Project report

TPG4560 Specialization Project

Student Eirin Lillevik

Student Ingvild Evensen Standal

Optimized Wellbore Trajectories

Trondheim, December 19, 2017

NTNU
Norwegian University of Science and Technology
Faculty of Engineering
Department of Geoscience and Petroleum

Summary

It is known from the media that Statoil is trying a new generation of platforms on the Oseberg field. According to them, this will cut costs dramatically. The reason is that dry well components are used. These are much cheaper than wet ones. They call the solution "subsea on a stick". Gathering all wellheads on one platform means that all wells have to be drilled from the same position. This increases the average well path length dramatically. A large part of the subsea field development costs are drilling related costs. How does this solution affect drilling costs?

This report is a starting point of a tool that can compare different subsea field layouts based on the overall field costs, including the drilling costs. The purpose is to identify the layout that yields the shortest average well path length. This is done by studying four different layouts with 12 fixed completion intervals. One layout consists of only satellite wells, while the others have one, two and three common drill centers for all the wells. The optimal position of the drill center(s) is identified for each field layout. The methods are based on achieving the shortest possible well path, and the drill center locations are calculated with the same criteria. The wells are constructed using trigonometric relations. Finally, the average well path lengths are calculated and all layouts are compared to each other.

The comparison of the different layouts show that the average well path length is highly sensitive to the number of drill centers. The case of satellite wells yields the shortest average well path length. As the number of drill centers decrease, the average well path length increase. A field layout with one common drill center, for example "subsea on a stick", is the solution that yields highest drilling costs because of longest average well path length. Further work, where subsea hardware and installation costs are taken into account, remains to be done to find the optimal field layout.

Preface

This study is the result of our Specialization Project at the Norwegian University of Science and Technology (NTNU). The project is written at the Department of Geoscience and Petroleum during the autumn semester of 2017. The problem description and issues discussed were described and given by Professor Tor Berge Gjersvik.

Trondheim, 2017-12-19

Eirin Lillevik Ingvild Evensen Standal

Acknowledgment

We would like to thank our supervisor Professor Tor Berge Gjersvik for providing us with this topic. His guidance and suggestions along the way are highly appreciated.

We would also like to thank our co-supervisors Professor Sigbjørn Sangesland and Audun Faanes for taking their time to discuss the subject.

E.L.

I.E.S.

Table of Contents

List of Figures

List of Tables

Abbreviations

BUA Build-up angle

BUR Build-up rate

DC Drill center

KOP Kickoff point

ROC Radius of curvature

ROT Radius of turn

TR Turn rate

WPL Well path length

XT Christmas tree

Chapter 1

Introduction

1.1 Background

Subsea field development costs consist largely of subsea well construction costs, subsea facilities hardware, and subsea installation and commissioning. In addition, there are the costs of tying the subsea field back to a production facility such as an existing platform, FPSO or "subsea to beach".

Industry practice for geologists, petrophysicists, reservoir engineers, and production engineers is jointly to determine the subsurface completion intervals for each well. Thereafter, these intervals are the starting point for the well construction engineers to model and propose wellbore trajectories and the required well program. In this work, the drilling engineers fit the placement of the wellheads into templates and/or satellites/cluster layouts in accordance with the subsea field development plan.

In Norway, it has become a standard solution to bring typically four wells into one template; a template with a roof offering trawl protection. This inevitably increases the average wellbore lengths, and well construction costs increase accordingly. There is a need to be able to compare different possible subsea layouts of wellhead positions (satellites/cluster or template) and the associated drilling, hardware, and installation costs to identify the best solution minimizing:

1. Drilling costs and drilling related technical risks (shorter and "simpler" wells).

2. The overall field development cost.

A more distributed solution may give shorter wells, but also more complex piping and more subsea structures in addition to the cost related to the re-placement of the drilling rig.

Literature Survey

A comprehensive literature search revealed that there is no relevant literature regarding this topic.

1.2 Objectives

The aim of this project work is to initiate the way towards an automatic tool that finds the optimal subsea field layout, with respect to cost of subsea facilities, flowlines and drilling/completion. As mentioned in chapter 1.1, there are many factors that have to be considered. This Project Report initiates the work of an automatic tool by studying the wellbore trajectories. Thus, the purpose of this Project Report is to compare the average well path lengths of 12 wells in different field layouts. The results will later be significant in the decision of choosing the optimal field development. The wells are constructed in a manner that minimizes the well paths, since longer wells increases the costs.

To find the average well path lengths, the objectives are:

1. Find the drill center(s) of each field architecture that will yield the shortest distance between the drill center and the completion(s).

2. Construct the mathematically shortest well path from the drill center to the completion.

3. Calculate the average well path lengths of the wells connected to the different field architectures.

All calculations are implemented in MATLAB. The tool requires data from the user, such as completion interval coordinates, build-up rate (BUR) and turn rate (TR).

1.3 Limitations

The limitations of the approach are mainly related to horizontal completion intervals. The mathematics applied in the calculations are derived from the fact that the completion intervals provided are horizontal. If the completion intervals are changed from horizontal, the program will require adjustment and new calculations, because of changing trigonometric relations.

Another important limitation is related to combinatorics and the number of completion intervals. The program is set up to handle 12 completion intervals and perform calculations based on four cases: all satellite wells, one drill center, two drill centers and three drill centers. If the number of completion intervals is changed or if the case of n ($n = \{n \in \mathbb{R}, n > 3\}$) drill centers is of interest, the program will fail. In other words, non-generalized formulas is a critical limitation.

In addition, the formulas for calculation of drill center placements require that the completion intervals are located at the same depth. The placement depends on the XY coordinates of the completion intervals. If the completion intervals are located at different depths, this will have impact on the drill center location in order to minimize the average well path lengths.

1.4 Approach

The approach is to first find the position of the drill centers. In the case of satellite wells this is done by constructing the well path. In the other cases, the placements are based on the arithmetic mean calculation of the completion coordinates. Further, the wells are constructed "bottoms up" by use of trigonometric relations. Finally, the well path lengths and their average are calculated. This is done to compare the different field layouts with respect to well path length.

The first chapter includes an introduction with problem definition and approach. The second chapter describes the methods used to obtain the objectives. The results and discussion are presented in chapter 3 and 4, respectively.

Chapter 2

Methods

2.1 Placement of Drill Centers

This section explains the methods used to find the optimal placement of the drill center(s) in each field layout. A drill center is the location where the drilling operations commence. The placement of the drill center(s) is critical to construct the shortest possible well paths.

Four different field layouts, described in chapter 1.3, are studied. The calculations differ from the case of satellite wells to the cases of common drill centers. In the case of satellite wells, the drill centers are found from the wellbore trajectory calculations. In the other cases, the drill centers are found by minimizing the average distance from the drill center to the start of the completion intervals in the XY-plane. The user enters the completion coordinates into the tool.

Satellite Wells

In the case of satellite wells, each completion interval has its own associated drill center. Each well path is constructed to have one build section and is oriented along the same azimuth angle as the completion interval. The placement of the drill center is consequently determined by the mathematically shortest well path. Figure 2.1 shows an example of 12 completion intervals (c) and the associated drill centers (DC) in the XY-plane.

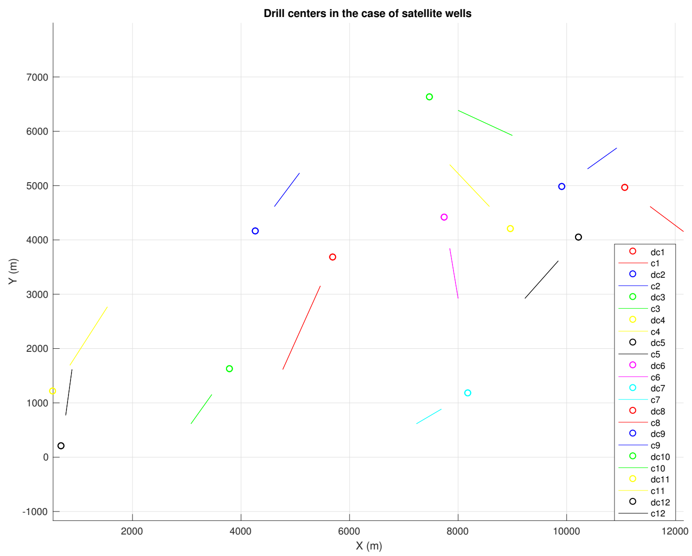


Figure 2.1: Completion intervals and associated drill centers.

One Drill Center

In the case of one drill center, every completion interval is reached from a common drill center. The location of the drill center is determined by the arithmetic mean of the completion start coordinates in the XY-plane, X_{cs} and Y_{cs}. Equations 2.1 and 2.2 show how the drill center location is calculated for n completion intervals. Figure 2.2 shows an example of one drill center and 12 completions in the XY-plane.

$$X_{DC} = \frac{1}{n}\sum_{i=1}^{n} X_{cs} \tag{2.1}$$

$$Y_{DC} = \frac{1}{n}\sum_{i=1}^{n} Y_{cs} \tag{2.2}$$

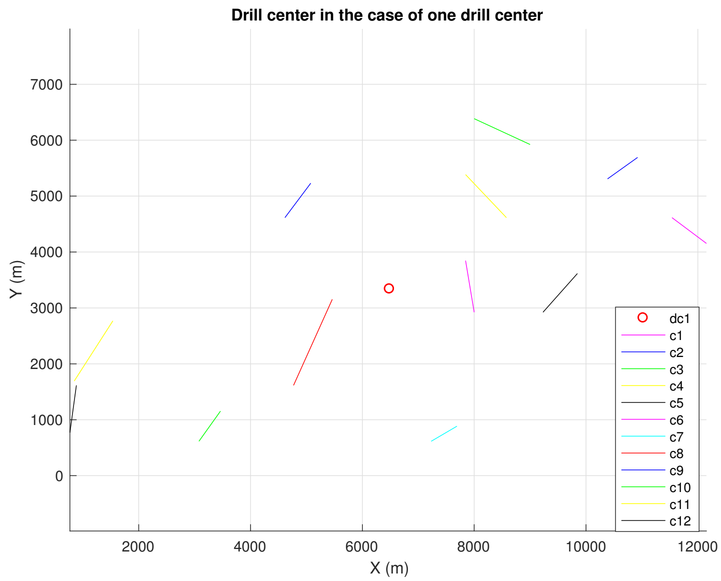


Figure 2.2: Completion intervals and one drill center.

Two Drill Centers

In the case of two drill centers, six and six completion intervals are reached from a common drill center. To find the optimal combination of six and six completion intervals, every possible combination is identified. The drill center positions for each combination are then found. The distances from the drill centers to the associated completion interval starts are then calculated in the XY-plane. The optimal combination is the one with the shortest average distance.

The binomial coefficient identifies the amount of unique combinations when k completions are chosen out of n completions:

$$\text{binomial coefficient} = \binom{n}{k} = \frac{n!}{k!(n-k)!} \tag{2.3}$$

Since there are six completion intervals per drill center, and there are twelve completions to choose from, the number of unique combinations are:

$$\binom{12}{6} = \frac{12!}{6!(12-6)!} = 924$$

To obtain these combinations, a matrix R is created with MATLAB's built-in function, *nchoosek*.

The command R=*nchoosek*(n,k) generates a matrix that contains every combination of n objects taken k at a time. A vector M is created, containing the numbers from 1 to 12. These numbers represent every completion interval. M consists of the n objects *nchoosek(n,k)* can choose from, and k is the six completions that are taken at a time. R is generated and results in a 924×6 matrix. Each row contains six numbers from 1 to 12, and every row contains a unique combination.

R represents the possible combinations of completions for the first drill center. The next step is to find the remaining completions for the second drill center. This is done by focusing on one row in R at a time. The combinations of the remaining completions are represented in a new matrix called R_rest. R_rest is created with the methodology explained in figure 2.3.

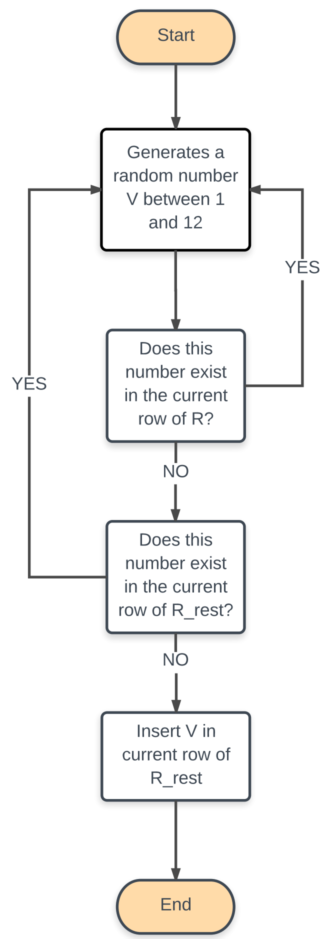


Figure 2.3: Flow chart explaining how the combinations in R_rest are generated.

A random number is assigned to the parameter V by use of the built-in MATLAB function *randi*. With the command V=*randi*(12), the number assigned will be between 1 and 12. Starting with the first row in R, if V is not equal to any of the numbers in this row, then it is inserted into the first row in R_rest. If V is equal to any of the numbers in this row, a new random number is generated. The first number in the first row of R_rest is now inserted. A new number is then assigned to V. The same procedure repeats itself, but V is also compared to the number in R_rest. When the first row of R_rest is filled out, the same procedure is performed for the remaining rows in R.

When R and R_rest are generated, both matrices have the same size, 924×6. Each row in R represent a unique grouping for the first drill center and each row in R_rest the unique corresponding grouping for the second drill center. The structures of matrix R and R_rest are illustrated below.

$$R = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 2 & 3 & 4 & 5 & 6 & 7 \\ 3 & 4 & 5 & 6 & 7 & 8 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix} \qquad R\_rest = \begin{bmatrix} 12 & 9 & 11 & 10 & 8 & 7 \\ 8 & 9 & 12 & 1 & 10 & 11 \\ 2 & 11 & 12 & 10 & 1 & 9 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix}$$

The next step is to calculate two optimum drill center locations for each combination. This is done by use of equations 2.1 and 2.2. When the drill centers to every combination are found, the distance from each completion interval start to the associated drill center is calculated using equation 2.4.

$$d = \sqrt{(X_{cs} - X_{DC})^2 + (Y_{cs} - Y_{DC})^2} \tag{2.4}$$

The average distance is computed for each combination. Finally, the optimal combination is the one with the shortest average distance. Figure 2.4 shows an example of two drill centers and 12 completions in the XY-plane.

