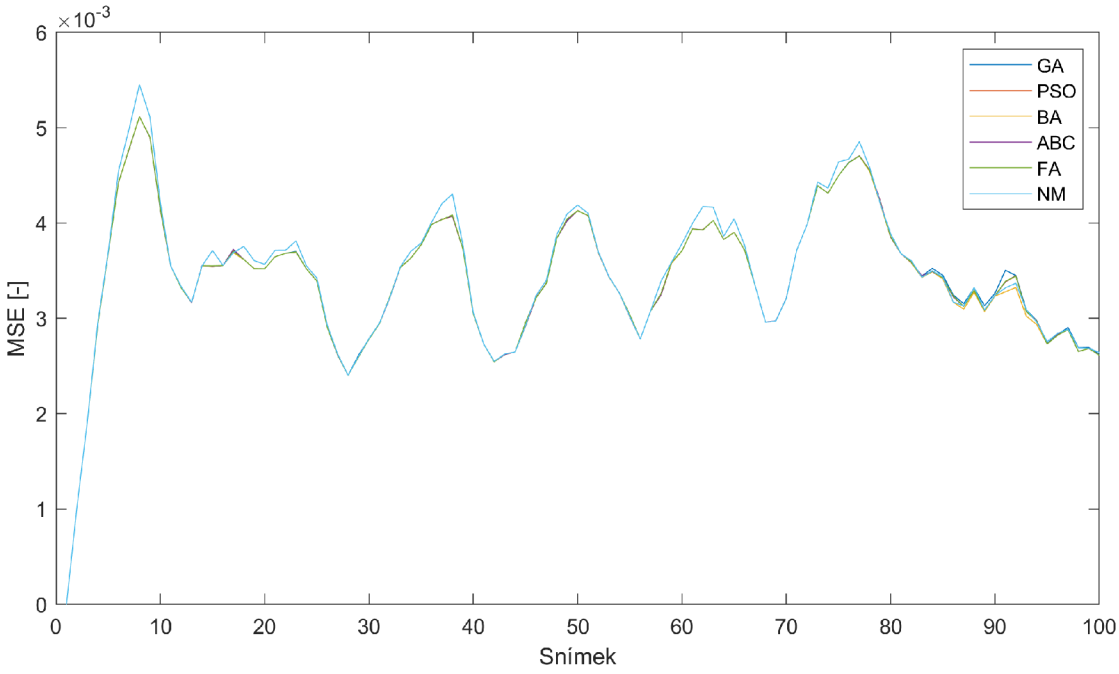
Tak jako v případě první klinické sekvence, i zde nebyly známy referenční hodnoty, pomocí kterých by bylo možné jednoznačně ověřit správnost registrace, která opět probíhala mezi prvním referenčním a zbylými snímky sekvence. Jednotlivé optimalizační algoritmy s optimálně zvolenými nastaveními z kapitoly 4 byly na těchto dynamických datech o délce 100 snímků spuštěny, kdy v rámci optimalizačního procesu dokázaly všechny metody opět minimalizovat podobnostní kritérium MSE na zanedbatelně rozdílné hodnoty, které jsou vyneseny v tabulce 6.1 a byly získány z celkem tří běhů každého algoritmu. Těchto velice si podobných výsledků však každý z algoritmů dosáhl za rozdílný čas, přičemž nejrychlejší byl algoritmus PSO, kterému se podařilo nalézt parametry transformace na celé sekvenci snímků během přibližně 4 minut. Zajímavé je, že neevoluční algoritmus Nelder-Meadův, který narozdíl od

klinické sekvence nádoru pankreatu, kde se globálně od ostatních evolučních metod lišil v řádu setin, zde vykazoval rozdíly mnohem menší. Jeho rozdíl ve výpočetním čase je oproti algoritmu BA více jak třínásobný, což je velmi dobrý výsledek s ohledem za to, jak si vedl na předchozí sekvenci.

