Aus dem
Deutschen Krebsforschungszentrum Heidelberg
(Vorstandsvorsitzender: Prof. Dr. Michael Baumann)

Abteilung Medizinische Physik in der Strahlentherapie
(Leiter: Prof. Dr. Oliver Jäkel)

Multi-parametric optimization of magnetic resonance-imaging sequences for magnetic resonance-guided radiotherapy

---

Inauguraldissertation
zur Erlangung des Doctor scientiarum humanarum (Dr. sc. hum.)
an der
Medizinischen Fakultät Heidelberg
der
Ruprecht-Karls-Universität

vorgelegt
von
Hafiz Muhammad Fahad
aus
Lahore, Pakistan
March, 2024

Doktorvater: Herr Prof. Dr. Christian P. Karger

Table of Contents

Glossary v

List of Figures vii

List of Tables viii

1. **Introduction** 1

1.1. Thesis Outline 2

2. Material and methods 3

2.1. Radiotherapy 3

2.1.1. Treatment Planning and Radiotherapy Process 3

2.1.2. Safety margins 5

2.1.3. Image-guided Radiotherapy 6

2.2. Magnetic resonance imaging 7

2.2.1. Basic principles of MRI 8

2.2.2. Imaging sequences 10

2.3. Image Quality 12

2.3.1. Image Contrast 13

2.3.2. Image Noise 13

2.3.3. Image Artifacts 14

2.4. Regression Problem 15

2.4.1. Generalized Additive Model 16

2.4.2. Neural Networks 18

2.4.3. Artificial Neural Networks 19

2.4.4. Activation Layer and Activation Functions 21

2.4.4.1. Loss function 23

2.4.4.2. Neural Network Optimization 24

2.5. Optimization 26

2.5.1. Covariance matrix adaptation evolution strategy 28

2.5.2. Genetic algorithm 29

2.5.3. Multi Objective Optimization 33

Non-dominated Sorting Genetic Algorithm –III: 34

2.6. Segmentation 35

2.7. Experimental setup 36

2.7.1. Phantom materials 37

2.7.2. Regression based SPS Optimization 38

2.7.2.1. Data Acquisition 39
2.7.2.2. Regression problem 39
2.7.2.3. Optimization Process 40
2.7.2.4. Clinical use cases 40
i) Achieving the same contrast as in a target image 40
ii) Maximizing the contrast between different tissue types 41
2.7.3. On-the-run SPS optimization 41
2.7.3.1. Optimization loop via MRI interface 42
2.7.3.2. Optimization Process 42
i) Achieving the same contrast as in a target image 42
ii) Maximizing the contrast between different tissue types 43
3. Results 44
3.1. Segmentation 44
3.2. Regression based SPS optimization 44
3.2.1. Regression 44
3.2.2. Optimization 47
i) Achieving the same contrast as in a target image 47
ii) Maximizing the contrast between different tissue types 49
3.3. On-the-run SPS Optimization 51
i) Achieving the same contrast as in a target image 51
ii) Maximizing the contrast between different tissue types 53
4. Discussion 55
4.1. Regression-based Optimization method 55
4.1.1. Optimization 56
i) Achieving the same contrast as in a target image 56
ii) Maximizing the contrast between different containers 57
4.2. On-the-run optimization 58
i) Achieving the same contrast as in a target image 58
ii) Maximizing the contrast between different containers 59
4.3. Comparison of the applied methods: regression-based vs. on-the-run optimization 61
4.4. Future work 62
4.5. Conclusion 62
5. Summary 63
6. Zusammenfassung 64
7. Bibliography 65
8. Personal Contributions 70

Curriculum vitue 71
Acknowledgements 72
Eidesstattliche Versicherung 73

Glossary

MRI	magnetic resonance imaging
RT	radiotherapy
SPS	sequence parameter sets
TR	repetition time
TE	echo time
FA	flip angle
BW	bandwidth
TF	turbo factor
MR	magnetic resonance
CMA-ES	covariance matrix adaptation evolution strategy
GA	genetic algorithm
CT	computed tomography
RF	radiofrequency
TPS	treatment planning system
PET	positron emission tomography
IMRT	intensity-modulated radiotherapy
TCP	tumour control probability
NTCP	normal tissue complication probability
GTV	gross tumor volume
CTV	clinical target volume
OAR	organs at risk
ITV	internal target volume
PTV	planning target volume
IGRT	image-guided radiotherapy
MRgRT	magnetic resonance-guided radiotherapy
SNR	signal-to-noise ratio
CNR	contrast-to-noise ratio
FID	free-induction decay
PD	proton density
SE	spin-echo
TSE	fast or turbo spin-echo
ETL	echo train length
MS	matrix size
FOV	field of view
ST	slice thickness
SAR	specific absorption rate
NSA	number of signal averages
OLS	ordinary least squares
GAM	generalized additive model
GLM	generalized linear model
ANN	artificial neural network

ML	machine learning
tanh	tangens-hyperbolicus
ReLU	rectified linear unit
MSE	mean squared error
MAE	mean absolute loss
GD	gradient descent
Adam	adaptive moment estimation
KKT	karush-kuhn-tucker
EA	evolutionary algorithms
MOO	multi-objective optimization
NSGA	non-dominated sorting genetic algorithm
DL	deep learning
ROIs	regions of interest
SGD	stochastic gradient descent

List of Figures

Figure 1: Illustration of a typical radiotherapy workflow.. 4
Figure 2: Graphical representation of the definition of the target. 6
Figure 3: Sequence diagram illustrating the steps of a Spin Echo sequence. 11
Figure 4: (a) Schematic representation of the longitudinal relaxation....12
Figure 5: Fitting a GAM model as an example to a random dataset....17
Figure 6: (a) Illustration of the fundamental structure of an artificial neural network.....20
Figure 7: Comparison of different Activation Functions in Neural Networks. 22
Figure 8: Sigmoid function and its derivative. 23
Figure 9: Visualization of Gradient Descent Optimization method. 26
Figure 10: CMA-ES flowchart. In each iteration(generation) step, ...29
Figure 11: Genetic Algorithm Pipeline. 33
Figure 12: Provides an overview of the various investigations conducted in this thesis. 37
Figure 13: Left) Shows the real phantom which is used for 38
Figure 14: Visualization of a deep learning model for our regression problem. 40
Figure 15: Flow chart of the proposed "on-the-run" optimization workflow. 43
Figure 16: Illustration of the segmentation process using the region growth method. 44
Figure 17: Measurement (dots) and model predictionl. 45
Figure 18: Contrast predicted with GAM (a, c) and DL-based regression (c, d)....46
Figure 19: A comparison of the MSE loss functions for the GA and CMA-ES....48
Figure 20: Comparison of the MSE loss functions for the CMA-ES optimization method 48
Figure 21: Development of TE (a), TR (b), TF (c), and FA (d) parameters 49
Figure 22: Results of the multi-objective optimization using NSGA-III 50
Figure 23: Pareto front optimal solutions obtained by using the NSGA-III....51
Figure 24: A comparison of the MSE loss functions 52
Figure 25: Shows the development of TE (a), TR (b) and FA (c) parameters. 53
Figure 26: Illustration of the impact of the two sets of weighting factors: 54

List of Tables

Table 1: Contrast materials providing specific T1 and T2 values at a 1.5 T MRI and 0.35 T MR-Linac. Values as reported in [87]. 38

Table 2: Quantitative evaluation of contrast regression models by using MAE and MSE losses. 45

Table 3: Signal and contrast for the measured substitutes for two different sets of weighting factors. 50

Table 4: Signal and contrast for the measured substitutes for two different sets of weighting factors. 54

1. Introduction

Magnetic Resonance Imaging (MRI) has been integrated into oncology for staging, assessing tumor response, and also for radiation therapy (RT) planning, with the advantages of superior soft-tissue imaging contrast and continuous real-time imaging, which can facilitate tumor and organ-at-risk delineation as well as image registration [1-5]. The large variety of imaging contrasts in MRI is associated with a large number of different pulse sequence parameter sets (SPS), which have a direct impact on image quality, contrast, acquisition time and efficiency of further image processing. Depending on the sequence and the clinical objective, these SPS can consist of up to 30 different parameters (repetition time (TR), echo time (TE), flip angle (FA), bandwidth (BW), turbo factor (TF) and averages, etc.). Each of these parameters directly influences image contrast, image quality, or acquisition time. As many pulse sequences are often not fully optimized to the needs of a specific clinical scenario, additional sequence optimization is often performed manually, which can be cumbersome and time-consuming. Machine learning-based models can help to simplify and automate such tasks, however, to train these models a large amount of data with different SPS needs to be collected and analyzed. Again, manual acquisition of this data at the scanner is a time-consuming procedure that requires repeated human interventions to change the SPS settings, and automation of this acquisition process is preferred. For this, several tools have been presented in literature. “Pulseq” [6] is a high-level, flexible, and hardware-independent open-source framework for the rapid development, representation, and execution of magnetic resonance (MR) sequences. This tool allows users to create customized sequences by applying different schemes of RF pulses and gradients. By utilizing the Pulseq interpreter, these sequences can be exported and executed on an MRI device.

Recently, the self-learning framework ‘MR-zero’ [7], utilizing the Pulseq-tool, has been proposed, which adapts and optimizes MRI sequences based on a Bloch equation simulation. The generated pulse sequence, still requires knowledge of Bloch simulation in order to perform MR sequence optimization. In the more advanced version "MR-double-zero" [8], the “Pulseq” tool is still utilized to remotely control the scanner, however, the optimization directly operates on the acquired imaging data without requiring a Bloch simulation model or any further human interaction. As a prerequisite for implementing clinical sequences with Pulseq, a detailed prior knowledge of the manufacturer pulse schemes with the exact timings of the gradients and RF pulses used in the respective clinical sequence is required. The clinical sequence has then to be built from scratch within Pulseq mimicking as closely as possible the selected sequence.

Just recently, a real-time scanner remote control tool ‘Access-i’ (Siemens Healthineers, Erlangen, Germany) has been introduced, which resolves this problem by allowing the user to access all sequences implemented on the scanner and to change MRI parameters via a script. This tool has been used to realize an automated sequence optimization tool that includes an iterative change of SPS that are executes on the MRI scanner followed by an automated evaluation of the resulting images based on a predefined optimization goal.

The proposed frameworks, I) regression-based optimization and II) on-the-run optimization, for automatic optimization of MRI sequences are based on SPS that are directly applied on the scanner. Two clinically relevant optimization goals were pursued: i) achieving the same signal and thus contrast as in a target image, and ii) maximizing the signal difference between specified tissue types. Furthermore, the proposed framework is evaluated using two different optimization methods, a covariance matrix adaptation evolution strategy (CMA-ES) and a genetic algorithm (GA). The obtained results demonstrate the potential of the proposed framework for automatic contrast optimization of MRI sequences, which can improve the application of MRI for application in radiotherapy planning. Most of the Introduction material have been extracted from our previous publication [88].

1.1. Thesis Outline

This thesis is structured into five chapters. Following the introduction section, Chapter 2 presents the materials and methods employed in the research. It discusses the physical foundations of radiotherapy, treatment planning and the radiotherapy process, as well as MRI and image quality. Additionally, it introduces the basics of regression and optimization problems, along with possible solution methods. The chapter further explains the experimental setup, including phantom materials and two optimization pipelines: I) Regression-based optimization and II) On-the-run optimization. It also compares two different evolutionary optimization methods: CMA-ES and GA.

Chapter 3 is dedicated to presenting the results achieved by using both SPS optimization workflows, providing a comprehensive overview of the outcomes and findings. In Chapter 4, a detailed discussion of the results from both SPS optimization workflows is provided, Finally, Chapter 5 presents the main conclusions drawn from this thesis, summarizing the key findings and their implications for future research in the field.

2. Material and methods

2.1. Radiotherapy

Radiotherapy is one of the most important strategies for cancer treatment together with surgery, immunotherapy, and chemotherapy [9-11]. Radiotherapy uses ionizing radiation to induce irreparable damage in tumor tissue [12]. The common goal of any radiotherapy is to deliver the medically prescribed dose, defined as the energy E absorbed by the mass m, to the treatment volume while sparing the surrounding healthy tissue as much as possible. This is particularly important for certain organs at risk (OARs) where explicit dose limits exist.

$$D = \frac{dE}{dm} \qquad \textit{Equation 2-1}$$

There are different types of radiation used, depending on the type and location of the tumor among other factors. Ion beam therapy uses positively charged protons or heavier ions to deliver the dose. Protons and ions can be precisely targeted to the tumor site, minimizing damage to surrounding healthy tissue. Proton and ion treatments are commonly used to treat tumors of the brain, spine, and prostate, among others [13]. Photon therapy uses high-energy photons from a linear accelerator to deliver the dose. Photons are commonly used to treat tumors of the lung, breast, and head and neck, among others [14].

2.1.1. Treatment Planning and Radiotherapy Process

The treatment planning process is comprised of several steps, starting with the acquisition of imaging data of the patient, usually computed tomography (CT) images that serves as patient model for treatment planning. After optimizing the treatment plan for the patient, plan is approved, verified my measurement or independent recalculation and finally delivered to the patient. Figure 1 depicts all the main components of the treatment planning workflow, emphasizing the structure of the treatment planning system (TPS) and the dose optimization.

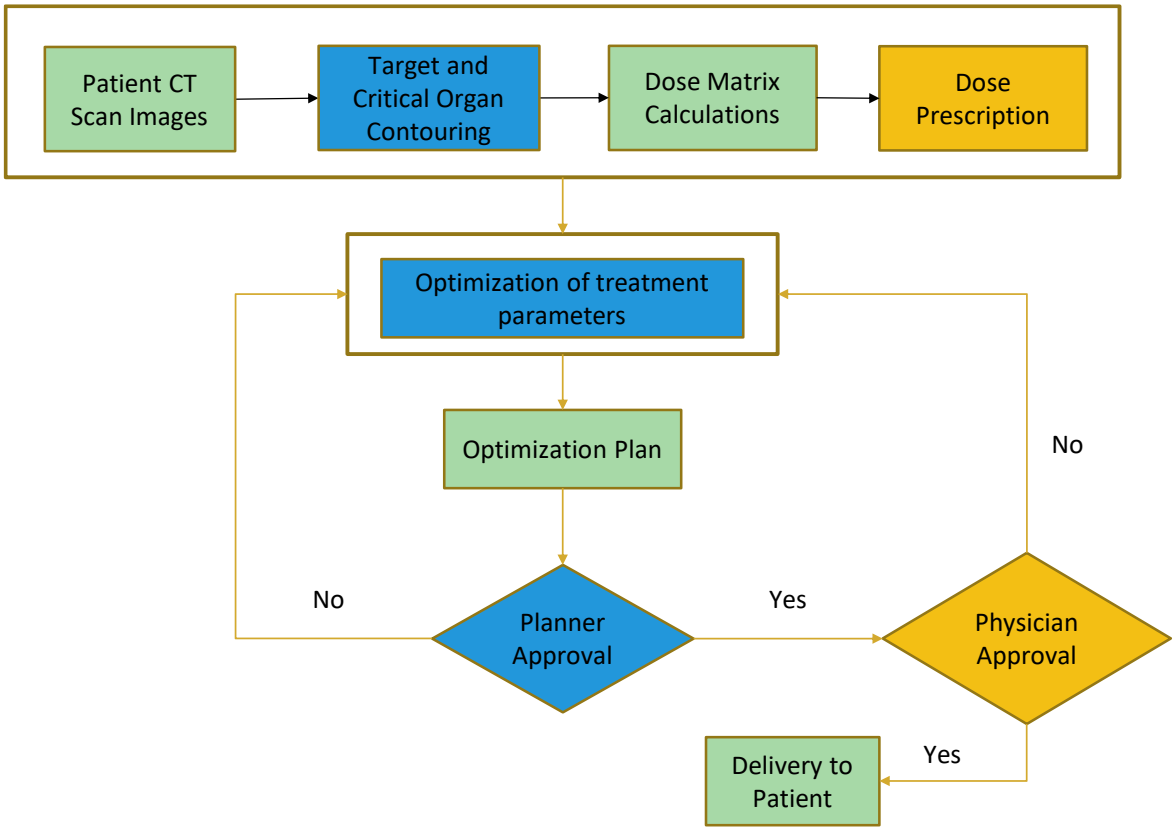

Figure 1: Illustration of a typical radiotherapy workflow. Green colored boxes show processes that are handled automatically by a clinical software or hardware, and blue boxes indicate processes that require a human treatment planner. Orange boxes show the processes that require physicians. Adopted from [15].

To start treatment planning, image data must be acquired for each patient. CT imaging is the current standard for 3D imaging in radiation oncology. There are two main reasons for choosing CT for radiotherapy planning: first, the CT image contains electronic density information, which is a key component for dose calculation. Second, a CT image is geometrically robust, allowing accurate targeting of the tumour during treatment planning and delivery [16] . However, geometric uncertainties may arise from the limited visibility of the tumour on CT. To overcome this problem and to better identify anatomical regions of interest, planning CT can be combined with other diagnostic imaging modalities such as positron emission tomography (PET), MRI or other diagnostic CT protocols.

The main goal of radiotherapy treatment planning is to optimize the dose in areas that need to receive the prescribed dose while reducing the dose in radiosensitive areas. For intensity-modulated radiotherapy (IMRT), the optimization process, uses an “inverse planning” technique, which is of fundamental importance in modern radiotherapy treatment planning [17]. Initially, the radiation oncologist prescribes the dose to the tumor based on his or her knowledge and experience, taking into account factors such as expected tumour control probability (TCP) and normal tissue complication probability (NTCP). The optimization process

aims to identify a set of treatment parameters, including beam angles and pencil beam intensities and locations that achieve the desired clinical goals best. During treatment plan optimization, the dose is recalculated iteratively in each step of the optimization loop.

$$d_i = \sum_j D_{ij} w_j \qquad \text{Equation 2-2}$$

To save the computation time, the dose contribution from each pencil beam (photon or proton) is pre-calculated and stored on the dose influence matrix D_{ij}, representing the dose to each voxel. The absorbed dose d_i is computed by summing dose contribution from intensity weighted elementary pencil beams w_j (with $w_j > 0$). The objective function in the treatment planning can be define as weighted/penalized least square:

$$\min_w F(d) = \arg\min_w (Dw - d^*)^T P (Dw - d^*)$$

Where $P = diag(p_1, p_2, \ldots . p_l)$ is the penalized diagonal matrix that allows to penalize between target volume and OARs.

Typically, the dose is delivered to the patient in a fractionated way whith typical doses of 2 Gy per day. This improves the sparing of normal tissue and OARs as their repair capacity is larger than that of the tumor [18, 19].

The fractionated approach of radiation therapy requires high precision in the application of radiation. This is especially true for highly conformal photon therapy or particle therapy, which includes high gradients in the dose distribution. Even a slight change in patient position may have a significant impact on the treatment outcome [14].

2.1.2. Safety margins

The concept of safety margins compensates for motion-induced under-dosage of tumor volume. The volumes used in treatment planning are defined by ICRU report 50 and 62 [20, 21]. The Gross Tumor Volume (GTV) is the visible or palpable tumor and any contiguous tumor extensions [21-23]. The Clinical Target Volume (CTV) is the GTV plus any subclinical or microscopic malignant disease that cannot be detected by imaging but is likely to be present based on the tumor type, location, and stage [21, 23]. The Planning Target Volume (PTV) is the CTV plus a margin that accounts for uncertainties in patient setup and organ motion during treatment delivery [21, 24]. Organs at Risk (OAR) are normal tissues or organs that are close to the target volume and are at risk of receiving a high dose of radiation during treatment. In addition to the four main volumes of interest, another volume is defined in the ICRU Report No. 62 to account for the effects of variation in size and position of the CTV, which accounts for organ motion. This results in an Internal Target Volume (ITV), which is the enveloping volume of the CTV in all motion states. All volumes are graphically presented in Figure 2

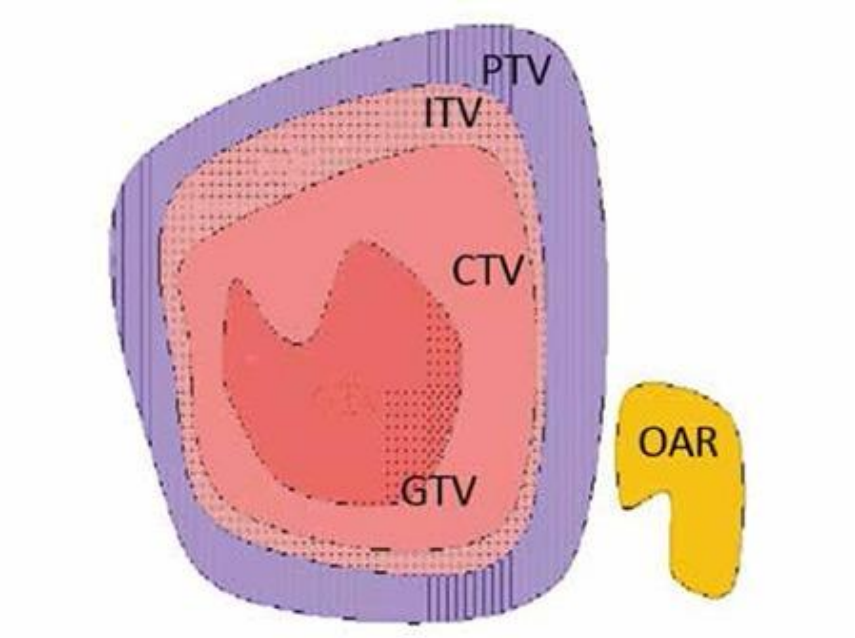

Figure 2: Graphical representation of the definition of the target volumes and organs at risk for use in treatment planning according to the ICRU reports 50 and 62. Adapter from [25].

Sufficient image contrast in imaging is important for radiotherapy to define the CTV, GTV and ITV as well as the OAR. This is especially important when the tumor is located close to critical organs or tissues. Contrast agents may be used to enhance the visibility of the tumor and to accurately delineate the target volume. This allows for more precise treatment planning and delivery, which can help to minimize the risk of normal tissue damage [26].

MRI offers superior contrast in imaging compared to CT, even without contrast agents and additional use of contrast agens additionally improves the contrast between tumor and normal tissue [27]. In MRI, contrast agents contain Gadolinium, which is a paramagnetic substance that enhances the signal intensity in T_1-weighted images [28].

2.1.3. Image-guided Radiotherapy

Image-guided radiotherapy (IGRT) is used to reduce the effect of set-up errors and anatomical changes in the patient in fractionated radiotherapy [29]. Prior to each fraction, an image of the patient is acquired. In the simplest case, a translation vector and/or rotation of the treatment table is determined using rigid image registration. The most common imaging technologies for IGRT are CT on rails and cone beam CT imaging using x-ray tubes attached to the gantry. In more advanced techniques, the treatment plan may be adapted to anatomical changes. This approach is denoted as adaptive radiothetrapy [30].

More recently, the use of magnetic resonance-guided radiotherapy (MRgRT) has been clinically introduced for photon therapy and is also proposed for proton therapy [31, 32]. The use of MRI provides excellent soft tissue contrast without additional radiation exposure and

in principle allows real-time imaging during irradiation. More details on MRI are given section 2.2. Early results suggest that MRgRT may be a promising approach for the treatment of certain tumor types. For example, a recent study found that MRgRT was effective in treating prostate cancer with a low incidence of side effects [33].

One of the currently available MRgRT devices is the Viewray MRIdian Linac (Viewray Inc., Oakwood, USA), which operates at a magnetic field of 0.3 T [34]. As spatial resolution, Signal-to-Noise Ratio (SNR) and Contrast-to-Noise Ratio (CNR) improves at higher magnetic field strengths, it may be beneficial to integrate diagnostic images of high-field MRI devices into the treatment**.** However, these images must be aligned with those in the Viewray MRIdian Linac for further use. This so-called image registration is performed by registration algorithms that either rigidly or deformably transform the diagnostic images.

2.2. Magnetic resonance imaging

MRI originated from the discovery of nuclear magnetic resonance by Bloch [35], followed by the first image acquisition by Lauterbur [36]. MRI is an imaging technique that provides detailed anatomical and physiological information about patients. Unlike CT scans, MRI does not use ionizing radiation. Instead, it utilizes a combination of a static magnetic field and radiofrequency (RF) pulses to excite nuclear hydrogen spins to obtain a spatially resolved signal from the body. When subjected to an external magnetic field, the spin of hydrogen nuclei aligns to the direction of the magnetic field and can be excited by applying oscillating magnetic fields (radiofrequency pulses). After excitation, the spins are precessing with a frequency proportional to the applied external magnetic field and in turn emit radio frequency signal which can be measured by a receiving coil. By using additional magnetic field gradients, the magnetization can be spatially encoded, allowing for the reconstruction of image slices through the body.

MRI is mainly used in medical diagnosis to visualize the structure and functionality of human tissues and organs. It offers high spatial resolution (approximately 1 mm) and superior soft tissue contrast. Various MRI sequences can provide not only spatial information but also functional insights. Examples include the measurement of perfusion, and diffussion (e.g. diffusion tensor imaging to visualize nerve fibers), dynamic contrast-enhanced MRI, used to measure microvascular parameters, or and magnetic resonance angiography, used to visualize blood vessels. MRI is often preferred over CT scans as it does not expose patients to radiation. However, individuals with certain metal implants containing are not suitable candidates for MRI scans due to possible induction of eddy-currents in the implants. Other drawbacks of MRI include loud sounds during the procedure, longer scan times, and the narrow space inside the scanner, which some patients may find uncomfortable.

2.2.1. Basic principles of MRI

All nuclei with an odd number of nucleons have a nuclear spin $\vec{J}$ in their ground state and a magnetic moment different from zero. These nuclei possess a dipole moment:

$$\vec{\mu} = \hbar\gamma\vec{J}$$

Where $\hbar$ is the reduced Planck constant and γ in the gyromagnetic ratio. For protons γ is $7.62259328547) \; MHz/T$.