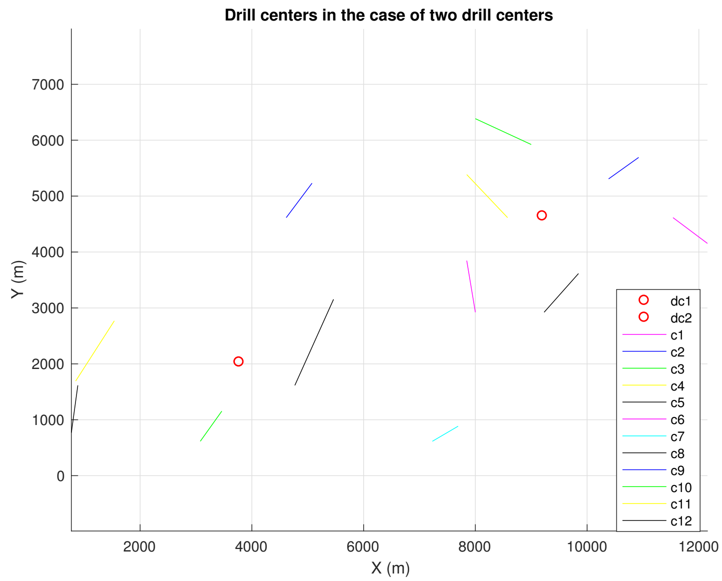


Figure 2.4: Completion intervals and two drill centers.

Three Drill Centers

In the case of three drill centers, four and four completion intervals are reached from the same drill center. The procedure used for the case of two drill centers is followed, but there are now three matrices instead of two.

Equation 2.3 is used to find the number of unique combinations. There are now four completion intervals per drill center, and twelve completions to choose from. The number of unique combinations in the first drill center is:

$$\binom{12}{4} = \frac{12!}{4!(12-4)!} = 495$$

R is created in the same way as for the case of two drill centers, except the value of k is now 4. This results in a 495×4 matrix. R represents the possible combinations of completions for the first drill center.

The combinations of the remaining completions are represented in a new matrix called R_mar. This matrix is created with the same methodology used for R_rest in the case of two drill centers.

The difference is that there are now eight remaining combinations, thus the dimension of R_mar is 495×8. The structures of R and R_mar are illustrated below.

$$R = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 2 & 3 & 4 & 5 \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix} \qquad \text{R\_mar} = \begin{bmatrix} 11 & 6 & 5 & 10 & 8 & 12 & 9 & 7 \\ 12 & 11 & 1 & 7 & 9 & 8 & 6 & 10 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix}$$

For the second drill center, there are eight remaining completions to choose from in each row of R_mar. Since there are four completions per drill center, the number of unique combinations for the second drill center are:

$$\binom{8}{4} = \frac{8!}{4!(8-4)!} = 70$$

Thus, for each combination in R, there are 70 different combinations for the second drill center. To list all of them, a new matrix called Mar_M is generated. The first four columns in Mar_M are generated by *nchoosek*, where k is equal to 4. Starting with the first row in R_mar, n is a vector that contains the numbers in the first row of R_mar. For the following four columns in Mar_M, n contains the numbers in the second row of R_mar. This is done for all 495 rows of R_mar, thus the number of columns in Mar_M becomes $495 \times 4 = 1980$. Since 70 combinations are created for each row in R_mar, the dimension of Mar_M is 70×1980. The generation of Mar_M is illustrated below.

$$\text{R\_mar} = \begin{bmatrix} 11 & 6 & 5 & 10 & 8 & 12 & 9 & 7 \\ 12 & 11 & 1 & 7 & 9 & 8 & 6 & 10 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix} \qquad \text{Mar\_M} = \begin{bmatrix} 6 & 10 & 9 & 7 & 11 & 1 & 9 & 8 & \dots \\ 6 & 5 & 12 & 8 & 11 & 7 & 9 & 10 & \dots \\ 11 & 12 & 6 & 5 & 10 & 7 & 9 & 12 & \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}$$

The next step is to identify the remaining completions for the third drill center. Since there are four completions per drill center, and only four completions left to choose from, there is only one combination left. A third matrix called M_gr3 is created. M_gr3 has the same size as Mar_M since each combination in Mar_M has one combination of the remaining completions. Figure 2.5 explains the methodology used to create M_gr3.

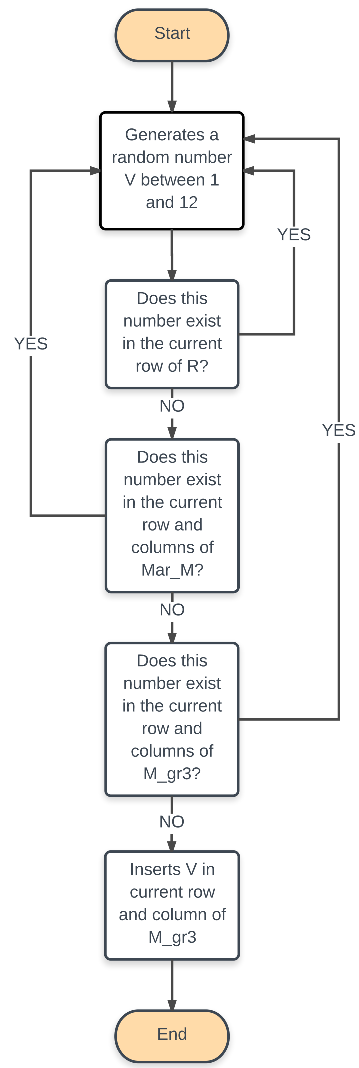


Figure 2.5: Flow chart explaining how the combinations in M_gr3 are generated.

As for the case of two drill centers, the parameter V is assigned a random number. Starting with the first row in R, V is compared to the numbers in this row. V is assigned a new value if it is equal to any of these numbers, if not, V remains the same. Then V is compared to the first four columns in the first row of Mar_M. V is assigned a new value if it is equal to any of these numbers, if not, V is inserted into the first row and first column of M_gr3. A new number is then assigned to V. The same procedure repeats itself, but V is also compared to the number in M_gr3. When the first four columns in the first row of M_gr3 are filled out, the same procedure is performed for the first row in R, and the first four columns in the second row of Mar_M. The procedure is followed for the remaining rows in Mar_M, before V is compared to the numbers in the second row of R, and the next four columns in Mar_M. The structure of matrix R, Mar_M and M_gr3 are illustrated below.

$$R = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 2 & 3 & 4 & 5 \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix}$$

$$\text{Mar\_M} = \begin{bmatrix} 6 & 10 & 9 & 7 & 11 & 1 & 9 & 8 & \cdots \\ 6 & 5 & 12 & 8 & 11 & 7 & 9 & 10 & \cdots \\ 11 & 12 & 6 & 5 & 10 & 7 & 9 & 12 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}$$

$$\text{M\_gr3} = \begin{bmatrix} 12 & 8 & 11 & 5 & 6 & 12 & 10 & 7 & \cdots \\ 7 & 10 & 9 & 11 & 6 & 1 & 12 & 8 & \cdots \\ 8 & 7 & 9 & 10 & 11 & 6 & 8 & 1 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}$$

All possible combinations are now found. There are 495 ways to group the completions for the first drill center. Each of these have 70 ways to group the remaining completions for the second drill center. Each of these have again only one way to arrange the remaining completions for the third drill center. Thus, the total amount of unique combinations are:

$$495 \times 70 \times 1 = 34650$$

The optimal drill center locations to each combination are found by use of equation 2.1 and 2.2. The distance from each completion start to the associated drill center is calculated using equation 2.4. The average distance is computed for each combination. Finally, the optimal combination is the one with the lowest average distance. Figure 2.6 shows an example of three drill centers and 12 completions in the XY-plane.

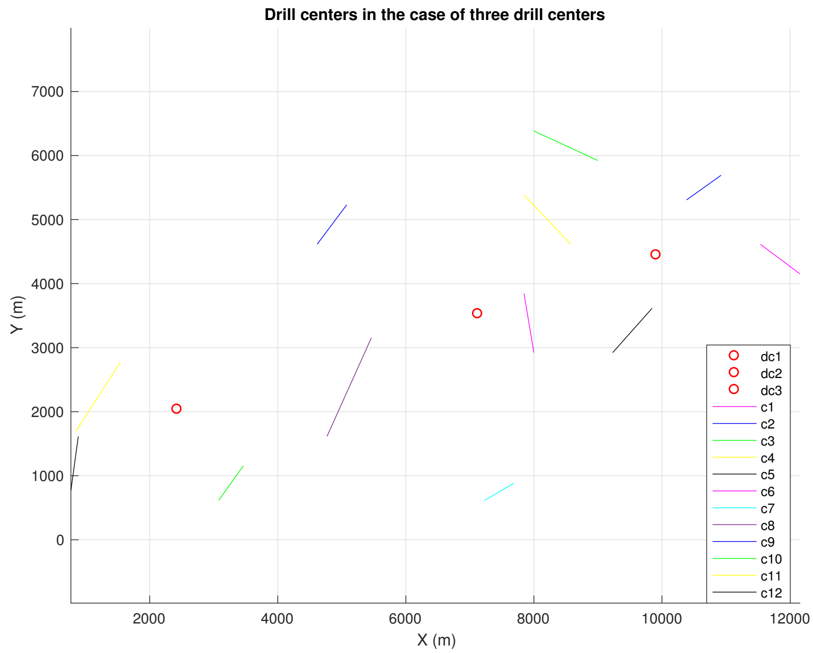


Figure 2.6: Completion intervals and three drill centers.

2.2 Wellbore Trajectory Calculations

This section explains the methods used to calculate the mathematically shortest well paths. Because of changes in the azimuth angle along the wellbore, the calculations differ from the case of satellite wells to the cases of common drill centers.

The well paths are calculated in a two-dimensional RZ-plane. The dimensions that are used are the measured horizontal displacement, R, and the true vertical depth of the well, Z.

The calculations are primarily for horizontal completion intervals. Some equations are adjusted for non-horizontal completion intervals. This is to make the transition easier if non-horizontal completion intervals will be of interest at a later occasion. Consequently, some of the equation requirements are unused.

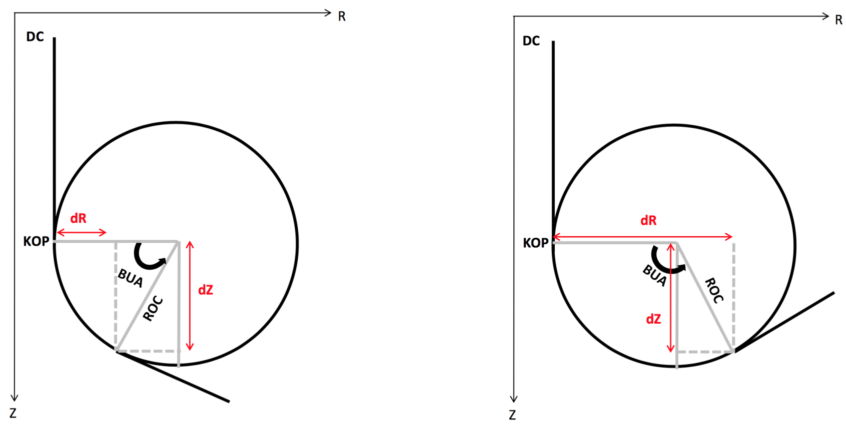
Common for each case is that the build-up rate (BUR), coordinates of completion start and coordinates of completion end are input parameters. This is data that the user of the model decides before the tool is ran. The BUR is a measure of how many degrees inclination the well builds per 30 meter drilled.

Satellite Wells

The case of satellite wells is special. As mentioned in chapter 2.1, each completion interval has its own associated drill center and the placement of these drill centers are determined by the mathematically shortest well path.

The wells are constructed from bottom to top, as the only known parameters are the target coordinates and the build-up rate. The shortest possible well path is constructed by building inclination toward the surface instantly. This approach causes the satellite well paths to have one build section. The build section connects the completion start to the kickoff point (KOP). The KOP is the location in a vertical wellbore where the directional drilling operation commence.

To make calculations in the build section, the well path is projected onto the RZ-plane, and the build section is expressed as an arc. The radius of the circle is a function of the BUR. The radius is called the radius of curvature (ROC). As the BUR increases, the ROC decreases. The ROC is illustrated in figure 2.7. Trigonometric relations are identified to calculate displacements in R and Z directions caused by the build section.



(a) BUA is less than 90° (b) BUA is more than or equal to 90°

Figure 2.7: Satellite well path with one build section expressed as an arc.

The build-up angle (BUA) is the total inclination required to hit the target with the correct angle. It is a function of the inclination of the completion interval. The inclination of the completion interval is calculated in degrees, and is defined as the angle from the vertical (0°) to the completion interval in a counterclockwise direction. When the height (dZ_c) and length (L_c) of the completion intervals are calculated by equations 2.5 and 2.6, the build-up angle is calculated from trigonometric identities in equation 2.7. The height is simply the difference in depth of the completion start and end. The length is computed using the Pythagorean theorem.

$$dZ_c = Z_{ce} - Z_{cs} \tag{2.5}$$

$$L_c = \sqrt{(X_{ce} - X_{cs})^2 + (Y_{ce} - Y_{cs})^2 + (Z_{ce} - Z_{cs})^2} \tag{2.6}$$

$$\text{BUA} = \begin{cases} \cos\frac{dZ_c}{L_c}, & \text{if } dZ_c \geq 0 \\ 90° - \sin\frac{dZ_c}{L_c}, & \text{if } dZ_c < 0 \end{cases} \tag{2.7}$$

Completion interval, completion start and completion end are denoted c, cs and ce, respectively. The ROC and the displacements in R (dR) and Z (dZ) caused by the build section, are also found by trigonometric relations in figure 2.7. The equations are listed as 2.8, 2.9 and 2.10 below. The last unknown regarding the build section, the arc length, is computed using equation 2.11.

$$\text{ROC} = \left(\frac{360° \times \text{arc length}}{2\pi \times \text{BUA}}\right) \tag{2.8}$$

$$dR = \begin{cases} \text{ROC} - \text{ROC}\cos(\text{BUA}), & \text{if BUA} \geq 90° \\ \text{ROC} + \text{ROC}\sin(\text{BUA} - 90°), & \text{if BUA} < 90° \end{cases} \tag{2.9}$$

$$dZ = \begin{cases} \text{ROC}\cos(\text{BUA} - 90°), & \text{if BUA} \geq 90° \\ \text{ROC}\sin(\text{BUA}), & \text{if BUA} < 90° \end{cases} \tag{2.10}$$

$$\text{arc length} = \left(\frac{\text{BUA}}{\text{BUR}}\right) \tag{2.11}$$

The next step is to find the coordinates of the KOP. To find the displacements between the completion start and the KOP in R and Z direction, the azimuth angle (azi_c) of each completion interval has to be computed. The azimuth angle is the angular direction in degrees with reference to the North. The satellite wells are constructed with constant azimuth angles. The azimuth of the completion intervals is calculated using the four-quadrant inverse tangent formula, equation 2.12.

$$azi_c = \text{atan2}(X_{ce} - X_{cs}, Y_{ce} - Y_{cs}) \tag{2.12}$$

The proof of equation 2.12 is found in Appendix A.1. The location of the KOP is then calculated using trigonometric identities in equations 2.13, 2.14 and 2.15.

$$\text{KOP}_X = X_{cs} - dR\sin(azi) \tag{2.13}$$

$$\text{KOP}_Y = Y_{cs} - dR\cos(azi) \tag{2.14}$$

$$\text{KOP}_Z = Z_{cs} - dZ \tag{2.15}$$

The well path length (WPL) of each satellite well is calculated by adding the length of the completion interval, the arc length and the depth of the KOP, using equation 2.16.

$$\text{WPL} = L_c + \text{arc length} + \text{KOP}_Z \tag{2.16}$$

The average well path length (WPL_{avg}) is calculated to compare the results from the different subsea layouts, using equation 2.17, where n is the number of wells.

$$\text{WPL}_{\text{avg}} = \sum_{i=1}^{n} \frac{\text{WPL}_i}{n} \tag{2.17}$$

Figure 2.8 illustrates the well paths in a subsea satellite field layout.

