



# An adaptive moment estimation framework for well placement optimization

Yazan Arouri<sup>1</sup> · Mohammad Sayyafzadeh<sup>1</sup>

Received: 2 March 2021 / Accepted: 25 January 2022 / Published online: 19 February 2022  
© The Author(s) 2022

## Abstract

In this study, we propose the use of a first-order gradient framework, the adaptive moment estimation (Adam), in conjunction with a stochastic gradient approximation, to well location and trajectory optimization problems. The Adam framework allows the incorporation of additional information from previous gradients to calculate variable-specific progression steps. As a result, this assists the search progression to be adjusted further for each variable and allows a convergence speed-up in problems where the gradients need to be approximated. We argue that under computational budget constraints, local optimization algorithms provide suitable solutions from a heuristic initial guess. Nonlinear constraints are taken into account to ensure the proposed solutions are not in violation of practical field considerations. The performance of the proposed algorithm is compared against steepest descent and generalized pattern search, using two case studies — the placement of four vertical wells and placement of 20 nonconventional (deviated, horizontal and/or slanted) wells. The results indicate that the proposed algorithm consistently outperforms the tested methods in terms computational efficiency and final optimum value. Additional discussions regarding nonconventional parameterization provide insights into simultaneous perturbation gradient approximations.

**Keywords** Well placement · Gradient-based algorithm · Adaptive moment estimation · Field development optimization

## 1 Introduction

Field development strategies are a crucial part of the efficient management of fluids in the subsurface. This includes water resources [1], carbon storage in geological formations [2] and hydrocarbon reserves [3–5]. A major component of a field development plan is the configuration of the wells. This may include the number of wells, type (injector or producer) and their trajectories, amongst other considerations. The reservoir response for a set of well configurations is used to assess the suitability for the given objective, such as Net-Present-Value (NPV) or cumulative oil production. Obtaining the reservoir response entails the running of an expensive reservoir simulation for each set

of well configurations. Computational demand increases when geological uncertainty needs to be considered, which is typically taken into account by using a suite of reservoir models. Additional considerations include field constraints, such as inter-well distance and maximum well length. As a result, optimization methods have been employed to efficiently find an optimal well configuration (solution).

Well placement optimization problems are characterized by the highly nonlinear relationship between the input (well locations and reservoir model) and the output (fluid production volumes). This results in nonconvex and multimodal objective function landscapes that require computationally efficient methods to traverse. Consequently, population-based algorithms have been investigated and are commonly applied to well placement optimization problems. These include the application of genetic algorithms (GA) [5–7], particle swarm optimization (PSO) [4, 8, 9] and covariance matrix adaptation evolution strategy [10].

Although these techniques produce promising solutions by exploring the space more globally, they arguably require a prohibitively large number of computationally expensive reservoir simulation calls. In some practical situations, the available computational budget may not allow the

---

✉ Yazan Arouri  
yazan.arouri@adelaide.edu.au

Mohammad Sayyafzadeh  
mohammad.sayyafzadeh@adelaide.edu.au

<sup>1</sup> Australian School of Petroleum and Energy Resources,  
The University of Adelaide, Adelaide, 5005, SA, Australia

application of these global optimizers to well placement problems. These techniques stochastically explore the search space using a pool of solutions (known as the population). Through learning parameters, solutions which are more promising undergo selection and stochastic manipulation which results in a new population. As such, in the case of heuristic initialization, it is difficult to ensure that the initial solutions based on reservoir engineering considerations are used or exploited adequately. As a result, such solutions may be absent in future generations and replaced by completely different ones. Although these generational mechanisms can be fruitful for global search, they inevitably increase the computational demand.

In these situations, local optimizers may provide suitable solutions in a more computationally efficient manner. Given the nature of local optimization methods, they are more likely to converge to a local optimum near the starting point. Consequently, their performance is heavily affected by the quality (location in the search space) of the initial guess. In an attempt to overcome this limitation, methods such as multi-start [11] and restart techniques [12] have been introduced. As their name suggests, multi-start methods initialize the algorithm from multiple different initial guesses which will (ideally) converge to different optimums. The aim is then to select the most optimal solution from all starting points. On the other hand, restart techniques execute the algorithm for a number of iterations before restarting the algorithmic parameters and using the last solution found as the initial guess. Although these approaches may succeed in delaying the convergence to an optimum, they inevitably reduce the computational efficiency of local approaches. This becomes counter-productive for optimization problems that are computationally constrained as is commonly the case for practical scenarios. By contrast, a less burdensome approach is to utilize prior knowledge to improve the quality of the initial guess. In this undertaking, the aim is to leverage the advantages of local methods to increase the convergence speed to an improved solution. It should be noted that this approach assumes the initial guess is based on sound knowledge, as is the case for the optimization problem under consideration. However, this does not preclude the use of local optimization methods to improve lower quality initial guesses either.

Optimization of well location using local methods has received some attention in the literature. These techniques can be categorized into two groups, gradient-based and pattern-search methods. Pattern-search algorithms have been used for well placement optimization, typically as local optimizers within hybrid approaches [13–16]. These methods provide a gradient-free procedure that relies on the direct search of the solution space. This search occurs through the use of a stencil, which is typically

a collection of points obtained by perturbations of equal size in all directions. The center of the stencil moves when there is an improvement in the objective function value. If no improvement occurs, the stencil size is reduced and the procedure is repeated. However, in high-dimensional optimization problems (e.g., placement of multiple nonconventional — deviated, horizontal and/or slanted — wells) the need to search perturbations in all directions may become a computational limitation.

Gradient-based approaches involve the utilization of a gradient computed through either an adjoint system [17–20] or approximation techniques [21, 22]. The adjoint method has been applied in several pieces of literature regarding vertical well placement optimization [19, 20, 23]. These methods rely on indirect approaches, which derive the well location sensitivity through gradients based on well control. Vlemmix et al. [23] extended this method to nonconventional wells by placing pseudo-sidetracks in grid blocks adjacent to the well path at specified trajectory points. However, this method is limited to only modifying the well along these trajectory points and not its overall location in the reservoir. More recently, Volkov and Bellout [24] use a combined technique for the optimization of nonconventional wells. The method uses the adjoint formulation to find key partial derivative terms that are then approximated using finite difference methods. The practical application of methods which rely on either adjoint formulations or adjoint-based gradients can be limited as they are not readily available in all commercial simulators in well placement optimization contexts. In addition, finite difference-based methods may lose efficiency when scaled to large dimensional problems, as they require one or two function evaluations for the perturbation in each dimension.

On the other hand, approximating the gradient, using simultaneous perturbation, provides an alternative. These methods rely on only objective function values and act as black-box methods. Bangerth et al. [21] applied an integer variant of the Simultaneous Perturbation Stochastic Approximation (SPSA) algorithm to well placement problems, including placing seven vertical wells in a simple two-dimensional model. Leeuwenburgh et al. [25] applied an ensemble method (EnOpt) to optimize the areal locations (x and y-coordinates) of nine vertical wells in a two-dimensional model. Li et al. [26] applied the integer variant of SPSA to the joint optimization of well controls and well placement on three case studies, including the benchmark PUNQ-S3 model. Jesmani et al. [22] applied continuous variants of SPSA to optimize the location of a single nonconventional well in the presence of four pre-existing injection wells. Simultaneous perturbation methods can provide an approximated gradient with only two function evaluations (if central), regardless of the number of decision variables. This makes the algorithms,

which employ simultaneous perturbation gradient approximation, computationally efficient for nonconventional well placement optimization problems.

Another important consideration is the parameterization of the decision variables in the problem formulation. Various parameterizations have been utilized in the literature. The simplest kind is the placement of vertical wells, which requires only two variables for each well location. In such a case, the well locations are defined either by the continuous Cartesian coordinates ( $x$  and  $y$ ) or by the cell indices ( $i$  and  $j$ ) [27]. In nonconventional wells (deviated, horizontal, slanted), the completion trajectory can be defined as two (heel and toe) points in space connected by a straight line. There are two common parameterizations for nonconventional wells: Cartesian coordinates and spherical coordinates. When using spherical coordinates, the completion trajectory is defined by the heel point using Cartesian coordinates ( $x$ ,  $y$  and  $z$ ), and spherical coordinates to define the length of the well ( $L$ ), the inclination angle ( $\phi$ ) and the azimuth angle ( $\theta$ ) in the horizontal plane [7, 28, 29]. In such parameterization, a well length constraint can be handled through simple bounds. On the other hand, when using Cartesian parameterization, both the heel and toe are defined by Cartesian coordinates [9, 30]. In Sayyafzadeh and Alrashdi [9], the implemented parameterization represents the heel and toe by  $x$  and  $y$ -coordinates and the  $z$ -coordinate is defined as a percentage between the bounding surfaces of the reservoir. The selection of parameterization can be important when approximating gradients using simultaneous perturbation. This is because the decision variables need to have similar sensitivities to the perturbation size in the gradient approximation. If this is not the case, the sensitivity to some parameters will be masked by others.

In practice, field development planning is a multi-disciplinary task, which includes an understanding of suitable well locations based on reservoir engineering judgement. To this end, we argue in situations under computational constraints, local methods can be leveraged to improve on these initial guesses in an efficient manner. These local methods will be able to produce improvements that are in line with considerations accounted for in the initial guess. In this study, we focus on the development of a first-order algorithm based on the adaptive moment estimation (Adam), for application to constrained well placement optimization. The algorithm is a combination of SPSA as a gradient approximation within an adaptive moment estimation framework. It is referred to as Adam-SPSA.

The adaptive moment estimation framework introduces the idea of using dimension-wise step-sizes in gradient-based algorithms. This allows the search direction to be

tailored for each dimension accordingly. In addition, this framework considers the accuracy of the approximated gradient by estimating the first and second order moments. As a result, these estimations aid the algorithm in guiding the search to more promising areas. The adaptive moment estimation (Adam) framework has found significant success in optimization problems in machine learning applications, including Google's translation system [31] and image processing [32, 33]. In these applications, the objective function is considered noisy as it is the summation of a random subset of cost (or loss) functions, from which a gradient is approximated. Although the objective function in well placement optimization using Adam-SPSA is not noisy (even if geological uncertainty is incorporated), stochasticity is still introduced by the random and simultaneous perturbations used in SPSA. This enables the extension of Adam to well placement optimization problems. More recently, olkov and Bellout [3] successfully applied the technique to well control optimization and shows its applicability to such problems.