There are several odd numbers of nuclei, such as ^{1}H, ^{3}He, ^{19}F, and ^{23}Na. Among these, the hydrogen atom (^{1}H) is the most commonly used in MRI due to its high natural abundance in human tissue. More than 80% of bodily tissue consists of water (H_2O) and fat (CH_2(OCOR)–CH(OCOR')–CH2(OCOR")) that contain a high number of hydrogen atoms, as evident in their chemical formulas [37]. Typically, an MRI image is generated based on the nuclei of these hydrogen atoms, i.e. protons. Each hydrogen nucleus possesses a magnetic moment, essentially acting as a tiny magnet. The high abundance of protons in tissue leads to a significantly stronger MRI signal as compared to other nuclei. Hence, the focus of the following description is on hydrogen nuclei.

Without an external magnetic field, the protons in the body are in thermal equilibrium and randomly distributed. However, when a strong external magnetic field B_0 is applied, the spins align along the direction of the B_0 field, either parallel or antiparallel. This creates a net magnetization vector along the field direction, called the longitudinal magnetization M_z . Under this condition, the net magnetization in the z-direction is static. The direction of the magnetization M can be manipulated by applying radio frequency (RF) pulses B_1 of different duration at Larmor frequency ω_0 to the body perpendicular to the main magnetic field B_0. This pulse causes the magnetization to be rotated away from the field direction by a certain angle, called the flip angle. The tilted magnetization results in a transverse magnetization M_{xy}, that rotates around the main field at a frequency that depends on the strength of the field, called the Larmor frequency.

$$\omega_0 = \gamma B_0 \qquad \textit{Equation 2-3}$$

The rotating transverse magnetization induces a voltage in a receiver coil placed around the body of the patient, which is detected and recorded by the scanner. This voltage is the MRI signal, and after spatial encoding, it contains information about the location and properties of the protons in the body.

Relaxation time

When an RF pulse is turned off, a relaxation process takes place, and the net magnetization immediately starts propagating back toward its equilibrium state, leading to an exponential decay of the transversal magnetization, called free-induction decay (FID). The relaxation process is caused by interactions between the spins and the molecular environment and can be divided into two processes, described by the Bloch Equation 2-4Equation 2-5.

Spin-lattice relaxation (also called longitudinal or T_1 relaxation time) is a measure of how quickly the net magnetisation vector (NMV) recovers to its ground state in the direction of B_0. The return of excited nuclei from the high energy state to the low energy or ground state is associated with loss of energy to the surrounding nuclei. T1 is the time required for the longitudinal magnetization to recover approximately 63% (1- (1/e)) of its initial value after being flipped into the transverse magnetic plane by a 90° radiofrequency pulse.

$$\frac{dM_z(t)}{dt} = \frac{M_o - M_z(t)}{T_1} \qquad \textit{Equation 2-4}$$

Spin-spin relaxation (also called transverse or T_2 relaxation) is a process that causes the progressive dephasing of the precessing dipoles, leading to a decay of the magnetisation in the transverse plane (M_{xy}). This type of relaxation occurs with the time constant T_2, which is the time it takes for the transverse magnetisation vector to decay to 1/e or 37% of its initial magnitude after a radiofrequency pulse.

$$\frac{dM_{xy}}{dt} = -\frac{M_{xy}(t)}{T_2} \qquad \textit{Equation 2-5}$$

In practice, the magnetic field B_0 will always have some spatial inhomogeneity ΔB_0, which results in a shorter relaxation time T_2^* compared to the spin-spin relaxation time T_2

Image acquisition and reconstruction

To generate a three-dimensional (3D) image, it is essential to capture spatial information in all three coordinates: x, y, and z. This is achieved through the use of a sequence of magnetic field gradients, namely G_x, G_y, and G_z, which lead to a spatial encoding of the Lamor-frequency.

First, a slice selection gradient G_{ss} is applied to the volume of interest, causing the frequency of the nuclei to vary in the direction of the gradient. Next, the position of each point is encoded vertically and horizontally within the selected slice by applying a phase encoding gradient G_{PE} and a frequency encoding gradient G_{FE}. The phase encoding gradient works by making the nuclei rotate with the same frequency but a different phase for each row. When the frequency encoding gradient is applied, the Larmor frequencies of the nuclei are modified to generate columns of different frequencies. This information is then combined to form the final image by using a 2D-Fourier transformation.

The delivery of these RF pulse and gradients significantly influences the resulting image. A pulse sequence refers to a defined sequence of pulses that are repeated systematically. In clinical practice, a wide range of sequences is available and can be utilized depending on the specific imaging requirements.

2.2.2. Imaging sequences

To acquire an MR image, a sequence of excitation pulses and gradient fields is applied. This sequence is characterized by three parameters: the FA, the TR, which is the time interval between two excitation pulses, and the TE, which is the time interval between the excitation pulse and signal acquisition. The resulting signal intensities in the image depend on tissue-specific properties such as relaxation times (T_1 and T_2) and proton density (PD) as described as.

$$S = PD\left(1 - e^{-TR/T_1}\right)\left(e^{-TE/T_2}\right) \qquad \textit{Equation 2-6}$$

In a typical spin-echo (SE) sequence, an additional B_1 pulse with a FA of 180° is applied after the initial 90° excitation pulse to reverse the dephasing of the nuclei caused by static field inhomogeneities and produce a signal echo at time TE. The signal intensities at the echo are governed by T_2. After a TR, the next excitation pulse is applied. During each excitation cycle, gradient fields for spatial encoding are changed to enable spatial encoding (Figure 1).

Applying consecutive 180° pulses in an MRI sequence generates multiple echoes located at a specific time interval (n • TE). This sequence, known as a multi-spin echo sequence, allows for the quantitative determination of the relaxation time (T_2) by analysing the signal intensity of the echoes [38]. To minimize the influence of T_1 relaxation, the TR should be chosen to be sufficiently high ($TR \geq 3T_1$).

Another related sequence is the fast or turbo spin-echo (TSE) sequence, which also utilizes consecutive 180° pulses [38]. The number of echoes are collected in one repetition time during TSE imaging called turbo factor (TF) or echo train length (ETL). However, in TSE, additional varying phase gradients are applied between each 180° pulse to speed up the k-space filling process. In regular SE and MSE sequences, the phase encoding gradient remains constant during each TR interval. By changing the phase encoding gradient for each 180° pulse, multiple phase-encoding steps (lines of k-space) can be acquired within a given repetition time, resulting in a significant reduction in imaging time.

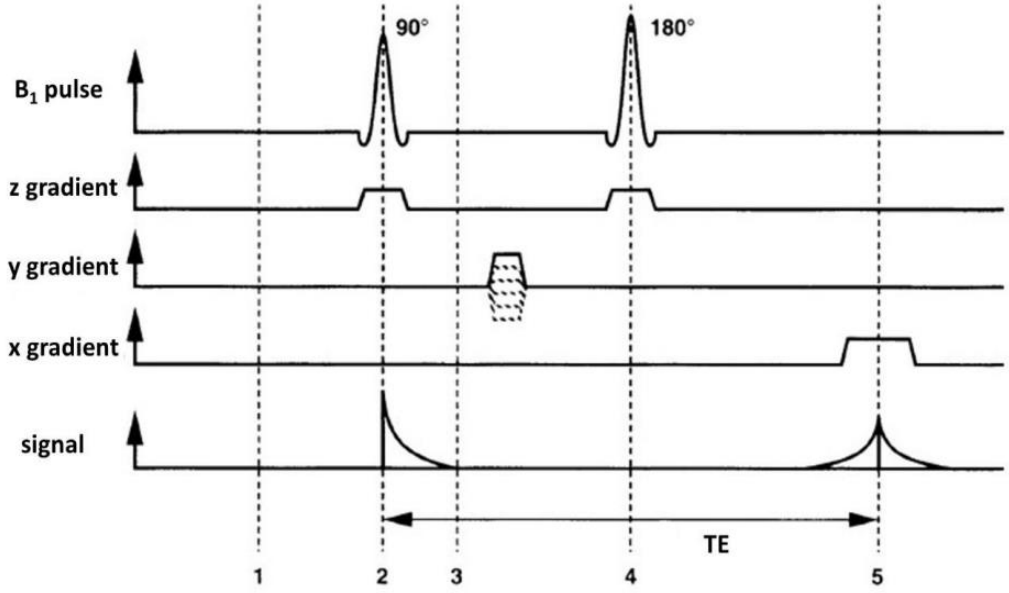

Figure 3: Sequence diagram illustrating the steps of a Spin Echo sequence. The sequence begins from the equilibrium state (1) and involves the application of a 90° excitation pulse with a simultaneous slice selective gradient to excite a 2D slice (2). Over time, the signal decays due to dephasing with the time constant T_2^* (3). At t = 1/2 TE, a 180°-pulse is applied (4), which reverse the dephasing caused by static field inhomogeneities and generating an echo at t = TE (5). The signal at the echo depends on T_2. (Adapted from [38], permission is obtained).

If the TR is sufficiently long compared to T_1 (the longitudinal relaxation time), all magnetization will fully relax before the next excitation. This leads to similar signal intensities at each excitation. In contrast, with a short TR, the magnetization in tissues with a short T_1 will fully relax, while the magnetization in tissues with a longer T_1 will not. As a result, tissues with short T_1 will have higher signal intensities at the next excitation, creating a T1-weighted (T_1W) contrast (Figure 4a). At a relatively long TE, tissues with short T_2 will have already undergone significant relaxation during signal acquisition, producing a weaker signal. On the other hand, tissues with longer T_2 will have higher signals due to less pronounced relaxation. These differences in measured signals create a T_2-weighted contrast (T_2W) (Figure 4b). However, at a very short TE, the signal intensities will be similar because none of the tissues has yet significantly relaxed. In addition, the T_1W and T_2W contrast can also be achieved by selecting small TR, TE and large TR, TE values of the sequence parameters during acquisition.

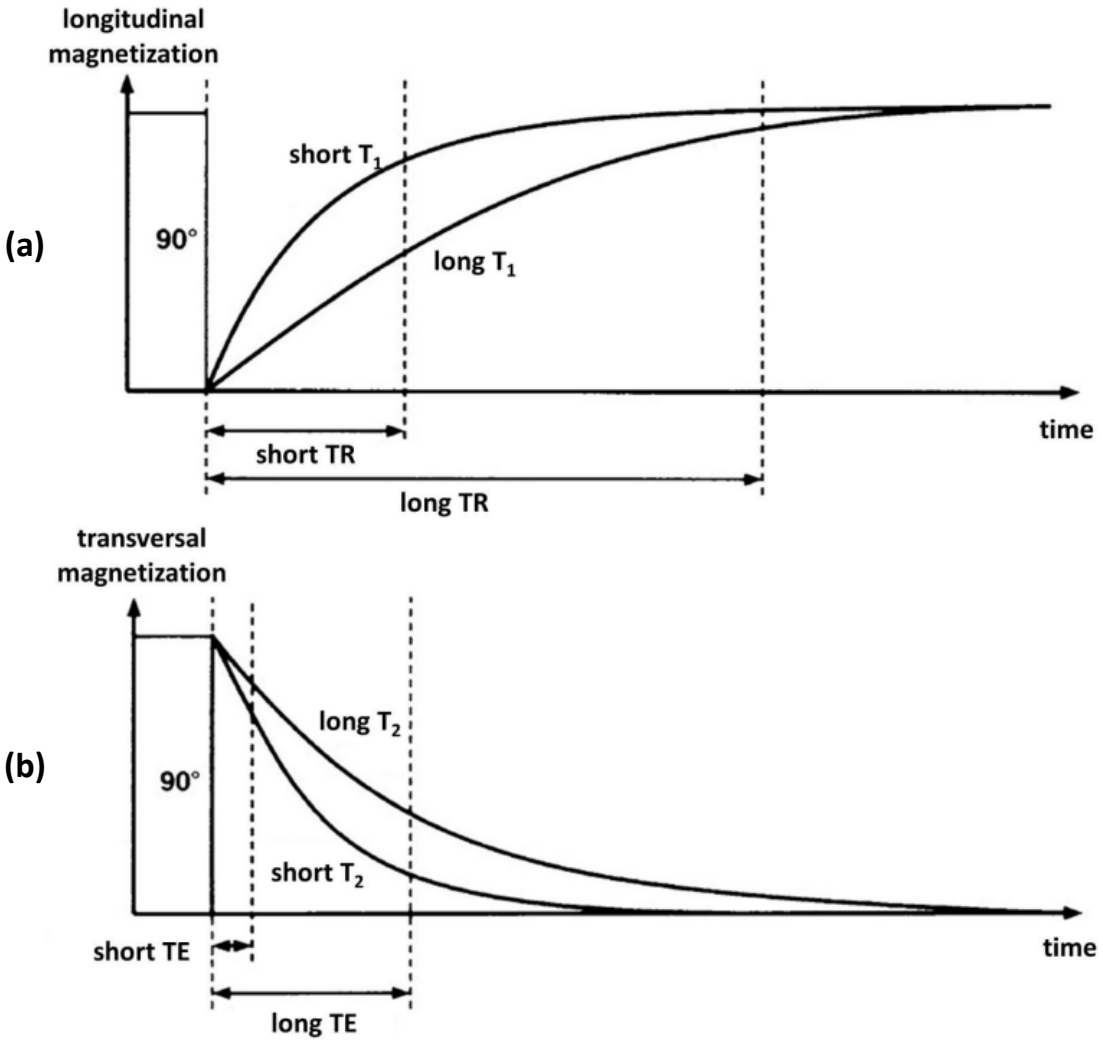

Figure 4: (a) Schematic representation of the longitudinal relaxation and the differences in signal intensities at different repetition times TR and (b) schematic representation of the transversal relaxation and the differences in signal intensities at different echo times TE Adapted from [38], permission is obtained.

2.3. Image Quality

Image quality is a crucial aspect of MRI, as it determines how well the scan can reveal the details and contrast of the tissues and organs in the human body. Several factors affect image quality, such as image contrast, noise, and presence of artifacts. Image quality assessment is a research field that seeks to evaluate and enhance the quality of MRI images using various methods and metrics. Image quality is essential for ensuring diagnostic accuracy, efficiency, and safety in MRI applications [39].

2.3.1. Image Contrast

The difference in signal intensity between different tissues or regions of interest, known as MRI contrast, depends on various factors including tissue properties and pulse sequence parameters. Proton density, representing the number of protons per unit volume, influences signal intensity, particularly in visualizing tissue anatomy and morphology like in brain or spine imaging. Additionally, the T_1 relaxation time of the tissue, which indicates the time for protons to return to the longitudinal vector quantity, and the T_2 relaxation time, representing the time for transverse magnetization to decay, play significant roles. Shorter T_1 relaxation times result in faster recovery and higher signal intensity, often used to depict differences in tissue composition and metabolism such as in fat or blood imaging. Conversely, longer T_2 relaxation times lead to slower decay and higher signal intensity, commonly utilized to highlight differences in tissue structure and pathology, such as in fluid or edema imaging.

The pulse sequence includes several parameters, such as the TR, TE, FA, BW, the matrix size (MS), the field of view (FOV), and the slice thickness (ST). These parameters determine the timing and characteristics of the RF pulses and the signal acquisition, and they affect the contrast, resolution, and acquisition time of the MRI images. For example, a short TR and a short TE will emphasize the T_1 contrast, while a long TR and a long TE will emphasize the T_2 contrast. A large FA will increase the signal intensity, but also increase the specific absorption rate (SAR) and the acquisition time. A high BW will reduce the noise and the susceptibility artifacts, but also reduce the SNR. A high MS and a small FOV will increase the spatial resolution, but also increase the scan time and the aliasing artifacts. A thin ST will increase the slice resolution, but also increase the partial volume effect and the scan time.

2.3.2. Image Noise

Image noise is the random variation in signal intensity that reduces image clarity and resolution, which originates e.g. from thermal noise in the receiver coils and the scanner electronics [40]. Thermal noise is proportional to temperature and bandwidth, and inversely proportional to coil resistance and the number of coil elements. It can be reduced by using a low bandwidth, employing a high-resistance coil, and using a multi-element coil. Other sources result from the scanner, which is the noise caused by the imperfections and instabilities of the scanner components, such as the main magnet, the gradient coils, the RF coils, and the shimming devices. The scanner noise is dependent on the scanner performance and maintenance, the pulse sequence parameters, and environmental factors. The scanner noise may be reduced by using high-quality and well-calibrated scanner components, using appropriate shimming and tuning, using gradient and RF spoiling, and by shielding of the scanner rooms.

MRI noise can be quantified by the SNR, which is the ratio of the mean signal intensity to the standard deviation of the noise intensity. The SNR can be increased by increasing the signal intensity or by decreasing the noise intensity using various MRI parameters. E.g. increasing the number of signal averages (NSA) reduces the noise intensity by the square root of the NSA, but also increases the scan time by the same factor. Increasing the voxel size increases signal intensity by the voxel size but also decreases spatial resolution by the same factor. Increased voxel sizes may generate partial volume effects or chemical shift artifacts, which

prevents resolving small tissue structures. BW is one of the MR parameters that affects image noise and its relationship to other parameters. BW defines the range of frequencies sampled during signal acquisition and is inversely proportional to the TE. BW affects SNR, susceptibility artifacts, and scan time of MRI images. It is inversely proportional to SNR, with noise intensity increasing by the square root of the BW. Therefore, low BW results in high SNR, while high bandwidth results in low SNR.

In addition, increasing the receiver gain, which is the amplification factor of the signal detected by the coil, increases the signal intensity but also increases the noise intensity by the same factor. Using parallel imaging, a technique that uses multiple coils to acquire signals simultaneously and to reconstruct the image using by a combination of the coil signals, reduces scan time and noise intensity, but it also reduces SNR by a geometry factor (g-factor), which depends on the coil configuration and acceleration factor.

2.3.3. Image Artifacts

Image distortions are image changes that do not reflect the true anatomy of the patient and may arise from various causes: Magnetic inhomogeneities of the main magnetic field in the imaged volume may lead to signal frequency shifts or broadening. This induces artifacts like chemical shift, magnetic susceptibility, zipper, and truncation. These artifacts can be reduced by appropriate shimming of the magnetic field, using a high bandwidth, a low magnetic field strength or frequency-selective fat suppression or water excitation.
The tissue heterogeneity describes the variation of the tissue properties and its magnetic environment across the imaging volume. This may cause a signal loss or distortion due to blood flow, motion, metal implants or dielectric effects. These effects can be reduced by using appropriate pulse sequence parameters, such as gradient moment nulling, flow compensation, motion correction, metal artifact reduction, and dielectric pads.

The technical errors originate from the mistakes or malfunctions of the scanner components or the operator. This can cause a corruption or a misregistration of the signal data, resulting in artifacts such as ringing, aliasing, ghosting, and cross-talk. These artifacts can be reduced by using appropriate quality control and maintenance of the scanner components, using appropriate pulse sequence parameters, such as oversampling, phase encoding, k-space filling, and slice selection, and assuring appropriate operator training and supervision

Overall, many parameters are involved in MR sequences, which directly affect the image quality. Therefore, optimizing all these MR sequence parameters is essential for image quality and achieving diagnostic accuracy in MRI examinations. To optimize the MRI image quality, several goals may be pursued: optimizing MR sequences to improve the signal intensity, the contrast, or reducing artifacts in the MRI images. This may be performed either manually or by using machine learning and artificial intelligence methods to automatically generate and optimize MRI sequences based on target contrasts or clinical application of the resulting images [7, 8, 41].

To analyze the behaviour of characteristic performance parameters of MR image acquisition automatically, such as contrast, we might need a continuous function of the target parameter

on the sequence parameters. With such a function, interpolated from previously performed measurements and explaining the physical behaviour of the respective sequence parameters, a regression model can be build. Such a regression model should reflect the measured values as accurately as possible.

2.4. Regression Problem

Regression analysis is a statistical technique used to model the relationship between a dependent variable and one or more independent variables. It aims to quantify how changes in the independent variables impact the dependent variable. Regression analysis is widely used in various fields, including economics, finance, social sciences, and machine learning.

The basic form of regression analysis is simple linear regression, which assumes a linear relationship between the dependent variable (Y) and multiple independent variables ($\vec{X}$). The simple linear regression model can be represented by the equation:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_\mathrm{p} X_\mathrm{p} + \varepsilon$$

In the regression model, Y denotes the dependent variable, while $\vec{X} = X_1, X_2, \ldots, X_\mathrm{p}$ represent the independent variables. The coefficients $\beta_0, \beta_1, \beta_2, \ldots, \beta_\mathrm{p}$ are the parameters to be estimated, representing the intercept and slopes of the linear relationship between the variables. Lastly, ε corresponds to the error term, capturing the unexplained variability in the dependent variable.

The objective of linear regression is to estimate the values of the coefficients that minimize the difference between the observed values of the dependent variable and the values predicted by the regression model. This is typically achieved using the Ordinary Least Squares (OLS) method, which minimizes the sum of squared residuals [42].

Nonlinear regression extends the concept of linear regression by allowing for more complex relationships between the dependent variable and the independent variables. It is used when the relationship between the variables cannot be adequately captured by a linear equation. The general form of a nonlinear regression model can be represented as:

$$Y = f(\beta, \vec{X}) + \varepsilon$$

Here, Y represents the dependent variable, while $f(\beta, \vec{X})$ represents a nonlinear function of the coefficients β and the independent variables $\vec{X}$. Again, ε represents the error term.

The nonlinear function can take various forms, such as exponential, logarithmic, polynomial, or trigonometric functions, among others. The choice of the specific form depends on the nature of the relationship between the variables. Unlike linear regression, estimating the coefficients in nonlinear regression models cannot be directly solved using OLS. Instead, iterative optimization algorithms are used to find the values of the coefficients that minimize the discrepancy between the observed and predicted values of the dependent variable.

Nonlinear regression is not bound by strict assumptions. However, certain considerations are commonly taken into account: Firstly, obtaining good initial parameter estimates is crucial as it aids in achieving convergence to the correct solution. Secondly, model selection plays a vital

role in nonlinear regression, as selecting an appropriate nonlinear model is essential for accurately capturing the underlying relationship between variables. Lastly, overfitting is a challenging issue in nonlinear regression, where selecting a too complex model relative to the available data can increase the chances of having an overfitting problem.

Nonlinear regression provides a flexible framework for modelling complex relationships between variables. It is commonly used in various fields, including biology, economics, engineering, and social sciences, where linear relationships may not accurately capture the data's underlying patterns. In this thesis, we focused on exploring and analysing two distinct learning-based regression models. Each of these models is described briefly in the following section.

2.4.1. Generalized Additive Model

A generalized additive model (GAM) is a statistical model that can capture complex and nonlinear relationships between a response variable and some predictor variables. A GAM is an extension of a generalized linear model (GLM), where the linear predictor is replaced by a sum of smooth functions of the predictor variables. These smooth functions can be estimated by various methods, such as splines, kernels, or trees. GAMs are useful for exploratory data analysis, regression, and classification problems, where the underlying relationship between the response and the predictors is not well understood or not easily parametrized. GAMs are also interpretable, as they allow one to visualize the effect of each predictor on the response, while accounting for the other predictors. The general form of a GAM can be written as follows:

$$g(E(Y)) = \beta_0 + f_1(x_1) + f_2(x_2) + \cdots + f_m(x_m)$$

Where Y is the response variable, g is a link function that relates the expected value of Y to the predictor variables, β_0 is a scalar value, and f_m are smooth functions of the independent variables x_m. The link function g depends on the distribution of Y, which can be any member of the exponential family, such as normal, binomial, Poisson, gamma, etc. The smooth functions f_m can have different forms and degrees of smoothness, depending on the data and the method of estimation. In GAM f_m can be calculated by a spline functions.

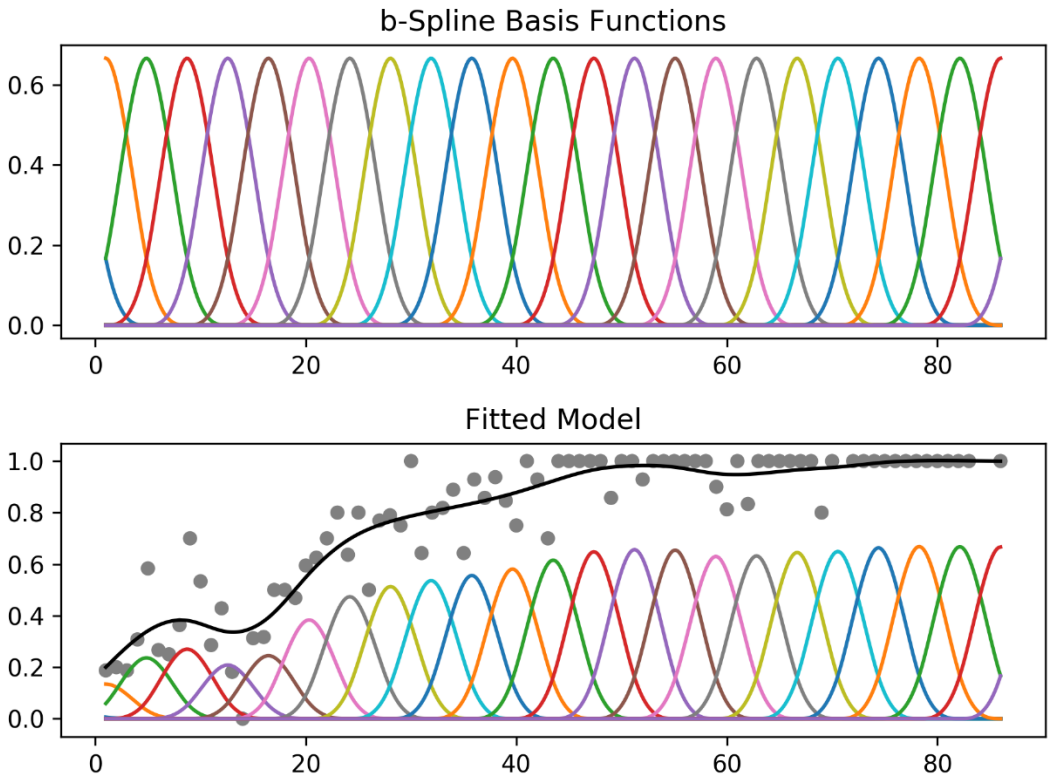

Figure 5: Fitting a GAM model as an example to a random dataset using B-spline basis functions as an example. Adapted from [43].

