

Well Control Optimization using Derivative-Free Algorithms and a Multiscale Approach

Xiang Wang · Ronald D. Haynes ·
Yanfeng He · Qihong Feng

Received: date / Accepted: date

Abstract Smart well technologies, which allow remote control of well and production processes, make the problem of determining optimal control strategies a timely and valuable pursuit. The large number of well rates for each control step make the optimization problem difficult and present a high risk of achieving a suboptimal solution. Moreover, the optimal number of adjustments is not known a priori. Adjusting well controls too frequently will increase unnecessary well management and operation cost, and an excessively low number of control adjustments may not be enough to obtain a good yield. In this paper, we explore the capability of three derivative-free algorithms and a multiscale regularization framework for well control optimization over the life of an oil reservoir. The derivative-free algorithms chosen include generalized pattern search (GPS), particle swarm optimization (PSO) and covariance matrix adaptation evolution strategy (CMA-ES). These algorithms, which cover a variety of search strategies (global/local search, stochastic/deterministic search), are chosen due to their robustness and easy parallelization. Although these algorithms have been used extensively in the reservoir development optimization literature, for the first time we thoroughly explore how these algorithms perform when hybridized within a multiscale regularization

X. Wang
School of Petroleum Engineering, Changzhou University, Changzhou, Jiangsu, China
Tel.: +86.132.8085.6026
E-mail: xiangwang@cczu.edu.cn

R. D. Haynes
Department of Mathematics & Statistics, Memorial University, St. John's, NL, Canada
Tel.: +1.709.864.8825
Fax: +1.709.864.3010
E-mail: rhaynes@mun.ca

Y. He
School of Petroleum Engineering, Changzhou University, Changzhou, Jiangsu, China
E-mail: heyanfeng@cczu.edu.cn

Q. Feng
School of Petroleum Engineering, China University of Petroleum (Huadong), Qingdao, Shandong, China
E-mail: fengqihong@126.com

framework. Starting with a reasonably small number of control steps, the control intervals are subsequently refined during the optimization. Results for the experiments studied indicate that CMA-ES performs best among the three algorithms in solving both small and large scale problems. When hybridized with a multiscale regularization approach, the ability to find the optimal solution is further enhanced, with the performance of GPS improving the most. Topics affecting the performance of the multiscale approach are discussed in this paper, including the effect of control frequency on the well control problem. The parameter settings for GPS, PSO, and CMA-ES, within the multiscale approach are considered.

Keywords Well Control · Production Optimization · Derivative-Free Algorithms · Multiscale Approach

1 Introduction

Determining the well production and injection rates is of paramount importance in modern reservoir development. The decision is difficult since the optimal rates depend on the heterogeneity of the rock and liquids, the well placements and other parameters. Indeed, these properties and input parameters are coupled in a highly nonlinear fashion. Moreover, the optimal production and injection rates are usually not constant throughout the life cycle of reservoir. The oil saturation distribution changes during the well injection and production processes. This will then affect the optimal production and injection rate for each well.

Well control planning can be formulated as an optimization problem, using economic or cumulative oil production as the objective function. The well rates or bottom hole pressures at different times are the optimization variables. Many optimization algorithms have been investigated to solve such problems. These algorithms can be broadly placed in two categories: derivative-based algorithms and derivative-free algorithms.

Derivative-based or gradient-based algorithms, take advantage of the gradient of the objective function to guide their search. This type of algorithm, commonly used in well control optimization, includes steepest ascent, conjugate gradient, and sequential quadratic programming methods [48, 66, 42]. Gradients of the objective function may be calculated by using an adjoint equation. This is an invasive approach, requiring a detailed knowledge of mathematics inside the reservoir simulator [13, 10, 4, 70, 8]. Other ways to approximate the gradients include finite difference perturbation [66, 71], or the simultaneous perturbation stochastic approximation [66, 43]. These algorithms assume a certain degree of smoothness of the objective function with respect to the optimization variables. Derivative-based algorithms are potentially very quick to converge but sometimes fall into local optimal.

Derivative-free algorithms can be subdivided into local search methods and global search methods. Local derivative-free algorithms include generalized pattern search (GPS) [35], mesh adaptive direct search (MADS) [38, 27], Hooke-Jeeves direct search (HJDS) [20], ensemble-based optimization (EnOpt) [14, 15, 19, 49, 11, 12], covariance matrix adaptation evolution strategy (CMA-ES) [7, 45], and so on. These methods have strong ability to find accurate optima in a local space, but may face some difficulties in finding global optima, especially when a good initial

guess is not available. Global derivative-free algorithms search through the entire space and provide techniques to avoid being trapped in local optima. Examples of global search algorithms include genetic algorithms (GAs) [2], particle swarm optimization (PSO) [37], and differential evolution (DE) [60,9]. Although these algorithms are robust and easy to use, they often require more function evaluations than local search and derivative-based algorithms to converge. However, most of these algorithms parallelize naturally and easily, which make their efficiency satisfactory [16]. Recently, some hybridization of these techniques such as PSO-MADS [35,37,34], multilevel coordinate search (MCS) [36], etc., have been developed and applied in well placement and/or well control optimization problems. These methods provide global search capabilities in addition to local convergence. The performance of MCS for well placement and control optimization is discussed in [67].

The optimization algorithms mentioned above can be further classified as either stochastic or deterministic. Stochastic methods use information from the previous iterations and a random component to generate new search points. The random component of the algorithm makes it more likely to avoid local optima, but it may also make the control of solution quality difficult, especially with a limited computational budget. The stochastic algorithms from the above list are MADS, CMA-ES, GA, PSO, EnOpt and DE. Deterministic methods have no random element. For a given problem, deterministic methods will give the same results for each trial, assuming the same initial guess. GPS, HJDS, and MCS are examples of deterministic algorithms.

All the above mentioned algorithms have been used in well control optimization and/or well placement optimization problems. The performance of the algorithms are problem-dependent. Some of the algorithms, like GPS (1960s), GA (1960s), and PSO (1990s), have been around for decades, and have been used in petroleum industrial problems for a relatively long time. People have accumulated a great deal of experience through case studies. CMA-ES, developed in the 2000s [30], was first used in petroleum related optimization problems only in 2012 [7]. Though CMA-ES performed very well in well placement and control optimization problems [56,53,7], to date the application of CMA-ES to well control optimization is still limited.

Although many optimization algorithms have been used, well control optimization is still a challenging problem and an active area of research. The number of optimization variables is large in many real-life scenarios. The required number of function evaluations will rise sharply with the increase in the number of variables. A single function evaluation requires one reservoir simulation which is often very demanding in terms of CPU time. The non-convex, non-smooth and multimodal objective surface further increases the optimization difficulty. To address these challenges, researchers have made many valuable contributions. In addition to the above optimization algorithm studies, several model predictive control and boundary control approaches have been applied to well production control, see [32,58]. Developing surrogate models by model reduction techniques which replace high-fidelity computational models (usually the numerical simulator) during the optimization, can save computational costs by sacrificing some model accuracy, see [47,46,59,1,63,41,26].

Furthermore, it is difficult to (automatically or a priori) choose a reasonable frequency for well control; an excessively low number of control adjustments may

not truly optimize oil recovery. Adjusting each well control too frequently imposes an unrealistic control burden on operations, increasing the total well management cost. Moreover, imposing a high number of control adjustments increases the complexity of the optimization problem so much that there is a high risk of optimization algorithms becoming trapped at local optima and hence missing the optimal strategy [57]. The control problem can become ill-posed, resulting in a “nervous” control strategy. In that case many different control strategies will result in (nearly) identical objective function values [21]. Multiscale regularization approaches have been developed to address these problems. The main idea of the multiscale approach is to start the optimization process with a very coarse control frequency (and thus, with a small number of control variables) and refine the number of control adjustments successively. The solution at the coarse-scale is used as the initial guess of controls for the next finer scale optimization [44,57,49]. The approach is called “multiscale” since the temporal scale of each control time step can range from years to days during the well production optimization problem. It is notable that this definition of multiscale is somewhat inconsistent with how it is used in other application areas. For example, multiscale refers to macroscopic, mesoscopic, and microscopic scales in modeling flow in porous media.

In the past, a number of multiscale regularization approaches have been investigated for well control optimization. The successive-splitting multiscale approach (SS-MS), also known as the ordinary multiscale approach [44,57], is the simplest strategy. It splits one control step into two equal control steps to construct a finer control problem to optimize. It is worth noting that in the work of Lien et al., they also refine in the spatial direction by initially grouping the injector and producer segments and then gradually splitting in space. The refinement indicator multiscale approach (RI-MS), was also proposed by Lien et al. [44]. RI-MS uses the magnitude of the components of the gradient of the objective to determine refinement indicators. The algorithm progressively increases the number of variables using the refinement indicators to choose the most-efficient partitioning of the current control steps to increase the value of the objective function. Hierarchical multiscale approach (Hi-MS), was devised and analysed by Oliveira et al. [49]. It is similar to SS-MS, but with an additional merge operation. The merge operation can merge existing control steps by considering the difference between well controls at two consecutive control steps and/or the gradient of the objective function with respect to the well controls. Most recently, Oliveira et al. proposed RHi-MS. The difference between RHi-MS and Hi-MS is that RHi-MS determines the split based on the refinement indicators.

The multiscale approach is a framework, which must be combined with an optimization algorithm to solve the well control optimization problem. Lien et al. combine SS-MS with the steepest-ascent method (a gradient-based optimization algorithm) to solve well control optimization problem. Shuai et al. [57] tested the performance of SS-MS when combined with EnOpt and BOBYQA (two derivative-free algorithms). Oliveira et al. [49,50] tested the performance of SS-MS, RI-MS, and Hi-MS when combined with the steepest-ascent method and EnOpt. Though Shuai et al. [57] and Oliveira et al. [49] tested the performance of multiscale approaches with derivative-free algorithms EnOpt and BOBYQA, we still know relatively little about how multiscale approaches and common derivative-free algorithm perform for well control optimization problems. The specific choice of the

optimizer affects the performance of the multiscale framework significantly and needs further investigation.

Given the prevalence of derivative-free algorithms for well control optimization and the need for a multiscale approach for problems with a large number of control variables, we consider the natural marriage of the two philosophies. For the multiscale approach, we use SS-MS since it is derivative-free, straightforward and easy to implement. Combining SS-MS with a derivative-free optimization algorithm avoids the gradient calculation or estimation. In this paper, we first discuss the effects of control frequency on well control optimization, including the effects on the production and the effects on the control strategy. Then we combine the SS-MS with three typical derivative-free optimization algorithms for the well control problem. We choose a deterministic local search method – generalized pattern search (GPS), a stochastic local search method – covariance matrix adaptation evolution strategy (CMA-ES), and a stochastic global search method – particle swarm optimization (PSO). The performance of each algorithm is analyzed using several reservoir models as test cases. Although GPS, PSO, and CMA-ES are widely used in petroleum engineering and many other areas, to the best of our knowledge there has been no attempt to combine these methods with a multiscale approach to solve the well control optimization problem. The influence of the two parameters, the split factor and the initial number of control steps, on the performance of SS-MS is also investigated. Moreover, we also discuss the performance of our SS-MS derivative-free approach in a parallel computing environment, a study which appears to be missing from the literature.

This paper is structured as follows. Section 2 describes the well control problem formulation. Section 3 gives detailed description of our multiscale approach and the framework of combining the multiscale approach with the three optimization algorithms. In Section 4, we detail the computational methodology and describe the reservoir models used in this paper. Section 5, we present the results and discussion for the experiments. Finally, in Section 6, we provide a summary and conclusions of this work.

2 The well control optimization problem

In this section, we describe the well control optimization problem, including the objective function of interest, the control variables and the imposed constraints.

The typical objective function associated with a well control problem evaluates an economic model and takes into account different costs such as the price of oil, the costs of the injection and the production of water. Another alternative is to use the cumulative oil production or the barrel of oil equivalent [7]. In this work, the objective function of interest is the net present value (NPV) of a time series of cash flows. For the three-phase flow of oil and water, the NPV is defined by

$$\text{NPV}(\mathbf{u}) = \sum_{k=1}^{N_t} \left[\frac{\Delta t_k}{(1+b)^{\frac{t_k}{\tau}}} \left(r_{gp} \mathbf{q}_{gp}^k(\mathbf{u}) + r_{op} \mathbf{q}_{op}^k(\mathbf{u}) - c_{wp} \mathbf{q}_{wp}^k(\mathbf{u}) - c_{wi} \mathbf{q}_{wi}^k(\mathbf{u}) \right) \right], \quad (1)$$

where \mathbf{u} is set of control variables during the reservoir's lifetime; \mathbf{q}_{gp}^k , \mathbf{q}_{op}^k and \mathbf{q}_{wp}^k , respectively, denote the average gas rate, the average oil rate and the average water rate for the n th time step; \mathbf{q}_{wi}^k is the average water-injection rate for the k th time step; r_{gp} and r_{op} are the gas and oil revenue; c_{wp} is the disposal cost of produced water; c_{wi} is the water injection cost; N_t is total number of time steps; t_k is the time at the end of k th time step; and Δt_k is k th time step size. The quantity τ provides the appropriate normalization for t_k , e.g., $\tau = 365$ days. The quantity b is the fractional discount rate.

The optimization variables \mathbf{u} could contain the well bottom hole pressures or the well liquid rates. In this work, we control wells by specifying the liquid rates. The vector \mathbf{u} is an N_u -dimensional column vector, where N_u is the total number of well controls. Assuming each well has the same frequency of control steps, then $N_u = N_t \cdot N_w$, where N_w is the total number of wells and N_t is the total number of control adjustments.