Tab. 6.1: Srovnání průměrných MSE dosažených minimálních hodnot na jednotlivých snímcích a průměrných časů pro jednotlivé optimalizační metody na druhé klinické sekvenci

Algoritmus	**MSE** [10^{-3}]	**Celkový čas** [s]
Spojitý GA	3,42706	614
PSO	3,42517	250
BA	3,42048	1476
ABC	3,42564	407
FA	3,42515	531
Nelder-Mead	3,42846	466

Průměrné MSE dosažené u jednotlivých registrovaných dvojic snímků byly vyneseny do obrázku 6.3, kdy na 38. snímku lze pozorovat rozdíl v MSE dosažené Nelder-Meadem a v průměru ostatními evolučními algoritmy o více jak 2 desetitisíciny, což byl v průměru přibližný rozdíl v posunu osy x o 3 pixely, v ose y o 4 pixely a v úhlu natočení o 4° Nedler-Meadovou metodou oproti ostatním algoritmům.



Obr. 6.3: Průběh průměrných MSE na druhé klinické sekvenci

Závěr

Prvním cílem této práce bylo prostudování potřebné teorie týkající se ultrazvukového zobrazování. Na tento popud byla vytvořena první kapitola práce, týkající se ultrazvuku v lékařství, principu a vzniku ultrazvuku a ultrazvukového zobrazení s kontrastními látkami.

V rámci druhého cíle práce je věnována pozornost teoretickému popisu registrace obrazu jako metodě zarovnání obrazů, pro které jsou hledány vhodné parametry transformace a kriteriální funkce, která popisuje kvalitu transformace a je během optimalizačního procesu maximalizována. V souladu s tímto cílem práce byla dále vytvořena kapitola týkající se optimalizačních a evolučních algoritmů, které se využívají k nalezení optimálních parametrů transformace, jež se při procesu registrace nastavují.

Praktická část práce ve čtvrté kapitole popisuje realizaci postupů hledání optimálních parametrů užitých optimalizačních metod pro registraci umělé sekvence v programovém prostředí MATLAB. Pro její zrealizování byla na simulované sekvenci nejprve nadefinována transformace obrazů, kdy byly uvažovány posuny a rotace, které obrazová data náhodně vychýlily. Dále došlo k implementaci kriteriální funkce MSE pro vyjádření podobnosti dvou obrazů, která byla během optimalizačního procesu minimalizována. Optimalizační evoluční metody, které byly pro tuto práci vybrány, byly spojitý genetický algoritmus (GA), Particle Swarm Optimalizace (PSO), včelí algoritmus (BA), algoritmus umělé včelí kolonie (ABC) a světluškový algoritmus (FA). Pro srovnání s jinými optimalizačními technikami byly algoritmy inspirované přírodou porovnány s neevolučním heuristickým Nelder-Meadovým algoritmem. Jelikož byly snímky na této sekvenci uměle zkresleny známými hodnotami posunů v osách x a y a rotací, mohla být správnost registrace ověřena porovnáním s nalezenými hodnotami na výstupu optimalizačních algoritmů. Jako směrodatné kritérium a hlavní metrika pro ověření úspěšnosti registrace dvojice obrazů však sloužila střední kvadratická chyba (MSE), kterou se podařilo na umělé sekvenci všemi algoritmy minimalizovat až na hodnoty blížící se téměř nule. Její hodnota je značně ovlivněna přítomným množstvím kontrastní látky v jednotlivých snímcích. Nejvýznamnějšími parametry z hlediska optimalizace se ukázaly být počet jedinců a maximální počet iterací, které nejvíce ovlivňovaly výpočetní náročnost jednotlivých optimalizačních metod. Výpočetní čas jednotlivých algoritmů byl pro nalezení optimálních parametrů transformace velmi variabilní. Nejdéle trval optimalizační proces algoritmu včel (BA), který byl oproti roji částic PSO téměř trojnásobný. Všechny algoritmy byly nastaveny na pevný počet iterací, v případě využití jiné ukončovací podmínky by výsledky mohly být jiné.

Poslední dvě kapitoly jsou věnovány registraci na reálných klinických sekvencích

nádoru pankreatu a tenkého střeva, na kterých byly jednotlivé algoritmy s nejlepšími optimálně nalezenými parametry otestovány.

