SPE-212228-MS

# Integrated Framework for Optimization of Horizontal/Deviated Well Placement and Control for Geological $CO_2$ Storage

Amy Zou and Louis J. Durlofsky, Stanford University



This paper was prepared for presentation at the SPE Reservoir Simulation Conference held in Galveston, Texas, USA, 28–30 March 2023.





## Abstract

A general framework for optimizing the locations and time-varying injection rates of a set of monobore wells for geological carbon storage is presented and applied. Two objective functions, minimization of mobile $CO_2$ fraction at the end of the operation, and maximization of storage efficiency, are considered. Appropriate linear and nonlinear constraints, involving the geometry of the well configuration, injection rates, and injected mass (for pressure management), are specified. Two derivative-free algorithms, particle swarm optimization (PSO) and differential evolution (DE), are applied and assessed. The various constraints are treated using a preprocessing repair procedure, penalty functions, and a filter method. The framework utilizes multifidelity optimization, in which increasing levels of grid resolution are applied during the course of the optimization run. For the minimization of mobile $CO_2$ fraction, the multifidelity approach is compared with high-resolution optimization. This treatment is shown to outperform high-resolution PSO and DE optimization in terms of both solution quality and computational requirements. The multifidelity DE optimization case provides the best (feasible) solution, with 0.090 mobile $CO_2$ fraction at 200 years, which represents a 68% improvement over a heuristic base-case. For the second objective function, multifidelity PSO provides a design that results in a storage efficiency of 0.074, which is about double the base-case value. For both objective functions, the optimized solutions contain horizontal and deviated wells placed near the bottom of the storage aquifer. The well configurations are much different for the two objective functions, with wells more closely spaced, resulting in a single merged plume, for the storage efficiency maximization case. For the mobile $CO_2$ minimization case, by contrast, wells are separated and pulsed, which facilitates dissolution and residual trapping.

## Introduction

Carbon capture and storage (CCS) is an essential component in many future low-carbon energy scenarios. Under the Net Zero scenario, for example, CCS capacity must increase from its current level of 0.044 gigatonnes in 2022 to 7.6 gigatonnes in 2050 (International Energy Agency., 2022). Geologically sequestered $CO_2$ is contained through four trapping mechanisms, referred to as stratigraphic, residual, dissolution, and mineral trapping (Smit et al., 2014). Mineralization occurs over very long time scales in many geological settings, so this trapping mechanism will not be considered in this work. Different

operational strategies (well locations and injection rates) lead to varying amounts of stratigraphic trapping relative to residual and dissolution trapping, as well as to large variations in storage efficiency (pore-space utilization) and plume footprint. Therefore, the objectives of the storage operation impact the design and real-time management. Objectives can include minimizing the fraction of $CO_2$ trapped stratigraphically (referred to as mobile $CO_2$), or maximizing storage efficiency. These and other specifications lead to challenging optimization problems involving expensive function evaluations and multiple nonlinear constraints.

In this paper, we develop and apply a general framework to optimize the locations and time-varying $CO_2$ injection rates for multiple 3D horizontal or deviated wells. Realistic nonlinear constraints, which act to provide acceptable well configurations and limit the amount of pressure buildup in the formation, are incorporated. Different objective functions, derivative-free optimization algorithms, and optimization strategies, including multifidelity optimization, are assessed. The framework builds on an earlier workflow applied for oil field problems (Zou et al., 2022). This workflow includes a 'geometric repair' treatment, in which infeasible well configurations are modified prior to flow simulation.

A number of previous investigators have developed optimization procedures for $CO_2$ storage applications. For optimization problems involving only well control (meaning well locations were fixed), the objective function is often continuous, which enables the use of gradient-based methods. Shamshiri and Jafarpour (2012), for example, used the BFGS algorithm to optimize the injection rates for vertical injectors, with the goals of minimizing mobile $CO_2$ volume and sweep efficiency (separately). Chen and Pawar (2019) applied a stochastic simplex approximate gradient (StoSAG) method to optimally control a set of vertical wells. Their objective was to maximize the life-cycle NPV for $CO_2$ EOR with storage (for which a $CO_2$ tax credit was applied). Derivative-free methods have also been used for these problems. Jin and Durlofsky (2018) utilized mesh adaptive direct search (MADS) to minimize the mole fraction of $CO_2$ in the confinement zone by controlling injection rates for multiple horizontal wells. Zhang and Agarwal (2012) optimized settings for both vertical and horizontal wells for $CO_2$ water-alternating-gas operations with storage. They used a genetic algorithm to minimize a metric involving the $CO_2$ plume radius. Park et al. (2021) applied the nondominated sorting genetic algorithm (NSGA-II) to find the optimal trade-offs between the cost associated with injection pressure and the amount of $CO_2$ stored. They optimized (constant-in-time) rates for vertical injectors.

