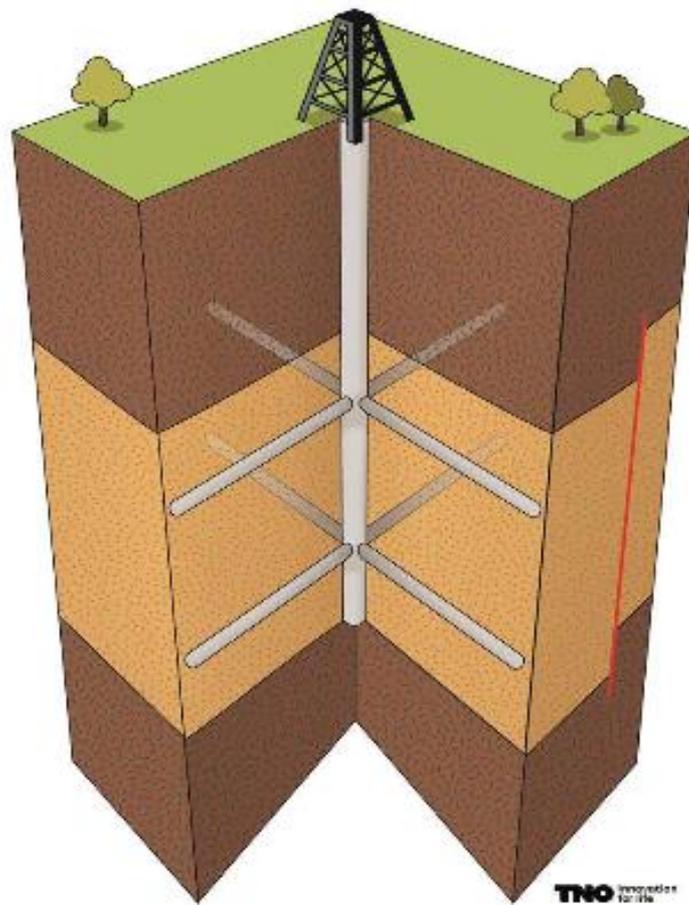


Diederik Troost

Optimization of Jetted Radial Wells



Optimization of Jetted Radial Wells

By

Diederik Troost

in partial fulfilment of the requirements for the degree of

Master of Science
in Applied Earth Sciences

at the Delft University of Technology,
to be defended publicly on Tuesday January 15, 2019 at 10:00 AM.

Supervisor:	Prof. dr. ir. J.D. Jansen	TU Delft
Thesis committee:	Prof dr. D.F. Bruhn,	TU Delft
	Dr. A. Barnhoorn,	TU Delft
	Dr. ir. E. Peters,	TNO

This thesis is confidential and cannot be made public until January 14, 2019.

An electronic version of this thesis is available at <http://repository.tudelft.nl/>.

List of symbols and abbreviations

az	Radial azimuth
BHP	Bottom hole pressure
C	Covariance matrix
c_c	Cumulation of CMA-ES distribution
c_{cov}	Covariance matrix rate of change
$CMA - ES$	Covariance Matrix Adaption – Evolution Strategy
c_p	Thermal capacity
C_u	Sample distribution EnOpt
c_w	CMA-ES control vector selection weights
C_x	Preconditioning matrix EnOpt
$C_{x,J(x)}$	Cross-covariance matrix, control vector and objective function
c_σ	Cumulation for CMA-ES step size
d_σ	CMA-ES step size adaption damping term
$EnOpt$	Ensemble Optimization
GA	Genetic Algorithm
G^T	$1 \times N_x$ search direction matrix
in	Radial inclination
I^R	Masking matrix
$J(\mathbf{x})$	Objective function
k	Iteration counter
KOD	Kick-off depth
$K_{x,y,z}$	Permeability in x,y or z direction
$LCOE$	Levelized cost of energy
len	Radial length
m	Mass
n	Number of radials in kick-off
N_c	Number of control vector elements
N_x	Number of ensemble control vectors
NPV	Net Present Value
p_i	Reservoir pressure in gridblock i
p_i^w	Well pressure in gridblock i
q_i^w	Well flow rate in gridblock i
\mathbf{p}	Search direction
\mathbf{p}_c	Evolution path
PSO	Particle Swarm Optimization
\mathbf{p}_σ	Conjugate evolution path
Q	Energy
R^{int}	Mixed integer mutation
$r_{o,x,y,z}$	Well equivalent radius on x,y or z projection
r_w	Well radius
R_x	Smoothing matrix EnOpt
S^{int}	Diagonal matrix of control vector with continuous values set to 0

T	Temperature
\mathbf{u}	Control vector
V	Velocity term PSO
w	Radial window
WI	Well Index
\mathbf{x}	Iteration initial control vector
X	Position term PSO
ρ	Density
ϕ	Porosity
σ	Step size
λ	Number of population members
λ_{reev}	Number of CMA-ES population members to reevaluate
μ	Number of control vectors to select for covariance matrix adaption
μ_{eff}	Sum of squared weights of selected control vectors for covariance matrix adaption

Contents

1	Introduction	1
2	Literature review	3
2.1	Radial Jet Drilling	3
2.1.1	Case Studies.....	4
2.2	Numerical Optimization	4
2.2.1	Gradient-based optimization.....	5
2.2.2	Gradient-free optimization.....	7
2.2.3	Optimization in Well Design & Placement	13
2.2.4	Robust Optimization.....	13
2.3	Well connectivity	14
2.3.1	Validation.....	16
3	Methodology.....	18
3.1	Objective Function	18
3.2	Radial Controls & Constraints	19
3.2.1	Radial Parametrization	19
3.2.2	Radial proximity.....	20
3.3	Optimization Test Case.....	22
3.3.1	Results	23
4	Optimization workflow.....	25
4.1	Workflow Cases	25
4.1.1	Klaipeda	26
4.1.2	Doublet	27
4.1.3	Fractured	28
4.2	Results.....	29
4.2.1	Klaipeda	29
4.2.2	Doublet	31
4.2.3	Fractured	32
4.3	Discussion.....	32
5	Conclusion & Future Work.....	35
	Bibliography.....	36

Abstract

Jetted radial wells are a relatively new form of hydraulic stimulation for oil and gas wells, where small diameter holes extend radially from a wellbore. This technique for increasing well connectivity could also be applied to geothermal wells. Radials can be freely placed along a backbone well, making the design of an economically viable well plan time intensive. Numerical optimization has been extensively applied to well design and will be applied for the design of a radial well plan. A simple optimization case is used to compare gradient-based and gradient-free optimization for jetted radial optimization, the gradient-free CMA-ES algorithm is chosen for the workflow. In addition to CMA-ES, mixed integer CMA-ES and uncertainty-handling CMA-ES are incorporated in the workflow. Three reservoir cases are evaluated for optimization using these algorithms. The workflow is able to determine well plans with higher NPV in each of these cases though finding a global optimum remains difficult.

Acknowledgements

The research for this paper received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 654662 (SURE). The content of this poster reflects only the authors' view. The Innovation and Networks Executive Agency (INEA) is not responsible for any use that may be made of the information it contains."

Summary

This thesis describes the development of an optimization workflow for radial jetted drilling (RJD), a recompletion technique in which small diameter lateral wells protrude radially from a wellbore. This technique has seen some use in hydrocarbon reservoirs but has not yet been widely used in geothermal reservoirs, which is the focus of this thesis.

Radials can be located anywhere along a wellbore in addition to it being possible for multiple radials to be jetted on the same depth. This makes the possible well designs incorporating radials large. To make the process of determining an economically viable plan quicker and easier, numerical optimization will be used. Numerical optimization adapts the inputs of some function with the aim of optimizing the result of that function over multiple iterations. There are many forms of numerical optimization. A distinction can be made between gradient-based optimization and gradient-free optimization. These methods differ in how function outputs are used to determine new input parameters. Several algorithms of these types are examined. As radials are not very common, the Well Index calculation method which represents wells in reservoir simulation is validated for simulating radials. It is found that the suggested Well Index calculation is satisfactory for the simulation of radials.

The radial well design will be optimized using a simplified form of net present value (NPV) as the objective function. The volume of water produced, as determined by a reservoir simulation, is used to calculate revenue from the geothermal reservoir. This is weighed against the cost of jetting radials to determine NPV. This requires a parametrization of radials so values can be input into the optimization algorithm which can then be translated to a well plan for the reservoir simulation. The effects of radials being in close proximity to one another are then examined. Numerical errors of roughly 2% of total production were observed. The decision about which algorithm to use for the optimization was made by comparing two types of optimization in a simple reservoir test case. The covariance matrix adaption – evolution strategy (CMA-ES) performed better than ensemble optimization (EnOpt) in this test case, this algorithm will be used for the workflow. In addition to CMA-ES, mixed integer CMA-ES (MI-CMA-ES) and uncertainty handling CMA-ES (UH-CMA-ES) will be used in the workflow.

These algorithms are run in three reservoirs and their results compared. The reservoirs are (1) a horizontally homogenous reservoir with thin layers based on the Klaipeda geothermal demonstration site (2) a doublet with horizontally extensive features along flow direction and (3) a dual porosity reservoir with a faulted block based on the Californië geothermal site. In each case, all algorithms are able to determine a well plan which increases NPV beyond that of the initially input well plan. Kick-off depth and number of radials have the most pronounced effect on NPV, the effect of radial direction is found to be limited.

1 Introduction

The use of high pressure fluids for well stimulation is not uncommon. Hydraulic fracturing is an established method of increasing reservoir connection by creating fractures near the wellbore. However, hydraulic fracturing is a controversial technique and prohibited in some EU countries. Radial jet drilling (RJD) is another method of utilizing a high-pressure fluid to increase well contact with a reservoir. A well completed with a RJD job has one or more small diameter holes extending radially from the wellbore (“radials”). These holes are made using a self-propelled jetting nozzle connected to a flexible hose. The result is similar to conventional reservoir perforation, though jetted radials can be longer and have a greater diameter than conventional perforations.

The RJD job can be performed in existing wells using a workover rig, or the well can be completed with radials after drilling. Jetting a radial also requires less fluids and chemicals compared to hydraulic fracturing. This would make RJD an interesting alternative to hydraulic fracturing if the radials consistently show an increase in reservoir connection.

RJD was first introduced in 1985 (2), however it has yet to be widely adopted as a technique to increase reservoir connection. Case studies of applying RJD in a field setting were mostly conducted after 2010. In these studies multiple wells in a (mature) field were jetted which led to an increased field production, even though some radials could not successfully be jetted.

A factor to consider when jetting radials is the uncertainty in radial trajectory. There is little information about the location of the jetting nozzle, besides the length of the hose used. As the hose is flexible and the nozzle has little steering capability, the path the jet ultimately takes while jetting is not known (3).

While RJD has been applied in oil fields, the use of jetted radials in geothermal wells is still limited. Increasing reservoir contact of one or both wells in a geothermal doublet would allow more water to circulate in the doublet and in turn, produce more water.

The placement of radials along the main wellbore has a significant effect on the injectivity. Ideally radials are jetted in layers that connect the injector and producer, and have high permeability. In addition, the jetting of a radial will incur some costs. The goal is to find a well plan in which the increased production is weighed against cost jetting. Manually making this well design is challenging and time consuming due to the large number of options when deciding where to jet radials. Automating the well design would make the selection of a suitable candidate well for an RJD job an easier process.

The goal of this thesis is stated as:

“Develop an optimization workflow for RJD well design in a geothermal well which optimizes economic potential while accounting for uncertainty in radial trajectory.”

First the background of jetted radial drilling and its application are examined. Numerical methods which could be applied to the optimization of radials are reviewed and the method of well connection to a reservoir in reservoir simulation are examined. The specifics for the optimization of jetted radials are then determined. An objective function is defined and a parametrization of radials is made so the optimization algorithm can create a well design incorporating radials. The effect of interference of nearby radials in production are then examined. Next, two optimization algorithms are compared in a simple radial jetted drilling test case to make a choice which algorithm to use in the workflow. Having made this choice, the workflow is run in three reservoir cases. The results of these optimizations are discussed before conclusions are drawn and future work is suggested.

2 Literature review

This chapter examines the background of radial jetted drilling as well as reviewing case studies of radial jetting jobs in hydrocarbon reservoirs. In addition to this RJD review, several numerical optimization methods are examined. A distinction is made between gradient-based and gradient-free optimization methods. Algorithms of both types which could be applied for the optimization of a radial well are reviewed. Finally, well connectivity to a reservoir in reservoir simulation is reviewed.

2.1 Radial Jet Drilling

Radial drilling was first proposed in 1985 to create additional drainage or injection holes in shallow heavy oil reservoirs (2). This work describes the general procedure of creating multiple radials protruding from a backbone well. A key feature of radial drilling is the option to drill multiple radials at the same level. The radial drilling system that Dickinson, et al (2) propose has a 4" diameter and includes the use of a 1¼" tubing to be used in the radial wells. Contemporary jetted radials are 1"-2" in diameter and do not use tubing, instead the radial is attached to a flexible high-pressure hose.

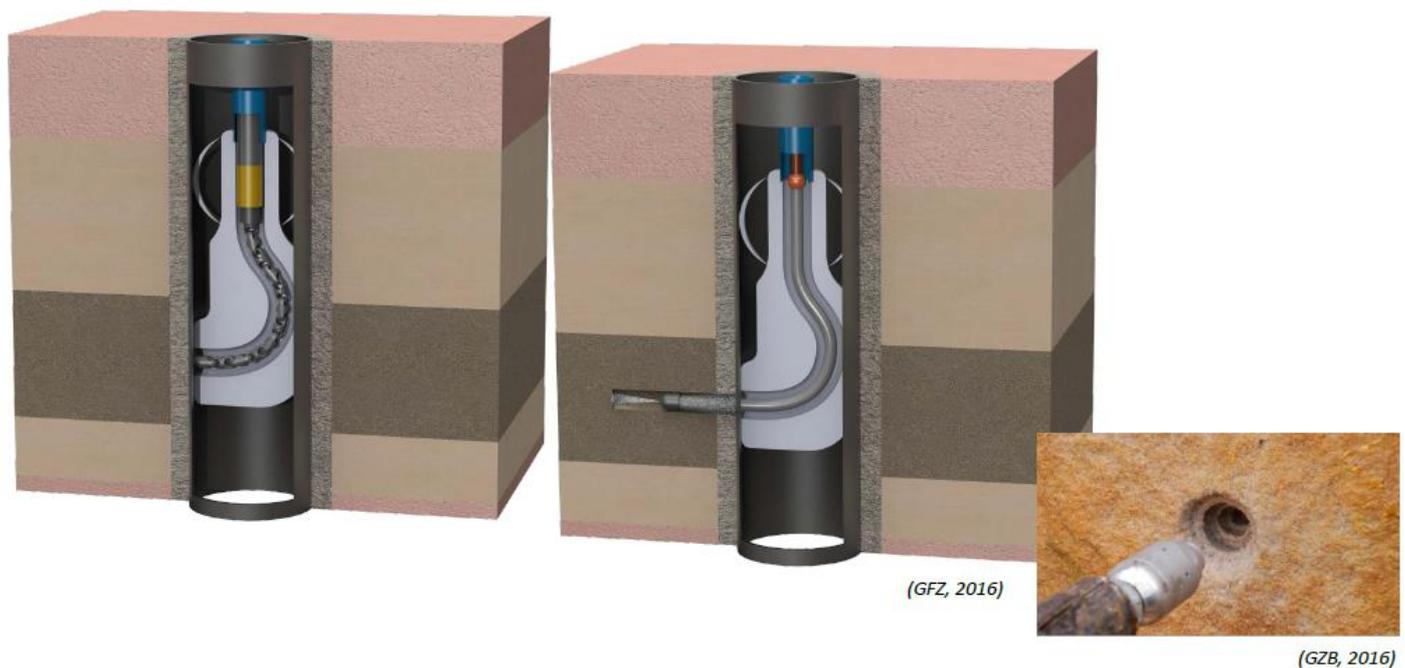


Figure 1 3D representation of jetting procedure. **Left:** Casing is milled through deflector shoe using milling bit. **Middle:** After retracting milling bit, jetting nozzle is lowered and jetting job commences. **Bottom right:** Photo of jetting nozzle and jetted hole (courtesy SURE Reinsch 2018)

While the jetting nozzle used differs by service provider, they all function using the same mechanics. A RJD nozzle has a forward jet to erode rocks and backward jets which allow the nozzle to propel itself through a reservoir. The nozzle will keep jetting through the reservoir, until the frictional drag force of the high-pressure hose being pulled along the jetted radial is equal of

greater to the force of the backward jets (4). In addition, the backward jets centre the nozzle in the jetted hole and keep tension on the high-pressure hose (4) (5). The tension on the hose is the only mechanism ensuring the jetted radial is straight.