In this study, we investigate the application of Adam-SPSA to vertical and nonconventional well placement optimization. This includes a two-dimensional visual example to compare the search directions of the adaptive moment estimation framework and the steepest descent framework. Additionally, Adam-SPSA is employed to two well placement optimization problems in the PUNQ-S3 model. The results are compared to the conventional steepest descent SPSA algorithm (SD-SPSA) and a pattern search technique, generalized pattern search (GPS).

The outline of the paper is as follows. The problem statement, including the formulation of the optimization problem, as well as the objective function are presented first. Next, the adaptive moment estimation framework is introduced. Following this, we begin the numerical results with a visualization of the gradient-based methods using a simple two-dimensional problem to reflect the improved search directions of the proposed algorithm. In addition, the numerical results for well placement optimization on two numerical case studies of increasing complexity are presented. The significance and implications of these results are then discussed, followed by concluding remarks.

## 2 Problem formulation

### 2.1 Problem statement

The optimization problem involves the minimization of a defined objective function where the well locations are the variables of interest. The well placement optimization

problem can be formulated as a general optimization problem as follows:

$$\begin{aligned} & \min_{\mathbf{x} \in \mathbb{R}^n} f(\mathbf{x}) \\ & \text{subject to :} \\ & c_i(\mathbf{x}) \leq 0 \quad i \in K \\ & c_i(\mathbf{x}) = 0 \quad i \in I, \end{aligned} \tag{1}$$

where,  $f(\mathbf{x})$  is the objective function,  $\mathbf{x}$  denotes the vector of decision variables,  $n$  is the number of decision variables,  $I$  and  $K$  are sets of indices for equality and inequality constraint functions,  $c_i(\mathbf{x})$ , respectively. These constraint functions may include linear and nonlinear constraints, as well as bound constraints.

### 2.2 Objective function

Here, we consider the objective function to be the negative of the NPV for a given lifespan. The objective function is defined as

$$\begin{aligned} f(\mathbf{x}) &= -\text{NPV}(\mathbf{x}) \\ &= \sum_{i=1}^{N_t} \frac{-r_{o,i} Q_{o,i}(\mathbf{x}) + c_{wp,i} Q_{wp,i}(\mathbf{x}) + c_{wi,i} Q_{wi,i}(\mathbf{x})}{(1+b)^{t_i}}, \end{aligned} \tag{2}$$

where  $r_{o,i}$ ,  $c_{wp,i}$  and  $c_{wi,i}$  are the sale price of oil and costs of water separation and injection, respectively, all of them per unit volume and defined from time  $t_i$  to  $t_{i+1}$  (there are  $N_t$  such intervals),  $Q_{o,i}$ ,  $Q_{wp,i}$  and  $Q_{wi,i}$  denote the field oil production, field water production and field water injection volumes during the mentioned output interval and  $b$  is the discount rate.

## 3 Methodology

### 3.1 Gradient approximation

The gradient approximation utilized in this study is SPSA. SPSA was first introduced by Spall [34] for problems whose analytical derivatives are unavailable. SPSA only requires two function evaluations to approximate a gradient using the central-difference method. This allows SPSA to be very efficient for high dimensional problems. The stochastic gradient approximation is:

$$\begin{aligned} \hat{g}_k(\mathbf{x}_k) &= \begin{bmatrix} \frac{f(\mathbf{x}_k + c_k \Delta_k) - f(\mathbf{x}_k - c_k \Delta_k)}{2c_k \Delta_{k1}} \\ \vdots \\ \frac{f(\mathbf{x}_k + c_k \Delta_k) - f(\mathbf{x}_k - c_k \Delta_k)}{2c_k \Delta_{kp}} \end{bmatrix} \\ &= \frac{f(\mathbf{x}_k + c_k \Delta_k) - f(\mathbf{x}_k - c_k \Delta_k)}{2c_k} \\ &\quad \times \left[ \Delta_{k1}^{-1}, \Delta_{k2}^{-1}, \dots, \Delta_{kp}^{-1} \right]^T, \end{aligned} \tag{3}$$

where, the mean-zero  $p$ -dimensional random perturbation vector,  $\Delta_k = [\Delta_{k1}^{-1}, \Delta_{k2}^{-1} \dots \Delta_{kp}^{-1}]^T$ , has a user-specified distribution and  $c_k$  is a positive scalar. The convergence theory of the SPSA algorithm can be found in Spall [34]. An important consideration for this theory is that Spall [34] recommends the use of the Bernoulli distribution. The selection of the optimal distribution perturbation vector was studied in Sadegh and Spall [35]. The gradient approximation is calculated using SPSA with the parameters following the recommendations given in Spall [36]. These are used to calculate the proposed perturbation size from the following gain sequence:

$$c_k = \frac{c}{(k+1)^\gamma}, \tag{4}$$

where,  $k$  is the iteration number,  $\gamma$  is a positive scalar constant and  $c$  is the initial perturbation size. Readers are referred to Spall [36] for additional implementation guidelines.

### 3.2 Adaptive moment estimation framework

Kingma and Ba [37] first introduced the adaptive moment estimation (Adam) framework as a gradient-based optimization algorithm which utilizes first-order information. The framework computes distinct progression steps for each decision variable (dimension) based on estimates of the first and second moments extracted from the gradient approximations [37]. In comparison, the steepest descent framework utilizes the approximated gradient as the progression step and, as such, the same step is taken for all decision variables.

The two pieces of information Adam extracts and utilizes are estimates of the first and second moments. For a random variable,  $X$ , the first moment is defined as the mean, or expected value,  $E[X]$ , about the origin [38]. This random variable,  $X$ , is taken to be the gradient approximation. Consequently, the estimation of the first moment is in fact an estimation of the expected value of the gradient. Following this, the definition of the second central moment of the random variable  $X$  is the variance about the mean, defined by Lefebvre [38]:

$$\text{Var}(X) = \text{Diag}(E[(X - E[X])^2]), \tag{5}$$

This can be further simplified with algebraic manipulation while assuming the variables of  $X$  are independent. Additionally, in the long-run, the mean of the gradients tends to approach zero as a local optimum is approached. This results in the following definition of the uncentred variance:

$$\text{unVar}(X) = \text{Diag}(E[X^2]), \tag{6}$$

Given that the random variable,  $X$ , is the gradient approximation, the uncentred variance is the expected value

of the element-wise gradient squared. The Adam framework then uses an exponential moving average to put greater weighting on the most recent approximated gradients when estimating the first and second moments. Details of the convergence theory for the adaptive moment estimation framework is presented in Kingma and Ba [37]. Framework 1 presents the steps of Adam.

**Framework 1** - Adaptive Moment Estimation

0. Initialize iteration counter, select initial guess,  $x_0$ , assign nonnegative constant parameters: step-size  $\omega$  and exponential decay rate for moment estimates  $\beta_1$  and  $\beta_2$
1. Initialize first and second moment vectors,  $m_0$  and  $v_0$ , which are  $n \times 1$  column vectors
2. Approximate stochastic gradient
3. Update the first moment vector using update rule:
  - (a)  $m_k = \beta_1 \times m_{k-1} + (1 - \beta_1) \times \hat{g}_k$ , where  $m_k, m_{k-1}$  and  $\hat{g}_k$  are  $n \times 1$  column vectors
  - (b)  $\hat{m}_k = \frac{m_k}{1 - \beta_1^k}$ , where  $\hat{m}_k$  is a  $n \times 1$  column vector
4. Update the second raw moment vector:
  - (a)  $v_k = \beta_2 \times v_{k-1} + (1 - \beta_2) \times (\hat{g}_k \odot \hat{g}_k)$ , where  $\odot$  is the element-wise multiplication and  $v_k$  and  $v_{k-1}$  are  $n \times 1$  column vectors
  - (b)  $\hat{v}_k = \frac{v_k}{1 - \beta_2^k}$ , where  $\hat{v}_k$  is a  $n \times 1$  column vector
5. Update the iterate:
  - (a)  $x_{k+1} = x_k - \omega \times \frac{\hat{m}_k}{\sqrt{\hat{v}_k + \epsilon}}$

It should be noted that all operations are done in an element-wise manner. The  $\odot$  represents the element-wise multiplication of two vectors. Also, the operation on the hyper-parameters  $(\beta_1^k, \beta_2^k)$  are denoted as  $\beta_1$  and  $\beta_2$  raised to the power  $k$  (iteration number). Typical values for  $\beta_1, \beta_2$  are 0.9 and 0.999, respectively [37]. A small positive value,  $\epsilon = 10^{-8}$ , is used to avoid the division by zero [37].

In machine learning applications, the objective function is typically in the form of a loss function representing a sum of differences over a set of observations. For example, a common loss function is the summation of the squared differences,  $f(x) = \sum_{i=1}^N (y_i - \hat{y}_i)^2$ , where  $N$  is the number of observations and  $y_i$  and  $\hat{y}_i$  are the true value and predicted value for observation  $i$ , respectively. In many occasions, instead of calculating the gradient as  $\nabla_k f(x) = \sum_{i=1}^N \nabla_k (y_i - \hat{y}_i)^2$ , a random set of observations is used in each iteration instead of all the observations ( $\nabla_k f(x) \approx \sum_{i=1}^n \nabla_k (y_i - \hat{y}_i)^2$ ), where  $n$  is smaller than  $N$ . This gives an approximate gradient in each iteration. If these gradient approximations are averaged, the result is in the same direction as the true gradient.