B splines are a type of basis function that can be used to fit GAMs. A basis function is a mathematical function that transforms an input variable into a new feature that can capture nonlinear patterns in the data. B splines are composed of polynomial segments that are joined smoothly at certain points called knots. The number and location of the knots affect the flexibility and smoothness of the B spline. B splines can also have different orders, which determine the degree of the polynomial segments and the continuity of the derivatives at the knots. One way to write the equation for a B spline basis function is:

$$b_j(x) = \sum_{i=0}^{k} c_{i,j} x^i$$

Where j is the index of the basis function, k is the order of the B spline, and $c_{i,j}$ are coefficients that depend on the knots and the order of the B spline. The coefficients can be computed using a recursive formula known as the de Boor algorithm [44]. Another way to write the equation for a B spline basis function is:

$$b_j(x) = \sum_{i=0}^{k} N_{i,k}(x) d_{i,j}$$

Where $N_{i,k}(x)$the normalized B spline basis are functions of order k, and $d_{i,j}$ are the control points that define the shape of the B spline. The normalized B spline basis functions can be computed using a recursive formula known as the Cox-de Boor formula [45]. To fit a GAM using B splines, we can use

a linear combination of B spline basis functions as the smooth term for each covariate. For example, if we have a covariate x, we can write the smooth term as:

$$s(x) = \sum_{j=1}^{J} \beta_j b_j(x)$$

j is the number of basis functions, β_j are the coefficients to be estimated, and $b_j(x)$ are the B spline basis functions.

2.4.2. Neural Networks

For many years machine learning (ML) methods have been successfully applied to solve various tasks, e.g. in robotics [46], natural language processing [47], and computer vision [48] that priorly were only achievable by humans. Yet, some capabilities, like speech and vision, are still challenging for computers, even though the human brain can perform these tasks almost effortlessly. With the advancements in computer hardware as well as the availability of huge amounts of data combined with the right algorithms, the performance of machine learning models finally reached human performance on the famous ImageNet challenge in 2015 [49].

Improved hardware does not only allow for processing more data in a shorter time but also to build more complex deep model neural net (NN) architectures. As in regular machine learning approaches, a common task is to estimate an implicit mapping between an input vector and an output vector. By adding more layers to neural network architectures in combination with non-linear activation functions, models can learn highly sophisticated mappings for underlying problems.

Despite active deep learning (DL) research in other fields such as natural language processing, speech recognition, and recommender systems, much attention is drawn to computer vision tasks, such as classifying, detecting and segmenting objects in an image or video.

Computer vision techniques have also been acknowledged in the medical context and thus been applied for solving tasks such as brain tumor or bone segmentation [50], but also personalized Alzheimer diagnosis [51], lung disease [52] and lesion classification [53]. However, medical computer vision encounters domain-specific challenges, which need to be addressed during the design and execution of machine learning models:
i) Few training samples: Finding large data sets in the medical context is difficult as it often covers rare pathological conditions, and data protection of patients is inhibitory. However, the more complex the underlying tasks and the more advanced machine learning models are, the more data is required for the training process as many parameters need to be learned.
ii) Large images with only small regions of interest: For some tasks, large 3D image volumes are available from CT or MR scans. However, tumors or other relevant areas are relatively small in comparison to the entire input volume. This increases the complexity of task solving as relevant information could be hidden by the remaining volume.

iii) Domain complexity: The same disease may vary among patients, and even from a medical point of view, it is not always fully understood which variations exist and how those diseases progress. Due to these uncertainties, data annotation can be more error-prone. In contrast to natural daily life objects, the medical domain reduces the interoperability of machine learning results even further as expert knowledge is required.

2.4.3. Artificial Neural Networks

The basic idea of the artificial neural network (ANN) started in 1943 when neurophysiologist Warren McCulloch and mathematician Walter Pitts tried to mimic the way neurons in a human brain work with a computational model-based threshold logic [54]. The idea kept getting developed over the years, but due to a lack of computation power and data, it did not gain popularity until the 2000s. A basic structure of the neural network, as shown in Figure 6*:*, contains 3 different types of layers:

i) Input Layer: This layer feeds the information to the whole model. Each neuron in this layer represents an input feature that has an effect on the final output.

ii) Hidden Layer: This layer takes in the information from the input layer and processes it to extract the relationship between the neurons of the input layer and their effect on the output. There can be more than one hidden layer, and each layer can have different numbers of neurons. These are the hyper-parameters, which are set, and they differ from task to task.

iii) Output Layer: This layer makes a decision using the information gathered from the whole network. The number of neurons in this layer depends on the task in question. In the case of a binary decision, this layer contains only one neuron.

To understand how a single neuron in a hidden or output layer works, refer to Figure 6. Here, the output of all the neurons from the previous layer is taken as input for the new layer. The task of this neuron is then divided into two parts. The first part calculates a linear combination of all the inputs and weights plus a bias term in Equation 2-7. The second part applies a nonlinear transformation with an activation function f in second equation.

$$z = \sum_{i=1}^{n} W_i X_i + b \qquad \textit{Equation 2-7}$$
$$Y = f(z)$$

The value of the linear combination could be between $-\infty$ to $+\infty$. The activation function brings this value into some range with non-linear transformation, which helps the model to learn complex non-linear relationships between the inputs and the output. This value is then the output of this neuron to the next layer. There are many options for activation functions in the literature [55] which are explained in the bellow section.

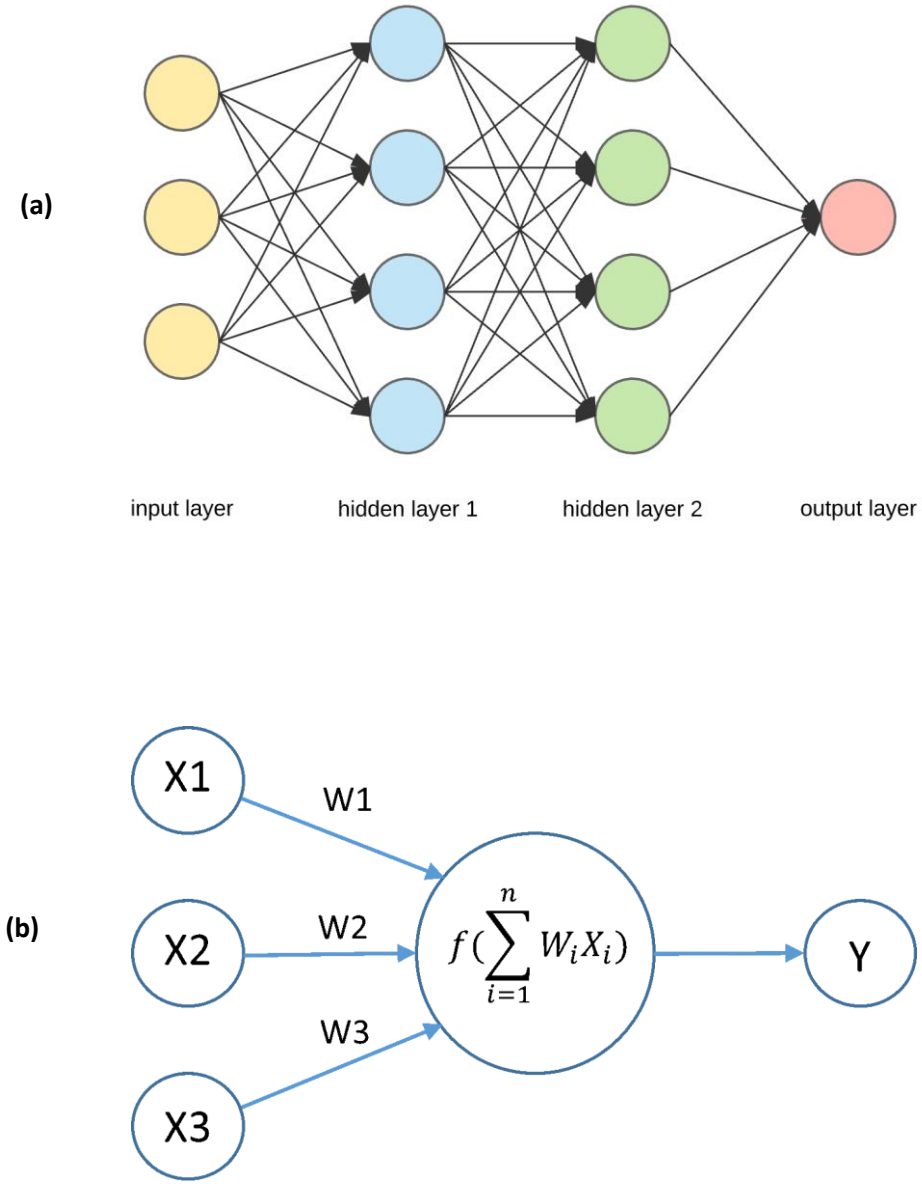

Figure 6: (a) Illustration of the fundamental structure of an artificial neural network. The network comprises three types of layers: the input layer responsible for receiving input features, the hidden layers for processing the information, and the output layer for making decisions. (b) Schematic representation of a single neuron in an artificial neural network. The neuron receives input from preceding layers, performs a linear combination operation, and applies a non-linear transformation using the activation function f. Adapted from [56].

2.4.4. Activation Layer and Activation Functions

After the neurons layer, the values could be between $-\infty$ to $+\infty$, and a non-linear activation function is applied to each feature map, which helps the model to learn the complex non-linear relationship between input and the output. There are many options for activation functions in the literature; the most common ones being:

1. The Sigmoid function is especially used for models where we have to predict a probability as an output for a binary classification. The term sigmoid means 'S-shaped', and logistic form of the sigmoid maps the interval $(-\infty, \infty)$ on to (0, 1) as seen in Figure 7. The logistic function has the form of [57]:

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

2. The tangens-hyperbolicus (tanh) activation function is also sigmoidal-shaped (s-shaped), but the range of the tanh function is from (-1,1). The output is transforms according to [58]:

$$tanh(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}}$$

3. The Rectified Linear Unit (ReLU) activation function is most commonly used, especially in the convolutional neural network (CNN). The ReLU maps from $\mathbb{R} \rightarrow [0, +\infty)$. For values larger or equal than one, ReLU is the identity function, while all negative values are mapped to zero, which decreases the ability of the model to fit or train from the data properly. ReLU are described by [58]:

$$ReLU(x) = max(0, x)$$

4. Leaky ReLUs are the attempt to solve the dying ReLU problem. Instead of the function being zero at $x < 0$, a Leaky *ReLU* function will have a small negative slop *a* (of 0.1, or so).

$$f(x) = \begin{cases} x, & if\ x > 0 \\ ax, & if\ \mathrm{x} \leq 0 \end{cases}$$

5. Softmax functions are a general form of sigmoid functions that result in a probability distribution dor different classes, adding up to 1. Mathematically, the softmax function is shown below, where x is a vector of the inputs to the output layer (if there are K classes, then there are K elements in x), and j is the index of the output units ($j = 1, 2, \ldots, K$):

$$\sigma(x) = \frac{e^{x_j}}{\sum_{k=1}^{k} e^{x_k}}$$

Figure 7: shows a graphical representation of these activation functions. The original perception first used a step function for activation. However, back-propagation requires the activation function to be differentiable. The widely used sigmoid function is a differentiable non-linear activation function that maps the output into a (0, 1) range allowing probability interpretations. However, sigmoid and similarly tanh exhibit so-called vanishing gradients, which prevents effective weight learning as weight updates might be close to zero. Figure 8: shows the sigmoid function and its derivative separately. The derivative is generally rather small with a maximum value of 0.25, and the sigmoid function reaches saturation for x ≤ −5

and x ≥ +5. Consequently, after many iterations, the weights are very slowly updated and and the training is accordingly slow. The effects of vanishing gradients become even more relevant in deep architectures as error gradients are back-propagated through many layers. To avoid this problem, the non-linear activation function ReLU the Leaky ReLU is commonly used [59]. In these cases, the vanishing gradients are replaced by a constant derivative of 1 or a, respectively. In addition, ReLU and its variations are less prone to exhibit in saturated gradients, another effect that prevents effective weight learning. Softmax is an activation function used in the final layer of, e.g., the classification network.

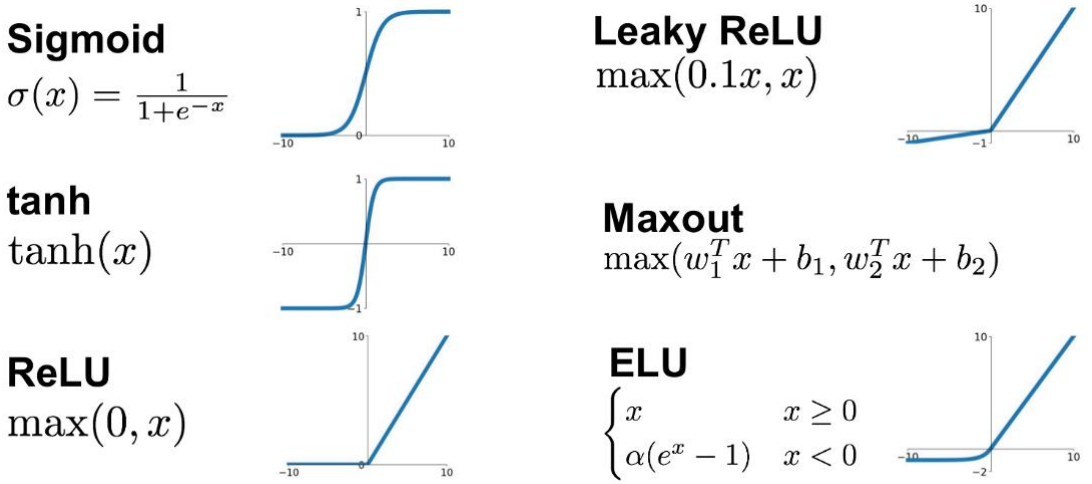

Figure 7: Comparison of different Activation Functions in Neural Networks. The figure illustrates various activation functions commonly used in neural networks. These activation functions introduce non-linearity to the network, enabling modelling of complex data. Adapter from [60].

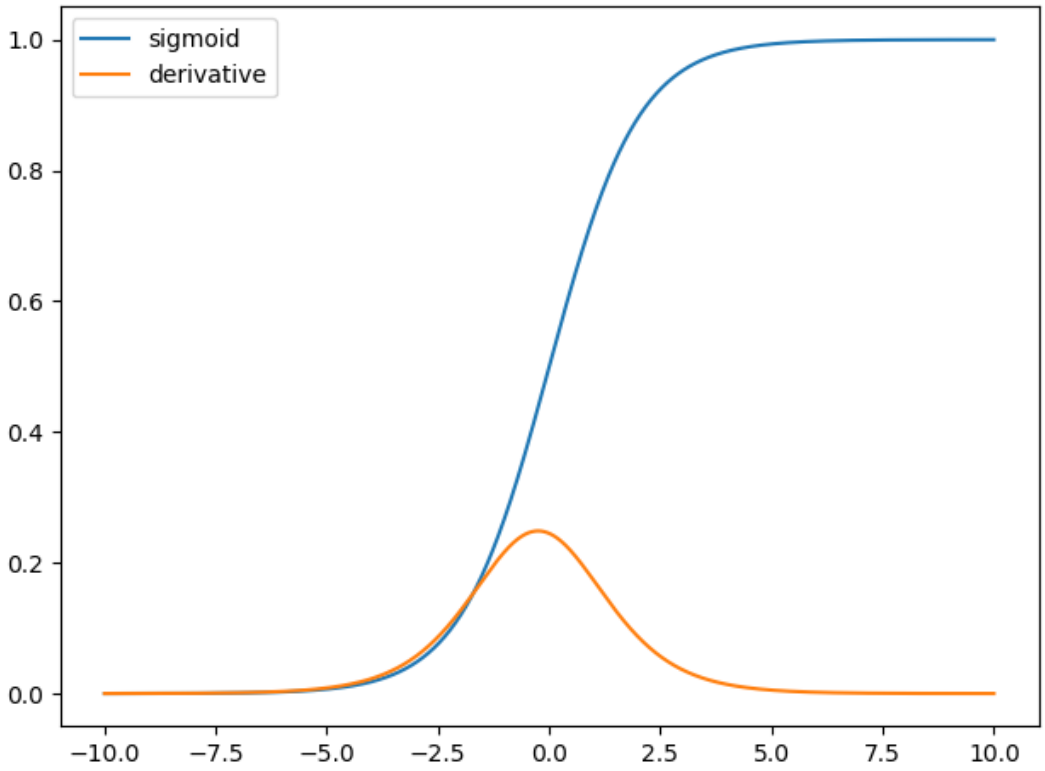

Figure 8: Sigmoid function and its derivative. The derivative has a maximum value of 0.25 and shows gradient saturation for x ≤ -5 and x ≥ +5. Adapted from [60], permission is obtained.

2.4.4.1. Loss function

The loss function plays an essential role in training neural networks (NNs) as it quantifies the error that influences the learning process. This error is determined by comparing the output of the neural network obtained during the forward pass with the ground truth. Backpropagation is then employed which calculates the gradient of the loss function with respect to all the weights in the network. The ultimate goal is to minimize the loss during the training process. The selection of an appropriate loss function depends on the specific problem that the network aims to solve. Loss functions can be customized based on the requirements of investigated task and it is also common to combine multiple loss functions into a single objective. There are several fundamental and widely used loss functions, including:

1. The mean squared error (MSE), which is also referred to as L_2 loss, calculates the average of the Euclidean distances (L_2) between the predicted value $\hat{y}_i$ and the ground truth y_i for each instance i.

$$L_{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

2. The mean absolute loss (MAE) is the absolute loss (referred to as L_1 loss) and calculates the mean absolute value between the predicted value $\hat{y}_i$ and the ground truth y_i for each instance i.

$$L_{MAE} = \frac{1}{n}\sum_{i=1}^{n} \left| y_i - \hat{y}_i \right|$$

2.4.4.2. Neural Network Optimization

Neural networks (NN) are trained using back-propagation, which is the implementation of a gradient descent algorithm based on a selected loss function. To compute the loss function, the input data is first propagated through the network and then compared to the ground truth. Let y be the ground truth, and $\hat{y}$ the result after the calculation. The loss function is defined as:

$$f(\theta) = L(y, \hat{y})$$

The Gradient Descent (GD) method updates the parameters, $\theta \in \mathbb{R}^d$, in the opposite direction of the gradient of loss function $\nabla_\theta f(\theta)$. Here, θ include all the weights and biases as described in Equation 2-3. The size of the step in the negative direction is called the learning rate (η). Figure 9: shows the graphical representation of gradient descent at work. Depending on our decision for speed vs. accuracy, there are three different versions of gradient descent method, which differ by the amount of data they require to make an update at the iteration number k:

1. Batch Gradient Descent method: Computes the gradient and makes the update to parameters after using the whole training Set:

$$\theta^{k+1} = \theta^k - \eta * \nabla_\theta f(\theta^k)$$

Since the whole dataset is used to make just a single update, Batch Gradient Descent is very slow and memory consuming but the number of steps might be very small.

2. Stochastic Gradient Descent (SGD) method updates the parameters on each dataset (x_i, y_i):

$$\theta^{k+1} = \theta^k - \eta * \nabla_\theta f(\theta^k; x_i; y_i)$$

Each update of SGD method is much faster than that for the batch gradient descent method, however, it needs more steps to converge.

3. Mini-batch Gradient Descent method updates the parameters after processing a mini-batch of n on training samples $x_{i:i+n}; y_{i:i+n}$.

$$\theta^{k+1} = \theta^k - \eta * \nabla_\theta f(\theta^k; x_{i:i+n}; y_{i:i+n})$$

With this method, the loss per iteration is minimized more than that in the stochastic gradient descent method, and each step is faster than that for the batch gradient descent method. This is the mostly employed method.

One of the problems with the gradient descent method is that it treats all the parameters in the same way, regardless of their magnitude, and uses the same learning rate throughout the learning process. In recent times, there are more powerful optimizers, which solve these issues and are used in training of neural networks. One of them is Adam (adaptive moment estimation). Adam uses an individual learning rate for each network weight and adapts these weights individually based on gradient back-propagation. An exponential moving average of the gradient $m(t)$ and the exponential moving average of past squared gradient $v(t)$ (variance) are calculated. Exponential moving averages of the gradient g are calculated as follows:

$$m(t) = \beta_1 m(t-1) + (1-\beta_1) g(t)$$

$$v(t) = \beta_2 v(t-1) + (1-\beta_2) g^2(t)$$

Here, t is a iteration number. Two additional parameters β_1 and β_2 are used to decay moving averages over time, recommended to be chosen close to 1. As $m(t)$ and $v(t)$ are initialized as vectors of 0's, that is why moment estimates tend to show a bias towards zero, especially during the initial time steps. Hence, this bias is corrected:

$$\widehat{m}(t) = \frac{m(t)}{1-\beta_1^t}$$

$$\hat{v}(t) = \frac{v(t)}{1-\beta_2^t}$$

Here, $\widehat{m}(t)$ and $\hat{v}(t)$ are the updated values, t is a exponent in case of β_1^t and β_2^t. These values are used to update the parameters by using the expression:

$$\theta^{k+1} = \theta^k - \frac{\eta}{\sqrt{(\hat{v}(t)} + \epsilon)} \widehat{m}(t)$$

Where, η is a learning rate using in the optimization method.

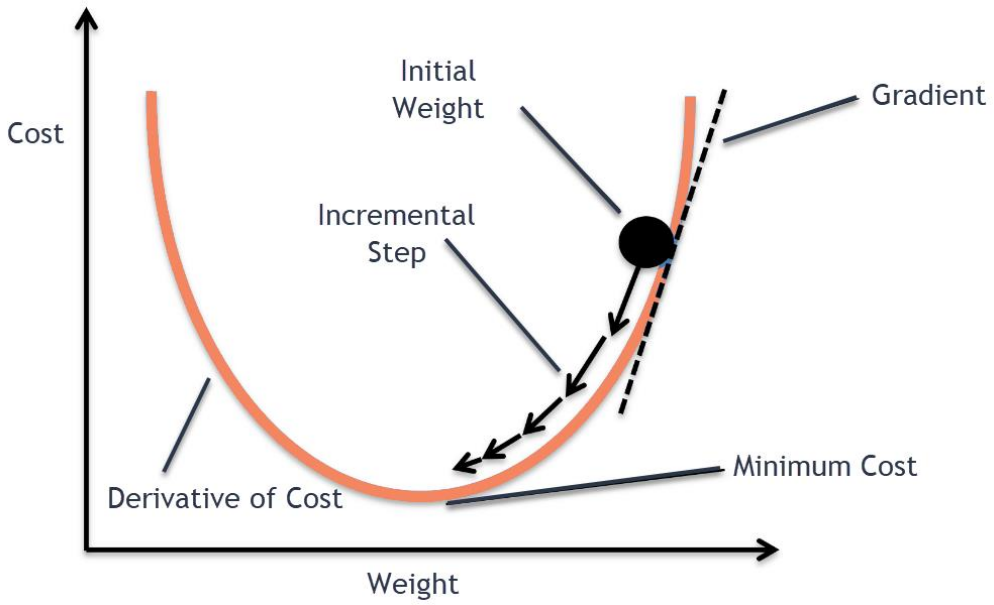

Figure 9: Visualization of Gradient Descent Optimization method. The figure shows the iterative process of gradient descent, in which the arrows represent the direction and magnitude of the gradient at each iteration, indicating the path taken towards the optimal solution. Adapted from [61].

2.5. Optimization

In unconstrained optimization, the goal is to find the optimal solution for a given objective function without any constraints. The objective function is typically defined over a continuous domain and represents a quantity to be maximized or minimized. Let's denote the objective function as f(x), where x is the input or decision variable. To find the optimal solution, we seek the value of x that either maximizes or minimizes the objective function. This can be represented as:

$$Minimize: f(x)$$

To determine the optimal solution, we often use gradient-based methods, such as gradient descent or Newton's method. These methods rely on the derivative or gradient of the objective function to guide the search for the minimum or maximum.

For instance, in gradient descent, we iteratively update the value of x based on the negative gradient of the objective function. The update rule can be expressed as:

$$x_{new} = x_{old} - learning\_rate * gradient(f(x_{old}))$$

Here, learning_rate represents a small positive value that controls the step size of each iteration. The goal is to update x in the direction that minimize the value of the objective

function until convergence. A detailed explanation of optimization can be found in the literature [62-65].

Constrained Optimization Problems:

Constrained optimization deals with finding the optimal solution while considering a set of constraints that restrict the search space. The constraints can be inequalities (<=, >=) or equalities (=) that must be satisfied when searching for the optimal solution. We can express the constrained optimization problem as follows:

$$Minimize: f(x)$$

$$Subject\ to:$$

$$g(x) <= 0\ (inequality\ constraints)$$

$$h(x) = 0\ (equality\ constraints)$$

Here, g(x) represents the vector of inequality constraints, and h(x) represents the vector of equality constraints. To solve constrained optimization problems, various techniques are available, such as the Lagrange multipliers method, Karush-Kuhn-Tucker (KKT) conditions, and nonlinear programming algorithms. The equation for the Lagrange multipliers method is [66]:

$$\nabla f(\mathbf{x}) = \lambda \nabla g(\mathbf{x})$$

$$\nabla f(\mathbf{x}) + \sum_{i=1}^{m} \lambda_i \nabla g_i(\mathbf{x}) + \sum_{j=1}^{n} \mu_j \nabla h_j(\mathbf{x}) = 0$$
$$g_i(\mathbf{x}) \leq 0, i = 1, \dots, m$$
$$h_j(\mathbf{x}) = 0, j = 1, \dots, n$$
$$\lambda_i \geq 0, i = 1, \dots, m$$
$$\lambda_i g_i(\mathbf{x}) = 0, i = 1, \dots, m$$

Where f(x) is the objective function, $g_i(x) \leq 0$ are the inequality constraints, $h_j(x) = 0$ are the equality constraints, ∇ denotes the gradient operator, and λ_i and μ_j are the KKT multiplier.

These methods aim to find the optimal solution by considering both the objective function and the constraints. By combining the objective function and the constraints, constrained optimization seeks to identify the values of x that simultaneously minimize the objective while satisfying all constraints.

Overall, optimization of unconstrained and constrained problems is a vast and rich field, encompassing a range of algorithms and techniques. These approaches allow to find the optimal solutions in various domains and make informed decisions based on mathematical modeling and problem-solving.

