Finding the perfect MRI sequence for your patient — Towards an optimisation workflow for MRI-sequences

Christina Plump*, Daniel C. Hoinkiss†, Jörn Huber†, Bernhard J. Berger‡§, Matthias Günther†¶, Christoph Lüth*¶ and Rolf Drechsler*¶

* *DFKI GmbH—Cyber-Physical Systems*
Bibliotheksstraße 5
28359 Bremen, Germany

† *Fraunhofer MeVis*
Max-von-Laue-Str. 2
Bremen, Germany

‡ *Hamburg University of Technology*
Am Schwarzenberg 1
21071 Hamburg, Germany

§ *University of Rostock*
18051 Rostock, Germany

¶ *University of Bremen*
Bibliotheksstraße 5
28359 Bremen, Germany

***Abstract*—Magnetic Resonance Imaging (MRI) is an essential tool for medical diagnosis. At the same time, its usage requires profound expert knowledge to determine the ideal MR sequence and protocol to be run. Until now, the contrast and quality of the resulting image have relied mainly on the radiologist's expertise. When confronted with clinical requirements and patient information, the radiologist chooses suitable sequence protocols for the examination. We propose a workflow that supports medical personnel in finding the optimal sequence for a given diagnostic task. To that end, we combine evolutionary algorithms for the optimisation, machine learning techniques for training a surrogate optimisation function from simulated MRI data, and domain-specific languages to allow non-programmers to formulate their requirements and constraints semi-formally. In this paper, we focus on the efficient usage of real-world application-motivated adaptions of the used evolutionary algorithm and evaluate their effects on four real-life sequence examples. We show that it is essential to use an adaption for the surrogate model to obtain realistic solutions and use correlation information about the search space to stay in feasible areas of the search space and thus improve optimisation quality. These findings are a first step in automating the entire MRI-sequence optimisation flow, which is necessary to allow a more widespread usage of this essential medical diagnostic technique.**

***Index Terms*—Evolutionary Algorithms, Surrogate Model, Constraint Handling, Medical Application, Domain-Specifc Language**

I. Introduction

Magnetic Resonance Imaging (MRI) is one of the most versatile modern imaging modalities in clinical routine. The important role arises from the ability to create a large variety of different image contrasts using so-called MRI sequences, which are computer programs that orchestrate the different hardware components of the MRI machine to yield the best contrast and image geometries for the specific application [1]. This flexibility allows MRI to both structurally and quantitatively separate healthy and malignant tissue and gives insights into the physiology of the patient. MRI is a non-invasive technique that offers accurate diagnosis without harming the patient, which is an additional benefit over other commonly used imaging modalities such as X-rays. Unfortunately, great flexibility often comes with great complexity, which also holds for MRI for the following reasons: First, implementing novel MRI sequences is usually a complex task, which requires a high amount of programming skills and thus is only possible for highly trained experts. Ongoing research, therefore, focussed on the development of tools for optimisation for specific aspects of sequence programming (see, e.g. [2] [3] [4]). Second, even if a specific MRI sequence is available, the correct adjustment of related sequence parameters, which define different timings of the pre-built sequence structure, usually requires months or years of expertise. As a consequence, a lack of trained operators can hinder the successful application of MRI in, e.g., emerging countries where many people would benefit from better diagnostics.

The current process in clinical day-to-day is depicted in Figure 1: The physician issues an MRI request to help diagnose a patient. This MRI request (along with the patient) is then transferred to a radiologist who has to assess the physician's request as well as the patient's condition (physical and mental) to choose the MRI sequence and protocol best suited for this combination in his experience. The radiologist then chooses one of the available sequences and adjusts the protocol to the best of his ability. This is then run on the MRI machine.

This poses two main issues: First, the restriction to the available sequences which might not be a perfect fit for the situation at hand and second, the extensive expertise that is required by the radiologist to fulfill the request as well as adapt to the patient's condition.

In [5], a general workflow of predicting image metrics from sequence description after training an ML model and

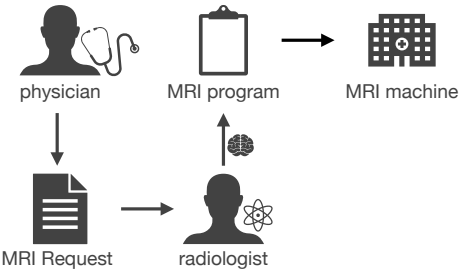

Fig. 1: Current Sequence Determination Process

subsequent optimisation was described and already showed promising results. We extend the work of Hoinkiss et al. in several ways. First, we provide a more detailed analysis of the optimisation techniques, identifying which of the used features are relevant to the improved optimisation results and if so, how this comes to pass. Second, we extend the workflow with two knowledge-based automation routines: One, to automatically translate the MRI request into a configuration for an evolutionary algorithm and second, an automatic repairing of the final optimisation results, which translates back to an MRI sequence that can be transformed into an MRI program which will guide the MRI process. While the former contribution improves knowledge about relevant features of the algorithm and might eventually improve performance, the latter contribution pushes the workflow one step further to automating the MRI-imaging process, reducing the required expert knowledge. This contribution helps to keep good medical diagnostics up even in times of reduced skilled workforce and allows good medical diagnostics in countries with a reduced number of available experts.

Our evaluation shows that an adaption technique for the surrogate model is essential to keep the results in a feasible area. Additionally, it shows that depending on population size and mutation rate, it is necessary or at least helpful to use recombination and mutation operators that comply with interdependencies in the data. Our repairing mechanism produces sensible MRI sequences that perform as specified in the optimisation for three out of four main setups.