Well control optimization during the reservoir life cycle can be expressed as the following mathematical problem:

$$\max \quad \text{NPV}(\mathbf{u}), \quad (2)$$

$$\text{subject to} \quad \mathbf{u}_{lb} \leq \mathbf{u} \leq \mathbf{u}_{ub}, \quad (3)$$

$$\mathbf{c}(\mathbf{u}) \leq 0, \quad (4)$$

$$\mathbf{e}(\mathbf{u}) = 0, \quad (5)$$

where $\text{NPV}(\mathbf{u})$ is the objective function given by equation (1). And, in order, equations (3–5) are the bound, inequality, and equality constraints (if any) imposed on the problem.

The bound constraints define the minimum and maximum values for control variables (i.e. well rates for problems in this paper). The inequality constraints include the limits on well bottom hole pressures, well water cut limits, facility constraints such as field-level production/injection limits, etc. The equality constraints usually are the flow equation constraints, which ensure the reservoir flow dynamics satisfy the governing reservoir flow equations.

Though there are large number and various types of constraints, we assume only bound constraints are imposed explicitly for our well control problems. This makes sense because: 1) the flow equation constraints are always satisfied since we use a reservoir simulator to calculate the objective function value. 2) some complex constraints, for example, the well water cut limits may be naturally enforced. If the water cut of a well is very high, the NPV will generally not be high. 3) the well bottom hole pressure constraints are imposed implicitly in our problems. Using the keywords setting in the reservoir simulation data file, if the bottom hole pressure is over the maximum limit for an injector, or under the minimum limit for a producer, the well will be controlled by the bottom hole pressure automatically, until the well pressure constraints are satisfied.

We use Eclipse 100 [25], a commercial reservoir simulation software from Schlumberger Ltd., to calculate the relevant time-dependent production information for all experiments in this paper.

For an optimization problem, the size of the search space decreases as the number of constraints increase, however, imposing constraints can make the search space more complicated and the optimization problem more difficult to solve. How

to balance the size and complexity of search space with the realism of the problem by choosing appropriate constraints needs further research, but is outside the scope of this paper.

3 A multiscale framework

In production optimization, specifying the frequency of needed well control adjustment is a challenge. On one hand, a high frequency adjustment of control parameters imposes unrealistic burden on operations, leading to an increase in well management costs. In addition, from an optimization perspective a high frequency of control adjustments implies an explosion in the number of control variables, requiring a great amount of computation and time to get an optimal solution. This may be especially true for derivative-free algorithms, which may need many more function evaluations than gradient-based algorithms. Many degrees of freedom also increase the risk of an optimization algorithm being trapped in a local optimum. On the other hand, imposing too few control adjustments may not truly optimize oil recovery.

Multiscale regularization provides a way to address the complexity of the optimization problem with a large number of control adjustments and also avoids the need to guess an appropriate number of control adjustment steps a priori. The multiscale approach starts with a coarse number of control steps and successively increases the frequency of control adjustments using the coarse-scale solution as the initial guess for the next finer scale optimization [44,49,57]. The refinement process is terminated when a specified stopping criteria is satisfied. For example, a maximum number of control adjustments or a minimum allowable change in the objective function could be imposed.

To the best of our knowledge, three related multiscale approaches have been investigated for the well control optimization problem. The first approach, first seen in [44], is referred to as ordinary multiscale or successive-splitting multiscale [57]. The optimization starts with a coarse number of control adjustments and subsequently splits each control step into two new ones at every iteration. The second optimization strategy, also proposed by Lien et. al. [44], uses the magnitude of the components of the gradient of the objective to determine refinement indicators. The algorithm progressively increases the number of variables using the refinement indicators to choose the most-efficient partitioning of the current control steps to increase the value of the objective function. The third approach is called the hierarchical multiscale method [49,50]. It is similar to the ordinary multiscale approach in [44,57], but the algorithm can also merge existing control steps by considering the difference between well controls at two consecutive control steps and the gradient of the objective function with respect to the well controls.

The goal of the present work is to explore the feasibility of improving the performance of derivative-free algorithms in solving large scale well control optimization problems by using a multiscale approach. We choose the successive-splitting multiscale approach because the other two methods, the refinement indicator multiscale approach and the hierarchical multiscale approach, require gradient information of the objective function – information we do not assume is available. Furthermore, one recent study compared the sophisticated refinement indicator and hierarchical

multiscale approaches with the simpler successive-splitting approach and showed similar performance [49].

Before we introduce our modified approach, we take a look at the original successive-splitting multiscale approach. As Shuai et al. describe in [57], the successive-splitting multiscale algorithm generally loops over the following steps:

- 1) **INITIALIZATION** One control step for each well (initial steps $n_0 = 1$); The number of unknowns is equal to the number of wells; Initial guesses of control are assigned to each well.
- 2) **OPTIMIZATION** Solve the well control optimization problem using an optimization algorithm.
- 3) **SPLITTING** Split each control step into two steps of equal length (split factor $n_s = 2$); This doubles the number of control variables; Use the solution from step 2) as the initial well control; Go to step 2).

Our experience indicates the efficacy of a multiscale approach depends on two key parameters: the number of control steps for each well at the beginning of the optimization (i.e. the number of initial steps n_0) and the multiplicative increase in the number of control steps at every iteration (i.e. the split factor n_s). As mentioned, the successive-splitting multiscale approach used in [57] starts the optimization procedure by finding the optimal control strategy assuming one control step ($n_0 = 1$). Subsequent optimizations split the number of control steps by a fixed split factor $n_s = 2$. We show that this configuration of the two parameters is not always the most efficient configuration. On one hand, the optimal well control strategies with a very coarse parametrization may be dramatically different than with a fine parametrization (or large number of control adjustments). Hence the solution found by a very coarse parametrization is not useful as an initial guess to find the optimal fine parametrization or will require many successive splittings. This observation has to be balanced with the realization and motivation that the problem with a large number of control adjustments is too difficult to solve immediately. The split factor is the key to balance the difficulty of optimization problem at each scale and the total number of scales. With a higher split factor, less scales are needed to reach the maximum number of control steps. We will show this is sometime more efficient.

Based on the above, in addition to coupling the multiscale approach with commonly used derivative free algorithms, we consider the effect of the choice of the initial number of control number steps n_0 and the choice of n_s in the overall efficiency of the multiscale optimization process. In our modified multiscale approach, we leave the choice of the initial number of steps and the choice of the split factor to the user. We start the multiscale algorithm with a reasonably small value of n_0 – the initial number of control steps, and then find the associated optimal controls. After maximizing objective function on the basis of the initial control steps, we split each control step into several steps depending on the split factor n_s as

$$\mathbf{x}_{i+1}(n) = \mathbf{x}_{i*}(\lceil n/n_s \rceil), \quad n = 1, 2, \dots, N_v \times n_s, \quad (6)$$

where $\mathbf{x}_{i+1}(n)$ is the n th variable in the initial guess for the $(i + 1)$ th scale; $\mathbf{x}_{i*}(\lceil n/n_s \rceil)$ is the $\lceil n/n_s \rceil$ th variable in the optimum solution for the i th scale, $\lceil \cdot \rceil$ is the ceiling function and N_v is the total number of variables for the i th scale. With this formula, the total number of variables for the $(i + 1)$ th scale becomes

$N_w \times n_s$, and every n_s variables for the $(i + 1)$ th scale use the optimum solution of the i th scale. This process of splitting the control steps and performing a new optimization is continued until the maximum number of control steps is reached.

Fig. 1 gives an illustration of how the successive-splitting multiscale approach splits the control steps to give the next finer scale. In this figure, we show the resulting number of control steps for two choices of n_s ($n_s = 2$ and $n_s = 4$) assuming the number of initial steps is $n_0 = 2$.

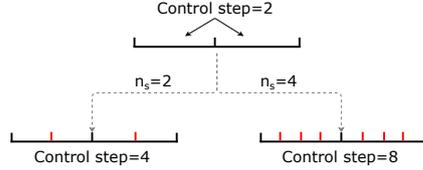


Fig. 1 Control steps split by the successive-splitting multiscale approach.

Our well control optimization procedure using a derivative-free multiscale approach is described by the steps given in Algorithm 1. A flow chart of the algorithm is given in Fig. 2.

Algorithm 1 The multiscale approach with derivative-free algorithms

```

select solver: GPS, PSO or CMA-ES
set initial control steps for each well  $n_0$ , and the split factor  $n_s$ 
set initial guess  $\mathbf{x}_0$ 
iteration  $i \leftarrow 0$ 
while not (global stopping criteria reached) do
  while not (scale stopping criteria) do
    solve  $\mathbf{x}_* = \operatorname{argmax} \text{ NPV}(\mathbf{x})$ 
  end while
  let  $\mathbf{x}_{i*} = \mathbf{x}_*$ 
  split, set control steps for each well  $n_{i+1} \leftarrow n_i \times n_s$ 
  update the initial guess,  $\mathbf{x}_{i+1}(n) = \mathbf{x}_{i*}(\lceil n/n_s \rceil)$ ,  $n = 1, 2, \dots, N_w \times n_s$ 
  test the scale stopping criteria
end while

```

We consider three derivative-free optimization algorithms in the multiscale approach: GPS, PSO, and CMA-ES. These are typical derivative-free, black-box optimization algorithms. Each method has distinct characteristics and all have been applied successfully to solve reservoir development problems as mentioned in the introduction.

GPS is a deterministic local search algorithm. The algorithm samples at a set of points on a grid around the current best point, looking for a point where the value of the objective function is better than the current value. The population size for GPS is decided automatically by the algorithm based on the problem dimension. For a D -dimensional problem, the population size equals $2D$ is using a maximal positive basis. For a further description of GPS and its application to well control optimization, see [3, 38, 5, 62, 69]. PSO is a population-based stochastic global search method. The PSO search mechanism mimics the social behaviour of

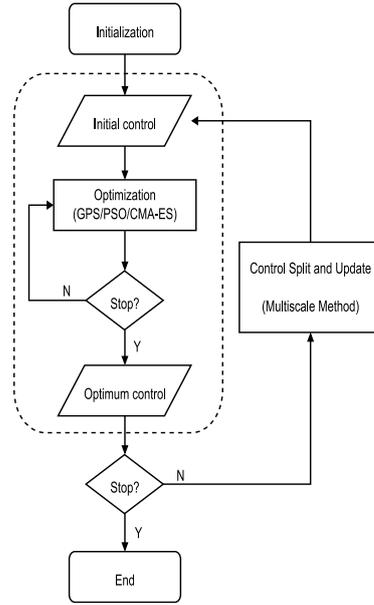


Fig. 2 Flow chart of well control optimization with multiscale approach and derivative-free algorithms.

biological organisms such as a flock of birds, see [40]. PSO initially chooses a population of candidate solutions (called a swarm of particles). These particles move through the search space in search of function improvement according to a random rule which updates each particle's position. The population size is generally varied from 20 to 100 and is usually decided by the user. For a more complete description of PSO and reservoir applications, see [40, 52, 51, 37, 64, 33, 18, 17, 54, 22]. CMA-ES is a population-based stochastic optimization algorithm. Unlike GA, PSO, and other classical population-based stochastic search algorithms, candidate solutions of CMA-ES are sampled from a probability distribution which is updated iteratively. Like GPS, the population size for CMA-ES depends only on the problem dimension. $4 + \lceil 3 \ln(D) \rceil$ solutions is generated initially for a D -dimensional problem. This algorithm performs better on the benchmark multimodal functions than all other similar classes of learning algorithms, see [68]. A more detailed description of CMA-ES and its applications can be found in [7, 23, 29, 68]. All three algorithms do not directly use or require the gradient of the objective function to be specified (or even to exist). Hence they can be used on functions that lack smoothness, those that are not continuous or differentiable.

At early scales of the algorithm, optimizations are performed with less stringent convergence tolerances to find an approximate solution. The tolerances are made smaller as the algorithm proceeds, using the smallest tolerance at the last scale. With this approach, we reduce the computational cost at early scales. The tolerance settings for the experiments can be found in Section 5.

Due to the complexity of the well control optimization problems, the objective functions are generally nonconvex with many minima, maxima, and saddlepoints. For these types of problems, GPS and CMA-ES are guaranteed to converge to

locally optimal solutions and can provide useful approximate solutions for the global optimization problems ([5, 62, 69, 68]). PSO, with the stochastic movement of the population, can reduce the chance of getting trapped at a local optimum ([40, 64]). However, PSO does not guarantee an optimal solution is ever found in practice.

In the multiscale framework, a sequence of well control optimization problems are constructed by increasing the number of variables during the optimization process. Each problem is solved by an optimization algorithm. The optimum found at one scale is used as the initial guess for the problem at the next scale. The multiscale framework can provide useful initial guesses for well control optimization problems at each subsequent scale. This strategy can accelerate the convergence of the approach to an optimal solution, but the convergence to the global optimum is not guaranteed. However, even a local optimum may often give a significant improvement over the initial control strategy ([4, 8, 61, 65, 55, 44]).

Finding the global optimum for well control optimization problem is challenging. There are several ways to improve the global convergence of the multiscale approach, such as using algorithms with better global convergence properties, increasing the computational budget, and adding more randomization to the multiscale framework. In this paper, we choose not to explore this topic.

4 Case studies

In this section, we list all approaches considered, and give a detailed description of the reservoir models used in this paper.

4.1 Optimization Approaches

The approaches considered in this paper include the three original optimization algorithms, GPS, PSO, and CMA-ES and three hybrid approaches that combine these algorithms with our modified multiscale method. The description of GPS, PSO, and CMA-ES are given in [5, 62, 64, 40, 28, 6]. The hybrid multiscale approaches are labeled as M-GPS, M-PSO, and M-CMA-ES.

