



An Automatic Well Planner for Complex Well Trajectories

Brage S. Kristoffersen¹  · Mathias C. Bellout¹ ·
Thiago L. Silva¹ · Carl F. Berg¹

Received: 23 December 2020 / Accepted: 18 May 2021
© The Author(s) 2021

Abstract A data-driven automatic well planner procedure is implemented to develop complex well trajectories by efficiently adapting to near-well reservoir properties and geometry. The procedure draws inspiration from geosteering drilling operations, where modern logging-while-drilling tools enable the adjustment of well trajectories during drilling. Analogously, the proposed procedure develops well trajectories based on a selected geology-based fitness measure using an artificial neural network as the decision maker in a virtual sequential drilling process within a reservoir model. While neural networks have seen extensive use in other areas of reservoir management, to the best of our knowledge, this work is the first to apply neural networks on well trajectory design within reservoir models. Importantly, both the input data generation used to train the network and the actual trajectory design operations conducted by the trained network are efficient calculations, since these rely solely on geometric and initial properties of the reservoir, and thus do not require additional simulations. Therefore, the main advantage over traditional methods is the highly articulated well trajectories adapted to reservoir properties using a low-order well representation. Well trajectories generated in a realistic reservoir by the automatic well planner are qualitatively and quantitatively compared to trajectories generated by a differential evolution algorithm.

✉ Brage S. Kristoffersen
brage.s.kristoffersen@ntnu.no

Mathias C. Bellout
mathias.bellout@ntnu.no

Thiago L. Silva
thiago.l.silva@ntnu.no

Carl F. Berg
carl.f.berg@ntnu.no

¹ Department of Geoscience and Petroleum, NTNU, Trondheim, Norway

Results show that the resulting trajectories improve productivity compared to straight line well trajectories, both for channelized and geometrically complex reservoirs. Moreover, the overall productivity with the resulting trajectories is comparable to well solutions obtained using differential evolution, but at a much lower computational cost.

Keywords Well Parameterization · Machine learning · Well trajectory optimization · Reservoir model · Numerical simulation · Artificial neural networks

1 Introduction

Subsurface reservoir development involves decisions regarding where to drill wells, drilling sequence and completion, and how to control existing production. Well representations implemented within reservoir models range from relatively simple parameterizations for vertical wells, to extended parameterizations for deviated wells that may have curved trajectories and complex completion designs. Advanced well representations are often required to honor important realistic geological features, such as undulating top reservoirs, faults or layers of good reservoir quality. When applied within optimization procedures, such realistic representations generally demand additional variables and sophisticated constraint handling. This increases problem dimension and complexity. In this work, a data-driven automatic well planner (AWP) procedure is proposed for effective design of well trajectories within field development decision-making.

Decisions concerning where to drill wells can be grouped into either the planning phase or the drilling phase. Information about the subsurface during the planning phase is very limited, relying mostly on indirect measurements obtained far from the planned path of the well. Decisions during the well planning phase are therefore characterized by high uncertainty. Decisions during the drilling phase, on the other hand, rely on information acquired during drilling; from readily available drilling parameters such as torque and weight on bit, to less available traditional logging-while-drilling measurements, and sophisticated logging tools embedded in the drill string that are able to acquire information from deeper in the formation.

The planning phase involves making decisions regarding the placement and trajectories of the wells using simulation models with considerable uncertainty. To some extent, the devised plans serve as guidelines for the subsequent drilling phase, where the well trajectories are often adjusted based on real-time information acquired through measure-while-drilling technology. This decision-making procedure is called geosteering, and it allows operators to change the direction of the well based on real-time measurements (Lesso Jr et al. 1996). Consequently, the drilling operation adapts in a reactive manner to the surroundings based on continuous processing and incorporation of measurements related to the current position of the drill bit. This technology enables wells to be placed inside thin oil zones or complex geological structures (Li et al. 2005). Several works present procedures that try to automate or assist the driller during the geosteering operation (Maus et al. 2020; Arbus and Wilson 2019; Winkler 2017).

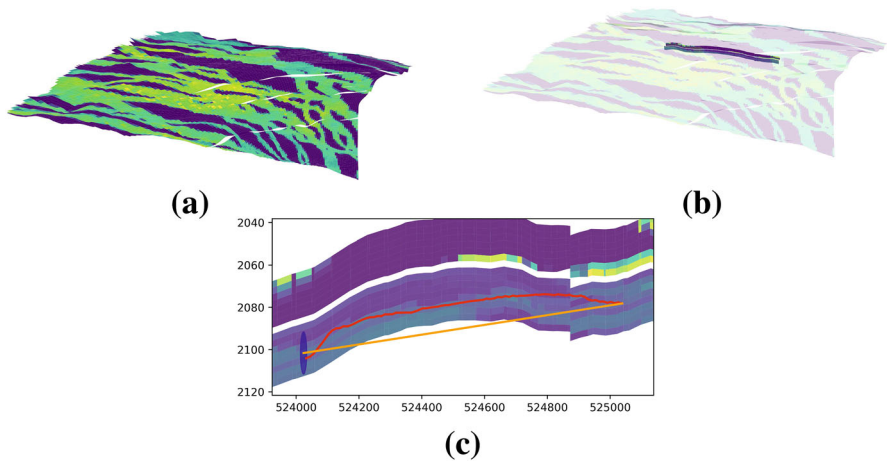


Fig. 1 **a** Distribution of a property, e.g., permeability or porosity, within a reservoir model. Bright colors indicate a good property, while dark colors indicate a poor property. **b** A cross section defined by the toe and heel for a proposed well path. **c** Examples of different well trajectories within the given cross section

The AWP procedure proposed in this work couples virtual geosteering with reservoir simulation models to provide for complex well trajectory design with low-order parameterization. Thus, geosteering concepts from the drilling phase are integrated within the well planning phase to improve the efficiency of field development and reservoir management operations. The procedure involves a machine learning application inspired by geosteering operations to efficiently adapt well trajectories to near-well reservoir properties and geometry while honoring engineering constraints. The concept is illustrated in Fig. 1, where Fig. 1a presents a property within a reservoir model, which we assume is available. This could be a static property such as permeability or porosity, a dynamic property such as oil saturation, or a combination of properties. The challenge is how to design well trajectories that adapt to the chosen property. To reduce the complexity, we consider only well paths in the cross section defined by the set of heel and toe coordinates for the proposed well. Such a cross section is shown in Fig. 1b. A simple straight-line representation might yield a well trajectory that is placed poorly with respect to both the geometry of the reservoir and the reservoir property in question. This is illustrated by the yellow line in Fig. 1c, where a large part of the well lies outside the reservoir, and the well also traverses low values of the property field. The proposed AWP aims to generate trajectories that traverse high values of the chosen property. An example is given by the red line in Fig. 1c, where the well remains within the reservoir and primarily traverses regions with high values of the property field.

Our AWP is based on machine learning. Machine learning methods are increasingly used for their predictive capability relying on processing of available data. Artificial neural networks (ANN), in particular, have been widely applied to treat tasks such as speech recognition (Mikolov et al. 2011), natural language processing (Collobert et al. 2011; Jean et al. 2014), and image recognition (Tompson et al. 2014; Szegedy et al. 2015). Oil and gas industry operations involve large amounts of data and data-

processing needs, as the industry relies on both direct and indirect measurements to model the past, current, and future state of an asset. Machine learning applications within oil and gas range from early stage characterization of reservoirs (Alizadeh et al. 2012), matching production history with simulation output (Shahkarami et al. 2014), and proxy-modeling to forecast the reservoir performance (Haghshenas et al. 2020). Despite the broad use of ANNs in oil and gas industry applications, to the best of our knowledge, the AWP procedure is the first to use ANNs for well trajectory design within reservoir simulation models.

The ANN driving the AWP procedure gathers local information around the current well head position and outputs the direction of the next section of well trajectory. While the training of the network has a significant computational cost, the time it takes to generate the individual well trajectories is negligible when compared to reservoir simulation runtimes. The neural network can be trained using a range of significantly different reservoir property data to allow the AWP to recognize subtle trends in the simulation model and suggest well trajectory changes accordingly. Fitness functions and required constraints are imposed during the training. These performance measures ensure both that the AWP trajectories adapt to the surrounding geology according to the specified fitness and drilling criteria, and that the resulting trajectories do not stray from the general well path proposed by the operator.