The remainder of the paper is structured as follows: Section II explains necessary information about the domain, the structure of the search and optimisation space, as well as the intricacies of the optimisation requirements. Then, Section III shows the interaction of our applied techniques and formulates our method, while Section IV comments on the implementation. The setup and results of our evaluation are described in Section V, whereas the overall discussion takes place in Section VI. Section VII then concludes the paper.

II. Problem Description

In this section, we introduce the reader to the terminology, search space description and explanation, as well as optimisation space and description to understand the challenges of the given optimisation problem and our modelling approach.

A. Search Space Description

A typical MRI sequence utilises four different principles. The main magnetic field aligns the proton spins in the body, creating a net magnetization along the magnetic field (through the bore). In addition, the proton spins start to precess around the external magnetic field at the so-called resonance frequency, which is linearly related to the external field strength. Radiofrequency pulses in resonance with the spin precession frequency are used for excitation of the spins and receiver coil elements sample the signal that can be received while the spins tend back to the equilibrium state. The time it takes the protons to return to their equilibrium state depends on their chemical environment and, thus, is an important factor in differentiating tissue types. A set of three coils (x, y, z) that create a gradient magnetic field in each of the three Cartesian directions is used to generate spatial resolution. These help to fill a parameter space called k-space, which can eventually be transformed into the final image using the Fourier Transform.

That said, all MRI sequences could be modelled using temporal functions of the different hardware components: Radiofrequency pulses, (x,y,z)-gradient coils, receiver coil. However, this form of representation has — while simple — a significant drawback. A majority of the so-defined search space is unfeasible in that it constitutes either impossible or insensible sequences. Additionally, it contains no information about the meaning (semantics) of the specific parts of the sequence. Therefore, another representation is necessary.

Instead of looking at five (independent) functions, their physical contribution to the image is considered. For an MRI to work, there are two major components: The echo signal formation, which describes the excitation of the spins and the creation of an MRI signal, and the readout type, which describes the spatial stepping through the k-space. These are typically followed by each other. The echo itself can be prolonged through a refocusing pulse that follows the original excitation to bring back the signal that has been degraded by magnetic field inhomogeneities. The flip angle, resembling the strength of the excitation, describes an important sequence parameter. The most important timing parameters of an MRI sequence are given by TE (echo time) and TR (repetition time), which describe the relative position of excitation, refocusing, and readout to each other. These are essential for acquiring the desired contrast in the image. The readout itself can be specified by its duration and the number of samples received by the coil, which will fill a single line in k-space. Finally, it is important to define the trajectory of the k-space to fill this parameter space by repeating the sequence parts in a looped fashion. In this paper, we restrict ourselves to cartesian trajectories, which fill the k-space line-by-line after a single or multiple excitations. More complex trajectories like spirals or radial sequences will be considered in future work. These cartesian trajectories can be specified through the total number of k-space rows, as well as the echo train length (ETL) and the Epi-factor, which describes the number of rows to be read out after a single excitation.

Search Space Variable	Domain	Opt Space Variable	Domain
numOfRows	$\mathbb{N}_{\leq 512}$	SNR_CSF	$\mathbb{R}_{>=0}$
numOfCols	$\mathbb{N}_{\leq 512}$	SNR_GM	$\mathbb{R}_{>=0}$
ETL	$\mathbb{N}_{\leq 512}$	SNR_WM	$\mathbb{R}_{>=0}$
Epi-Factor	$\mathbb{N}_{\leq 512}$	GWC	$\mathbb{R}_{>=0}$
Refocusing Angle	$\mathbb{N}_{\leq 360}$	CGC	$\mathbb{R}_{>=0}$
TE	$\mathbb{N}$	CWC	$\mathbb{R}_{>=0}$
TR	$\mathbb{N}_0$	Ghost	$[0,1] \in \mathbb{R}$
readout duration	$\mathbb{N}_0$	Sharpness	$[0,1] \in \mathbb{R}$
measurements	$\mathbb{N}_0$	Homogeneity	$[0,1] \in \mathbb{R}$
prescans	$\mathbb{N}_0$	Motion.Sens	$[0,1] \in \mathbb{R}$
reference lines	$\mathbb{N}_0$	Distortion	$[0,1] \in \mathbb{R}$
		acquisition	$\mathbb{R}_{>=0}$

TABLE I: Description of search and optimisation space variables of the optimisation problem, their domain-related name as well as their boundaries.

The above explanation is a very simple description of the MRI search space. Additional parameters have to be considered, like, for example, the number of prescans (to guarantee a steady state of the magnetisation before receiving data) or the number of total measurements (for task-related imaging or to increase signal-to-noise ratio). Also, some MRI acceleration techniques can be used to subsample the k-space. Missing lines are then generated by utilising information, which is extracted from additional multi-coil reference data. Whether such parameters deviate from the default value, is highly dependent on the chosen type of sequence.

All of the above-mentioned information can be structured in a search space consisting of mainly integer values. A sequence is thus — for this work — defined by a vector $\mathbf{x} \in \mathbb{N}^{11} = \mathcal{S}$ containing the variables (together with their boundaries) shown in the left part of Table I.

Several of these variables are dependent on each other, however not in a strict way that could be expressed by a constraint. It can rather be described as a correlation. For example, low TE values are more likely to occur with low TR values than with high TR values.