Determining the trajectory of a radial during or after jetting is impractical, so checks whether a radial has deviated from a straight trajectory is rarely examined. In a quarry test, several radials were jetted and their trajectory after jetting was measured (3). It was observed that radials deviated heavily from the planned straight trajectory.

2.1.1 Case Studies

Due to the low cost and possibility to recomplete old wells with RJD, it has seen use in mature fields to improve production (6) (7) (8). The RJD completion jobs were usually done in wells that were not or barely producing, except in the Donelson field where 2 new wells were drilled and completed with RJD (6). While most of the radials were jetted successfully, a successful RJD job did not guarantee increased production. A few wells in the Assam cases were jetted but did not produce more (7) (8). Other wells in the same field showed a significant production increase. The results of the jetting jobs are shown in Table 1, the studied cases did not provide production data over longer periods.

Recompleting with RJD has the potential to extract more oil from minimally performing wells, although there is no indication how much of an increase can be expected or whether a well will not produce more. Completing a new well with RJD has yet to see widespread adoption, though in the Donelson field the newly drilled wells with RJD did contribute to the production increase.

The studied cases were considered successful by the field owners so there seems to be promise in this technology. So, there is an incentive to examine whether RJD would also be beneficial in geothermal applications.

Table 1 - Evaluation of Radial Jetting case studies

Case/field	Length jetted/planned [m/m]	# wells	Prod. per well before [m ³ /day]	Prod. per well after [m ³ /day]	Increase [%]
Donelson	7200/7200	10	0.06	0.105	175%
Upper Assam	949/1300	3	1	19.3	1930%
Assam Arakan	1590/1700	4	0.25	9.5	292%

2.2 Numerical Optimization

Mathematical optimization aims to find the input variable x , for which the function $f(x)$ returns an optimal value (9). The optimal value is a global maximum or minimum, depending on the type of optimization. In Figure 2 a basic function $f(x,y)$ is shown where the global maximum is found at $x = y = 0$. Here x and y are the design variables. While the global optimum of the function in Figure 2 can also be found analytically, the goal in optimization is to find the global optimum value for functions where this is no longer possible.

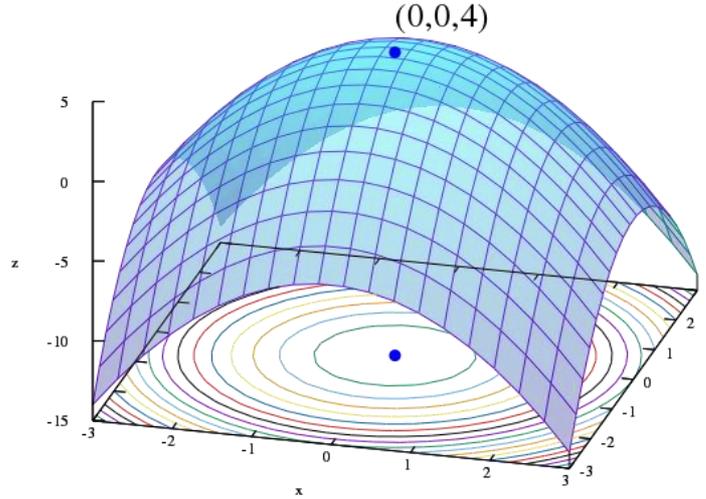


Figure 2 - $f(x,y) = -(x^2+y^2)+4$, the global optimum is located at $x=0$ and $y=0$

2.2.1 Gradient-based optimization

In a gradient-based optimization method, new design variables are suggested based on making a step in the search direction computed using the gradient (eq. [1]).

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \sigma^k \mathbf{p}^k \quad [1]$$

A new vector of design variables for the next iteration \mathbf{x}^{k+1} is suggested by taking the control vector at the current iteration \mathbf{x}^k and making a step with size α^k in the search direction of \mathbf{p}^k . This is the main loop that will perform the optimization of the objective function. To determine the step size σ^k , another loop is required. In this loop a line search is used to find a step size such that $f(\mathbf{x}^k + \sigma^k \mathbf{p}^k) > f(\mathbf{x}^k)$.

How the search direction \mathbf{p}^k is computed, depends on the optimization method but the principle remains the same. These methods gain local information, the gradient, of the objective function using function evaluations. Utilization of the gradient to find a search direction can lead to fast localization of optimal values, as the gradient gives information on where to find the greatest change in objective function value. The gradient does not differentiate between local and global optima. This type of optimization does not perform well on non-smooth objective functions where computing the gradient is difficult or impossible.

2.2.1.1 Ensemble Optimization

A gradient-based optimization method used in reservoir control is ensemble optimization (EnOpt) proposed by Chen et al. (10). The control vector update step in this optimization method (eq. [2]), is a slightly modified form of eq. [1].

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \sigma^k \mathbf{C}_x \mathbf{G}^{T,k} \quad [2]$$

With the control vector \mathbf{x}^k having N_x controls. Where $\mathbf{G}^{T,k}$, a $1 \times N_x$ matrix is the search direction, C_x a preconditioning matrix and σ^k is a parameter tuning the step size. The function evaluations of the perturbed control vectors are used to compute the gradient of the objective function. A correlated gaussian distribution with zero mean and distribution C_u is used to sample N_e perturbed controls vectors \mathbf{u}_i (eq. [3]), such that the mean of all perturbed controls vectors is approximately equal to the initial control vector as in eq. [4].

$$\mathbf{u}_i \sim \mathbf{x}^k + \mathcal{N}(0, C_u) \quad i = 1, \dots, N_e \quad [3]$$

$$\mathbf{x}^k \approx \langle \mathbf{u}_i \rangle = \frac{1}{N_e} \sum_{i=1}^{N_e} \mathbf{u}_i \quad [4]$$

$$\langle J(\mathbf{u}_i) \rangle = \frac{1}{N_e} \sum_{i=1}^{N_e} J(\mathbf{u}_i) \quad [5]$$

A cross-covariance matrix $C_{x,J(x)}$ between the objective function $J(x)$ and control vectors is approximated using eq. [6].

$$C_{x,J(x)} = \frac{1}{N_e - 1} \sum_{i=1}^{N_e} (\mathbf{u}_i - \mathbf{x}^k) (J(\mathbf{u}_i) - \langle J(\mathbf{u}_i) \rangle) \quad [6]$$

This cross-covariance matrix approximates the search direction terms $C_x \mathbf{G}^{T,k}$ in eq. [2]. The cross-covariance matrix from eq. [6] can then be substituted in eq. [2] using C_x as a smoothing matrix (eq. [7]). Creating a new smoothing matrix $R_x = C_x C_{x,J(x)}$, eq. [7] can be rewritten into eq. [8].

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \sigma^k C_x C_{x,J(x)} \quad [7]$$

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \sigma^k R_x \mathbf{G}^{T,k} \quad [8]$$

Do & Reynolds (11) showed that the cross-covariance matrix calculated in eq. [6], is a first-order approximation of the preconditioned covariance matrix times the true gradient. They also determined this is the case for the simplex gradient and preconditioned simplex gradient. Fonseca et al. (12) used this to make an alteration to the EnOpt algorithm for robust optimization using the stochastic simplex gradient (eq. [9]). The steps of this algorithm is shown in a simplified form in Algorithm 1.

$$C_U \nabla J_E(\mathbf{x}^k) = \frac{1}{N_e} \sum_{j=1}^{N_e} \left[\frac{1}{N_p} \sum_{i=1}^{N_p} (E_x[(\mathbf{u}_i - \mathbf{x}^k)(\mathbf{u}_i - \mathbf{x}^k)^T]) \nabla_x J(\mathbf{m}_j, \mathbf{x}^k) \right] \quad [9]$$

Algorithm 1 – Ensemble Optimization

- 
- i. Initial radial settings \mathbf{x}^k are input
 - ii. Ensemble of controls U^k generated from initial settings \mathbf{x}^k
 - iii. Perform Function Evaluation for all ensemble members
 - iv. Compute (approximate) gradient from FE results (eq. [6])
 - v. Calculate step-size parameter σ^k to satisfy $J(\mathbf{x}^k + \sigma^k \mathbf{p}^k) > J(\mathbf{x}^k)$
 - vi. Get radial settings \mathbf{x}^{k+1} for next generation (eq. [2])

Figure 3 shows a sample of an objective function in three dimensions where the x and y axis are controls and z the objective function value. An initial control vector x^k is shown in red, with several perturbed control vectors u_i in black. Using the function evaluation results of each perturbed control vector, the slope of the objective function is determined so a step can be made in the, in this case, descent direction shown by the arrow.

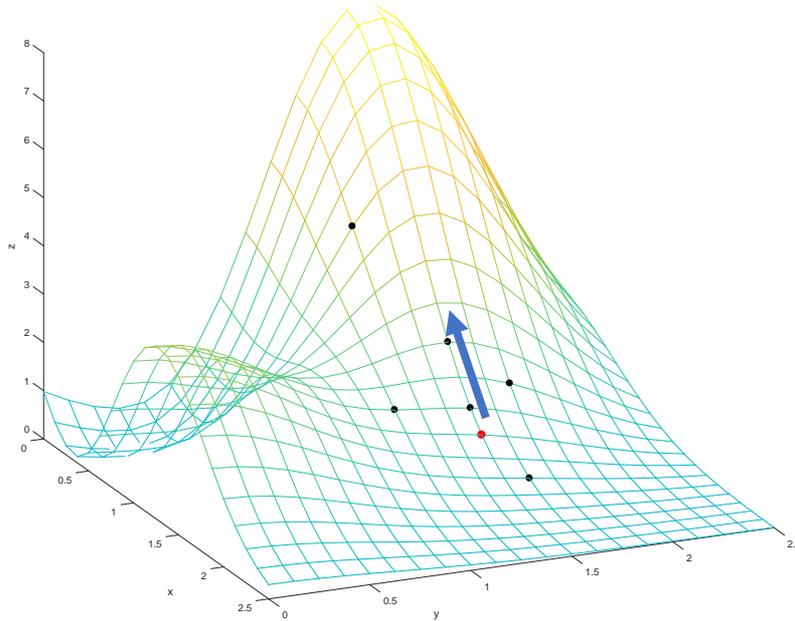


Figure 3 - Sample 3D objective function with initial control vector (red) and perturbed ensemble members (black). The arrow shows the direction of the update step.

2.2.2 Gradient-free optimization

Gradient-free optimization methods find new design variables without the assistance of a search-direction. As with ensemble optimization, most gradient-free methods maintain a population of control vectors. The optimization of the control vector is controlled by either evolution or swarm intelligence. The distinction between evolutionary and swarm methods is the mechanism with which members of the population are adapted. In evolutionary schemes, members of the population are combined (crossover) or perturbed (mutation) to find control vectors with favourable objective function values, the update step of two parent vectors is shown in Figure 4.

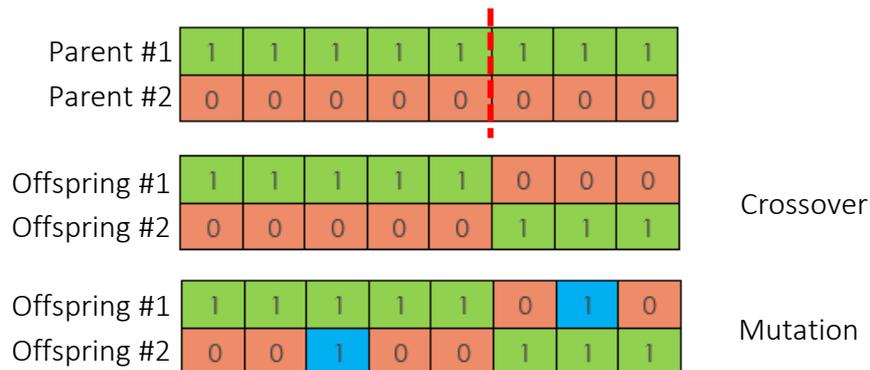


Figure 4 - Genetic Algorithm update steps

Swarm intelligence, on the other hand, uses the position and velocity of population members in the search space to determine a search direction (13). Shown in Figure 5 is the update step of the particle swarm optimization (PSO) algorithm. The control vector for particle i , \mathbf{X}_i^k has a position in the search space and a velocity \mathbf{V}_i^k . The new control vector \mathbf{X}_i^{k+1} is attained using the location of the previously attained best result of the particle $Pbest_i$ and the best result of the entire swarm in the neighbourhood of particle i $Gbest_i$. The distances to $Pbest_i$ and $Gbest_i$ are the velocities \mathbf{V}_i^{Pbest} and \mathbf{V}_i^{Gbest} . These velocities are used to together with \mathbf{V}_i^k to make a step of \mathbf{V}_i^{k+1} (eq. [10]) the new particle location \mathbf{X}_i^{k+1} in eq. [11].

$$\mathbf{V}_i^{k+1} = \mathbf{V}_i^k + c_1\rho_1\mathbf{V}_i^{Pbest} + c_2\rho_2\mathbf{V}_i^{Gbest} \quad [10]$$

$$\mathbf{X}_i^{k+1} = \mathbf{X}_i^k + \mathbf{V}_i^{k+1} \quad [11]$$

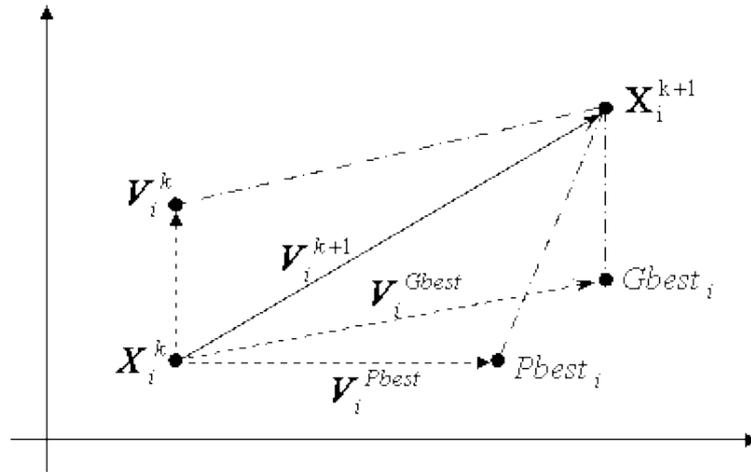


Figure 5 - Particle Swarm Optimization update step (33)

Gradient-free algorithms can be implemented in varying ways, however they adhere to the basic principles of their respective methods when suggesting new control vectors. No information about the objective function is used other than the objective function result of a control vector. This makes gradient-free methods an alternative to gradient-based optimization if computing the gradient is computationally expensive or impossible. If the calculation of the gradient is possible, gradient-based optimization is preferred as the extra information about the objective function will most likely find an optimal value faster. Yang (14) gives an overview of nature inspired optimization methods such as evolutionary and swarm optimization.