Importantly, while AWP generates trajectories that closely adapt to realistic geological heterogeneities, the procedure enables a low-level representation of these complex trajectories. Specifically, the data-driven approach enabling this well parameterization takes as input the heel and toe values of a well and yields, according to the specified fitness target, trajectories that honor the underlying geological surroundings. This capability simplifies and can help decrease the computational load required by field development operations, e.g., well completion design and well placement, especially if these operations involve iterative procedures and extensive reservoir simulation. A particular application is towards the heavy computational cost associated with using derivative-free algorithms, e.g., for well placement search (Bellout et al. 2012; Kristoffersen et al. 2020), since sampling requirements for these type of algorithms, and therefore overall performance, are typically proportional to the number of variables.

This paper is structured as follows: Sect. 2 presents an overview of the AWP methodology. Section 3 is an introduction to the reservoir model used in this work and an analysis of how AWP performs against a traditional approach on static data. Section 4 presents the problem formulation and integration of AWP in a numerical model, where AWP trajectories are compared with a straight-line well description in terms of productivity. Section 5 contains a discussion of the work. Finally, conclusions are presented in Sect. 6.

2 Methodology

This section presents the main components and operations of the AWP methodology. Though inter-related, the AWP components and operations are first presented sequentially as separate topics. At the end of the section, the functions and characteristics of the individual parts are discussed in the context of the overall procedure. The goal

of this presentation structure is to make it easier for the reader to understand the core parts of the AWP procedure, and then its overall functionality.

2.1 AWP Components

In essence, AWP is a method to automatically define a well path within the spatial grid of a simulation model. The main input to the AWP methodology is a single-line parameterization given by the heel and toe end points of the well. Additionally, the AWP methodology requires knowledge about both the spatial geometry and the grid properties in the reservoir simulation model. A property field obtained from the reservoir grid model data is a representation of the geological structure where the well will be drilled. When a property field and a heel-toe description are provided, the AWP methodology can create a new trajectory based on decisions made by an ANN. These three main components of the AWP methodology, i.e., well parameterization, the reservoir property field, and the ANN, are described below.

2.1.1 Well Parameterization

Well parameterization refers to how wells are specified as variables within an encompassing routine. In this work, wells are specified by the location of their heel $\mathbf{x}_h = (x_h, y_h, z_h)$ and toe $\mathbf{x}_t = (x_t, y_t, z_t)$. These are the start and end points of the perforated part of the well, i.e., the part of the well that interacts with the reservoir. Thus, a well is defined by a vector of the two positions, $(\mathbf{x}_h, \mathbf{x}_t)$, where $\mathbf{x} \in \mathbb{R}^3$, similar to the well parameterization found in (Sayyafzadeh and Alrashdi 2019). Note that the coordinate for the toe \mathbf{x}_t can be replaced by (θ, ϕ, l) , where the elevation θ and azimuth ϕ are the two angles giving the direction of the well, while l denotes its length. This type of parameterization is used in (Forouzanfar et al. 2012; Hassani et al. 2011; Bouzarkouna et al. 2011).

Given the heel and toe, the simplest well trajectory is a straight line (hereinafter abbreviated “SL”) between the two end points. However, such an SL representation is unlikely to yield optimal production, since it does not discriminate between or take into account the layout of surrounding reservoir properties. Specifically, such a simple representation disregards detailed model property data contained in the grid model. Herein, such model property data, especially pertaining to the near-wellbore area, are the type of data typically obtained by logging-while-drilling procedures during drilling.

In this work, all wells are either defined by the two end points of the perforated interval $(\mathbf{x}_h, \mathbf{x}_t)$, or given as refined representations defined by a set of connected straight lines. Crucially, even though the AWP produces refined well path representations consisting of a set of connected segments, the variables treated at a well trajectory design level are only the heel and toe end point coordinates.

2.1.2 Reservoir Property Field

Reservoir properties of importance for fluid flow can be divided into two groups: static and dynamic properties. Static properties include petrophysical data such as porosity and permeability, as well as static multiphase flow parameters such as relative permeability and capillary pressure curves. Dynamic properties typically involve data such as saturation and pressure.

Reservoir properties need to be defined for each point corresponding to the spatial discretization of the reservoir. The property field f of a real-valued property, e.g., permeability, is represented by a map $f: \Omega \rightarrow \mathbb{R}$ from the volume of the reservoir Ω to real values \mathbb{R} . Thus, for every point $\mathbf{x} \in \Omega$, a property $f(\mathbf{x})$ is defined. As an example, the permeability gives a property field f , where $f(\mathbf{x}) = k(\mathbf{x})$ for all points $\mathbf{x} \in \Omega$ inside the reservoir Ω .

The spatial grid is a representation of the geometry of the reservoir, and typically captures the main structural features in the reservoir, such as faults and geological layering. For a grid cell identified by an index (i, j, k) , there is a single property defined for the full volume inside this cell. Therefore, different property fields may be associated with different grid representations of the reservoir.

For dynamic reservoir simulations, the models are usually coarsened versions of geological models of the same reservoir. $f(\mathbf{x})$ will denote the corresponding property function for the cell containing the point \mathbf{x} . In the next subsection we will discuss how the reservoir property field data are treated by the neural network component of the AWP procedure.

2.1.3 Artificial Neural Network

A main component of the AWP methodology is an artificial neural network. This subsection describes the type and qualities of the ANN used in the AWP methodology. The training of the network is described in the following section.

The ANN serves as decision maker within the AWP procedure. While other types of ANNs can also be applied, in this work the network is generated through an unsupervised reinforcement learning scheme based on the NeuroEvolution of Augmenting Topologies (NEAT) (Stanley and Miikkulainen 2002; Stanley 2004). The reason for using NEAT is that the framework is capable of designing a minimalistic ANN through a process mimicking natural selection, in which several of the ANNs compete against each other for survival. The simplistic design of ANNs created by NEAT is expected to produce robust networks with limited data sets compared to other alternatives such as supervised learning, which require substantially more data.

In NEAT, a population of networks is evolved over multiple generations through a multi-speciated genetic algorithm as outlined in the flowchart in Fig. 2. The initial population consists of equal networks with only one hidden node and zero connections, as illustrated by the zeroth generation in Fig. 3. Each member of the population in the genetic algorithm is an ANN, and its performance is ranked based on a fitness value obtained from a predefined fitness function.

The algorithm depicted in Fig. 2 starts by generating mutations or performing crossovers between the parents, i.e., sets of members. The parents chosen for mutation

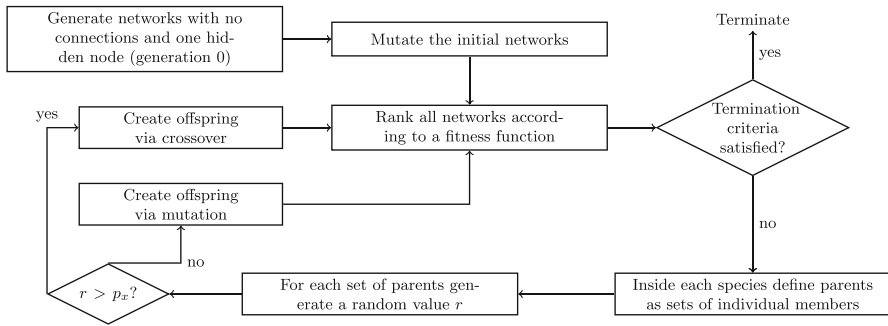


Fig. 2 Flowchart presenting the general implementation of the genetic algorithm within the NEAT framework