B. Optimisation Space Description

The physician has — depending on the diagnostic task — different demands regarding the properties of the resulting image: First, the contrast of gray matter and white matter to the main image as well as to one another is of interest. Usually, high contrast is preferred. Second, the signal-to-noise ratio in Cerebrospinal Fluid (CSF), as well as both gray and white matter, is essential. All six values are real-valued and non-negative. Third, usually, the absence/presence of ghost artefacts is relevant. Ghost artefacts are shadow copies of the actual image that occur at different places in the image. Fourth, the image's sharpness and homogeneity are relevant to the physician. A measure that is directly linked to the patient is motion sensitivity: In principle, the patient should lie still; however, often, this is impossible. Motion sensitivity describes how reactive the image quality is to the patient's motion. A comparable measure is distortion sensitivity. These last five values are all real-valued as well and lie between 0 and 1. Finally, an essential factor is the acquisition time, i.e. how long it takes to complete the examination. Some patients may not be able to lie in the MR tube for longer than a few seconds. Then, the sequence definition produces a quick examination. All in all, this leads to an optimization space (with variables that may be subject to optimisation or constraints for optimisation) of $\mathcal{O} = \mathbb{R}^{12}_{>=0}$, with the above-mentioned restrictions (see the right part of Table I).

C. Optimisation Task

Before we finally describe the optimisation task, we comment on the functional relationship between $\mathcal{S}$ and $\mathcal{O}$. Obviously, the optimisation in $\mathcal{O}$ results from variations in $\mathcal{S}$. However, no explicit function

$$f : \mathcal{S} \rightarrow \mathcal{O} \tag{1}$$

exists. As is usual in this kind of situation, we fall back to a surrogate model (see [6] for exists. As is usual in this kind of situation, we fall back to a surrogate model (see [6] for an extensive survey on types of surrogate models and forms of usage). We therefore use the trained function $\tilde{f}_D : \mathcal{S} \rightarrow \mathcal{O}$ based on a set of training data $\mathcal{D}$, as has been discussed in [5]. Hoinkiss et al. extensively describe their decision process for the model as well as the analysis of the training data. Additionally, they state goodness-of-fit values like the mean squared error (MSE), which allows for the informed usage of their model. Using a surrogate model increases the difficulty of the optimisation task significantly due to the following reasons: First, $\tilde{f}_D$ is error-prone, even when well-trained. Second, its quality is highly dependent on the training data D. Third, it decreases the chances of applying gradient-based optimisation techniques.

We will now discuss the structure of potential optimisation tasks. In general, the physician needs the $i-th$ of the optimisation space components to be maximised or minimised, i.e.

$$\mathcal{S} \ni \mathbf{x}_{opt} = \arg\max_{\mathbf{x}}(y_i) \tag{2}$$

where y_i denotes the i-th component of $\mathcal{O} \ni \mathbf{y} = \mathbf{f}(\mathbf{x})$ and $\arg\max$ may be replaced with $\arg\min$ depending on the optimisation task. In [5], it was stated that this main optimisation is usually accompanied by further restrictions, however, usually on the optimisation space rather than the search space. These restrictions can be strict, i.e. must not be violated, or tendencies, i.e. should not be violated. Hence,besides usual constraints of the form

$$cs_j(x_1, x_2, ..., x_s) \leq 0 \ , \ 1 \leq j \leq m, s = dim(\mathcal{S}) \tag{3}$$

which restrict the search space, we additionally have constraints of the form:

$$co_k(y_1, ..., y_o) \leq 0 \ \ 1 \leq k \leq l, o = dim(\mathcal{O}) \tag{4}$$

cs stands for *constraint-searchspace* and co for *constraint-optspace*. Please note that the form of constraint is the general form, which includes expressions like $ETL \leq 512 \Rightarrow ETL - 512 \leq 0$. Remember that some constraints may denote tendencies, e.g. avoiding ghost artefacts, that need not be be

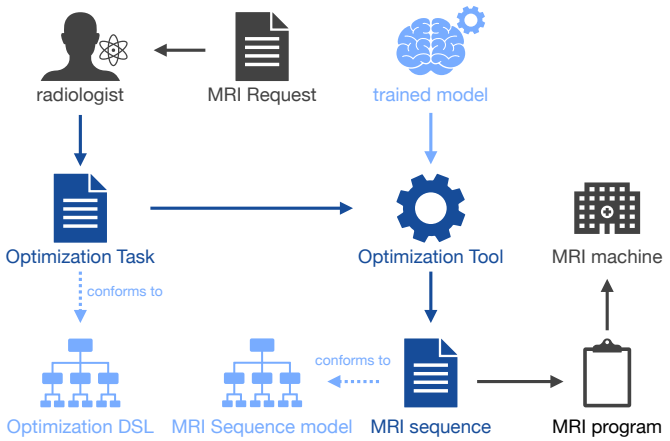

Fig. 2: Automated Sequence Determination Process

fulfilled but should be. All in all, we search for an element of the search space that fulfills the optimisation goal, while keeping the constraints on the optimisation space as well as the interdependencies of the search space to make sure it corresponds to a valid sequence.

III. Methodology

In Figure 1, we depict the usual workflow of an MRI program specification. The aspired automated workflow we are working towards is depicted in Figure 2. When the radiologist receives the physician's request, he formulates it in a formal manner that is still close to natural language. To that end, in [5], a domain-specific language (DSL) was developed. This formal optimisation requirement is transformed into an evolutionary algorithm configuration. The optimisation process is then carried out using a trained model $\tilde{f}$ of f. Finally, it results in an element of the defined search space, which is then transformed into a sequence formulation that—again—conforms to a DSL, which is close to natural domain language, i.e., easy to understand. This sequence can finally be automatically transferred to an MRI program. Instead of having a profound knowledge of the different aspects of sequence generation and programming, the radiologist now only needs to transform the MRI request to the optimisation task language.