In contrast, the objective function in well placement problems is not of similar form to a loss function and the gradient approximation can be computationally expensive. That is, the objective function,  $f(x) = -NPV(x)$ , requires reservoir simulations to obtain a objective value. To reduce the computational costs of gradient approximations, we use a simultaneous perturbation approach. It is worth mentioning that, in both cases (subset of objective functions or simultaneous perturbations), the gradient approximation accuracy increases if averaged over a number of approximations. That is, let  $g(x) = \nabla_k f(x)$  then the averaged gradient approximation is given by  $\hat{g}(x) = \sum_{j=1}^s g_j(x)$ , where  $s$  is the number of gradient approximations. Additionally, the theory of simultaneous perturbation suggests the approximated gradient is an unbiased estimator of the true gradient within a bias bound [34]. Adam has been shown to work reasonably well in such stochastic optimization problems, and because of the similarities, we use this framework with a simultaneous perturbation to address computational intensity of well placement problems.

**3.3 Parameter selection**

In this section, we present the rationale for the parameter values selected for perturbation size and step-size for each algorithm. The selection of appropriate parameter values for perturbation size and step-size are pivotal in the convergence of local algorithms. Previous work has investigated this topic for well control optimization problems [3].

**3.3.1 Perturbation size**

The perturbation size determines the amount of perturbation taken in a direction when calculating the gradient. In this work, we use a central difference to approximate the gradient using SPSA. As such, the perturbation is done in the forward and backward directions from the current solution. The gradient aims to capture the sensitivity of the objective function to changes in the solution. For well placement problems, such as those under consideration, this is guided by the grid block size. Considering this, the numerical simulation may not properly capture the sensitivity if the perturbation is too small. Similarly, a perturbation too large will not be representative of the local landscape. Due to spatially varying property values (e.g., saturations and permeabilities), there is an inherent discontinuity in the landscape. The selection of an appropriate perturbation becomes important as it is one of the remedies for this issue, to some extent.

Consequently, an initial perturbation size ( $c$ ) value of 0.05 was found to be suitable for a perturbation of

approximately one grid block. Adam-SPSA and SD-SPSA both employ the decaying sequence given by Eq. 4 to update the perturbation size at each iteration. A value of 0.101 was used for  $\gamma$  as recommended by Spall [36]. It is important to note that although Eq. 4 represents a decaying sequence the sensitivities are still captured by the gradient.

Firstly, this work is focused on optimization under practical computational constraints. As such, the decay of the perturbation size is limited for the computational budget considered. In addition, in this work we employ a continuous parameterization for well location for vertical and nonconventional wells. Lastly, the wells are not necessarily at the centre of the grid blocks as the search progresses.

In turn, this rationale was used to guide the selection of an initial stencil size for optimization using GPS. The stencil size in GPS represents the perturbation taken in each direction (decision variable) that forms the search points of the stencil at the current solution. For this reason, a stencil size that represents a perturbation of approximately one grid block is suitable. Consequently, the selected initial stencil size for GPS optimization runs is similar to the perturbation size value for Adam-SPSA and SD-SPSA.

### 3.3.2 Step-size

The step-size determines the progression of an algorithm from the current solution to the next proposed solution in the search direction. The effective step-size can be thought of as the step-size multiplied by the search direction. In other words, this is the difference between the current solution and the proposed solution. As such, a suitable effective step-size is one where the search does not swing from one bound to another as this may result in overstepping local minimums. On the other hand, an effective step-size that is too small may cause a prohibitive slowdown in convergence. Accordingly, for this work an effective step-size which represented approximately one grid block was used. To do this, the first two iterations of Adam-SPSA were used to find a suitable value for  $\omega$  (step-size) that gave an effective step-size of one grid block. In Adam-SPSA, the selected  $\omega$  value is constant throughout the optimization run (i.e. the step-size does not change as the search progresses). In addition, previous studies have shown that Adam-SPSA does not require a backtracking line-search given its adaptive search direction [3].

A similar routine was undertaken to find a similar effective step-size for SD-SPSA. SD-SPSA updates the step-size based on Eq. 7, where  $a$  is the initial step-size,  $A$  is a stability constant, and  $\alpha$  is a positive scalar constant. From Spall [36] we use the recommended values of 0.602 and  $0.1 \times \text{maximum number of iterations}$  for  $\alpha$  and  $A$ , respectively. Additionally, for SD-SPSA a backtracking

line-search is implemented to help guide the search to an improved objective function value. In this work, a cut-back value of 0.5 and a maximum of 5 cut-backs are used, after which the gradient is re-calculated and the line-search is repeated.

$$a_k = \frac{a}{(A + k + 1)^\alpha}, \quad (7)$$

### 3.3.3 Constraint handling

The consideration of physical field constraints is an important aspect for practical application of optimization in well location problems. In this work, three nonlinear constraints are considered to ensure the proposed solutions do not violate engineering principles. In addition, the decision variables are normalized between 0 (lower bound) and 1 (upper bound) for which simple bound constraints are applied. The first constraint considers a minimum inter-well distance between any pair of wells. When placing three-dimensional nonconventional wells, this constraint considers the minimum distance between any two points on the line segments representing the two wells. The second constraint is a polygon reservoir bound. In this constraint, the reservoir bound is approximated using a polygon shape to ensure any proposed well (both heel and toe for nonconventional wells) lies within this polygon. The third constraint considered relates to a maximum well length for nonconventional wells.

The violation of any of these nonlinear constraints is treated through the projection of an infeasible point onto the feasible domain. This projection is formulated into an optimization problem where a distance metric, the Euclidean distance, with respect to the infeasible point is minimized. That is, the distance between the original violating set of well location/s and a set of proposed well location/s is minimized. This optimization problem is subject to the same constraints as in the original problem. In this work, the constraint handling is implemented through MATLAB's *fmincon* function, which is a nonlinear programming solver. The solver is set to use a sequential quadratic programming (SQP) algorithm to solve this optimization problem. Details can be found on the MathWorks reference manual [39].

### 3.3.4 Constraint handling in gradient approximation

A sensitivity analysis was undertaken to gain insights into the effect that different types of constraints in gradient approximation had on the algorithm's performance. All parameter values were kept constant for each method. The difference between each method is the constraints considered when approximating the gradient using SPSA. This means once the forward ( $x_k + c_k \Delta_k$ ) and backward

$(x_k - c_k \Delta_k)$  perturbations are proposed they are subjected to the assigned constraints.

Three different combinations were tested. The first considered simple bound constraints to ensure the proposed forward and backward perturbations are within the lower and upper bounds. The second implemented a refined bound constraint which included simple bound constraints and the reservoir polygon constraint. That means if a forward or backward perturbation proposed well locations outside the defined reservoir polygon, they are projected back into the reservoir. This was to ensure the number of wells was consistent throughout the gradient approximation. The third set-up considered all three nonlinear constraints (mentioned in Section 3.3.3) as well as the bound constraints.

The results showed the worst performing combination was the simple bound set-up which considered only bound constraints for the forward and backward perturbation. This could indicate that although the perturbations are within the bounds, they are not corrected enough to result in a useful gradient. On the other hand, including all the nonlinear constraints could be over-correcting the perturbations. This may result in a gradient that is not representative of the local landscape causing it to perform worse than the refined bounds set-up. The best performing combination is the refined bound case containing bound constraints and the reservoir polygon constraint. This insight shows that the constraint handling considered for the perturbations has an effect on the quality of the gradient approximation. It is worth mentioning that only one simultaneous perturbation is used for gradient approximation.

## 4 Case studies

This section begins with a simple two-dimensional example to illustrate the differences in search direction between Adam-SPSA and SD-SPSA. Next, the results for a case study investigating the placement of four infill vertical

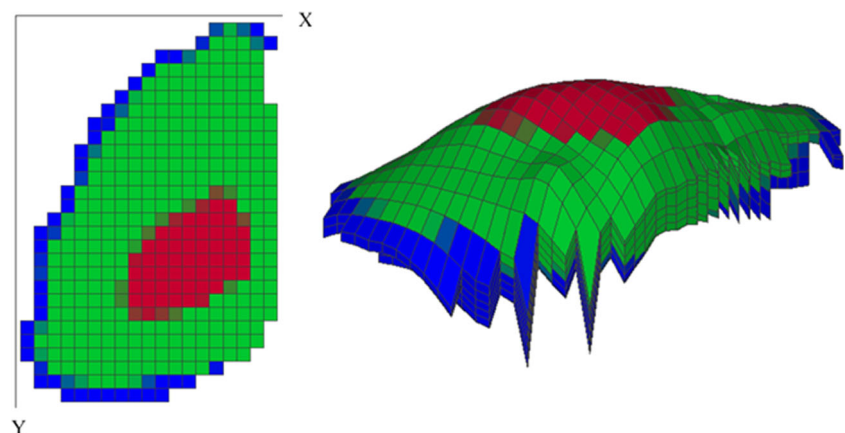
production wells are presented. The third case study considers the placement of 20 nonconventional wells considering three physical field constraints. All case studies use the three-dimensional PUNQ-S3 benchmark model, shown in Fig. 1 [40]. The model is an oil-saturated heterogeneous reservoir with a small gas-cap and strong aquifer support with bottom-drive and side support on two sides. The other two sides are defined as no-flow boundaries. The model is discretized into 19 by 28 by 5 grid blocks, in which 1,761 are active. The areal extent of the model is  $17 \times 10^6 m^2$ , with varying thicknesses between 20 and 30 meters. The reservoir has a lifespan of 10 years.