Literatura

[1] HRAZDIRA, Ivo. *ÚVOD DO ULTRASONOGRAFIE* [online]. Dostupné z URL: <https://docplayer.cz/4239393-Uvod-do-ultrasonografie.html>

[2] NAVRÁTIL, Leoš a Jozef ROSINA. *Medicínská biofyzika*. [Praha]: Grada, 2005. 524 s. ISBN 978-80-247-1152-2.

[3] SEDLÁŘ, Martin, Erik STAFFA a Vojtěch MORNSTEIN. *Zobrazovací metody využívající neionizující záření*. 1. vydání. [Brno]: Masarykova univerzita, 2014. 210 s. ISBN 978-80-210-7156-8.

[4] PAEFGEN, Vera, Dennis DOLESCHEL a Fabian KIESSLING. *Evolution of contrast agents for ultrasound imaging and ultrasound-mediated drug delivery*. Frontiers in Pharmacology. 2015, 6. Dostupné z doi: 10.3389/fphar.2015.00197.

[5] GOLDBERG, Barry B., Ji-Bin LIU, a Flemming FORSBERG. *Ultrasound contrast agents: a review*. Ultrasound in medicine & biology, 1994, 20, 4, s. 319–333. Dostupné z doi: 10.1016/0301-5629(94)90001-9.

[6] LIU, Ji-Bin, Gervais WANSAICHEONG, Daniel A. MERTON, Flemming FORSBERG a Barry B. GOLDBERG. *Contrast-enhanced Ultrasound Imaging: State of the Art*. Journal of Medical Ultrasound. 2005, 13, 3, s. 109-126. ISSN 0929-6441. Dostupné z URL: <https://doi.org/10.1016/S0929-6441(09)60100-4>.

[7] JAN, Jiří. *Medical Imgae Processing, Reconstruction and Restoration: Concepts and Methods*. Taylor & Francis Group, 2006. ISBN 0-8247-5849-8.

[8] SARVAMANGALA, D. R. a Raghavendra V. KULKARNI. *A Comparative Study of Bio-inspired Algorithms for Medical Image Registration*. Advances in Intelligent Computing, 2019, 27-44. ISBN 978-981-10-8973-2. Dostupné z doi: 10.1007/978-981-10-8974-9_2.

[9] BHATTACHARYA, Mahua a Arpita DAS. *Multimodality Medical Image Registration and Fusion Techniques Using Mutual Information and Genetic Algorithm-Based Approaches*. Software Tools and Algorithms for Biological Systems, 441–449. Dostupné z doi: 10.1007/978-1-4419-7046-6_44.

[10] SARVAMANGALA, D. R. a Raghavendra V. KULKARNI. *Swarm Intelligence Algorithms for Medical Image Registration: A Comparative Study*. International Conference on Computational Intelligence, Communications, and Business Analytics, 2017. Dostupné z doi: 10.1007/978-981-10-6430-2_35.

[11] VALSECCHI, Andrea, Sergio DAMAS a José SANTAMARIA. *An image registration approach using genetic algorithms*. IEEE Congress on Evolutionary Computation, 2012, 1-8. Dostupné z doi: 10.1109/CEC.2012.6256162.

[12] VALSECCHI, Andrea, Sergio DAMAS a José SANTAMARIA. *Evolutionary Intensity-based Medical Image Registration: A Review*. Current Medical Imaging Reviews. 2013, 9.

[13] TVRDÍK, Josef. *Evoluční algoritmy*. Skripta, Přírodovědecká fakulta, Ostravská univerzita, 2004.