2.2.2.1 Covariance Matrix Adaption – Evolution Strategy

Covariance Matrix Adaption – Evolution Strategy (CMA-ES) is a gradient-free optimization method with an evolutionary update scheme (15). This algorithm differs from other gradient-free methods in the sampling strategy, where the sample space is changed dependent on the results of the optimization. Each iteration, a population of λ members are sampled from a multivariate normal distribution around a mean vector \mathbf{x}^k .

$$\mathbf{u}_i \sim \mathbf{x}^k + \sigma^k \times \mathcal{N}(0, \mathbf{C}^k) \quad [12]$$

Here, \mathbf{x}_i is a population member of that iteration, σ_k the step size and C^k the covariance matrix of the multivariate distribution. Each iteration, the values in the covariance matrix are adapted to influence where new values are sampled. This is shown in Figure 6, where the dots are population members and the orange line represents where new population members can be sampled. As more iterations elapse, the covariance matrix will progressively sample points closer to an optimal control setting.

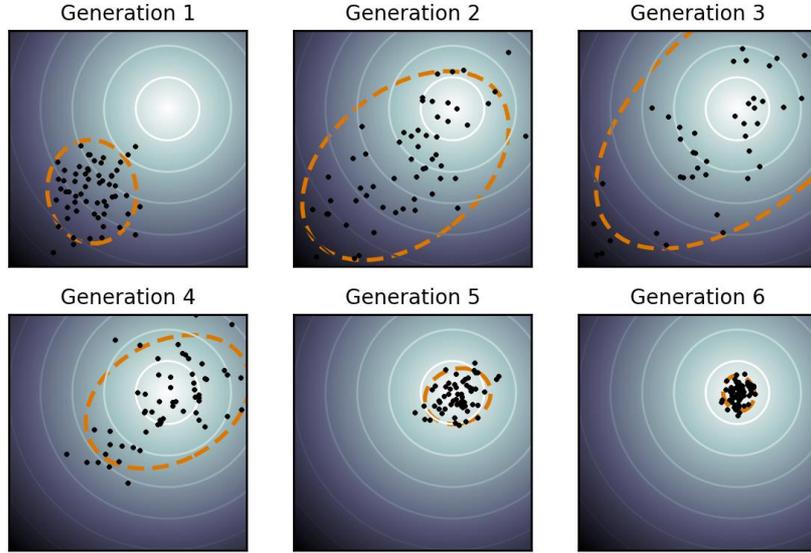


Figure 6 - CMA-ES algorithm search method. Dots are population members, the orange circles show the range given by the covariance matrix for sampling new population members. (From Wikipedia CMA-ES)

The mean vector is comparable to the initial control vector in ensemble optimization and is updated at the end of each iteration. To update the mean, the objective function values of the population members are evaluated and ranked. The best μ solutions are taken from the ranked population members and used to generate a new mean vector.

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \sum_{i=1}^{\mu} w_i (\mathbf{u}_{i:\lambda} - \mathbf{x}^k) \quad [13]$$

Where the weights $w_1 \geq w_2 \geq \dots \geq w_{\lambda}$ sum to one. The CMA-ES has some memory of previous solutions which are retained in the evolution path \mathbf{p}_c . This evolution path is used to update the covariance matrix for the next iteration. Having generated a new mean vector for the next iteration, the covariance matrix and evolution path of the algorithm are updated to reflect the new solution.

$$\mathbf{p}_c^{k+1} = (1 - c_c) \cdot \mathbf{p}_c^k + \sqrt{c_c - (2 - c_c)} \cdot \frac{c_w}{\sigma^k} (\langle \mathbf{x} \rangle_w^{k+1} - \langle \mathbf{x} \rangle_w^k) \quad [14]$$

$$C^{k+1} = (1 - c_{cov}) \cdot C^k + c_{cov} \cdot \mathbf{p}_c^{k+1} (\mathbf{p}_c^{k+1})^T \quad [15]$$

The constants c_c and c_{cov} determine how much information from the previous solutions is retained in the next step, $\langle \mathbf{x} \rangle_w$ is the weighted mean, c_w is $\frac{\sum_{i=1}^{\mu} w_i}{\sum_{i=1}^{\mu} w_i^2}$ and σ^k the step size. The step size governs the rate of change in covariance matrix and scales the solutions samples.

This changes depending on how close sampled solutions are to an optimum value. Step size is increased if few favourable objective function results are found in the current population.

As the optimization proceeds and gets closer to an optimum, step size is decreased to not “over-shoot” this optimum. Step size is determined using cumulative step size adaption eq. [16] & [17]. This ensures steps become smaller when there are small changes between the mean vectors of the current and next iteration. Conversely, large changes between the mean vectors leads to a larger step size. Algorithm 2 shows a simplified form of the CMA-ES algorithm.

$$\mathbf{p}_\sigma^{k+1} = (1 - c_\sigma)\mathbf{p}_\sigma^k + \sqrt{c_\sigma(2 - c_\sigma)}\mu_{eff}C^k \frac{x^{k+1} - x^k}{\sigma^k} \quad [16]$$

$$\sigma^{k+1} = \sigma^k \exp\left(\frac{c_\sigma}{d_\sigma} \left(\frac{\|\mathbf{p}_\sigma^{k+1}\|}{E\|\mathcal{N}(0, I)\|}\right)\right) \quad [17]$$

Algorithm 2 – Covariance Matrix Adaption – Evolution Strategy

- i. Initial radial settings \mathbf{x}^k are input
- ii. Sample control settings using covariance matrix (eq. [12])
- iii. Perform reservoir simulation for control setting population
- iv. Rank solutions according to simulation result
- v. Adapt covariance matrix and stepsize (eqs. [14] – [17])
- vi. Determine new radial settings from best solutions for the next generation (eq. [13])

2.2.2.2 Mixed-Integer CMA-ES

In the optimization of radial well plans, the number of radials used has a significant effect on the objective function. This makes the optimization of radial well design a mixed-integer problem where at least one of the controls to be optimized is bound to an integer value. A modification for CMA-ES is proposed (16) to handle mixed-integer control vectors. Problems arise with integer values if the standard deviation used to sample new controls vectors is small. If both variance and step size are small, new integers might no longer be sampled. To prevent this, an integer mutation is applied to variables where variance is small. This is done with a slight change in the CMA-ES algorithm.

$$\mathbf{u}_i \sim \mathbf{x}^k + \sigma^k \times \mathcal{N}(0, C^k) + S^{int} R_i^{int} \quad [18]$$

A new term is introduced when sampling vectors for the population. The matrix S^{int} has the control vector on the main diagonal with continuous values of the control vector set to zero.

The integer mutation R_i^{int} [19] is performed on variables according to a diagonal masking matrix $I^{R,k}$. The elements of the masking matrix are one where $2\sigma^k C^{\frac{1}{2}k} < S^{int}$, this is the condition suggested in (16), and zero otherwise. Simply, values in the masking matrix are one for integer controls where new integers are unlikely to be sampled.

$$R_i^{int} = I_i^{\pm 1} (R_i' + R_i'') \quad [19]$$

The integer mutation consists of a sign-switching matrix $I_i^{\pm 1}$ with diagonal values of 1 and -1 with equal probability. Matrix R_i' has one of the values unmasked by $I^{R,k}$ set to one, ensuring at most one integer value is mutated. The R_i'' matrix gives variables unmasked by $I^{R,k}$ a value from a geometric distribution according to $p=0.7^{|I_k^R|^{-1}}$. The bold lines in Algorithm 3 show where the MI-CMA-ES algorithm differs from normal CMA-ES.

Algorithm 3 – Mixed Integer CMA-ES

- i. Initial radial settings are input
- ii. **Determine if perturbation is too small to sample new integers**
- iii. Sample control settings using covariance matrix
- iv. **Perform integer mutation on controls determined in ii. (eq. [19])**
- v. Perform reservoir simulation for control setting population
- vi. Rank solutions according to simulation result
- vii. Adapt covariance matrix from best solutions
- viii. Determine new radial settings from best solutions for the next generation

2.2.2.3 Uncertainty Handling CMA-ES

The radial jetting process has no capability to direct the trajectory of the radial. In (3) observed radials deviated from the planned trajectory. Accounting for the uncertainty in radial trajectory ensures the optimization returns a robust optimum. The uncertainty in trajectory can be described by perturbations in the control vector.

This requires a robust optimization method which can handle uncertainty in the input vector. The Uncertainty-Handling CMA-ES (17) is an alteration of the CMA-ES algorithm which can find robust optima with uncertain parameters in the control vector. The UH-CMA-ES uses the rank-based approach of CMA-ES to determine the level of uncertainty in a population of control vectors. After the ranking step of regular CMA-ES, λ_{reev} population members are selected for reevaluation.

$$\lambda_{reev} = f_{pr}(r_\lambda \cdot \lambda) \tag{20}$$

$$f_{pr}(x) = \begin{cases} \lfloor x \rfloor + 1 & \text{with probability } x - \lfloor x \rfloor \\ \lfloor x \rfloor & \text{otherwise} \end{cases} \tag{21}$$

The factor r_λ determines the amount of population members to reevaluate. The first λ_{reev} control vectors from the population are then perturbed according the measure of uncertainty on the respective controls and function evaluations are performed using these members. Each solution is then ranked again, where non-reevaluated members retain their previous rank. A new ranking is obtained by summing the old and new ranks (eq. [21]). Ties in the new ranking are resolved using the absolute rank change $|\Delta_i|$, for reevaluated members rank change is computed using eq. [22] and eq. [23] otherwise. Smaller rank changes are preferred as these solutions are more robust. The combined ranking is used to determine the new mean control vector and to update the covariance matrix. The differences between UH-CMA-ES and CMA-ES are shown in Algorithm 4

$$\text{rank}(J_i^{\text{comb}}) = \text{rank}(J_i^{\text{new}}) + \text{rank}(J_i^{\text{old}}) \quad [21]$$

$$\Delta_i = \text{rank}(J_i^{\text{new}}) - \text{rank}(J_i^{\text{old}}) \quad [22]$$

$$\Delta_i = \frac{1}{\lambda_{\text{reev}}} \sum_{j=1}^{\lambda_{\text{reev}}} |\Delta_j| \quad [23]$$

Algorithm 4 – Uncertainty Handling CMA-ES

- i. Initial radial settings are input
- ii. Sample control settings using covariance matrix
- iii. Perform reservoir simulation for control setting population
- iv. Rank solutions according to simulation result
- v. **Select control vectors to reevaluate (eqs. [20] & [21])**
- vi. **Perturb these control vectors and perform new function evaluations**
- vii. **Create new ranking, combining original and reevaluated rankings (eqs. [21] – [23])**
- viii. Adapt covariance matrix from best solutions
- ix. Determine new radial settings from best solutions for the next generation

2.2.2.4 Mixed Integer Uncertainty Handling CMA-ES

The CMA-ES algorithm modifications are affect different parts of the base CMA-ES. MI-CMA-ES integer mutation is performed during the sampling of a control vector population. The adaptations required for UH-CMA-ES are added later in the algorithm, at the ranking of FE results. This makes it possible to combine the MI-CMA-ES and UH-CMA-ES algorithms for an uncertainty handling CMA-ES with mixed integer mutation. In this MI-UH-CMA-ES, integer mutation is only applied during the sampling of the control vector population and not when reevaluating control vectors during the uncertainty handling step. In Algorithm 5 the orange text shows the mixed integer part of the algorithm, blue is the uncertainty handling operation.

Algorithm 5 – Mixed Integer Uncertainty Handling CMA-ES

- i. Initial radial settings are input
- ii. **Determine if perturbation is too small to sample new integers**
- iii. Sample control settings using covariance matrix
- iv. **Perform integer mutation on controls determined in ii.**
- v. Perform reservoir simulation for control setting population
- vi. Rank solutions according to simulation result
- vii. **Select control vectors to reevaluate**
- viii. **Perturb these control vectors and perform new function evaluations**
- ix. **Create new ranking, combining original and reevaluated rankings**
- x. Adapt covariance matrix from best solutions
- xi. Determine new radial settings from best solutions for the next generation

2.2.3 Optimization in Well Design & Placement

Well placement has a large impact on the production potential of a well. A skilled engineer can plan a well that will produce satisfactorily, though doing this manually is time consuming and options which may perform better could be missed. Numerical optimization is an approach to more exhaustively evaluate possible well designs during this process. Performing an optimization has shown to result in favorable well placement strategies in both gradient-based (18) and gradient-free (19) methods. In addition to location, the gradient free method in (19) also determined an optimum well type when the algorithm had the freedom to place a vertical or nonconventional (multilateral) well.

Designing a trajectory for multilateral wells is more complex than only determining well placement. Finding the optimum location for one or more laterals protruding from somewhere along the main wellbore manually, is extremely challenging when the costs of drilling should be considered. In these cases, applying a numerical optimization scheme is necessary to determine favorable multilateral well designs. Bukhamsin et al. (20) used an unmodified Genetic Algorithm to compare a binary Genetic Algorithm (bGA), where all design parameters are converted to binary values, with a continuous Genetic Algorithm (cGA). Over the course of an optimization run, a stochastic optimization algorithm performs many function evaluations. Yeten et al. (21) used several helper functions such as a neural-network and near-well upscaling to decrease the amount of required function evaluations, while Bouzarkouna et al. (22) applied the CMA-ES algorithm in combination with a local meta-model.

2.2.4 Robust Optimization

The reservoir simulations used in numerical optimization are based on an interpretation of the subsurface. Such an interpretation is made using data from well logs and seismic surveys, though these methods cannot fully describe the reservoir geology. Well logs have limited reach into the reservoir and seismic is limited by resolution in the survey. The resulting reservoir model is therefore subject to geologic uncertainty.