The optimization of well locations (configuration), or the joint optimization of well locations and controls, is more challenging because the objective function can be strongly discontinuous due to geological heterogeneity. In such cases, derivative-free algorithms may be more attractive than gradient-based methods. Cameron and Durlofsky (2012) utilized Hooke–Jeeves direct search (HJDS) to optimize the locations and controls of multiple horizontal $CO_2$ injectors, with the goal of minimizing mobile $CO_2$. Stopa et al. (2016) applied both a genetic algorithm (GA) and particle swarm optimization (PSO) to optimize the locations and controls of vertical $CO_2$ injectors to minimize $CO_2$ in the top layer of the model. Goda and Sato (2013) applied iterative Latin hypercube sampling (ILHS) to maximize immobile $CO_2$ by optimizing the locations and rates for vertical injectors. Zheng et al. (2021) used controlled NSGA-II to solve multiobjective problems with coupled flow and geomechanics. Their work involved vertical $CO_2$ injection wells, with the objectives of maximizing the mass of $CO_2$ injected while minimizing quantities associated with geomechanical risk. Other algorithms, such as differential evolution, DE (Cihan et al., 2015), and covariance matrix adaptation evolution strategy, CMA-ES (Miyagi et al., 2018), have also been applied for $CO_2$ storage operations.

Simulations involving $CO_2$ injection are often sensitive to the level of grid refinement. Yamamoto and Doughty (2011) and Youssef et al. (2021), for example, demonstrated that low-resolution models can considerably underestimate the areal extent of the plume. Li et al. (2013) considered cases involving different grid resolutions for different capillary pressure models and showed that more finely gridded models displayed faster $CO_2$ dissolution. Shao et al. (2022) constructed reservoir models from high-resolution

geomodels generated from field data. They identified factors that influence plume shape, extent, and internal saturation distribution. Because fine-scale simulation of $CO_2$ storage problems is very expensive computationally, highly resolved models are not practical for use in derivative-free optimization, where thousands of function evaluations may be required. Thus multifidelity treatments, in which a succession of models with increasing resolution/accuracy is used over the course of the optimization, may be appropriate for these problems. Multifidelity optimization has been successfully applied in oil field problems (Aliyev and Durlofsky, 2017) and has been considered to a limited extent for $CO_2$ storage (Cameron and Durlofsky, 2012). If a highly resolved geological description is available, multifidelity procedures can make use of properly upscaled models. Aliyev and Durlofsky (2017), for example, used global single-phase transmissibility upscaling in this setting. Two-phase upscaling procedures, such as those applied by Rabinovich et al. (2015) or Mishra et al. (2020), are more accurate and may be even more effective.

Constraints appear in practical optimization problems, and it is important that these are handled efficiently. Bound constraints on the decision variables (e.g., maximum and minimum injection rate) can be easily accommodated, whereas linear and nonlinear constraints require additional treatments. In well control optimization for $CO_2$ storage, pressure buildup appears as a nonlinear constraint that can only be evaluated after the simulation is performed. Allen et al. (2017) and Miyagi et al. (2018) treated this constraint using a penalty method. Cihan et al. (2015) optimized the controls for brine extraction with a constraint specifying no $CO_2$ breakthrough. They used a modified DE algorithm with infeasible solutions penalized during the selection process. Constrained problems can also be formulated in the context of multiobjective optimization. Park et al. (2021), for example, considered two objectives – maximizing the amount of trapped $CO_2$, and minimizing a metric associated with well bottomhole pressures (BHPs).

Previous studies have demonstrated the advantages of using horizontal wells for $CO_2$ storage problems (Al-Khdheeawi et al., 2017; Jun et al., 2019). It is, therefore, reasonable to introduce additional generality by incorporating deviated wells. As far as we are aware, optimization of injection scenarios involving deviated wells has not been previously considered for CCS operations. Constraints on the arrangement of such wells, e.g., minimum well-to-well and well-to-boundary distance, can be challenging to satisfy for arbitrary 3D configurations. Here we treat general monobore wells, which can be horizontal or deviated. We apply the general repair procedure originally introduced by Volkov and Bellout (2018) to reduce or eliminate geometric infeasibility. This approach was incorporated into derivative-free optimization by Zou et al. (2022), where it was shown to significantly outperform a standard penalty-method treatment for the cases considered.