[14] SIVANANDAM, S. N. a S. N. DEEPA. *Genetic Algorithm Optimization Problems*. Introduction to Genetic Algorithms, 165–209. Dostupné z doi:10.1007/978-3-540-73190-0_7

[15] RAIDA, Zbyněk. *Optimalizace v elektrotechnice*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav radioelektroniky. Dostupné z URL: <`http://www.urel.feec.vutbr.cz/~raida/optimalizace/index.htm`>

[16] ŠTEFKA, David. *Systém pro studium genetických algoritmů v kontextu tradičních stochastických optimalizačních metod*. Praha: České vysoké učení technické v Praze, Fakulta jaderná a fyzikálně inženýrská, Katedra matematiky, 2004. Dostupné z URL: <`http://home.tiscali.cz/davidstefka/vyzkumak/vyzkumak.html`>

[17] DOSTÁL, Zdeněk a Petr BEREMLIJSKI. *Metody optimalizace*. Skripta VŠB TU Ostrava a ZU v Plzni, 2012. Dostupné z URL: <`https://homel.vsb.cz/~ber95/MetodyOptimalizace/metody_optimalizace.pdf`>

[18] ANON., nedatováno. *Genetic Algorithms Tutorial* [online]. Tutorialspoint, 2016. 36 s. Dostupné z URL: <`https://www.tutorialspoint.com/genetic_algorithms/index.htm`>

[19] WU, Ruoyun, Keck Voon LING, Wei SHAO a Wan Sing NG. *Registration of Organ Surface with Intra-operative 3D Ultrasound Image Using Genetic Algorithm*. Medical Image Computing and Computer-Assisted Intervention - MICCAI 2003. MICCAI 2003. Lecture Notes in Computer Science, vol 2878, 383–390. ISBN 978-3-540-20462-6. Dostupné z doi: 10.1007/978-3-540-39899-8_48.

[20] INGOLE, V. T., C. N. DESHMUKH, A. JOSHI a D. SHETE. *Medical Image Registration Using Genetic Algorithm*. "Second International Conference on

Emerging Trends in Engineering & Technology, 2009, 63-66. Dostupné z doi: 10.1109/ICETET.2009.143.

[21] LEITE, J. V., S. L. AVILA, N. J. BATISTELA, W. P. CARPES, Jr., N. SADOWSKI, P. KUO-PENG a J. P. A. BASTOS. *Real coded genetic algorithm for Jiles-Atherton model parameters identification.* IEEE Transactions on Magnetics, 40(2), 888–891. Dostupné z doi:10.1109/tmag.2004.825319.

[22] ZELINKA, Ivan, Zuzana OPLÁTKOVÁ, Pavel OŠMERA, Miloš ŠEDA a František VČELAŘ. *Evoluční výpočetní techniky – principy a aplikace.* BEN, Praha, 2009. ISBN 80-7300-218-39788073002183.

[23] HAUPT, Randy L. a Sue Ellen HAUPT. *Practical Genetic Algorithms.* John Wiley & Sons, New Jersey, 2004. ISBN 0-471-45565-2.

[24] YUCE, Baris, Michael S. PACKIANATHER, Ernesto MASTROCINQUE, Duc Truong PHAM a Alfredo LAMBIASE. *Honey Bees Inspired Optimization Method: The Bees Algorithm.* Insects 2013, 4, s. 646-662. Dostupné z doi: 10.3390/s151025033.

[25] PHAM, D. T., Afshin GHANBARZADEH, Ebubekir KOÇ, Sameh OTRI, Sahra RAHIM a Mb ZAIDI. *The Bees Algorithm Technical Note.* Manufacturing Engineering Centre, 2005, 1–57. Dostupné z URL:<`https://www.researchgate.net/publication/260985621_The_Bees_Algorithm_Technical_Note`>

[26] ALOMARI, Osama a Zulaiha Ali OTHMAN. *Bees Algorithm for feature selection in Network Anomaly detection.* Journal of Applied Sciences Research, 2012, 8(3). Dostupné z URL:<`https://www.researchgate.net/publication/260985621_The_Bees_Algorithm_Technical_Note`>

[27] NEBTI, Salima. *Bio-Inspired Algorithms for Color Image Segmentation.* International Journal of Computer Applications, 2013, 73(18), 11-16. Dostupné z doi: 10.5120/12840-9810.

[28] KUMAR, Balwant a Dharmender KUMAR. *A review on Artificial Bee Colony algorithm.* International Journal of Engineering and Technology, 2 (3), 2013, s. 175-186. Dostupné z URL:<`https://www.sciencepubco.com/index.php/ijet/article/view/1030/759`>

[29] AKAY, Bahriye a Dervis KARABOGA. *Parameter Tuning for the Artificial Bee Colony Algorithm.* Lecture Notes in Computer Science, 608–619. Dostupné z doi: 10.1007/978-3-642-04441-0_53.