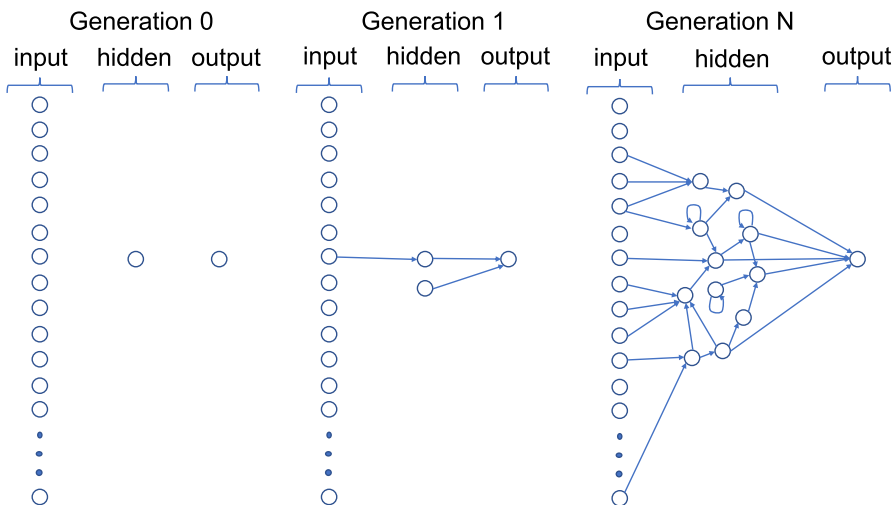


Fig. 3 Illustration of the topological evolution of the best network using NeuroEvolution of Augmenting Topologies. The circles represent nodes, while blue arrows represent the connection between the nodes and their direction

or crossover are selected stochastically from a prescribed crossover probability p_x : for each set of parents we generate a random number $r \in [0, 1]$; if $r < p_x$, the parents get an offspring through crossover, while if $r > p_x$, the parents get an offspring from mutation. The members of the new generation are ranked based on their fitness function. Members that display novel features are protected by speciation. For the remaining members, the highest ranked are kept for further breeding, while the lowest ranked are terminated.

Mutation and speciation should ensure that there is sufficient genetic diversity within the population. This evolutionary approach allows modification of the topology of the network by adding or removing nodes and connections. The algorithm is furthermore allowed to change weights, biases, activation function, nodes, and connections.

In this decision-making, the reservoir properties in the vicinity of the current position serve as input to the network, while the direction for the next well section is the output. While the training process could be computationally costly, applying the ANN is computationally light. This makes the introduction of an ANN within an optimization workflow a reasonable strategy to reduce both problem complexity and computational load. This is especially true when using search routines that require a high number of function evaluations.

2.2 AWP Operations

A well, as parameterized by its heel and toe coordinates ($\mathbf{x}_h, \mathbf{x}_t$), is provided to the AWP procedure to develop a trajectory according to a specified fitness measure. The fitness measure for the AWP procedure used in this work is to traverse the most permeable reservoir sands, though other fitness measures can be used. This section describes main AWP operations: decision-making, objective formulation, constraint handling, and the training of the network.

2.2.1 Decision-Making

The underlying idea for AWP decision-making is to mimic geosteering operations. Modern logging tools integrated in the drill string enable the real-time gathering of information regarding local geology and fluid content around the drill bit. Based on this information, drillers can make informed decisions of whether to alter the current trajectory of the well. Such decisions are made according to different objectives, for example improving the productivity of the well or placing the well in a thin oil layer between an aquifer and a gas cap. The drilling decisions happen sequentially based on the current local information, while previous decisions are irreversible because that well segment has already been drilled.

The AWP procedure uses the same operational principles as a geosteering process, i.e., the AWP procedure adapts the trajectory of the well based on near-wellbore information, in this case obtained from a simulated environment. As mentioned, in this article only the permeability field $k(\mathbf{x})$ is used as the property field. However, any other property fields involving different data types can be applied. The target properties for well trajectory adjustment is a decision for the reservoir/drilling engineer. Ultimately, this decision is dependent on the purpose of the well in the context of the overall drainage strategy.

In our implementation, the AWP makes consecutive decisions with a predefined step length. Artificially, the amount of information made available to the method is limited by distance (corresponding to a look-ahead of a predefined distance). Both short- and long-distance information is provided as input to the network, thus driving the forward propagation of the well trajectory. The short-distance information is concerned with going directly through zones that have immediate potential according to the current local information. The long-term information is concerned with reaching the target (the well toe), and ensuring that the current decisions will not be damaging to later stages of the drilling process, e.g., that the overall trajectory is still feasible to drill.

The ANN training procedure determines the appropriate influence between these two types of information sources.

2.2.2 Trajectory Objectives

As noted in Sect. 2.1.3, the AWP procedure is dependent on a ranking of the networks, where the ranking is given by a fitness function. The AWP can be represented by a mapping

$$N: (\mathbf{x}_h, \mathbf{x}_t) \rightarrow \{\mathbf{x}_h, \mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_{n-1}, \mathbf{x}_n\}, \quad (1)$$

where the well trajectory generated by the AWP gives the well segments defined by $\{\mathbf{x}_i, \mathbf{x}_{i+1}\}$. Note that the last point, while in the vicinity of the toe \mathbf{x}_t , is not necessarily in exactly the same position, and thus $\mathbf{x}_n \neq \mathbf{x}_t$. This will be elaborated on later.

Given a property field $f(\mathbf{x})$, a simple fitness function g_f can be defined by the sum

$$g_f(N(\mathbf{x}_h, \mathbf{x}_t)) = \frac{\sum_{i=0}^n \int_{\mathbf{x}_i}^{\mathbf{x}_{i+1}} f(\mathbf{x}) d\mathbf{x}}{\sum_{i=0}^n \|\mathbf{x}_{i+1} - \mathbf{x}_i\|}. \quad (2)$$

If, for example, the property field is the permeability field, $f(\mathbf{x}) = k(\mathbf{x})$, then this fitness function is just an average of the permeability values along the well path.

Assume we have a predefined length l for the straight sections of our well trajectory. Then a three-dimensional well trajectory can be defined by a set of elevation angles $\Theta = \{\theta_i\}$ and azimuth angles $\Phi = \{\phi_i\}$, together with an initial starting point $\mathbf{x}_0 = \mathbf{x}_h$. The points along the well trajectory, at a given iteration $i + 1$, are then

$$\begin{aligned} x_{i+1} &= x_i + l \sin(\theta_i) \cos(\phi_i), \\ y_{i+1} &= y_i + l \sin(\theta_i) \sin(\phi_i), \\ z_{i+1} &= z_i + l \cos(\theta_i). \end{aligned} \quad (3)$$

As previously mentioned, in this work we are applying the AWP in two-dimensional cross sections only; thus we have a constant azimuth, $\phi_i = \phi_a$, for each well. The coordinate \mathbf{x}_{i+1} is then determined by the previous spatial coordinate \mathbf{x}_i and the elevation angle θ_i .

The input into the ANN is the set of local properties around the current location \mathbf{x}_i , the current position and angles, in addition to the toe coordinate \mathbf{x}_h . The output consists of two angles $\{\theta_{i+1}, \phi_{i+1}\}$ that provide the direction of the next coordinate point. Starting at the heel \mathbf{x}_i , and with a fixed step length l , the ANN is thus applied repeatedly to produce a set of spatial points defining a well path.

Consider a given AWP network N , as formulated in Eq. (1). When ranking our ANN as indicated in Fig. 2, a fitness function would be to apply Eq. (2) for the well paths over a set of starting points $\{(\mathbf{x}_h, \mathbf{x}_t)\}$

$$\sum_{\{(\mathbf{x}_h, \mathbf{x}_t)\}} g_f(N(\mathbf{x}_h, \mathbf{x}_t)). \quad (4)$$

Here, $g_f(N(\mathbf{x}_h, \mathbf{x}_t))$ is as given by Eq. (1).

When using the fitness function given in Eq. (4), the trained network yields trajectories traversing parts of the reservoir with high values for the property field $f(\mathbf{x})$. To ensure that the resulting trajectory lies between the given heel and toe coordinates, we introduce a second fitness term $b_t(\mathbf{x}_n)$. This additional term is proportional to the distance between the last point \mathbf{x}_n of the developing trajectory $N(\mathbf{x}_h, \mathbf{x}_t)$, and the target toe \mathbf{x}_t coordinate. As such, this term serves as a “reward” measure for the developing trajectory, and is defined as

$$b_t(\mathbf{x}_n) = \begin{cases} R_g & ||\mathbf{x}_t - \mathbf{x}_n|| \leq t_g \\ (\boldsymbol{\alpha} \cdot (\mathbf{x}_t - \mathbf{x}_n))^2 & ||\mathbf{x}_t - \mathbf{x}_n|| > t_g. \end{cases} \quad (5)$$

The first condition in Eq. (5) applies to the case when the distance to the toe is less than t_g , which yields a positive contribution of R_g to the fitness. R_g is a constant value equal to many times the expected value of Eq. (4). Accordingly, R_g serves as a discriminator between those networks that do succeed in reaching close to the toe and those that do not. If the first condition is not met, the contribution to the fitness is set equal to the square of the distance to the toe (here $\boldsymbol{\alpha}$ is a vector that provides weights for the different directions).