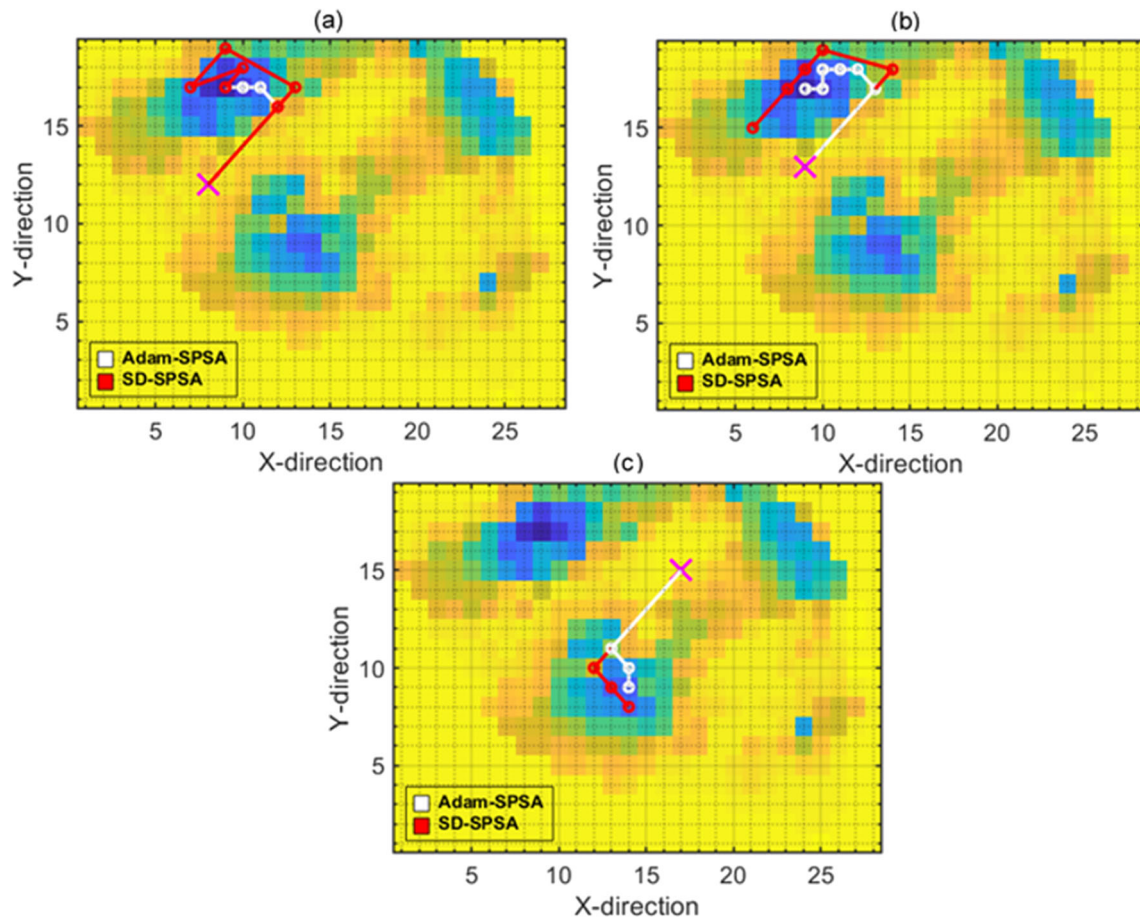
### 4.1 Case study 1 - two-dimensional visual example

In this case study, we visually compare the search directions from Adam-SPSA and SD-SPSA. The optimization problem is the placement of one vertical production well, resulting in 2 decision variables (x- and y- coordinates). Figure 2 shows the search steps undertaken by Adam-SPSA and SD-SPSA for three different initial starting points (represented by the cross). In Fig. 2a, both Adam-SPSA and SD-SPSA are able to reach the closest local minimum to the initial guess. However, Adam-SPSA only requires 4 iterations to do so while SD-SPSA requires 6. Figure 2b shows a different result where only Adam-SPSA is able to reach the closest local minimum after 6 iterations, whilst SD-SPSA converges after 5 iterations to a lower quality solution. It must be noted that SD-SPSA was allowed to continue for 3 additional iterations without any improvement. Similarly, Fig. 2c shows that Adam-SPSA was able to reach the closest local minimum in 3 iterations, whilst SD-SPSA did not converge to the same solution after 3 iterations.

As shown in Fig. 2 the search directions of SD-SPSA are always 45 degrees from the previous solution. This is attributed to the use of the Bernoulli distribution in the gradient (when using only one gradient approximation), which is directly used as the search direction. This problem

**Fig. 1** Top view of Layer 1 and three-dimensional representation of the PUNQ-S3 benchmark model showing the ternary saturations. Red is gas, green is oil and blue is water





**Fig. 2** Visual representations of search steps of optimization of one vertical well using three different initial starting points (pink cross). The numbers inside the box indicates the number of iterations needed

is more severe when the number of dimensions is greater and/or the sensitivity to variables is not similar. Unlike SD-SPSA, Adam-SPSA uses a dimension-wise adjusted search step which gives the algorithm more control and flexibility over movement in both x and y directions.

#### 4.2 Case study 2 - four vertical infill production wells

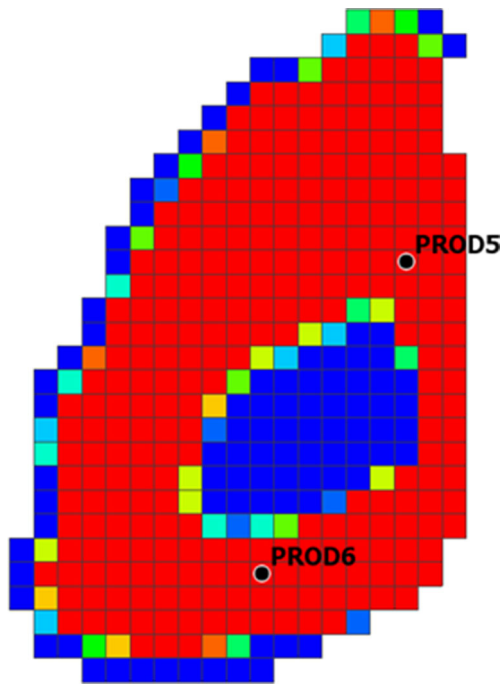
In this case study, two producers are considered pre-existing. These wells are PROD5 and PROD6 with x and y-coordinates of (2970,1890) and (1890, 4230), respectively, shown in Fig. 3. The objective is to minimize the function given by Eq. 2 by placing four vertical infill producers, resulting in a total of 8 decision variables. The wells are controlled by bottom-hole pressure (BHP) with a pressure of 2900 psi (200 bars) and a maximum liquid rate of 5660 STB/day (900  $sm^3/day$ ). The economic parameters are given in Table 1. Since the number of wells are fixed (i.e., is not a decision variable) and all wells are vertical, there is no impact of drilling and completion costs on NPV calculation. As such, the drilling and completion costs are

to reach final value for each algorithm. Yellow is high objective function value and blue is low objective function value

not considered in this case study, and the associated values in Eq. 2 are equated to zero. The optimization variables were normalized and bounded between 0 and 1. In this case study, a minimum inter-well distance of 300 meters is used. A box constraint was used to represent the reservoir bounds by placing upper and lower bounds on the x and y-coordinates.

As previously discussed, the step-sizes were used to ensure a similar effective step was taken in each algorithm. Adam-SPSA had parameter values of 0.07 and 0.05 for  $\omega$  and  $c$ , respectively. SD-SPSA had parameter values of 0.2 and 0.05 for  $a$  and  $c$ , respectively. To reduce the effect of the stochastic nature of the SPSA gradient approximation, both Adam-SPSA and SD-SPSA results were averaged over 10 optimization runs from the same reservoir engineering initial guess. Since GPS is a deterministic method, it was only run once. That is, for the same initial guess, GPS will result in the same solution. GPS was implemented using MATLAB's pattern search optimization tool [39]. The polling method used was the *Positive basis 2N* with an initial and maximum mesh size of 0.05. An incomplete polling was employed where the first search direction at





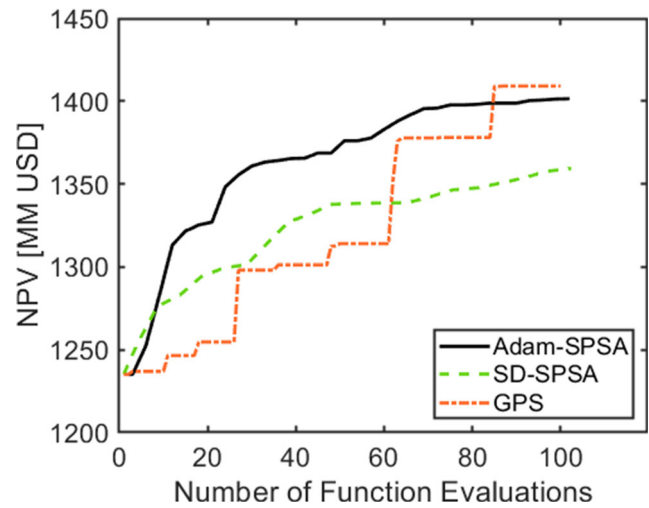
**Fig. 3** Initial oil saturation of Layer 1 in two-dimensions with two pre-existing production wells (PROD5 and PROD6). Red represents high oil saturation and blue represented low oil saturations

each iteration is the direction which gave the best solution in the previous iteration. The stopping criterion was set to a maximum number of function evaluations of 100 in this case study. For all optimization runs, the initial guess was taken to be an example of a possible solution based on reservoir engineering judgement, represented by the initial well locations from the PUNQ-S3 benchmark model. These wells are located around the gas/oil contact (GOC) to ensure that the water breakthrough is delayed whilst maximizing oil production.