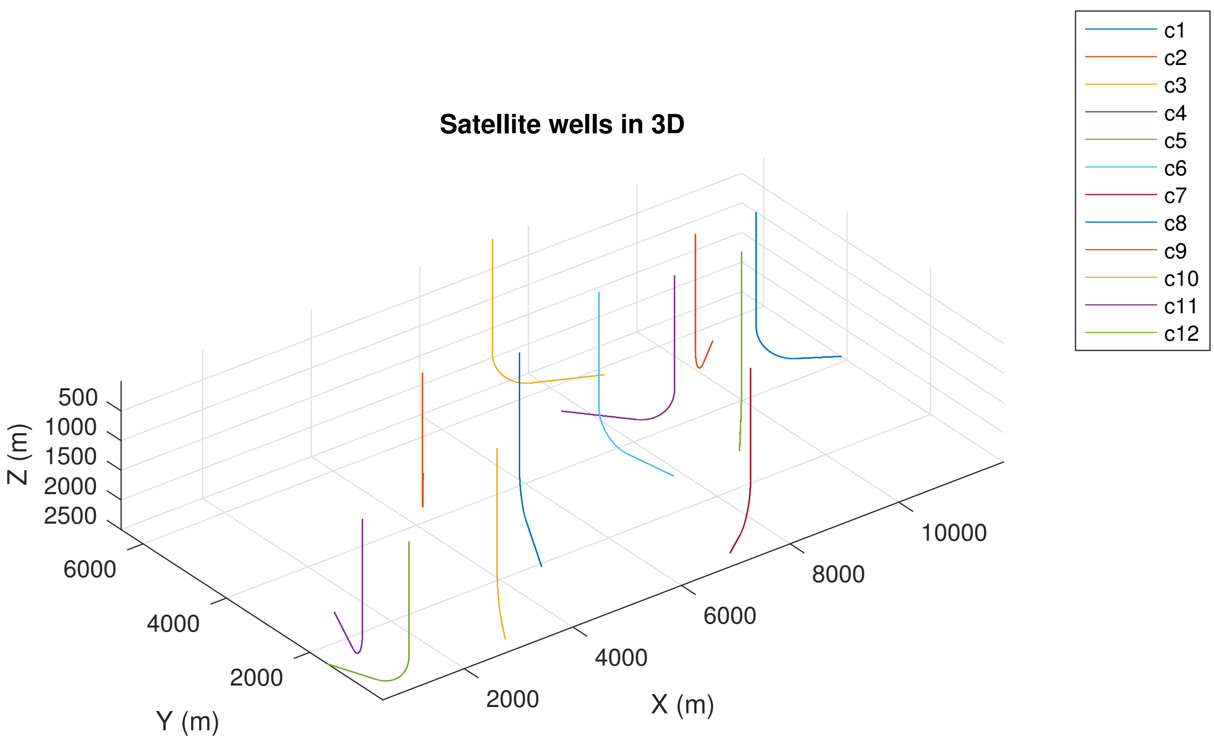


Figure 2.8: Satellite well paths.

One Drill Center

The case of one drill center is more complex because the azimuth angle changes along the well path. To calculate the length of the well projected onto the XY-plane, new equations and new trigonometric identities are introduced.

As with the satellite wells, these wells are constructed from bottom to top because of limited known parameters. The difference in this approach is that the wells need to curve around a cylinder in the three-dimensional XYZ-plane. When the wells are projected onto the XY-plane, the wells turn around a circle, illustrated in figure 2.9.

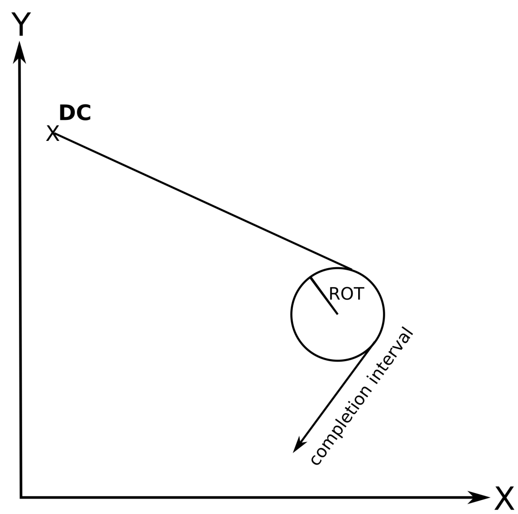


Figure 2.9: The projection of a well onto the XY-plane.

The circle has a radius that is a function of the turn rate (TR). The TR is an input parameter determind by the user. It is a measure of how many degrees the well turns in the XY-plane per 30 meter drilled. The radius of the associated circle is called the radius of turn (ROT), and is calculated by equation 2.18.

$$\mathrm{ROT} = \frac{360°}{2\pi \times \mathrm{TR}} \tag{2.18}$$

Although the TR is an input parameter, it will be adjusted if the distance between the drill center and the completion start in the XY-plane is too short for the well to follow the arc. This is done by use of the critiera in equation 2.19. Whenever this criteria is fulfilled, the turn rate is increased by $1°/30m$, and the criteria is checked again until the criteria not is fulfilled.

$$\sqrt{(X_{DC} - X_{cs})^2 + (Y_{DC} - Y_{cs})^2} < 2 \times \text{ROT} \tag{2.19}$$

To find the arc length in the XY-plane, the point on the circle where the well will start to turn must be located. In order to find this point, the coordinate system is transformed. Each completion interval is given its own transformed coordinate system.

The process of transforming the coordinate system requires two steps. In the first step all coordinates are shifted into a coordinate system (a, b) that has its origin in the drill center, by use of equations 2.20 and 2.21.

$$a = X - X_{DC} \tag{2.20}$$

$$b = Y - Y_{DC} \tag{2.21}$$

The axes are still oriented in the same directions as the axes in the XY-plane. All completion intervals share this system.

The second step is to rotate the (a, b) coordinate system to transform the coordinates into the final coordinate system (x_i, y_i). This is where every completion interval gets its own distinctive system. The characteristics of these coordinate systems are that they have the origin in the drill center, the y_i-axis is parallel to the completion interval and y_i increases from the completion end to the completion start. This orientation is chosen to simplify the upcoming calculations.

The rotation angle (rotate) varies depending on the azimuth of the completion interval. To make the y_i-axis parallel to the completion interval and its positive direction to go from the completion end to the completion start, the rotation angle is computed by use of equation 2.22.

$$\text{rotate} = 180° - azi_c \tag{2.22}$$

Equation 2.22 can be verified by sketching the situations. The azimuth of the completion intervals are calculated by equation 2.12. The next step is to find the formulas that can transform the shifted coordinates into the rotated coordinate system. A sketch of the situation is illustrated in figure 2.10.

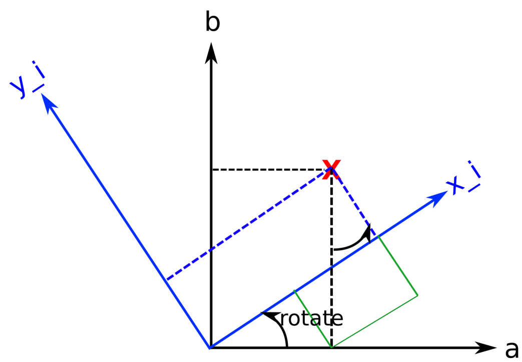


Figure 2.10: Transformation of coordinate system.

The green lines in figure 2.10 are help lines that make it easier to identify the trigonometric relations. Two triangles form, and both have one angle equal to the rotation angle. In addition, one has the *a* coordinate as hypotenuse and the other has the *b* coordinate as hypotenuse. Equation 2.23 and 2.24 is set up by the trigonometric relations identified.

$$x_i = \mathrm{a} \times \cos(\mathrm{rotate}) + \mathrm{b} \times \sin(\mathrm{rotate}) \tag{2.23}$$

$$y_i = -\mathrm{a} \times \sin(\mathrm{rotate}) + \mathrm{b} \times \cos(\mathrm{rotate}) \tag{2.24}$$

The next step is to find the center of the turn circle, (x_{cci}, y_{cci}). The circle is intentionally oriented in a manner that makes the completion interval a tangent to the circle. Their intersection point, is the point of the completion start (x_{i1}, y_{i1}). Thus, the circle center has the same y_i value as the completion start, see figure 2.11.

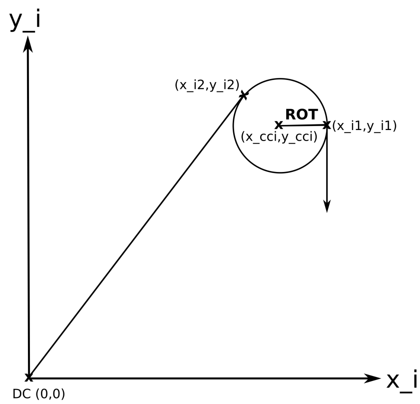


Figure 2.11: Sketch of one completion interval's transformed coordinate system.

The x_i coordinate of the circle center depends on the TR and how the completion start is oriented relative to the drill center. Because the x_i coordinate of the point of turn always is closer to the drill center than the completion start x_i coordinate, the coordinates of the circle center is calculated using equation 2.25 and 2.26.

$$x_{cci} = \begin{cases} x_{i1} + \mathrm{ROT}, & \text{for } x_{i1} < 0 \\ x_{i1} - \mathrm{ROT}, & \text{for } x_{i1} > 0 \end{cases} \tag{2.25}$$

$$y_{cci} = y_{i1} \tag{2.26}$$

To find the point of turn (x_{i2}, y_{i2}), support lines are added in figure 2.11, illustrated in figure 2.12.

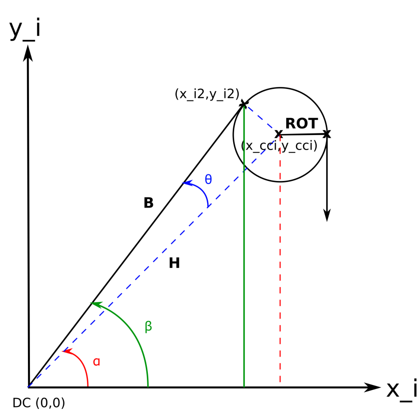


Figure 2.12: All necessary support lines.

It turns out that several solutions are required to find the point of turn. The correct solution depends on which quadrant the circle center is placed in and whether the well turns clockwise or counterclockwise. The well turns clockwise when $x_{cci} < x_{i1}$ and counterclockwise when $x_{cci} > x_{i1}$. Some equations are still common. The parameters H, B, θ and β, illustrated in figure 2.12, are calculated using equation 2.27, 2.28, 2.29 and 2.30 for every completion interval.

$$H = \sqrt{(x_{cci})^2 + (y_{cci})^2} \tag{2.27}$$

$$B = \sqrt{H^2 - \mathrm{ROT}^2} \tag{2.28}$$

$$\theta = \arcsin\left(\frac{\mathrm{ROT}}{H}\right) \tag{2.29}$$

$$\beta = \theta + \alpha \tag{2.30}$$

The parameters that change for each completion interval are α, x_{i2} and y_{i2}. Depending on which criteria that is fulfilled, there are eight unique solution sets. The solutions are listed in equation 2.31, 2.32, 2.33, 2.34, 2.35, 2.36, 2.37 and 2.38.

$$\text{if } x_{cci} > 0,\ y_{cci} > 0, \text{and } x_{cci} > x_{i1}, \text{ then } \begin{cases} \alpha & = \arctan\frac{x_{cci}}{y_{cci}} \\ x_{i2} & = B\sin(\beta) \\ y_{i2} & = B\cos(\beta) \end{cases} \tag{2.31}$$

$$\text{if } x_{cci} > 0,\ y_{cci} > 0, \text{and } x_{cci} < x_{i1}, \text{ then } \begin{cases} \alpha & = \arctan\frac{y_{cci}}{x_{cci}} \\ x_{i2} & = B\cos(\beta) \\ y_{i2} & = B\sin(\beta) \end{cases} \tag{2.32}$$

$$\text{if } x_{cci} > 0,\ y_{cci} < 0, \text{and } x_{cci} > x_{i1}, \text{ then } \begin{cases} \alpha & = -\arctan\frac{y_{cci}}{x_{cci}} \\ x_{i2} & = B\cos(\beta) \\ y_{i2} & = -B\sin(\beta) \end{cases} \tag{2.33}$$

$$\text{if } x_{cci} > 0,\ y_{cci} < 0, \text{and } x_{cci} < x_{i1}, \text{ then } \begin{cases} \alpha & = -\arctan\frac{x_{cci}}{y_{cci}} \\ x_{i2} & = B\sin(\beta) \\ y_{i2} & = -B\cos(\beta) \end{cases} \tag{2.34}$$

$$\text{if } x_{cci} < 0,\ y_{cci} < 0, \text{and } x_{cci} > x_{i1},\ \text{then} \begin{cases} \alpha & = \arctan \frac{x_{cci}}{y_{cci}} \\ x_{i2} & = -B\sin(\beta) \\ y_{i2} & = -B\cos(\beta) \end{cases} \tag{2.35}$$

$$\text{if } x_{cci} < 0,\ y_{cci} < 0, \text{and } x_{cci} < x_{i1},\ \text{then} \begin{cases} \alpha & = \arctan \frac{y_{cci}}{x_{cci}} \\ x_{i2} & = -B\cos(\beta) \\ y_{i2} & = -B\sin(\beta) \end{cases} \tag{2.36}$$

$$\text{if } x_{cci} < 0,\ y_{cci} > 0, \text{and } x_{cci} > x_{i1},\ \text{then} \begin{cases} \alpha & = -\arctan \frac{y_{cci}}{x_{cci}} \\ x_{i2} & = -B\cos(\beta) \\ y_{i2} & = B\sin(\beta) \end{cases} \tag{2.37}$$

$$\text{if } x_{cci} < 0,\ y_{cci} > 0, \text{and } x_{cci} < x_{i1},\ \text{then} \begin{cases} \alpha & = -\arctan \frac{x_{cci}}{y_{cci}} \\ x_{i2} & = -B\sin(\beta) \\ y_{i2} & = B\cos(\beta) \end{cases} \tag{2.38}$$