The final fitness function is given as

$$\sum_{\{\mathbf{x}_h, \mathbf{x}_t\}} g_f(N(\mathbf{x}_h, \mathbf{x}_t)) + b_t(\mathbf{x}_n), \quad (6)$$

where \mathbf{x}_n is the last point in the output of $N(\mathbf{x}_h, \mathbf{x}_t)$.

2.2.3 Constraint-Handling

The specified fitness function ensures that a generated network produces well trajectories that span the distance from the heel to (the vicinity of) the toe. In this subsection, three geometric constraints are introduced to impose further realism on the trajectories produced by the network.

Two of these constraints are hard constraints that cause a particular adjustment to be unsuccessful, while the third constraint is a simple bound that cannot be violated. In this context, the measure of a successful run uses the first condition of Eq. (5), i.e., if $||\mathbf{x}_t - \mathbf{x}_n|| \leq t_g$, then the run is a success. When in application mode, if a run is deemed unsuccessful, then the constraint handler will simply return those steps that have been made, in addition to alterations of the trajectory that will adhere to the geometrical constraints. Importantly, after a constraint has been violated, properties and features will be ignored in favor of picking steps that produce a trajectory whose last point is close to the specified toe coordinate. Finally, if remedying the situation by this measure is impossible, the constraint handler will revert to returning the straight line.

The first geometric constraint imposed on the production of trajectories limits the possible deviation of the trajectory, i.e.,

$$\frac{\|(\mathbf{x}_i - \mathbf{x}_h) \times (\mathbf{x}_t - \mathbf{x}_h)\|}{\|\mathbf{x}_t - \mathbf{x}_h\|} < t_d \quad \forall \mathbf{x}_i \in N(\mathbf{x}_h, \mathbf{x}_t), \quad (7)$$

where t_d is the maximum distance allowed from the heel-toe line.

The second constraint is implemented to avoid unnecessary oscillations. The constraint is applied to the number of well segments $\|N(\mathbf{x}_h, \mathbf{x}_t)\|$ as

$$\|N(\mathbf{x}_h, \mathbf{x}_t)\| < \frac{\|\mathbf{x}_t - \mathbf{x}_h\|}{l} \beta, \quad (8)$$

where β is a predetermined total length coefficient ($\beta = 1.1$ in this work). Maintaining a low number of well segments reduces the flexibility of the trajectory, and thus restricts the neural network in artificially increasing the fitness by creating, for example, unreasonable bending of trajectories within good reservoir sands.

A common restriction in drilling operations is the dogleg severity (DLS), which is a measure of how many degrees of elevation and azimuth the trajectory can change per drill-string length. The DLS constraint is a constraint implemented as bounds limiting the deviation in elevation (θ) and azimuth (ϕ) angles between two trajectory points $i - 1$ and i , i.e.,

$$\|\theta_i - \theta_{i-1}\| + \|\phi_i - \phi_{i-1}\| < \gamma l, \quad (9)$$

and γ is a coefficient determining the DLS magnitude.

2.3 AWP Implementation

The AWP implementation comprises the various components, operations, and constraints defined up to now. The overall implementation is outlined in Fig. 4. First, we initialize the drilling operation at the heel location. We then collect information from a predefined radius d_m . This information, along with geometric relations, is passed through the ANN, which then decides whether to increase or decrease the elevation. If any constraints are violated or the target (toe) is reached, the well trajectory is returned to the encompassing procedure. If not, the procedure is repeated until either the procedure has reached the target, or a constraint is violated. The operational sequence, as well as any interdependency and interaction between the various operations and components, is discussed next. Some relevant concepts regarding unsupervised reinforcement learning of neural networks are also discussed.

For computational efficiency, the AWP is trained exclusively on 2D cases. In all cases, the azimuth is kept constant and equal to the angle given by the heel and toe; thus

$$\phi_i = \phi_a = \arctan\left(\frac{x_t - x_h}{y_t - y_h}\right), \quad (10)$$

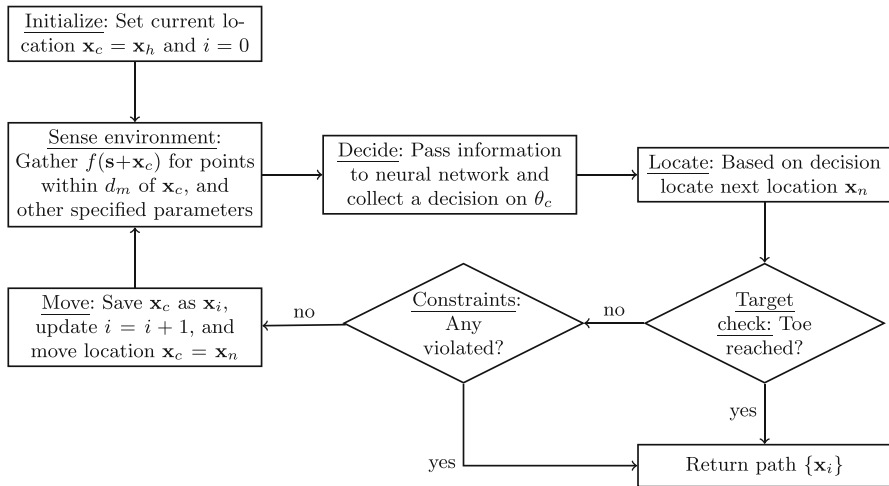


Fig. 4 Flowchart outlining the series of operations and constraint checks performed by the AWP procedure to obtain the well path trajectory

where ϕ_i is the azimuth at the individual AWP decisions.

In this work, we distribute the points radially using a maximum distance d_m . The number of sampling points along a particular direction is given by n_d . We also use a maximum elevation, γ_m , with the number of sampling points in this dimension being n_γ . This gives two sets of parameters, one of all the lengths $\mathcal{L} = \left\{ \frac{d_m}{n_d}, \frac{2d_m}{n_d}, \dots, d_m \right\}$, and one of all the elevations $\Psi = \left\{ 0, \pm \frac{\gamma_m}{n_\gamma}, \pm \frac{2\gamma_m}{n_\gamma}, \dots, \pm \gamma_m \right\}$. The combination of these two sets of variables creates a set of vectors $\mathbf{s} = (x_s, y_s, z_s)$ given as

$$\begin{aligned} x_s &= d \sin(\psi) \cos(\phi_a) \\ y_s &= d \sin(\psi) \sin(\phi_a) \\ z_s &= d \cos(\psi) \quad \forall \quad d \in \mathcal{L}, \psi \in \Psi. \end{aligned} \quad (11)$$

If $\gamma_m = \pi/2$, then the set of coordinates $\{\mathbf{s}\}$ spans the half sphere of radius d_m in front of the current position. When \mathbf{x}_i is the current position and $\gamma_m = \pi/2$, then the sample points $\{\mathbf{x}_i + \mathbf{s}\}$ are as illustrated in Fig. 5a. At each sampling point we collect the value of the property field f , so that a set of values $\{f(\mathbf{x}_i + \mathbf{s})\}$ is passed as input to the network. In our implementation, we use a constant step length l determined by the approximate length of one drill pipe. The only decision parameter out of the neural network is the continuous elevation angle θ_i , as indicated in Fig. 5. The cyan bars at the current location in Fig. 5 indicate the bound constraint with regard to the DLS, as given by Eq. (9). As indicated in the flowchart of the AWP process in Fig. 4, whenever the DLS constraint is violated, the iterative loop is stopped and the AWP returns a straight line as the obtained well trajectory. The flowchart in Fig. 4 includes the full iterative loop for the AWP procedure. The figure also includes the stopping