This uncertainty can be accounted for and mitigated using numerical optimization. The optimization algorithms perform numerous reservoir simulations over the course of a run. Using multiple reservoir models for these simulations allows the optimization algorithm to find solutions that take into account multiple geologic scenarios. Optimizing under uncertainty is commonly referred to as robust optimization. These schemes are widely applicable in the oil and gas industry as geological uncertainty is a constant factor when dealing with the subsurface.

Well placement is a field development activity where taking geologic uncertainty into account can significantly influence the prediction of production potential of a reservoir, making this an interesting case for robust optimization. Ramirez et al. (23) compared deterministic and robust optimization results for NPV on several reservoir realizations. The robust optimization resulted in an increased NPV compared to deterministic optimization. As robust optimization is based on the expected outcomes of several realizations, the probability of an outcome can also be weighed in the scheme. Güyagüler and Horne (24) propose a robust optimization method using utility theory to assign an expected utility to a well location based on NPV and the probability of that outcome.

In addition to well placement, other steps in the field development process such as designing a well trajectory, benefit from robust optimization. Schulze-Riegert et al. (25) developed a stochastic optimization method to optimize a straight well path under geologic uncertainty, using the probability of an outcome as weight similar to (24). Well trajectory optimization under uncertainty was also conducted using a gradient-based optimization scheme where the trajectory was not constrained to a straight line (26).

2.3 Well connectivity

In many field applications of numerical reservoir simulation, the grid block dimensions of the model are much larger than the radius of a well and therefore the numerically calculated pressure in a grid block is different than the well bottom hole pressure (BHP). To account for this difference the concept of a Well Index (WI) was introduced. The well index relates the difference of well pressure $p_{w,i}$ and grid block pressure $p_{b,i}$ to the flow rate q_i of well segment i (eq. [24]).

$$WI_i = \frac{q_i^w \mu}{(p_i - p_i^w)} \quad [24]$$

Assuming single-phase flow, this equation gives the WI in grid block i with grid block pressure p_i , well pressure and flowrate are p_i^w and q_i^w respectively. The WI has a significant effect on the accuracy of the subsequent simulation. Several methods have been suggested to calculate the WI. The method of WI calculation used in most reservoir simulators, was introduced by Peaceman (27) and later modified to work with anisotropy (28) and horizontal wells (29). Peaceman's method uses an equivalent radius (eq. [26]). This is the radius at which the analytical solution for radial flow is equal to the gridblock pressure. In an orthogonal grid with anisotropic permeability, Peaceman showed that WI can be calculated as:

$$WI = \left(\frac{2\pi \sqrt{k_x k_y} \Delta z}{\ln\left(\frac{r_o}{r_w}\right) + S} \right) \quad [25]$$

$$r_o = 0.28 \frac{\sqrt{\frac{k_y}{k_x} \Delta x^2 + \frac{k_x}{k_y} \Delta y^2}}{\sqrt[4]{\frac{k_y}{k_x} + \frac{k_x}{k_y}}} \quad [26]$$

This method of WI calculation is used extensively in commercial reservoir simulators and is accurate for wells that are aligned to the numerical grid. The projection method for WI computation is an adaption of the Peaceman equations to account for unaligned well sections. A WI is calculated for each orthogonal axis [27] depending on the projected length of the well (Figure 7). The total WI for a gridblock is the norm of these WIs [29].

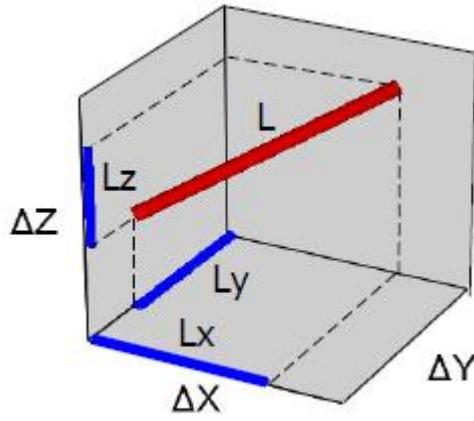


Figure 7 - Projection of well onto orthogonal axis, the WI projection method assigns WI to a gridblock according to the length of the well on each projection L_x , L_y and L_z (30)

$$WI_x = \left(\frac{2\pi\sqrt{k_y k_z} L_x}{\ln\left(\frac{r_{o,x}}{r_w}\right) + S} \right) \quad [27]$$

$$r_{o,x} = 0.28 \frac{\sqrt{\frac{k_y}{k_z} \Delta z^2 + \frac{k_z}{k_y} \Delta y^2}}{\sqrt[4]{\frac{k_y}{k_z} + \frac{k_z}{k_y}}} \quad [28]$$

$$WI_t = \sqrt{WI_x^2 + WI_y^2 + WI_z^2} \quad [29]$$

If a grid block contains multiple well sections, the length of the projection in each direction is summed. These summed values are then used as L_x , L_y and L_z to calculate the WI for that block. For his MSc. thesis Jones Shu did a comparison of these methods for well index calculation for both straight and deviated wells (30). The projection method came close to the reference solution in most cases he studied. When a well section was close to the model boundary the WI calculation began to deviate significantly. The projection method also has difficulty correctly calculating pressure and inflow when multiple wells are in the same grid block.

A concern with the Well Index calculation is that errors could propagate when a well has a short grid block intersection. When an intersection is short compared to intersections in neighboring grid blocks, this should be reflected in the Well Index (weighting). This can be mitigated by using an analytical model to calculate the Well Indices (31). Using a local analytical model is time consuming so the projection method will first be evaluated.

2.3.1 Validation

The optimization is expected to generate laterals that are not aligned to the grid. Therefore, the projection method will be validated to check whether it can accurately compute inflow and bottom hole pressure when a well or radial is not aligned to the grid, as in Figure 8.

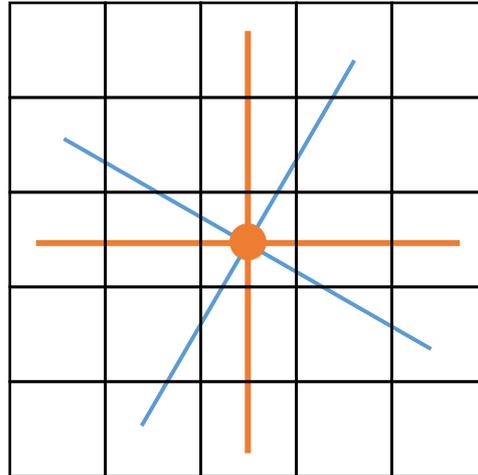


Figure 8 – Representation of numerical grid with 4 aligned radials (orange) and Unaligned Radials (light blue)

To test this, first a vertical well with four radials aligned to the grid was placed in a 1000x1000x250m homogenous box reservoir with 10x10x5m gridblocks and 200 mD permeability. For the base case, four radials aligned to the grid are added to the well. A reservoir simulation is run with a single vertical well with constant BHP boundary condition for the base case and for radials rotated up to 60 degrees. The results should be the same because the permeability is homogenous. The simulation is run over a time period of one year. Injection rate and cumulative injection for the base case with radials aligned to the grid is shown in Figure 9. Each simulation with a set of rotated radials is compared to this base case to check for any deviations. The difference between a rotated set of radials and the base case is shown in Figure 10. For large rotations there is initially a significant difference in injection rates (~25% in the first days), this quickly decreases to less than 2% after around 15 days. For cumulative injection this difference does not exceed 2.5%

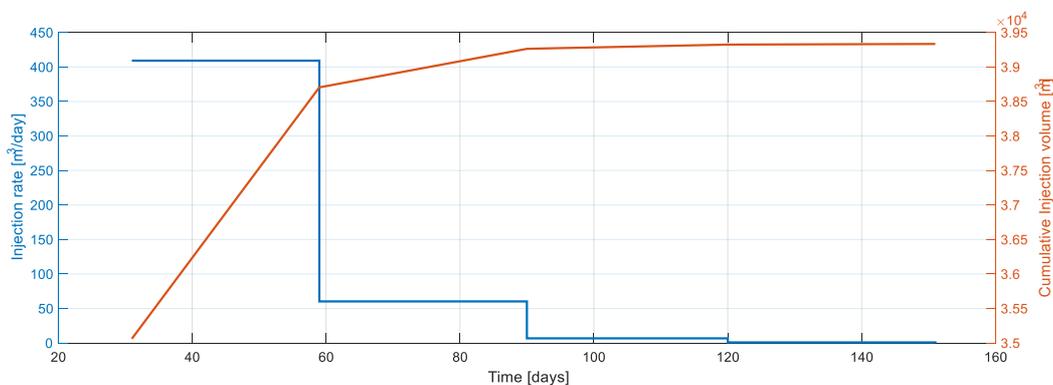


Figure 9 - Injection of Aligned Radials over 160 days (simulation was run for one year). Left axis (blue) shows injection rate, right axis (orange) shows cumulative injection

Note that this comparison between aligned and unaligned radials is only done in a homogenous reservoir, while the simulations will also be performed on heterogeneous reservoirs. Due to the difficulty of verifying a numerical approximation in a heterogeneous grid, a heterogeneous case is not considered. The small errors in the rotated radials indicate that the Well Index weighs the influence of each well section correctly and this is assumed to hold for heterogeneous grids. Radial sensitivity to inclination was examined in (32) and it was determined that radials with a slight upward inclination from horizontal, have a higher injectivity.

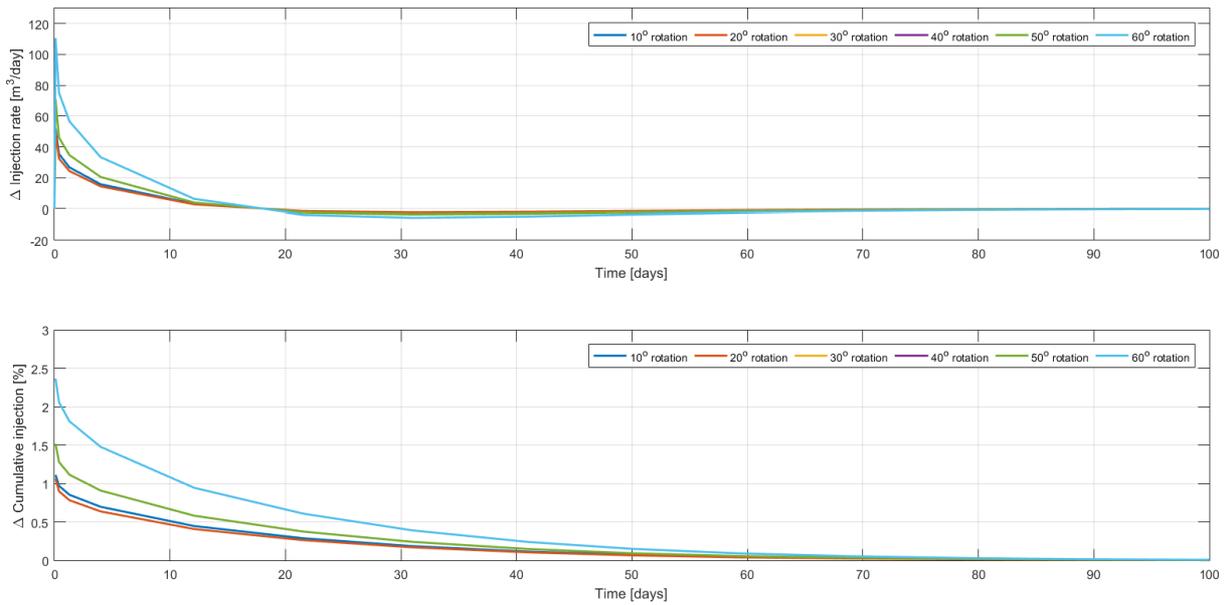


Figure 10 – **Top figure:** Injection rate differences for rotated radials compared to aligned radials. **Bottom Figure:** Percentage difference in cumulative injection unaligned radials compared to aligned radials.

3 Methodology

This chapter describes the methods used for the radial optimization workflow and explores issues which may be encountered when simulating radial jetted wells. To perform an optimization, an objective function which the algorithm will attempt to increase is determined. Radials are parametrized so the optimization algorithm has a method to place radials in the reservoir. The effect of interference of radials in close proximity to each other is evaluated and finally a decision is made as to which optimization algorithm will be used for the workflow.

3.1 Objective Function

An integral component of any optimization method is the objective function. This function describes the property which the algorithm will maximize or minimize. A commonly used metric for optimization in oil and gas is net present value (NPV) (eq. [30]). The NPV is P_t the result of generated revenue minus the cost incurred in year t . In geothermal projects the levelized cost of energy (LCOE) is also used measure the economic value of a project. LCOE gives a relation between energy generated versus the cost of generation and is commonly used to compare the economic efficiency of power generation, the definition is given in eq. [31].

$$NPV = \sum_{t=0}^n \frac{P_t}{(1+r)^t} \quad [30]$$

$$LCOE = \frac{\text{sum of costs incurred}}{\text{energy generated}} = \frac{\sum_{t=1}^n \frac{I_t + M_t + F_t}{(1+r)^t}}{\sum_{t=1}^n \frac{E_t}{(1+r)^t}} \quad [31]$$

With I_t the capital costs, M_t the maintenance costs and F_t fuel costs incurred in year t . E_t the energy generated in year t and r the discount rate. As the optimization is done for a geothermal project, the source of revenue is heat generated by the geothermal well. The reservoir simulation will output a volume of water produced, from which the amount of energy produced can be calculated using [32].

$$Q = m \cdot C_p \cdot \Delta T \quad [32]$$

Where Q is energy generated, m the mass of water, C_p the heat capacity of water and ΔT is the change in temperature. It is assumed that produced water is at reservoir temperature and the temperature inside the reservoir stays constant, water breakthrough is not considered. To determine ΔT the temperature to which the water is cooled is taken to be 20°C.

The cost of a radial is based on the time it takes to jet a radial. The total time required to jet all the radials in the plan proposed by the optimization is multiplied by a day rate. It is assumed that 2.2 radials can be jetted in one day (24 hours). Radials are assumed to be jetted to their maximum length, jetting costs for length are not considered. A day rate of €20000 is used to determine the capital expense (capex) (roughly 9.1×10^3 € per radial). The cost to drill the backbone is not considered. The reservoir simulations are performed with BHP control, meaning

the fuel cost is constant for all well plans. Simplified versions of [30] and [31] will be compared to determine which objective function to use in the optimization and whether this will be a maximization (NPV) or minimization (LCOE) problem. Q_{ww} is the energy generated with warm water from the reservoir in both cases and g_{cost} is the gas cost per kWh which is used to calculate revenue.

$$J_{NPV}(\mathbf{x}) = capex - (Q_{ww} \cdot g_{cost}) \quad [33]$$

$$J_{lcoe}(\mathbf{x}) = \frac{capex}{Q_{ww}} \quad [34]$$

3.2 Radial Controls & Constraints

A radial has several input parameters that control its position in the reservoir. Note that this is also dependent on the well from which the radials protrude, referred to as the backbone. The inputs are shown in Figure 11. While a radial can be any length between 1-100m, in practice a radial is generally jetted to its full length of 100m if possible. Therefore radial length will not be considered as a variable to optimize.