2.5.1. Covariance matrix adaptation evolution strategy

Covariance Matrix Adaptation Evolution Strategy (CMA-ES) is a popular and powerful stochastic derivative-free global optimization method. It belongs to the family of Evolutionary Algorithms (EAs) and is designed to solve continuous optimization problems, where the objective function is expensive to evaluate or has no explicit gradient information. CMA-ES, an optimization technique originally introduced by Hansen *et al.* [67], and is a method that aims to find the global optimum in a given solution space. A population of potential solutions is iteratively sampled from this distribution and evaluated using a so-called "black-box function", which is a function that can be used without knowing its internal implementation or logic. The pairs of solutions and their evaluations form a data set that CMA-ES uses to update its search distribution, specifically the mean and covariance matrix. A more detailed description on CMA-ES can be found in [68-70].

To elaborate, let's consider a fitness function $f: \mathbb{R}^n \mapsto \mathbb{R}$ parameterized by $\theta \in \mathbb{R}^n$, denoted $f(\theta)$. The goal is to find an optimal parameter θ^* that minimizes $f(\theta)$. In CMA-ES, it is common to represent the solution space with a multivariate normal distribution, with $\theta \sim N(\theta; m, C)$. Here, m is the n-dimensional mean vector and C is the $n \times n$ covariance matrix.

At each iteration k, CMA-ES generates the k^{th} population of λ offspring by sampling from the k^{th} distribution given by $\theta_i \sim N(\theta; m_k, C_k)$ for $i = 1, \cdots, \lambda$. Here, m_k and C_k correspond to the mean vector and covariance matrix at iteration k after k updates respectively. The offspring are then sorted in ascending order based on their evaluations $f(\theta_i)$. Only the top μ $(< \lambda)$ candidates are selected for updating m_k and C_k. Another parameter, the global step size $\sigma \in \mathbb{R}$, is used to control the convergence rate of the covariance matrix update. σ is defined as the global standard deviation. Thus, the complete set of parameters in CMA-ES is $\{m, C, \sigma\}$.

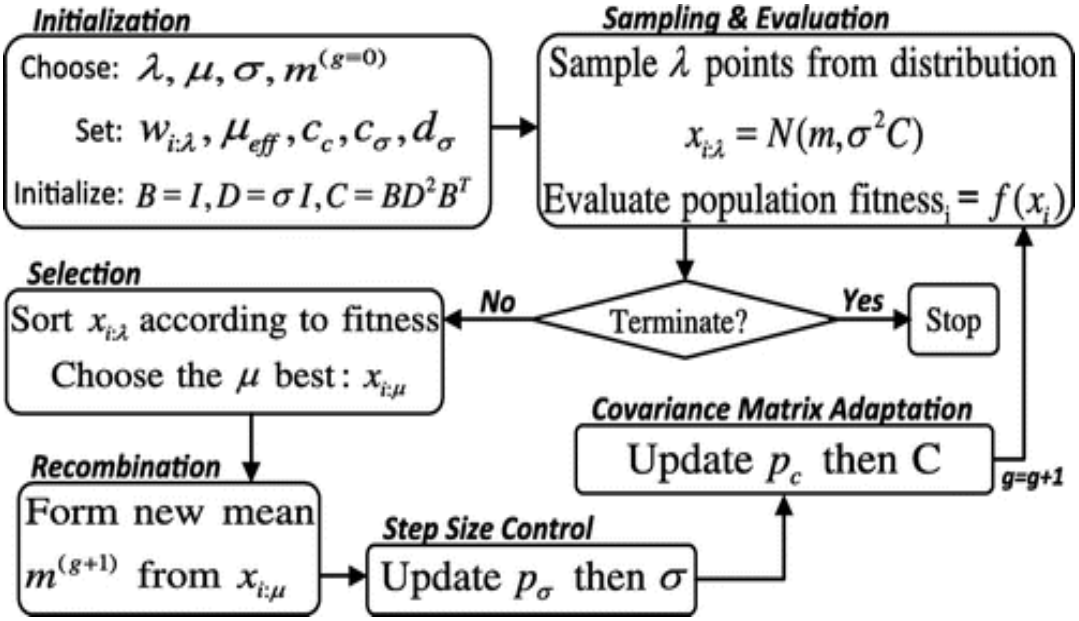

Figure 10: CMA-ES flowchart. In each iteration(generation) step, a weighted combination of the μ best out of λ new candidate solutions is used to update the distribution parameters p σ, p μ, C. Here, N represents a normal distribution and f is the fitness function. Adapter from [71].

2.5.2. Genetic algorithm

Genetic algorithms (GA) have emerged as a powerful optimization technique inspired by natural selection and genetic principles (see Figure 11:). GA is widely used in various fields, including engineering, computer science, and economics, to solve complex optimization problems. The key characteristic of GAs lies in their ability to mimic the process of evolution through iterative generations of candidate solutions. This section explains the dynamics of genetic algorithm optimization, shedding light on the underlying mechanisms and factors influencing its performance. For more detailed explanation, the reader is refer to the literature [72, 73]

The representation of the problem space and encoding of the candidate solutions play a crucial role in genetic algorithm optimization and significantly impacts the genetic algorithm's efficiency and effectiveness. Here, we explore different representation schemes and encoding techniques commonly used in genetic algorithm optimization.

Representation Schemes:

1. Binary representation is one of the most widely used schemes in genetic algorithms. It represents each candidate solution as a string of binary digits (0 and 1). This scheme is suitable for problems where the solution can be represented as a binary sequence, such as combinatorial optimization problems or Boolean function optimization.

2. Real-valued representation is used when the problem requires variables with continuous values. Each candidate solution is represented as a vector of real numbers in this scheme. It

allows for directly mapping the solution space, making it suitable for optimization problems with continuous or numerical variables.

3. Permutation representation is employed when the problem involves finding an optimal ordering or arrangement of elements. It represents the candidate solution as a permutation of the elements being optimized. Permutation representation is commonly used in problems such as traveling salesman or job scheduling.

Encoding Techniques:

1. Fixed-length encoding is used when the length of the chromosome representing a candidate solution remains constant throughout the optimization process. Each chromosome has a fixed number of bits or elements representing different variables or characteristics of the solution. Fixed-length encoding simplifies the implementation of genetic operators but may be less efficient for problems with varying lengths.

2. Variable-length encoding is employed when the length of the chromosome can change during the optimization process. It allows for the representation of solutions with varying numbers of variables or features. Variable-length encoding offers flexibility but requires additional mechanisms to handle varying chromosome lengths, such as special termination conditions or dynamic memory allocation.

3. Direct encoding represents the problem-specific features directly in the chromosome without any transformation. For example, in a scheduling problem, the chromosome may directly encode the time slots assigned to different tasks. Direct encoding simplifies the representation but may lead to a large search space or lack of generality.

4. Indirect encoding represents the candidate solution using a set of rules or transformations that generate the desired solution. It employs a higher-level representation that maps to the actual solution. Indirect encoding allows for compact representation and can exploit problem-specific knowledge, but it adds complexity to the decoding process.

The choice of representation and encoding depends on the problem's nature, the variables' characteristics, and the optimization task's requirements. It is essential to select a representation scheme and encoding technique that effectively captures the problem structure and facilitates efficient exploration of the search space. Careful consideration should be given to the choice of representation and encoding to ensure compatibility with the problem domain and maximize the performance of genetic algorithm optimization.

Population Initialization:

The initial population is the foundation for the evolutionary process in genetic algorithms. This section explores different techniques for population initialization, such as random initialization, heuristic-based initialization, and biased initialization. The influence of population size, diversity, and structure on the algorithm’s exploration and exploitation abilities will be examined, highlighting the trade-offs involved.

Genetic Operators:

The genetic operators, namely selection, crossover, and mutation, are exploitation characteristics of genetic algorithms. This section delves into various selection strategies, including tournament selection, roulette wheel selection, and rank-based selection. It investigates different crossover techniques, such as single-point crossover, multi-point crossover, and uniform crossover. The effects of different mutation operators, mutation rates, and adaptive mutation schemes on the algorithm's ability to escape local optima and maintain diversity will be discussed.

Selection:

Selection is a vital genetic operator in the genetic algorithm optimization process, responsible for choosing individuals from the population to serve as parents for the next generation. The selection process is based on proportionate fitness selection, where individuals with higher fitness are more likely to be selected. There are several common selection strategies:

1. Tournament Selection: In this strategy, individuals are randomly selected from the population, and a tournament is held among them. The fittest individual from the tournament is chosen as a parent. The tournament size determines the selection pressure, with larger tournament sizes favoring individuals with better fitness.

2. Roulette Wheel Selection also known as stochastic selection or proportionate fitness selection, this strategy assigns a probability to each individual in the population based on their fitness. A roulette wheel is then spun, and individuals are selected based on the portion of the wheel they occupy. Individuals with higher fitness have larger portions and are more likely to be selected.

3. Rank-Based Selection assigns ranks to individuals based on their fitness, with higher ranks assigned to fitter individuals. Selection probabilities are derived from these ranks, favoring individuals with higher ranks. This technique reduces the influence of extreme fitness values and encourages diversity.

Crossover:

Crossover is the genetic operator responsible for recombining genetic information from two parent individuals to create offspring. It mimics the genetic recombination that occurs during sexual reproduction. Common crossover techniques include:

1. Single-Point Crossover is a technique, in which a single random point is selected along the chromosome of the parent individuals. The genetic material beyond that point is swapped between the parents, producing two offspring.

2. Multi-Point Crossover: Similar to single-point crossover, multi-point crossover involves selecting multiple random points along the chromosome. The genetic material between these

points is exchanged between the parents, generating offspring with mixed genetic information.

3. Uniform Crossover, each gene in the chromosome is independently selected from either parent with a predefined probability. This allows for a more diverse recombination of genetic material, leading to increased exploration of the search space.

Mutation:

Mutation is a genetic operator that introduces random changes in individual chromosomes, helping to maintain diversity and explore new regions of the search space. It prevents premature convergence to local optima. Common mutation techniques include:

1. Bit Flip Mutation is commonly used for binary-encoded chromosomes. It selects random bits in the chromosome and flips their values, introducing small changes to the genetic material.

2. Gaussian Mutation is suitable for real-valued chromosomes. It adds a small random value drawn from a Gaussian distribution to each gene in the chromosome, causing a slight perturbation in the values.

3. Swap Mutation is often used for permutation-encoded chromosomes. It selects two random positions in the chromosome and swaps the values at those positions, introducing changes in the ordering of genes.

It is important to note that the mutation rate determines the probability of applying mutation to each individual gene or chromosome. A higher mutation rate promotes exploration but may hinder convergence, while a lower mutation rate may lead to premature convergence.

The genetic operators selection, crossover, and mutation work in combination to drive the evolutionary process in genetic algorithm optimization. They allow for the exploration of the search space, exploitation of promising solutions, and maintenance of genetic diversity, ultimately converging towards high-quality solutions to complex optimization problems.

-

Convergence Analysis:

Understanding the convergence properties of genetic algorithms is crucial for assessing their performance and optimizing their behaviour. This section presents theoretical and empirical convergence analyses, exploring the impact of population size, selection pressure, crossover rate, mutation rate, and termination criteria on the algorithm's convergence speed and solution quality. Moreover, the role of fitness landscape characteristics, such as ruggedness, multimodality, and separability, in determining convergence behaviour will be investigated.

Parameter Tuning and Adaptation:

The performance of genetic algorithms heavily relies on the proper configuration of their parameters. This section investigates various techniques for parameter tuning and adaptation, including classical methods like grid search and evolutionary algorithms, as well as more advanced approaches such as self-adaptive algorithms and machine learning-based methods. The challenges, benefits, and limitations of different parameter tuning strategies will be discussed, along with their impact on the algorithm's performance and robustness.

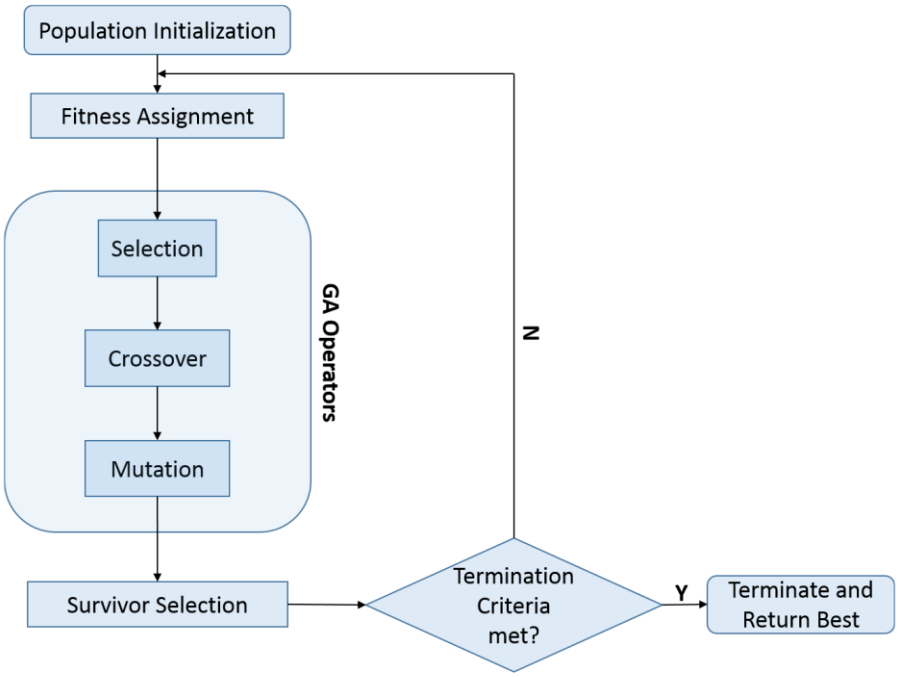

Figure 11: Genetic Algorithm Pipeline. The figure illustrates the pipeline of a genetic algorithm, including the steps of initialization, fitness assignment, selection, crossover, mutation, and selection. Adapted from [74].

2.5.3. Multi Objective Optimization

Multi-objective optimization (MOO) also knows as Pareto optimization is a branch of mathematical optimization that deals with problems involving more than one objective function to be optimized simultaneously. Such problems arise in many fields of science, engineering [75], economics, and logistics, where optimal decisions need to be made in the presence of trade-offs between two or more conflicting objectives. MOO problems are challenging because they do not have a single optimal solution, but rather a set of solutions that are equally good in terms of the objective functions. These solutions are called Pareto optimal solutions. The primary challenge in MOO is the presence of conflicting objectives, making it impossible to improve one objective without degrading another. This leads to the concept of Pareto optimality, where a solution is considered Pareto optimal if none of the

objective functions can be improved without worsening at least one other objective function. The goal in MOO is to find a representative set of such Pareto optimal solutions.

Several solution methods have been developed of MOO, including interactive methods [76], a priori methods [77], evolutionary algorithms [78]. One way to solve MOO problems is to use scalarization methods. In which convert a MOO problem into a single-objective optimization problem by combining the multiple objectives into a scalar function. The scalar function is usually a weighted sum of the objectives, but other forms are possible. The equation for the weighted sum scalarization method or classical method is [79]:

$$f(x) = \sum_{i=1}^{n} w_i f_i(x)$$

where $f(x)$ is the scalar function, $f_i(x)$ are the objective functions, w_i are the weights, and n is the number of objectives. The weights reflect the relative importance of each objective and must be non-negative and sum to one.

Scalarization methods are simple and easy to implement, but they have some limitations. For example, they might not be able to find all the Pareto optimal solutions, especially if the Pareto optimal set is non-convex or discontinuous. They also require prior knowledge of the objective functions and their ranges, which might not be available in some cases. Therefore, other methods, such as evolutionary algorithms, have been developed to overcome these challenges and find more diverse and robust solutions for MOO problems [80].

Non-dominated Sorting Genetic Algorithm –III:

Non-dominated Sorting Genetic Algorithm-III (NSGA-III) [81] is an evolutionary algorithm specifically designed to solve multi-objective optimization problems. It is an improved version of the well-known NSGA-II [82] algorithm and is known for its ability to efficiently handle multi-objective optimization problems. The algorithm works by maintaining a diverse set of solutions that are not dominated by each other. It uses a three-level approach to maintain diversity and convergence, making it suitable for many-objective optimization problems. NSGA-III improves upon NSGA-II by introducing the following features:

Reference points: NSGA-III uses a set of predefined reference points to guide the search towards a diverse and well-distributed Pareto front. The reference points are uniformly distributed in the objective space and represent the ideal trade-offs among the objectives. NSGA-III tries to find one solution for each reference point, or as close as possible.

Reference line: NSGA-III uses a reference line to measure the distance between a solution and a reference point. The reference line is the line that passes through the origin and the reference point. NSGA-III prefers solutions that have a smaller perpendicular distance to the reference line, as they are closer to the ideal trade-off.

Selection scheme: NSGA-III uses a two-step selection scheme to maintain diversity and convergence. First, it applies the non-dominated sorting as in NSGA-II to rank the solutions

according to their dominance level. Second, it selects the solutions from the last front that are needed to fill the population. It does so by assigning each solution to the nearest reference point and then choosing the solutions that belong to the least populated reference points. If there is more than one solution for a reference point, it selects the one with the smallest perpendicular distance to the reference line.

Elitism: NSGA-III maintains an external archive of the best solutions found so far, called the elite set. The elite set is updated at each generation by combining the current population and the offspring, and applying the non-dominated sorting procedure. The elite set ensures that the algorithm does not lose any good solution and preserves the diversity of the Pareto optimal front. A detailed description of this method and related methods can be found elsewhere [81, 82].

2.6. Segmentation

Segmentation in the context of image processing refers to the process of dividing an image into meaningful and distinct regions or objects. It is a fundamental technique used to extract specific regions of interest or separate different objects in an image. Segmentation plays a crucial role in various applications such as object recognition, image analysis, computer vision, and medical imaging.

The main purpose of segmentation is to simplify and analyze images by partitioning them into semantically coherent regions. This enables subsequent analysis and understanding of the image content, such as object detection, tracking, and feature extraction.

There are several popular methods for image segmentation, and each has its strengths and limitations. Some common segmentation methods include thresholding [83], edge-based methods [84], clustering algorithms [85], and region-based methods [86]. Each method employs different techniques and algorithms to achieve the desired segmentation. One of the widely used segmentation methods is the Region Growth method. It is a region-based approach that starts with a seed point or region and gradually expands the region by including neighboring pixels based on specific criteria. The region growth process continues iteratively until a stopping criterion is met.

The region growth method typically involves the following steps: Seed Selection, Region, Expansion, Similarity measure and finally the stopping criterion. During the seed Selection an initial seed point or region is chosen either manually or automatically based on specific characteristics or user input. This seed point or region is expanded during the region expansion by iteratively examining neighboring pixels and determining if they satisfy certain criteria, such as intensity similarity, color similarity, texture similarity, or gradient similarity. Thereafter, a similarity measure is used to assess the similarity between the candidate pixel and the current region. If the similarity criterion is met, the pixel is added to the region, and the process continues. The region growth process continues until a stopping criterion is fulfilled, which could be a predefined size, a specific image feature, or reaching the boundary of the image. This method is effective in segmenting regions with uniform properties, such as homogeneous regions or objects with consistent characteristics. However, it may face

challenges in handling complex scenes with varying textures, illumination changes, or overlapping objects.

2.7. Experimental setup

This section presents two different optimization workflows developed in this thesis: I) Regression-based optimization and II) On-the-run optimization. In the regression-based optimization workflow, two different regression models were evaluated: (a) DL and (b) GAM. For both optimization workflows, two different EA optimizations were also evaluated: (i) CMA-ES and (ii) GA. As a clinical application in radiotherapy, this study demonstrated how MR sequences can be optimized with respect to two different goals: (1) achieving the same contrast as in a target image and (2) maximizing the contrast between different tissue types. *Figure 12* gives an overview on the different investigation in this thesis.

Pursuing the first goal is reasonable, e.g., if diagnostic MR images are to be included into treatment planning for a 0.35 T MR-linac, which can exhibit a different image contrast. Using the images from the diagnostic MRI and optimizing the SPS to obtain a similar contrast as the MR-linac may facilitate image processing steps like registration. Other scenarios, where the target image is acquired at the same device with different sequences or SPS are also conceivable. The second use case maximizes the contrast between specified adjacent tissues to improve the conditions for automated segmentation of tumors and/or organs at risk.

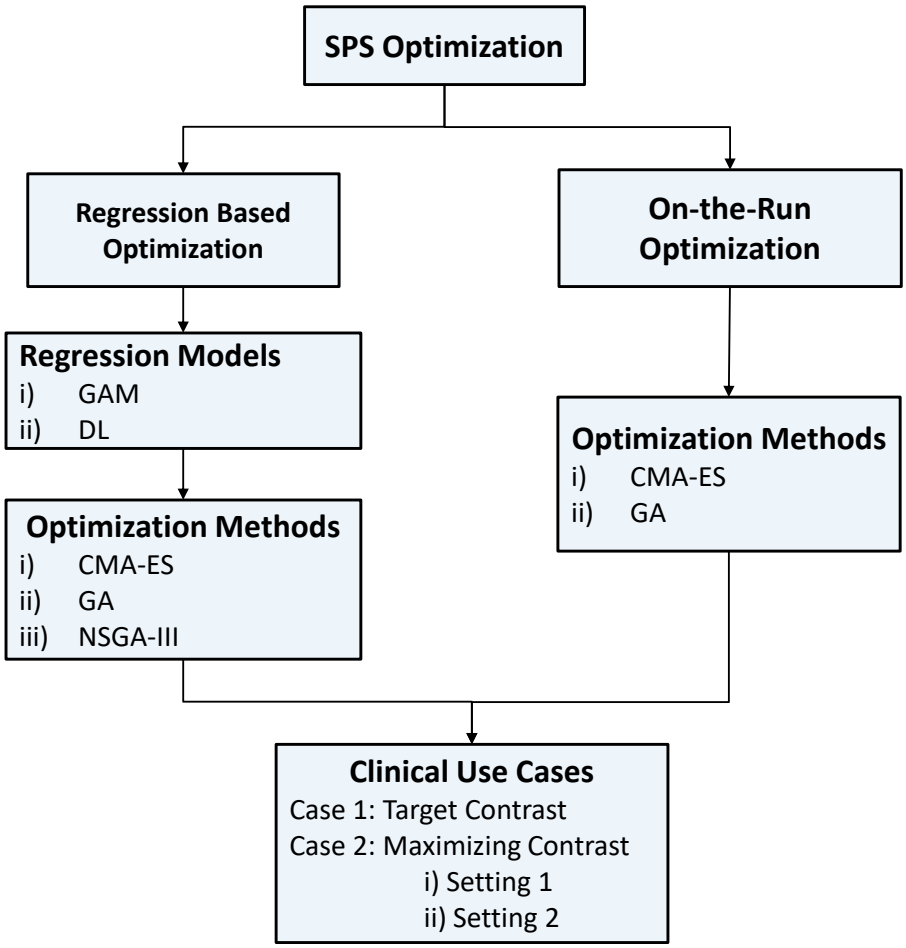

Figure 12: Provides an overview of the various investigations conducted in this thesis.

2.7.1. Phantom materials

Measurements were performed using a cylindrical water phantom that was equipped with 7 in-house fabricated substitutes with different contrasts. A PMMA-ring was used to hold the substitutes in place. The substitutes and they were made with different concentrations of agarose (Agar) (AgaroseHEEO Ultra-Quality, Carl RothGmbH&Co. KG, Karlsruhe, Germany), in-house produced nickel-diethylenetriaminepentaacetic acid (Ni-DTPA), and potassium chloride (KCl) (≥99, 5%, Carl RothGmbH&Co. KG, Karlsruhe, Germany). The contrast of the Ni-DTPA doped agarose gel was adjusted to achieve different T_1 and T_2 relaxation times in MRI by varying the amounts of Ni-DTPA and agarose. Ni-DTPA mainly reduced T_1 relaxation time, while agarose mainly decreased T_2 relaxation time. Additionally, KCL was added to change the CT value. The substitutes had different amounts of Agar, Ni, and KCL to create 7 different contrasts (see Table 1). Plastic conical centrifuge tubes (50 ml, diameter: 28 mm, FalconTM, Thermo Fisher Scientific Inc., Waltham, USA) served as containers for the substitutes. *Figure 13* shows a phantom alongside its corresponding scanned image.

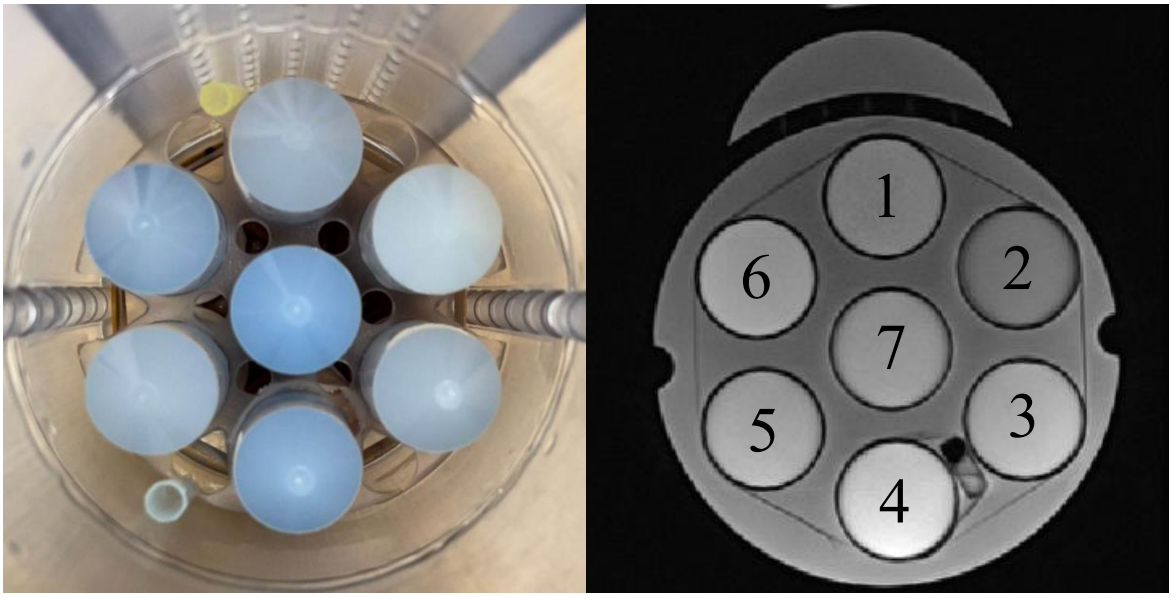

Figure 13: Left) Shows the real phantom which is used for all the investigation, and (Right) displays the image acquired from MR SOLA scanner using random parameter combinations.