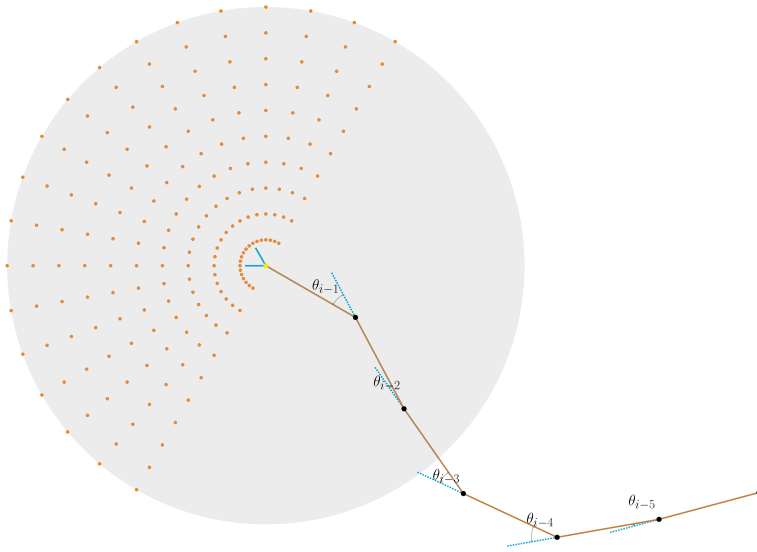


Fig. 5 Description of the AWP procedure. The well trajectory is built in steps of equal length l , generated by the AWP procedure. The set of sampling points for the next decision are indicated by orange circles, distributed within the grey information horizon. Dogleg severity bounds for the next elevation θ_i are indicated by cyan-colored bars from the current location. The change in elevation at the previous decision is shown as the angle between the brown and dotted cyan line

criterion (shown in the lower right box) indicating that the procedure stops when the well path has progressed close enough to the toe.

2.4 General Training and Application

As discussed, the AWP procedure operates on a straight-line well path to produce a trajectory better adapted to the local distribution of some fitness measure, in our case the average permeability along the well path. The well path end points provided to the AWP procedure, i.e., the heel and toe, may come from any encompassing procedure.

In this method, it is important to distinguish between the two main stages of operation. The first stage always consists of the training process as outlined by the flowchart in Fig. 2. At this stage, the ANN(s) are exposed to the training data, which in our setting are a set of reservoir models.

As with all neural networks, we face the problem of over-fitting, i.e., the situation when the network remembers the correspondence between the input and optimal solution instead of learning *how* to find the solution. To make the neural network general enough to solve for new circumstances, the well coordinates in the training data have to contain hundreds of different scenarios.

To maintain generality in the neural networks, each member of the genetic algorithm was evaluated through all the training data and given an aggregate fitness. For each member of the NEAT algorithm, the aggregate fitness for the member is given by summing the fitness function in Eq. (6) for all the training data.

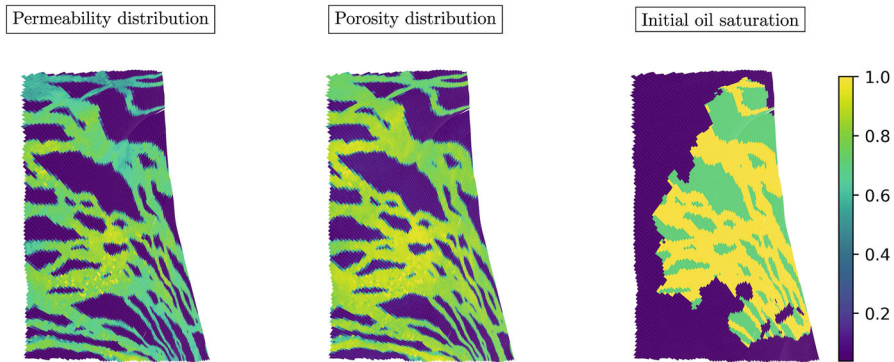


Fig. 6 Permeability, porosity, and initial oil saturation of the upper layer of realization 25 in the Olympus ensemble. All values are normalized from 0 to 1

3 Analysis

This section tests the performance of the AWP for a fitness-function-driven development of new trajectories while honoring multiple geometric constraints, as outlined in the previous section. Five AWP trajectory solutions are highlighted to demonstrate the capability of the methodology to adapt to important geometric features of the reservoir. The Olympus reservoir model used for these tests is presented first, and then the model setup for the ANN is described.

3.1 Reservoir Model

The AWP procedure is applied to a case built from the Olympus reservoir model, which is a synthetic reservoir created to resemble features found at the Norwegian Continental Shelf (Fonseca et al. 2018). The model consists of a highly channelized reservoir formation, as seen in Fig. 6, with high permeability contrasts between the channels and the surrounding geology, and a lower, more homogeneous formation. The Olympus benchmark case originally consists of an ensemble of 50 realizations that represent the geological uncertainty of the model. For the purpose of our testing, we have chosen two of these realizations, one for neural network training (realization 1) and one for the dynamic test (realization 25).

Permeability, porosity, and initial oil saturation for realization 25 of the Olympus ensemble are shown in Fig. 6. The original Olympus benchmark grid consists of $118 \times 181 \times 16$ cells, of which 192,750 are active. The analysis performed in this section uses the full Olympus model grid, while the dynamic validation test performed in Sect. 4 uses the upper channelized formation (giving a reservoir section consisting of $118 \times 181 \times 6$ cells only). The inactive cells are omitted when applying the AWP.

Table 1 The final configuration for the network utilized in Sects. 3 and 4. The number of training coordinates and approximate training time are also given

Parameter	Value
Training time (wall time for a single core)	18 h
Number of hidden nodes	87
Number of connections	235

Table 2 The parameters applied to both case studies

Parameter	Name	Value	Unit
α_x	Horizontal weighing	0.20	
α_z	Vertical weighing	0.80	
β	Length coefficient	1.1	
γ	DLS coefficient	0.1	°/m
L	Step length	12	m
d_m	Max distance	100	m
n_d	Distance resolution	10	
γ_m	Max angle	110	°
n_γ	Angular resolution	10	
t_g	Goal tolerance	10	m
t_d	Distance tolerance	20	m

3.2 Automatic Well Planner

The training data for the AWP neural network involves generating a random set of trajectories applied to the Olympus realization chosen for training. The ultimate goal of the training process is to generate a network that is able to produce trajectories with high fitness function values while honoring the constraints. The training set consisted of 1,000 arbitrary heel and toe pairs (\mathbf{x}_h , \mathbf{x}_t). The only limitations applied to these coordinate pairs were a regional and a length restriction. The regional restriction makes sure that both \mathbf{x}_h and \mathbf{x}_t lie within the reservoir, while the length restriction enforces the internal distance $\|\mathbf{x}_t - \mathbf{x}_h\| \leq 3,000$ m. Table 1 lists training time and the size of the final ANN with the number of hidden nodes and connections. Although the training is time-consuming, the resulting ANN is small, and therefore applying the network is fast. Constraint parameters used in the training of the AWP procedure are given in Table 2.

For comparison, the problem of generating trajectories with high fitness values was also solved using a differential evolution (DE) algorithm (Price et al. 2005). DE is a global optimization algorithm that has seen extensive use within several engineering applications (Chakraborty 2008). Throughout this paper, whenever DE is involved, it is subjected to the same constraints as the AWP. To find the best possible solution, DE is provided with information for the entire cross section. DE solutions are expected to produce higher fitness function values because they rely on information from the full cross section, not just the well vicinity.

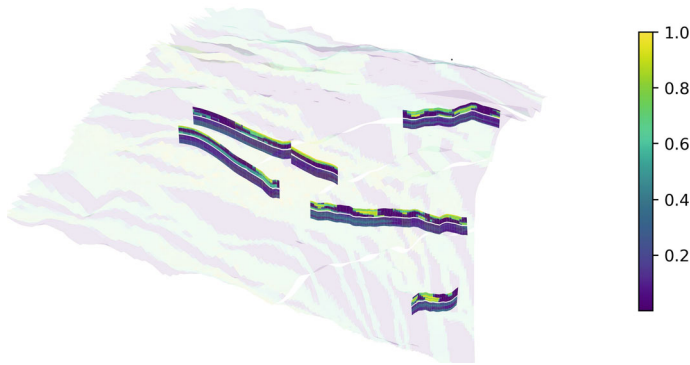


Fig. 7 Location of cross sections for the five example cases, illustrated on the permeability field of the reservoir; all values are normalized from 0 to 1

3.3 Test Results

This section presents the results regarding the efficiency of the AWP procedure to develop a fitness-increasing trajectory.