3.2.1 Radial Parametrization

When jetting a well, it is common to jet multiple radials at the same kick-off depth. For this reason, radial controls are parametrized with jetting multiple radials per kick-off in mind. These controls are 1. kick-off depth, 2. number of radials, 3. azimuth, 4. inclination, 5. radial length and 6. the window, as in eq. [35] & Figure 11. The window parameter determines how multiple radials are distributed when around the well bore (Figure 12). The window parameter w defaults to a full circle in the parametrization if the condition in eq. [37] is met. The control vectors for each kick-off \mathbf{s}_r combine to the complete control vector for a given well plan.

$$\mathbf{s}_r = [KOD_r \ n_r \ az_r \ in_r \ len_r \ w_r] \quad r = 1, \dots, kicks \quad [35]$$

$$\mathbf{u} = [\mathbf{s}_1 \ \mathbf{s}_2 \ \dots \ \mathbf{s}_{kicks}] \quad [36]$$

$$w_r(w) = \begin{cases} w, & w < 2\pi(1 - \frac{1}{n_r}) \\ 2\pi, & w > 2\pi(1 - \frac{1}{n_r}) \end{cases} \quad [37]$$

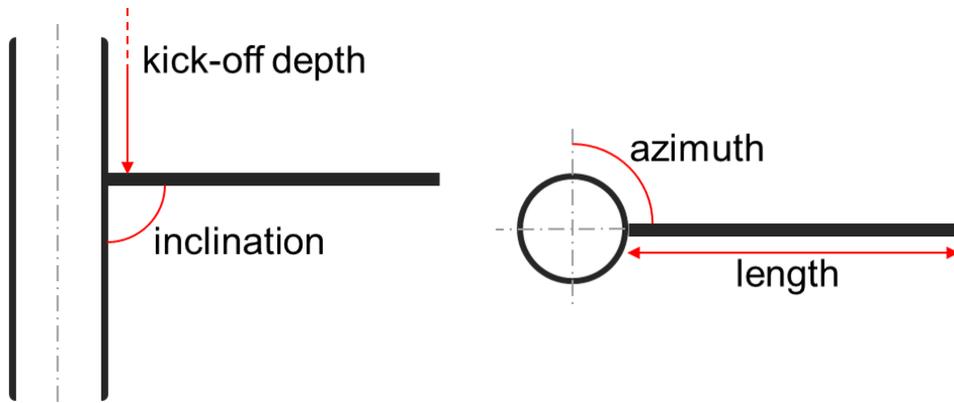


Figure 11 - Radial controls for kick-off.

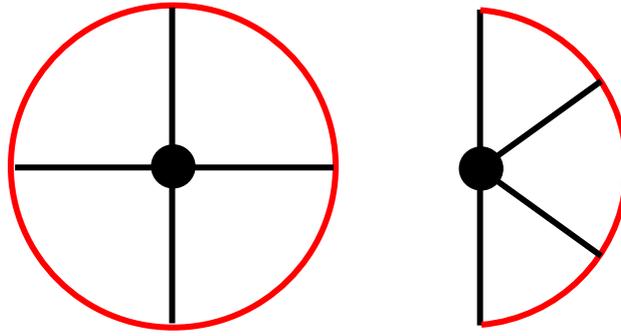


Figure 12 - Radial Window Parameter, four radials with window 2π (left) and π (right) both with azimuth pointing

3.2.2 Radial proximity

The parametrization of radials allows for large numbers of radials to be jetted at a kick-off depth, as well as in close vertical proximity. Having multiple jetted radials close to one another is expected to decrease the individual contribution of each radial. As the cost of the radial well plan is dependent on the number of jetted radials, the optimization should identify when an additional radial is not worth the investment.

To evaluate the effect of adding radials to a kick-off, a vertical well in the center of a homogenous reservoir with a single kick-off in the middle of the reservoir is considered. Radials are added to the kick-off which are horizontally evenly distributed. The result is shown in Figure 14. As expected the inflow per radial decreases as the number of radials in the kick-off increases, though the decrease is not constant. The number of radials which can profitably be jetted depends on the flow characteristics of the reservoir rock, though jetting a large number of radials will result in diminishing returns. A second order polynomial was fitted to this data, which clearly shows the relative decrease in production when adding radials. From this polynomial it was found that an additional radial has about 3% less production than the previous radial.

The increase in production per radial shows irregularities. Repeating the simulation in the same homogenous reservoir with a higher grid resolution reduces these irregularities (Figure 13). Note that the production increase from zero to one radial was not considered in the high resolution case.

In addition to horizontal proximity of radials in the same kick-off, vertical distance between radials is also expected to influence the production of the well. This is examined by evaluating a homogenous reservoir with a vertical well which has two kick-offs. One of these kick-offs has a fixed kick-off depth while the other is incrementally moved away, increasing the vertical distance between the two. Both kick-offs have 4 jetted radials. In addition to the change in vertical distance, the fixed set of radials also has a varying azimuth.

This creates an offset between the two sets of radials and should decrease the influence of the nearby kick-off as distance between radials increases. Figure 15 shows that jetting radials with offset has a significant positive effect on production even with small difference in azimuth. The results for 15° and 30° offset both had higher production than the 0° and 45° cases. This difference was $\sim 3.2e5 \text{ m}^3$ at each data point, the simulation data and data corrected for this factor are plotted. The corrected data matches well with the other simulation results. This error is attributed to numerical errors in the simulations

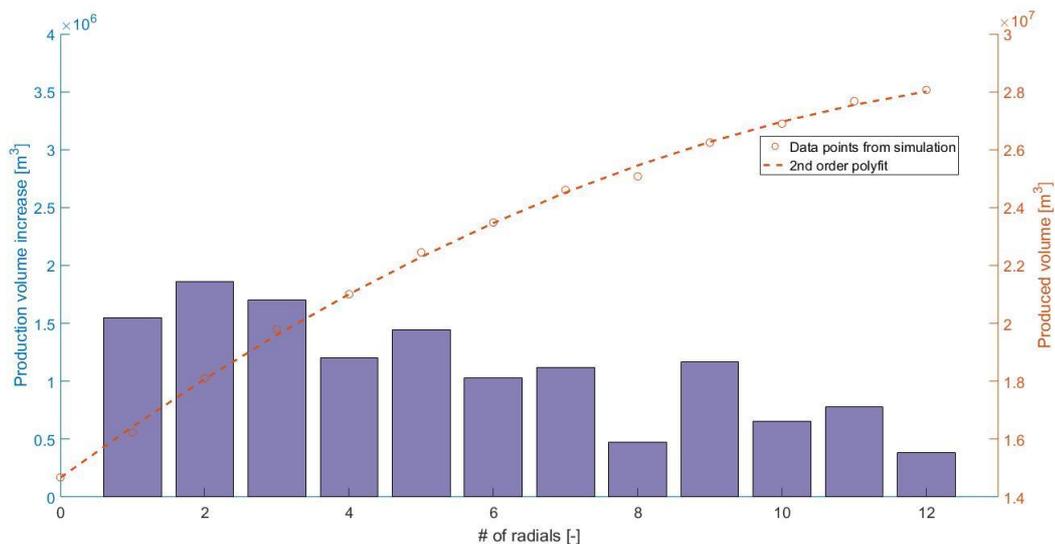


Figure 14 – Production increase with increasing number of radials. The left axis corresponds to the bar chart showing increase in production for each added radial. Left axis shows the produced volume of the well including radials with a second order fitted polynomial (dashed line) through simulation data (points).

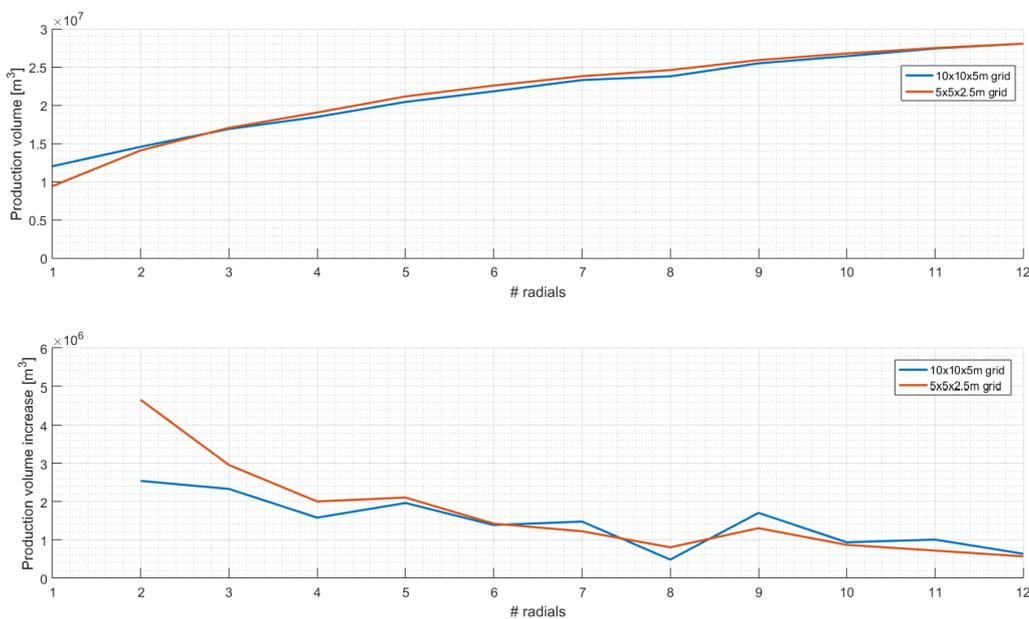


Figure 13 - **Top figure:** Produced volume comparison for an increasing number of radials in a low (blue) and high (orange) resolution grid. **Bottom figure:** Production increase per radial starting from 1 to 2 radials in a low and high resolution grid.

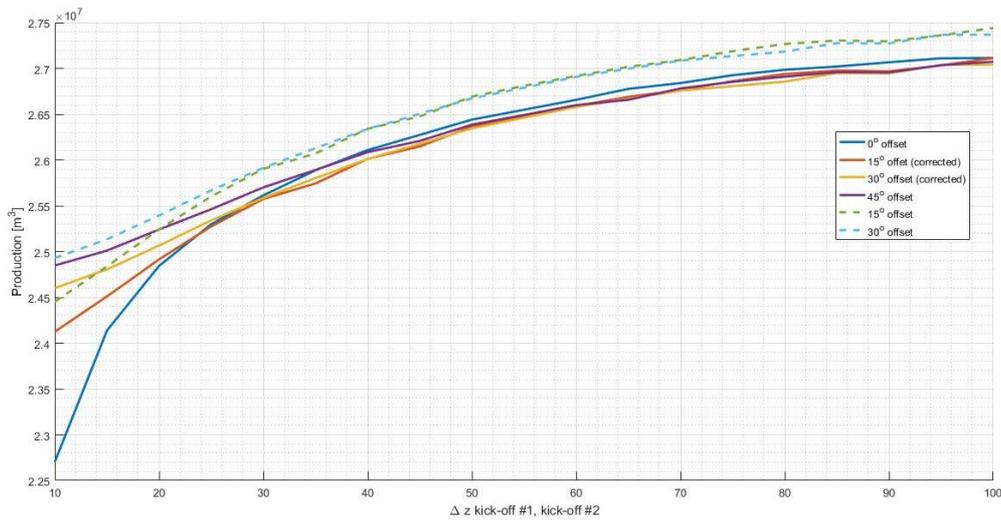


Figure 15 – Production of radial with increasing distance between kick-offs. Solid lines show production for varying radial offset. Production for 15° and 30° offset is corrected by a constant to align with other data, the dashed lines show this data without correction.

3.3 Optimization Test Case

A choice has to be made whether the workflow will use a gradient-based or gradient-free method. The evaluated gradient-based optimization is EnOpt and gradient-free optimization is CMA-ES. Both these methods are compared in a simple reservoir with one vertical well and two high permeability layers while the rest of the reservoir is restrictive to flow (Figure 16). The expected result is that radials will be placed in the high permeability layers. In this test case azimuth and inclination are not considered since the high permeability layers extend over the entire reservoir. The only controls are kick-off depth, number of radials and radial length. All laterals are aligned to the grid, making the maximum number of radials per kick-off 4. Azimuth and inclination are not considered. When lateral is added to a star this is done in the same pattern for each case. The first lateral always points North, the second South, then East and finally West is added if the kick-off point has four laterals. This was done so that if the same lateral controls were simulated twice, the same objective function value is returned. Adding laterals randomly was considered but this caused two sets of the same lateral controls to produce different volumes. In this case, one lateral might be pointed North and another at a different depth South, in which case these wells are not interfering with each which could cause a higher produced volume. Neither kick-off starts in a high permeability layer, and both have the same distance to one of the high permeability layers.



Reservoir properties

- 1000m x 1000m x 100m
- 2x 20m high perm. layer
- $K_{high} = 1000 \text{ mD}$
- $K_{low} = 10 \text{ mD}$
- $K_v = 50 \text{ mD}$
- $\phi_{high} = 0.21$
- $\phi_{low} = 0.07$

Figure 16 – Representation of test case reservoir, high permeability layers are shown in green. The initial kick-off locations with 2 radials are shown in orange.

3.3.1 Results

Both methods were run for 50 iterations with an ensemble size of 11 members for 10 controls. At each iteration, the function evaluations for all ensemble and population members are saved to create a map of the objective function. The results of the optimization are compared in Figure 17. CMA-ES optimization results in a higher objective function value in this test case. Examining the cumulative lateral length it is clear that EnOpt sets the number of radials in one of the kick-offs to zero, trapping the optimization in a local optimum. Figure 18 shows the number of radials in each ensemble member for EnOpt and every population member in CMA-ES. At 4 radials EnOpt finds the highest NPV, when the number of radials increases NPV only decreases indicating a negative gradient. The CMA-ES algorithm has no problems finding controls vectors where more than 4 radials are used and result in a higher NPV. The CMA-ES algorithm is chosen to perform the optimization in the workflow.