To investigate the effect of n_0 and n_s , we test four different configurations for each hybrid approach. We use the Roman numerals I, II, III, and IV to represent the four configurations. The configurations used are:

- Configuration I: the initial number of control steps for each well is $n_0 = 1$ and the split factor is $n_s = 2$. With this configuration, the multiscale method is the same as the successive-splitting multiscale method from [57].
- Configuration II: the initial number of control steps for each well is $n_0 = 2$ and the split factor is $n_s = 2$.
- Configuration III: the initial number of control steps for each well is $n_0 = 2$ and the split factor is $n_s = 4$.
- Configuration IV: the initial number of control steps for each well is $n_0 = 1$ and the split factor is $n_s = 4$.

Fig. 3 provides an overview of all approaches considered in our experiments. The approaches fall into different quadrants according to their search features.

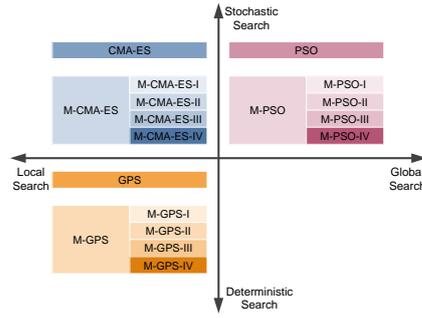


Fig. 3 An overview of all optimization approaches considered in our experiments.

4.2 Model description

Two reservoir models are considered in this paper. The first one is a simple 2-D reservoir model. This model is used to analyze the performance of the approaches mentioned in Section 4.1. The second model is a real-world reservoir model, and we apply the multiscale approaches to this model to optimize the control strategy.

4.2.1 Model 1: 5-spot model

The first model is a single-layer reservoir containing four producing wells and one injection well in a five-spot well pattern [49]. The reservoir model is represented by a 51×51 uniform grid ($\Delta x = \Delta y = 10m$; $\Delta z = 5m$). We consider only oil-water two phase flow. The water compressibility is 4×10^{-4} 1/MPa, water viscosity is 0.42 mPa \cdot s, oil compressibility is 5×10^{-6} 1/MPa and the oil viscosity is 1.6 mPa \cdot s. The rock compressibility is 2.1×10^{-3} 1/MPa. The oil-water relative permeability curves are shown in Fig. 5. The reservoir permeability field and well placements are shown in Fig. 4. We note that there are four different regions of homogeneous permeability. The permeabilities are 1000 mD for the two high-permeability regions, and 100 mD for the two low-permeability regions. The porosity, net-to-gross ratio, and initial water saturation are set to 0.2 at all grid blocks.

The reservoir lifetime is set to 720 days. The injection well INJ-01 (in Fig. 4) is not controlled, the liquid rate is fixed at 240 m³/d. The liquid rates of four producing wells are the optimization variables. Bound constraints are considered for the producing wells. The lower bound is set to 0 m³/d and the upper bound is 80 m³/d for PRO-01 and PRO-03 while 0 m³/d and 40 m³/d are the lower and upper bounds for PRO-02 and PRO-04. The initial rates of all producing wells are 20 m³/d. The BHP bounds are set in the simulator. The lower BHP is set to 5 MPa and the upper BHP is set to 40 MPa for all wells.

The objective function we use for this model is the NPV (see equation (1)) and the corresponding economic parameters are given in Table 1.

4.2.2 Model 2: PUNQ-S3

The second reservoir model is the PUNQ-S3, which is a small-size reservoir model based on the North Sea reservoir [24]. The model contains a three phase gas-oil-

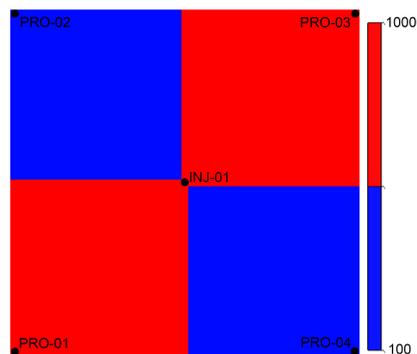


Fig. 4 The permeability field (mD) for model 1.

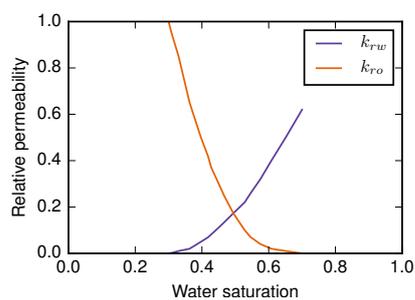


Fig. 5 The oil-water relative permeability curves for model 1.

Table 1 Economic parameters used for model 1.

Parameter	Value
Oil revenue	USD 500.0/ m^3
Water-production cost	USD 250.0/ m^3
Water-injection cost	USD 80.0/ m^3
Annual discount rate	0

water system with $19 \times 28 \times 5$ grid blocks, of which 1761 blocks are active. The field contains 6 production wells but no injection wells are present due to the strong aquifer. Fig. 6 shows the porosity, permeability and oil saturation present in the model.

We use a production period of 3840 days (about 10 years), with a minimum control interval of 120 days. The initial liquid rates for all wells are $100 m^3/d$. The lower bound is set to $0 m^3/d$ and the upper bound is $200 m^3/d$ for all wells. BHP bounds are also considered in this example. The lower BHP bounds are set to 12 MPa and no upper bound is enforced for any producers. The economic parameters for the NPV calculation are given in Table 2.

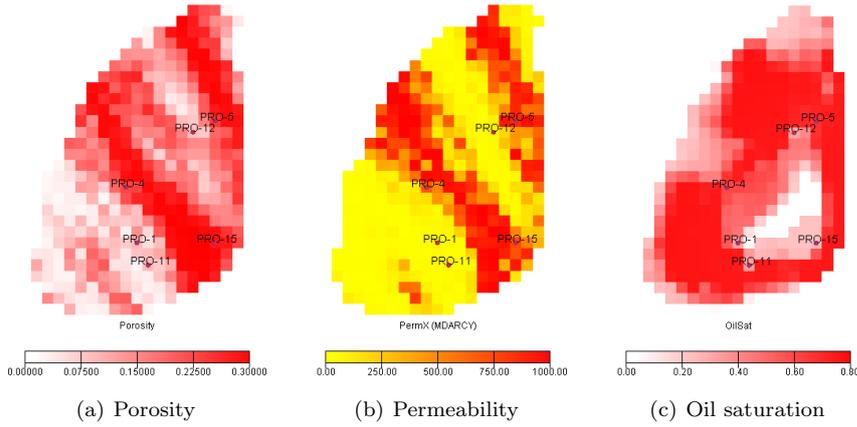


Fig. 6 Property and wells of PUNQ-S3 field.

Table 2 Economic parameters used for PUNQ-S3.

Parameter	Value
Gas revenue	USD $0.5/m^3$
Oil revenue	USD $500.0/m^3$
Water-production cost	USD $80.0/m^3$
Annual discount rate	0

5 Results and discussion

5.1 Effects of control frequency on well control optimization

To show the effect of the control frequency on the NPV for the first model, as described in Section 4.2.1, we turn off the multiscale approach and optimize using four different, fixed control frequencies. Four control frequencies are considered and these constitute four variations of the optimization problem:

- Case 1A: each well is produced under a liquid rate throughout its lifetime. This gives 4 optimization variables in total.
- Case 1B: the liquid rate for each well is updated every 360 days (2 control periods). This gives 8 optimization variables in total.
- Case 1C: the liquid rate for each well is updated every 90 days (8 control periods). This gives 32 optimization variables in total.
- Case 1D: the liquid rate for each well is updated every 22.5 days (32 control periods). This gives 128 optimization variables in total.

Three optimization algorithms, GPS, PSO, and CMA-ES, are applied to each case to find the optimal controls and the corresponding NPV.

5.1.1 Optimal NPV under different control frequencies

Fig. 7 compares the optimal NPV under the four different control frequencies for this model. The results shown are the best values found using all the optimization

approaches. Well control with a reasonable frequency is necessary — we obtain a significantly higher NPV than what is possible when using a fixed rate over the life cycle (Case 1A). It is clear that with a continued increase of the number of control adjustments, the optimal NPV grows more and more slowly. The increase in maximum NPV found is very slight (0.28 %) when the number of control steps for each well increases from 8 (Case 1C) to 32 (Case 1D). There is no need to adjust well rates too frequently. We will not see a considerable revenue increase and the increase in the number of control adjustments will increase operation costs. Also the problem with a large number of control adjustments is harder to optimize and the algorithms have a higher risk of falling into a local optima (see Section 5.2.2). This justifies the use of multiscale approach to determine an appropriate control frequency.

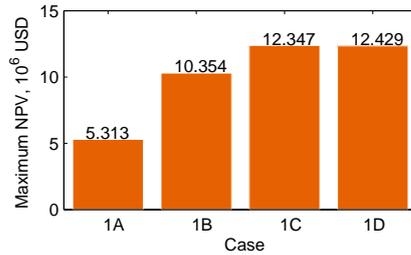


Fig. 7 Optimum NPV for cases with different control frequencies.

5.1.2 Optimal controls under different control frequencies

Fig. 8 presents the optimum controls for wells PRO-01 and PRO-02 under different control frequencies. We omit the results for well PRO-03 and PRO-04 because the reservoir is symmetric. The optimum controls become more like a bang-bang solution for all wells with an increase in the number of control steps. It is worth noting that the optimum controls for Case 1A are significantly different that those for Cases 1B–1D. This reflects the different production strategies for wells using a static rate compared to using dynamic well controls in water flooding reservoirs. The similarity of optimum controls between different control frequencies is important for the success of a multiscale framework. As the multiscale approach uses the optimal controls found in iteration i as the initial guess for iteration $i + 1$, a good initial guess could accelerate the optimization process and a bad initial guess may guide the optimizer to wrong search areas and directions (see Section 5.2.1). Indeed Fig. 8 shows this required similarity as the number of control steps is increased.

5.2 Performance of GPS, PSO, and CMA-ES for well control optimization

In this section we address the performance of GPS, PSO, and CMA-ES for well control optimization without the use of the multiscale framework. We use the same

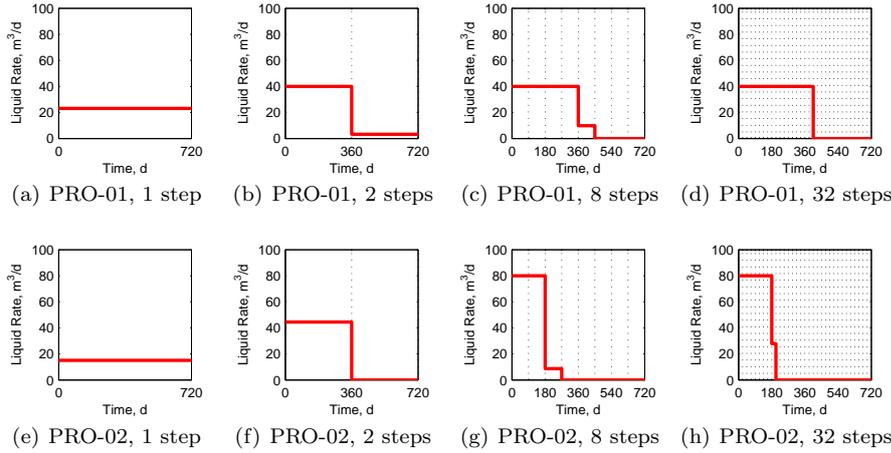


Fig. 8 The optimum well controls for different control frequencies.

test cases as in Section 5.1. We use the maximum number of simulation runs as the stopping criterion, and this value is set to 100 times the number of control variables. As PSO and CMA-ES are stochastic algorithms, 10 trials are performed for these two algorithms to assess the average performance.

5.2.1 Parameter tuning and the effect of the initial guess on GPS, PSO, and CMA-ES

The performance of the optimization algorithms are affected by the choice of their parameter values. In this section, we complete parameter tunings for GPS, PSO, and CMA-ES to improve their performance in solving well control optimization problems. Here we perform a tuning study for two choices of initial guesses. The *good* initial guess is chosen to mimic the initial guess provided by the multiscale algorithm. The *bad* initial guess is purposely chosen to be far away from the optimal controls.

We hypothesize that the performance of the local search algorithms are highly affected by the initial guess, while the stochastic global search algorithms are not. We take Case 1B as an example and use the three different initial guesses shown in Table 3. For each initial guess, 10 trials were performed for PSO and CMA-ES and 1 trial for GPS (since it is a deterministic algorithm).

Table 3 Three initial guesses for GPS, PSO, and CMA-ES.

Type	Initial guess	NPV, $\times 10^6$ USD
good	[20, 20, 20, 20, 20, 20, 20, 20]	5.0009
medium	[20, 20, 40, 40, 40, 40, 20, 20]	2.6484
bad	[0, 40, 0, 80, 0, 80, 0, 40]	-4.2826

Fig. 9 shows the plots of NPV versus the number of simulations for GPS, PSO, and CMA-ES. Each line represents one trial. The early convergence history

of GPS and PSO is affected by the choice of the initial guess. CMA-ES recovers quite quickly from the bad initial guess, and even converges more quickly than from a good initial guess in some cases. With a large number of simulation runs, the effect of initial guess for all three algorithms is quite small. This suggests that if we want to make efficient use of the multiscale approach, the number of simulations at each scale should be limited.