3.3.1 Five Cross Sections

In this case study, we consider five example cases. These examples highlight the behavior of the AWP method compared to solutions provided by the DE and SL. Each cross section represents one aspect of the AWP procedure that we want to highlight. The positions inside the reservoir grid of the cross sections used to develop the five example trajectories are shown in Fig. 7.

Figure 8 shows the five example cross sections. Note that the models have been stretched vertically, and therefore the circular target appears as an ellipse.

DE should yield better-adapted well trajectories compared to the “short-sighted” AWP procedure. DE solutions are therefore viewed as best-case solutions for comparison with the AWP procedure.

The first cross section in Fig. 8 is an example where the solutions provided by both the AWP procedure (red) and DE (pink) adapt to fit the more permeable zones of the reservoir. The example shows that the AWP methodology is able to trade length for increased permeability to place more of the well within the permeable zone. The DE solution shows that the well path could be further improved if it left the lower zone at an earlier point. However, since the AWP is guided by its limited sampling range given by d_m , it will stay on the straight-line path until some of the sampling points register the higher-permeability zone above.

The second cross section in Fig. 8 represents an example of grid geometry that can be problematic for well path development, since the fault represents a discontinuity in the high-permeability layers. However, the resulting trajectories show that the AWP procedure is capable of overcoming this type of discontinuity.

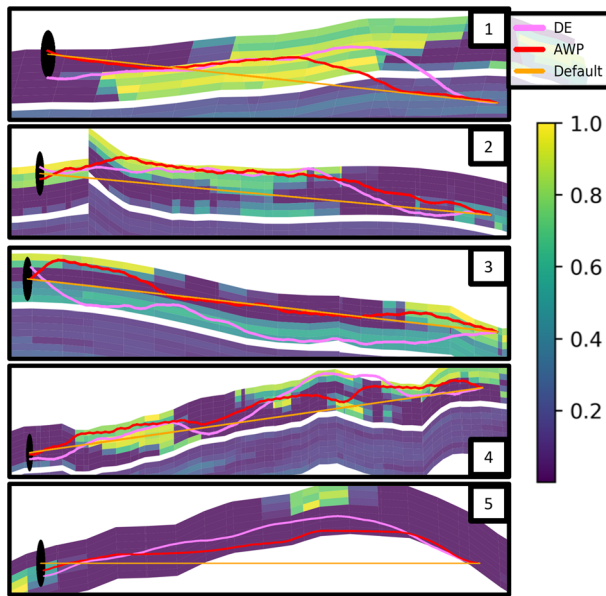


Fig. 8 Cross sections of the reservoir highlighting different features of the AWP methodology. The black circle represents the well toe (i.e., the target). The red connected lines represent the solution provided with AWP, the pink connected lines represent the solution provided with DE, and the orange line represents the straight line (default) running from the heel to the toe. The labels on the right indicate the cross section number

The third example highlights a potential problem regarding the application of the AWP procedure. In this example, the intermediate layer would give a higher aggregate permeability, and is therefore preferred by the DE method. The AWP procedure leads the trajectory towards the higher-permeability zone above the starting point. This encourages the AWP to stay close to the top of the reservoir, causing a lower fitness function value than the path obtained by the DE method.

The Olympus reservoir has a highly channelized upper part, and these channels meander downwards throughout the reservoir. Hitting these channels of high permeability is important for the productivity of the well. The fourth example shows that the AWP is able to adapt to the different parts in such a way that it increases the contact with productive regions. The path chosen by the AWP has clear similarities to the path obtained by the DE method.

The fifth example illustrates the cross section of a dome-like structure within a reservoir. In some reservoirs where the topography is rapidly changing, it is impossible to create straight-line trajectories that are wholly inside the reservoir. In this example, a sizable part of the straight-line well ends up outside the reservoir. In contrast, the AWP procedure yields a trajectory which follows the curvature of the reservoir much more closely. Similarly to the DE solution, the AWP procedure situates the entire well within the reservoir, thus increasing the production potential significantly.

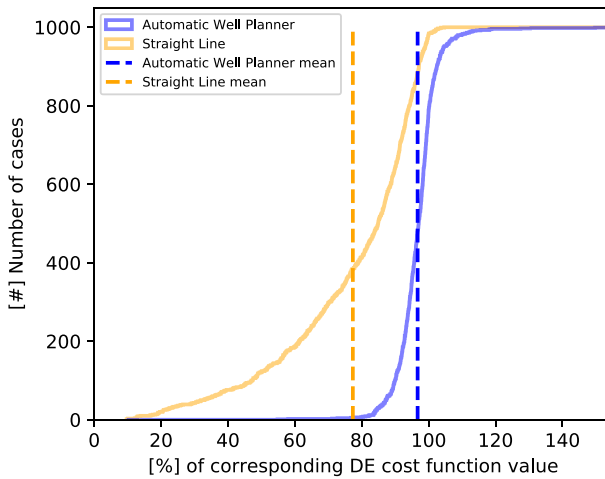


Fig. 9 Illustration of fitness function value distribution, as given by Eq. (6), relative to the fitness function values obtained by the DE algorithm

3.3.2 Statistical Comparison

The five examples above illustrate, in a case-by-case fashion, important aspects of the application of the AWP procedure. Here a statistical analysis is presented that compares the fitness function distribution corresponding to AWP-generated well paths against the distribution obtained when using SL well paths. The analysis is performed using an additional set of 1,000 new cross sections.

The fitness function values obtained using both the straight-line well paths and the AWP trajectories are compared against fitness function values corresponding to DE trajectories on the same cross sections. Keep in mind, however, that these values are obtained at a significantly higher computational cost per individual well. Using a modern CPU, the average time spent for DE per well was 250 seconds, while the AWP spent 0.022 seconds creating a similar trajectory. Leveraging the built-in parallelization capabilities of AWP, the training time was 3 hours. Cost function distributions for the SL and AWP trajectories are shown in Fig. 9. Note that each cost function point in these distributions is normalized by the DE solution value obtained for the same cross section; the abscissa therefore represents the percentage relative to the corresponding DE cost function value. As expected, Fig. 9 confirms that cost function values obtained using the AWP procedure are significantly higher compared to SL well paths for the same cross sections. Furthermore, the AWP procedure yields values that are close to those obtained using the DE algorithm. Given its performance in terms of both speed and fitness function value, the effectiveness of the AWP procedure is further assessed in a verification study in the next section.

4 Application

In this section, the AWP procedure developed in the preceding sections is coupled with the Olympus reservoir to investigate the impact of substituting the straight-line parameterization with AWP when generating and simulating a random set of producers in the reservoir model. To further test the generality of the AWP methodology, we graphically display the results of applying the AWP on a few wells in the Volve reservoir. The Volve reservoir is described in detail in Sen et al. (2019).

4.1 Dynamic Verification with AWP

The AWP is incorporated within the open-source field development framework FieldOpt (Baumann et al. 2020). The AWP procedure interacts as a drop-in replacement for the SL parameterization already included in the framework. A net-present-value (NPV) formulation is used as the objective function

$$\text{NPV} = \sum_{t=1}^T \frac{R_t}{(1+d)^t} - w_t. \quad (12)$$

Here we use a discount factor of $d = 0.08$, while the well costs w_t are accumulated by the length of the well times a constant. The time interval is 1 year and is evaluated over a total of 20 years. R_t here represents the cash flow

$$R_t = c_o C_{op} - c_{wp} C_{wp} - c_{wi} C_{wi}. \quad (13)$$

where C_{op} is the production of oil in sm^3 , C_{wp} is the water production in sm^3 , and C_{wi} is the water injection in sm^3 , within the period t . Oil price (c_o) is set to 377 \$/sm³, while water production (c_{wp}) and injection costs (c_{wi}) are set to 50 \$/sm³ and 17 \$/sm³, respectively.