[30] ABDEL-BASSET, Mohamed, Ahmed E. FAKHRY, Ibrahim EL-HENAWY, Tie QIU a Arun Kumar SANGAIAH. *Feature and Intensity Based Medical Image Registration Using Particle Swarm Optimization*. Journal of Medical Systems, 41, 197 (2017). Dostupné z doi: 10.1007/s10916-017-0846-9.

[31] YANG, Xin-She. *Nature-Inspired Metaheuristic Algorithms, Second Edition*. Luniver Press, 2010. ISBN 978-1-905986-28-6.

[32] YANG, Xin-She. *Firefly Algorithms for Multimodal Optimization*. Stochastic Algorithms: Foundations and Applications, SAGA 2009, Lecture Notes in Computer Science, Vol. 5792, 169–178. Dostupné z doi: 10.1007/978-3-642-04944-6_14.

[33] COCIANU, Catalina-Lucia a Alexandru STAN. *New Evolutionary-Based Techniques for Image Registration*. Applied Sciences, 9(1), 176. Dostupné z doi: 10.3390/app9010176.

[34] MO, Yuan-bin, Yan-zhui MA a Qiao-yan ZHENG. *Optimal Choice of Parameters for Firefly Algorithm*. Fourth International Conference on Digital Manufacturing & Automation, 2013. Dostupné z doi:10.1109/icdma.2013.210.

[35] HERIS, Mostapha Kalami. *Practical Genetic Algorithms in Python and MATLAB*. Yarpiz, 2020. Dostupné z URL:<`https://yarpiz.com/632/ypga191215-practical-genetic-algorithms-in-python-and-matlab`>

[36] SORSA, A., R. PELTOKANGAS a K. LEIVISKA. *Real-coded genetic algorithms and nonlinear parameter identification*. 4th International IEEE Conference Intelligent Systems, 2008. Dostupné z doi: 10.1109/IS.2008.4670495.

[37] MARIE-SAINTE, Souad Larabi, Reham ALSKIREEN a Sawsan ALHALAWANI. *Emerging Applications of Bio-Inspired Algorithms in Image Segmentation*. Electronics 2021, 10(24), 3116. Dostupné z URL:<`https://doi.org/10.3390/electronics10243116`>

[38] HERIS, Mostapha Kalami. *Particle Swarm Optimization (PSO) in MATLAB*. Yarpiz, 2016. Dostupné z URL:<`https://yarpiz.com/440/ytea101-particle-swarm-optimization-pso-in-matlab-video-tutorial`>

[39] HERIS, Mostapha Kalami. *Bees Algorithm (BeA) in MATLAB*. Yarpiz, 2015. Dostupné z URL:<`https://yarpiz.com/315/ypea115-bees-algorithm`>

[40] HERIS, Mostapha Kalami. *Artificial Bee Colony in MATLAB*. Yarpiz, 2015. Dostupné z URL:<`https://yarpiz.com/297/ypea114-artificial-bee-colony`>

[41] HERIS, Mostapha Kalami. *Firefly Algorithm (FA) in MATLAB*. Yarpiz, 2015. Dostupné z URL:<https://yarpiz.com/259/ypea112-firefly-algorithm>

Seznam příloh

A Obsah CD **95**

A Obsah CD

/ ..... Soubory DP Votýpka

- ABC.m
- BA.m
- FA.m
- GA.m
- mutace.m
- PSO.m
- reg_ABC_MSE.m
- reg_BA_MSE.m
- reg_FA_MSE.m
- reg_NM.m
- reg_PSO_MSE.m
- reg_spojity_GA_MSE.m
- ruletova_selekce.m
- serazeni_populace.m
- tanecni_ritual.m
- uniformni_prekrizeni.m
- vazena_ruleta.m
- vytvoreni_nove_sekvence.m
- READ_ME.m