When the turn points are identified they must be transformed back to the normal coordinate system. This is also a two step process. First, the coordinate system (x_i, y_i) is rotated back to (a, b) by use of equations 2.39 and 2.40.

$$a = x_i \cos(\text{rotate}) - y_i \sin(\text{rotate}) \tag{2.39}$$

$$b = x_i \sin(\text{rotate}) + y_i \cos(\text{rotate}) \tag{2.40}$$

These equations can be derived by following the same procedure as when equations 2.23 and 2.24 were derived. The last step is to shift the coordinate system (a, b) back to the normal coordinate system (X, Y), by use of equations 2.41 and 2.42:

$$X = a + X_{DC} \tag{2.41}$$

$$Y = b + Y_{DC} \tag{2.42}$$

The objective of calculating the turn point is to find the length of the well in the R-plane. To find the length of the turn section, the azimuth of the tangent (azi_t) from the drill center to the turn point (X_2, Y_2) must be calculated. It is calculated by use of equation 2.43.

$$azi_t = \text{atan2}(X_2 - X_{DC}, Y_2 - Y_{DC}) \tag{2.43}$$

The length of the arc, arc_{azi}, is found by calculating the angle between the two vectors, vec_t and vec_c, see equation 2.46. The length from the DC to the turn point in the XY-plane is vectorized as vec_t, see equation 2.44. The completion interval is projected onto the XY-plane and vectorized as vec_c, see equation 2.45. The built in MATLAB functions cross and dot are used to calculate the cross and dot product of the two vectors, respectively.

$$vec_t = [(X_2 - X_{DC}), (Y2 - Y_{DC}), 0] \tag{2.44}$$

$$vec_c = [(X_{ce} - X_{cs}), (Y_{ce} - Y_{cs}), 0] \tag{2.45}$$

$$arc_{azi} = \frac{\text{atan2}(\text{cross}(vec_t, vec_c), \text{dot}(vec_t, vec_c))}{\text{TR}} \tag{2.46}$$

Equation 2.46 does not apply to all scenarios. When the well turns more than 180° in the XY-plane, equation 2.47 has to be used. The criteria, $N > 0$, for this equation is explained in Appendix A.2.

$$arc_{azi} = \frac{360° - \text{atan2}(\text{cross}(vec_t, vec_c), \text{dot}(vec_t, vec_c))}{\text{TR}} \quad \text{if} \quad N > 0 \tag{2.47}$$

The length of the well from the drill center to the start of the completion in the R-plane (dR_{tot}) is then calculated using equation 2.48.

$$dR_{tot} = \sqrt{(X_2 - X_{DC})^2 + (Y_2 - Y_{DC})^2} + arc_{azi} \tag{2.48}$$

When the length of the well in the $R-$plane is known, the well path around the cylinder is unwrapped. By doing this, the well can be projected onto the RZ-plane. The next step is to construct the well path from the completion start to the drill center. To make the wells as short as possible, they are constructed with two build sections that are separated by a tangent section. The calculations for these build sections follow the same concepts as for the satellite wells' build section.

One more known parameter is needed to construct the well from bottom to top. The Z coordinate of the KOP is chosen to be an input parameter. This makes it possible to calculate the total depth available for the two build sections and the tangent section.

The total displacement in R (dR) and Z (dZ) direction in the two build sections are calculated by use of equation 2.9 and 2.10. The components, dR_{tan} and dZ_{tan}, of the tangent section are calculated using the equations 2.49 and 2.50. The length of the tangent section (L_{tan}) in the RZ-plane is then calculated by use of equation 2.51.

$$dR_{tan} = dR_{tot} - dR \tag{2.49}$$

$$dZ_{tan} = Z_{cs} - dZ - KOP_Z \tag{2.50}$$

$$L_{tan} = \sqrt{dR_{tan}^2 + dZ_{tan}^2} \tag{2.51}$$

The total arc length of the two build sections is calculated using equation 2.11. Consequently, the well path length (WPL) of each well is calculated by summing the lengths of the completion interval, tangent section, build sections/arc and distance from wellhead to KOP. Equation 2.52 is used.

$$\text{WPL} = L_c + L_{tan} + arc + \text{KOP}_Z \tag{2.52}$$

Finally, the average well path length is calculated by equation 2.17.

Two Drill Centers

To find the average well path length in the case of two drill centers, the same method as for one drill center is used. The only difference is that two different (a, b) coordinate systems are used, because both drill centers generate its associated coordinate system. One has its origin in the first drill center while the other has its origin in the second drill center. In addition, only half as many completions belong to each of the two coordinate systems.

Three Drill Centers

To find the average well path length in the case of three drill centers, the same method as for one drill center is used. There are now three different (a, b) coordinate systems. The first has its origin in the first drill center, the second has its origin in the second drill center while the third has its origin in the third drill center. In addition, only one third of the completions belong to each of the three coordinate systems.

Chapter 3

Results

3.1 Combinations and Placement of Drill Centers

The completion intervals and the associated drill center(s) are presented in the following tables. The results of the satellite well-, one drill center-, two drill centers-, and three drill centers field layouts are given in tables 3.1, 3.2, 3.3 and 3.4, respectively.

Table 3.1: Results in the case of satellite wells.

Completion start coordinates		Drill center coordinates	
X	Y	X	Y
11357	4500	10871	4803
10214	5214	9779	4841
7857	6286	7337	6528
8429	4500	8796	4060
9643	3571	10001	4018
7714	3786	7627	4352
7571	893	8029	1237
5357	3143	5625	3649
4500	4500	4148	4048
3357	1143	3660	1629
1071	1714	845	1187
786	786	982	248

Table 3.2: Results in the case of one drill center.

Completion start coordinates		Drill center coordinates	
X	Y	X	Y
11357	4500	6488.0	3336.3
10214	5214		
7857	6286		
8429	4500		
9643	3571		
7714	3786		
7571	893		
5357	3143		
4500	4500		
3357	1143		
1071	1714		
786	786		

Table 3.3: Results in the case of two drill centers.

	Completion start coordinates		Drill center coordinates	
	X	Y	X	Y
1st drill center	11357	4500	9202.3	4642.8
	10214	5214		
	7857	6286		
	8429	4500		
	9643	3571		
	7714	3786		
2nd drill center	3357	1143	3773.7	2029.8
	1071	1714		
	5357	3143		
	7571	893		
	786	786		
	4500	4500		

Table 3.4: Results in the case of three drill centers.

	Completion start coordinates		Drill center coordinates	
	X	Y	X	Y
1st drill center	7857	6286	7124.7	3527.0
	7714	3786		
	7571	893		
	5357	3143		
2nd drill center	9643	3571	9910.8	4446.3
	11357	4500		
	8429	4500		
	10214	5214		
3rd drill center	1071	1714	2428.5	2035.8
	786	786		
	4500	4500		
	3357	1143		

3.2 Wellbore Trajectory Calculations

This section presents the average well path lengths calculated from the methods in chapter 2.2. As mentioned, some parameters are set by the user; the build-up rate (BUR), turn rate (TR) and kickoff point (KOP_Z). Table 3.5 displays the input parameters that are used.

Table 3.5: Input parameters used in the calculations

Input parameter	Value	Unit
BUR	3/30	°/m
TR	3/30	°/m
KOP_Z	500	m

Due to the requirement from equation 2.18, some of the turn rates were adjusted. Table 3.6 shows the resulting turn rates:

Table 3.6: Resulting turn rates

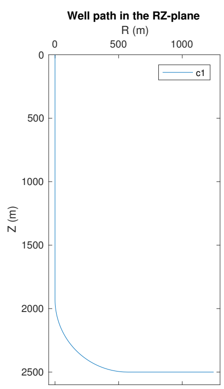
One drill center TR (°/m)	Two drill centers TR (°/m)	Three drill centers TR (°/m)
3/30	5/30	6/30

The adjusted turn rates are used in the calculations. Table 3.7 shows the resulting average well path lengths (WPL):

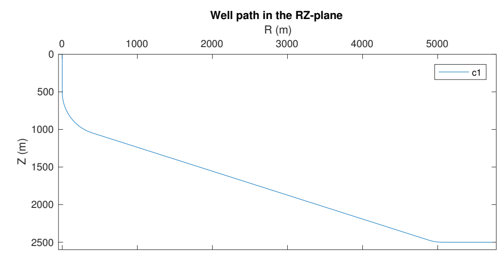
Table 3.7: Average well path lengths

Satellite wells WPL (m)	One drill center WPL (m)	Two drill centers WPL (m)	Three drill centers WPL (m)
3721.1	5930.2	4662.9	4303.7

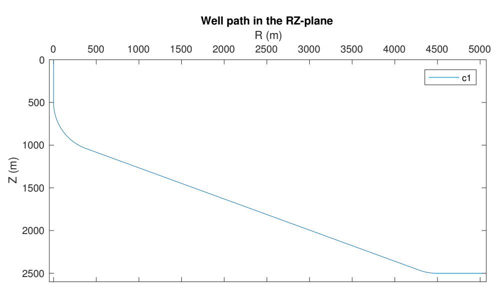
The results are shown graphically in figure 3.1. The figure shows the construction of the different well types.



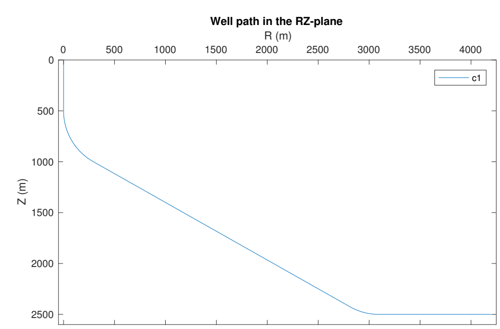
(a) Satellite wells



(b) One drill center



(c) Two drill centers



(d) Three drill centers

Figure 3.1: Wellbore trajectories for the different scenarios.

Chapter 4

Discussion

4.1 Discussion

The results from chapter 3.2 show that the average well path length decreases with increasing number of drill centers, as expected. As the well becomes longer, the more time it will take to drill it. This is primarily because of increased trip time since the length of the drill string increases. Considering that the drilling rigs are payed on a daily rate, the total well construction costs increase. Thus, the results indicate that a subsea field layout with only satellite wells is the cheapest solution.

Figure 3.1 illustrates that the deviated drilling length increases when the number of drill centers decreases. Deviated wells increase the technical risks related to an unstable wellbore. In addition, during tripping, long wells are left open longer. To carry out a controlled killing procedure, the drill string must be on bottom. A kick taken below the drill string can cause serious well control problems. These examples indicate that shorter and less deviated wells minimize technical risk in addition to costs.

The methods presented in chapter 2 are mathematically reliable, but because of the limitations they are only applicable for unrealistic field data. Although the completion intervals are unreal, the resulting comparison of the average well path lengths indicate a realistic difference. To make the input data more realistic, the methods can be adjusted for non-horizontal completions. The calculations can also consider different kickoff points (KOP), build-up rates (BUR) and turn rates (TR) for the different completion intervals. These proposals are minor adjustments that require a lot of work without making the differences in well path lengths significantly larger. Adjustments that are considered valuable are calculations that can compare more than 12 completion intervals, completion intervals located at different depths and subsea field layouts with more than three common drill centers. The scale of the field data will then be closer to the reality.

As mentioned in chapter 1.1, additional costs such as hardware and installation costs are part of the total subsea field development costs. An additional study of these costs will be necessary to identify the optimal subsea layout. Costs related to flowlines, control umbilicals, trawl protection, and service and injection lines are examples of subsea facilities hardware costs. Installation costs vary depending on for example vessel re-positioning and the methods used for installing flowlines.

4.2 Conclusion

The average well path length decreases as the number of drill centers increases. Consequently, the case of satellite wells is the optimal solution to minimize the well path lengths.

The average well path lengths are of interest to optimize subsea field development. Instead of minimizing the cost of individual contracts (drilling contractor, subsea EPC and subsea installation), the goal is to minimize the sum of these. As the results indicate, the case of satellite wells minimizes the drilling cost. However, saving on drilling cost has penalties, e.g. template structures are cheaper per well than individual wells and then there are the tie-back costs from satellite wells to manifold. Consequently, the results provided in this project can not identify the optimal field layout, but they are crucial in order to minimize the drilling cost.

4.3 Recommendations for Further Work

Drilling costs

The most important work that remains to be done in the drilling aspect of this model, is to tie the well path lengths and drilling costs together. This can be done by using equation 4.1, or by using a more precise equation. In addition, the costs of rig re-positioning during the drilling operations can be taken into account.

$$\text{drilling cost} = \frac{\text{well length}}{\text{daily construction rate}} \times \text{daily rig cost rate} \tag{4.1}$$

The limitation of 12 completion intervals should be extended. More completion intervals will cause the differences in average well path lengths to be greater. As these differences increase, the cost savings based on drilling will increase as well.

The program should be updated and improved to handle calculations for more than three drill centers. This causes the average well path lengths and cost savings for several field layouts to be evaluated.

Subsea Hardware and Installation Costs

To find the optimal subsea layout, factors like those mentioned in chapter 4.1 have to be accounted for. When comparing a subsea satellite layout and a layout with one or more common drill centers, the subsea hardware and installation costs have to be evaluated. These costs must be combined with the drilling costs, to optimize the overall field development costs.

The factors that have the highest impact on the subsea hardware costs are listed in table 4.1. The expected effects on costs per well are also listed.

Table 4.1: Effects of subsea hardware costs

Subsea hardware	Satellite	Common drill centers
Christmas tree	Same	Same
Template	None	"Higher"
Manifold	Higher	Lower
Trawl protection	Higher	Lower
Tie-back	Higher	Lower
Boosting	Higher	Lower

Christmas tree (XT) costs are the same for all field layouts since each well needs its own XT and their design is independent of the foundation around the wells. There is no need for templates on satellite wells as the XTs are supported by the conductor and the surface casing. Thus, the template related costs (excluding manifold and trawl protection) are "higher" per well for the cases of common drill centers since templates are required. If there are more than four or six satellite wells tied back to the manifold, the manifold costs will be higher for the case of satellites. This is mainly due to the size of the manifold. In the case of satellite wells, the manifold is placed on a suction anchor or piled base. In the case of common drill centers, the manifold is integrated in the template. If the field requires trawl protection, each satellite well and manifold must be equipped with its own trawl protection. As mentioned in chapter 4.2, the protection structure makes integrated template structures cheaper per well. Tie-back costs increase as the length of the flowlines, control umbilicals and service and injection lines increase. If the flowlines become too long, then pressure support is needed to transport the wellstream and extra boosting equipment is needed.