Figure 4 shows the convergence plot for the results of the well location optimization for case study 2. The results show that SD-SPSA, on average, converged to a final optimum NPV of 1,359 MM USD after 103 function evaluations. In comparison, Adam-SPSA, on average, required only 30 function evaluations to achieve a similar NPV value (1,361 MM USD). This represents up to a 71% decrease in the required number of functional evaluations. Another important consideration is the standard deviation of the results from Adam-SPSA compared to SD-SPSA. The

**Table 1** Economic parameters for the two example problems

Parameter	Case study 2	Case study 3
Oil price ( $r_o$ )	\$80 USD/bbl	\$40 USD/bbl
Water handling costs ( $c_{wp}$ )	\$13 USD/bbl	\$5 USD/bbl
Discount rate (b)	0%	0%

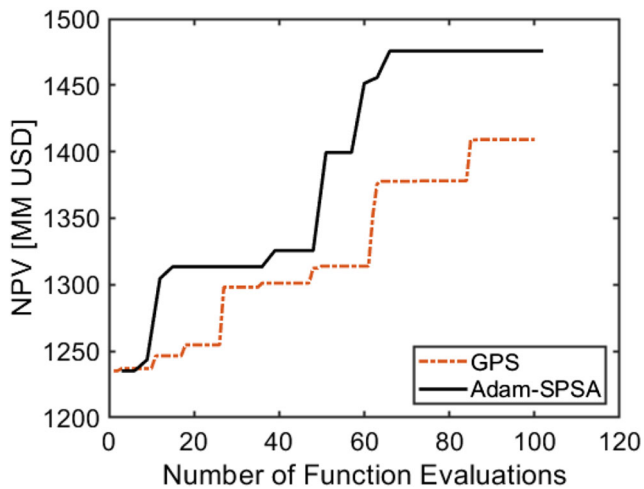


**Fig. 4** Convergence plot for the optimization of four vertical infill wells comparing Adam-SPSA (solid line), SD-SPSA (dashed line), and GPS (dash-dotted line). The results of Adam-SPSA and SD-SPSA represent an average of 10 runs

standard deviation of the 10 runs for both these algorithms gives an indication of the effect that the stochastic nature of the gradient approximation has. Although an element of stochasticity can be advantageous, it can become a hindrance if an algorithm is susceptible to it. For case study 2, Adam-SPSA had a standard deviation in the final optimum value of 41.69 MM USD across the 10 runs. In comparison, SD-SPSA had a standard deviation of 51.22 MM USD across the 10 runs. This is up to almost a 19% difference in standard deviation. That is, the optimization runs of SD-SPSA showed a 19% more spread in the final optimum values found when compared to Adam-SPSA.

The average results of Adam-SPSA and the result of GPS are competitive with regards to the final optimum value. The GPS optimization run results in a final optimum NPV value of 1,409 MM USD after 100 function evaluations. On the other hand, Adam-SPSA, on average, results in a final optimum value of 1,401 MM USD. The convergence plot of GPS exhibits a characteristic pattern of a local optimizer with improvements occurring after searching the local landscape. Given the stencil-based search of GPS, it is not uncommon to see large increases in NPV after one function evaluation. An example of this is shown in Fig. 4 where a significant jump in NPV occurs at 61 function evaluations.

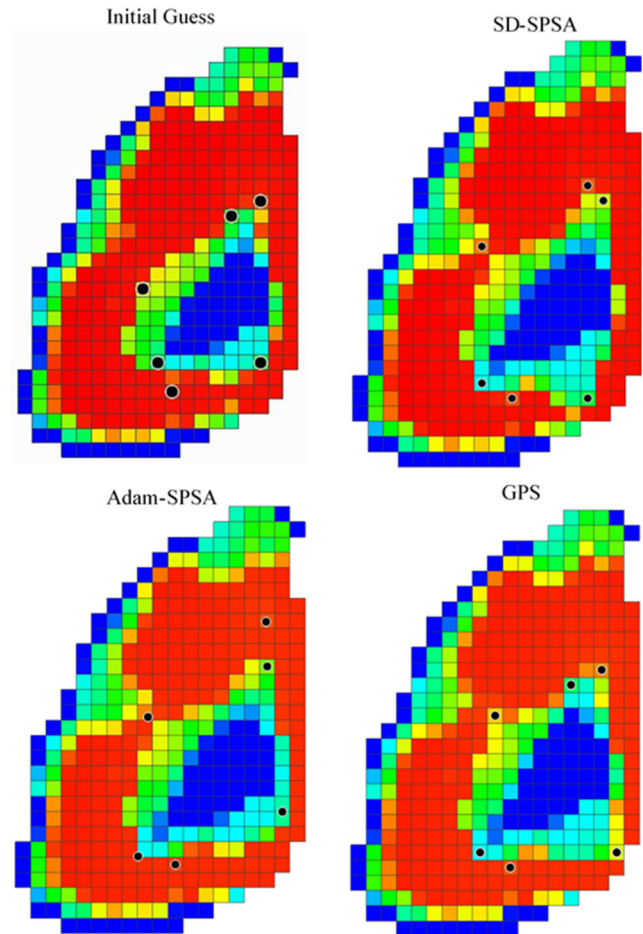
Although the one GPS run outperforms the average runs of Adam-SPSA in terms of final NPV, it should be mentioned that three of the 10 runs of Adam-SPSA outperform GPS. An example is shown in Fig. 5, which shows the best run of Adam-SPSA against the GPS run. These results show that Adam-SPSA outperforms GPS in terms of convergence speed and final optimum value. For



**Fig. 5** Convergence plot for the best run (out of 10) of Adam-SPSA (solid line) and GPS (dash-dotted line) run for the optimization of four vertical infill wells

example, GPS converges to an optimum value of 1,409 MM USD after 85 function evaluations, whilst Adam-SPSA converges to similar value (1,399 MM USD) after only 51 function evaluations. This translates to a 40% improvement in number of function evaluations to achieve a similar NPV value. This Adam-SPSA optimization run converges to a final optimum value of 1,476 MM USD after 66 function evaluations. On the other hand, after 66 function evaluations, GPS reaches an NPV value of 1,378 MM USD, which is a 7.11% reduction compared to Adam-SPSA. The convergence plots shown in Fig. 5 are characteristic of local optimizers. The staggered pattern represents the algorithms traversing the local landscape in search of an improvement in objective function value.

Figure 6 shows the final oil saturation maps for the best optimized solutions of all three algorithms. The major difference is seen when comparing the locations of the wells relative to the gas cap, where the wells were initially located (shown in top left map). Given this work investigates the use of local optimization algorithms for well placement problems, it is expected that the final solutions will be relatively close (with respect to well locations) to the initial guess. The aim is to improve on an initial guess that is based on reservoir engineering judgement to maximize the objective function. This is shown by the well locations from the solutions of the three algorithms. The wells are slightly moved away from the gas cap to produce more oil from the northern and western flanks of the reservoir. However, this needs to be balanced as a move too far north or westerly would result in a large water influx from the strong aquifer on either side. From the final oil saturations shown, the Adam-SPSA solution produces a solution that best balances



**Fig. 6** Oil saturation maps of Layer 1 at the end of the production time for the initial guess (top left) and the three best solutions from SD-SPSA (top right), Adam-SPSA (bottom left) and GPS (bottom right). Red represents high oil saturation and blue represented low oil saturations. Wells are represented by solid black circles

this as shown by the drainage of oil saturation in the western side of the reservoir model.

### 4.3 Case study 3 - 20 nonconventional wells

The third case study investigates the placement of 20 non-conventional wells in the PUNQ-S3 model. Consequently, this problem has a total of 120 decision variables. The optimization variables were normalized and bounded between 0 and 1. In this paper, we use a parameterization similar to Sayyafzadeh and Alrashdi [9]. Each nonconventional well is defined with 6 variables. The heel and toe are defined using  $x$  and  $y$  Cartesian coordinates, while the  $z$ -coordinate is defined as a percentage between the top and bottom layer. For a proposed set of  $x$ - and  $y$ -coordinates, the corresponding  $z$ -coordinates of the top and bottom layers are found using an interpolation surface. Then the proposed  $z$ -coordinate is calculated using the proposed percentage

(i.e. the decision variable) between these two z-coordinates. This helps ensure that the full perforation length of the well intersects the reservoir model in the z-direction. The wells are defined as fully perforated straight lines between the heel and toe. Lastly, this parameterization allows the nonconventional wells to be defined as vertical, slanted or horizontal.

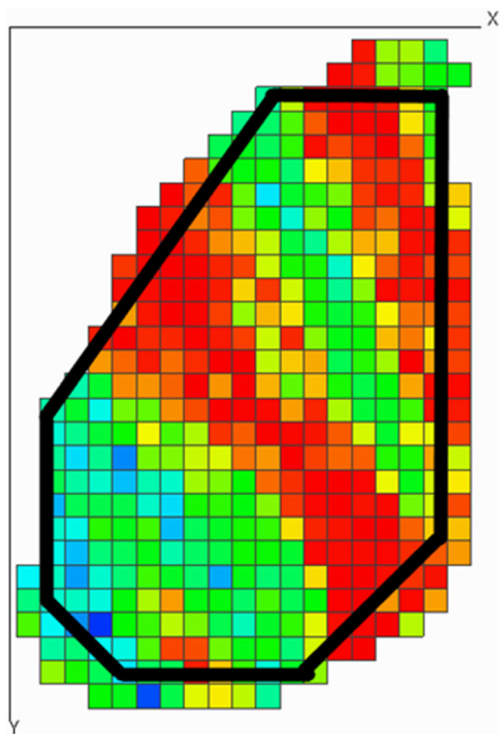
The objective is to place 20 nonconventional wells. As in case study 2, the wells are controlled by BHP with a pressure of 2900 psi (200 bars) and a maximum liquid rate of 5660 STB/day (900  $sm^3/day$ ). The economic parameters used in this case study are given in Table 1. As previously mentioned, three nonlinear constraints are considered in this case study. This includes a maximum well length of 500 meters, an inter-well constraint (in three-dimensions) of 200 meters and a reservoir polygon bound (Fig. 7) constraint.

The GPS employed is the same as that employed in case study 2, with an initial stencil size of 0.05. Adam-SPSA had parameter values of 0.1 and 0.05 for  $\omega$  and  $c$ , respectively. SD-SPSA had parameter values of 0.25 and 0.05 for  $a$  and  $c$ , respectively. In this case study, an initial guess was used based on possible reservoir engineering considerations. The wells were placed in a manner that would take advantage of the added contact with the reservoir when using nonconventional wells. However, the effect of

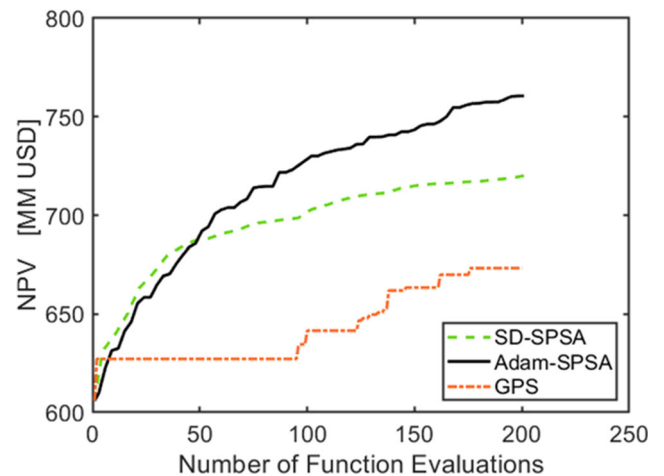
the strong aquifer needed to be considered to ensure the water breakthrough did not force wells to be shut-in. The stopping criterion was set to a maximum number of function evaluation of 200.