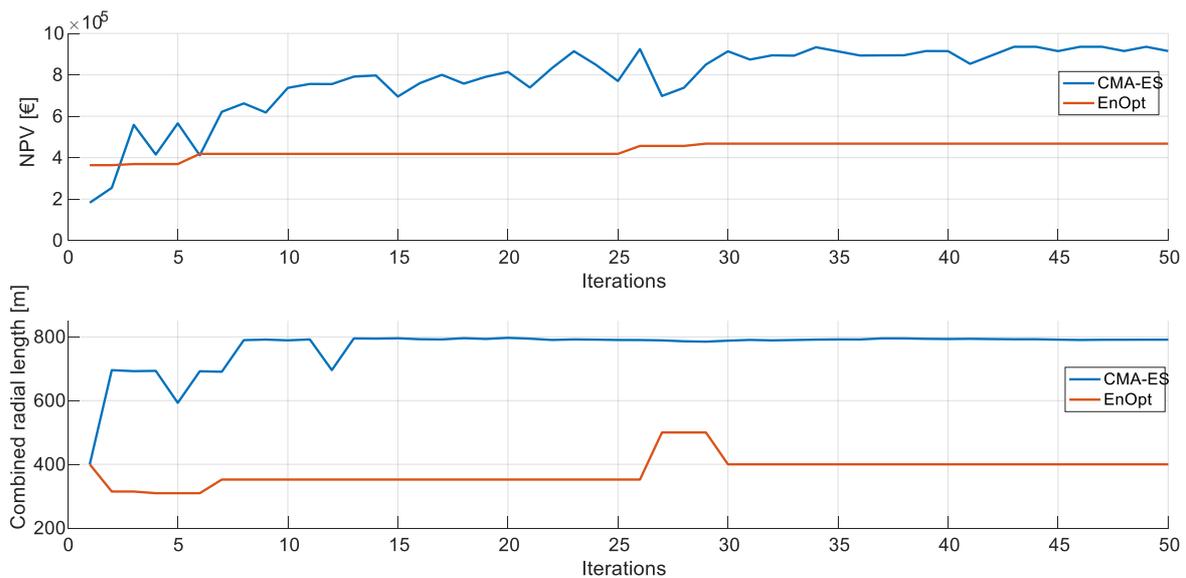


Figure 17 – **Top figure:** NPV comparison of test case optimization for CMA-ES (blue) and EnOpt (orange). **Bottom figure:** Comparison of total jetted radial length for test case optimization for CMA-ES and EnOpt.

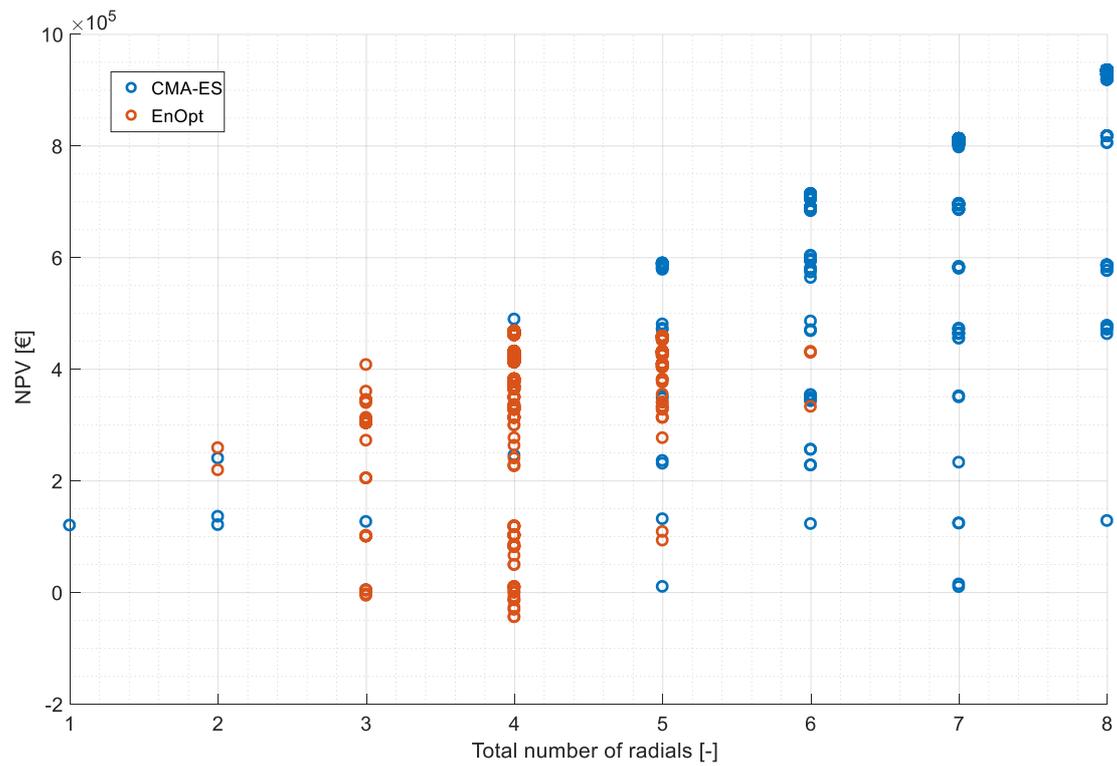


Figure 18 – Comparison of NPV for number of radials of all population data from test case optimization using CMA-ES (blue) and EnOpt (orange).

4 Optimization workflow

The optimization algorithm aims to find the lateral settings which results in highest objective function value. These settings control the attributes of laterals such as kick-off depth and radial length and will influence the production of a well. Having determined an optimization method, the algorithm is incorporated into a workflow which utilizes the algorithm to optimize a radial well plan in realistic field cases. The base CMA-ES algorithm will be compared to MI-CMA-ES and UH-CMA-ES in these cases. The constants used in the different CMA-ES algorithms are the suggested values found in (15) and (17), which are also listed in Table 2.

Table 2 – Constants used in CMA-ES optimization. The r_λ term is used to determine number of reevaluated control vectors for UH-CMA-ES.

Constant	Value
λ	$\lfloor 4 + 3 \cdot \ln(N_x) \rfloor$
μ	$\lfloor \frac{\lambda}{2} \rfloor$
μ_{eff}	$\sum_{i=1}^{\mu} w_i^2$
c_σ	$\frac{\mu_{eff} + 2}{N_x + \mu_{eff} + 3}$
c_c	$\frac{4}{N_x + 4}$
c_{cov}	$\frac{2}{(N_x + \sqrt{2})^2}$
d_σ	$1 + 2 \cdot \max[0, \sqrt{\frac{\mu_{eff} - 1}{N_x + 1}} - 1] + c_\sigma$
r_λ	0.3

4.1 Workflow Cases

To gauge whether the optimization can output well plans with optimally jetted radials, the optimization algorithms are tested on three field cases. The cases increase in reservoir complexity so the effectiveness of the algorithms can be compared in various reservoir settings. This also allows some optimization parameters to be ignored in some cases. Table 3 gives a quick overview of the cases and their properties. The reservoir simulation for each reservoir is performed for a time of one year. All backbone wells have a 5" radius, radials are 2" radius.

Table 3 - Petrophysical Properties of Workflow Cases

Reservoir	K_h [mD]	K_v [mD]	Φ [%]	Reservoir temp. [°C]	Notes
Klaipeda	250-500	150-250	14-21	80	Horizontally homogenous, thin layers
Doublet	1000-1500	500	15-20	80	Synthetic model with geothermal doublet
Fractured	0,1-10.000	0,1-10.000	5-25	80	Faulted block in reservoir with fractures near the faults

4.1.1 Klaipeda

The Klaipeda geothermal site in Lithuania (1) is a relatively small geothermal reservoir with thin high permeability sand layers separated by shales. The average gridblock in the reservoir is 10x10x2.6m, reservoir dimensions are 100x100x59 gridblocks. A subsurface model based on this reservoir with horizontally homogenous layers (32), is used as one of the optimization cases and shown in Figure 19. The temperature in the Klaipeda reservoir is ~40°C, this is low for a geothermal reservoir. For the optimization, warm water temperature is assumed to be 80°C to make the case less marginal economically. Low NPV might result in radials not being jetted to save costs.

The reservoir has a single production well along which radials can be jetted. In most cases radial wells are used in injection wells, the choice to use a production well in this reservoir was to simplify the script. Using a production well, the produced volume can be read directly from the simulation result, allowing the workflow to run without adaptations to the code. As injection and production are essentially the same process reversed, an increased production will indicate an increase in injectivity. As there is only a single production well, the reservoir has a constant pressure boundary constraint, applied using a pore volume multiplier at the edges of the reservoir.

As the layers are horizontally homogenous, azimuth is not considered in this optimization. The parameters to optimize in this case are the kick-off depth and the number of radials per kick-off. For the uncertainty handling algorithm inclination is the uncertain parameter. In addition to the three discussed CMA-ES algorithms the MI-UH-CMA-ES algorithm is also tested in this reservoir.

Table 4 – Facies & properties in Klaipeda reservoir case

Facies	K_h [mD]	K_v [mD]	ϕ [%]
Coarse sand	500	150	21
Fine sand	250	150	14
Clay	50	50	7

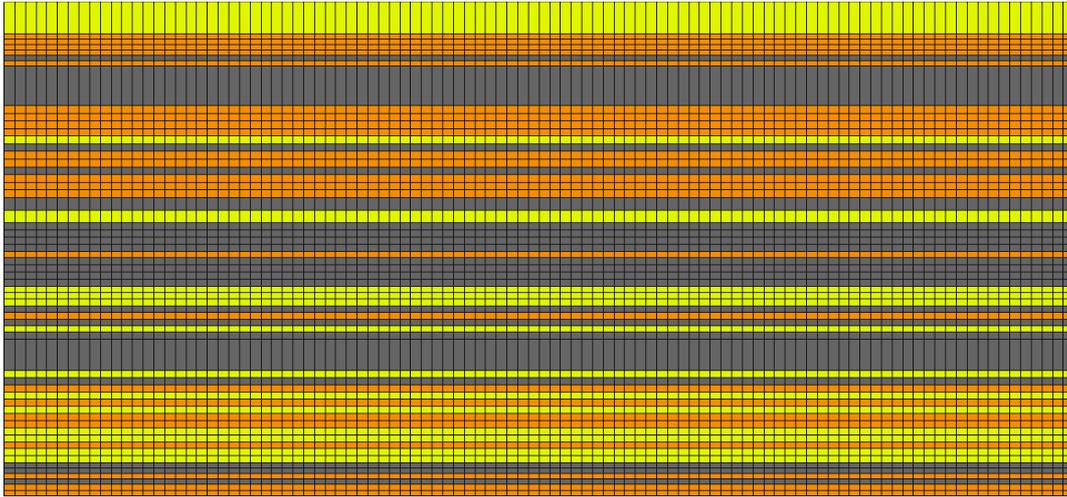


Figure 19 – Vertical slice of Klaipeda reservoir. Colors correspond to those in Table 4.

4.1.2 Doublet

Geothermal reservoirs are generally exploited using a doublet, where water is heated as it flows through the reservoir from an injection to a production well. As these doublets are common, the optimization workflow should be able to work in these systems. A synthetic reservoir with features oriented along the flow direction (Figure 20) is used to apply the optimization in a doublet. The reservoir is comprised of 15x15x2.5m gridblocks in a 100x200x40 reservoir. The features were generated such that sections of both wells are connected with high permeability rock. Both wells are the same in terms of diameter and have a 45° inclination, with the injector and producer having an azimuth of 0° and 180° respectively. The wells are controlled using a BHP constraint, making this a balanced system. Jetting radials in a single well is not expected to be beneficial to NPV, as the BHP constraint on the injector will ensure the pressure near the producer will remain the same. The expected result is that radials are not beneficial in this reservoir when jetted in a single well. The optimization algorithm should identify this and not jet radials. The azimuth and window are used as optimization parameters in addition to number of radials. The uncertain parameter is inclination.

Table 5 – Facies & properties Doublet reservoir

Facies		K_h [mD]	K_v [mD]	ϕ [%]
	Coarse sand	1500	500	20
	Fine sand	1000	500	15
	Clay	50	50	5

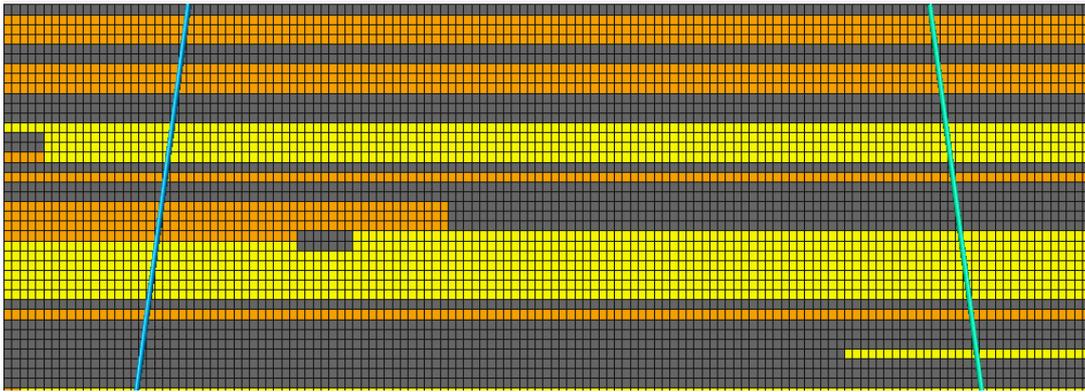


Figure 20 – Vertical slice of Doublet reservoir. Colors correspond to Table 5. The vertical scale is multiplied 5x, the left line shows the producer, the right the injector.

4.1.3 Fractured

The final case evaluated by this workflow is a reservoir based on a geothermal site in the Netherlands in a fractured reservoir (dual porosity case). The geological setting is based on the Californië geothermal site near Venlo in The Netherlands. A fault separates the reservoir into two sections. Fractures are present in the reservoir and occur with greater frequency near the faults (Figure 21 & Figure 22). The average gridblock is 46x50x21m, the reservoir dimensions are 111x61x41. The expected result of the optimization is that radials are placed closer to the fractures where permeability is higher. In this reservoir only the injection well is considered for radial jetting, this well is vertical and gets closer to the fractures as depth increases. The production well has been drilled in a section of the reservoir which easily flows, reducing the need for radials in this well. The optimization controls are kick-off depth, number of radials, azimuth and window, uncertainty is present in inclination.

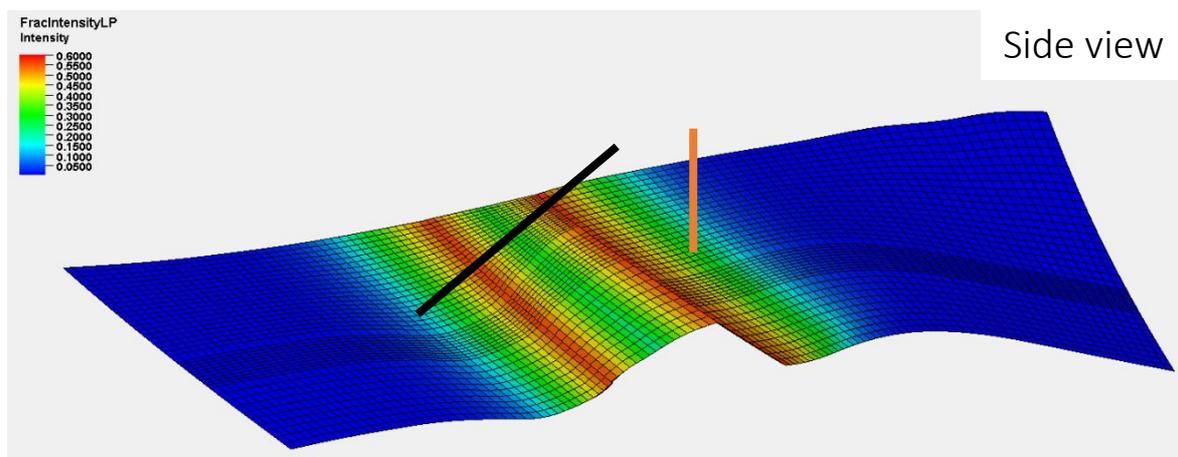


Figure 21 – Vertical slice of Fractured reservoir showing fracture intensity. Injector well is shown in orange, producer in black.