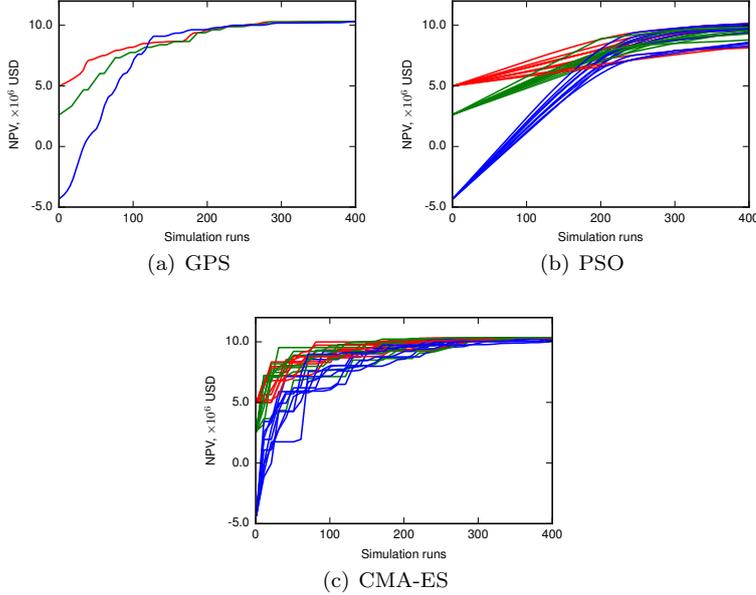


Fig. 9 Plots of NPV versus the number of simulation runs for GPS, PSO, and CMA-ES with different initial guesses with 8 control steps. Each line represents one trial. The colors red, green, and blue denotes good, medium, and bad initial guesses, respectively.

Again using a good and a bad initial guess, we analyzed the effect of the other parameters for PSO and CMA-ES, to identify the primary parameters which affect the performance of the algorithms. For PSO, the algorithm parameters include the population size λ , and the parameters ω , c_1 , and c_2 . Three levels are chosen for each parameter. For CMA-ES, the algorithm parameters include the population size λ , the parent number μ (number of candidate solutions used to update the distribution parameters), the recombination weights ω , and the parameter σ , which determines the initial coordinate-wise standard deviations for the search.

For each parameter choice, 10 trials are performed for Case 1B with a good and a bad initial guess to start the optimization. We use the best NPV obtained after 20% of the maximum number of simulation runs as the evaluation criterion of the algorithm's performance. We mainly focus on the performance of the algorithms at an early stage because the multiscale framework requires a good early stage performance for the hybrid optimization algorithm. Also our test results showed that the performance of the algorithms are less sensitive to the parameter values at a later stage. The performance for each parameter choice is shown in the beanplots

in Fig. 10 and Fig. 11. A beanplot [39] promotes visual comparison of univariate data between groups. In a beanplot, the individual observations are shown as small points or small lines in a one-dimensional scatter plot. In addition, the estimated density of the distributions is visible and the mean (bold line) and median (marker '+') are shown.

From Fig. 10 we can see that for PSO the population size plays the most important role in the algorithm's ability to utilize the good initial guess. When the population size equals 20 or 50, PSO with a bad initial guess obtains an optimal NPV in a similar number of iterations as PSO with a good initial guess. This is because, with the same number of simulation runs, PSO with a small population size can evolve more generations, and this decreases the affect of the initial guess. The bigger the population size, the smaller the variability in the NPV results, with a similar mean value. For these reasons, we choose $\lambda = 100$ for all subsequent PSO experiment. Parameter c_2 is one of the weighting parameters, the bigger c_2 , the greater the tendency for the particles to fly towards the best location found so far. We suggest a bigger c_2 when combining with the multiscale approach. The parameters ω and c_1 have no obvious affect in this case. Generally for all parameter values PSO responds favorably to the better initial guess.

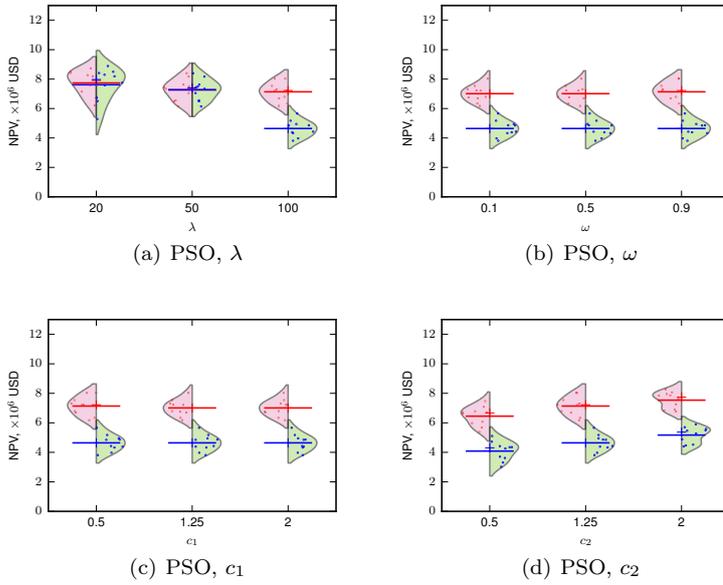


Fig. 10 Beanplots of the NPV for various parameter settings of PSO. The left side of each beanplot gives the results obtained with a good initial guess, and the right side gives the results obtained with a bad initial guess. The individual dots show the NPV obtained by each trial. The background pink and green colors show the distribution of results. The short horizontal line and the marker '+' denote the mean and median of all 10 trials, respectively.

From Fig. 11 we can see that the good initial guess gives a higher average NPV with smaller variability for CMA-ES. For this problem, the best configuration is

$\sigma = 0.3$, $\lambda = 10$, $\mu = 2$, and $\omega = \text{superlinear}$; this is also the default configuration of CMA-ES. In fact, according to the work of [31], CMA-ES does not require significant parameter tuning for its application. The choice of parameters is not generally left to the user (arguably with the exception of population size σ).

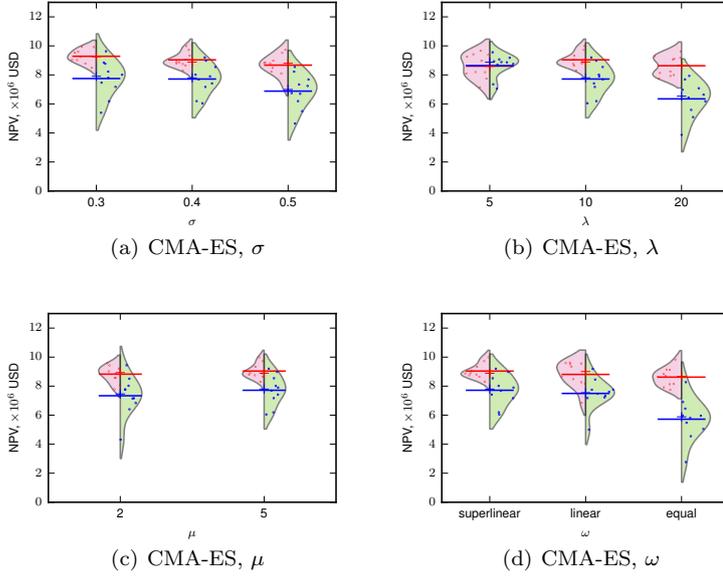


Fig. 11 Beanplots of the NPV for various parameter settings of CMA-ES. The left side of each beanplot gives the results obtained with a good initial guess, and the right side gives the results obtained with a bad initial guess. The individual dots show the NPV obtained by each trial. The background pink and green colors show the distribution of results. The short horizontal line and the marker ‘+’ denote the mean and median of all 10 trials, respectively.

5.2.2 Performance with different control frequencies

Table 4 and Fig. 12 show the performance of GPS, PSO, and CMA-ES for well control optimization problems with different control frequencies. In Table 4, the maximum, minimum, mean, median, and standard deviation of the NPV for each case are given. From the table we can see that, for Case 1A, which has only 4 variables, GPS obtains the highest NPV after 400 simulation runs. Similar results are found in Case 1B. In Case 1C, the maximum NPV of CMA-ES exceeds the result obtained by GPS, but the mean and median NPV for CMA-ES are lower than those of GPS. In these three cases, although the final NPV of GPS is larger than the final NPV for CMA-ES and PSO, the difference of the mean/median NPV for the three algorithms is quite small (less than 2%). In Case 1D, which has 128 variables, the NPV obtained by GPS is obviously lower than that of CMA-ES. Generally, CMA-ES showed excellent performance in most cases. GPS performs best when the problem dimension is very small.

Table 4 Results for Cases 1A-1D for Model 1 using GPS, PSO, and CMA-ES. Values shown are NPV ($\times 10^6$ USD).

Case	Algorithm	Trials	Max	Min	Mean	Median	Std.
1A	GPS	1	5.3132	5.3132	5.3132	5.3132	-
	PSO	10	5.2850	5.1850	5.2603	5.2720	0.0310
	CMA-ES	10	5.3121	5.2969	5.3034	5.3045	0.0048
1B	GPS	1	10.3539	10.3539	10.3539	10.3539	-
	PSO	10	10.3200	9.4220	10.0840	10.1700	0.2819
	CMA-ES	10	10.3536	10.3511	10.3527	10.3528	0.0008
1C	GPS	1	12.3470	12.3470	12.3470	12.3470	-
	PSO	10	12.2700	11.3800	11.9660	12.1050	0.2966
	CMA-ES	10	12.3474	12.3447	12.3466	12.3467	0.0007
1D	GPS	1	9.5083	9.5083	9.5083	9.5083	-
	PSO	10	11.7000	10.2300	11.1290	11.1700	0.4910
	CMA-ES	10	12.4285	12.3466	12.4054	12.4178	0.0315

Fig. 12 shows the plots of NPV versus the number of simulation runs for the four cases. In this figure, we use a solid line to show the median NPV of each algorithm, and use the same color as the line to fill the area between the maximum and minimum NPV for each algorithm. These plots clearly show the performance of GPS, PSO, and CMA-ES using different computational budgets. GPS obtains the highest NPV for Case 1A–1C at the end of optimization. But the budget (number of simulation runs) required for GPS grows rapidly as the dimension of the problem increases. GPS converged with no more than 50% of total budget for Cases 1A and 1B, and about 80% of the total budget for Case 1C. For Case 1D, GPS did not converge after 100D simulation runs. CMA-ES obtains almost as high a NPV as GPS for Case 1A–1C, and it obtains highest NPV for Case 1D. Furthermore, CMA-ES showed an excellent performance when the budget is limited. PSO also outperforms GPS for a low budget and a large problem dimension, but it still can not beat CMA-ES in these cases.

Since PSO and CMA-ES are stochastic algorithms, the performance is different for each trial. In Table 4 we can see the standard deviation for PSO is larger than the standard deviation for CMA-ES. In Figure 12 we can see that the best NPV obtained has a higher variation for low computational budgets than for high budgets. For PSO, the variability did not decrease in Case 1C and 1D as the algorithm converged.

To investigate further, we choose 2 of the 32 variables and 5 of the 10 trials for Case 1C and then compare the population distribution of CMA-ES and PSO at different iterations. The resulting scatter diagrams are shown in Fig. 13. In Case 1C, the population size is 100 for PSO, and 14 for CMA-ES. Hence after same number of simulation runs, CMA-ES and PSO are at different iteration numbers. After 500 simulation runs, CMA-ES is at the 36th iteration, while PSO is at the 5th iteration. After about 3000 simulation runs (Fig. 13(d) and 13(h)), we can see that PSO has converged to different locations for each trial. Compared with CMA-ES, PSO is more easily falls into local optima in our test cases, in spite of the larger population size and the ability search the entire space.

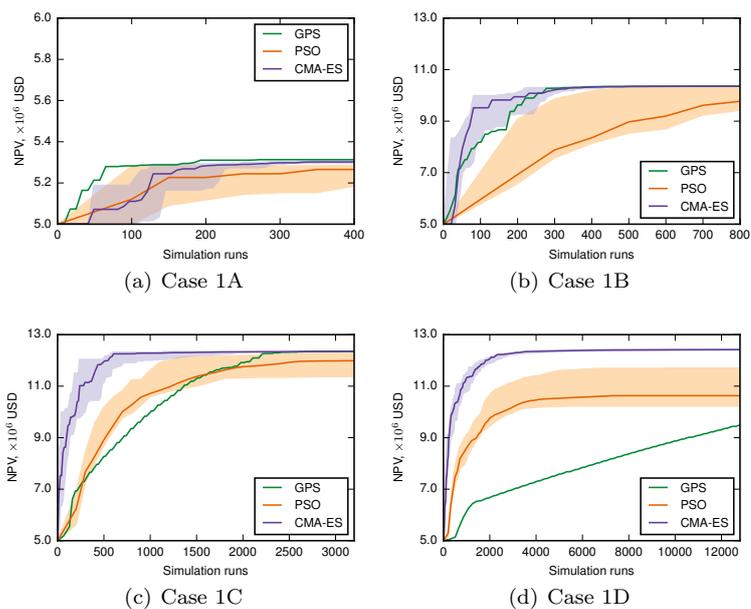


Fig. 12 Optimization performance for well control problems using GPS, PSO and CMA-ES. The solid lines give the median NPV over all 10 runs of PSO and CMA-ES without the multiscale framework. The areas between maximum and minimum NPV are filled with the corresponding color. Note that the x -axis scale is different for each case.