Table 1: Contrast materials providing specific T1 and T2 values at a 1.5 T MRI and 0.35 T MR-Linac. Values as reported in [87].

	Container	**#1**	**#2**	**#3**	**#4**	**#5**	**#6**	**#7**
1.5 T	T1	420±1	523±1	984±3	1097±4	629±1	876±4	882±2
	T2	67±1	93±2	110±2	46±1	57±1	296±3	107±2
0.35 T	T1	575±4	733±4	1108±4	1155±4	707±4	1106±5	1051±5
	T2	75±5	100±5	119±4	45±1	61±3	311±13	110±5

2.7.2. Regression based SPS Optimization

In this section, a regression-based optimization workflow was implemented. Initially, a large dataset was acquired by varying combinations of four sequence parameters to construct a regression model. This model aims to predict the signal value for any combination of SPS. These regression models served as the signal prediction functions. The optimization methods were then integrated with the regression models. Overall, the workflow continuously updated the SPS and predicted the signals for each substitution with the new SPS based on the regression model. In this iterative process, the SPS was optimized for two clinical use cases.

2.7.2.1. Data Acquisition

All measurements were performed on a 1.5 T MAGNETOM Aera MR scanner (Siemens Healthineers, Erlangen, Germany) using the 20-channel head coil. Measurements were performed on a 1.5 T MRI Aera scanner. The SPS was optimized for a 2D TSE sequence with a fixed BW of 186 Hz/pixel, acquired resolution of 0.4 x 0.4 mm^2, slice thickness of 5 mm and echo spacing of 11 ms. TE, TR, TF and refocusing FA, as main contributors to contrast, were varied in a specific range (TE: 12 ms – 114 ms, TR: 500 ms – 2300 ms, TF: 10 – 40 and FA: 140° – 180°) to generate images with 1114 different SPS combinations.

In our experiments, we used an image from a 0.35 T MR-Linac (MRIdian Linac, Viewray Inc., Oakwood, USA) as a target image specially for use case 1: achieving the same contrast as a target image, with the following parameters: TR = 2000 ms, TE = 35 ms, TF = 15, BW = 202 Hz/pixel, FA = 180° and resolution = 0.78 x 0.78 mm^2.

2.7.2.2. Regression problem

After measuring the parameter space, the next step was to develop a model that can predict the signal values based on the measured data. The model should match the measured values as closely as possible. In this study, two different types of regression models were explored: (i) GAM and (ii) DL model (see section 2.4.)

Depending on the sequence and parameters, modelling interaction effects may occur. Therefore, the interaction variable strategy was used for GAM. To select the best number of splines, a grid search method was additionally used. The DL model had an architecture consisting of five blocks with hidden layers using ReLU activation function and one input and output layer each. The number of hidden layers were optimized by random search method. The number of nodes in the input and output layers depends on the total number of input parameters and predicted signal output values, respectively. MSE and MAE were used as the loss function and regularizer, respectively, during the training process. In addition, an Adam optimizer with a learning rate of 0.01 was also used for optimization. For both methods, 10% of the dataset was randomly selected as the test dataset, and another 10% of the remaining data was used for validation in DL model and the remaining dataset was used during regression training.

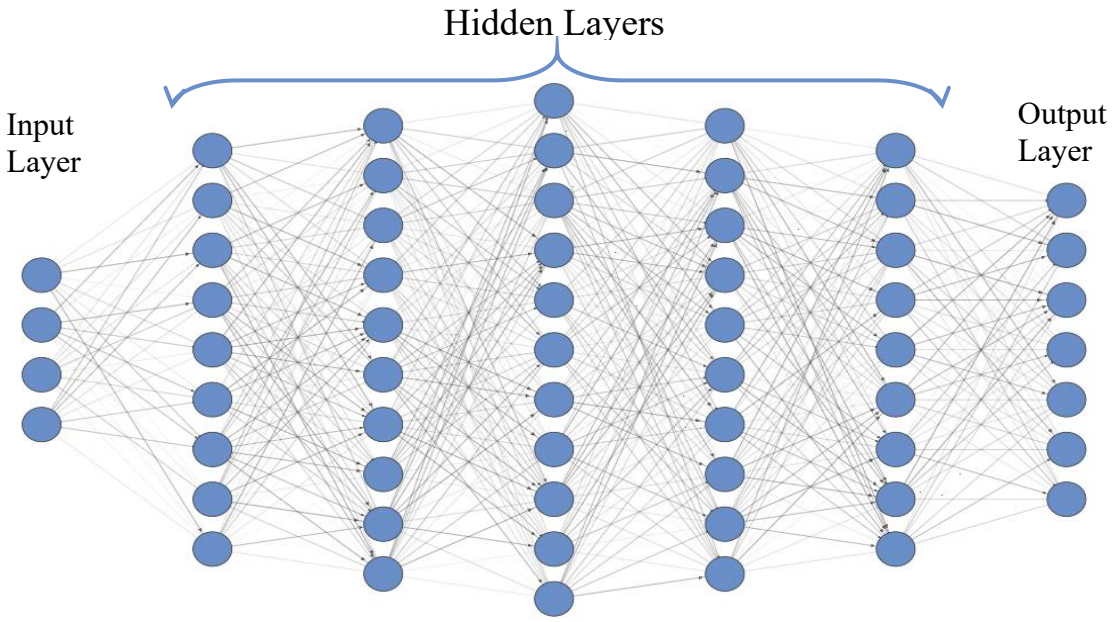

Figure 14: Visualization of a deep learning model for our regression problem. This DL model has 1 input, 5 hidden layers, and 1 output layer. The output layer has 7 nodes that correspond to the 7 signal values of the containers.

2.7.2.3. Optimization Process

After evaluating the regression models, the next step was to implement optimization methods based on them (for more detail, see section 2.5). The approximation function based on regression models may not be a convex function and may have undefined derivatives. Therefore, gradient-based optimization cannot be applied or cannot guarantee a global optimum. Three different optimization methods based on evolutionary algorithms were employed: (i) GA and (ii) CMA-ES for single objective optimization and (iii) NSGA-III for multi-objective optimization (see section 2.5 for more details). These algorithms were implemented using the PyMOO framework for multi-objective optimization in Python. For GA, tournament selection, simulated binary crossover, and polynomial mutation strategies with a default probability of 0.9 were used. For CMA-ES, a sigma value of 0.2 was used. Elitism was not incorporated into the execution of either algorithm. For NSGA-III, a population size of 360 and a number of iterations of 50 were used. The workflow continuously updated the SPS and predicted the signals for each substitute with the new SPS based on the regression model. In this loop, the SPS were optimized for two clinical use cases.

2.7.2.4. Clinical use cases

i) Achieving the same contrast as in a target image

To achieve this, we consider that the regression models predict a signal m_i in substitute i that should match the signal t_i of the same substitute in the target image acquired at 0.35T MR LINAC with the parameters combination described in section 2.7.2.1. The optimization of the SPS involves several steps: First, a set of either predefined or randomly selected parameters

is sent to the regression model. Then the regression model predicts the average signal values for each individual ROI m_i based on these parameters. The differences between the signals m_i and t_i determine the MSE loss function, which is sent to the optimizer to update the SPS. Finally, the updated parameters are transferred back to the regression model for the next iteration. This process is repeated until the objective function is below a user defined threshold value 10^{-3}.

$$\mathrm{Loss(MSE)} = \frac{\sum_{\mathrm{i}}^{\mathrm{n}}(\mathrm{t_i} - \mathrm{m_i})^2}{\mathrm{n}}$$

ii) Maximizing the contrast between different tissue types

First, a set of either predefined or randomly chosen parameters are selected and send them to regression models to predict the average signal values for each individual ROI m_i based on these parameters. Then the contrast between adjacent pairs of substitutes is calculated as the difference between mi and mj (i<j). Since each contrast represents a different clinical objective, this problem can be considered as a MOO problem (see more detail in 2.5.3). To maximize the contrast between each substitute, two different optimization approaches are evaluated. The NSGA-III method is used to optimize the MOO and obtained the Pareto front optimal solution. However, since MOO is computationally expensive and time consuming, the classical approach of considering a weighted sum of the individual objective functions was also applied to arrive at a single objective function:

$$F = \sum_{i,j} \lambda_{ij}\left(m_i - m_j\right)^2 \qquad with \quad i = 1 \dots (n-1),\ j = (i+1)$$

Here, λ_{ij} are the weights of the objective function terms $\left(m_i - m_j\right)^2$. To evaluate the performance of the multi-objective optimization algorithm, two different settings of weighting factors were evaluated. Setting 1: the weights for four containers (λ_{12}, λ_{23}, λ_{34}) were initialized to 1 and optimization was performed. Setting 2: the weighting factors λ_{12} and λ_{23} were then changed to 5 while keeping the weight λ_{34} equal to 1 and the resulting contrasts were evaluated. This allowed us to investigate how the classical MOO approach would prioritize the associated objectives in comparison to others and also compare with Pareto front optimal solutions.

2.7.3. On-the-run SPS optimization

This section describes the implementation of an innovative on-the-run optimization workflow. Most of the presented materials for “on-the-run” SPS optimization workflow have been extracted from our previous publication [88]. The workflow continuously updates the SPS and executes it on an MR scanner using the Access-i tool, resulting in the generation of new image data. This new data is then used to calculate the loss value based on the clinical objective function. The SPS is carefully optimized for two different clinical use cases within this iterative loop. Unlike a regression-based optimization workflow, the on-the-run optimization approach eliminates the need to acquire a large initial dataset. Additionally, since this workflow runs directly on the scanner, predictive models are not required.

2.7.3.1. Optimization loop via MRI interface

All measurements were performed on a 1.5 T MAGNETOM Sola MR scanner (Siemens Healthineers, Erlangen, Germany) using the 20-channel head coil. To enable the "on-the-run" optimization process, the MR scanner was remotely controlled by Access-i (Siemens Healthineers, Erlangen, Germany). The optimization process was run on a local computer (Intel(R) Core(TM) i5-9400, 2.9 GHz CPU, 6 cores and 16 GB RAM) instead on the host computer of the MRI scanner. By running the optimization process on a separate computer, we were able to optimize the MR sequence parameters based on the acquired images without interfering with the operation of the scanner using an in-house developed Python code. In the next step, optimized SPS are automatically transferred back and executed on the MRI scanner, and the optimization loop is terminated if the objective function or parameter values do no longer change with respect to a predefined threshold value.

2.7.3.2. Optimization Process

The SPS was optimized for a 2D TSE sequence with a constant BW of 186 Hz/pixel, acquired resolution of 0.4 x 0.4 mm2, slice thickness of 5 mm and TF) of 30 with an echo spacing of 11 ms. TE, TR and refocusing FA, as main contributors to contrast, were used as optimization parameters and were allowed to vary within a specific range (TE: 12 ms – 114 ms, TR: 500 ms – 2300 ms and FA: 140° – 180°). These parameters and ranges were the same as for the regression method, expect for TF, as Access-i has limitations that do not allow us to change a few sequence parameters.

The optimization workflow deals with images for discrete SPS obtained directly from the scanner via the Access-i tool. Consequently, gradient-based optimization methods were not applicable, since they require a smooth function to calculate the derivatives. We therefore employed the same evolutionary algorithms for evaluation: The GA and the CMA-ES. The workflow was designed to continuously update the SPS and to subsequently execute the SPS on an MR scanner via the Access-i tool, which in turn results in new image data. In this loop, the SPS were optimized for two clinical use cases

2.7.3.3. Clinical use cases

i) Achieving the same contrast as in a target image

For this, we consider that the scanner generates a signal m_i in substitute i which shall match the signal t_i of the same substitute in the target image acquired under different conditions, without an initial image registration process. For target image (T), the same parameter setups as for regression-based optimization workflow (described in section 2.7.2.1) were used.

The optimization of the SPS involves several steps. First, a set of either predefined or randomly selected parameters is sent to the scanner for data acquisition. Then, the substitutes are automatically segmented using a region growing segmentation algorithm and the average signal of each ROI m_i is calculated. The MSE loss function (Figure 15) is derived from the differences between the signals m_i and t_i and is used by the optimizer to update the SPS. The updated parameters are then transferred back to the scanner for another iteration. This process continues until the objective function reaches a user defined threshold value 10^{-3}. Figure 15 illustrates the pipeline for on-the-run SPS optimization.

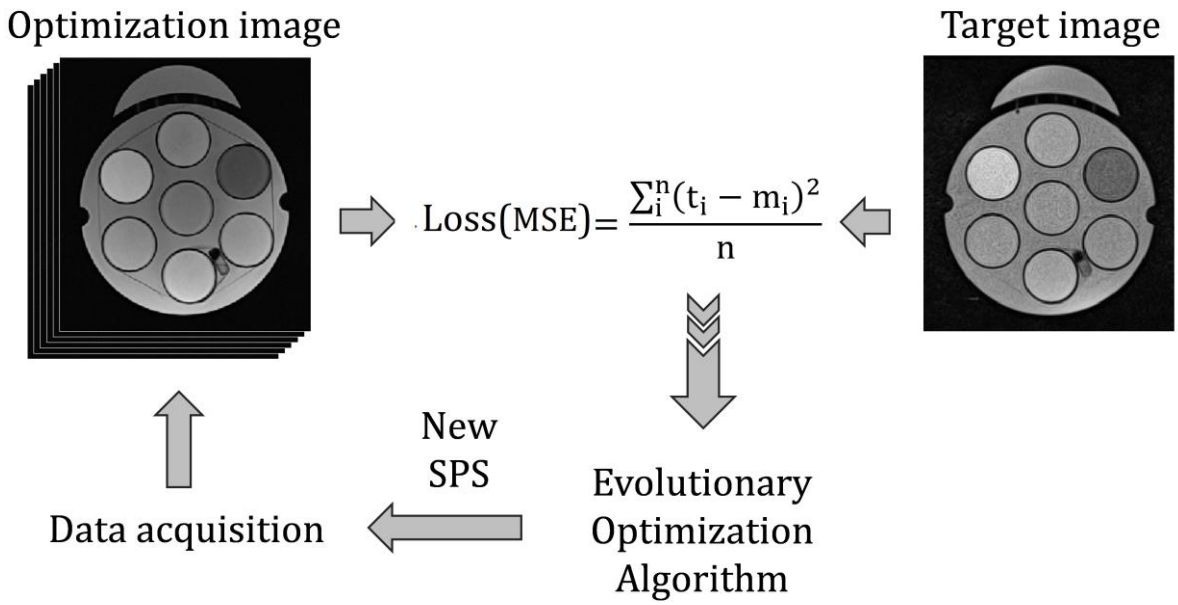

Figure 15: Flow chart of the proposed "on-the-run" optimization workflow. The optimizer uses the Access-i interface to send the sequence parameters to the MR scanner, which acquires a set of 4 MR images (population size of 4). These images are then compared to the target image to calculate the MSE. The MSE is then fed back to the optimizer to update the sequence parameters until the optimum solution is achieved. Adapted from [88]

ii) Maximizing the contrast between different tissue types

For this use case, a set of either predefined or randomly parameters are selected and automatically segment the substitutes in the acquired images. Then, the contrast between neighbouring pairs of substitutes is calculated as the difference of m_i and m_j ($i<j$). As each contrast represents a different clinical objective, this problem was considered as a MOO problem. As a MOO is computationally expensive and time-consuming, it is difficult to solve such a problem directly by measurements on the scanner. Therefore, the classical approach of considering a weighted sum of the individual objective functions is applied to arrive at a single objective function [79]:

$$F = \sum_{i,j} \lambda_{ij} \left(m_i - m_j\right)^2 \qquad with \quad i = 1 \dots (n-1),\ j = (i+1)$$

Where the λ_{ij} are the weights of the objective functions terms $\left(m_i - m_j\right)^2$. To evaluate the performance of the MOO algorithm, two different experiments were again conducted with different settings of weighting factors. In setting 1, all weights (λ_{12}, λ_{23}, λ_{34}, λ_{45}, λ_{56}, λ_{67}) were set to 1 an. In setting 2: weights λ_{34} and λ_{45} were set to 5, while keeping the other weights (λ_{12}, λ_{23}, λ_{56}, λ_{67}) equal to 1. For both settings the optimization were performed and the contrasts were evaluated.

3. Results

This chapter presents the results obtained from the different optimization experiments. Section 3.1 shows the results for segmentation, which were obtained by the region growing method. Section 3.2 describes the results obtained by utilizing a regression-based SPS optimization pipeline. Section 3.3 describes the results obtained by the on-the-run SPS optimization directly on the scanner using the Access-i interface. For both methods, two clinically relevant optimization goals were pursued: i) achieving the same contrast as in a target image, and ii) maximizing the contrast between specified tissue types.

3.1. Segmentation

After dataset acquisition, regions of interest (ROIs) are identified through an automated segmentation technique known as the region growth method (see Section 2.6 for detail). *Figure 16* shows the segmentation steps carried out by the region growth method.

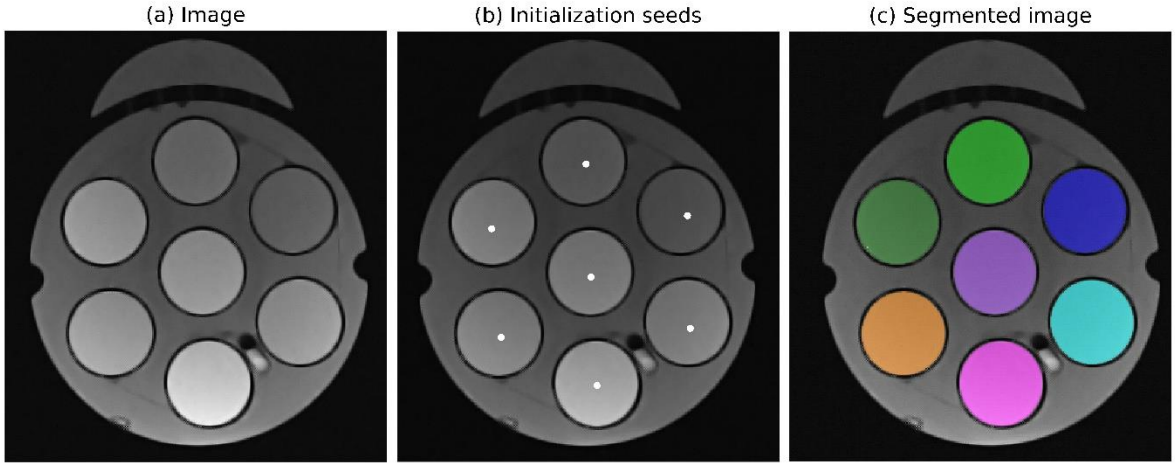

Figure 16: Illustration of the segmentation process using the region growth method: initial image (a), image after seed initialization through a manual process involving mouse clicks (b), and final segmented image using the region growth method (c), respectively.

3.2. Regression based SPS optimization

This section demonstrates the results of the SPS optimization based on two different regression methods: (a) GAM and (b) DL based model. Additionally, a detailed evaluation of different optimization methods, including CMA-ES and GA for single-objective optimization and NSGA-III for multi-objective optimization, is shown. Furthermore, the results obtained by applying the regression-based optimization method in a clinical use case are also shown.

3.2.1. Regression

Figure 17 compares the results of two different regression models: (a) GAM and (b) DL, applied to the dataset for varying TR and TE. The contrast was calculated from two neighboring vials 1/2 within the segmented ROIs as an example, after segmentation. Each graph represents the contrast values for varying TR parameters with fixed TE value. Figure 18shows surface plots of the contrast with a grid resolution of 100x100a.u. In Figure 18 (a) and (b), TF and TR are varied with a fixed TE of 11ms. In Figure 18 (c) and (d), TR and TE are varied with a constant TF of 10.

As can be seen in Figure 18, the prediction of the GAM is smoother compared to the DL regression due to the characteristic features of the splines.
To evaluate the performance of the DL and GAM regression models, a quantitative analysis was performed. For training purposes, 10% of the dataset was randomly selected as the test dataset, and another 10% of the remaining data was used for validation during DL-based regression training. After training, the performance of these two regression models was compared by evaluating the loss functions using the MAE and MSE metrics (see Table 2). Table 2 shows that the GAM exhibits lower MSE and MAE values within the training dataset compared to the DL, but shows larger deviations in the test data.

Table 2: Quantitative evaluation of contrast regression models by using MAE and MSE losses.

Methods	MAE		MSE	
	Train	Test	Train	Test
GAM	0.6	2.6	0.4	5.2
DL	1.3	1.4	1.9	2.1

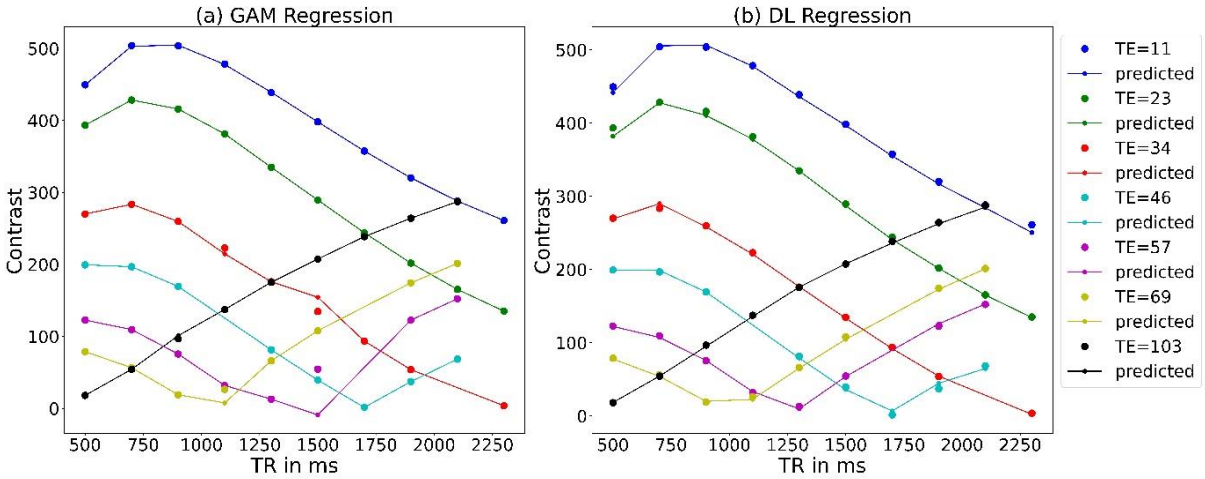

Figure 17: Measurement (dots) and model prediction (dots&line) of the contrast between 1/2 vials for varying TR and discrete values of TE for the (a) GAM and (b) deep learning-based model.

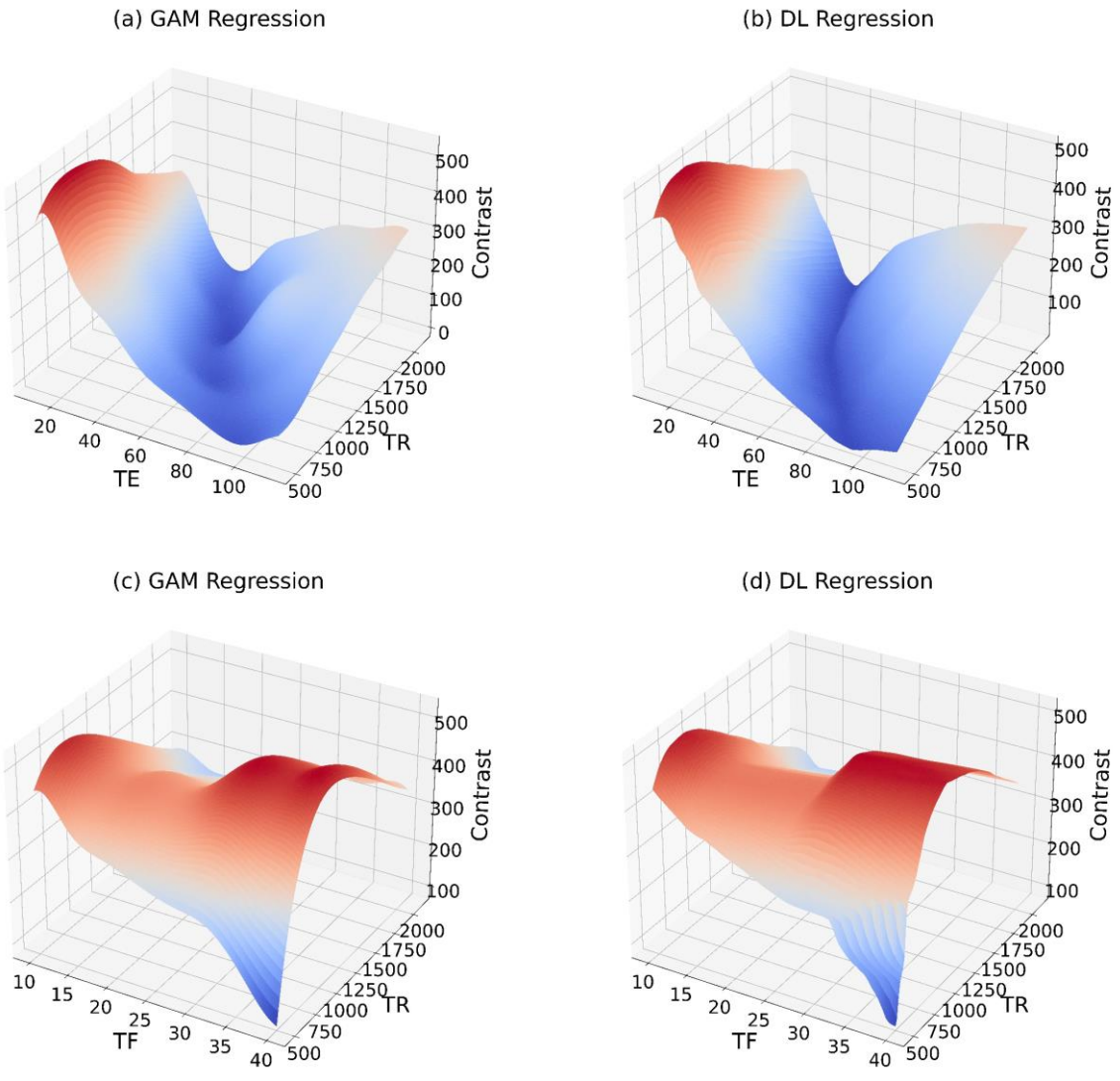

Figure 18: Contrast predicted with GAM (a, c) and DL-based regression (c, d) for varying TF and TR and fixed TE = 11 ms (a, b) and varying TR, TE and fixed TF = 10 (c, d).