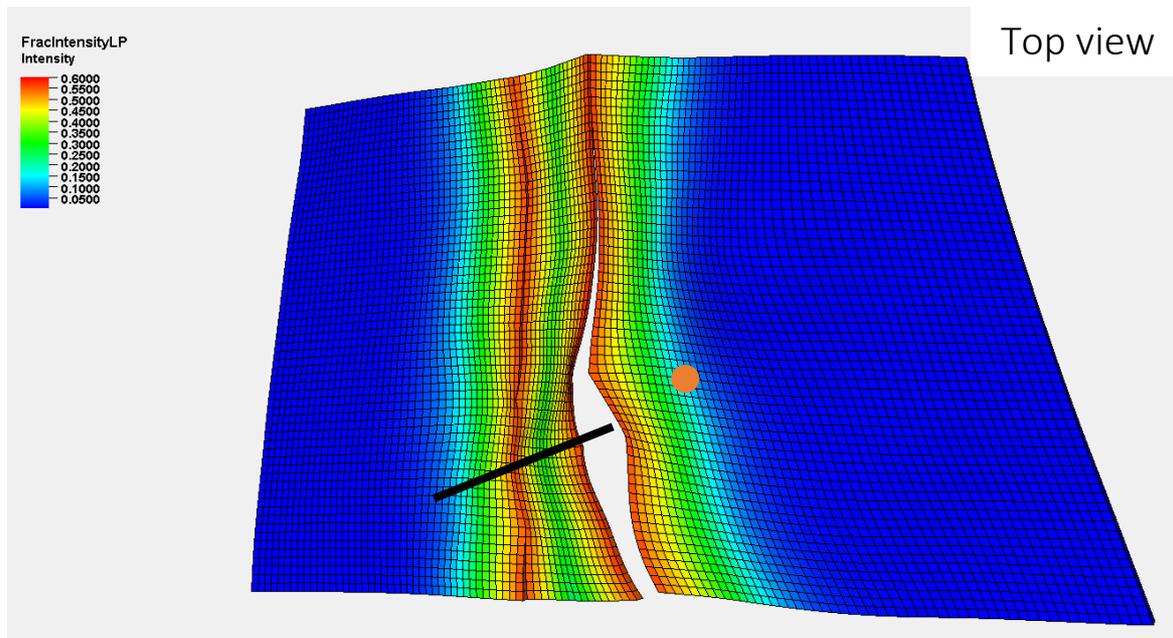


Figure 22 – Top view of Fractured reservoir showing fracture intensity. This is a horizontal slice roughly halfway down the reservoir. The black line represents the producer, the orange dot shows injector location.

4.2 Results

This chapter will evaluate the results gathered from the various optimization runs in the three reservoirs. Optimization results are shown in three plots for each reservoir. The first plot describes kick-off depth as a location along the backbone i.e. a value of .5 means the kick-off is jetted halfway along the backbone. Next is the number of radials jetted per kick-off, the bar colors correspond with the kick-off colors, the final plot shows objective function value.

4.2.1 Klaipeda

All CMA-ES algorithms are able to determine a radial jetting plan that increases the objective function. Figure 23 shows the NPV for the control vector \mathbf{x} at the end of each iteration. There is significant difference between the results generated by each CMA-ES, with the UH-CMA-ES finding the highest NPV in this reservoir. The amount of radials to jet is also the lowest in the well plan, 13 compared to 19 and 17 using the CMA-ES and MI-CMA-ES algorithms respectively. This would lead to a 5.4×10^4 € cost reduction compared to CMA-ES, which is a small difference compared to the resulting NPVs. Indicating the UH-CMA-ES is placing the radials more favorably. Examining the kick-off depths at the end of each optimization shows the UH-CMA-ES spreads out the kick-offs over the length of the backbone more compared to CMA-ES and MI-CMA-ES. Over the course of the optimization the UH-CMA-ES algorithm is also better at avoiding placing radials in low permeability layers.

The common trend in all these algorithms is placing radials toward the bottom of the reservoir. This is not unexpected as the reservoir has the largest section of high permeability rock in the bottom, allowing for the most drainage. This is especially clear in the MI-CMA-ES and UH-CMA-ES algorithms.

The MI-CMA-ES is essentially combining three kick-offs at the bottom of the reservoir by placing them in extreme close proximity and increasing the number of radials this way. This leads to a few radials being placed on top of one another, decreasing the effectiveness of individual radials. This might be avoided in UH-CMA-ES with uncertainty in the inclination. If kick-offs are close vertically, a change in inclination may result in radials being even closer to each other, resulting in a lower NPV and thus making that well plan less influential to future iterations.

While the MI-UH-CMA-ES optimization results in a similar NPV to UH-CMA-ES, a greater number of radials jetted in this well plan. Furthermore, two kick-offs are placed at almost the same depth. Indicating this NPV is likely a local optimum.

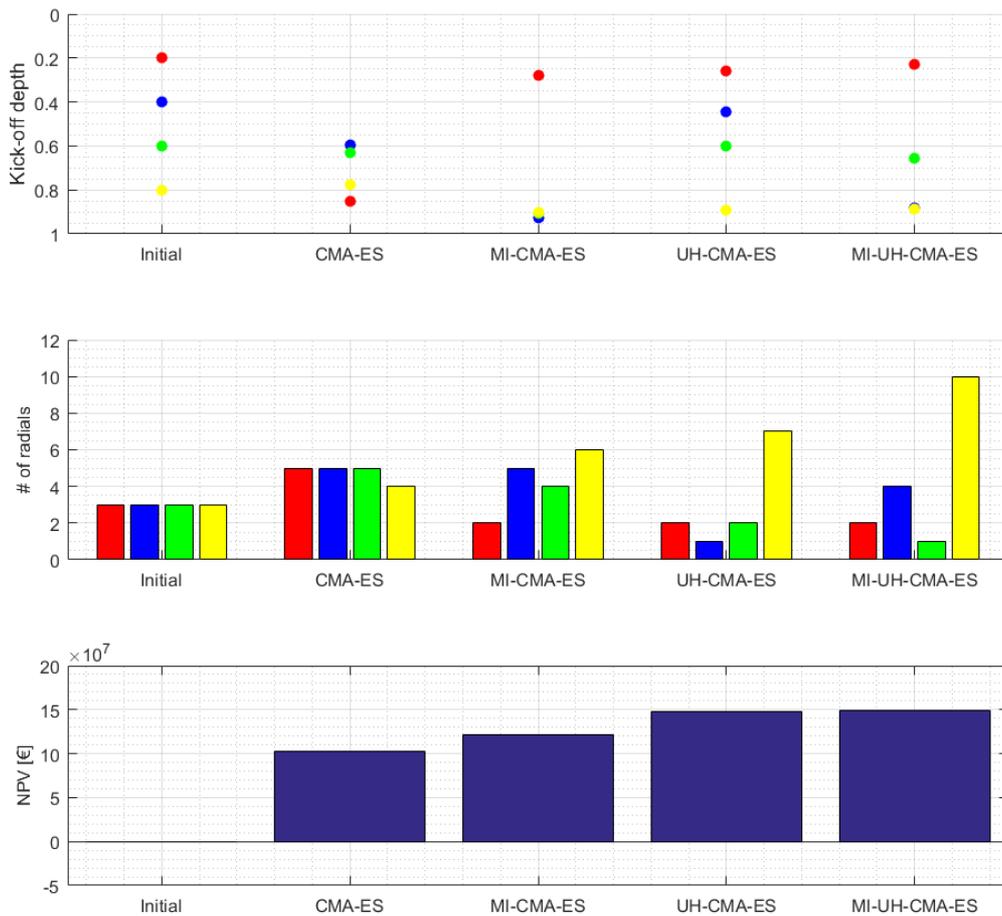


Figure 23 – Klaipeida optimization results, the first entry in each figure shows results for the optimization initial control vector. **Top figure:** Kick-off depth along backbone. **Middle figure:** Number of radials in each kick-off. **Bottom figure:** NPV for control vector returned after optimization. The initial control vector NPV is close to 0

4.2.2 Doublet

Optimization in the Doublet reservoir returns the same well plan for each algorithm (Figure 24), in which no radials are present. The difference between algorithms is observed in the number of iterations required for the solution converges to zero radials. Due to the highest NPV being found without radials, the number of radials has a significant effect on NPV. The MI-CMA-ES should therefore perform best in this case. CMA-ES and UH-CMA-ES, not having the mixed integer mutation, take longer to set the number of radials to zero. Both these algorithms have the same sampling strategy to generate control vectors, though it takes the CMA-ES algorithm longer to remove all radials from the well. This is attributed to the optimization being stochastic. Examining the population of control vectors in the first 4 iterations shows the CMA-ES algorithm starts with more radials compared to UH-CMA-ES, before reducing this number to 0.

To determine the influence of radial jetting costs on NPV in this reservoir, the costs of radials are compared to the difference in NPV for the three algorithms. The results from iteration 4 are used for this comparison. For this iteration the CMA-ES algorithm has 3 radials, MI-CMA-ES has 0 and UH-CMA-ES has 1 radial. Jetting a single radial costs $\sim 9.1 \times 10^3$ € and jetting 3 would add $\sim 2.7 \times 10^4$ € to capex. The difference in NPV between MI-CMA-ES and UH-CMA-ES is $\sim 3.8 \times 10^3$ €, for MI-CMA-ES and CMA-ES this difference is $\sim 9.8 \times 10^3$ €. This shows the radials are contributing to inflow into the reservoir, though not enough to make jetting radials economically viable.

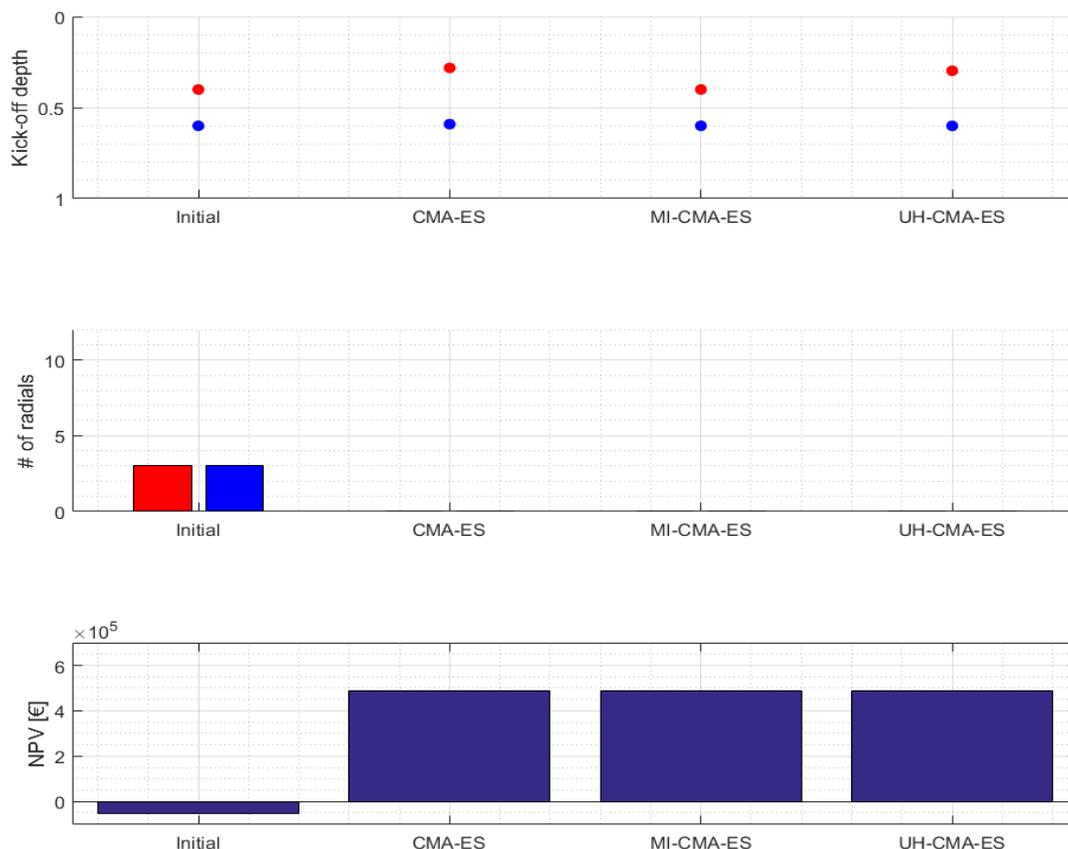


Figure 24 - Doublet optimization results, the first entry in each figure shows results for the optimization initial control vector. **Top figure:** Kick-off depth along backbone. **Middle figure:** Number of radials in each kick-off. **Bottom figure:** NPV for control vector returned after optimization.

4.2.3 Fractured

The fractured reservoir does not feature discrete sections of high or low permeability, as is the case in the previous reservoirs. This would mean less discontinuities in the objective function, making for an easier optimization case. The obtained results (Figure 25) reflect this, the three evaluated methods find similar NPVs after 20 iterations indicating a similar performance. There are some discrepancies between the well plans, especially in the number of radials per kick-off. Though each algorithm places most of the radials near the bottom of the backbone, which is the expected behavior as this is where permeability is highest. The differences between the well plans indicate these are local optima.

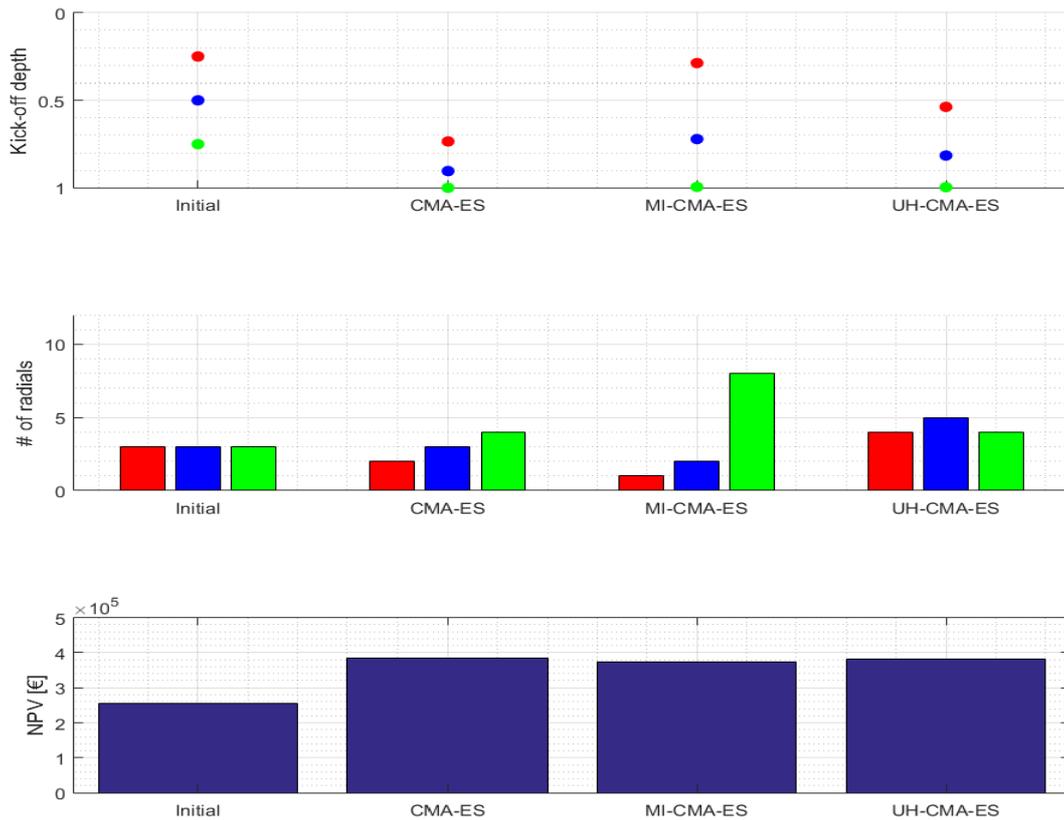


Figure 25 - Fractured optimization results, the first entry in each figure shows results for the optimization initial control vector. **Top figure:** Kick-off depth along backbone. **Middle figure:** Number of radials in each kick-off. **Bottom figure:** NPV for control vector returned after optimization.