Figure 8 compares the performance of Adam-SPSA to GPS and SD-SPSA for case study 3. Similar to case study 2, Adam-SPSA and SD-SPSA were run with the same initial guess using 10 different seeds. On average, SD-SPSA required 203 function evaluations to converge to a final optimum NPV value of 719.9 MM USD. In comparison, Adam-SPSA required 87 function evaluations to reach a similar NPV value (721.7 MM USD). This translates to a convergence speed-up of up to 57% to reach the SD-SPSA final optimum value.

An interesting insight shown in Fig. 8 is the behaviour of Adam-SPSA and SD-SPSA in early iterations. Between 0 and 50 function evaluations SD-SPSA, on average, seems to be performing slightly better than Adam-SPSA. However, after this initial period, the convergence speed of Adam-SPSA increases, while SD-SPSA slows significantly, resulting in a noticeable difference in the average final optimums. This initial difference can be attributed to the nature in which the search directions are calculated by Adam-SPSA, where a running exponential average is used to estimate the first and second moments. As such, in order to improve the search direction, Adam-SPSA requires first-order information from a number of iterations. Once this is achieved, Adam-SPSA is able to perform significantly better based on this extracted information. This is a noticeable difference compared to case study 2 where only 8 dimensions were considered where Adam-SPSA was able to update the dimension-wise search directions more efficiently.



**Fig. 7** X-directional permeability of Layer 1 showing the reservoir polygon boundary (solid black lines) used to define the piecewise linear polynomials for boundary constraint. Red represents a high permeability value and green represents a low permeability

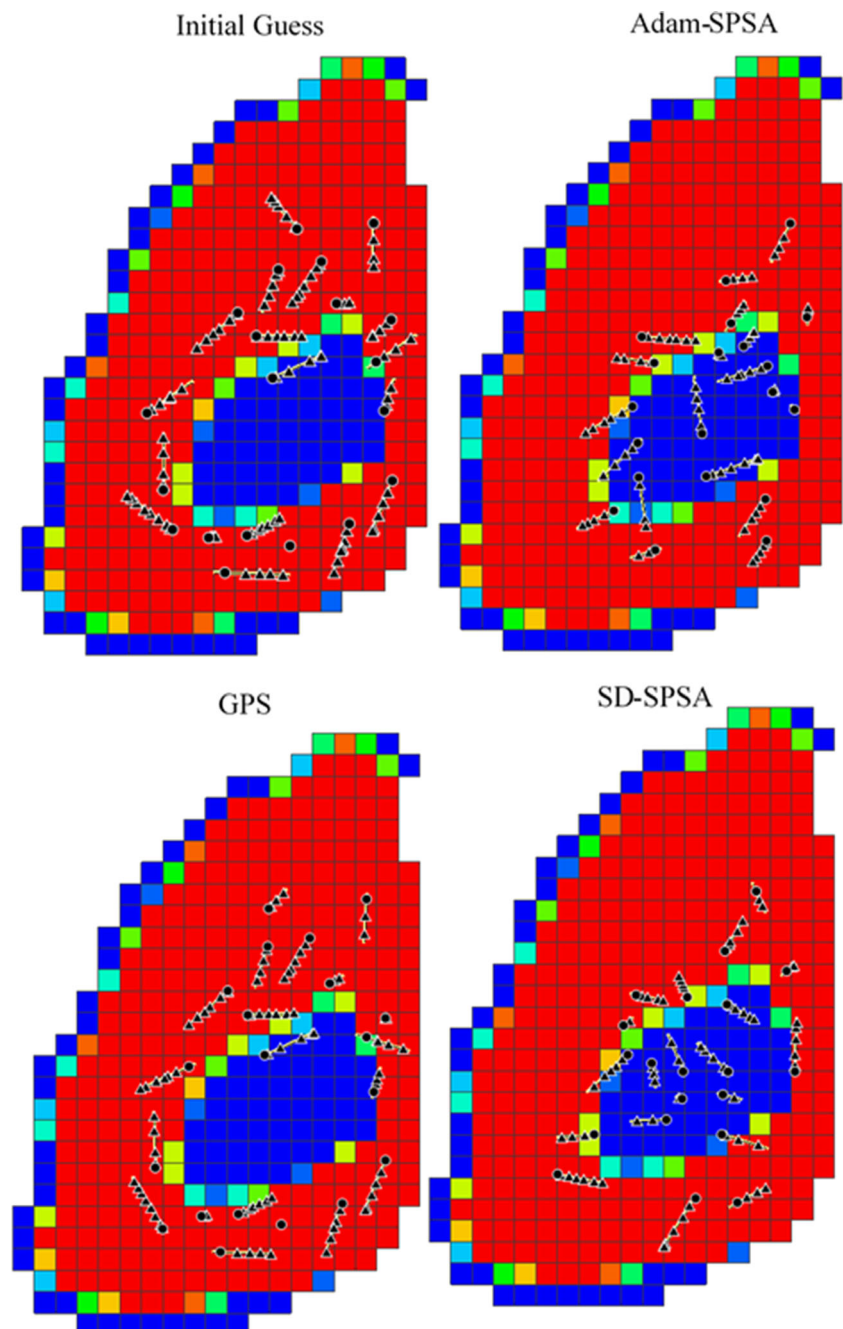


**Fig. 8** Convergence plot showing the results of Adam-SPA (solid line), SD-SPSA (dashed-dotted line) and GPS (dashed line) for the placement of 20 nonconventional wells

The large difference in performance of the best run of SD-SPSA and its average over 10 runs is the stochastic nature of the gradient calculation. As such, the standard deviation can give insights into the spread of optimization results. For case study 3, Adam-SPSA has a standard deviation in the final optimum value of 15.90 MM USD. On the other hand, SD-SPSA has a standard deviation over 10 runs of 26.38 MM USD, which represents up to 40% more spread in final optimum values. From a practical standpoint, Adam-SPSA results in a more consistent solution with respect to the final optimum NPV.

The comparison of the GPS and Adam-SPSA results is a reflection of the “curse of dimensionality” that pattern search methods are susceptible to. For a 120-dimensional problem, as in case study 3, it is computationally inefficient to traverse the local landscape using a stencil-based approach. Even when employing an incomplete polling approach, this search technique may result in prolonged periods of no improvement in function value, as shown in Fig. 8. The outperformance of GPS by Adam-SPSA (and SD-SPSA) gives insight into the advantage that a stochastic element (i.e., in gradient approximation) can

**Fig. 9** The well locations for initial guess (top left) and initial oil saturation maps with the optimal well locations found by Adam-SPSA (top right), SD-SPSA (bottom right), and GPS (bottom left) in two-dimensions. Circles represent well head location and triangles represent a connection. Red represents high oil saturation and blue represents low oil saturation



have, in addition to the simultaneous perturbation. Although these local methods are still susceptible to the same dimensionality curse, having a stochastic element gives these methods an opportunity to traverse the landscape more efficiently. This results in a more computationally efficient search and hence final optimum value. It is worth mentioning that there is potential for these advantages to extend to larger models containing finer grid cells. However, in such cases careful consideration would need to be given when selecting appropriate step-sizes and perturbation sizes. As a step-size and perturbation size too small may have an affect on the convergence of the algorithm.

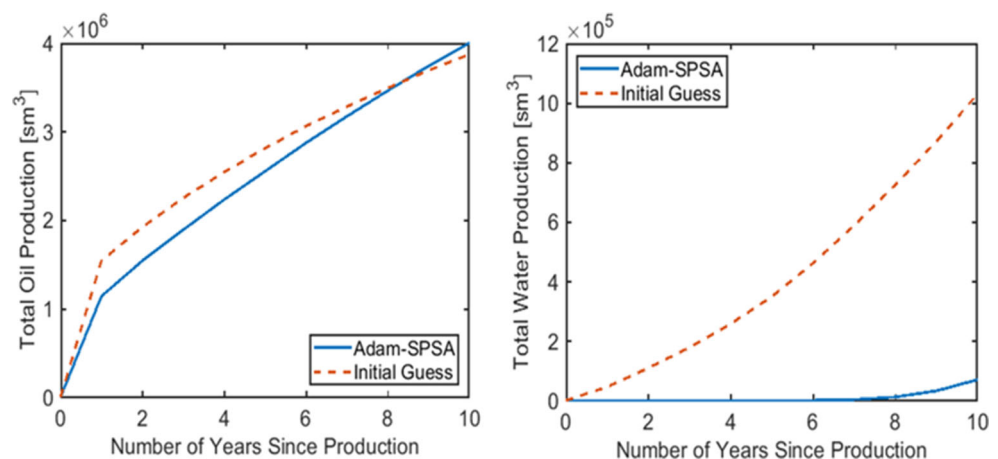
The well locations of the initial guess as well as selected optimization results for the three algorithms are presented in Fig. 9. The first observation is the similarity of the well locations from the initial guess to each of the algorithm results. This is what is expected with the use of local methods, which exploit the initial guess and progress to the closest minimum. The well locations exhibited in Fig. 9 show that placing wells closer to the gas cap assists in lowering the water production due to aquifer encroachment. However, placing too many wells in (or too close to) the gas cap, as in the SD-SPSA solution, may undermine the oil production total and as such reduce the objective function (NPV) value. Overall, the solution found by Adam-SPSA is able to balance these two aspects of water production and oil production to improve NPV over the initial guess. This gives an indication that local methods, both gradient-based and pattern-search (although to a lesser degree) type algorithms can be used to improve on initial guesses for well location optimization. Furthermore, although the well locations are similar to the initial guess, it reflects the ability for local methods to handle nonconventional well trajectory placement. In computationally constrained practical scenarios, these methods may provide suitable solutions within the budget.