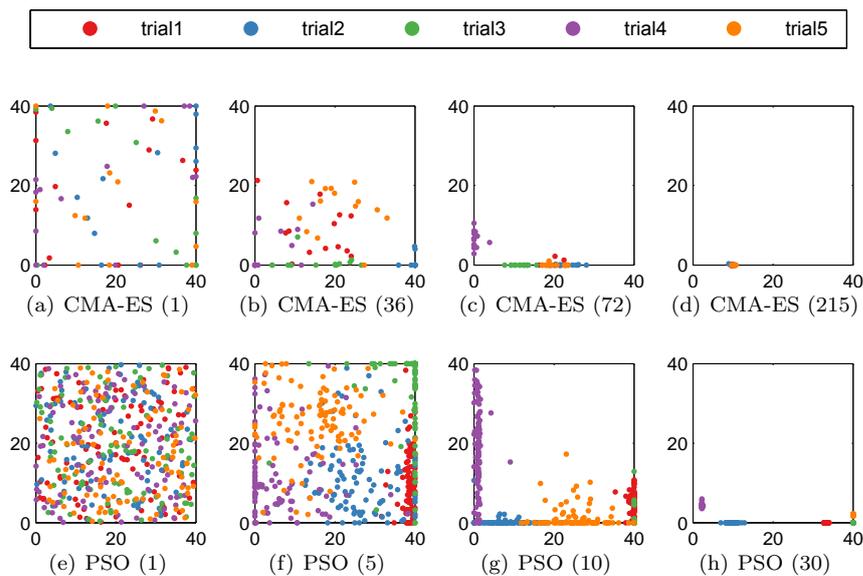


Fig. 13 Scatter diagrams for CMA-ES and PSO for Case 1C. The bracketed number in the caption of each sub-figure is the iteration number. Points represent the candidate solutions at this specific iteration.

5.2.3 Performance in parallel environments

GPS, PSO, and CMA-ES parallelize naturally. Here we investigate the performance of these three algorithms in parallel environments. Table 5 gives the population sizes of GPS, PSO, and CMA-ES in Cases 1A–1D. The population size for GPS and CMA-ES are decided automatically by the algorithms based on the problem dimension. For a D -dimensional problem, the population size equals $2D$ for GPS, and $4 + \lfloor 3 \ln(D) \rfloor$ for CMA-ES. The population size for PSO is usually decided by the user and we use 100 for all cases (more discussion on the population size is given in Section 5.2.1).

In a parallel environment, we can evaluate a number of individuals, up to the number of processors, simultaneously. Note that we are not able to evaluate the individuals from different iterations at the same time. For well control optimization problems, the computation time is mainly spent evaluating the reservoir simulation, a parallel environment can greatly reduce the time of optimization.

Table 5 Population size of GPS, PSO, and CMA-ES for Case 1A-1D. D is the number of variables in the problem.

Case	D	GPS	PSO	CMA-ES
1A	4	8	100	8
1B	8	16	100	10
1C	32	64	100	14
1D	128	256	100	18

Assume we have three parallel environments, with 8, 32, and an infinite number of processors, respectively. Fig. 14 compares the parallel performance of GPS, PSO, and CMA-ES in the parallel environments to the performance in a sequential environment. We use the number of *runs* as the y -axis in this figure. One run evaluates a number of potential solutions up to the number of processors. In an iteration, if the number of potential solutions is less than the number of processors, then all the potential solutions are evaluated in a single run, with some processors idle. The number of runs is equal to the number of simulations if we have only one processor.

In Fig. 14, we compare the number of runs needed to get from the initial NPV to 50% of the final NPV, as well as the number of runs needed to reach the maximum number of simulator evaluations (100 times the problem dimension). From this figure we can see that, with the increase of processors, the number of runs required for GPS, PSO, and CMA-ES decrease, until the number of processors is larger than the population size. For each algorithm, the larger the population size, the greater the benefits from the parallel environment. With an increase in the number of processors, the order of performance of the three algorithms changes. For Case 1A, GPS performs best followed by CMA-ES and PSO in the sequential environment (number of processors equals 1). The order becomes GPS, PSO, CMA-ES (in decreasing order of performance) when the number of processors reaches 32. And with enough processors (≥ 100), the order becomes PSO, GPS, CMA-ES. This is because for Case 1A, the population size for GPS, PSO and CMA-ES is 8, 100, and 8, respectively. Once the number of processors exceeds 8, the total number of runs required for GPS and CMA-ES no longer decreases. The

total number of runs required for PSO keeps falling until the number of processors exceeds 100. The order also changes depending on the number of processors for Case 1B–1D. Generally, a parallel environment can greatly reduce the time spent for these algorithms. PSO can outperform GPS and CMA-ES in performance if the number of processors is large enough.

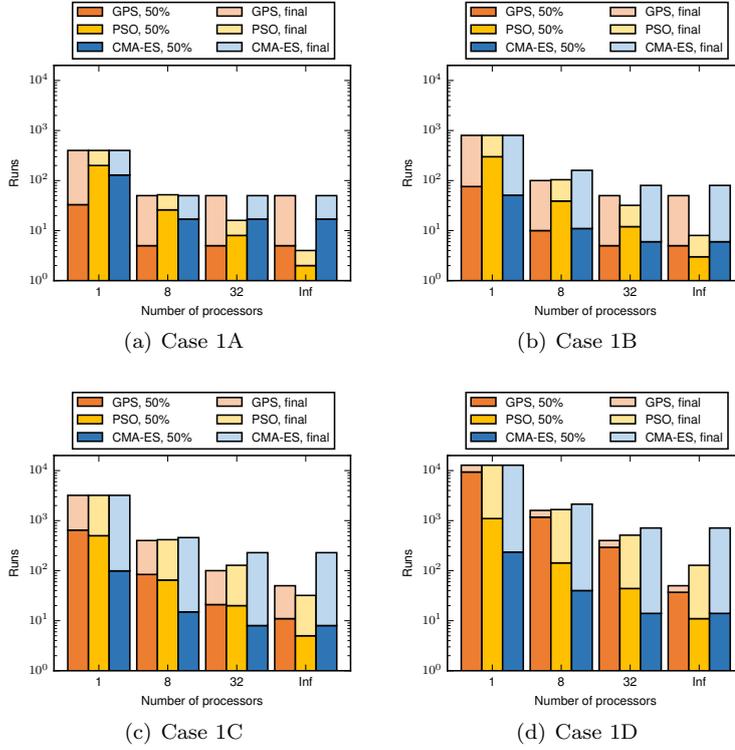


Fig. 14 Number of runs required for the well control optimization problems in parallel environments with different numbers of processors. “50%” in the legend denotes the number of runs required to reach 50% of the maximum NPV for the algorithms. “final” in the legend denotes the total number of runs required to reach the maximum number of simulator evaluations for the algorithms.

5.3 Multiscale optimization for Model 1

In this section we address the performance of the multiscale approaches (M-GPS, M-PSO, and M-CMA-ES) for well control optimization. We use the first model as described in Section 4.2. We stop the optimization at each scale when the average relative well rate change is less than 10% of the distance between the upper and lower bounds on the well rates. No further scales will be completed when the relative change in the NPV is $< 10\%$ between two neighboring scales.

The maximum number of simulation runs for the problem is set to 3200. As M-PSO and M-CMA-ES are stochastic algorithms, 10 trials were performed for these two algorithms to assess the overall performance.

5.3.1 Performance of the multiscale approaches

As in Section 4.1, we consider four configurations for each multiscale approach. As a first test, the multiscale optimization process is terminated when the number of control steps reaches 8 each well for configuration I–III, and 16 each well for configuration IV. The values of n_0 and n_s in different configurations are given in Section 4.1.

The plots of NPV versus the number of simulation runs for the different multiscale approaches and the different configurations, as well as the plots for the standard algorithms (GPS, PSO, CMA-ES) with 8 pre-set control steps for each well, are shown in Fig. 15. From the figure we see that compared with direct optimization with 8 well control adjustments, both GPS and PSO converge faster when using the multiscale approach. GPS convergence improves the most amongst the three algorithms. Fig. 15(b) shows that for this test case, M-PSO locates a control strategy which gives a higher NPV than PSO. The performance of CMA-ES (Fig. 15(c)) is quite different. The results show that CMA-ES converges faster than M-CMA-ES for this relatively small scale optimization problem. This is because CMA-ES is less sensitive to the quality of the initial guess and hence it does not take great advantage of the multiscale framework to speed up its convergence rate. The multiscale framework for CMA-ES still does, however, give us a way to automatically detect a good control frequency for well control optimization.

As a second test of the multiscale framework we increase the number of control steps to 32 for each well. The plots of NPV versus the number of simulation runs for the different multiscale approaches and the different configurations, as well as the plots for the standard algorithms (GPS, PSO, CMA-ES) are shown in Fig. 16. From the figure we see that compared with direct optimization, all algorithms converge faster when using the multiscale approach. GPS convergence improves the most amongst the three algorithms, followed by PSO and CMA-ES.

5.3.2 Parameter tuning of the multiscale approaches

To investigate the effect of n_0 and n_s in more detail, we test the performance of our multiscale approach with different configurations. The tests are divided into two groups:

- Group 1: the initial number of control steps for each well is varied from 1 to 4 and the split factor is fixed at $n_s = 2$.
- Group 2: the initial number of control steps for each well is fixed at $n_0 = 2$ and the split factor is varied from 2 to 5.

We use all three derivative free algorithms to tune of the multiscale parameters n_0 and n_s .

The effect of the initial number of control steps, n_0 , is shown in Fig. 17 and Fig. 19. With a low computational budget, a higher value of $n_0 \geq 2$ is generally better for all three algorithms. A low budget allows very few scales to complete.

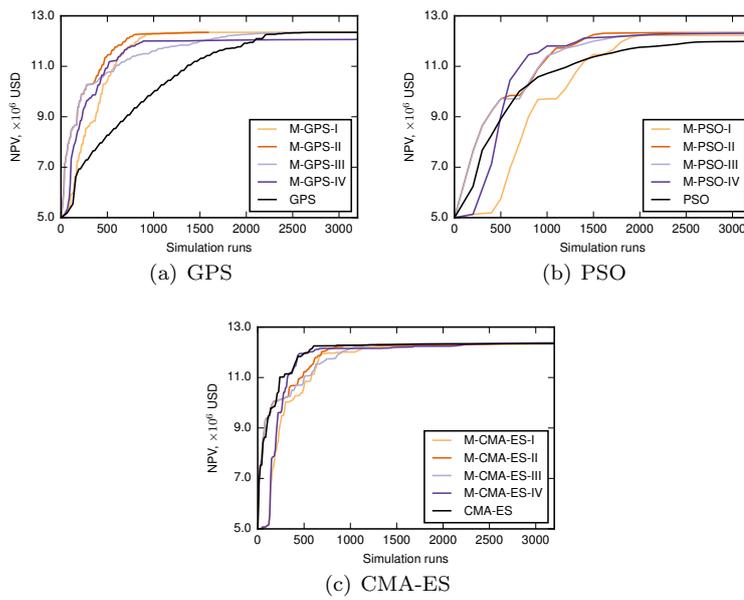


Fig. 15 Comparison of the performance of multiscale approaches with different configurations with 8 final control adjustments for each well.

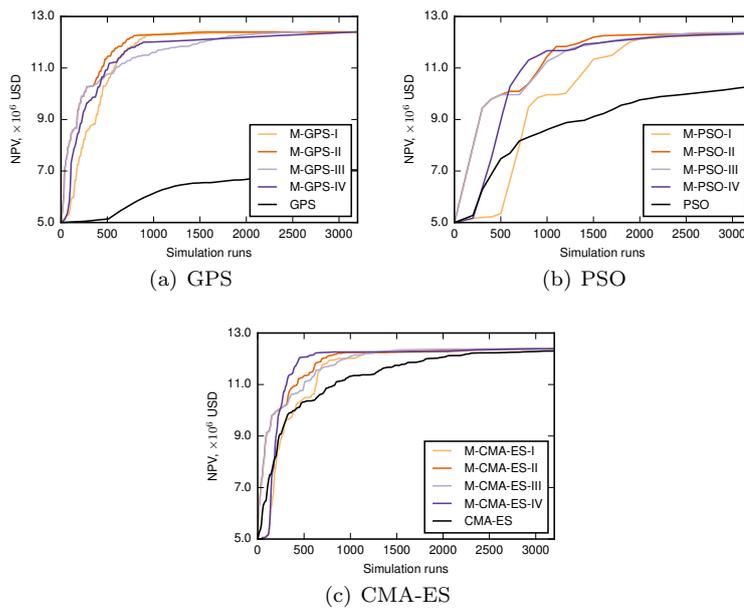


Fig. 16 Comparison of the performance of multiscale approaches with different configurations with 32 final control adjustments for each well.

Hence it is best, in terms of the final NPV obtained, to increase the initial number of control steps. Medium budgets obtain the best results with $n_0 = 2$. With a higher budget, the choice $n_0 = 2$ performs best for GPS, while PSO and CMA-ES are insensitive to the choice of n_0 . Unless using a very low budget, $n_0 = 2$ is an all round good choice.

The effect of the split factor, n_s , is shown in Fig. 18 and Fig. 20. With a low budget $n_s = 2$ is best while PSO and CMA-ES are insensitive to the choice of n_s . With a higher budget a lower values of n_s performs best for GPS and PSO, while CMA-ES is relatively insensitive. The choice $n_s = 2$ appears to perform well for all methods and budgets.

For low to medium budgets, the performance of PSO lags behind the performance of GPS and CMA-ES. This is likely due to the generation of the random swarm at the beginning of each scale (explicitly using only the best particle from the end of the last scale). With higher budgets PSO is able to eventually recover as an optimizer within the multiscale framework. The reinitialization of the swarm also affects the performance of PSO within the multiscale framework with a high split factor. GPS and CMA-ES, on the other hand, choose a new set of candidate solutions at the beginning of each scale based closely on the best solution found at the end of the last scale.

Overall the choice of n_0 affects the performance of PSO and CMA-ES more than the choice of n_s (this is true for PSO unless n_s is very large). GPS shows a slightly larger sensitivity on the choice of n_s .