Figure 2 also graphically depicts this work's contribution in a dark blue colour. The parts of the non-automated process are in dark gray, whereas the work of [5] are in light blue.

A. Optimisation technique

Because the problem at hand is multi-modal and non-differentiable (see [7] for works on differentiability), it is necessary to use an algorithm for optimisation that requires function evaluations only and does not depend on gradient techniques. The multi-modality suggests a stochastic approach to keep the possibility of leaving local optima. Considering both requirements, we propose a bio-inspired approach, namely genetic algorithms [8]. Their stochastic, population-based nature, together with a strictly function-based evaluation technique, can handle the challenges of the given task.

Genetic algorithms model the evolutionary process of selecting the fittest individuals for the next generation and recombination. Additionally, mutation introduces some variance to the population. When a given termination criterion is met, usually a fixed number of generations, the algorithm terminates, and the *best* candidate is returned.

Algorithm 1 Adapted Genetic Algorithm

$p_0 \leftarrow p_{initial}$
for $z = 1$ **to** $i \leq generations$ **do**
  $\theta \leftarrow computeMalusForVagueConstraints$
  $p_z^s \leftarrow select(p_{z-1}, d(\tilde{f}(p_{z-1})) + \theta)$
  $p_z^p \leftarrow select(p_{z-1}, d(\tilde{f}(p_{z-1})) + \theta)$
  $p_z^o \leftarrow mutate(recombinate(p_z^p), correlationInfo)$
  filterStrictConstraintsViolations(p_z^o)
  $p_z \leftarrow p_z^s \cup p_z^o$
end for

Genetic algorithms are a versatile tool that is adaptable to many use cases and requirements. During our description of the optimisation task, it became apparent that several adaptions need to be made to the standard algorithm. Algorithm 1 shows the principle of a standard genetic algorithm, which is enhanced with the problem-specific adaptions. We will describe these adaptions in the following paragraphs.

a) Handling of constraint types: There are several possibilities for dealing with constraints [9]. They can be divided into two categories. The first category does not allow the existence of individuals that violate any constraint, and the second one discourages their existence by worsening their fitness. As the optimization requirements can contain constraints that must not be violated (e.g., acquisition time < 5 min) as well as constraints that may slightly, however not massively, be violated, we adapt both strategies in a configurable way. Individuals are killed at birth when a strict constraint is violated. For non-strict constraints, we add the following malus to the fitness function:

$$\theta(\mathbf{x}) = \sum_{j=1}^{m_{nonStrict}} \theta_j^{\mathcal{S}} \cdot |cs_j(\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_s)| + \sum_{k=1}^{l_{nonStrict}} \theta_k^{\mathcal{O}} \cdot |co_k((\mathbf{f}(\mathbf{x}))_1, (\mathbf{f}(\mathbf{x}))_2, ..., (\mathbf{f}(\mathbf{x}))_o)| \quad (5)$$

where $\theta_k^{\mathcal{O}}, \theta_j^{\mathcal{S}}$ are weights with negative values for maximisation tasks and positive values for minimsation tasks (each time *worsening* the fitness, therefore making it harder for the individual in the population). The summation only runs over all non-strict constraints, as the strict constraints are handled by the killed-at-birth strategy.

b) Treatment of surrogate function: As mentioned in Section II, there is no explicit relationship known between image metrics and sequence parameters. Therefore, there is no canonical fitness function for the evolutionary algorithm. In situations like this, a reliable technique is the usage of surrogate functions (see [6], [10]). We use the trained function from [5] as a surrogate function and propose the usage of a density

estimation *de* [11] to soften the influence of prediction errors. This technique estimates the density of search space variables in the training data and uses this information to smooth out extreme optimisation function results in search space areas where little training data is present.

c) Treatment of interdependent search space: The search space contains dependencies between dimensions. For example, ETL, Epi-Factor and Refocusing Angle influence each other in their choice, as well as TE, TR, and readout duration, when talking about reasonable solutions. Through mutation and recombination operators in the genetic algorithm, it is possible to produce generations that do not contain these dependencies, i.e. $Cov(X_i^1, X_j^1) \neq Cov(X_i^2, X_j^2) \neq ...Cov(X_i^{z_{max}}, X_j^{z_{max}})$, where X_i^z denotes the random variable of the i-th search space variable in the z-th iteration of the genetic algorithm. To contain the covariance similar between generations, we apply correlation-aware mutation and recombination operators as proposed in [12]. They ensure that all correlated components of the search space are mutated (or recombined, resp.) together. This keeps the dependencies similar throughout the algorithm. The solution improvement stems from a change in the expected value and variance of the search space variables.

B. Translation of optimisation requirements to configuration

The presented optimisation requirement DSL of [5] contains three main parts: First, the optimisation goal itself, which states optimisation space variable and direction. Second, restrictions to *obey*, which contain constraints that must not be violated. Third, properties to *aim* for, which can be of a constraint nature or categorical type (*very high — very low*). We generate the fitting configuration for the optimisation tool as follows: For the optimisation goal, the direction is given as `maximize := true|false`, and the goal is specified by setting the corresponding weight for the fitness calculation to 1.0. The restrictions of type *obey* are directly transformed to constraint configurations of the form `constraint(expression, "strict");`, where `expression` equals the *left* side of the constraint. The properties to aim for that resemble constraints are transformed analogously. However, they receive the value `"vague"` instead of `"strict"`. The properties to aim for—that are categorical—are transformed into constraints with cardinal values `constraint(variable - toCardinal(categorical value), "vague");` where the categorical value depends on variables it refers to, e.g. *very high sharpness* translates to `constraint(Sharpness >= 0.9, "vague");`.