The installation costs vary depending on the vessels and methods used. To include these in the model, the user must be able to specify the installation vessels and methods as input. Assuming that lighter vessels can be used when there is no template to install, the costs will decrease for the case of satellites since lighter vessels are cheaper. On the other hand, time will be lost in moving the installation vessels between each installation. In addition, tie-back distances increase. If these distances become large enough, then reel-based installation will be cheaper than installing for example flowlines piece by piece with a crane. These are just examples of considerations that have to be made, more factors will be included in the future model.

Intervention Costs

If the field is developed subsea, subsea intervention will be required. This will increase the costs of the subsea developments. Additionally, well intervention will be required in all scenarios. Well intervention in a subsea field is challenging and expensive because intervention vessels or mobile offshore drilling units are needed. Subsea intervention may not be profitable because of these cost. As a result, the recovery factor can be reduced compared to a fixed platform where intervention is cheaper.

Appendix A

Derivations

A.1 Azimuth with the Four-Quadrant Inverse Tangent

To be able to see any pattern in the calculation of the azimuth, all arrangements of the completion interval can be drawn in a coordinate system. A sketch of the first quadrant arrangements are shown in figure A.1.

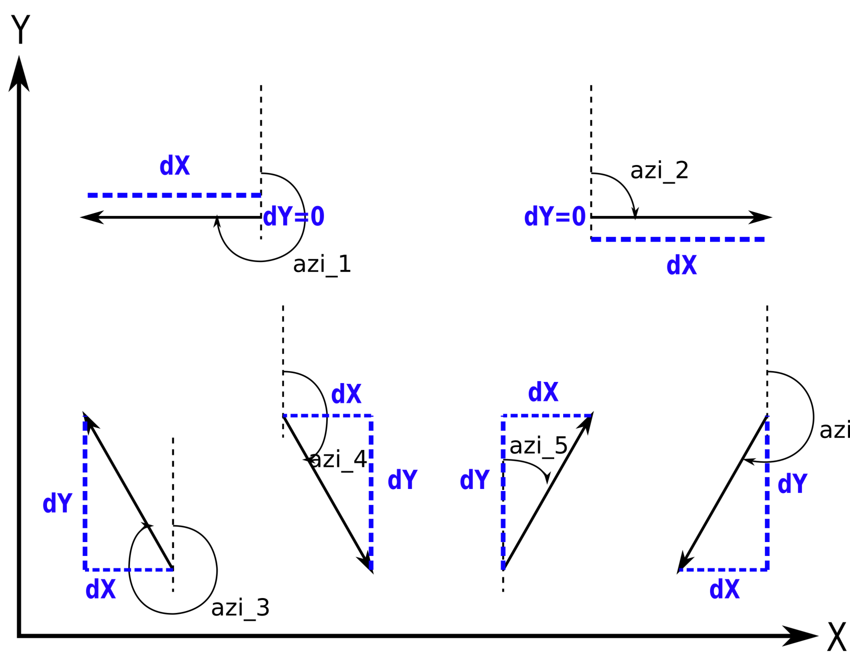


Figure A.1: Completion arrangements in the first quadrant

Using trigonometric relations the azimuth can be found. dX and dY are defined as the completion start coordinate subtracted from the completion end coordinate. This yields the following solutions:

$$azi_1 = \frac{3\pi}{2} \tag{A.1}$$

$$azi_2 = \frac{\pi}{2} \tag{A.2}$$

$$azi_3 = \frac{3\pi}{2} - \arctan\frac{dX}{dY} \tag{A.3}$$

$$azi_4 = \frac{\pi}{2} - \arctan\frac{dY}{dX} \tag{A.4}$$

$$azi_5 = \arctan\frac{dX}{dY} \tag{A.5}$$

$$azi_6 = \pi + \arctan\frac{dX}{dY} \tag{A.6}$$

If one defines the azimuth in the range $-\pi < azi < \pi$, the same expressions become:

$$azi_1 = -\frac{\pi}{2} \tag{A.7}$$

$$azi_2 = \frac{\pi}{2} \tag{A.8}$$

$$azi_3 = \arctan\frac{dX}{dY} \tag{A.9}$$

$$azi_4 = \arctan\frac{dX}{dY} + \pi \tag{A.10}$$

$$azi_5 = \arctan\frac{dX}{dY} \tag{A.11}$$

$$azi_6 = \arctan\frac{dX}{dY} - \pi \tag{A.12}$$

The equations for azi_1, A.1 and A.7, are only valid when dX is negative and dY is equal to zero. The equations for azi_2, A.2 and A.8, are only valid when dX is positive and dY is equal to zero. The equations for azi_3, A.3 and A.9, are valid when dX is negative or equal to zero, and dY is positive. The equations for azi_4, A.4 and A.10, are only valid when dX is positive and dY is negative. The equations for azi_5, A.5 and A.11, are only valid when dX is positive or equal to zero, and dY is positive. The equations for azi_6, A.6 and A.12, are only valid when dX is negative or equal to zero, and dY is negative. This set of equations have the same conditions as the four-quadrant inverse tangent, atan2(dX, dY):

$$\operatorname{atan2}(dX, dY) = \begin{cases} \arctan \frac{dX}{dY}, & \text{if } dY > 0 \\ \arctan \frac{dX}{dY} + \pi, & \text{if } dY < 0 \text{ and } dX \geq 0 \\ \arctan \frac{dX}{dY} - \pi, & \text{if } dY < 0 \text{ and } dX < 0 \\ +\frac{\pi}{2}, & \text{if } dY = 0 \text{ and } dX > 0 \\ -\frac{\pi}{2}, & \text{if } dY = 0 \text{ and } dX < 0 \\ \text{undefined} & \text{if } dY = 0 \text{ and } dX = 0 \end{cases} \tag{A.13}$$

The final expression becomes:

$$azi = \operatorname{atan2}(dX, dY) \tag{A.14}$$

A.2 Criteria for Equation 2.47

Some of the wells turn more than 180° in the XY-plane. When calculating the angle between two vectors using equation 2.46, it is always the smallest angle that is calculated. Thus, equation 2.47 must be used when the wells turn more than 180°. The criteria used for these cases checks if the completion vector, vec_c, has a direction towards an infinite line that is an extension of the tangent vector vec_t. Figure A.2 shows the concept of the criteria.

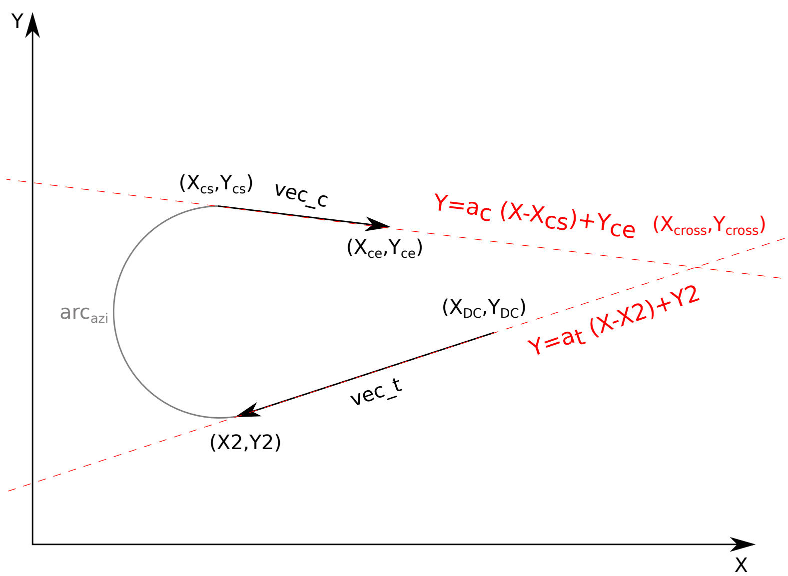


Figure A.2: Sketch of the vectors vec_c and vec_t and the lines that passes through them.

The extensions of vector vec_t and vec_c are expressed in equation A.17 and A.18, respectively. Their slopes are expressed in equation A.15 and A.16, respectively.

$$a_t = \frac{Y2 - Y_{DC}}{X2 - X_{DC}} \tag{A.15}$$

$$a_c = \frac{Y_{ce} - Y_{cs}}{X_{ce} - x_{cs}} \tag{A.16}$$

$$Y = a_t(X - X2) + Y2 \tag{A.17}$$

$$Y = a_c(X - X_{cs}) + Y_{cs} \tag{A.18}$$

The crossing point is identified by finding the X when equation A.17 and A.18 are the same, X_{cross}. The expression for this X, X_{cross}, is shown in equation A.19.

$$X_{cross} = \frac{Y_{cs} - Y2 + a_t \times X2 - a_c \times X_{cs}}{a_t - a_c} \tag{A.19}$$

The next step is to find out if this crossing X, X_{cross}, lies in front of the completion interval vector, vec_c, or behind it. This is solved by a ratio, N, see equation A.20 and A.21.

$$X_{cross} = X_{cs} + N \times (X_{ce} - X_{cs}) \tag{A.20}$$

$$N = \frac{X_{cross} - X_{cs}}{X_{ce} - X_{cs}} \tag{A.21}$$

If the ratio, N, is positive, the completion interval has a direction towards the tangent, and equation 2.47 should be used.

Appendix B

MATLAB

B.1 run tool

```
%completion coordinates (must be positive coordinates) 1.4/1000
C=[11357 4500 2500 11929 4143 2500; 10214 5214 2500 10714 5643 2500; 7857 6286 2500 8857 5821 2500; ...
    8429 4500 2500 7714 5357 2500; 9643 3571 2500 9071 2857 2500; 7714 3786 2500 7857 2857 2500; ...
    7571 893 2500 7143 571 2500; 5357 3143 2500 4714 1929 2500; 4500 4500 2500 5000 5143 2500; ...
    3357 1143 2500 3000 571 2500; 1071 1714 2500 1500 2714 2500; 786 786 2500 500 1571 2500];

%input parameter
BUR=3/30; %deg/m
KOPz=500; %m
TR=BUR; %deg/m
N_dc=3;

%find average wellbore length - only satelites
WPL_avg_sat=plot_satellite(C,BUR);

%T displays the results in a table
if N_dc == 1
    %find average wellbore length - one drill center
    WPL_avg_dc1=plot_one_dc(C,BUR,KOPz,TR);
    T=table(WPL_avg_sat,WPL_avg_dc1);
    disp(T)
elseif N_dc == 2
    %find average wellbore length - one and two drill centers
    WPL_avg_dc1=plot_one_dc(C,BUR,KOPz,TR);
    WPL_avg_dc2=plot_two_dc(C,BUR,KOPz,TR);
    T=table(WPL_avg_sat,WPL_avg_dc1,WPL_avg_dc2);
    disp(T)
elseif N_dc == 3
    %find average wellbore length - one, two and three drill centers
    WPL_avg_dc1=plot_one_dc(C,BUR,KOPz,TR);
    WPL_avg_dc2=plot_two_dc(C,BUR,KOPz,TR);
    WPL_avg_dc3=plot_three_dc(C,BUR,KOPz,TR);
    T=table(WPL_avg_sat,WPL_avg_dc1,WPL_avg_dc2,WPL_avg_dc3);
    disp(T)
end
```

B.2 get BUA

```
function [L_c,BUA]=get_BUA(C)
%calculates the length and inclination of each completion interval

%lengt and height of completion interval
L_c=(sqrt((C(:,4)-C(:,1)).^2+(C(:,5)-C(:,2)).^2+(C(:,6)-C(:,3)).^2))';
dZ_c=C(:,6)'-C(:,3)';

%inclination of completion interval (build-up angle)
BUA=acosd(dZ_c./L_c);
for i=1:size(C,1)
    if dZ_c(i) < 0
        BUA(i)=90-asind(dZ_c/L_c(i));
    end
end
end
```

B.3 plot satellite

```
function WPL_avg=plot_satellite(C,BUR)
%calculates the average well path length of all satellite wells

%completion interval length and build-up angle
[L_c,BUA]=get_BUA(C);

%arc length and radius of curvature
arc=BUA/BUR;
ROC=(360*arc)./(2*pi*BUA);

%azimuth and completion start coordinates
dR=ROC+ROC.*sind(BUA-90);
dZ=ROC.*cosd(BUA-90);
azi_c=atan2d((C(:,4)'-C(:,1)'),(C(:,5)'-C(:,2)'));
for i=1:size(C,1)
    if BUA(i)<90
        dR(i)=ROC(i)-ROC(i)*cosd(BUA(i));
        dZ(i)=ROC(i)*sind(BUA(i));
    end
end

%coordinates of kick off point
KOP(:,1)=C(:,1)-dR'.*sind(azi_c');
KOP(:,2)=C(:,2)-dR'.*cosd(azi_c');
KOP(:,3)=C(:,3)-dZ';

%average well path length of all satellite wells
WPL=L_c+arc+KOP(:,3)';
WPL_avg=sum(WPL)/length(WPL);

%the following calculations are only for the intention of plotting the wells in 2D

%coordinates of the circle center
R_cc1=ROC;
Z_cc1=KOP(:,3)';

%amount of columns needed in the R and Z matrices
K=C(:,6)';
```

```
for i=1:size(C,1)
    if C(i,3)-C(i,6)==0
        K(i)=C(i,3)+L_c(i);
    end
end

%creating the R and Z matrices
R=zeros(size(C,1),ceil(max(K)));
Z=zeros(size(C,1),ceil(max(K)));

%R and Z coordinates of completion start (end of build) and completion end
Rc1e=ROC-ROC.*cosd(BUA);
Zc1e=ROC.*sind(BUA)+KOP(:,3)';
R_ce=dR+(sqrt((C(:,4)-C(:,1)).^2+(C(:,5)-C(:,2)).^2))';
m1=(R_ce-Rc1e)./(C(:,6)-C(:,3))';
b1=Rc1e-m1.*Zc1e;

%filling the Z matrix with numbers from 1 to depth of completion end
for i=1:size(C,1)
    for j=1:C(i,6)
        A=1:C(i,6);
        Z(i,j)=A(j);
    end
end

%filling the R matrix with the corresponding coordinates
N=zeros(1,size(C,1));
for j=1:size(C,1)
for i=1:C(j,6)
    %coordinates above KOP
    if Z(j,i)<=KOP(j,3)
        R(j,i)=0;
    %coordinates of the build section
    elseif Z(j,i)>KOP(j,3) && Z(j,i)<=Zc1e(j)
        R(j,i)=-sqrt(ROC(j)^2-(Z(j,i)-Z_cc1(j))^2)+R_cc1(j);
    %coordinates of the completion interval
    else
        R(j,i)=m1(j)*Z(j,i)+b1(j);
    end

    %coordinates of a horizontal completion interval
    if C(j,3)-C(j,6)==0
        B=dR(j):(dR(j)+L_c(j));
        R(j,C(j,3)+length(B))=dR(j)+L_c(j);
        N(j)=C(j,3)+length(B);
                for k=1:length(B)
                    R(j,C(j,3)+k-1)=B(k);
                    Z(j,C(j,3)+k)=Z(j,C(j,3)+k-1);
                end
     end
end
end

%plotting all wells as a two-dimensional figure
% for i=1:size(C,1)
%     figure()
%     plot(R(i,1:N(i)),flipud(Z(i,1:N(i))));
%     set(gca,'XAxisLocation','top','YAxisLocation','left','ydir','reverse')
%     axis equal
%     title('Well path in the RZ-plane')
%     xlabel('R (m)')
%     ylabel('Z (m)')
```

```
%      xlim([-50 (R(i,N(i))+50)])
%      ylim([0 2600])
% end
end
```