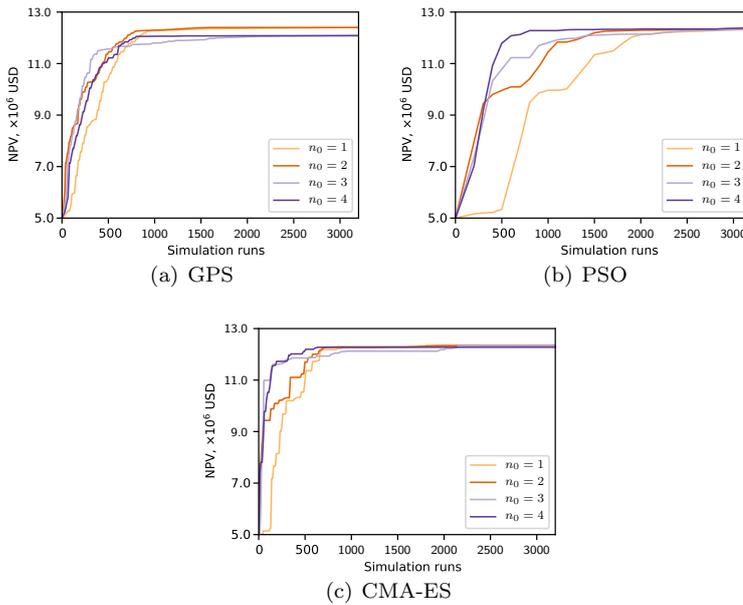


Fig. 17 Comparison of the performance of multiscale approaches with different initial numbers of control steps n_0 (the split factor n_s is fixed at 2 for each trial).

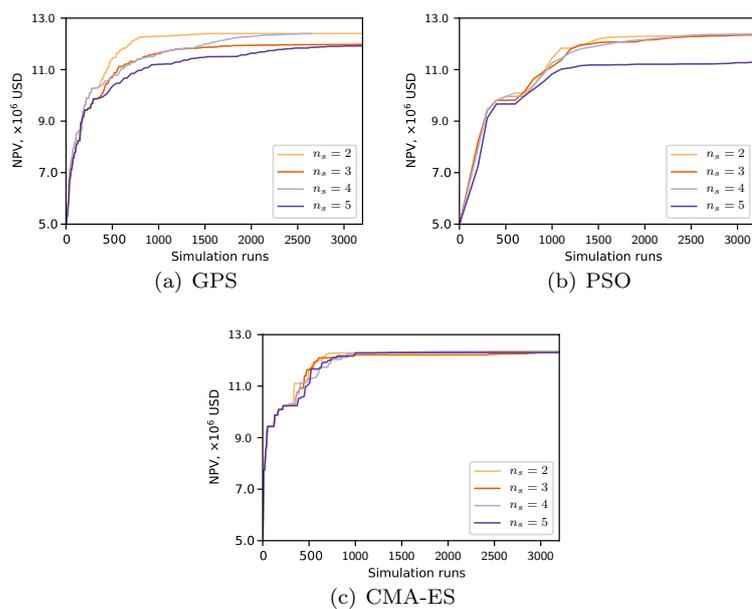


Fig. 18 Comparison of the performance of multiscale approaches with different split factors n_s (the initial number of control steps n_0 is fixed at 2 for each trial).

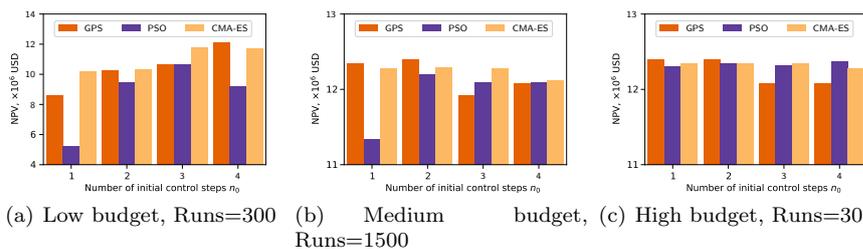


Fig. 19 The performance of multiscale approaches with different initial number of control steps, n_0 , under different budgets.

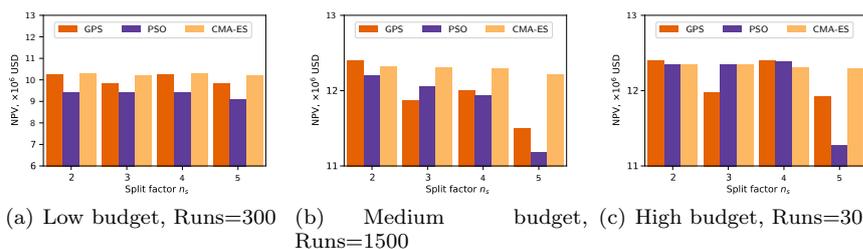


Fig. 20 The performance of multiscale approaches with different split factor, n_s , under different budgets.

5.3.3 Performance in parallel environments

Fig. 21 shows the performance of the multiscale approaches in parallel environments with 8, 32, and an infinite number of processors using 8 pre-set final control steps for configurations I–III and 16 control steps for configuration IV. As in Section 5.2.3, we use the number of runs as the horizontal axis. The lines for PSO, CMA-ES, M-PSO, and M-CMA-ES are the medians of 10 trials. From this figure we can see that, not surprisingly, the more processors the less runs needed to converge. In a parallel environment, the improved convergence of the multiscale approach is less apparent. The multiscale approach still benefits, however, if the optimal control frequency is unknown and hence not specified a priori.

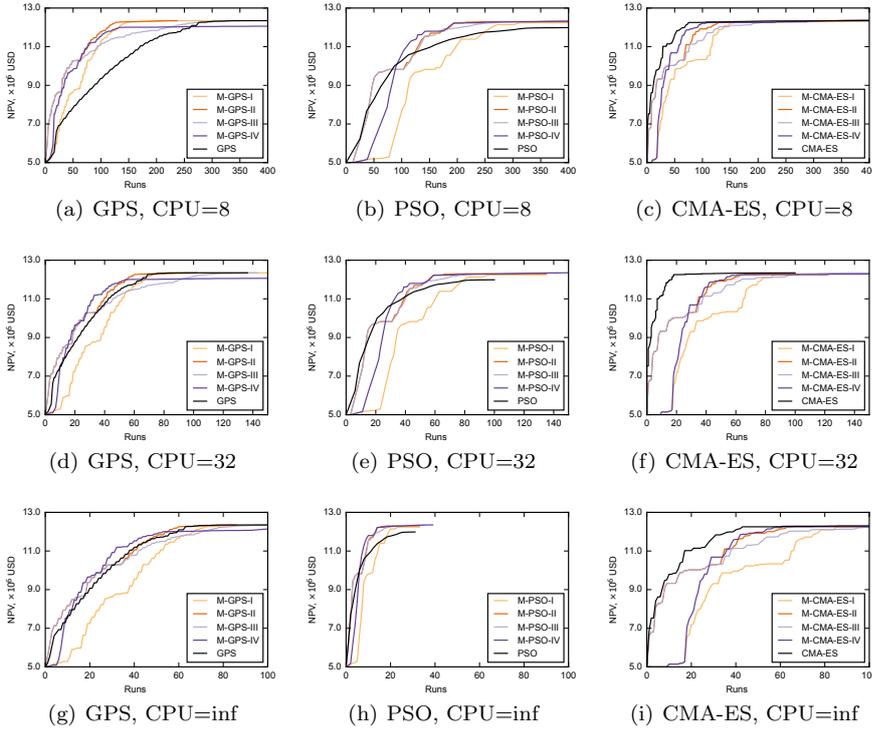


Fig. 21 Comparison of the performance of multiscale approaches with 8 final control adjustments for each well in parallel environments with 8, 32, and an infinite number of processors.

In Fig. 22 we repeat this experiment for the more difficult problem with 32 control steps for each well. The results are similar.

5.3.4 Performance with different computational budgets

Here we assess the effect of computational budget on the efficacy of the multiscale approach. We use 300, 1500, and 3000 simulation runs as a low, medium, and

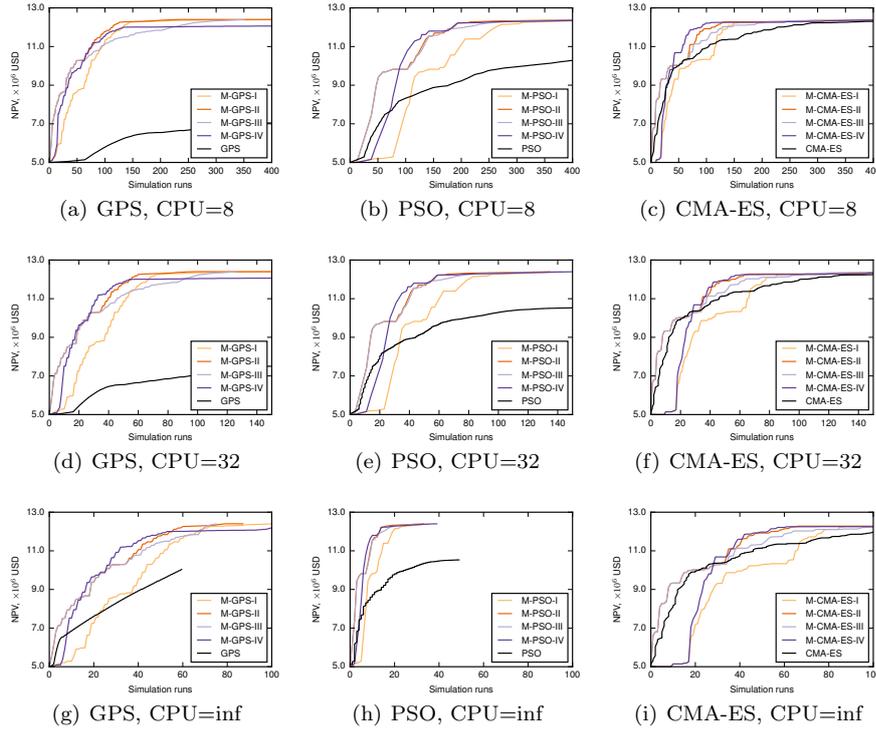


Fig. 22 Comparison of the performance of multiscale approaches with 32 final control adjustments for each well in parallel environments with 8, 32, and an infinite number of processors.

high budget. For different budgets, the performance of the different multiscale approaches and the different configurations are shown in several beanplots in Fig. 23. In the beanplots, the individual observations are shown as small lines in a one-dimensional scatter plot. The estimated density of the distributions is visible and the mean (blue bold line) and median (red '+') are shown.

Fig 23(a) shows results for a low budget. Since GPS is a deterministic algorithm, we have no distribution for the four configurations of M-GPS. M-GPS-II and M-GPS-III obtained the highest NPV amongst all four configurations. Although some configurations of M-PSO and M-CMA-ES could obtain a relatively high NPV, there is also has a risk of obtaining a low NPV due to the high variability. For a medium budget, Fig 23(b), the variation of all four configurations of M-CMA-ES is relatively small. The variation of the M-PSO is quite large. With this budget of function evaluations, M-GPS and M-CMA-ES are good choices. For a large budget, we see that M-GPS-IV obtained a higher NPV than the other configurations. With the fourth configuration, the approach terminated at 16 control steps for each well, while the other three configurations terminated at 8 control steps for each well. With more control steps, a higher NPV could be obtained. Configuration I performs less well with a low budget. This is because the initial number of control steps for configuration I is 1. The optimal control found with only 1 step is very different when compared with the optimal control found with

more steps, and hence does not provide a good initial guess for the next step of the multiscale process. In general, we see that the second configuration performs better than the other three configurations for M-GPS, M-PSO and M-CMA-ES. Furthermore, M-GPS-II is highly recommended for all budgets.

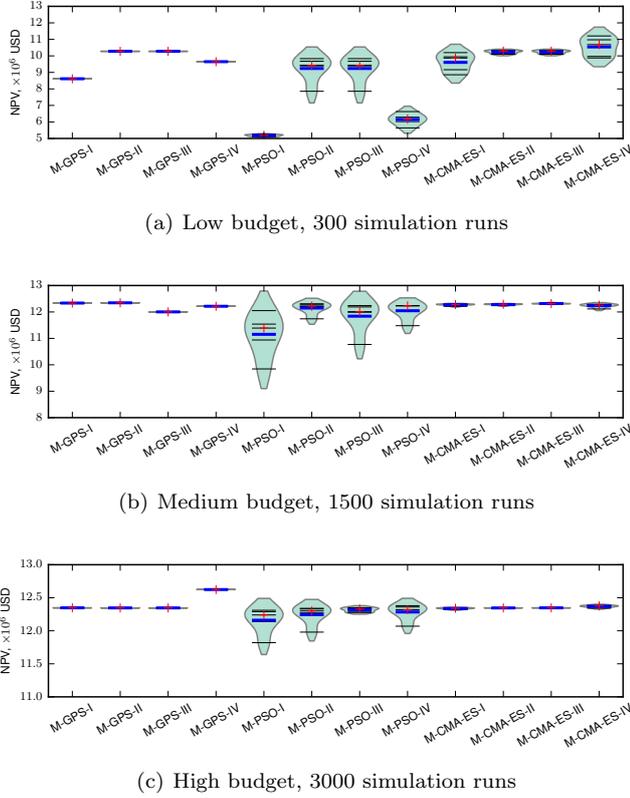


Fig. 23 Beanplots for different configurations of our multiscale approaches.

5.4 Multiscale optimization for a real-world reservoir

Based on the results of the previous section we now apply the multiscale approaches with configuration II ($n_0 = 2$ and $n_s = 2$) to solve the well control problem of reservoir PUNQ-S3 (see Section 4.2.2). The maximum number of function evaluations is set to 10000. We use an average relative well rate change of $< 10\%$ of the gap between the upper and lower bound as the stopping criterion for each scale. No further scales will be completed if the relative change in the NPV is $< 10\%$ between two neighboring scales. Due to the large computational time we perform only 3 trials for M-PSO and M-CMA-ES.