C. Repairing the optimisation result and translating it to sequence description

The result of an optimisation process is the vector representation of the search space. However, to (a) make it easier readable for the radiologist, (b) allow human improvement, and (c) allow the processing of this result to a scanner-compatible sequence information, we translate the result in the MRI sequence model DSL of [5]. This has the advantage that the model already includes the necessary semantics through the nesting of the information. Most parameters, however, can be directly processed as information after a short sanity check. Others need to be subjected to a repairing procedure. This is usually to account for actual applicability as sequence and compliance with basic interdependencies in the data. Additionally, the most upper level of sequence description (which type of echo and which type of readout) is not part of the optimisation output and is computed through the information in the variables `refocussing angle`, `ETL`, and `Epi-factor`.

IV. Implementation

We implemented the proposed methodology—shown in Figure 2—using the model-driven development paradigm [13] using different Java frameworks. In model-driven development, the (data) model is the essential aspect and the basis for data transformations. To this end, we are using Xtext [14], a framework from the Eclipse modeling framework [15], for specifying the optimisation task DSL. The DSL specification defines the grammar for optimisation tasks and how a file—that conforms to this grammar—is translated into an optimisation task model. We then use Xpand—a model-to-text framework—to automatically generate optimisation task-specific configuration files for the open-source optimisation tool suite EvoAl [16]. The optimisation tool is partially built on Jenetics [17] to search for an optimal solution to the configured problem. Jenetics is a Java library providing an adaptable evolutionary algorithm. In this search process, we use Smile [18] to pre-train a model to predict the fitness of MRI sequences. The final solution is then translated into an MRI sequence model that allows the description of MRI sequences that are independent of the used MRI scanner.

V. Evaluation

As the main contribution of this paper is the chosen adapted evolutionary algorithm (genetic algorithm), this section mainly focuses on its evaluation. We aim to answer the following evaluation questions:

Research Question 1: First, does the proposed optimisation strategy find MRI sequences that fulfil the given requirements and are optimal concerning the used training data?

Research Question 2: Second, how does the chosen algorithm compare to a standard evolutionary algorithm that does not consider the training data's peculiarities and the given search space interdependencies?

Research Question 3: Third, what is the relationship between optimisation result and time effort, i.e., to what degree does an increase in population size and generations improve the algorithm's result?

A. Evaluation Setup

We specify four sets of requirements that model different clinical situations, translate them into constraints and optimisation tasks and analyse them using varying parameter sets of the evolutionary algorithm. The four sets of requirements are modelled with respect to the training data in the sense that—given the restricted set sequence types in the training

data—sequences do exist that fulfil the requirements. That does not mean the set of requirements has to be present in the training data; however, it is reasonably close to some that are.

Every setup (requirement and parameter set) is run ten times to account for stochastic effects. For every individual x of each run, we collect the relative optimality $opt_r = f_r(x)/opt\ data_r$, the proportion of violated strict constraints, and the proportion of violated soft constraints. For each setup, we report averages of these values over ten runs and the maximum.

We vary the following parameters for the evolutionary algorithm: As general features for the evolutionary algorithm, we vary whether we use the fitness adaption and/or correlation-aware operators (*no* vs. *with density* and *correlation* vs. *normal*). Additionally, we have a look at the maximum number of generations to get some understanding of what maximum number of generations may be sufficient. This keeps the goal in mind to produce a feasible setup for reality, which would include a timely computation. Last but not least, we experiment with different population sizes, ranging from 20 to 500.

Note that the variation of the maximum number of generations does not require separate evaluation runs. All setups are run for 1000 generations, however, we evaluate the above-mentioned metrics after 10, 50, 100, 200, and 500 evaluations.

B. Evaluation results—description

We focus on describing the results for a target specification, which aims to maximise the signal-to-noise ratio in CSF, while having a white and gray matter contrasts of at least 20 and 15, resp. and an avoidance of ghosting. We will comment on the behaviour of the other target specifications later on. In Figures 3 - 6, the evolution of the mean of the population (over all runs) and mean of the maximum of the population (over all runs) is depicted (dark blue points stand for max, light blue points for mean). Please note that only those individuals that satisfy the weak constraints, as well as the strict constraints, were included in the computation of this data. Furthermore, the reported values are those achieved by using the surrogate model as the predictor of the individuals. The adaptions through constraints or density adaption are not included in these values to keep the variants comparable.

We first have a look at the optimisation results for a high population count ($m = 500$) and a low probability for mutation. The results (as described above) are shown in Figure 3. The left side shows the results for *no density*, i.e., the surrogate model is used as such for the fitness function, and the right side shows the result for the density adaption. The upper facets show the results for using the correlation adaption, and the lower facets show these for using standard mutation operators and crossover.