In the problem formulation there are five water injectors with fixed positions surrounding a large oil-containing volume limited by an injection pressure threshold (BHP) of 230 barsa. The problem formulation contains a single producer set to a constant drawdown pressure of 150 barsa.

4.2 Results from Random Well Placement

In the following, the performance of trajectories obtained using the AWP procedure and SL well paths is compared in terms of liquid production and NPV value. For this comparison, a set of 1,990 random heel and toe values $\{(\mathbf{x}_h, \mathbf{x}_t)\}$ were generated. SL and AWP trajectories are generated using each pair of heel and toe coordinates. Wells generated using the AWP are expected to yield higher production than the SL trajectory, due to traversing parts of the reservoir with higher permeability. A higher liquid production does not necessarily translate into a higher NPV. The resulting liquid

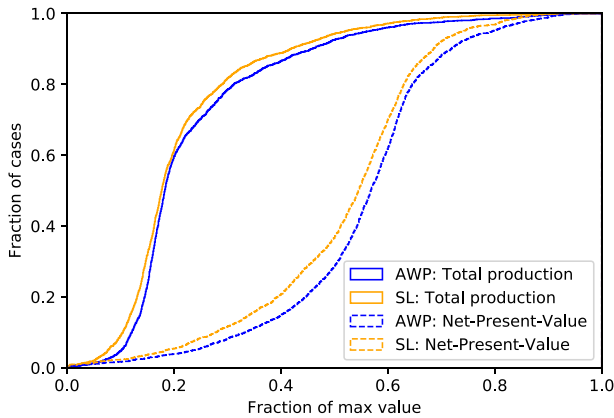


Fig. 10 Cumulative histogram of total liquid production (solid) and NPV (dashed) distribution, for the SL (blue) and AWP trajectories (orange). The x -axis is normalized to the highest liquid production or NPV value

Table 3 Means, standard deviation, and relative standard deviation of NPV and total liquid production (Liq.) for the random well placement comparison

Solution	NPV [\$]	NPV std [\$]	NPV rsd [%]
AWP	8.39E8	2.44E8	0.290
SL	7.87E8	2.53E8	0.325
	Liq. [SM3]	Liq. std [SM3]	Liq. rsd [%]
AWP	4.40E6	2.80E6	0.63
SL	4.03E6	2.53E6	0.62

production and NPV values are plotted in Fig. 10. It shows that wells with AWP-generated trajectories clearly yield higher productivity and NPV compared to SL wells. In particular, there is a significant gap between the two liquid production curves from the 0.2 to the 0.6 fraction.

The production and NPV results for all cases are also collected in Table 3, including the standard deviation for the reservoir simulations of these 1,990 well placements. In accordance with the plot in Fig. 10, the table shows that the AWP method increases both NPV and liquid production. The AWP method also reduces the standard deviation in NPV between the cases. Since the AWP places the wells in high-permeability sands, less variation in the geology traversed by the AWP is expected, and therefore less variation in the NPV.

For the liquid production values, the AWP gives a similar relative standard deviation. Earlier water breakthrough and higher water cut will counteract the increased income from higher oil production. Despite this, about 65% of the cases experienced an improvement when the SL was replaced with the AWP, in both NPV and liquid production.

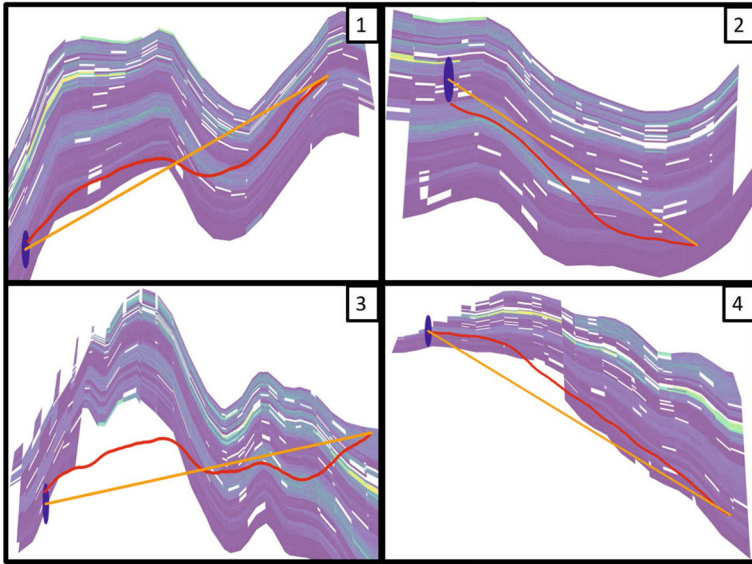


Fig. 11 Four cross sections 1, 2, 3, and 4 collected from the Volve reservoir. The blue circle represents the well toe (i.e., the target). The red connected lines represent the solution provided with AWP, and the orange line represents the straight line (default) running from the heel to the toe

4.3 Applying the AWP on Volve

In order to test the generality of the AWP procedure, the AWP that was trained on Olympus is used to make trajectories on the Volve field. Volve is less channelized than Olympus. Although Volve has permeable streaks, the main challenge is to account for the greater contrast in height and geometry. In the training procedure for Olympus, we put $t_d = 20$ in Eq. (7). In order to better account for the greater variation of depth in Volve, the AWP was additionally trained for 100 iterations with a $t_d = 80$ on a random set of wells within Volve. This enabled the procedure to become more familiar with the less strict deviation constraints.

As can be observed in Fig. 11, the AWP methodology is capable of ensuring that a greater portion of the trajectory remains within the reservoir, compared to the straight-line alternative. In Fig. 11, cross sections 1, 3, and 4 clearly demonstrate this behavior. Interestingly, cross section 2 shows that even when unfamiliar channels with higher permeability are detected in the vicinity of the trajectory, the AWP procedure follows a reasonable contour towards its target point.

5 Discussion

In this work we have used a fitness function, Eq. (6), that is essentially a measure of the permeability along well paths. From Table 3, which summarizes the results in Sect. 3.3.2, we observe that AWP trajectories yield an average fitness function result of 96.7%, compared to an average of 77.3% when using traditional SL on the set

of random well paths. This improvement implies that learning has occurred during the training process. In particular, these results show that the AWP procedure, when trained sufficiently, is capable of adapting the straight line into a trajectory that better fits the geometry and property field of the reservoir.

As the current AWP is trained to increase the permeability along the well path, the AWP is expected to increase productivity versus the SL. The verification study in Sect. 4.2 indeed shows that AWP trajectories in general yield higher liquid production than SL well paths. Interestingly, the standard deviation of NPV is smaller for AWP trajectories compared to the standard deviation of corresponding SL well paths. This could indicate that AWP smooths the response surface. A smoother response surface could remove local optima, which would be a beneficial feature for most optimization algorithms.

While the ultimate goal in field development is to increase the NPV, there is no direct link between productivity and NPV, as increased productivity can lead to more water production, with an adverse effect on the NPV. While the training of our AWP is expected to increase productivity, the NPV is also increased for the AWP wells compared to the SL wells, as illustrated by Fig. 10. Actually, the separation between the productivity curves is not visibly larger than the separation between the NPV curves; thus the current AWP also seems well fitted to increase NPV in addition to production.

By including other penalties and parameters into the fitness function, we expect to be able to train an AWP that is more adapted to increasing the NPV instead of productivity. Parameters might include porosity and saturation, while penalties could depend on distance to oil-water contact and distance to other wells. The set and weighting of parameters and penalties would need to be tailored to the objectives for the wells, such as different fitness functions for injectors and producers.

When dealing with artificial neural networks, it becomes evident that there are both benefits and drawbacks to applying a method like the AWP procedure. The black-box nature of an artificial neural network makes it difficult to identify the situations where the procedure encounters problems or excels. Additionally, there are strict requirements when it comes to providing the necessary quality of the training data. The training data should represent situations the network might be exposed to; when increasing the complexity of the task, the only way to make it better is to increase the volume and quality of the training data set.