B.4 get one DC

```
function [dc]=get_one_dc(C)
%calulates the optimized coordinates of one common drill center

%compute average X and Y coordinates
X_avg=sum(C(:,1))/size(C,1);
Y_avg=sum(C(:,2))/size(C,1);

%gather the coordinates in a common drill center vector
dc=[X_avg Y_avg 0];
end
```

B.5 get turn one DC

```
function [X2,Y2,TR]=get_turn_one_dc(C,DC,TR)
%calculates the turn point in the XY-plane

%radius of turn
ROT=360/(2*pi*TR);

%check if distance between WH and completion start is less than 2*ROT
k=1;
while k <= size(C,1)
    if (sqrt((DC(1)-C(k,1))^2+(DC(2)-C(k,2))^2) < 2*ROT)
        %increase the turn rate
        TR=TR+1/30;
        ROT=360/(2*pi*TR);
    else
        k=k+1;
    end
end

%azimuth of completion interval
azi_c=atan2d((C(:,4)'-C(:,1)'),(C(:,5)'-C(:,2)'));

%completion start coordinates in a new coordinate system (a,b) with origin in the drill center
a1=C(:,1)'-DC(1);
b1=C(:,2)'-DC(2);

%rotation angle needed to place the coordinates in a rotated coordinate system
rotate=180-azi_c;

%completion start coordinates in the rotated coordinate system (xi,yi) with origin in the drill center
xi1=a1.*cosd(rotate)+b1.*sind(rotate);
yi1=-a1.*sind(rotate)+b1.*cosd(rotate);

%circle of turn center coordinates in (xi,yi)
x_cci=xi1+ROT;
y_cci=yi1;
for i=1:size(C,1)
```

```
    if xi1(i)>0
        x_cci(i)=xi1(i)-ROT;
    end
end

%distances from drill center to the circles of turn
H=sqrt((x_cci).^2+(y_cci).^2);
B=sqrt((H.^2-ROT^2));
theta=(asind(ROT./H));

%coordinates of turn point in (xi,yi)
beta=zeros(1,size(C,1));
dxi=zeros(1,size(C,1));
dyi=zeros(1,size(C,1));
for i=1:size(C,1)
if x_cci(i)>0 && y_cci(i)>0
    if x_cci(i)>xi1(i)
        beta(i)=theta(i)+atand(x_cci(i)/y_cci(i));
        dxi(i)=B(i)*sind(beta(i));
        dyi(i)=B(i)*cosd(beta(i));
    elseif x_cci(i)<xi1(i)
        beta(i)=theta(i)+atand(y_cci(i)/x_cci(i));
        dxi(i)=B(i)*cosd(beta(i));
        dyi(i)=B(i)*sind(beta(i));
    end
elseif x_cci(i)>0 && y_cci(i)<0
    if x_cci(i)>xi1(i)
        beta(i)=theta(i)-atand(y_cci(i)/x_cci(i));
        dxi(i)=B(i)*cosd(beta(i));
        dyi(i)=-B(i)*sind(beta(i));
    elseif x_cci(i)<xi1(i)
        beta(i)=theta(i)-atand(x_cci(i)/y_cci(i));
        dxi(i)=B(i)*sind(beta(i));
        dyi(i)=-B(i)*cosd(beta(i));
    end
elseif x_cci(i)<0 && y_cci(i)<0
    if x_cci(i)>xi1(i)
        beta(i)=theta(i)+atand(x_cci(i)/y_cci(i));
        dxi(i)=-B(i)*sind(beta(i));
        dyi(i)=-B(i)*cosd(beta(i));
    elseif x_cci(i)<xi1(i)
        beta(i)=theta(i)+atand(y_cci(i)/x_cci(i));
        dxi(i)=-B(i)*cosd(beta(i));
        dyi(i)=-B(i)*sind(beta(i));
    end
elseif x_cci(i)<0 && y_cci(i)>0
    if x_cci(i)>xi1(i)
        beta(i)=theta(i)-atand(y_cci(i)/x_cci(i));
        dxi(i)=-B(i)*cosd(beta(i));
        dyi(i)=B(i)*sind(beta(i));
    elseif x_cci(i)<xi1(i)
        beta(i)=theta(i)-atand(x_cci(i)/y_cci(i));
        dxi(i)=-B(i)*sind(beta(i));
        dyi(i)=B(i)*cosd(beta(i));
    end
end
end

%coordinates of turn point in (a,b)
a2=dxi.*cosd(rotate)-dyi.*sind(rotate);
b2=dxi.*sind(rotate)+dyi.*cosd(rotate);
```

```
%coordinates of turn point in (X,Y)
X2=DC(1)+a2;
Y2=DC(2)+b2;
end
```

B.6 plot one DC

```
function WPL_avg=plot_one_dc(C,BUR,KOPz,TR)
%calculates the average well path length of all wells drilled from one drill center

%compute optimized drillcenter
DC=get_one_dc(C);

%find turn point in the XY-plane
[X2,Y2,TR]=get_turn_one_dc(C,DC,TR);

%coordinates of the 1st kickoff point
KOP=[DC(1) DC(2) KOPz];

%completion interval length and build-up angle
[L_c,BUA]=get_BUA(C);

%arc length in the XY-plane
vec_t=[X2'-DC(1) Y2'-DC(2) zeros(size(C,1),1)];
vec_c=[C(:,4)-C(:,1) C(:,5)-C(:,2) zeros(size(C,1),1)];
alpha_azi=zeros(1,size(C,1));
a_t=(vec_t(:,2)./vec_t(:,1))';
a_c=(vec_c(:,2)./vec_c(:,1))';
x=(C(:,2)'-Y2+a_t.*X2-a_c.*C(:,1)')./(a_t-a_c);
N=(x-C(:,1)')./(C(:,4)-C(:,1))';
for i=1:size(C,1)
    %calculate angle between two vectors
    alpha_azi(i)=atan2d(norm(cross(vec_t(i,:),vec_c(i,:))),dot(vec_t(i,:),vec_c(i,:)));
    if N(i)>0
        alpha_azi(i)=360-alpha_azi(i);
    end
end
arc_azi=alpha_azi./TR;

%RZ coordinates of start of completion interval
dRtot=sqrt((X2-DC(1)).^2+(Y2-DC(2)).^2)+arc_azi;
dZtot=C(:,3)'-DC(3);

%arc length in the RZ plane and radius of curvature
arc=BUA/BUR;
ROC=(360*arc)./(2*pi*BUA);

%displacements in R and Z due to both build sections
dR=ROC+ROC.*sind(BUA-90);
dZ=ROC.*cosd(BUA-90);
for i=1:size(C,1)
    if BUA(i) < 90
        dR(i)=ROC(i)-ROC(i)*cosd(BUA(i));
        dZ(i)=ROC(i)*sind(BUA(i));
    end
end

%length of tangent section
dRtan=dRtot-dR;
```

```
dZtan=dZtot-dZ-KOP(3);
Ltan=sqrt(dRtan.^2+dZtan.^2);

%first build-up angle
BUA1=atand(dRtan./dZtan);

%average wellpath length
WPL=L_c+arc+Ltan+KOP(3);
WPL_avg=sum(WPL)/length(WPL);

%the following calculations are only for the intention of plotting the wells in 2D

%coordinates of the 1st circle of build center
R_cc1=ROC;
Z_cc1=KOP(3);

%amount of columns needed in the R and Z matrices
K=C(:,6)';
for i=1:size(C,1)
    if C(i,3)-C(i,6)==0
        K(i)=C(i,3)+L_c(i);
    end
end

%creating the R and Z matrices
R=zeros(size(C,1),ceil(max(K)));
Z=zeros(size(C,1),ceil(max(K)));

%R and Z coordinates of 1st and 2nd build sections and completion coordinates
Rc1e=ROC-ROC.*cosd(BUA1);
Zc1e=ROC.*sind(BUA1)+KOP(3);
Rc2s=Ltan.*sind(BUA1)+Rc1e;
Zc2s=Ltan.*cosd(BUA1)+Zc1e;
m1=(Rc2s-Rc1e)./(Zc2s-Zc1e);
b1=Rc1e-m1.*Zc1e;
R_cc2=dRtot;
Z_cc2=C(:,3)'-ROC;
R_ce=dRtot+(sqrt((C(:,4)-C(:,1)).^2+(C(:,5)-C(:,2)).^2))';
Rc2e=dRtot;
Zc2e=C(:,3)';
m2=(R_ce-Rc2e)./(C(:,6)'-Zc2e);
b2=Rc2e-m2.*Zc2e;

%filling the Z matrix with numbers from 1 to depth of completion end
for i=1:size(C,1)
    for j=1:C(i,6)
        A=1:C(i,6);
        Z(i,j)=A(j);
    end
end

%filling the R matrix with the corresponding coordinates
N=zeros(1,size(C,1));
for j=1:size(C,1)
for i=1:C(j,6)
    %coordinates above the 1st kickoff point
    if Z(j,i)<=KOP(3)
        R(j,i)=0;
    %coordinates of the 1st build section
    elseif Z(j,i)>KOP(3) && Z(j,i)<=Zc1e(j)
        R(j,i)=-sqrt(ROC(j)^2-(Z(j,i)-Z_cc1)^2)+R_cc1(j);
    %coordinates of the tangent section
```

```
    elseif Z(j,i)>Zc1e(j) && Z(j,i)<=Zc2s(j)
        R(j,i)=m1(j)*Z(j,i)+b1(j);
    %coordinates of the 2nd build section
    elseif Z(j,i)>Zc2s(j) && Z(j,i)<C(j,3)
        R(j,i)=-sqrt(ROC(j)^2-(Z(j,i)-Z_cc2(j))^2)+R_cc2(j);
    %coordinates of the completion interval
    else
        R(j,i)=m2(j)*Z(j,i)+b2(j);
    end

    %coordinates of a horizontal completion interval
    if C(j,3)-C(j,6)==0
        B=dRtot(j):(dRtot(j)+L_c(j));
        R(j,C(j,3)+length(B))=dRtot(j)+L_c(j);
        N(j)=C(j,3)+length(B);
                for k=1:length(B)
                    R(j,C(j,3)+k-1)=B(k);
                    Z(j,C(j,3)+k)=Z(j,C(j,3)+k-1);
                end
     end
end
end

%plotting all wells as a two-dimensional figure
% for i=1:size(C,1)
%     figure()
%     plot(R(i,1:N(i)),flipud(Z(i,1:N(i))));
%     set(gca,'XAxisLocation','top','YAxisLocation','left','ydir','reverse')
%     axis equal
%     title('Well path in the RZ-plane')
%     xlabel('R (m)')
%     ylabel('Z (m)')
%     xlim([-50 (R(i,N(i))+50)])
%     ylim([0 2600])
% end
end
```

B.7 get two DC

```
function [DC1,DC2,X_opt,Y_opt,Z_opt]=get_two_dc(C,N_dc)
%calulates the optimized coordinates of two drill centers

%number of completion intervals
N=size(C,1);
M=(1:N);

%all possible combinations of template 1
R=nchoosek(M,N/N_dc);

%the corresponding combinations of template 2
R_rest=zeros(size(R));
for i=1:size(R,1)
    Rtemp=R(i,:);
    R2=zeros(1,N/N_dc);
    j=1;
    while j <= size(R,2)
        V=randi(N);
        if sum(Rtemp==V)==0 && sum(R2==V)==0
            R2(j)=V;
```

```
            j=j+1;
        end
    end
    R_rest(i,:)=R2;
end

%coordinates corresponding to values in R and R_rest
X=zeros(N/N_dc,size(R,1));
Y=zeros(N/N_dc,size(R,1));

X_rest=zeros(N/N_dc,size(R_rest,1));
Y_rest=zeros(N/N_dc,size(R_rest,1));
for i=1:size(R,1)
    for j = 1:N/N_dc
    X(j,i) = C(R(i,j),1);
    Y(j,i) = C(R(i,j),2);

    X_rest(j,i) = C(R_rest(i,j),1);
    Y_rest(j,i) = C(R_rest(i,j),2);
    end
end

%drillcenters
X_dc=zeros(1,size(X,2));
Y_dc=zeros(1,size(Y,2));

X_dc_rest=zeros(1,size(X,2));
Y_dc_rest=zeros(1,size(Y,2));
for i=1:size(X,2)
    X_dc(i)=sum(X(:,i))/size(X,1);
    Y_dc(i)=sum(Y(:,i))/size(Y,1);

    X_dc_rest(i)=sum(X_rest(:,i))/size(X_rest,1);
    Y_dc_rest(i)=sum(Y_rest(:,i))/size(Y_rest,1);
end

%distances from drill center to all points
dist=zeros(size(X));
dist_rest=zeros(size(X));
for i=1:size(dist,2)
    for j=1:size(dist,1)
        dist(j,i)=sqrt((X(j,i)-X_dc(i))^2+(Y(j,i)-Y_dc(i))^2);
        dist_rest(j,i)=sqrt((X_rest(j,i)-X_dc_rest(i))^2+(Y_rest(j,i)-Y_dc_rest(i))^2);
    end
end

%average distances for each group
dist_avg=zeros(1,size(dist,2));
dist_avg_rest=zeros(1,size(dist,2));
for i=1:size(dist,2)
        dist_avg(i)=sum(dist(:,i))/size(dist,1);
        dist_avg_rest(i)=sum(dist_rest(:,i))/size(dist_rest,1);
end

%the combination that yields the shortest well path lengths
dist_tot=(dist_avg+dist_avg_rest)./2;
[g,col]=min(dist_tot);

X_opt=[C(R(col,:),1)' C(R_rest(col,:),1)'];
Y_opt=[C(R(col,:),2)' C(R_rest(col,:),2)'];
Z_opt=[C(R(col,:),3)' C(R_rest(col,:),3)'];

```

```
X_opt(2,:)=[C(R(col,:),4)' C(R_rest(col,:),4)'];
Y_opt(2,:)=[C(R(col,:),5)' C(R_rest(col,:),5)'];
Z_opt(2,:)=[C(R(col,:),6)' C(R_rest(col,:),6)'];

DC1=[X_dc(col) Y_dc(col) 0];
DC2=[X_dc_rest(col) Y_dc_rest(col) 0];
end
```