Fig. 24 shows the performance of M-GPS-II, M-PSO-II and M-CMA-ES-II. The performance of GPS, PSO, and CMA-ES using a pre-set control frequency of

32 control steps for each well is shown in this figure as well. The results show that for the same number of reservoir simulations, combining these three algorithms with the multiscale framework gives higher NPV values as compared to directly optimizing with the largest number of control steps.

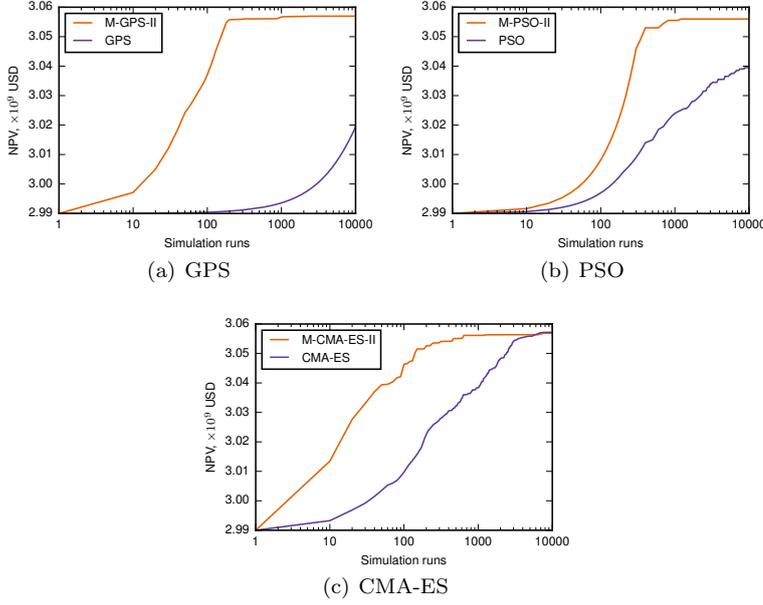


Fig. 24 Comparison of the performance of multiscale approach and optimizers without multiscale for PUNQ-S3. Here are the median NPV of trials for PSO and CMA-ES.

The values of NPV found by the approaches after 1000, 5000, and 10000 simulation runs are shown in Table 6. From the table we can see that, M-GPS-II outperforms among all three multiscale approaches, even when the number of simulation runs is limited. Of course the deterministic M-GPS has the additional advantage of having no variability in the outcome. Without the multiscale framework, CMA-ES performs the best.

Table 6 Median NPV (10^9 USD) with different budgets for PUNQ-S3.

Algorithm	Trails	1000	5000	10000
GPS	1	2.9936	3.0061	3.0194
M-GPS-II	1	3.0567	3.0570	3.0570
PSO	3	3.0240	3.0365	3.0395
M-PSO-II	3	3.0555	3.0560	3.0560
CMA-ES	3	3.0384	3.0558	3.0572
M-CMA-ES-II	3	3.0561	3.0563	3.0568

Fig. 25 shows the optimum well controls found with the different optimization approaches. The approaches include GPS, PSO, CMA-ES, which starts the op-

timization immediately with a high number of control intervals, and M-GPS-II, M-PSO-II, M-CMA-ES-II, which use the multiscale framework. From the figure we can see that, without the multiscale framework, all three optimization algorithms obtain “nervous” (fluctuating in time) control strategy for this example. PSO obtains the most “nervous” solution, and GPS the least. When combined with the multiscale framework, GPS and PSO show a significant improvement in the control strategy. M-GPS-II obtains the most stable control strategy with the highest NPV. For the stochastic algorithms, PSO and CMA-ES, the multiscale framework can decrease the degree of the “nervousness”. The reason for the observed nervous control strategy is that many different control strategies will result in (nearly) identical objective function values. With a limited computational budget, the optimization algorithms easily get trapped in a local (and nervous) optimum which leads to premature convergence. Considering the well management cost, the nervous control should be avoided in oil production in practice. We see that the multiscale framework can decrease the degree of the “nervousness”. [49] provided an alternate method to avoid the “nervous” solution. We note that, for some complex reservoirs, a nervous control may outperform since the rapid change of well production/injection rate causes fluctuations of the pressure field, resulting in a larger flooding area.

6 Conclusions

In this paper we have considered three derivative-free optimization algorithms combined with a multiscale framework to solve well control optimization problems. The optimization algorithms used include GPS, which is a deterministic local search approach; PSO, which is a stochastic global search method; and CMA-ES, which is a stochastic local search method. A generalization of the successive-splitting multiscale approach from [44,57] was introduced to combine with the derivative-free optimization algorithms.

Based on thorough numerical experiments the following conclusions can be drawn:

- The control frequency does have a significant effect on well control optimization problems. The more frequent the well control adjustment, the higher the NPV that can be obtained but at the cost of a harder optimization problem. This increase in optimal NPV becomes less significant as we continue to increase the number of control steps. Considering the operation costs, each reservoir has an optimal control frequency. The optimal controls are similar with different control frequencies when every well is produced under a liquid rate throughout its lifetime.
- Without the multiscale framework, GPS performs best when the problem dimension is very small and the budget is large enough. CMA-ES showed excellent performance when the budget is limited. A parallel environment can greatly reduce the time spent for these algorithms. PSO can outperform GPS and CMA-ES in performance if the number of processors is large enough. The choice of the initial guess has a significant effect on the convergence speed for GPS and PSO at the early stage of optimization. CMA-ES, by contrast, is less sensitive to the choice of initial guess. The performance of PSO is affected

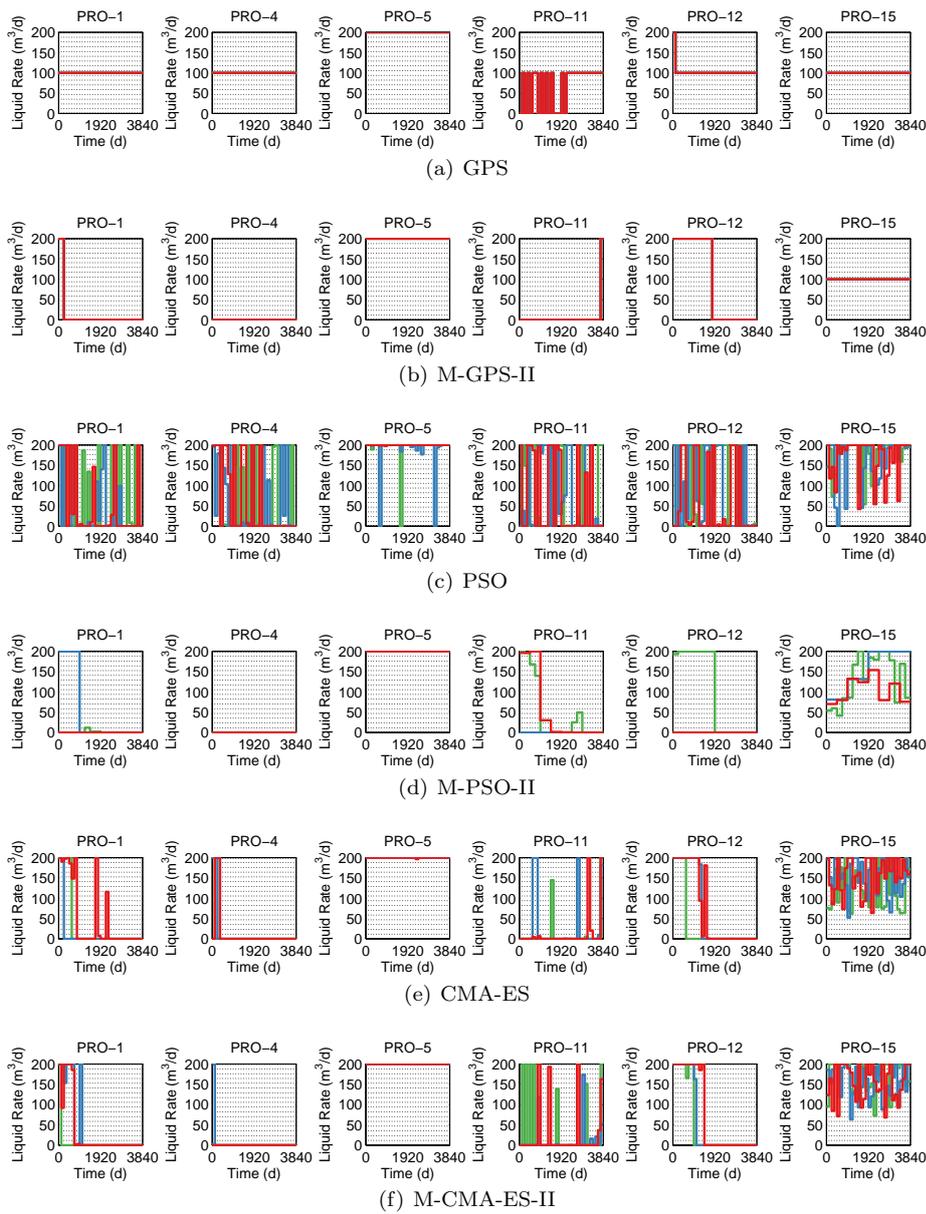


Fig. 25 Optimum well controls for the different optimization approaches for PUNQ-S3 example. Results for 3 trials are shown with different line colors (red, blue, and green) for each stochastic algorithm (PSO, CMA-ES, M-PSO-II, M-CMA-ES-II).

dramatically by the population size. Parameter tuning for CMA-ES showed that the default settings work quite well.

- The multiscale approaches have two advantages in solving well control problem. (1) they provide a way to optimize the control frequency and the well controls simultaneously. (2) when compared to the standalone algorithms the multiscale approach can speed-up the convergence. Based on the results of the test cases, the convergence of GPS and PSO improve the most when combined with the multiscale framework. The multiscale framework is more efficient as the number of control steps increases. The difference in performance between the multiscale hybrid algorithms and the stand-alone algorithms decreases as the number of processors increases.
- The multiscale framework has two key parameters, the choice of the initial number of control steps n_0 and the split factor n_s . The choice $n_0 = 2$ and $n_s = 2$ gave the best performance. In the multiscale framework, M-GPS-II is highly recommended for any computational budget.

All above conclusions are based on the experiments in this paper. Although the multiscale approach has shown its potential to solve complex well control problems, there are still many possible avenues for future work. There is flexibility to choose different n_0 and n_s values for each well. The proposed multiscale approach results in the same number of control steps for each well in the reservoir. In some cases, a different control frequency for individual wells can better balance the production yield and made best use of computational budget. Hence, a heuristic multiscale framework with an adaptive initial number of control steps and splitting factor for each well is suggested for further study. The performance of multiscale approaches with nonlinear constraints still needs additional study. As does the development of robust stopping criteria within the multiscale framework.

Acknowledgements The authors acknowledge funding from the Natural Sciences and Engineering Research Council of Canada (NSERC) Discovery Grant Program and the Natural Science Research Project of Higher Education of Jiangsu, China (No. 17KJB440001).

References

1. Alenezi, F., Mohaghegh, S.: A data-driven smart proxy model for a comprehensive reservoir simulation. In: 2016 4th Saudi International Conference on Information Technology (Big Data Analysis) (KACSTIT), pp. 1–6 (2016). DOI 10.1109/KACSTIT.2016.7756063
2. Almeida, L., Tupac, Y., Pacheco, M.A., Vellasco, M., Lazo, J.: Evolutionary optimization of smart-wells control under technical uncertainties. In: Proceedings of Latin American & Caribbean Petroleum Engineering Conference. Society of Petroleum Engineers (SPE) (2007). DOI 10.2523/107872-ms
3. Asadollahi, M., Nvdal, G., Dadashpour, M., Kleppe, J.: Production optimization using derivative free methods applied to Brugge field case. *Journal of Petroleum Science and Engineering* **114**, 22–37 (2014). DOI 10.1016/j.petrol.2013.12.004
4. Asheim, H.: Maximization of water sweep efficiency by controlling production and injection rates. In: European Petroleum Conference. Society of Petroleum Engineers (1988)
5. Audet, C., Dennis, J.: Analysis of Generalized Pattern Searches. *SIAM Journal on Optimization* **13**(3), 889–903 (2002). DOI 10.1137/S1052623400378742
6. Auger, A., Hansen, N.: A restart CMA evolution strategy with increasing population size. In: The 2005 IEEE Congress on Evolutionary Computation, 2005, vol. 2, pp. 1769–1776 (2005). DOI 10.1109/CEC.2005.1554902