Another important aspect of practical application for nonconventional wells is the ability for algorithms to

navigate the local landscape in the presence of physical field constraints. As mentioned earlier, three nonlinear constraints were considered: a maximum well length, a minimum three-dimensional inter-well distance and a reservoir polygon bound. The local methods were still able to improve on the initial guess whilst operating with the nonlinear constraint handling employed. The constraint handling technique is successful in ensuring the proposed solutions are not violating the physical constraints.

Additional insight can be obtained when reflecting on the oil and water production totals, presented in Fig. 10, for the Adam-SPSA solution presented in Fig. 9. These totals give an indication of the reasons for the differences in the objective function value, NPV, obtained from Adam-SPSA compared to the initial guess. The placement of the wells plays a pivotal role in the volume of fluids produced from a field. As previously mentioned, this solution places more wells closer to the gas cap, with a number of wells directly on top of the dome structure. By the same token, this means that the wells in this solution are further away from the strong aquifers that are present on the outer boundaries. After 10 years of production, the Adam-SPSA solution and the initial guess have similar oil production totals of  $4.00 \times 10^6 \text{sm}^3$  and  $3.87 \times 10^6 \text{sm}^3$ , respectively. However, there are more substantial differences between the water production totals. Specifically, the Adam-SPSA solution results in a water production total of  $7.02 \times 10^4 \text{sm}^3$  compared to  $1.03 \times 10^6 \text{sm}^3$  produced in the initial guess, which is more than a magnitude difference. Given there are no associated costs with gas handling, the key driver for the difference in NPV between the Adam-SPSA solution and the initial guess is the significant reduction in water production. This gives additional insight to possible field development considerations with regards to water handling costs. Although not considered in this study, further analysis of these results in practical applications may lead to the consideration of the gas production. This could be limited through additional constraints or the introduction

**Fig. 10** Oil (left) and water (right) production totals for the best Adam-SPSA (solid lines) solution compared to the initial guess (dashed line)



of gas handling costs into the objective function. In such a scenario, it is suitable to expect the solutions from the optimization will look different.

## 5 Discussion

In practical scenarios under computational constraint the use of optimization algorithms may be limited by a number of expensive reservoir simulation calls. As such, the employment of local optimization algorithms to these problems, such as well location optimization, may provide suitable improvements to proposed solutions. Often in practice an in-depth understanding of the reservoir dynamics is available. This may culminate itself through multiple possible development plans or potential infill well locations. This provides an ideal starting point for local optimization techniques to exploit and fine-tune these development plans to improve returns and highlight potential bypassed opportunities. Previous applications of local optimization algorithms have shown promise for vertical well placement problems. A limited number of studies have also shown applications for local optimization algorithms for nonconventional well placement.

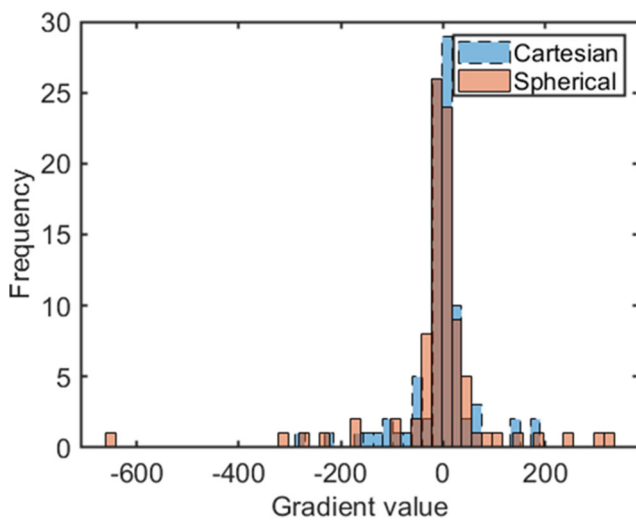
In this study, we compare a derivative-free local algorithm, GPS, and two gradient-based algorithms, Adam-SPSA and SD-SPSA (with some form of stochasticity) for well placement optimization under a limited computational budget. For problems of relatively low dimensions, as in case study 2 (8 dimensions), the performance of the algorithms was competitive as the local landscape could be exploited efficiently. However, as the dimension increases to more realistic problems, as in case study 3 (120 dimensions), the simultaneous perturbation stochastic nature of SPSA offers some advantages. The element of randomness introduced by SPSA, especially when using one approximation, increases the probability of the search covering the local landscape more efficiently.

In addition, by simultaneously perturbing all the decision variables at once the progression may occur in all dimensions. On the other hand, the deterministic nature of pattern search methods, such as GPS, makes this type of progression less likely. In addition, pattern search methods employ perturbations to each dimension individually, which is computationally inefficient for problems with a large number of decision variables. The decrease in performance of GPS from case study 2 to case study 3 reflects this inefficiency and susceptibility to the dimensionality curse. This further becomes an issue under computational constraint where an incomplete polling is more favourable, which may result in more promising directions being overlooked.

Although Adam-SPSA and SD-SPSA both employ SPSA as the gradient approximation, the first-order information extracted and utilized differs. The steepest descent framework employs the gradient as the search direction to progress the search in each dimension. However, the step-size taken is identical across all dimensions. A backtracking line-search was employed for SD-SPSA to improve the search progression by adjusting the step-size. On the other hand, Adam-SPSA, utilizes an adaptive framework that allows dimension-wise steps. This is done by estimating the first and second order moments of the gradients, which represents the accuracy of the gradient in each direction. The search direction in Adam-SPSA is the ratio of the first moment (mean) to the second moment (variance). This can be thought of as a signal-to-noise ratio (SNR). A smaller SNR indicates that there is substantial uncertainty as to whether the estimated first moment corresponds to the true gradient. As a result, the effective step in such a direction should be small. On the other hand, a large SNR indicates that the estimated first moment gives a better approximation of the true gradient as the estimated second moment would be relatively low. Consequently, this allows Adam-SPSA to adaptively change the search direction for each decision variable to improve the search. This is shown through the results for the two case studies involving both the placement of vertical wells (case study 2) and nonconventional wells (case study 3) where Adam-SPSA outperformed SD-SPSA in both convergence speed and final optimum value.

Another important consideration for gradient approximations which use simultaneous perturbation is the parameterization. A gradient approximation attempts to find the sensitivity of the objective function to each decision variable. In simultaneous perturbation methods, this is done for all decision variables often using the same perturbation size (namely,  $c$  in SPSA). However, if the decision variables do not have the same sensitivity, this will result in the less sensitive decision variables being masked by more sensitive decision variables. As a result, the gradient approximation will not be representative of the landscape.

In our nonconventional well placement studies, we trialed both spherical coordinates and Cartesian coordinates to define the well trajectory. Given the different types of variables in spherical parameterization, the sensitivity of the objective function to each will be varied. Our results, which are not shown, found that when using spherical parameterization for gradient approximations that use simultaneous perturbation were susceptible to this and performed very poorly. To further investigate this difference in performance, we calculated the gradients using finite difference for both parameterizations for 15 nonconventional wells. Figure 11 presents the histogram showing the (finite-difference) gradient values across all



**Fig. 11** Histogram of gradient values calculated using finite difference for all decision variables using a perturbation size ( $c$ ) of 0.1 comparing Cartesian and spherical parameterization

90 decision variables. The mean absolute deviation for the Cartesian parameterization histogram is 36.44, whilst it is 51.83 for the spherical parameterization histogram. This indicates that the gradient values when using spherical coordinates exhibit more spread when using the same perturbation size. This additional sensitivity can be attributed to the dependence of the location of the toe to the location of the heel in spherical parameterization. When a coordinate of the heel is perturbed, the physical location of the toe will also move even though the values for the decision variables related to the toe ( $\phi$ ,  $\theta$ ,  $L$ ) are unchanged. This gives insight into the effect that different types of parameterization can have on gradient approximations.

Although the performance of simultaneous perturbation gradients improve when utilizing a Cartesian parameterization, an additional nonlinear constraint for well length needed to be considered. In addition, it is more difficult to put constraints on the dogleg severity when using this parameterization. This could be the reason why spherical parameterization is popular in the literature as population-based methods do not require a gradient approximation.

The practicality of optimized solutions is dependent on their feasibility when physical constraints are present. This further complicates the optimization problem, especially for nonconventional wells where we considered three practical field constraints. These included a minimum inter-well distance, a maximum well length and a reservoir polygon bound. These constraints were useful in obtaining final optimum solutions that were plausible and in line with general engineering knowledge (e.g. no intersecting wells). Similar constraint handling techniques applied in this work could be extended to deal with practical constraints surrounding a minimum distance between a well and a fault.

In this case, the fault would need to be represented by a plane in 3-dimensional space.

Furthermore, the role these constraints play when approximating gradients and their effects on algorithm performance was investigated. Although not extensive, the preliminary results indicated that there is an ideal amount of (forward and backward) perturbation correction needed during gradient approximation. The correction needs to keep the integrity of the underlying direction, yet ensure the perturbations fulfil certain constraints. There is potential to extend this investigation further by comprehensively studying the available constraint handling techniques and their effects on gradient approximation and algorithm performance.

Also, the consideration of geological uncertainty in gradient-based methods is an important extension. One may extend the proposed method by simply using the expected NPV across a number of realizations as the objective function. Other more efficient, but less accurate, approaches may approximate a gradient using an average of gradient approximations which utilize stochastically selected realizations. This requires additional research to investigate the implications of such techniques on convergence.

## 6 Concluding remarks

This study compared the proposed algorithm (Adam-SPSA) with the conventional steepest descent SPSA (SD-SPSA) and a derivative-free pattern search algorithm (GPS) for constrained well placement optimization. The results presented showed the successful application of local optimization algorithms to well placement optimization, including vertical wells and nonconventional wells. These algorithms leveraged local exploitation to improve the objective function value from an initial guess. The gradient-based methods (Adam-SPSA and SD-SPSA) performed effectively in both the low-dimensional and high-dimensional case studies by significantly improving on the initial objective function value. However, the derivative-free method (GPS) was not competitive in the high-dimensional case. This can be attributed to the simultaneous perturbation used in the gradient-based methods, which allows for the progression in all dimensions during one step. When comparing the gradient-based methods, Adam-SPSA consistently outperformed SD-SPSA in both case studies investigated. This can be ascribed to the adaptive nature of the search direction, which incorporates estimates of the first and second moment. This allows the search direction to be calculated in a dimension-wise manner leading to more suitable progression steps for each decision variable. Lastly, the decision variables must have a similar sensitivity to the objective

function for the parameterization employed. For nonconventional wells, the Cartesian parameterization showed lower sensitivity compared to the spherical parameterization.