3.2.2. Optimization

In this section, two clinical use cases are illustrated: (1) achieving the same contrast as the target image, and (2) maximizing the contrast in specified tissue types. Furthermore, these two clinically relevant optimization goals are evaluated using different optimization methods: (i) CMA-ES and (ii) GA for single-objective optimization for clinical use case 1 and 2, and (iii) NSGA-III for multi-objective optimization (MOO) specifically for clinical use case 2.

i) Achieving the same contrast as in a target image

Figure 20: shows the comparison between the two different optimization methods, GA and CMA-ES, used to achieve the same contrast as a target image. Figure 19: shows that GA required a larger number of iterations compared to CMA-ES; however, both methods converged at the same parameter values (FA = 173°, TF = 26, TE = 47 ms, TR = 1763 ms) and objective function values. The CMA-ES optimization method required 59 iterations to converge, while the GA optimization required 159 iterations. In addition, the finally optimized image obtained by the CMA-ES optimization method is displayed for visualization and comparison with the target image (T) acquired at the 0.35 T MR linac.

Figure 20: compares the results for population sizes of 2, 3, 4, 5, and 6, which required 185, 126, 59, 108, and 85 iterations, respectively. It was found that the 4 acquisitions per iteration is most suitable for the presented optimization case. In Figure 20:, the performance of the optimization method is shown using an intelligent initial guess (close to the previous optimization solution) of (FA = 165°, TF = 20, TE = 35 ms, TR = 1650 ms), which converged to the optimal parameter values in only 37 iterations compared to 59 iterations when starting at a random position within the parameter space. Figure 21 illustrates the progression of the parameter development during the optimization process.

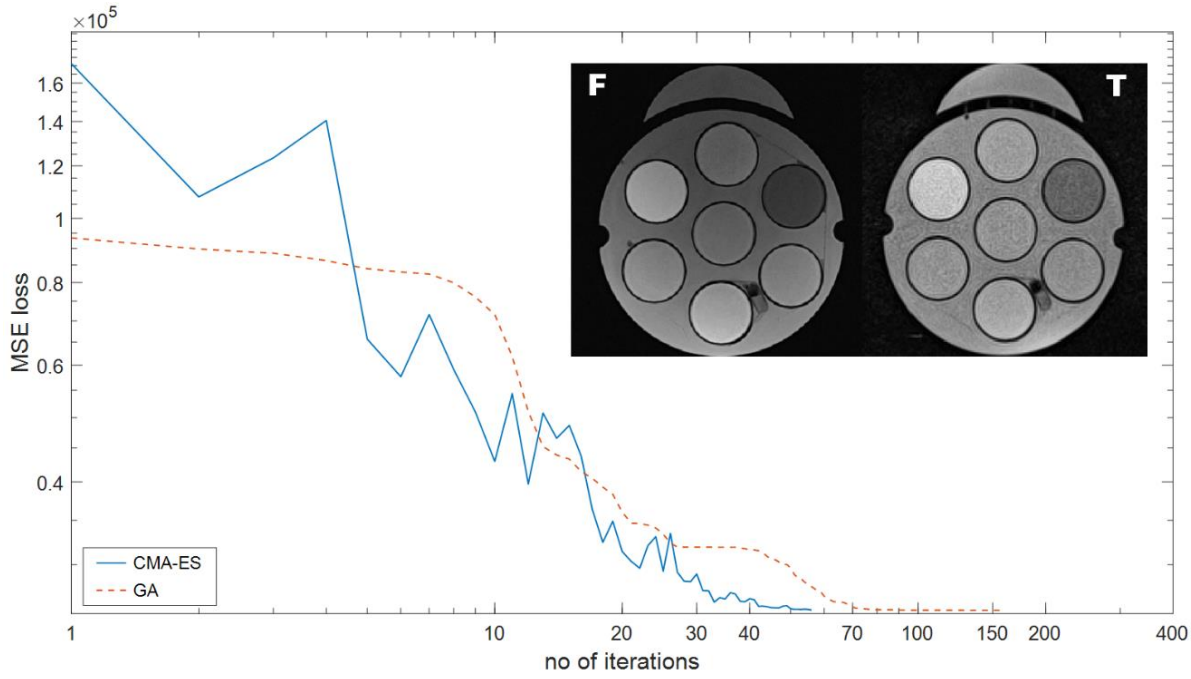

Figure 19: A comparison of the MSE loss functions for the GA and CMA-ES optimization methods is shown for clinical use case i). Additionally, the finally optimized image (F) obtained by the CMA-ES method as well as the target image (T) are displayed.

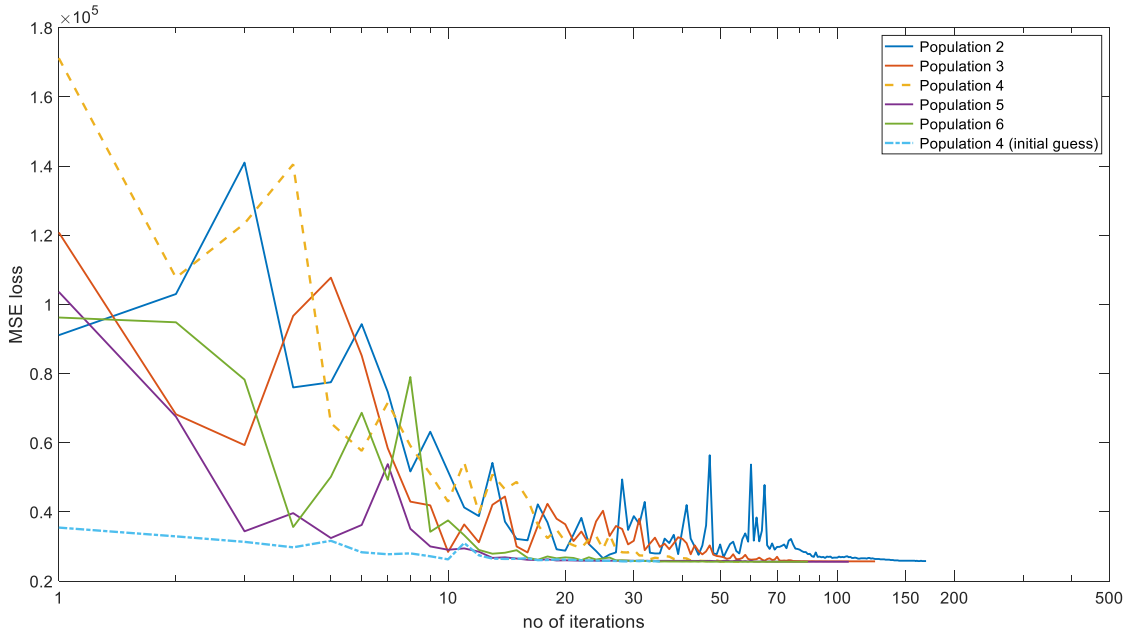

Figure 20: Comparison of the MSE loss functions for the CMA-ES optimization method with varying population sizes for clinical case i). The number of iterations required for convergence with population sizes of 2, 3 4, 5, and 6 were 185,126, 59, 108, and 85, respectively. An evaluation with an initial guess and a population size of 4 converged in just 37 iterations.

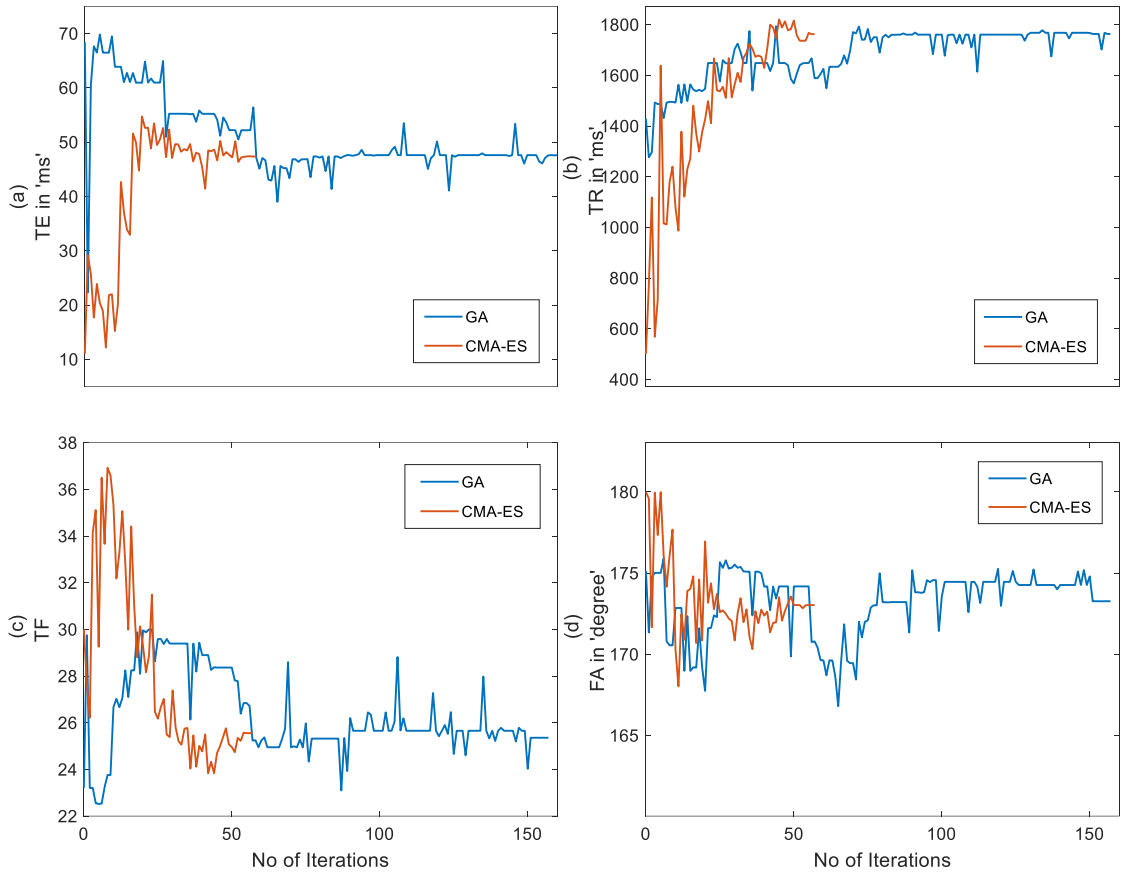

Figure 21: Development of TE (a), TR (b), TF (c), and FA (d) parameters during the regression-based optimization process for clinical use case i). CMA-ES converged in fewer iterations compared to GA, although both methods result in the same SPS.

ii) Maximizing the contrast between different tissue types

Figure 22: Figure 22: shows the Pareto front MOO optimal solutions for the contrast between substitutes 1/2, 2/3, and 3/4 obtained by the NSGA-III method. For better visualization, only the contrast between substitutes 1/2, 2/3, and 3/4 are presented. The CMA-ES was also evaluated as a classical MOO in terms of maximizing the contrast between selected substitutes. Table 3 shows the results of the CMA-ES optimization method for two analyzed weight settings. Setting 1 (λ_{12}, λ_{23}, λ_{34} = 1) resulted in a well-distributed contrast between substitutes. Setting 2 (increasing the weighting factors λ_{12} and λ_{23} to 5) increased the contrast between substitutes 1/2, but at the cost of decreased contrast between substitutes 3/4. The contrast between substitutes 2/3 remained the same in this case. We found (FA: 170°, TF = 10, TE: 57 ms, TR: 2010 ms) and (FA: 179°, TF = 36, TE: 39 ms, TR: 2100 ms) as the optimal parameter values for setting 1 and setting 2, respectively.

Figure 23: shows the Pareto front of optimal solutions obtained using the NSGA-III method. These plots also include solutions obtained using CMA-ES for the two different weight settings

1 and 2. The results show that by varying the weighting factors λ_{ij}, different optimal contrast combinations between all the substitutes can be obtained in the Pareto front optimal solutions.

Table 3: Signal and contrast for the measured substitutes for two different sets of weighting factors.

Container	1		2		3		4
Case 1 ($\lambda_{12}=\lambda_{23}=\lambda_{34}=1$)							
Signal [a.u.]	975.01		618.63		1312.71		1627.69
Contrast [a.u.]		356.41		696.17		312.98	
Case 2 ($\lambda_{12}=\lambda_{23}=5$, $\lambda_{34}=1$)							
Signal [a.u.]	885.01		434.51		1130.51		1300.86
Contrast [a.u.]		450.49		695.99		170.34	

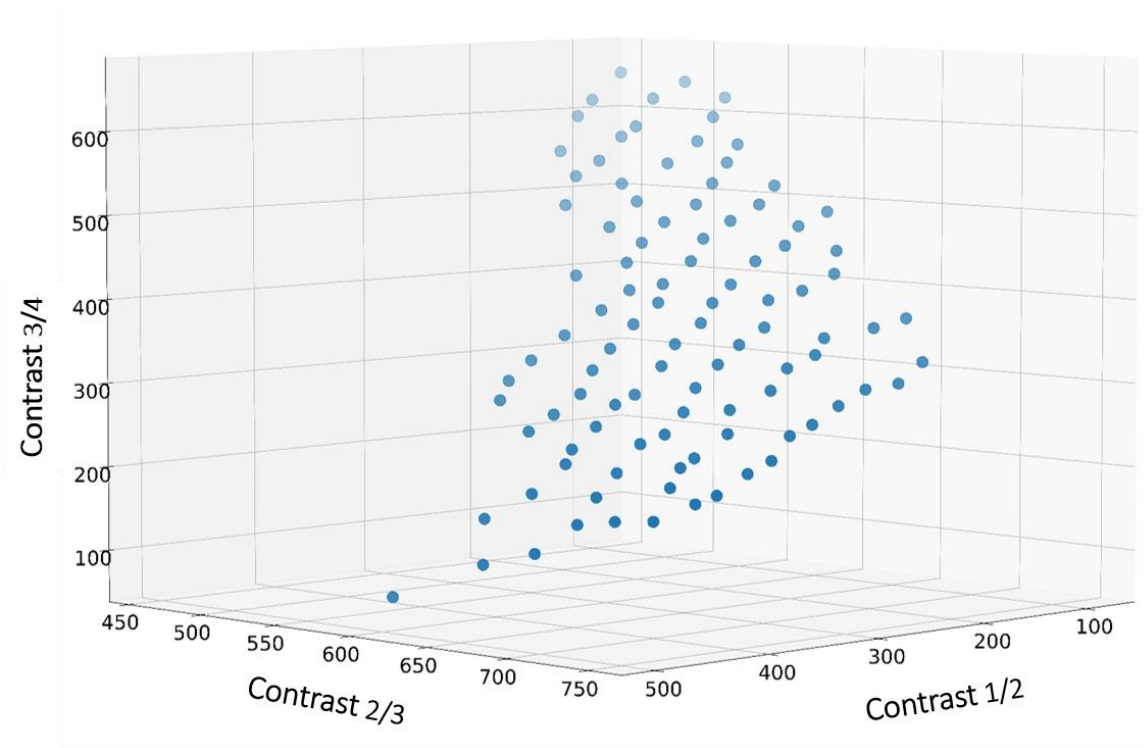

Figure 22: Results of the multi-objective optimization using NSGA-III to maximize the contrast between the substitutes 1/2, 2/3, and 3/4. The Pareto front shows all the optimal contrast combinations with in the given parameter space.

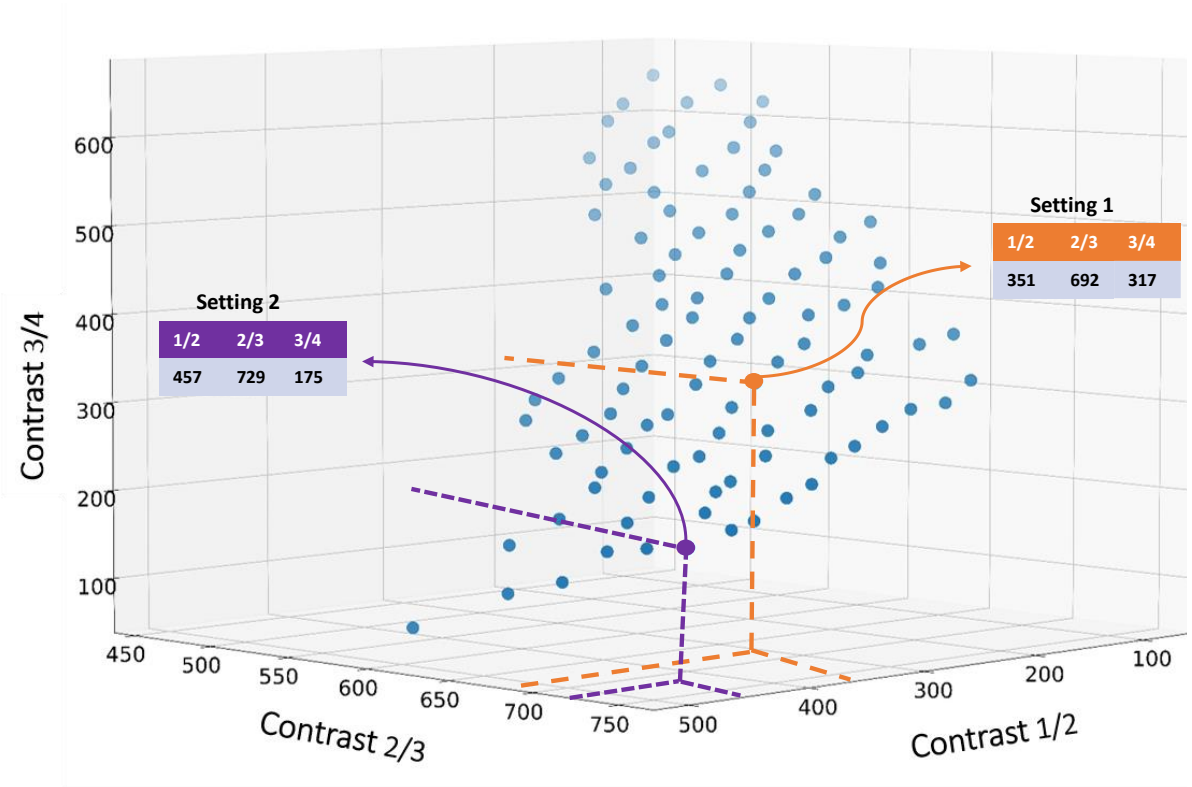

Figure 23: Pareto front optimal solutions obtained by using the NSGA-III method for clinical case ii). The graphs also represent the solution obtained by using CMA-ES for two different weight settings: setting 1 (all weights set to 1) and setting 2 (λ_{12} and λ_{23} = 5 and λ_{34} = 1).

3.3. On-the-run SPS Optimization

This section presents the detailed results of the on-the-run SPS optimization results, in which the MR scanner was remotely controlled using the Access-I interface. Some of the presented results have been extracted from our previous publication [88]. Both, CMA-ES and GA optimization methods were used for both clinical use cases. For contrast maximization (case ii), only classical multi-objective optimization was utilized by introducing weights, as MOO requires a large number of acquisitions to find the optimal Pareto front, which is not feasible for the on-the-run optimization.

i) Achieving the same contrast as in a target image

Figure 24: shows the loss functions for the both GA and CMA-ES optimization techniques. The results show, that also here, the CMA-ES method is almost 1.5 times faster than the GA method, taking approximately 2.5 hours to converge to the same objective function and parameter values (FA: 180°, TE: 50 ms, TR: 2010 ms) as compared to 4 hours for the GA method. The CMA-ES optimization took 73 iterations to achieve the optimal value, while the GA optimization took 172 iterations. The overall optimization time is the result of the acquisition time for the sequences per iteration (10 - 40s), total number of acquisitions (73 vs 172, respectively), the time required for remotely controlling the MRI via Access-i, and the execution of the optimization algorithm, with the first two being the limiting factors.

Figure 24: visualized the progress of the optimization process by displaying the image contrast at 10, 20, 30, 50, and 73 iterations for the CMA-ES algorithm. These corresponding images were then compared with the target image (T) acquired at the 0.35 T MR linac. The excellent agreement between the optimized and target images indicates that the contrast optimization was successful. Figure 25*:* provides a visualization of how the parameters evolved throughout the optimization process.

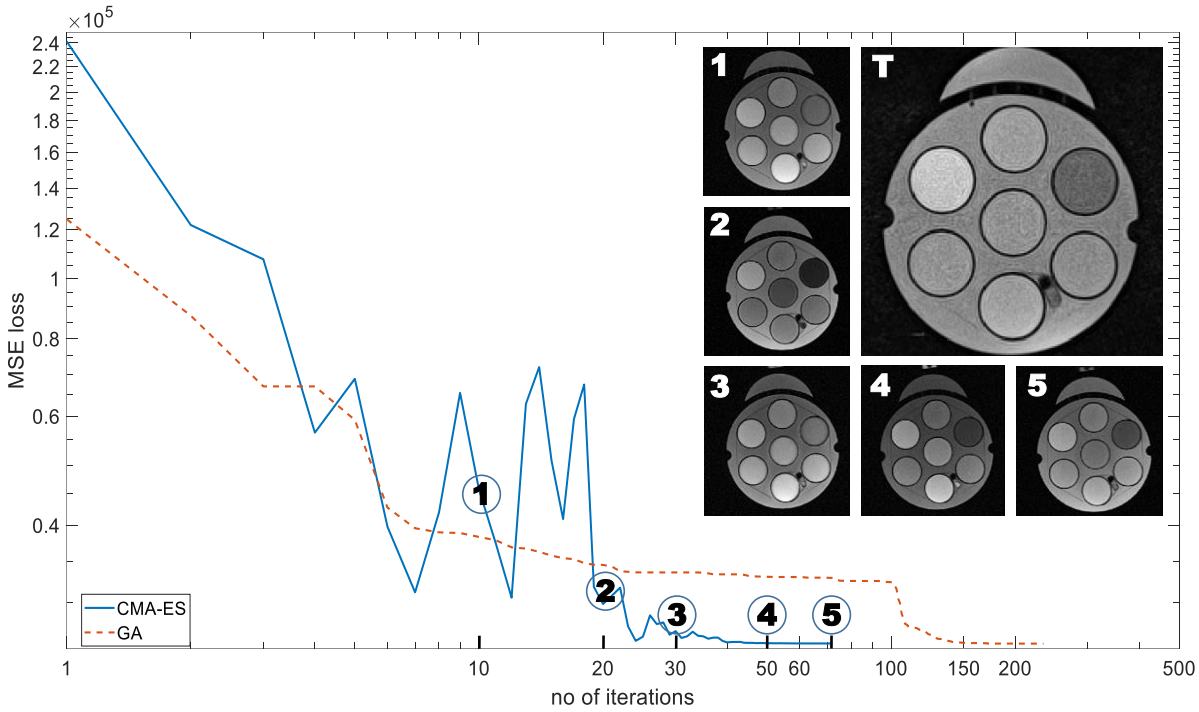

Figure 24: A comparison of the MSE loss functions for the GA and CMA-ES on-the-run optimization methods for clinical use case i). Additionally, five images at different iteration numbers (10, 20, 30, 50 and 73) for the CMA-ES method as well as the target image (T) are shown. Adapted from [88]

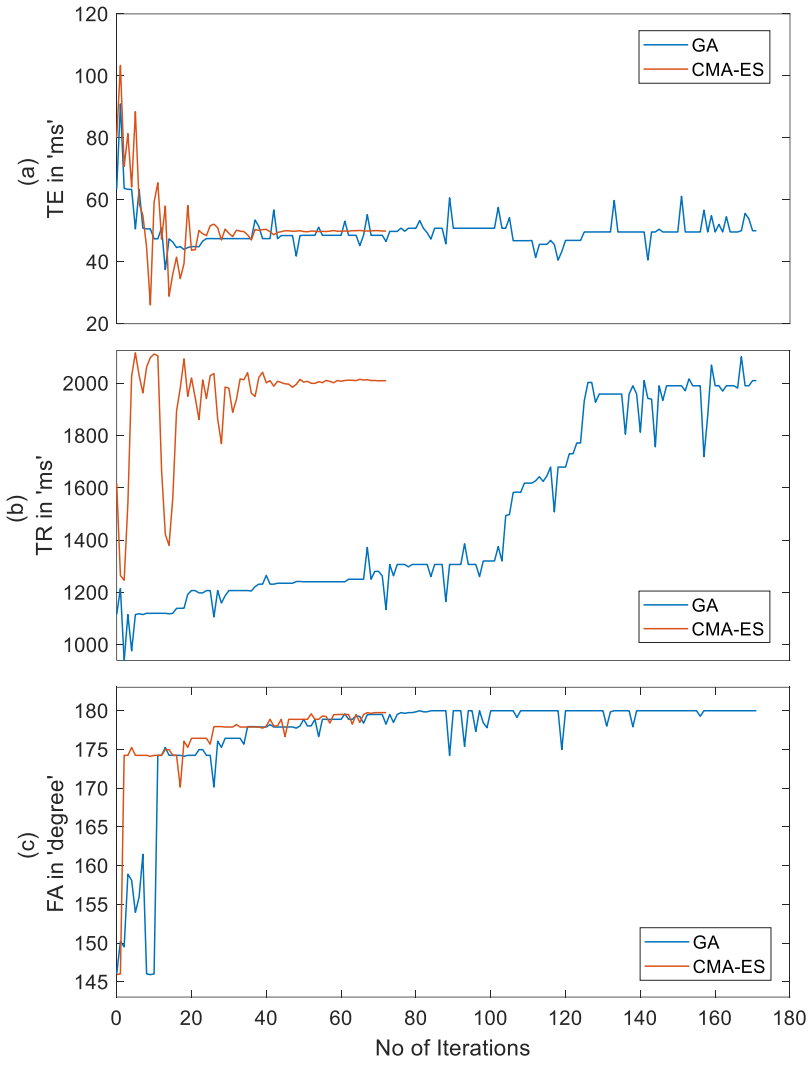

Figure 25: Shows the development of TE (a), TR (b) and FA (c) parameters during the on-the-run optimization process for the clinical use case i). CMA-ES converges in fewer iterations compared to GA, although both methods result in the same final parameter values. Adapted from [88].

ii) Maximizing the contrast between different tissue types

The CMA-ES and GA methods were also evaluated to optimize for contrast maximization between selected substitutes. Table 4 shows only the results of CMA-ES, since CMA-ES converged in fewer iterations and achieved the same optimal parameter values. Two settings with different weighting factors were analyzed: setting 1, where all weights were set to 1, and setting 2, where the weighting factors λ_{34} and λ_{45} were increased to 5. In setting 1, the contrast between the 3/4 and 4/5 substitutes was relatively low. However, in setting 2, the contrast between these substitutes improved significantly, but at the expense of decreased contrast between the other substitutes. The optimal SPS for settings 1 and 2 were found to be (FA: 173°, TE: 91 ms, TR: 2090 ms) and (FA: 180°, TE: 11 ms, TR: 700 ms), respectively.