First, we see that the mean and max show comparable behaviour as was to be expected. They follow the same trajectory, with the maximum a little more unstable than the mean. This was to be expected through the population size of 500. The results using no density adaption reach very high values very fast after around 50 generations. The density and correlation adapted results first drop and then increase until

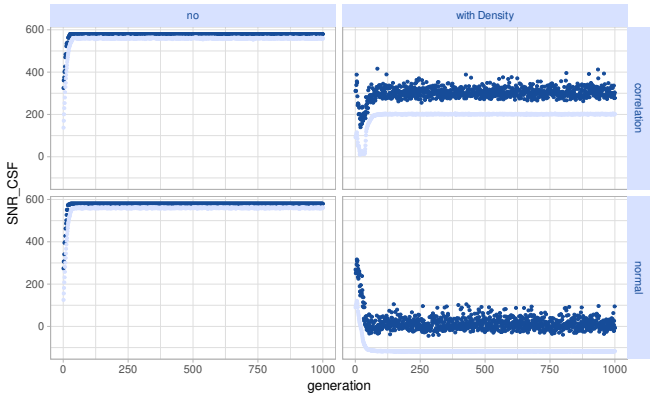

Fig. 3: Results for BSSFP-Setup (well represented in training data, differentiated by correlation and density adaption usage) with low mutation probability and population size of 500

they stabilise at around 100 generations but at a much lower value. The density-adapted but not correlation-adapted setup decreases and then stabilises at a value below zero.

These results need some domain context to be correctly interpreted. A signal-to-noise ratio is always higher than zero, and the highest signal-to-noise ratios for CSF lie at around 300 in the training data. With this in mind, we can turn to an interpretation of the presented results. Both variants with no density adaption do perform significantly better in maximising the respective value; however, this value reaches unrealistic areas with respect to the domain and training data. This leads to the assumption that the high values reported result from a worse prediction of the fitness value at this position. An expert review of the actual resulting sequence confirms its invalidity. The density approach does lead to *worse* results in terms of sheer prediction value; however, it seems to be in a more realistic area. This is exactly in line with the explanation by Plump et al. when presenting this technique: Areas with little training data are *flattened*, i.e. high fitness predictions are decreased, and low fitness predictions are increased to prevent extreme values with small confidence to steer the population in the wrong direction [11]. Validating the final sequences leads to a positive result. The lower right corner shows an entirely unexpected behaviour as it seems to minimise the fitness results. We assume that this is the case because the considered sequences do not follow the given interdependencies and, therefore, result in wrong fitness predictions.

Figure 6 shows the results for a population of $m = 500$ and a high mutation rate ($p = 0.8$). The behaviour of the first three facets is comparable to Figure 3, besides the slower convergence and the greater distance between the mean and the maximum, which is, however, to be expected for a high mutation rate. The right lower facet shows a different behaviour than before: It works now as expected, resulting in results comparable to the setup with both density and correlation adaption. We assume that the negative results from Figure 3 are prevented through the high mutation rate, thus more likely achieving valid sequences and thus resulting in good

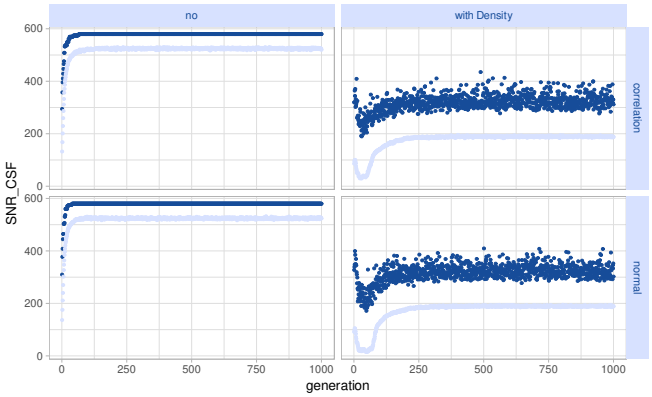

Fig. 4: Results for BSSFP-Setup (well represented in training data, differentiated by correlation and density adaption usage) with high mutation probability and population size of 500

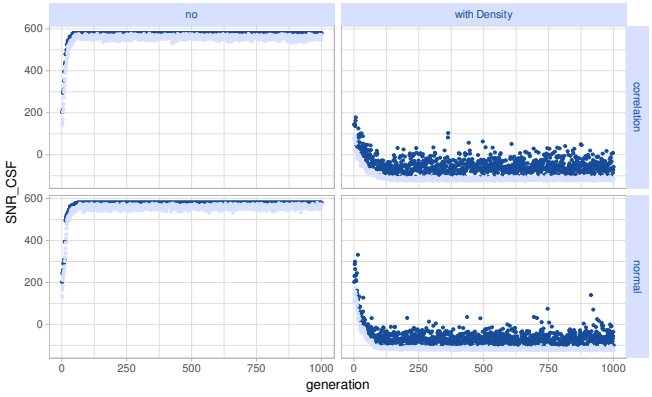

Fig. 5: Results for BSSFP-Setup (well represented in training data, differentiated by correlation and density adaption usage) with low mutation probability and population size of 20

predictions through the surrogate model.

After describing the results for the highest instance of population variations, we now turn to the smallest one to discuss the other end, namely $m = 20$. Figure 5 shows the results for a low mutation rate. First of all, the comparison of mean and maximum values confirms the small population size, as they are much closer to one another and the mean values show more deviation over the course of time, i.e. single extremal values in the population have a higher influence which is to be expected with a smaller population size. Second, the left-hand side variant (i.e. with no adaption of the surrogate model) performs in a comparable manner to the high population setup. Hence, it shows the same weaknesses, i.e. unrealistic results through a bad fit of the surrogate model. The density-adapted variants both show negative behaviour as well, as they seem to minimise their results. We assume that due to the small population size the correlation-adapted alterers cannot show their potential, as their computation relies on population-based measurements which are less reliable in small population sizes. This could also explain the comparable