B.8 get turn two DC

```
function [X2,Y2,TR] = get_turn_two_dc(C,DC1,DC2,X_opt,Y_opt,TR)
%calculates the turn point in the XY-plane

%radius of turn
ROT=360/(2*pi*TR);

%check if distance between WH and completion start is less than 2*ROT
k=1;
while k <= (size(C,1)/2)
    if (sqrt((DC1(1)-X_opt(1,k))^2+(DC1(2)-Y_opt(1,k))^2) < 2*ROT) || ...
            (sqrt((DC2(1)-X_opt(1,k+6))^2+(DC2(2)-Y_opt(1,k+6))^2) < 2*ROT)
        %increase the turn rate
        TR=TR+1/30;
        ROT=360/(2*pi*TR);
    else
        k=k+1;
    end
end

%azimuth of completion interval
azi_c=atan2d((X_opt(2,:)-X_opt(1,:)),(Y_opt(2,:)-Y_opt(1,:)));

%completion start coordinates in two new coordinate systems (a,b) with originin the drill centers
a1(1:6)=X_opt(1,1:6)-DC1(1);        a1(7:12)=X_opt(1,7:12)-DC2(1);
b1(1:6)=Y_opt(1,1:6)-DC1(2);        b1(7:12)=Y_opt(1,7:12)-DC2(2);

%rotation angle needed to place the coordinates in a rotated coordinate system
rotate=180-azi_c;

%completion start coordinates in the rotated coordinate systems (xi,yi) with origin in the drill centers
xi1=a1.*cosd(rotate)+b1.*sind(rotate);
yi1=-a1.*sind(rotate)+b1.*cosd(rotate);

%circle of turn center coordinates in (xi,yi)
x_cci=xi1+ROT;
y_cci=yi1;
for i=1:size(C,1)
    if xi1(i)>0
        x_cci(i)=xi1(i)-ROT;
    end
end

%distances from drill centers to the circles of turn
H=sqrt((x_cci).^2+(y_cci).^2);
B=sqrt(H.^2-ROT^2);
tetha=(asind(ROT./H));

%coordinates of turn point in (xi,yi)
betha=zeros(1,size(C,1));
```

```
dxi=zeros(1,size(C,1));
dyi=zeros(1,size(C,1));
for i=1:size(C,1)
if x_cci(i)>0 && y_cci(i)>0
    if x_cci(i)>xi1(i)
        betha(i)=tetha(i)+atand(x_cci(i)/y_cci(i));
        dxi(i)=B(i)*sind(betha(i));
        dyi(i)=B(i)*cosd(betha(i));
    elseif x_cci(i)<xi1(i)
        betha(i)=tetha(i)+atand(y_cci(i)/x_cci(i));
        dxi(i)=B(i)*cosd(betha(i));
        dyi(i)=B(i)*sind(betha(i));
    end
elseif x_cci(i)>0 && y_cci(i)<0
    if x_cci(i)>xi1(i)
        betha(i)=tetha(i)-atand(y_cci(i)/x_cci(i));
        dxi(i)=B(i)*cosd(betha(i));
        dyi(i)=-B(i)*sind(betha(i));
    elseif x_cci(i)<xi1(i)
        betha(i)=tetha(i)-atand(x_cci(i)/y_cci(i));
        dxi(i)=B(i)*sind(betha(i));
        dyi(i)=-B(i)*cosd(betha(i));
    end
elseif x_cci(i)<0 && y_cci(i)<0
    if x_cci(i)>xi1(i)
        betha(i)=tetha(i)+atand(x_cci(i)/y_cci(i));
        dxi(i)=-B(i)*sind(betha(i));
        dyi(i)=-B(i)*cosd(betha(i));
    elseif x_cci(i)<xi1(i)
        betha(i)=tetha(i)+atand(y_cci(i)/x_cci(i));
        dxi(i)=-B(i)*cosd(betha(i));
        dyi(i)=-B(i)*sind(betha(i));
    end
elseif x_cci(i)<0 && y_cci(i)>0
    if x_cci(i)>xi1(i)
        betha(i)=tetha(i)-atand(y_cci(i)/x_cci(i));
        dxi(i)=-B(i)*cosd(betha(i));
        dyi(i)=B(i)*sind(betha(i));
    elseif x_cci(i)<xi1(i)
        betha(i)=tetha(i)-atand(x_cci(i)/y_cci(i));
        dxi(i)=-B(i)*sind(betha(i));
        dyi(i)=B(i)*cosd(betha(i));
    end
end
end

%coordinates of turn point in (a,b)
a2=dxi.*cosd(rotate)-dyi.*sind(rotate);
b2=dxi.*sind(rotate)+dyi.*cosd(rotate);

%coordinates of turn point in (X,Y)
X2(1:6)=DC1(1)+a2(1:6);    X2(7:12)=DC2(1)+a2(7:12);
Y2(1:6)=DC1(2)+b2(1:6);    Y2(7:12)=DC2(2)+b2(7:12);
end
```

B.9 plot two DC

```
function WPL_avg=plot_two_dc(C,BUR,KOPz,TR)
%calculates the average well path length of all wells drilled from two drill centers
```

```

%input parameter
N_dc=2;

%compute optimized drillcenters and corresponding groups
[DC1,DC2,X_opt,Y_opt,Z_opt]=get_two_dc(C,N_dc);

%rearrange the completion intervals
C(:,1)=X_opt(1,:);    C(:,2)=Y_opt(1,:);    C(:,3)=Z_opt(1,:);
C(:,4)=X_opt(2,:);    C(:,5)=Y_opt(2,:);    C(:,6)=Z_opt(2,:);

%find turn point in the XY-plane
[X2,Y2,TR]=get_turn_two_dc(C,DC1,DC2,X_opt,Y_opt,TR);

%coordinates of the 1st kickoff points
KOP=[DC1(1) DC1(2) KOPz; DC2(1) DC2(2) KOPz];

%completion interval length and build-up angle
[L_c,BUA]=get_BUA(C);

%arc length in the XY-plane
vec_t(1:6,:)=[(X2(1:6)-DC1(1))' (Y2(1:6)-DC1(2))' zeros(size(C,1)/2,1)];
vec_t(7:12,:)=[(X2(7:12)-DC2(1))' (Y2(7:12)-DC2(2))' zeros(size(C,1)/2,1)];
vec_c=[C(:,4)-C(:,1) C(:,5)-C(:,2) zeros(size(C,1),1)];
alpha_azi=zeros(1,size(C,1));
a_t=(vec_t(:,2)./vec_t(:,1))';
a_c=(vec_c(:,2)./vec_c(:,1))';
x=(C(:,2)'-Y2+a_t.*X2-a_c.*C(:,1)')./(a_t-a_c);
N=(x-C(:,1)')./(C(:,4)-C(:,1))';
for i=1:size(C,1)
    %calculate angle between two vectors
    alpha_azi(i)=atan2d(norm(cross(vec_t(i,:),vec_c(i,:))),dot(vec_t(i,:),vec_c(i,:)));
    if N(i)>0
        alpha_azi(i)=360-alpha_azi(i);
    end
end
arc_azi=alpha_azi./TR;

%RZ coordinates of start of completion interval
dRtot(1:6)=sqrt((X2(1:6)-DC1(1)).^2+(Y2(1:6)-DC1(2)).^2)+arc_azi(1:6);
dRtot(7:12)=sqrt((X2(7:12)-DC2(1)).^2+(Y2(7:12)-DC2(2)).^2)+arc_azi(7:12);
dZtot(1:6)=C(1:6,3)'-DC1(3);
dZtot(7:12)=C(7:12,3)'-DC2(3);

%arc length in the RZ plane and radius of curvature
arc=BUA/BUR;
ROC=(360*arc)./(2*pi*BUA);

%displacements in R and Z due to both build sections
dR=ROC+ROC.*sind(BUA-90);
dZ=ROC.*cosd(BUA-90);
for i=1:size(C,1)
    if BUA(i) < 90
        dR(i)=ROC(i)-ROC(i)*cosd(BUA(i));
        dZ(i)=ROC(i)*sind(BUA(i));
    end
end

%length of tangent section
dRtan=dRtot-dR;
dZtan=dZtot-dZ-KOP(1,3);
Ltan=sqrt(dRtan.^2+dZtan.^2);
```

```

%first build-up angle
BUA1=atand(dRtan./dZtan);

%average wellpath length
WPL=L_c+arc+Ltan+KOP(1,3);
WPL_avg=sum(WPL)/length(WPL);

%the following calculations are only for the intention of plotting the wells in 2D

%coordinates of the 1st circle of build center
R_cc1=ROC;
Z_cc1=KOP(1,3);

%amount of columns needed in the R and Z matrices
K=C(:,6)';
for i=1:size(C,1)
    if C(i,3)-C(i,6)==0
        K(i)=C(i,3)+L_c(i);
    end
end

%creating the R and Z matrices
R=zeros(size(C,1),ceil(max(K)));
Z=zeros(size(C,1),ceil(max(K)));

%R and Z coordinates of 1st and 2nd build sections and completion coordinates
Rc1e=ROC-ROC.*cosd(BUA1);
Zc1e=ROC.*sind(BUA1)+KOP(1,3);
Rc2s=Ltan.*sind(BUA1)+Rc1e;
Zc2s=Ltan.*cosd(BUA1)+Zc1e;
m1=(Rc2s-Rc1e)./(Zc2s-Zc1e);
b1=Rc1e-m1.*Zc1e;
R_cc2=dRtot;
Z_cc2=C(:,3)'-ROC;
R_ce=dRtot+(sqrt((C(:,4)-C(:,1)).^2+(C(:,5)-C(:,2)).^2))';
Rc2e=dRtot;
Zc2e=C(:,3)';
m2=(R_ce-Rc2e)./(C(:,6)'-Zc2e);
b2=Rc2e-m2.*Zc2e;

%filling the Z matrix with numbers from 1 to depth of completion end
for i=1:size(C,1)
    for j=1:C(i,6)
    A=1:C(i,6);
    Z(i,j)=A(j);
    end
end

%filling the R matrix with the corresponding coordinates
N=zeros(1,size(C,1));
for j=1:size(C,1)
for i=1:C(j,6)
    %coordinates above the 1st kickoff point
    if Z(j,i)<=KOP(1,3)
        R(j,i)=0;
    %coordinates of the 1st build section
    elseif Z(j,i)>KOP(1,3) && Z(j,i)<=Zc1e(j)
        R(j,i)=-sqrt(ROC(j)^2-(Z(j,i)-Z_cc1)^2)+R_cc1(j);
    %coordinates of the tangent section
    elseif Z(j,i)>Zc1e(j) && Z(j,i)<=Zc2s(j)
        R(j,i)=m1(j)*Z(j,i)+b1(j);
```

```
    %coordinates of the 2nd build section
    elseif Z(j,i)>Zc2s(j) && Z(j,i)<C(j,3)
        R(j,i)=-sqrt(ROC(j)^2-(Z(j,i)-Z_cc2(j))^2)+R_cc2(j);
    %coordinates of the completion interval
    else
        R(j,i)=m2(j)*Z(j,i)+b2(j);
    end

    %coordinates of a horizontal completion interval
    if C(j,3)-C(j,6)==0
        B=dRtot(j):(dRtot(j)+L_c(j));
        R(j,C(j,3)+length(B))=dRtot(j)+L_c(j);
        N(j)=C(j,3)+length(B);
                for k=1:length(B)
                    R(j,C(j,3)+k-1)=B(k);
                    Z(j,C(j,3)+k)=Z(j,C(j,3)+k-1);
                end
     end
end
end

%plotting all wells as a two-dimensional figure
% for i=1:size(C,1)
%     figure()
%     plot(R(i,1:N(i)),flipud(Z(i,1:N(i))));
%     set(gca,'XAxisLocation','top','YAxisLocation','left','ydir','reverse')
%     axis equal
%     title('Well path in the RZ-plane')
%     xlabel('R (m)')
%     ylabel('Z (m)')
%     xlim([-50 (R(i,N(i))+50)])
%     ylim([0 2600])
% end
end
```

B.10 get three DC

```
function [DC1,DC2,DC3,X_opt,Y_opt,Z_opt]=get_three_dc(C,N_dc)
%calulates the optimized coordinates of three drill centers

%number of completion intervals
N=size(C,1);
M=(1:N);

%all possible combinations of template 1
R=nchoosek(M,N/N_dc);

%remaining combinations in R_mar
R_mar=zeros(size(R,1),N-N/N_dc);
for i=1:N-N/N_dc
    for j=1:size(R,1)
    Rtemp=R(j,:);
    R2=zeros(1,N-N/N_dc);
    k=1;
    while k <= N-N/N_dc
        V=randi(N);
        if sum(Rtemp==V)==0 && sum(R2==V)==0
            R2(k)=V;
            k=k+1;
```

```
            end
        end
        R_mar(j,:)=R2;
        end
end

%all possible combinations of template 2
Mar_M=zeros(70,size(R,1)*size(R,2));
for i=1:size(R_mar,1)
        Mar_M(:,4*i-3:4*i)=nchoosek(R_mar(i,:),N/N_dc);
end

%the remaining combinations of template 3
M_gr3=zeros(70,size(R,1)*size(R,2));
  for i=1:size(R,1)
        Rtemp1=R(i,:);
        for j=1:size(M_gr3,1)
        R2=zeros(1,N/N_dc);
        Rtemp2=Mar_M(j,4*i-3:4*i);
        k=1;
        while k <= N/N_dc
            V=randi(N);
            if sum(Rtemp2==V)==0 && sum(R2==V)==0 && sum(Rtemp1==V)==0
                R2(k)=V;
                k=k+1;
            end
        end
        M_gr3(j,4*i-3:4*i)=R2;
        end
  end

%combines all 3 templates
M=size(R,1)*size(R,2)*N_dc;
Pos=zeros(size(M_gr3,1),M);
for i=1:size(R,1)
        Pos(:,12*i-3:12*i)=M_gr3(:,4*i-3:4*i);
        Pos(:,12*i-7:12*i-4)=Mar_M(:,4*i-3:4*i);
        for j=1:size(M_gr3,1)
            Pos(j,12*i-11:12*i-8)=R(i,:);
        end
end

%coordinates corresponding to values in Pos
X=zeros(size(M_gr3,1),M);
Y=zeros(size(M_gr3,1),M);
Z=zeros(size(M_gr3,1),M);
for i=1:size(M_gr3,1)
        for j=1:M
            X(i,j)=C(Pos(i,j),1);
            Y(i,j)=C(Pos(i,j),2);
            Z(i,j)=C(Pos(i,j),3);
        end
end

%drill center to every template
X_dc=zeros(size(M_gr3,1),M);
Y_dc=zeros(size(M_gr3,1),M);
for j=1:size(M_gr3,1)
        for i=1:M/(N/N_dc)
            X_dc(j,4*i-3:4*i)=sum(X(j,4*i-3:4*i))/4;
            Y_dc(j,4*i-3:4*i)=sum(Y(j,4*i-3:4*i))/4;
        end
```

```
end

%distance from drill center to every completion coordinate
dist_dc=zeros(size(M_gr3,1),M);
for j=1:size(M_gr3,1)
    for i=1:M
        dist_dc(j,i)=sqrt((X(j,i)-X_dc(j,i))^2+(Y(j,i)-Y_dc(j,i))^2);
    end
end

%average distance from completion interval to dc for every combination
dist_avg=zeros(size(M_gr3,1),size(R,1));
for j=1:size(M_gr3,1)
    for i=1:size(R,1)
        dist_avg(j,i)=sum(dist_dc(j,12*i-11:12*i))/12;
    end
end

%the combination that yields the shortest well path lengths
[min_avg_col,J]=min(dist_avg,[],1);
[min_avg_dist,I]=min(min_avg_col);
row_min=J(I);

X_opt(1,:)=X(row_min,I*12-11:I*12);
Y_opt(1,:)=Y(row_min,I*12-11:I*12);
Z_opt(1,:)=Z(row_min,I*12-11:I*12);

X_opt(2,:)=C(Pos(row_min,I*12-11:I*12),4);
Y_opt(2,:)=C(Pos(row_min,I*12-11:I*12),5);
Z_opt(2,:)=C(Pos(row_min,I*12-11:I*12),6);

%corresponding dc coordinates
DC1=[sum(X_opt(1,1:4))/4 sum(Y_opt(1,1:4))/4 0];
DC2=[sum(X_opt(1,5:8))/4 sum(Y_opt(1,5:8))/4 0];
DC3=[sum(X_opt(1,9:12))/4 sum(Y_opt(1,9:12))/4 0];
end
```