7. Bouzarkouna, Z., Ding, D.Y., Auger, A.: Well placement optimization with the covariance matrix adaptation evolution strategy and meta-models. *Computational Geosciences* **16**(1), 75–92 (2012). DOI 10.1007/s10596-011-9254-2
8. Brouwer, D.R., Jansen, J.D.: Dynamic Optimization of Waterflooding With Smart Wells Using Optimal Control Theory. *SPE Journal* **9**(04), 391–402 (2004)
9. Carosio, G.L.C., Humphries, T.D., Haynes, R.D., Farquharson, C.G.: A closer look at differential evolution for the optimal well placement problem. In: *Proceedings of the 2015 on Genetic and Evolutionary Computation Conference - GECCO'15*. Association for Computing Machinery (ACM) (2015). DOI 10.1145/2739480.2754772
10. Chavent, G.: Identification of functional parameters in partial differential equations. In: *Joint Automatic Control Conference*, pp. 155–156 (1974)
11. Chen, B., Reynolds, A.C.: Ensemble-based optimization of the water-alternating-gas-injection process. *SPE Journal* **21**(03), 786–798 (2016)
12. Chen, B., Reynolds, A.C.: Optimal control of ICV's and well operating conditions for the water-alternating-gas injection process. *Journal of Petroleum Science and Engineering* **149**, 623–640 (2017). DOI 10.1016/j.petrol.2016.11.004
13. Chen, W.H., Gavalas, G.R., Seinfeld, J.H., Wasserman, M.L.: A New Algorithm for Automatic History Matching. *Society of Petroleum Engineers Journal* **14**(06), 593–608 (1974)
14. Chen, Y.: Efficient ensemble based reservoir management. Ph.D. thesis, University of Oklahoma, USA (2008)
15. Chen, Y., Oliver, D.S., others: Ensemble-based closed-loop optimization applied to Brugge field. *SPE Reservoir Evaluation & Engineering* **13**(01), 56–71 (2010)
16. Ciaurri, D.E., Mukerji, T., Durlofsky, L.J.: Derivative-Free Optimization for Oil Field Operations. In: X.S. Yang, S. Koziel (eds.) *Computational Optimization and Applications in Engineering and Industry*, no. 359 in *Studies in Computational Intelligence*, pp. 19–55. Springer Berlin Heidelberg (2011)
17. Clerc, M.: Stagnation analysis in particle swarm optimization or what happens when nothing happens. Tech. Rep. CSM-460, Department of Computer Science, University of Essex (2006)
18. Clerc, M.: Particle swarm optimization, vol. 93. John Wiley & Sons (2010). DOI 10.1002/9780470612163
19. Do, S.T., Reynolds, A.C.: Theoretical connections between optimization algorithms based on an approximate gradient. *Computational Geosciences* **17**(6), 959–973 (2013). DOI 10.1007/s10596-013-9368-9
20. Echeverria Ciaurri, D., Isebor, O.J., Durlofsky, L.J.: Application of derivative-free methodologies to generally constrained oil production optimisation problems. *International Journal of Mathematical Modelling and Numerical Optimisation* **2**(2), 134–161 (2011). DOI 10.1504/IJMMNO.2011.039425
21. van Essen, G., Van den Hof, P., Jansen, J.D.: Hierarchical Long-Term and Short-Term Production Optimization. *SPE Journal* **16**(01), 191–199 (2011). DOI 10.2118/124332-PA
22. Feng, Q., Zhang, J., Zhang, X., Hu, A.: Optimizing well placement in a coalbed methane reservoir using the particle swarm optimization algorithm. *International Journal of Coal Geology* **104**, 34 – 45 (2012)
23. Forouzanfar, F., Poquioma, W.E., Reynolds, A.C.: A covariance matrix adaptation algorithm for simultaneous estimation of optimal placement and control of production and water injection wells. In: *SPE Reservoir Simulation Symposium*. Society of Petroleum Engineers (SPE) (2015). DOI 10.2118/173256-ms
24. Gao, G., Zafari, M., Reynolds, A.C.: Quantifying uncertainty for the PUNQ-s3 problem in a bayesian setting with RML and EnKF. *SPE Journal* **11**(04), 506–515 (2006). DOI 10.2118/93324-pa
25. GeoQuest, S.: ECLIPSE reference manual. Schlumberger, Houston, Texas (2014)
26. Grimstad, B., Foss, B., Heddle, R., Woodman, M.: Global optimization of multiphase flow networks using spline surrogate models. *Computers & Chemical Engineering* **84**, 237–254 (2016). DOI 10.1016/j.compchemeng.2015.08.022
27. Gunnerud, V., Conn, A., Foss, B.: Embedding structural information in simulation-based optimization. *Computers & Chemical Engineering* **53**, 35–43 (2013). DOI 10.1016/j.compchemeng.2013.02.004
28. Hansen, N.: The CMA evolution strategy: A tutorial. <http://www.lri.fr/hansen/cmaesintro.html>

29. Hansen, N., Niederberger, A., Guzzella, L., Koumoutsakos, P.: A method for handling uncertainty in evolutionary optimization with an application to feedback control of combustion. *IEEE Transactions on Evolutionary Computation* **13**(1), 180–197 (2009). DOI 10.1109/tevc.2008.924423
30. Hansen, N., Ostermeier, A.: Completely derandomized self-adaptation in evolution strategies. *Evolutionary Computation* **9**(2), 159–195 (2001). DOI 10.1162/106365601750190398
31. Hansen, N., Ros, R.: Benchmarking a weighted negative covariance matrix update on the BBOB-2010 noiseless testbed. In: *Proceedings of the 12th annual conference on Genetic and evolutionary computation - GECCO'10*, pp. 1673–1680. Association for Computing Machinery (ACM) (2010). DOI 10.1145/1830761.1830788
32. Hasan, A., Foss, B., Sagatun, S.: Flow control of fluids through porous media. *Applied Mathematics and Computation* **219**(7), 3323–3335 (2012). DOI 10.1016/j.amc.2011.07.001
33. Helwig, S., Neumann, F., Wanka, R.: Particle swarm optimization with velocity adaptation. In: *2009 International Conference on Adaptive and Intelligent Systems*, pp. 146–151. Institute of Electrical & Electronics Engineers (IEEE) (2009). DOI 10.1109/icaais.2009.32
34. Humphries, T.D., Haynes, R.D.: Joint optimization of well placement and control for nonconventional well types. *Journal of Petroleum Science and Engineering* **126**, 242–253 (2015). DOI 10.1016/j.petrol.2014.12.016
35. Humphries, T.D., Haynes, R.D., James, L.A.: Simultaneous and sequential approaches to joint optimization of well placement and control. *Computational Geosciences* **18**(3-4), 433–448 (2013). DOI 10.1007/s10596-013-9375-x
36. Huyer, W., Neumaier, A.: Global optimization by multilevel coordinate search. *Journal of Global Optimization* **14**(4), 331–355 (1999). DOI 10.1023/a:1008382309369
37. Isebor, O.J., Ciaurri, D.E., Durlafsky, L.J.: Generalized field-development optimization with derivative-free procedures. *SPE Journal* **19**(05), 891–908 (2014). DOI 10.2118/163631-pa
38. Isebor, O.J., Durlafsky, L.J., Ciaurri, D.E.: A derivative-free methodology with local and global search for the constrained joint optimization of well locations and controls. *Computational Geosciences* **18**(3-4), 463–482 (2014). DOI 10.1007/s10596-013-9383-x
39. Kampstra, P.: Beanplot: A boxplot alternative for visual comparison of distributions. *Journal of Statistical Software* **28**(1) (2008). DOI 10.18637/jss.v028.c01
40. Kennedy, J.: Particle Swarm Optimization. In: C. Sammut, G.I. Webb (eds.) *Encyclopedia of Machine Learning*, pp. 760–766. Springer Science + Business Media (2011). DOI 10.1007/978-0-387-30164-8
41. Knudsen, B.R., Foss, B.: Designing shale-well proxy models for field development and production optimization problems. *Journal of Natural Gas Science and Engineering* **27**, 504–514 (2015). DOI 10.1016/j.jngse.2015.08.005
42. Li, L., Jafarpour, B.: A variable-control well placement optimization for improved reservoir development. *Computational Geosciences* **16**(4), 871–889 (2012)
43. Li, L., Jafarpour, B., Mohammad-Khaninezhad, M.R.: A simultaneous perturbation stochastic approximation algorithm for coupled well placement and control optimization under geologic uncertainty. *Computational Geosciences* **17**(1), 167–188 (2012). DOI 10.1007/s10596-012-9323-1
44. Lien, M.E., Brouwer, D.R., Mannseth, T., Jansen, J.D.: Multiscale regularization of flooding optimization for smart field management. *SPE Journal* **13**(02), 195–204 (2008). DOI 10.2118/99728-pa
45. Loshchilov, I.: CMA-ES with restarts for solving CEC 2013 benchmark problems. In: *2013 IEEE Congress on Evolutionary Computation*. Institute of Electrical & Electronics Engineers (IEEE) (2013). DOI 10.1109/cec.2013.6557593
46. Narasingam, A., Kwon, J.S.I.: Development of local dynamic mode decomposition with control: Application to model predictive control of hydraulic fracturing. *Computers & Chemical Engineering* **106**, 501–511 (2017). DOI 10.1016/j.compchemeng.2017.07.002
47. Narasingam, A., Siddhamshetty, P., Kwon, J.S.I.: Temporal clustering for order reduction of nonlinear parabolic PDE systems with time-dependent spatial domains: Application to a hydraulic fracturing process. *AIChE Journal* **63**(9), 3818–3831 (2017). DOI 10.1002/aic.15733
48. Nocedal, J., Wright, S.: *Numerical optimization*. Springer Science & Business Media (2006)
49. Oliveira, D.F., Reynolds, A.: An adaptive hierarchical multiscale algorithm for estimation of optimal well controls. *SPE Journal* **19**(05), 909–930 (2014). DOI 10.2118/163645-pa
50. Oliveira, D.F., Reynolds, A.: Hierarchical multiscale methods for life-cycle production optimization: A field case study. In: *SPE Reservoir Simulation Symposium*. Society of Petroleum Engineers (SPE) (2015). DOI 10.2118/173273-ms

51. Onwunalu, J.E., Durlofsky, L.: A new well-pattern-optimization procedure for large-scale field development. *SPE Journal* **16**(03), 594–607 (2011). DOI 10.2118/124364-pa
52. Onwunalu, J.E., Durlofsky, L.J.: Development and application of a new well pattern optimization algorithm for optimizing large scale field development. In: *SPE Annual Technical Conference and Exhibition*. Society of Petroleum Engineers (SPE) (2009). DOI 10.2118/124364-ms
53. Pajonk, O., Schulze-riegert, R., Krosche, M., Hassan, M., Nwakile, M.M., others: Ensemble-based water flooding optimization applied to mature fields. In: *SPE Middle East Oil and Gas Show and Conference*. Society of Petroleum Engineers (2011)
54. Perez, R.E., Behdinan, K.: Particle swarm approach for structural design optimization. *Computers & Structures* **85**(1920), 1579–1588 (2007). DOI 10.1016/j.compstruc.2006.10.013
55. Sarma, P., Aziz, K., Durlofsky, L.J.: Implementation of adjoint solution for optimal control of smart wells. In: *SPE Reservoir Simulation Symposium*. Society of Petroleum Engineers (2005)
56. Schulze-riegert, R., Bagheri, M., Krosche, M., Kueck, N., Ma, D., others: Multiple-objective optimization applied to well path design under geological uncertainty. In: *SPE Reservoir Simulation Symposium*. Society of Petroleum Engineers (2011)
57. Shuai, Y., White, C.D., Zhang, H., Sun, T.: Using multiscale regularization to obtain realistic optimal control strategies. In: *SPE Reservoir Simulation Symposium*. Society of Petroleum Engineers (SPE) (2011). DOI 10.2118/142043-ms
58. Siddhamshetty, P., Kwon, J.S.I.: Model-based feedback control of oil production in oil-rim reservoirs under gas coning conditions. *Computers & Chemical Engineering* **112**, 112–120 (2018). DOI 10.1016/j.compchemeng.2018.02.001
59. Sidhu, H.S., Narasingam, A., Siddhamshetty, P., Kwon, J.S.I.: Model order reduction of nonlinear parabolic PDE systems with moving boundaries using sparse proper orthogonal decomposition: Application to hydraulic fracturing. *Computers & Chemical Engineering* **112**, 92–100 (2018). DOI 10.1016/j.compchemeng.2018.02.004
60. Storn, R., Price, K.: Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization* **11**(4), 341–359 (1997). DOI 10.1023/a:1008202821328
61. Sudaryanto, B.: Optimization of displacement efficiency of oil recovery in porous media using optimal control theory. Ph.D. thesis, University of Southern California (1999)
62. Torczon, V.: On the Convergence of Pattern Search Algorithms. *SIAM Journal on Optimization* **7**(1), 1–25 (1997). DOI 10.1137/S1052623493250780
63. Trehan, S., Durlofsky, L.J.: Trajectory piecewise quadratic reduced-order model for sub-surface flow, with application to PDE-constrained optimization. *Journal of Computational Physics* **326**, 446–473 (2016). DOI 10.1016/j.jcp.2016.08.032
64. Vaz, A.I.F., Vicente, L.N.: A particle swarm pattern search method for bound constrained global optimization. *Journal of Global Optimization* **39**(2), 197–219 (2007). DOI 10.1007/s10898-007-9133-5
65. Virnovsky, G.A.: Waterflooding strategy design using optimal control theory (1991). DOI 10.3997/2214-4609.201411270
66. Wang, C., Li, G., Reynolds, A.C.: Production optimization in closed-loop reservoir management. *SPE Journal* **14**(03), 506–523 (2009). DOI 10.2118/109805-pa
67. Wang, X., Haynes, R.D., Feng, Q.: A multilevel coordinate search algorithm for well placement, control and joint optimization. *Computers & Chemical Engineering* **95**, 75–96 (2016). DOI <https://doi.org/10.1016/j.compchemeng.2016.09.006>
68. Willjuice Iruthayarajan, M., Baskar, S.: Covariance matrix adaptation evolution strategy based design of centralized PID controller. *Expert Systems with Applications* **37**(8), 5775–5781 (2010). DOI 10.1016/j.eswa.2010.02.031
69. Yin, S., Cagan, J.: An extended pattern search algorithm for three-dimensional component layout. *Journal of Mechanical Design* **122**(1), 102–108 (2000). DOI 10.1115/1.533550
70. Zakirov, I., Aanonsen, S.I., Zakirov, E.S., Palatnik, B.M.: Optimizing reservoir performance by automatic allocation of well rates. In: *5th European Conference on the Mathematics of Oil Recovery* (1996)
71. Zhou, K., Hou, J., Zhang, X., Du, Q., Kang, X., Jiang, S.: Optimal control of polymer flooding based on simultaneous perturbation stochastic approximation method guided by finite difference gradient. *Computers & Chemical Engineering* **55**, 40–49 (2013). DOI 10.1016/j.compchemeng.2013.04.009