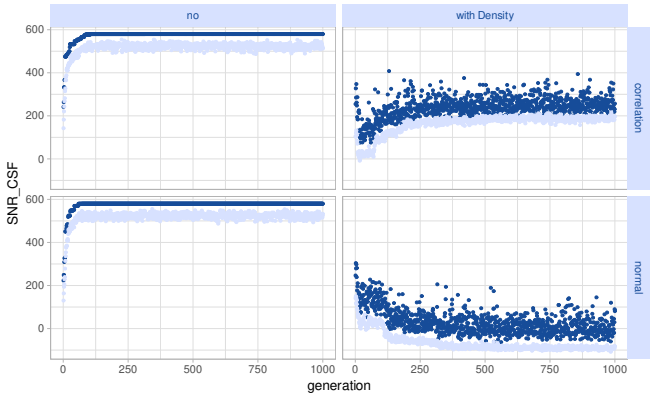

Fig. 6: Results for BSSFP-Setup (well represented in training data, differentiated by correlation and density adaption usage) with high mutation probability and population size of 20

behaviour for the non-adapted alterers in the lower right facet.

Last but not least, Figure 6 shows the results for a population size of $m = 20$ and a high mutation rate of $p = 0.8$. Means and Maximum show the expected behaviour resulting from a small population size (more deviation in the mean, less distance between both measures) as well as the effects of a high mutation rate (slower convergence). The non-density adapted left-handed results again behave as before. Nevertheless, the right facets show a different behaviour now. The evolutionary algorithm using the correlated adaption performs comparably well again. The high mutation rate seems to remedy the negative effects of the small population size from before. The non-correlated approach, however, minimises again instead of maximising - here, the higher mutation rate can not counteract the small population size.

Investigating the remaining setups leads to comparable results for two of them with feasible sequences that fulfil the experts' expectations as optimisation results. The fourth, however, does not show the expected result with the surrogate model, nor are the results valid for MRI imaging. We assume this is due to the fact that this type of sequence is underrepresented in the ML model presented by Hoinkiss et al. and is, therefore, at a disadvantage during optimisation.

VI. Discussion

A. Evaluation Results

Regarding our presented evaluation questions, we can answer question 1 positively, however, in a rather anecdotical manner. For three of our target specifications, the proposed variants with correlation mutation operators and the density fitness adaption resulted in valid sequences that (according to the trained model) fulfill the constraints and maximize the specified value (w.r.t the training data). Nevertheless, a more thorough target variation might be necessary to answer this question more confidently.

Regarding our second question, we find that the density adaption on the surrogate model is necessary to achieve feasible sequences instead of being steered to an unknown region of

sequences, which simply perform well on the extrapolation of the surrogate model. The effectiveness of the correlation adaption depends on population size and mutation rate. For a high population size, it performs well but not always better than the normal alterers, and for a low population, it only performs well for a high mutation rate but always outperforms the standard approach.

For question 3 - from the given results, 200-300 generations are usually enough for the algorithm to stabilise and thus yield a result. Population size, however, is another matter. It seems that in order to achieve positive effects through the correlation adaption on the altering operators, a population size of 100 is necessary.

B. Threats to validity

In this paragraph, we discuss evaluation decisions that may invalidate or weaken our results, as well as workflow decisions that may threaten the actual applicability of our approach.

In our evaluation, we measure the success of an evolutionary algorithm run with regard to the training data. However, while the training data is a good representation of both search and optimisation space (as reported in [5]), it is not clear that the overall optimum (given all possible sequences) is in it. Therefore, a good algorithm result— as evaluated in Section V— does not mean that the actual optimum has been found. Furthermore, when the solution is a sequence in a sparse area of the training data, the estimated function may differ from the true function. Therefore, the resulting fitness value is only partially reliable. A possibility to truly evaluate this result would be an additionally run simulation or an actual run on an MRI. However, for the amount of evaluation runs, the latter is not feasible.

C. Features to improve and further steps

Our goal is a fully automated MRI sequence optimiser for practical use. While this work shows several necessary steps towards that goal, some aspects still need more work from our point of view. First, we are interested in automatically adapting the evolutionary algorithm parameters to the optimisation task. The presented solutions show a good, general approach; however, optimisation tasks with unusual constraints or optimisation parameters may benefit from an automatic adjustment. The idea is to build a model predicting the evolutionary algorithm's success from the set parameter. However, the training of this model requires a much more profound database for the optimisation function and valid optimisation results. Second, the estimated optimisation function needs improvement. On the one hand, the machine learning technique might benefit from more domain information about dependencies in the data. On the other hand, domain experts are researching AI-supported simulations of image metrics given the sequence parameters [19]–[21]. Our approach could benefit from this work, as the possibility of having faster simulations may allow for a merged approach in the evolutionary algorithm. Third, to make this automated workflow useful for researchers (and not only practitioners) in the MRI domain, an extension of the sequence language would allow the study of more sophisticated MRI sequences. It would be beneficial for developing new sequences because possible sequence definitions could be proposed by the algorithm, possibly reducing experimental time at the MRI.

VII. Conclusion and Outlook

The non-invasiveness of MRI, which allows for a precise diagnosis without harming the patient, makes it an essential tool in medical diagnosis. However, sequence programming and adjustment of related parameters with respect to the clinician's request is a demanding task requiring a lot of experience and expert knowledge. The necessary experience and knowledge are critical bottlenecks for the availability of this tool, especially in underdeveloped countries. This paper proposes the next critical step in building a fully automated sequence optimizer—a flexible, automatically configurable evolutionary algorithm for the optimisation process. We successfully evaluated this optimizer regarding the supplied training data and showed that the choice of features improves the optimiser compared to a standard evolutionary algorithm. In further work, we plan to develop an automatic adaption of the optimizing parameters to the optimisation task and an improvement of the surrogate function replacing the unknown optimisation function.