Figure 26: provides a visual representation of the results by showing the initial and final images for both cases.

Table 4: Signal and contrast for the measured substitutes for two different sets of weighting factors.

Container	1	2	3	4	5	6	7
Case 1 ($\lambda_{12}=\lambda_{23}=\lambda_{34}=\lambda_{45}=\lambda_{56}=\lambda_{67}=1$)							
Signal [a.u]	715.43	232.46	826.68	940.52	894.82	1227.79	494.94
Contrast [a.u]	482.97	594.22	113.84	45.70	332.97	732.85	
Case 2 ($\lambda_{12}=\lambda_{23}=\lambda_{56}=\lambda_{67}=1$, $\lambda_{34}=\lambda_{45}=5$)							
Signal [a.u]	688.79	612.78	1030.27	1627.86	1126.77	1095.36	1213.98
Contrast [a.u]	76.01	417.49	569.59	501.09	31.41	118.59	

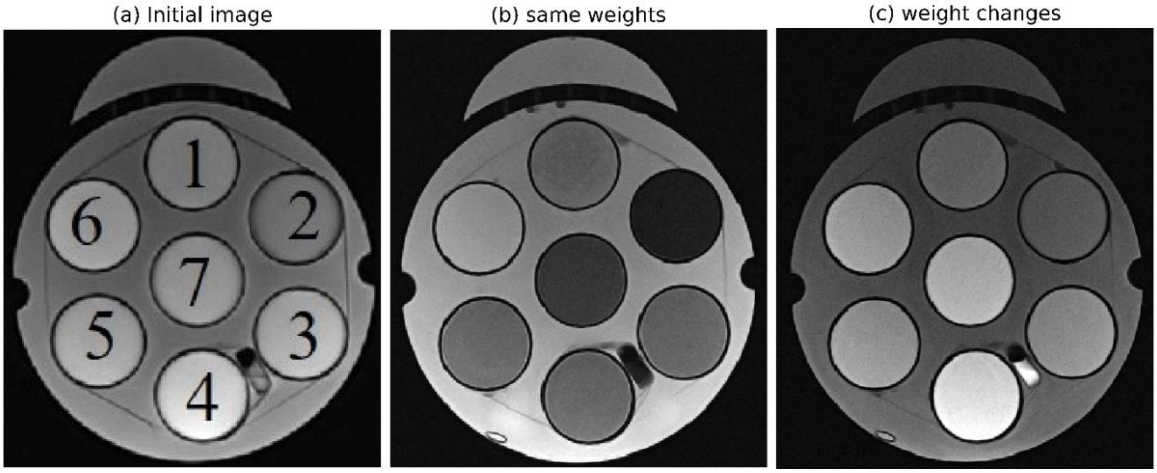

Figure 26: Illustration of the impact of the two sets of weighting factors: initial image (a) and image after optimization with $\lambda_{12}=\lambda_{23}=\lambda_{34}=\lambda_{45}=\lambda_{56}=\lambda_{67}=1$ (b, setting 1) and λ_{34} and λ_{45} raised to 5 (c, setting 2), respectively. Adapted from [88]

4. Discussion

In this chapter, the two previously introduced optimization methods are discussed. Some of the presented materials have been extracted from our previous publication [88]. In the present work, a proof-of-concept for the automatic optimization of contrast in MRI sequences was demonstrated by applying the sequence of SPS using two different optimization approaches: (I) Regression-based optimization method and (II) On-the-run optimization method. Furthermore, for both methods, it was demonstrated how MR sequences can be optimized specifically for application in radiotherapy using two clinical use cases: (1) achieving the same signal as in a target image and (2) maximizing the signal difference between different tissue types. Use case (1) improves radiotherapy planning in MRgRT, where diagnostic MRI images may be easier registered to images obtained at the MR-Linac [89, 90]. This registration may be facilitated, if the image from the diagnostic MRI is optimized to have the same signal as the target image from the MR-Linac. Use case (2) on the other hand, is useful to better distinguish adjacent tissue structures and to automatically delineate them for treatment planning.

4.1. Regression-based Optimization method

A proof-of-concept for automatic contrast optimization in MRI sequences was demonstrated using the SPS sequence based on a regression model. The optimization was performed with images of a 1.5T MR Aera scanner using 2D TSE sequences due to their relatively short acquisition time. The contrast was optimized by adjusting four parameters: TR, TE, TF and FA. However, it is important to note that additional parameters could potentially be included in this optimization process. Overall, the final data set of acquired images comprised over 1114 individual SPS.

For the regression-based optimization method, two regression models were compared: (a) GAM and (b) DL to predict contrast between given substitutes. GAMs are useful when the relationships in the data are not strictly linear and can handle complex, non-linear patterns without relying on a predetermined functional form. One of the key features of GAMs is their ability to incorporate spline functions to effectively model non-linearities. These spline functions allow GAMs to flexibly capture intricate relationships between predictors and the response variable, making them valuable tools for regression analysis in various fields. In contrast, the DL model is a neural network-based model that can learn linear or complex nonlinear relationships between variables. Unlike traditional regression models, DL models can automatically discover intricate patterns and hierarchies in the data by learning from large amounts of dataset. Both models were trained on the acquired dataset and their performance was evaluated using two different metrics: the MSE and MAE. These MAE and MSE metrics are mainly used to evaluate the prediction error rates and model performance in regression analysis. The DL model achieved an MSE of 1.9 and an MAE of 1.3 on the training dataset, and an MSE of 2.1 and an MAE of 1.4 on the test dataset. In comparison, the GAM model achieved an MSE of 0.4 and an MAE of 0.6 on the training dataset, and an MSE of 5.2

and an MAE of 2.6 on the test dataset (see Table 2). Results showed that the GAM model had lower errors for the training dataset than the DL model. However, there was a notable deviation for the test dataset due to overfitting, particularly visible in Figure 17 at TR=1500 ms and both TE=57 ms and TE=34 ms in the GAM model, as these points were specifically chosen for the test dataset. In contrast, the DL model performed well on both the training and test datasets. Mitigating overfitting is relatively easy since a validation test can be applied using validation data during training to analyze the model and prevent overfitting. Conversely, controlling overfitting in GAM during training is challenging because validation tests cannot be applied. In addition, selecting the optimal number of splines to train the GAM to avoid overfitting is another challenging task. To address this, the Python implementation of the GAM tool [91] used in this study offers additional optimization methods, such as grid search and random search. These techniques systematically explore various spline configurations, allowing the most suitable number of splines for the specific GAM to be identified. Despite the utilization of an additional grid search method to address this problem, some overfitting was still observed in the model.

Depending on the sequence and SPS, interaction effects within the variable may occur. While the integration of interaction variables into a GAM is possible, these interaction variables have to be pre-defined. For this, however, a prior knowledge of the physical dependence of these variables is required. The DL-based model on the other hand can automatically detect the described interactions between all parameters without the need for additional information. On the other hand, the GAM model is much faster and requires fewer computational resources (no GPU) compared to the DL model, even with a large dataset to train. The DL models take several hours to be trained and require high computational power (GPU).

DL is a powerful technique that can predict multiple outcomes using only one model, as shown in the Figure 14. Despite the already mentioned time-consuming nature and the need for high computational power in the training-phase of a deep learning model, it only needs to undergo training once to support multiple predictions, which may consequently save both time and computational resources. On the other hand, for a GAM training a separate model for each contrast combination is required. This means that for n objective functions, the number of GAM models is n, while in the DL-based model approach only one can be sufficient. Therefore, a DL model is a more efficient approach for multi-prediction tasks. Once a reliable regression model has been established, optimization can be conducted using the regression model.

4.1.1. Optimization

i) Achieving the same contrast as in a target image

In In this work, two single-objective optimization algorithms based on evolutionary strategies: GA and CMA-ES, were evaluated. Each iteration of the algorithms runs on a fixed population size, determining how many samples have to be acquired and evaluated per iteration. As shown in Figure 19, CMA-ES only required 59 iterations (resulting in a total of 236 acquisitions), while GA required 159 iterations (resulting in a total of 636 acquisitions) using

a population size of 4. Since, CMA-ES required fewer iterations than GA, the results obtained from CMA-ES were presented for further analysis. In *Figure 20*, different population sizes were tested, with 2–6 acquisitions and found that a population size of 4 was most suitable for both GA and CMA-ES, based on time efficiency. The number of iterations required for CMA-ES for convergence with population sizes of 2, 3, 5, and 6 was 185, 126, 108, and 85, respectively. The termination criteria were consistently applied across all population sizes: The objective function value remained unchanged beyond the user define threshold of 10^{-3}, or none of the variables exhibited changes greater than 1, as the variables were considered only as integer values. Additionally, the PyMOO framework that has been used for the implementation of the individual algorithms, has the ability to select an initial guess, which is utilized within one of the population selections. The performance of such an initial guess in CMA-ES was evaluated by employing a parameter set close to the optimal value, leading to convergence with a population size of 4 in just 37 iterations. These findings could be helpful for optimizing the sequence SPS in-vivo and in clinical practice, this initial guess during an in-vivo optimization could consist of clinically suggested optimized parameter sets or by optimized parameters from a phantom study.

Overall, the comparison of the two presented optimization methods shows that CMA-ES converges faster and should therefore be preferred over the GA algorithm for the presented optimization cases. It is important to note, however, that the required population size and the number of iterations may differ significantly depending on the number of sequence parameters included in the optimization and the resulting complexity of the objective function. Regarding population size, there is a trade-off between slowing down the optimization by larger population sizes and insufficient diversity if the size is too small.

ii) Maximizing the contrast between different containers

Here we wanted to maximize the contrast between the adjacent vails. As these represent different clinical objective, this problem is considered as a MOO problem, and therefore a different optimization approach as presented in i) had to be used. For this, two different MOO methods were implemented: the classical approach, which is a scalarization method that transforms a multi-objective function into a scalar function by applying a weighted sum of each objective function (see section 2.5.3 for more details), and NSGA-III, an evolutionary algorithm designed to find the best Pareto optimal solutions using elitism and a fast non-dominated sorting approach. In the multi-objective optimization analysis, the results demonstrated that NSGA-III produced a well-spread pareto-front optimal solution. Despite the extensive time required for multi-objective optimization, using 250 populations and 50 iterations, resulting in a total of 250 x 50 = 12,500 measurements, the NSGA-III method proved to be a powerful technique for solving MOO. It provided all the objective contrast solutions through a single optimization method, saving a significant amount of time and computational resources. In contrast, the classical approach requires optimization from scratch with new combinations of weighting factors, which is not an efficient method.

Furthermore, to evaluate both classical and NSGA-III multi-objective optimization approaches and also to understand the impact of weighting factors (λ_{ij}) in the classical approach, two

settings of weighting factors were investigated: Setting 1) (λ_{12}, λ_{23}, λ_{34} to 1) resulted in a well-distributed contrast between substitutes. Setting 2 (increasing the weighting factors λ_{12} and λ_{23} to 5) which in turn increased the contrast between substitutes 1/2 and 2/3. The obtained results by using a classical approach of considering a weighted sum of the individual objective functions, also show the trade-off between NSGA-III and classical approach optimization method. For both settings 1 and 2, the optimal solution of classical method was found within the pareto-optimal solutions which was obtained using NSGA-III. By altering the λ values, an investigation was conducted into how the classical optimization algorithm prioritizes the associated objectives in comparison to the others.

For multi-objective optimization problems, the NSGA-III optimization algorithm has also been implemented. NSGA-III is a MOO algorithm that extends the classical genetic algorithm to solve problems with multiple conflicting objectives. It has several advantages over classical weighting methods in multi-objective optimization. Unlike classical methods that require the specification of weighting factors for each objective, NSGA-III eliminates this need, making it more applicable to real-world problems. It uses non-dominated sorting and crowding distance to maintain a diverse population of solutions, which is critical when dealing with multiple conflicting objectives and a diverse set of solutions is desired. Designed to handle problems with different types of Pareto fronts, including convex, concave, and disconnected fronts, NSGA-III adapts well to different problem structures without requiring specific knowledge of the problem domain.

4.2. On-the-run optimization

In the present study, a proof-of-concept was demonstrated for fully automated optimization of contrast in MRI sequences by applying the sequence of SPS directly on the scanner. The optimization was performed for 1.5T MR Sola scanner using 2D TSE sequences due to its relatively short acquisition time. The contrasts were optimized by changing the three parameters TR, TE, and FA, however, more parameters could in principle be included.

i) Achieving the same contrast as in a target image

In this study, SPS optimization was performed by aiming at a similar contrast as the target image from different scanners or the same scanner with different SPS. The same two optimization algorithms based on evolutionary strategies: GA and CMA-ES were evaluated. Different population sizes, ranging from 2 to 5 acquisitions, were tested and obtained consistent results similar to those obtained in the regression-based approach. Again, a population size of 4 was found to be optimal for both GA and CMA-ES in terms of time efficiency. In particular, CMA-ES required 73 iterations, resulting in a total of 292 acquisitions and a total optimization time of approximately 2.5 hours. Meanwhile, GA required 172 iterations, resulting in a total of 688 acquisitions and a total optimization time of approximately 4 hours. The total optimization time also depends on the parameter combinations chosen during optimization, as small TR values require less acquisition time compared to long TR values. The number of iterations required for CMA-ES to converge with population sizes of 2, 3, and 5 were 162, 125, and 93, respectively.

ii) Maximizing the contrast between different containers

As shown before, in multi-objective optimization in clinical use case 2 (maximization the contrast between adjacent vials), NSGA-III is computationally expensive and time consuming compared to the classical approach of a weighted sum of the individual objective functions to one single objective function. Due to the large number of acquisitions, it is difficult to solve such a problem directly from measurements on the scanner as shown in regression-based optimization method. Here, two different settings of weight factors were investigated for maximizing the signal between adjacent substitutes: Setting 1, where all weights were set to one (λ_{12}, λ_{23}, λ_{34}, λ_{45}, λ_{56}, λ_{67} = 1), resulted in a rather low contrast between substitutes 3/4 and 4/5 as shown in Figure 26. Changing the weights of individual containers, as presented in setting 2 ($\lambda_{34} = \lambda_{45} = 5$), significantly improved the contrast between the respective substitutes 3/4 and 4/5, but at the expense of reduced contrast between the other substitutes. For both setting 1 and setting 2, the number of iterations required to converge using CMA-ES with a population size of 4 was 67 and 59, respectively.

In this work, a real-time scanner remote control tool "Access-i", developed by Siemens Healthineers in Erlangen, Germany, was used. Access-i provides access to all sequences implemented on the scanner by the manufacturer. In addition, it exists in several versions, including a graphical interface and a Python script version. For full control of the scanner, the Python version of Access-i was used in this work by integrating Access-i into the optimization pipeline. However, this tool has some limitations. Certain MR parameters remain inaccessible via Access-i. For example, parameters such as echo spacing and echo train length/turbo factor play a critical role in contrast formation, but are not adjustable through Access-i. Another limitation is that the Access-i tool has a fixed timer for executing the second command. However, after including the optimization algorithm, the optimization process takes additional time, resulting in code crashes. To address this issue, an additional timer has been introduced to consider the additional time of the optimization. In addition, a recently released real-time scanner remote tool known as Pulseq is an open-source framework specifically designed for the development and execution of MR pulse sequences used in imaging and spectroscopy. With Pulseq, MRI sequences can be programmed directly in MATLAB or Python and executed on real MR devices [6, 92]. However, working with Pulseq requires a detailed understanding of the manufacturer's pulse schemes, including the fixed timing of gradients and RF pulses used in the clinical sequence.

As an alternative to the employed optimization algorithms, one could also use a discrete gradient descent optimization method [93], a derivative-free approach for solving unconstrained non-smooth optimization problems. This method is based on the concept of discrete gradients, which can approximate the sub-gradients of a wide range of non-smooth functions. Furthermore, this method can improve computational efficiency, as it requires a small population size, therefore requiring a smaller number of image acquisitions. Previous studies [94-96] have shown that this method is computationally efficient in solving non-smooth optimization problems. However, it is important to note that the performance of this method may vary depending on the specific characteristics of the optimization problem.

In contrast to the present approach, SPS optimization based on Bloch equations can be performed, which may be faster compared to real acquisition-based optimization as it does

not require a MR device. However, it heavily relies on the knowledge of Bloch simulation to implement a specific sequence and accurate T_1 and T_2 relaxation times for each tissue, which are not necessarily known. Recently, a self-learning framework called 'MR-zero' [7], utilizing the Pulseq-tool, has been proposed, which adapts and optimizes MRI sequences based on a Bloch equation simulation. The generated pulse sequence, still requires knowledge of Bloch simulation in order to perform MR sequence optimization. In the more advanced version "MR-double-zero" [8], the "Pulseq" tool is still utilized to remotely control the scanner, however, the optimization directly operates on the acquired imaging data without requiring a Bloch simulation model or any further human interaction. However, implementing a sequence in Pulseq remains a challenging task. In contrast, the method used in this thesis operates directly on the MR scanner, accessing predefined clinical sequences using the Access-i interface of the manufacturer without prior knowledge about sequence details. This approach simplifies the optimization process and makes it feasible.

In recent times, several methods for optimizing MR sequence parameters have been proposed. One notable approach is Joint MR Sequence Optimization [97], which combines MR physics knowledge with neural network techniques to enhance super-resolution in spin-echo MRI. This approach simultaneously optimizes the RF pulse train for PD and T_2-weighted TSE sequences. By integrating known-operator learning, the RF pulse design generates optimal signals for the NN. Importantly, this method bridges the gap between physics-based optimization and AI-driven approaches. Additionally, AI-driven and automated optimization [98] revolutionizes MR sequence design. ML within the Pulseq framework enables direct MRI sequence formulation in MATLAB. Using simulated phantom data as a training database, ML algorithms capture the intricate relationship between sequence parameters and simulated outcomes. Both methods employ Bloch simulation with the Pulseq tool to implement the sequence. However, our proposed "on-the-run" optimization method has a unique property: it optimizes the SPS directly on clinical sequences implemented by the manufacturer.

In addition, some limitations of on-the-run optimization method should be addressed. First, the potential influence of gradient heating and field drifts due to the long image acquisition time must be acknowledged, as they may affect image quality and measurement accuracy [99, 100]. Secondly, the phantom was newly prepared following the recipe as described in [87]. However, the T_1 and T_2 values were not verified again. Finally, the optimal parameters obtained by using the proposed workflow may be different from typical clinical parameters and while the new contrast may be beneficial for a certain application, it may come along with a suboptimal diagnostic image quality.

4.3. Comparison of the applied methods: regression-based vs. on-the-run optimization

Both the regression-based method and the on-the-run optimization method have been used to optimize the sequence parameter sets. However, the regression-based method is a inefficient method that requires acquisition of a very large dataset including image acquisitions at all potentially relevant parameter combinations of the parameter space, the implementation of a regression model, and the application of the optimization method. In contrast, the on-the-run optimization is a fully automated MR sequence optimization method without any human interaction. This model automatically segments all the vials, calculates the required clinical objectives, and performs the optimization directly on the MR scanner using the Access-i tool.

For instance, to build a regression method, obtaining S = 10 entries in each of the parameter spaces (number of parameters N = 4) would lead to $S^N = 10^4$ necessary measurements and the problem of defining appropriate boundary conditions. In contrast, the on-the-run optimization method can be seen as an advanced, sophisticated and efficient search in the MR parameter space to solve both clinical use cases. The autonomous on-the-run optimization required only 300 acquisitions, which took about 3 hours directly at the MRI scanner. This is still long for an MRI examination, but faster compared to the regression-based optimization approach. In addition, the scanning time in the on-the-run optimization method could be decreased further by intelligently selecting the initial guess. The results show that by selecting a suitable initial guess, e.g., from a previously optimized SPS could significantly decrease the acquisition time.

For the clinical use case of maximizing the signal difference between tissues, we discussed that this problem is a MOO problem and has more than one solution. To solve such problems with a population size of 250 and 50 iterations, 12,500 measurements are required. However, this large number of measurements performed directly on the MRI scanner not feasible. Therefore, only the classical approach was investigated for the on-the-run optimization, and NSGA-III does not appear to be possible for this case. However, in the regression-based method, a large number of data sets were acquired by varying different parameters, and two regression models (GAM and DL) were built. These models can accurately predict the unknown contrast values for given SPS. Therefore, during the optimization process, the regression model was used as an objective function that gives the contrast values for all substitutes without performing any measurements on the scanner. Therefore, NSGA-III was used and analyzed for regression-based optimization.

4.4. Future work

In this proof-of-principle study, the optimization was performed on a phantom and for clinical implementation, the optimized sequences need also to be tested in-vivo. While phantoms measurements are useful to establish and calibrate an optimized sequence, images may additionally be affected by differences in object size, conductivity and by complex physiological and/or dynamical conditions within the human body. Therefore, the contrast obtained with the optimized SPS has to be validated in humans. As the optimization of the rather simple and fast TSE already took almost 3h, it may not be feasible to perform the full optimization in-vivo, however, the SPS could be pre-optimized in anthropomorphic phantoms and the obtained SPS may be used as a starting point for further optimization in vivo. This approach may significantly reduce the required optimization times in humans.

Further, more clinical sequences may be integrated into the optimization pipeline, for example, the 3D True Fast Imaging with Steady State Free Precession used for planning at the MRIdian MR-Linac. Finally, further studies have to investigate additional optimization parameters such as CNR, SNR and acquisition time.

4.5. Conclusion

In conclusion, both proposed frameworks, (I) regression-based optimization and (II) on-the-run optimization," for an automated multi-parametric optimization of SPS on MRI scanners have the potential to enhance the quality of MRI images for specialized purposes in MRgRT. The optimization workflow has been established and exemplarily tested for two clinical use cases: (i) achieving the same signal as in a target image and (ii) maximizing the signal difference (contrast) between different tissue types. The evaluation of two optimization methods based on evolutionary strategies suggests that Covariance Matrix Adaptation Evolution Strategy (CMA-ES) is an efficient approach to optimize the signal towards a target image or to optimize signal difference between two given tissues for both frameworks.

Moreover, the evaluation of Non-Dominated Sorting Genetic Algorithm (NSGA-III) method for MOO and classical approach for Multi-Objective-Optimization (MOO) in regression-based approach showed that NSGA-III has significant advantages over classical approaches because it provides a full pareto-front in one problem solution. The presented methods may be extended by including additional sequence parameters and image quality goals, providing a flexible tool for optimizing MR image sequences for different clinical needs.

5. Summary

Magnetic Resonance Imaging (MRI) is widely used in oncology for tumor staging, treatment response assessment, and radiation therapy (RT) planning. However, optimization of MRI sequences for specific clinical needs is complex and very time-consuming due to the large number of parameter settings. This study proposes two different frameworks for the automatic optimization of MRI sequences addressing two clinical use cases in RT planning based on the sequence parameter sets (SPS): I) a regression-based optimization and II) an on-the-run optimization. A phantom with 7 in-house fabricated contrasts was used for measurements. In the regression-based optimization, two prediction models, i) a Generalized Additive Model (GAM), and ii) a Deep Learning (DL) model, were implemented based on a large number of acquired datasets. In contrast, the on-the-run optimization of the SPS was applied directly on the MR scanner using the interface Access-i. Both frameworks used a derivative-free optimization algorithm to iteratively update a parameterized sequence based on the prediction model or on the use of the MR scanner. In each iteration, the mean squared error (MSE) was calculated. Two clinically relevant optimization goals were pursued: achieving the same contrast as in a target image and maximizing the contrast between specified tissue types. Both goals were evaluated using two optimization methods: a covariance matrix adaptation evolution strategy (CMA-ES) and a genetic algorithm (GA). The obtained results demonstrated the potential of the framework for automatic contrast optimization of MRI sequences. Both CMA-ES and GA methods showed promising results in achieving the two optimization goals; however, CMA-ES converged much faster compared to GA. The proposed frameworks enable fast automatic contrast optimization of MRI sequences based on SPS and may be used to enhance the quality of MRI images for dedicated applications in MR-guided RT.