**Acknowledgements** The authors would like to thank Steve Begg for his support and fruitful discussion. The authors would also like to thank David Echeverría Ciaurri for his help with the code relating to the inter-well constraint. The authors are also grateful to Rock Flow Dynamics for providing licenses for tNavigator. This work was supported with supercomputing resources provided by the Phoenix HPC service at the University of Adelaide.

**Funding** Open Access funding enabled and organized by CAUL and its Member Institutions.

## Declarations

**Conflict of interest** The authors declare that they have no conflict of interest.

**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

- Ahmad, A., El-Shafie, A., Razali, S.F.M., Mohamad, Z.S.: Reservoir optimization in water resources: a review. *Water Resour. Manag.* **28**(11), 3391–3405 (2014)
- Cameron, D.A., Durlofsky, L.J.: Optimization of well placement, CO<sub>2</sub> injection rates, and brine cycling for geological carbon sequestration. *Int. J. Greenhouse Gas Control* **10**, 100–112 (2012). <https://doi.org/10.1016/j.ijggc.2012.06.003>, <https://www.sciencedirect.com/science/article/pii/S1750583612001296>
- Aroui, Y., Sayyafzadeh, M.: An accelerated gradient algorithm for well control optimization. *J. Pet. Sci. Eng.* **190**, 106872 (2020). <https://doi.org/10.1016/j.petrol.2019.106872>, <http://www.sciencedirect.com/science/article/pii/S0920410519312884>
- Alrashdi, Z., Sayyafzadeh, M.: (+) evolution strategy algorithm in well placement, trajectory, control and joint optimisation. *J. Pet. Sci. Eng.* **177**, 1042–1058 (2019). <https://doi.org/10.1016/j.petrol.2019.02.047>, <http://www.sciencedirect.com/science/article/pii/S0920410519301846>
- Sayyafzadeh, M.: Reducing the computation time of well placement optimisation problems using self-adaptive metamodelling. *J. Pet. Sci. Eng.* **151**, 143–158 (2017)
- Bittencourt, A.C., Horne, R.N.: Reservoir development and design optimization. In: SPE Annual Technical Conference and Exhibition. Society of Petroleum Engineers (1997). <https://doi.org/10.2118/38895-MS>
- Yeten, B., Durlofsky, L.J., Aziz, K.: Optimization of nonconventional well type, location, and trajectory. *SPE J.* **8**(03), 200–210 (2003)
- Onwunalu, J.E., Durlofsky, L.J.: Application of a particle swarm optimization algorithm for determining optimum well location and type. *Comput. Geosci.* **14**(1), 183–198 (2010). <https://doi.org/10.1007/s10596-009-9142-1>
- Sayyafzadeh, M., Alrashdi, Z.: 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> (2019)
- Bouzarkouna, Z., Ding, D.Y., Auger, A.: Well placement optimization with the covariance matrix adaptation evolution strategy and meta-models. *Comput. Geosci.* **16**(1), 75–92 (2012)
- Martí, R.: *Multi-start Methods*, pp. 355–368. Springer (2003)
- O'Donoghue, B., Candàs, E.: Adaptive restart for accelerated gradient schemes. *Found. Comput. Math.* **15**(3), 715–732 (2015). <https://doi.org/10.1007/s10208-013-9150-3>
- Bellout, M.C., EcheverríaCiaurri, D., Durlofsky, L.J., Foss, B., Kleppe, J.: Joint optimization of oil well placement and controls. *Comput. Geosci.* **16**(4), 1061–1079 (2012). <https://doi.org/10.1007/s10596-012-9303-5>
- Isebor, O.J., EcheverríaCiaurri, D., Durlofsky, L.J.: Generalized field-development optimization with derivative-free procedures. *SPE J.* 891–908. <https://doi.org/10.2118/163631-PA> (2014)
- Wang, H., Ciaurri, D.E., Durlofsky, L.J.: Use of retrospective optimization for placement of oil wells under uncertainty. In: Proceedings of the 2010 Winter Simulation Conference, pp. 1750–1757 (2010)
- Wang, H., Echeverría-Ciaurri, D., Durlofsky, L., Cominelli, A.: Optimal well placement under uncertainty using a retrospective optimization framework. *SPE J.* **17**(01), 112–121 (2012). <https://doi.org/10.2118/141950-PA>
- Jansen, J.D.: Adjoint-based optimization of multi-phase flow through porous media - a review. *Comput. Fluids* **46**(1), 40–51 (2011). <https://doi.org/10.1016/j.compfluid.2010.09.039>, <http://www.sciencedirect.com/science/article/pii/S0045793010002677>
- Sarma, P., Chen, W.H.: Efficient well placement optimization with gradient-based algorithms and adjoint models. In: Intelligent Energy Conference and Exhibition (2008)
- Wang, C., Li, G., Reynolds, A.C.: Optimal well placement for production optimization. In: Eastern Regional Meeting, p. 5. Society of Petroleum Engineers (2007). <https://doi.org/10.2118/111154-MS>
- Zandvliet, M., Handels, M., van Essen, G., Brouwer, R., Jansen, J.-D.: Adjoint-based well-placement optimization under production constraints. *J. Pet. Sci. Eng.* <https://doi.org/10.2118/105797-PA> (2008)
- Bangerth, W., Klie, H., Wheeler, M.F., Stoffa, P.L., Sen, M.K.: On optimization algorithms for the reservoir oil well placement problem. *Comput. Geosci.* **10**(3), 303–319 (2006). <https://doi.org/10.1007/s10596-006-9025-7>
- Jesmani, M., Jafarpour, B., Bellout, M.C., Hanea, R.G., Foss, B.: Application of simultaneous perturbation stochastic approximation to well placement optimization under uncertainty. In: ECMOR XV-15th European Conference on the Mathematics of Oil Recovery, pp. cp-494–00133. European Association of Geoscientists & Engineers (2016)
- Vlemmix, S., Joosten, G., GerardJP., Brouwer, R., Jansen, J.-D.: Adjoint-based well trajectory optimization. In: EUROPEC/EAGE Conference and Exhibition. Society of Petroleum Engineers (2009)
- Volkov, O., Bellout, M.C.: Gradient-based constrained well placement optimization. *J. Pet. Sci. Eng.* **171**, 1052–1066 (2018). <https://doi.org/10.1016/j.petrol.2018.08.033>, <http://www.sciencedirect.com/science/article/pii/S0920410518306995>



25. Leeuwenburgh, O., Egberts, P.J., P. Abbink, O.A.: Ensemble methods for reservoir life-cycle optimization and well placement. In: SPE/DGS Saudi Arabia Section Technical Symposium and Exhibition, p. 8. Society of Petroleum Engineers (2010). <https://doi.org/10.2118/136916-MS>
26. Li, L., Jafarpour, B., Mohammad-Khaninezhad, M.R.: A simultaneous perturbation stochastic approximation algorithm for coupled well placement and control optimization under geologic uncertainty. *Comput. Geosci.* **17**(1), 167–188 (2013). <https://doi.org/10.1007/s10596-012-9323-1>
27. Guyaguler, B., Horne, R.N.: Uncertainty assessment of well placement optimization. In: SPE Annual Technical Conference and Exhibition. Society of Petroleum Engineers (2001)
28. Ding, Y.: Optimization of well placement using evolutionary methods. In: Europec/EAGE Conference and Exhibition. Society of Petroleum Engineers (2008)
29. Forouzanfar, F., Reynolds, A.C.: Well-placement optimization using a derivative-free method. *J. Pet. Sci. Eng.* **109**, 96–116 (2013). <https://doi.org/10.1016/j.petrol.2013.07.009>, <http://www.sciencedirect.com/science/article/pii/S0920410513001952>
30. Emerick, A.A., Silva, B., Almeida, L.F., Szwarcman, D., Pacheco, M.A.C., Vellasco, M.M.B.R.: Well placement optimization using a genetic algorithm with nonlinear constraints. In: SPE Reservoir Simulation Symposium. Society of Petroleum Engineers (2009)
31. Wu, Y., Schuster, M., Chen, Z., Le, Q.V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K.: Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv:1609.08144* (2016)
32. Gregor, K., Danihelka, I., Graves, A., Rezende, D.J., Wierstra, D.: Draw: A recurrent neural network for image generation. *arXiv:1502.04623* (2015)
33. Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J., Wang, Z.: Photo-realistic single image super-resolution using a generative adversarial network. In: Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, pp. 4681–4690 (2017)
34. Spall, J.C.: Multivariate stochastic approximation using a simultaneous perturbation gradient approximation. *IEEE Trans. Autom. Control* **37**(3), 332–341 (1992)
35. Sadegh, P., Spall, J.C.: Optimal random perturbations for stochastic approximation using a simultaneous perturbation gradient approximation. *IEEE Trans. Autom. Control.* **43**(10), 1480–1484 (1998)
36. Spall, J.C.: Implementation of the simultaneous perturbation algorithm for stochastic optimization. *IEEE Trans. Aerospace. Electron. Syst.* **34**(3), 817–823 (1998)
37. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. *arXiv:1412.6980* (2014)
38. Lefebvre, M. *Applied Probability and Statistics*, 1st edn. Springer, New York (2006)
39. MathWorks. *Matlab* (2018)
40. Floris, F.J.T., Bush, M.D., Cuypers, M., Roggero, F., Syversveen, A.-R.: Methods for quantifying the uncertainty of production forecasts: a comparative study. *Pet. Geosci.* **7**(S), S87–S96 (2001). <https://doi.org/10.1144/petgeo.7.S.S87>, <https://www.earthdoc.org/content/journals/10.1144/petgeo.7.S.S87>

**Publisher's note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.