4.3 Discussion

In two out of the three cases, the optimization algorithms find similar NPV results, the exception being the Klaipeda reservoir. In this reservoir, both the algorithms incorporating the uncertainty handling scheme performed significantly better than the basic CMA-ES and MI-CMA-ES. The re-evaluations of the objective function with uncertainty on input parameters, allow the uncertainty handling algorithms to identify in which kick-off more radials should be placed and increasing NPV. Even with uncertainty on the inclination, the MI-UH-CMA-ES algorithm places two kick-off at effectively the same depth, without significantly impacting the NPV of the well plan.

Having multiple kick-offs in close proximity is expected to have a negative effect on NPV due to interference between radials. In addition, several kick-offs in well plans feature small numbers of radials. Whether jetting these radials contribute enough to NPV to offset the jetting cost is unknown. These factors may limit the algorithms in finding the global optimum.

Kick-offs in close proximity and with small numbers of radials are mostly observed in the Klaipeda reservoir, so these will be re-evaluated using an engineered well-plan with the exception of the CMA-ES result. Kick-offs within 20m of another or combined, using the average depth of the combined kick-offs for the new kick-off. Kick-offs with less than two radials are removed, UH-CMA-ES has no specific method for integer optimization and the mixed integer mutation is only ever performed on one integer value. Kick-offs with small numbers of radials could therefore be “artefacts” of the optimization. Removing these will simplify the well plan and lead to a lower capex.

Table 6 - Comparison optimization results and engineered solution Klaipeda reservoir

Algorithm	Optimization NPV [€]	Engineered NPV [€]
MI-CMA-ES	1.22×10^8	1.35×10^8
UH-CMA-ES	1.48×10^8	1.17×10^8
MI-UH-CMA-ES	1.49×10^8	1.33×10^8

Comparing the results in Table 6 shows that results generated from the uncertainty handling scheme are better than the proposed engineered well plan. The increase of NPV with the engineered solution compared to the MI-CMA-ES algorithm can be attributed to the combination of kick-offs. Combining three kick-offs eliminated a lot of radial overlap which has a significant positive effect on NPV. It is still unknown what the global optimum is, while UH-CMA-ES and MI-UH-CMA-ES find similar NPVs, the well plans have differences between kick-off depth and number of radials.

The Doublet reservoir has good connection between injector and producer. Jetting radials in this reservoir did not lead to increased NPV, which all algorithms were able to identify and subsequently removed radials from the well plan. Adding radials to the injector leads to a small increase in pressure in the reservoir as the BHP constraint is quickly satisfied. As such the pressure near the producer does not change significantly. This leads to small production increases which do not offset radial cost, making this reservoir a case in which there is no benefit to jetting radials. Which is confirmed by the optimization results.

Azimuth and window in the fractured reservoir were also expected to have an impact on NPV, where radials directed toward the production well and closer to the fractures are expected in the optimization result. Azimuth and window did get iteratively changed in CMA-ES and UH-CMA-ES, whereas the result of MI-CMA-ES is dominated by a kick-off with 8 radials near the bottom of the well with a window term of 2π . To judge whether the azimuth and window have an effect on NPV with CMA-ES and UH-CMA-ES, the well plans are simulated again with a window term of 2π in combination with the azimuth from the well plan, as well as 0 azimuth.

Table 7 shows the results of these simulations. Changing the azimuth and window did not lead to significant changes in NPV. In fact the differences when comparing the optimization results against the engineered solution are generally less than 2%, which was the error margin attributed to numerical errors. While the optimization results are higher, the effect azimuth and window have on NPV seems limited.

Table 7 - Comparison Optimization results and engineered solution fractured reservoir case

Algorithm	Optimization	w = 2π az from opt.		w = 2π az = 0	
		NPV [€]	% opt. result	NPV [€]	% opt. result
CMA-ES	3.85×10^5	3.82×10^5	99.2%	3.78×10^5	98.2%
UH-CMA-ES	3.73×10^5	3.60×10^5	96.5%	3.66×10^5	98.1%

Examining the gridblocks in which the backbone and radial wells are located shows that a radial does not protrude far from the backbone in terms of gridblocks. This offers an explanation as to the limited effect of azimuth and window on NPV. If a radial extends the, in this case horizontal, reach of a well by one or two gridblocks, this does not have a significant effect on injectivity beyond increasing the WI in that area. The number of radials has a more pronounced effect as this directly influences the WI of the well. This of course depends on gridblock dimensions compared to radial length, but this should be taken into account in the optimization.

5 Conclusion & Future Work

The stated goal of developing an optimization workflow for jetted radial wells which can handle uncertainty on the inputs has been met. The algorithms present in the workflow are all able to find control vectors that increase NPV from a starting control vector. The most important factors on NPV in a radial well plan were found to be kick-off depth and the number of radials per kick-off. Azimuth and window did not contribute much to NPV in the examined Fractured case where these parameters were part of the optimization.

There several avenues for further development of the workflow. The CMA-ES optimizations in this thesis have all used the default constants suggested in literature, so there may be room for improvement by adapting these values to be more suited to the case of RJD well design. Especially the r_λ factor determining the amount of reevaluated control vectors for the UH-CMA-ES. The number of reevaluated control vectors is small compared to the population, increasing this may improve the performance of this algorithm with finding robust optima. Increasing population size could also be considered. The population size determined from literature is dependent on the number of elements in the control vector. In every optimization case examined in this thesis, at least one of the radials controls per kick-off (radial length) was not considered in the optimization, effectively increasing the population size compared to what is suggested in literature.

In addition, incorporating geologic uncertainty would be a significant improvement on the current workflow for real world implementation. Though this comes with the added complexity of having both uncertainty on the input parameters and in the objective function, likely requiring a large amount of function evaluations. This increase in computational cost could be limited by evaluating the effect of radial trajectory uncertainty on the objective function. Given that the radials are small compared to the grid blocks in the reservoir simulation, there will likely be situations where a radial does not exit a grid block even when considering the uncertainty in trajectory.

The method of determining how the control vector population is sampled is another area for further investigation, for example by limiting the covariance update to the main diagonal and ignoring the correlation with other parameters as is the case now.

Finally there are some practical changes that could be made to the code. The current workflow has the option to place kick-offs very close to one another. Having a function to combine kick-offs when this occurs would allow the algorithm to eliminate parameters in the control vector.

This could either be used to reduce the dimension of the optimization or to add a kick-off location, using the parameters eliminated from combining two or more kick-offs. The parametrization used by the workflow is only suitable for straight backbones and straight radials. Expanding the parametrization to allow for deviated backbones would be a significant step to making the workflow useable in more real life reservoir cases. Whether the parametrization of radials should also be capable of deviated radials would depend on the effect of the uncertainty on trajectory on the simulation output.

Bibliography

1. *Klaipeda Geothermal Demonstration Project*. **Radeckas B., Lukosevicius**. Tokyo : s.n., 2000.
2. *Horizontal Radial Drilling System*. **Dickinson, W. and Dickinson, R.W.** Bakersfield : Society of Petroleum Engineers, 1985. SPE California Regional Meeting.
3. **Reinsch, T. et al.** *Insights Into the Radial Water Jet Drilling Technology - Application in a Quarry*. 2017.
4. *Theoretical and experimental study of the pulling force of jet bits in radial drilling technology*. **Ruichang G., Gensheng L., Zhongwei H., Shouceng T., Xiaoning Z., Wei W.** 2009, Petroleum Science, Vol. 6, pp. 395-399.
5. *Radial Jet Drilling: A Technical Review*. **Kamel, A.H.** Manama : s.n., 2017. SPE Middle East Oil & Gas Show and Conference.
6. *Radial Drilling Revitalizes Aging Field in Tarim: A Case Study*. **Teng et al.** The Woodlands, Texas : s.n., 2014. SPE/ICoTA Coiled Tubing & Well Intervention Conference & Exhibition.
7. *Production Enhancement in Mature Fields in Assam Arakan Basin by Radial Jet Drilling- A Case Study*. **Maut P.P., Jain D., Mohan R., Talukdar D., Baruah T., Sharma P., Verma S.** Kuala Lumpur : s.n., 2017. SPE Symposium: Production Enhancement and Cost Optimization.
8. *Radial Jet Drilling in Mature Fields of Oil India Limited- An Experimental Approach* . **Jain D., Maut P.P., Saharia P., Dutta R., Yomda S., Hatchell I., Mukherjee A.** Mumbai : s.n., 2017. SPE Oil and Gas India Conference and Exhibition.
9. **Nocedal, Wright.** *Numerical Optimization*. New York : Springer, 2006.
10. *Efficient Ensemble-Based Closed-Loop Production Optimization*. **Chen Y., Oliver D.S., Zhang D.** 4, s.l. : Society of Petroleum Engineers, 2009, Vol. 14.
11. *Theoretical connections between optimization algorithms based on an approximate gradient*. **Do S.T., Reynolds A.C.** s.l. : Springer Science, 2013, Computational Geoscience, Vol. 17, pp. 959-973.
12. *A stochastic simplex gradient (StoSAG) for optimization under uncertainty*. **Fonseca R., Chen B., Jansen J.D., Reynolds A.C.** s.l. : Wiley Online Library, 2016, International Journal for Numerical Methods in Engineering.
13. *Particle Swarm Optimization*. **Kennedy J., Eberhart R.** Perth : IEEE, 1992. International Conference on Neural Networks.
14. **X.S., Yang.** *Nature-Inspired Metaheuristic Algorithms*. Cambridge : Luniver Press, 2010.
15. *Completely Derandomized Self-Adaption in Evolution Strategies*. **Hansen N., Ostermeier A.** 2001, Evolutionary Computation, Vol. 9, pp. 159-195.
16. **Hansen, N.** *A CMA-ES for Mixed-Integer Nonlinear Optimization*. Paris : INRIA, 2011.
17. *A Method for Handling Uncertainty in Evolutionary Optimization with an Application to Feedback Control of Combustion*. **Hansen N., Niederberger A., Guzzella L., Koumoutsakos P.** Dublin : Institute of Electrical and Electronics Engineers, 2011, IEEE Transactions on Evolutionary Computation.
18. *Efficient Well Placement Optimization with Gradient-Based Algorithms and Adjoint Models*. **Sarma P., Chen, W.H.** Amsterdam : s.n., 2008. SPE Intelligent Energy Conference & Exhibition.
19. *Application of particle swarm optimization algorithm for determining optimum well location and type*. **Onwunalu, J.E. and Durlofsky, L.J.** 2009, Computational Geosciences, pp. 183-198.
20. *Optimization of Multilateral Well Design and Location in a Real Field Using a Continuous Genetic Algorithm*. **Bukhamsin, A.Y., Farshi, M.M. and Aziz, K.** Al-Khobar : s.n., 2010. SPE/DGS Annual Symposium and Exhibition.
21. *Optimization of Nonconventional Well Type, Location, and Trajectory*. **Yeten, B., Durlofsky, L.J. and Aziz, K.** San Antonio : s.n., 2003. SPE Annual Technical Conference and Exhibition.
22. *Well placement optimization with covariance matrix adaption evolution strategy and meta-models*. **Bouzarkouna Z., Ding D.Y., Auger A.** 16, 2012, Computational Geoscience, pp. 75-92.

23. *Model-Based Well Location Optimization - A Robust Approach*. **Ramirez B.A., Joosten G.J.P., Kaleta M.P., Gelderblom P.P.** Montgomery, TX : s.n., 2017. SPE Reservoir Simulation Conference.
24. *Uncertainty Assessment of Well-Placement Optimization*. **Güygüler B., Horne R.N.** 2004, SPE Reservoir Evaluation & Engineering, pp. 24-32.
25. *Multiple-Objective Optimization Applied to Well Path Design under Geological Uncertainty*. **Schulze-Riegert R., Bagheri M., Krosche M., Kück N., Ma D.** The Woodlands, TX : s.n., 2011. SPE Reservoir Simulation Symposium.
26. *Well Trajectory Optimization Constrained to Structural Uncertainties*. **Hanea, R.G., et al.** Montgomery : s.n., 2017. SPE Reservoir Simulation Conference.
27. *Interpretation of Well-Block Pressures in Numerical Simulation*. **Peaceman, D.W.** 1978, Society of Petroleum Engineers Journal, pp. 183-194.
28. *Interpretation of Well-Block Pressures in Numerical Reservoir Simulation with Nonsquare Grid Blocks and Anisotropic Permeability*. **Peaceman, D.W.** 3, s.l. : Society of Petroleum Engineers, 1983, Vol. 23.
29. *Representation of a Horizontal Well in Numerical Reservoir Simulation*. **Peaceman, D.W.** 1, s.l. : Society of Petroleum Engineers, 1993, Vol. 1.
30. **Shu, J.** *Comparison of Various Techniques for Computing Well Index*. 2005.
31. *Well Inflow Modelling for Wells not Aligned to a Numerical Grid*. **Egberts, P.J.P., Shatyrbayeva, S.G.S. and Fokker, P.A.** s.l. : Society of Petroleum Engineers, 2013.
32. *A case study of radial jetting technology for enhancing geothermal energy systems at Klaipėda geothermal demonstration plant*. **Nair, R. et al.** Stanford : s.n., 2017. 42nd Workshop on Geothermal Reservoir Engineering.
33. *Neuro-Fuzzy DC Motor Speed Control Using Particle Swarm Optimization*. **Allaoua B., Laoufi A., Gasbaoui B., Abderrahmani A.** 15, 2009, Leonardo Electronic Journal of Practices and Technologies, pp. 1-18.
34. *A case study of radial jetting technology for enhancing geothermal energy systems at Klaipėda geothermal demonstration plant*. **Nair R., Peters E., Sliupa S., Valickas R., Petrauskas S.** Stanford : s.n., 2017.