6. Zusammenfassung

Die Magnetresonanztomographie (MRT) wird in der Onkologie häufig für das Tumor-Staging, die Beurteilung des Therapieerfolgs und die Planung der Radiotherapie (RT) eingesetzt. Die Optimierung von MRT-Sequenzen für spezifische klinische Anforderungen ist jedoch aufgrund der großen Anzahl von Parametereinstellungen komplex und sehr zeitaufwändig. In dieser Studie werden zwei verschiedene Verfahren für die automatische Optimierung von MRT-Sequenzen für zwei klinische Anwendungsfälle in der RT-Planung basierend auf den Sequenz-Parametersätzen (SPS) vorgeschlagen: I) eine regressionsbasierte Optimierung und II) eine On-the-Run-Optimierung. Für die Messungen wurde ein Phantom mit 7 selbst hergestellten Kontrasten verwendet. Bei der regressionsbasierten Optimierung wurden zwei Vorhersagemodelle, i) ein verallgemeinertes additives Modell (GAM) und ii) ein Deep-Learning-Modell (DL), auf Basis einer großen Anzahl von aufgenommenen Datensätzen implementiert. Die On-the-Run-Optimierung der SPS wurde dagegen direkt auf dem MR-Scanner über die Access-i Schnittstelle durchgeführt. Beide Methoden verwendeten einen ableitungsfreien Optimierungsalgorithmus, um eine parametrisierte Sequenz basierend auf dem Vorhersagemodell oder der direkten Verwenduung des MR-Scanner iterativ zu aktualisieren. In jeder Iteration wurde der mittlere quadratische Fehler (MSE) berechnet. Es wurden zwei klinisch relevante Optimierungsziele verfolgt: Das Erreichen des gleichen Kontrasts wie in einem Zielbild und die Maximierung des Kontrasts zwischen bestimmten Gewebetypen. Beide Ziele wurden mit zwei Optimierungsmethoden bewertet: einer Covariance Matrix Adaptation Evolution Strategy (CMA-ES) und einem genetischen Algorithmus (GA). Die erzielten Ergebnisse zeigten das Potenzial der Methode für die automatische Kontrastoptimierung von MRT-Sequenzen. Sowohl die CMA-ES- als auch die GA-Methode zeigten vielversprechende Ergebnisse beim Erreichen der beiden Optimierungsziele, wobei die CMA-ES-Methode im Vergleich zur GA-Methode wesentlich schneller konvergierte. Die vorgeschlagene Methode ermöglicht die schnelle automatische Kontrastoptimierung von MRT-Sequenzen auf der Grundlage von SPS und kann zur Verbesserung der Qualität von MRT-Bildern für spezielle Anwendungen in der MR-geführten RT eingesetzt werden.

7. Bibliography

[1] Aker M, Boone D, Chandramohan A, Sizer B, Motson R, Arulampalam T. Diagnostic accuracy of MRI in assessing tumor regression and identifying complete response in patients with locally advanced rectal cancer after neoadjuvant treatment. Abdominal Radiology. 2018;43:3213–9.

[2] Horvat N, Carlos Tavares Rocha C, Clemente Oliveira B, Petkovska I, Gollub MJ. MRI of rectal cancer: tumor staging, imaging techniques, and management. Radiographics. 2019;39:367–87.

[3] Kalisz KR, Enzerra MD, Paspulati RM. MRI evaluation of the response of rectal cancer to neoadjuvant chemoradiation therapy. Radiographics. 2019;39:538–56.

[4] Michael J, Neuzil K, Altun E, Bjurlin MA. Current Opinion on the Use of Magnetic Resonance Imaging in Staging Prostate Cancer: A Narrative Review. Cancer Management and Research. 2022;14:937.

[5] Nascimento D, Suchard G, Hatem M, de Abreu A. The role of magnetic resonance imaging in the evaluation of bone tumours and tumour-like lesions. Insights into imaging. 2014;5:419–40.

[6] Layton KJ, Kroboth S, Jia F, Littin S, Yu H, Leupold J, et al. Pulseq: a rapid and hardware-independent pulse sequence prototyping framework. Magnetic resonance in medicine. 2017;77:1544–52.

[7] Loktyushin A, Herz K, Dang N, Glang F, Deshmane A, Weinmüller S, et al. MRzero-Automated discovery of MRI sequences using supervised learning. Magnetic Resonance in Medicine. 2021;86:709–24.

[8] Glang F, Mueller S, Herz K, Loktyushin A, Scheffler K, Zaiss M. MR-double-zero–proof-of-concept for a framework to autonomously discover MRI contrasts. Journal of Magnetic Resonance. 2022:107237.

[9] Asna N, Livoff A, Batash R, Debbi R, Schaffer P, Rivkind T, et al. Radiation therapy and immunotherapy—a potential combination in cancer treatment. Current Oncology. 2018;25(5):454-60.

[10] Wookey H, Ash C, Welsh WK, Mustard R. The treatment of oral cancer by a combination of radiotherapy and surgery. Annals of Surgery. 1951;134(4):529.

[11] Northover J, Meadows H, Ryan C, Gray R. Combined radiotherapy and chemotherapy for anal cancer. The Lancet. 1997;349(9046):205-6.

[12] Bortfeld PDT, Bendl R, Dams J, Dobler B, Gölz S, Grosser KH, et al. Division Medical Physics (E0400).

[13] Yuan T-Z, Zhan Z-J, Qian C-N. New frontiers in proton therapy: applications in cancers. Cancer Communications. 2019;39(1):1-7.

[14] Chen Z, Dominello MM, Joiner MC, Burmeister JW. Proton versus photon radiation therapy: A clinical review. Frontiers in Oncology. 2023;13:1133909.

[15] Gozbasi HO. Optimization approaches for planning external beam radiotherapy: Georgia Institute of Technology; 2010.

[16] Richter C, Wohlfahrt P. Dual-Energy CT in Radiation Oncology. Spectral Imaging: Dual-Energy, Multi-Energy and Photon-Counting CT: Springer; 2022. p. 333-46.

[17] Chui C-S, Spirou SV. Inverse planning algorithms for external beam radiation therapy. Medical Dosimetry. 2001;26(2):189-97.

[18] Cooper JS, Fu K, Marks J, Silverman S. Late effects of radiation therapy in the head and neck region. International Journal of Radiation Oncology* Biology* Physics. 1995;31(5):1141-64.

[19] Brahme A. Development of radiation therapy optimization. Acta oncologica. 2000;39(5):579-95.

[20] Wambersie A, Landberg T, Chavaudra J, Dobbs J, Hanks G, Johansson K, et al. Prescribing, recording, and reporting photon beam therapy presentation of the ICRU report# 50. Journal of Medical Physics. 1992;17(4):5-7.

[21] Stroom JC, Heijmen BJ. Geometrical uncertainties, radiotherapy planning margins, and the ICRU-62 report. Radiotherapy and oncology. 2002;64(1):75-83.

[22] Thieke C. Gross tumor volume (GTV) Basics, methods, registration, limitations. Der Radiologe. 2018;58:722-9.
[23] Berthelsen AK, Dobbs J, Kjellén E, Landberg T, Möller TR, Nilsson P, et al. What's new in target volume definition for radiologists in ICRU Report 71? How can the ICRU volume definitions be integrated in clinical practice? Cancer Imaging. 2007;7(1):104.
[24] Fisher BJ, Daugherty LC, Reiff JE. Planning Target Volume (PTV). In: Brady LW, Yaeger TE, editors. Encyclopedia of Radiation Oncology10.1007/978-3-540-85516-3_300. Berlin, Heidelberg: Springer Berlin Heidelberg; 2013. p. 624-5.
[25] Berberoğlu K. Use of Positron Emission Tomography/Computed Tomography in radiation treatment planning for lung cancer. Molecular imaging and radionuclide therapy. 2016;25(2):50.
[26] Burnet NG, Noble D, Paul A, Whitfield GA, Delorme S. Target volume concepts in radiotherapy and their implications for imaging. Der Radiologe. 2018;58:708-21.
[27] Pereira GC, Traughber M, Muzic RF. The role of imaging in radiation therapy planning: past, present, and future. BioMed research international. 2014;2014.
[28] Suarez-Gironzini V, Khoo V. Imaging advances for target volume definition in radiotherapy. Current Radiology Reports. 2015;3:1-9.
[29] Jain V, Soni TP, Singh DK, Patni N, Jakhotia N, Gupta AK, et al. A prospective study to assess and quantify the setup errors with cone-beam computed tomography in head-and-neck cancer image-guided radiotherapy treatment. Journal of Cancer Research and Therapeutics. 2023;19(3):783-7.
[30] Yan D, Vicini F, Wong J, Martinez A. Adaptive radiation therapy. Physics in Medicine & Biology. 1997;42(1):123.
[31] Moteabbed M, Smeets J, Hong TS, Janssens G, Labarbe R, Wolfgang JA, et al. Toward MR-integrated proton therapy: modeling the potential benefits for liver tumors. Physics in Medicine & Biology. 2021;66(19):195004.
[32] Surla S. Dosimetry of Carbon-Ion Reference and Small Photon Fields for Magnetic Resonance-Guided Radiation Therapy 2024.
[33] Westley RL, Valle LF, Tree AC, Kishan AU, editors. MRI-Guided Radiotherapy for Prostate Cancer: Seeing is Believing. Seminars in Radiation Oncology; 2024: Elsevier.
[34] Klüter S. Technical design and concept of a 0.35 T MR-Linac. Clin Transl Radiat Oncol 2019; 18: 98–101.
[35] Bloch F. Nuclear induction. Physical review. 1946;70(7-8):460.
[36] Lauterbur PC. Image formation by induced local interactions: examples employing nuclear magnetic resonance. nature. 1973;242(5394):190-1.
[37] Kagawa T, Yoshida S, Shiraishi T, Hashimoto M, Inadomi D, Sato M, et al. Basic principles of magnetic resonance imaging for beginner oral and maxillofacial radiologists. Oral Radiology. 2017;33:92-100.
[38] Weishaupt D, Köchli VD, Marincek B. Wie funktioniert MRI?: eine Einführung in Physik und Funktionsweise der Magnetresonanzbildgebung: Springer; 2009.
[39] Ma JJ, Nakarmi U, Kin CYS, Sandino CM, Cheng JY, Syed AB, et al., editors. Diagnostic image quality assessment and classification in medical imaging: Opportunities and challenges. 2020 IEEE 17th International Symposium on Biomedical Imaging (ISBI); 2020: IEEE.
[40] Soto ME, Pezoa JE, Torres SN, editors. Thermal noise estimation and removal in MRI: a noise cancellation approach. Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications: 16th Iberoamerican Congress, CIARP 2011, Pucón, Chile, November 15-18, 2011 Proceedings 16; 2011: Springer.
[41] Simkó A, Ruiter S, Löfstedt T, Garpebring A, Nyholm T, Bylund M, et al. Improving MR image quality with a multi-task model, using convolutional losses. BMC Medical Imaging. 2023;23(1):148.
[42] Kilmer JT, Rodríguez RL. Ordinary least squares regression is indicated for studies of allometry. Journal of evolutionary biology. 2017;30(1):4-12.
[43] Servén D, Brummitt C. pyGAM Documentation. 2020.

[44] Hong X, Mitchell RJ, Chen S. Modelling and control of Hammerstein system using B-spline approximation and the inverse of De Boor algorithm. International Journal of Systems Science. 2012;43(10):1976-84.
[45] Ma X, Shen W. Generalized de Boor–Cox formulas and pyramids for multi-degree spline basis functions. Mathematics. 2023;11(2):367.
[46] Thrun S, Montemerlo M, Dahlkamp H, Stavens D, Aron A, Diebel J, et al. Stanley: The robot that won the DARPA Grand Challenge. Journal of field Robotics. 2006;23(9):661-92.
[47] Wu Y, Schuster M, Chen Z, Le QV, Norouzi M, Macherey W, et al. Google's neural machine translation system: Bridging the gap between human and machine translation. arXiv preprint arXiv:160908144. 2016.
[48] Girshick R, Donahue J, Darrell T, Malik J, editors. Rich feature hierarchies for accurate object detection and semantic segmentation. Proceedings of the IEEE conference on computer vision and pattern recognition; 2014.
[49] Russakovsky O, Deng J, Su H, Krause J, Satheesh S, Ma S, et al. Imagenet large scale visual recognition challenge. International journal of computer vision. 2015;115:211-52.
[50] Biratu ES, Schwenker F, Ayano YM, Debelee TG. A survey of brain tumor segmentation and classification algorithms. Journal of Imaging. 2021;7(9):179.
[51] Zhu Y, Kim M, Zhu X, Yan J, Kaufer D, Wu G, editors. Personalized diagnosis for Alzheimer's disease. Medical Image Computing and Computer Assisted Intervention− MICCAI 2017: 20th International Conference, Quebec City, QC, Canada, September 11-13, 2017, Proceedings, Part III 20; 2017: Springer.
[52] Li Q, Cai W, Wang X, Zhou Y, Feng DD, Chen M, editors. Medical image classification with convolutional neural network. 2014 13th international conference on control automation robotics & vision (ICARCV); 2014: IEEE.
[53] Suzuki K. Overview of deep learning in medical imaging. Radiological physics and technology. 2017;10(3):257-73.
[54] McCelloch W, Pitts W. A logical calculus of the idea immanent in neural nets. Bulletin ofMathematical Biophysics. 1943;5:115-33.
[55] Lederer J. Activation functions in artificial neural networks: A systematic overview. arXiv preprint arXiv:210109957. 2021.
[56] Dertat A. Applied deep learning–part 1: artificial neural networks. Towards data science. 2017.
[57] Kyurkchiev N, Markov S. Sigmoid functions: some approximation and modelling aspects. LAP LAMBERT Academic Publishing, Saarbrucken. 2015;4.
[58] Dubey SR, Singh SK, Chaudhuri BB. Activation functions in deep learning: A comprehensive survey and benchmark. Neurocomputing. 2022.
[59] Dubey AK, Jain V, editors. Comparative study of convolution neural network's relu and leaky-relu activation functions. Applications of Computing, Automation and Wireless Systems in Electrical Engineering: Proceedings of MARC 2018; 2019: Springer.
[60] Bhalley R. Deep Learning with Swift for TensorFlow: Differentiable Programming with Swift: Springer; 2021.
[61] Liu Y. Amodal Instance Segmentation and Multi-Object Tracking with Deep Pixel Embedding. 2019.
[62] Maier HR, Razavi S, Kapelan Z, Matott LS, Kasprzyk J, Tolson BA. Introductory overview: Optimization using evolutionary algorithms and other metaheuristics. Environmental modelling & software. 2019;114:195-213.
[63] Ruder S. An overview of gradient descent optimization algorithms. arXiv preprint arXiv:160904747. 2016.
[64] Kochenderfer MJ, Wheeler TA. Algorithms for optimization: Mit Press; 2019.
[65] Datta R, Deb K. Evolutionary constrained optimization: Springer; 2014.
[66] Bertsekas DP. Constrained optimization and Lagrange multiplier methods: Academic press; 2014.

[67] Hansen N, Müller SD, Koumoutsakos P. Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (CMA-ES). Evolutionary computation. 2003;11(1):1-18.
[68] Varelas K, Auger A, Brockhoff D, Hansen N, ElHara OA, Semet Y, et al., editors. A comparative study of large-scale variants of CMA-ES. Parallel Problem Solving from Nature–PPSN XV: 15th International Conference, Coimbra, Portugal, September 8–12, 2018, Proceedings, Part I 15; 2018: Springer.
[69] Hansen N, Auger A, editors. CMA-ES: evolution strategies and covariance matrix adaptation. Proceedings of the 13th annual conference companion on Genetic and evolutionary computation; 2011.
[70] Arnold DV, Hansen N, editors. A (1+ 1)-CMA-ES for constrained optimisation. Proceedings of the 14th annual conference on Genetic and evolutionary computation; 2012.
[71] Agarwal R. Evaluation Function Approximation for Scrabble. arXiv preprint arXiv:190108728. 2019.
[72] Sivanandam S, Deepa S, Sivanandam S, Deepa S. Genetic algorithm optimization problems. Introduction to genetic algorithms. 2008:165-209.
[73] Yadav PK, Prajapati N. An overview of genetic algorithm and modeling. International Journal of Scientific and Research Publications. 2012;2(9):1-4.
[74] Ansari MSA, Pal K, Govil MC. Revisiting of peer-to-peer traffic: taxonomy, applications, identification techniques, new trends and challenges. Knowledge and Information Systems. 2023:1-58.
[75] Pereira JLJ, Oliver GA, Francisco MB, Cunha Jr SS, Gomes GF. A review of multi-objective optimization: methods and algorithms in mechanical engineering problems. Archives of Computational Methods in Engineering. 2022;29(4):2285-308.
[76] Mosavi A. Applications of interactive methods of MOO in chemical engineering problems. Global Journals of Engineering Research. 2010;10(3):08-15.
[77] Petchrompo S, Coit DW, Brintrup A, Wannakrairot A, Parlikad AK. A review of Pareto pruning methods for multi-objective optimization. Computers & Industrial Engineering. 2022;167:108022.
[78] Zitzler E. Evolutionary algorithms for multiobjective optimization: Methods and applications: Shaker Ithaca; 1999.
[79] Murata T, Ishibuchi H, Tanaka H. Multi-objective genetic algorithm and its applications to flowshop scheduling. Computers & industrial engineering. 1996;30:957–68.
[80] Seshadri A. A fast elitist multiobjective genetic algorithm: NSGA-II. MATLAB Central. 2006;182:182-97.
[81] Yuan Y, Xu H, Wang B, editors. An improved NSGA-III procedure for evolutionary many-objective optimization. Proceedings of the 2014 annual conference on genetic and evolutionary computation; 2014.
[82] Deb K, Pratap A, Agarwal S, Meyarivan T. A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE transactions on evolutionary computation. 2002;6(2):182-97.
[83] Al-Amri SS, Kalyankar NV. Image segmentation by using threshold techniques. arXiv preprint arXiv:10054020. 2010.
[84] Farag AA. Edge-based image segmentation. Remote sensing reviews. 1992;6(1):95-121.
[85] Dhanachandra N, Manglem K, Chanu YJ. Image segmentation using K-means clustering algorithm and subtractive clustering algorithm. Procedia Computer Science. 2015;54:764-71.
[86] Bose S, O'sullivan F. A region-based segmentation method for multichannel image data. Journal of the American Statistical Association. 1997;92(437):92-106.
[87] Elter A, Hellwich E, Dorsch S, Schäfer M, Runz A, Klüter S, et al. Development of phantom materials with independently adjustable CT-and MR-contrast at 0.35, 1.5 and 3 T. Physics in Medicine & Biology. 2021;66:045013.

[88] Fahad HM, Dorsch S, Zaiss M, Karger CP. Multi-parametric optimization of magnetic resonance imaging sequences for magnetic resonance-guided radiotherapy. Physics and Imaging in Radiation Oncology. 2023;28:100497.
[89] Fallone BG, editor The rotating biplanar linac–magnetic resonance imaging system. Seminars in radiation oncology; 2014.
[90] Raaymakers BW, Lagendijk JJW, Overweg J, Kok JGM, Raaijmakers AJE, Kerkhof EM, et al. Integrating a 1.5 T MRI scanner with a 6 MV accelerator: proof of concept. Physics in Medicine & Biology. 2009;54:N229.
[91] Servn D, Brummitt C. pygam: Generalized additive models in python, 2018. URL: https://doi org/105281/zenodo.1208723.
[92] Ravi KS, Geethanath S, Vaughan JT. PyPulseq: A python package for mri pulse sequence design. Journal of Open Source Software. 2019;4(42):1725.
[93] Bagirov AM, Karasözen B, Sezer M. Discrete gradient method: derivative-free method for nonsmooth optimization. Journal of Optimization Theory and Applications. 2008;137:317-34.
[94] Grimm V, McLachlan RI, McLaren DI, Quispel G, Schönlieb C. Discrete gradient methods for solving variational image regularisation models. Journal of Physics A: Mathematical and Theoretical. 2017;50(29):295201.
[95] McLachlan RI, Quispel G. Discrete gradient methods have an energy conservation law. arXiv preprint arXiv:13024513. 2013.
[96] Quispel G, Turner GS. Discrete gradient methods for solving ODEs numerically while preserving a first integral. Journal of Physics A: Mathematical and General. 1996;29(13):L341.
[97] Dang HN, Golkov V, Wimmer T, Cremers D, Maier A, Zaiss M. Joint MR sequence optimization beats pure neural network approaches for spin-echo MRI super-resolution. arXiv preprint arXiv:230507524. 2023.
[98] Hoinkiss DC, Huber J, Plump C, Lüth C, Drechsler R, Günther M. AI-driven and automated MRI sequence optimization in scanner-independent MRI sequences formulated by a domain-specific language. Frontiers in Neuroimaging. 2023;2:1090054.
[99] Koolstra K, O'Reilly T, Börnert P, Webb A. Image distortion correction for MRI in low field permanent magnet systems with strong B 0 inhomogeneity and gradient field nonlinearities. Magnetic Resonance Materials in Physics, Biology and Medicine. 2021:1-12.
[100] Cooley CZ, Stockmann JP, Armstrong BD, Sarracanie M, Lev MH, Rosen MS, et al. Two-dimensional imaging in a lightweight portable MRI scanner without gradient coils. Magnetic resonance in medicine. 2015;73(2):872-83.
[101] Dang V-H, Vien NA, Chung T. A covariance matrix adaptation evolution strategy in reproducing kernel Hilbert space. Genetic Programming and Evolvable Machines. 2019;20:479-501.

8. Personal Contributions

This work has received funding from the German Federal Ministry of Education and Research (BMBF) within the funding program “Bildgeführte Diagnostik und Therapie – Neue Wege in der Intervention” for the ARTEMIS research project.

Journal Articles:

Publication: Fahad, H.M., Dorsch, S., Zaiss, M. and Karger, C.P.: Multi-parametric optimization of magnetic resonance imaging sequences for magnetic resonance-guided radiotherapy. Physics and Imaging in Radiation Oncology, 2023, 28, p.100497

Publication is based on the method described in chapters 2.7.1 and 2.7.3 of this dissertation. this publication is based on the results described in chapters 3.3 of this dissertation. The discussion related to proposed workflow (Chapter 4.2, 4.3, 4.4 and 4.5) is also depicted in this publication. My personal contribution to this publication consisted of: Conceptualization, Methodology, Software, Visualization, Investigation, Writing – original draft.

Conference Contributions

1. **Fahad, H.M.**, Dorsch, S., Zaiss, M. and Karger, C.P.: “Multi-parametric optimization of magnetic resonance imaging sequences for magnetic resonance-guided radiotherapy”. SGSMP, ÖGMP, DGMP, digital congress, 2021 (poster presentation)

2. **Fahad, H.M.**, Dorsch, S., Zaiss, M. and Karger, C.P.: “Multi-parametric optimization of magnetic resonance imaging sequences for magnetic resonance-guided radiotherapy”. ESTRO Conference 2022, Copenhagen, Denmark, (poster presentation)

3. **Fahad, H.M.**, Dorsch, S., Zaiss, M. and Karger, C.P.: “Multi-parametric optimization of magnetic resonance imaging sequences for magnetic resonance-guided radiotherapy”. DKFZ PhD poster presentation, Heidelberg, 2022 (poster presentation)

Curriculum vitue

Personal Information

Name	Hafiz Muhammad Fahad
Date of birth	03.12.1992
Place of birth	Lahore
Nationality	Pakistani

Education

2020 – present	German Cancer Research Center (dkfz.) & Medical Faculties of Heidelberg University **Ph.D. Researcher** **Project title:** "multi-parametric optimization of magnetic resonance imaging sequences for magnetic resonance-guided radiotherapy"
2016 – 2020	Ruprechts-Karls-Universität Heidelberg, Heidelberg, Germany **Master Thesis Topic:** "Mono and Multi-Modal Diffeomorphic Medical Image Registration using Deep learning" **Degree**: Master of Science (M.Sc.) in Scientific Computing
2011 – 2015	University of the Punjab, Lahore, Pakistan **Bachelor Thesis Topic:** "Data science analysis for high energy field data using ROOT software" **Degree**: Bachelor of Science (B.Sc.) in Computational Physics

Research Experience:

Sep 2023- present	Research Scientist at Ludwig Maximilian University of Munich Faculty of Physics
Jan – Oct 2018	Research Assistant at Visual Computing Group, Heidelberg University Faculty for Mathematics and Informatics
Jan- May 2018	Research Assistant at Heidelberg Collaboratory for Image Processing (HCI), Heidelberg University Faculty for Mathematics and Informatics
Jan- June 2017	Research Assistant at Mathematical Methods of Simulation group, Heidelberg University Faculty for Mathematics and Informatics

Acknowledgements

I extend my heartfelt gratitude to all those who have supported me throughout this journey. Their influence has been invaluable.

First and foremost, I express sincere appreciation to my thesis supervisors, Prof. Dr. Christian Karger, for unwavering support, patience, and guidance throughout my research. Additionally, I am grateful to my advisor, Dr. Stefan Dorsch, whose expertise and mentorship have been a guiding light. His patience during challenging times has taught me resilience and strength of character. He helped me translate thoughts into words and ideas into practice.

I also acknowledge Prof. Dr. Oliver Jäkel, Head of the Department of Medical Physics in Radiation Oncology at DKFZ, for his support.

I am grateful for the whole department E040 for the nice working atmosphere and the colleagues who became friends. Thanks, my E0402 group, for all the activities outside the office. And thank you all who helped with proofreading my thesis!

My parents have been unwavering pillars of strength, accompanying me through every frenzied step of this journey. Special thanks to my wife and son, whose unwavering love and emotional support have been my driving force. Their belief in my abilities pushed me beyond my limits, achieving what once seemed impossible.

Lastly, I am indebted to individuals like Sarmad Ullah, Dr. Noman Shakir, Malik Majeed, Dr. Kafeel Sarwar, Farhan Rasheed, and Muhammad Umair. Their unwavering assistance helped me navigate tough days and overcome my weakest moments. Although I cannot name everyone, their guidance was pivotal in completing this project. Thank you all for being just a phone call or appointment away. My heart is filled with gratitude.

Eidesstattliche Versicherung

1. Bei der eingereichten Dissertation zu dem Thema **Multi-parametric optimization of magnetic resonance-imaging sequences for magnetic resonance-guided radiotherapy** handelt es sich um meine eigenständig erbrachte Leistung.

2. Ich habe nur die angegebenen Quellen und Hilfsmittel benutzt und mich keiner unzulässigen Hilfe Dritter bedient. Insbesondere habe ich wörtlich oder sinngemäß aus anderen Werken übernommene Inhalte als solche kenntlich gemacht.

3. Die Richtigkeit der vorstehenden Erklärungen bestätige ich.

4. Die Bedeutung der eidesstattlichen Versicherung und die strafrechtlichen Folgen einer unrichtigen oder unvollständigen eidesstattlichen Versicherung sind mir bekannt. Ich versichere an Eides statt, dass ich nach bestem Wissen die reine Wahrheit erklärt und nichts verschwiegen habe.

Eppelheim, 14.03.20224 _____