References

[1] M. A. Bernstein, K. F. King, and X. J. Zhou, *Handbook of MRI Pulse Sequences*. Elsevier, 2004.

[2] M. Shinnar, S. Eleff, H. Subramanian, and J. S. Leigh, "The synthesis of pulse sequences yielding arbitrary magnetization vectors," *Magnetic Resonance in Medicine*, vol. 12, no. 1, pp. 74–80, oct 1989.

[3] B. A. Hargreaves, D. G. Nishimura, and S. M. Conolly, "Time-optimal multidimensional gradient waveform design for rapid imaging," *Magnetic Resonance in Medicine*, vol. 51, no. 1, pp. 81–92, 2003.

[4] M. Lustig, S.-J. Kim, and J. M. Pauly, "A fast method for designing time-optimal gradient waveforms for arbitrary k-space trajectories," *IEEE Transactions on Medical Imaging*, vol. 27, no. 6, pp. 866–873, jun 2008.

[5] D. C. Hoinkiss, J. Huber, C. Plump, C. Lüth, R. Drechsler, and M. Günther, "AI-Driven and Automated MRI Sequence Optimization in Scanner-Independent MRI Sequences Formulated by a Domain-Specific Language," *Frontiers*, 2023.

[6] Y. Jin, "Surrogate-assisted evolutionary computation: Recent advances and future challenges," *Swarm and Evolutionary Computation*, vol. 1, no. 2, pp. 61–70, 2011. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2210650211000198

[7] A. Loktyushin, K. Herz, N. Dang, F. Glang, A. Deshmane, S. Weinmüller, A. Doerfler, B. Schölkopf, K. Scheffler, and M. Zaiss, "MRzero - automated discovery of MRI sequences using supervised learning," *Magnetic Resonance in Medicine*, vol. 86, no. 2, pp. 709–724, mar 2021.

[8] J. H. Holland, "Genetic algorithms," *Scientific American*, vol. 267, no. 1, pp. 66–73, 1992. [Online]. Available: http://www.jstor.org/stable/24939139

[9] C. A. Coello Coello, "Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: a survey of the state of the art," *Computer Methods in Applied Mechanics and Engineering*, vol. 191, no. 11, pp. 1245–1287, 2002. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0045782501003231

[10] T. Chugh, C. Sun, H. Wang, and Y. Jin, "Surrogate-assisted evolutionary optimization of large problems," *High-Performance Simulation-Based Optimization*, pp. 165–187, 2020. [Online]. Available: https://doi.org/10.1007/978-3-030-18764-4_8

[11] C. Plump, B. J. Berger, and R. Drechsler, "Using density of training data to improve evolutionary algorithms with approximative fitness functions," in *2022 IEEE Congress on Evolutionary Computation (CEC)*, 2022, pp. 1–10.

[12] C. Plump, B. J. Berger, and R. Drechsler, "Domain-driven correlation-aware recombination and mutation operators for complex real-world applications," in *2021 IEEE Congress on Evolutionary Computation (CEC)*, 2021, pp. 540–548.
[13] D. Schmidt, "Guest editor's introduction: Model-driven engineering," *Computer*, vol. 39, no. 2, pp. 25–31, 2006.
[14] M. Eysholdt and H. Behrens, "Xtext: Implement your language faster than the quick and dirty way," in *Proceedings of the ACM International Conference Companion on Object Oriented Programming Systems Languages and Applications Companion*, ser. OOPSLA '10. New York, NY, USA: Association for Computing Machinery, 2010, pp. 307–309.
[15] D. Steinberg, F. Budinsky, M. Paternostro, and E. Merks, *EMF: Eclipse Modeling Framework 2.0*, 2nd ed. Addison-Wesley Professional, 2009.
[16] B. J. Berger, C. Plump, and R. Drechsler, "Evoal: A domain-specific language-based approach to optimisation," in *2023 IEEE Congress on Evolutionary Computation (CEC)*, 2023, pp. 1–10.
[17] F. Wilhelmstötter, "Jenetics," https://jenetics.io, Jan. 2023.
[18] H. Li, "Smile," https://haifengl.github.io, Jan. 2023.
[19] A. Loktyushin, K. Herz, N. Dang, F. Glang, A. Deshmane, S. Weinmüller, A. Doerfler, B. Schölkopf, K. Scheffler, and M. Zaiss, "MRzero - Automated discovery of MRI sequences using supervised learning," *Magnetic Resonance in Medicine*, vol. 86, no. 2, pp. 709–724, 2021. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/mrm.28727
[20] Q. Yang, Z. Wang, K. Guo, C. Cai, and X. Qu, "Physics-driven synthetic data learning for biomedical magnetic resonance: The imaging physics-based data synthesis paradigm for artificial intelligence," *IEEE Signal Processing Magazine*, vol. 40, no. 2, pp. 129–140, Mar. 2023. [Online]. Available: http://dx.doi.org/10.1109/MSP.2022.3183809
[21] H. Haji-Valizadeh, R. Guo, S. Kucukseymen, Y. Tuyen, J. Rodriguez, A. Paskavitz, P. Pierce, B. Goddu, L. H. Ngo, and R. Nezafat, "Comparison of complex k-space data and magnitude-only for training of deep learning–based artifact suppression for real-time cine MRI," *Frontiers in Physics*, vol. 9, Sep. 2021. [Online]. Available: https://doi.org/10.3389/fphy.2021.684184