B.11 get turn three DC

```
function [X2,Y2,TR]=get_turn_three_dc(C,DC1,DC2,DC3,X_opt,Y_opt,TR)
%calculates the turn point in the XY-plane

%radius of turn
ROT=360/(2*pi*TR);

%check if distance between WH and completion start is less than 2*ROT
k=1;
while k <= (size(C,1)/3)
    if (sqrt((DC1(1)-X_opt(1,k))^2+(DC1(2)-Y_opt(1,k))^2) < 2*ROT) ||...
            (sqrt((DC2(1)-X_opt(1,k+4))^2+(DC2(2)-Y_opt(1,k+4))^2) < 2*ROT) ||...
            (sqrt((DC3(1)-X_opt(1,k+8))^2+(DC3(2)-Y_opt(1,k+8))^2) < 2*ROT)
        %increase the turn rate
        TR=TR+1/30;
        ROT=360/(2*pi*TR);
    else
        k=k+1;
    end
end

```

```
%azimuth of completion interval
azi_c=atan2d((X_opt(2,:)-X_opt(1,:)),(Y_opt(2,:)-Y_opt(1,:)));

%completion start coordinates in three new coordinate systems (a,b) with originin the drill centers
a1(1:4)=X_opt(1,1:4)-DC1(1);        b1(1:4)=Y_opt(1,1:4)-DC1(2);
a1(5:8)=X_opt(1,5:8)-DC2(1);        b1(5:8)=Y_opt(1,5:8)-DC2(2);
a1(9:12)=X_opt(1,9:12)-DC3(1);      b1(9:12)=Y_opt(1,9:12)-DC3(2);

%rotation angle needed to place the coordinates in a rotated coordinate system
rotate=180-azi_c;

%completion start coordinates in the rotated coordinate systems (xi,yi) with origin in the drill centers
xi1=a1.*cosd(rotate)+b1.*sind(rotate);
yi1=-a1.*sind(rotate)+b1.*cosd(rotate);

%circle of turn center coordinates in (xi,yi)
x_cci=xi1+ROT;
y_cci=yi1;
for i=1:size(C,1)
    if xi1(i)>0
        x_cci(i)=xi1(i)-ROT;
    end
end

%distances from drill centers to the circles of turn
H=sqrt((x_cci).^2+(y_cci).^2);
B=sqrt(H.^2-ROT^2);
tetha=(asind(ROT./H));

%coordinates of turn point in (xi,yi)
betha=zeros(1,size(C,1));
dxi=zeros(1,size(C,1));
dyi=zeros(1,size(C,1));
for i=1:size(C,1)
if x_cci(i)>0 && y_cci(i)>0
    if x_cci(i)>xi1(i)
        betha(i)=tetha(i)+atand(x_cci(i)/y_cci(i));
        dxi(i)=B(i)*sind(betha(i));
        dyi(i)=B(i)*cosd(betha(i));
    elseif x_cci(i)<xi1(i)
        betha(i)=tetha(i)+atand(y_cci(i)/x_cci(i));
        dxi(i)=B(i)*cosd(betha(i));
        dyi(i)=B(i)*sind(betha(i));
    end
elseif x_cci(i)>0 && y_cci(i)<0
    if x_cci(i)>xi1(i)
        betha(i)=tetha(i)-atand(y_cci(i)/x_cci(i));
        dxi(i)=B(i)*cosd(betha(i));
        dyi(i)=-B(i)*sind(betha(i));
    elseif x_cci(i)<xi1(i)
        betha(i)=tetha(i)-atand(x_cci(i)/y_cci(i));
        dxi(i)=B(i)*sind(betha(i));
        dyi(i)=-B(i)*cosd(betha(i));
    end
elseif x_cci(i)<0 && y_cci(i)<0
    if x_cci(i)>xi1(i)
        betha(i)=tetha(i)+atand(x_cci(i)/y_cci(i));
        dxi(i)=-B(i)*sind(betha(i));
        dyi(i)=-B(i)*cosd(betha(i));
    elseif x_cci(i)<xi1(i)
        betha(i)=tetha(i)+atand(y_cci(i)/x_cci(i));
        dxi(i)=-B(i)*cosd(betha(i));
```

```
        dyi(i)=-B(i)*sind(betha(i));
    end
elseif x_cci(i)<0 && y_cci(i)>0
    if x_cci(i)>xi1(i)
        betha(i)=tetha(i)-atand(y_cci(i)/x_cci(i));
        dxi(i)=-B(i)*cosd(betha(i));
        dyi(i)=B(i)*sind(betha(i));
    elseif x_cci(i)<xi1(i)
        betha(i)=tetha(i)-atand(x_cci(i)/y_cci(i));
        dxi(i)=-B(i)*sind(betha(i));
        dyi(i)=B(i)*cosd(betha(i));
    end
end
end

%coordinates of turn point in (a,b)
a2=dxi.*cosd(rotate)-dyi.*sind(rotate);
b2=dxi.*sind(rotate)+dyi.*cosd(rotate);

%coordinates of turn point in (X,Y)
X2(1:4)=DC1(1)+a2(1:4);     Y2(1:4)=DC1(2)+b2(1:4);
X2(5:8)=DC2(1)+a2(5:8);     Y2(5:8)=DC2(2)+b2(5:8);
X2(9:12)=DC3(1)+a2(9:12);   Y2(9:12)=DC3(2)+b2(9:12);
end
```

B.12 plot three DC

```
function WPL_avg=plot_three_dc(C,BUR,KOPz,TR)
%calculates the average well path length of all wells drilled from three drill centers

%input parameter
N_dc=3;

%compute optimized drillcenters and corresponding groups
[DC1,DC2,DC3,X_opt,Y_opt,Z_opt]=get_three_dc(C,N_dc);

%rearrange the completion intervals
C(:,1)=X_opt(1,:);     C(:,2)=Y_opt(1,:);     C(:,3)=Z_opt(1,:);
C(:,4)=X_opt(2,:);     C(:,5)=Y_opt(2,:);     C(:,6)=Z_opt(2,:);

%find turn point in the XY-plane
[X2,Y2,TR]=get_turn_three_dc(C,DC1,DC2,DC3,X_opt,Y_opt,TR);

%coordinates of the 1st kickoff points
KOP=[DC1(1) DC1(2) KOPz; DC2(1) DC2(2) KOPz; DC3(1) DC3(2) KOPz];

%completion interval length and build-up angle
[L_c,BUA]=get_BUA(C);

%arc length in the XY-plane
vec_t(1:4,:)=[(X2(1:4)-DC1(1))' (Y2(1:4)-DC1(2))' zeros(size(C,1)/3,1)];
vec_t(5:8,:)=[(X2(5:8)-DC2(1))' (Y2(5:8)-DC2(2))' zeros(size(C,1)/3,1)];
vec_t(9:12,:)=[(X2(9:12)-DC3(1))' (Y2(9:12)-DC3(2))' zeros(size(C,1)/3,1)];
vec_c=[C(:,4)-C(:,1) C(:,5)-C(:,2) zeros(size(C,1),1)];
alpha_azi=zeros(1,size(C,1));
a_t=(vec_t(:,2)./vec_t(:,1))';
a_c=(vec_c(:,2)./vec_c(:,1))';
x=(C(:,2)'-Y2+a_t.*X2-a_c.*C(:,1)')./(a_t-a_c);
N=(x-C(:,1)')./(C(:,4)-C(:,1))';
```

```
for i=1:size(C,1)
    %calculate angle between two vectors
    alpha_azi(i)=atan2d(norm(cross(vec_t(i,:),vec_c(i,:))),dot(vec_t(i,:),vec_c(i,:)));
    if N(i)>0
        alpha_azi(i)=360-alpha_azi(i);
    end
end
arc_azi=alpha_azi./TR;

%RZ coordinates of start of completion interval
dRtot(1:4)=sqrt((X2(1:4)-DC1(1)).^2+(Y2(1:4)-DC1(2)).^2)+arc_azi(1:4);
dRtot(5:8)=sqrt((X2(5:8)-DC2(1)).^2+(Y2(5:8)-DC2(2)).^2)+arc_azi(5:8);
dRtot(9:12)=sqrt((X2(9:12)-DC3(1)).^2+(Y2(9:12)-DC3(2)).^2)+arc_azi(9:12);
dZtot(1:4)=C(1:4,3)'-DC1(3);
dZtot(5:8)=C(5:8,3)'-DC2(3);
dZtot(8:12)=C(8:12,3)'-DC3(3);

%arc length in the RZ plane and radius of curvature
arc=BUA/BUR;
ROC=(360*arc)./(2*pi*BUA);

%displacements in R and Z due to both build sections
dR=ROC+ROC.*sind(BUA-90);
dZ=ROC.*cosd(BUA-90);
for i=1:size(C,1)
    if BUA(i) < 90
        dR(i)=ROC(i)-ROC(i)*cosd(BUA(i));
        dZ(i)=ROC(i)*sind(BUA(i));
    end
end

%length of tangent section
dRtan=dRtot-dR;
dZtan=dZtot-dZ-KOP(1,3);
Ltan=sqrt(dRtan.^2+dZtan.^2);

%first build-up angle
BUA1=atand(dRtan./dZtan);

%average wellpath length
WPL=L_c+arc+Ltan+KOP(1,3);
WPL_avg=sum(WPL)/length(WPL);

%the following calculations are only for the intention of plotting the wells in 2D

%coordinates of the 1st circle of build center
R_cc1=ROC;
Z_cc1=KOP(1,3);

%amount of columns needed in the R and Z matrices
K=C(:,6)';
for i=1:size(C,1)
    if C(i,3)-C(i,6)==0
        K(i)=C(i,3)+L_c(i);
    end
end

%creating the R and Z matrices
R=zeros(size(C,1),ceil(max(K)));
Z=zeros(size(C,1),ceil(max(K)));

%R and Z coordinates of 1st and 2nd build sections and completion coordinates
```

```
Rc1e=ROC-ROC.*cosd(BUA1);
Zc1e=ROC.*sind(BUA1)+KOP(1,3);
Rc2s=Ltan.*sind(BUA1)+Rc1e;
Zc2s=Ltan.*cosd(BUA1)+Zc1e;
m1=(Rc2s-Rc1e)./(Zc2s-Zc1e);
b1=Rc1e-m1.*Zc1e;
R_cc2=dRtot;
Z_cc2=C(:,3)'-ROC;
R_ce=dRtot+(sqrt((C(:,4)-C(:,1)).^2+(C(:,5)-C(:,2)).^2))';
Rc2e=dRtot;
Zc2e=C(:,3)';
m2=(R_ce-Rc2e)./(C(:,6)'-Zc2e);
b2=Rc2e-m2.*Zc2e;

%filling the Z matrix with numbers from 1 to depth of completion end
for i=1:size(C,1)
    for j=1:C(i,6)
    A=1:C(i,6);
    Z(i,j)=A(j);
    end
end

%filling the R matrix with the corresponding coordinates
N=zeros(1,size(C,1));
for j=1:size(C,1)
for i=1:C(j,6)
    %coordinates above the 1st kickoff point
    if Z(j,i)<=KOP(1,3)
        R(j,i)=0;
    %coordinates of the 1st build section
    elseif Z(j,i)>KOP(1,3) && Z(j,i)<=Zc1e(j)
        R(j,i)=-sqrt(ROC(j)^2-(Z(j,i)-Z_cc1)^2)+R_cc1(j);
    %coordinates of the tangent section
    elseif Z(j,i)>Zc1e(j) && Z(j,i)<=Zc2s(j)
        R(j,i)=m1(j)*Z(j,i)+b1(j);
    %coordinates of the 2nd build section
    elseif Z(j,i)>Zc2s(j) && Z(j,i)<C(j,3)
        R(j,i)=-sqrt(ROC(j)^2-(Z(j,i)-Z_cc2(j))^2)+R_cc2(j);
    %coordinates of the completion interval
    else
        R(j,i)=m2(j)*Z(j,i)+b2(j);
    end

    %coordinates of a horizontal completion interval
    if C(j,3)-C(j,6)==0
        B=dRtot(j):(dRtot(j)+L_c(j));
        R(j,C(j,3)+length(B))=dRtot(j)+L_c(j);
        N(j)=C(j,3)+length(B);
                for k=1:length(B)
                    R(j,C(j,3)+k-1)=B(k);
                    Z(j,C(j,3)+k)=Z(j,C(j,3)+k-1);
                end
     end
end
end

%plotting all wells as a two-dimensional figure
% for i=1:size(C,1)
%     figure()
%     plot(R(i,1:N(i)),flipud(Z(i,1:N(i))));
%     set(gca,'XAxisLocation','top','YAxisLocation','left','ydir','reverse')
%     axis equal
```

```
%      title('Well path in the RZ-plane')
%      xlabel('R (m)')
%      ylabel('Z (m)')
%      xlim([-50 (R(i,N(i))+50)])
%      ylim([0 2600])
% end
end
```

NTNU