Although the algorithm for producing the network is stochastic, the artificial neural network remains deterministic. If the AWP were to be compared with DE, the search space is large enough to make it unlikely for DE to find the global maximum every time it is applied. DE will frequently find a local maximum close in value to the global maximum. However, the stochastic nature of the algorithm might lead to a slightly different trajectory. Using the AWP, on the other hand, will result in the exact same decisions being made. This is an important feature, as no information is lost throughout the trajectory-making process and can easily be recreated based on coordinates provided to the AWP.

The Volve cross sections presented in Fig. 11 demonstrate that even though we apply the AWP on a different reservoir, with slightly more training, the features of the problem remain the same. The AWP would similarly deliver increased accuracy

if it were trained from the ground up using cross sections from both reservoir models. Further, it is conceivable that with training on more data consisting of multiple reservoirs and properties, the procedure could become general enough to be applied on any reservoir without additional training.

6 Conclusion

The proposed AWP methodology is a fast and computationally efficient approach for deriving trajectories that adhere to local geological and geometric features. AWP draws inspiration from drilling operations where information about the entire section is unavailable. The procedure relies on a data acquisition process to incorporate data into the method, allowing the AWP to make decisions based on information from multiple sources of data. In this paper, the data sources are a set of permeability values spatially distributed in front of the current position.

The goal of the AWP procedure, as presented in this work, is to enhance current well parameterization techniques. The enhancement could be especially fruitful for well placement optimization efforts. The AWP is also well suited for optimization under uncertainty when the uncertainty is represented by an ensemble of models. In this setting, the low-order parameterization of the AWP allows for customized well paths for each member in the ensemble, thereby obtaining a well placement which is optimized with respect to adaptive well paths (Kristoffersen et al. 2020).

The AWP application shows that some traits of the decision-making process can be generalized and incorporated within an artificial neural network. AWP solutions, compared to traditional SL trajectories, result in generally higher productivity given the objective of traversing high-permeability zones. Future developments include coupling of the methodology with optimization, including robust optimization, as well as expanding the AWP to deal with three dimensions, and testing other fitness function formulations.

Acknowledgements This research is a part of BRU21 – NTNU Research and Innovation Program on Digital and Automation Solutions for the Oil and Gas Industry (www.ntnu.edu/bru21), and Brage S. Kristoffersen is supported by Equinor ASA through BRU21. Carl Fredrik Berg is supported by the Research Council of Norway (Centers of Excellence funding scheme, project number 262644, PoreLab).

Funding Open access funding provided by NTNU Norwegian University of Science and Technology (incl St. Olavs Hospital - Trondheim University Hospital).

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Alizadeh B, Najjari S, Kadkhodaie-Ilkhchi A (2012) Artificial neural network modeling and cluster analysis for organic facies and burial history estimation using well log data: A case study of the south pars gas field, Persian Gulf Iran. *Computers Geosci* 45:261–269
- Arbus T, Wilson S (2019) Cybersteering: Automated geosteering by way of distributed computing and graph databases in the cloud. In: Unconventional resources technology conference (URTEC); Society of Petroleum Engineers, pp 1361–1368
- Baumann EJM, Dale SI, Bellout MC (2020) FieldOpt: A powerful and effective programming framework tailored for field development optimization. *Computers Geosci*. <https://doi.org/10.1016/j.cageo.2019.104379>
- Bellout MC, Echeverría Ciaurri D, Durlowsky LJ, Foss B, Kleppe J (2012) Joint optimization of oil well placement and controls. *Comput Geosci* 16(4):1061–1079
- Bouzarkouna Z, Ding DY, Auger A (2011) Well placement optimization with the covariance matrix adaptation evolution strategy and meta-models. *Comput Geosci*. <https://doi.org/10.1007/s10596-011-9254-2>
- Chakraborty UK (2008) *Advances in differential evolution* vol. 143. Springer, New York. <https://doi.org/10.1007/978-3-540-68830-3>
- Collobert R, Weston J, Bottou L, Karlen M, Kavukcuoglu K, Kuksa P (2011) Natural language processing (almost) from scratch. *J Mach Learn Res* 12:2493–2537
- Fonseca R, Della Rossa E, Emerick A, Hanea R, Jansen J (2018) Overview of the Olympus field development optimization challenge. In: 16th European Conference on the Mathematics of Oil Recovery, pp. 1–10. European Association of Geoscientists & Engineers. <https://doi.org/10.3997/2214-4609.201802246>
- Forouzanfar F, Reynolds AC, Li G (2012) Optimization of the well locations and completions for vertical and horizontal wells using a derivative-free optimization algorithm. *J Petrol Sci Eng* 86:272–288. <https://doi.org/10.1016/j.petrol.2012.03.014>
- Haghshenas Y, Emami Niri M, Amini S, Amiri Kolajoobi R (2020) Developing grid-based smart proxy model to evaluate various water flooding injection scenarios. *Petrol Sci Technol* 38(17):870–881
- Hassani H, Sarkheil H, Foroud T, Karimpooli S, et al (2011) A proxy modeling approach to optimization horizontal well placement. In: 45th US Rock Mechanics/Geomechanics Symposium. American Rock Mechanics Association
- Jean S, Cho K, Memisevic R, Bengio Y (2014) On using very large target vocabulary for neural machine translation. arXiv preprint [arXiv:1412.2007](https://arxiv.org/abs/1412.2007)
- Kristoffersen B, Silva T, Bellout MC, Berg, CF (2020) An automatic well planner for efficient well placement optimization under geological uncertainty. In: 17th european conference on the mathematics of oil recovery, pp. 1–16. European Association of Geoscientists & Engineers
- Lesso Jr W, Kashikar S, et al (1996) The principles and procedures of geosteering. In: International association of drilling contractors international/society of petroleum engineers drilling conference and exhibition. Society of Petroleum Engineers
- Li Q, Omeragic D, Chou L, Yang L, Duong K, et al (2005) New directional electromagnetic tool for proactive geosteering and accurate formation evaluation while drilling. In: SPWLA 46th annual logging symposium. Society of Petrophysicists and Well-Log Analysts
- Maus S, Gee T, Mitkus A.M, McCarthy K, Charney E, Ferro A, Liu Q, Lightfoot J, Reynerson P, Velozzi D.M, et al (2020) Automated geosteering with fault detection and multi-solution tracking. In: International Association of Drilling Contractors International/Society of Petroleum Engineers Drilling Conference and Exhibition. Society of Petroleum Engineers
- Mikolov T, Deoras A, Povey D, Burget L, Černocký J (2011) Strategies for training large scale neural network language models. In: 2011 IEEE Workshop on Automatic Speech Recognition & Understanding, pp. 196–201. IEEE
- Price KV, Storn RM, Lampinen JA (2005) *Differential evolution: A practical approach to global optimization*. Springer, New York
- Sayyafzadeh M, Alrashdi Z (2019) Well controls and placement optimisation using response-fed and judgement-aided parameterisation: Olympus optimisation challenge. *Comput Geosci*. <https://doi.org/10.1007/s10596-019-09891-y>
- Sen S, Ganguli S.S, et al (2019) Estimation of pore pressure and fracture gradient in Volve field, Norwegian North Sea. In: SPE Oil and Gas India Conference and Exhibition. Society of Petroleum Engineers

- Shahkarami A, Mohaghegh S.D, Gholami V, Haghighat S.A, et al (2014) Artificial intelligence (AI) assisted history matching. In: SPE western North American and Rocky Mountain joint meeting. Society of Petroleum Engineers
- Stanley K.O (2004) Efficient evolution of neural networks through complexification. Ph.D. thesis, Department of Computer Sciences, The University of Texas at Austin
- Stanley KO, Miikkulainen R (2002) Evolving neural networks through augmenting topologies. *Evolut Comput* 10(2):99–127. <https://doi.org/10.1162/106365602320169811>
- Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A (2015) Going deeper with convolutions. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1–9
- Tompson J.J, Jain A, LeCun Y, Bregler C (2014) Joint training of a convolutional network and a graphical model for human pose estimation. In: *Proceedings of the 27th international conference on neural information processing systems*, vol 1. *Advances in neural information processing systems*, pp. 1799–1807
- Winkler H (2017) Geosteering by exact inference on a bayesian network. *Geophysics* 82(5):D279–D291