In this study, we generalize the optimization framework developed by Zou et al. (2022) to treat objective functions and constraints applicable in $CO_2$ storage problems. We also introduce a multifidelity workflow to accelerate the optimization runs. The optimizations involve the determination of the locations and time-varying injection rates for multiple wells, which can be horizontal or deviated. The objective functions considered are the minimization of the mobile $CO_2$ fraction (which corresponds to maximizing residual and dissolution trapping), and the maximization of storage efficiency. Constraints include maintaining the target field injection rate, limiting the injection well BHPs, and satisfying specifications on well-configuration geometry. Both PSO and DE optimization algorithms are applied and evaluated. The target problems involve the injection of 4 megatonnes $CO_2$ per year, for 30 years, through four wells. To our knowledge, geological $CO_2$ storage optimization with multiple deviated wells and realistic objective functions and constraints has not been considered in previous studies. The use of multifidelity treatments in this setting is also new.

This paper proceeds as follows. In the next section, we define the optimization problem and objective functions considered in this work, describe the integrated optimization framework, constraint handling methods and multifidelity treatment, and present the aquifer model used in the examples. Next, we present results using PSO and DE to determine the optimal locations and controls for four horizontal or deviated $CO_2$ injectors. Two different objective functions are considered, and multifidelity results are compared to

those from high-resolution optimization (for the first objective function). We conclude with a summary and suggestions for future work in this area.

## Methodology and Aquifer Model

In this section, we first define the optimization problem along with the objective functions and constraints applied in this study. We then introduce the integrated framework for optimization, which includes the optimization algorithms and constraint handling methods. Next, we describe the aquifer model used in this study. Finally, grid-refinement results for a base-case setup are presented.

### Problem Statement and Objective Function

The optimization problem entails determining the locations and time-varying injection rates for a set of monobore (linear) wells. There are $N_w$ wells, each defined by the ($x$, $y$, $z$) values of their two endpoints. Injection fractions for each well, in $N_t$ control periods, will also be determined. The optimization problem is written as:

$$\min_{\mathrm{u}\in U} f(\mathbf{x}, \mathbf{u}), \tag{1a}$$

$$\text{s.t.}\, \mathbf{g}(\mathbf{x}, \mathbf{u}) = 0, \quad \mathbf{C}(\mathbf{u}) = 0. \tag{1b}$$

In Eq. 1, $f$ is the objective function we seek to minimize. The vector $\mathrm{u} \in U \subset \mathbb{R}^n$ ($n = 6N_w + N_w \times N_t$) contains the set of decision variables. This can be expressed as $\mathbf{u} = [\mathbf{u}_l, \mathbf{u}_c]$, where $\mathbf{u}_l \in \mathbb{R}^{6N_w}$ is the set of well locations and $\mathbf{u}_c \in \mathbb{R}^{N_w \times N_t}$ is the set of time-varying injection fractions (controls). The subspace $U$ contains all bound constraints imposed on the well locations and controls. The numerical solution g is a function of both the state variables x and well variables u. Constraints (linear and nonlinear) are represented by C, which depends on both well locations and controls. Constraints are satisfied when C = 0. The constraints considered in this study will be described in the next section.

The well location variable $\mathrm{u}_l$ can be further expanded as $\mathrm{u}_l = [\tilde{\mathrm{u}}_1, \ldots, \tilde{\mathrm{u}}_i, \ldots, \tilde{\mathrm{u}}_{N_w}]$, where $\tilde{\mathrm{u}}_i \in \mathbb{R}^6$ and $i = 1, \ldots, N_w$. Each well $\tilde{\mathrm{u}}_i$ is treated as a linear segment in 3D, defined by the coordinates of its heel $\tilde{\mathbf{u}}_i^h = \left[u_{i,x}^h, u_{i,y}^h, u_{i,z}^h\right]$ and toe $\tilde{\mathbf{u}}_i^t = \left[u_{i,x}^t, u_{i,y}^t, u_{i,z}^t\right]$. Wells are considered to be perforated along the full heel-to-toe trajectory. The determination of the intersected grid blocks and the computation of well index follow the approaches described by Volkov and Bellout (2018). The well control variables $\mathrm{u}_c$ can be expressed as $\mathrm{u}_c = [\mathrm{u}^\circ_1, \ldots, \mathrm{u}^\circ_i, \ldots, \mathrm{u}^\circ_{N_w}]$, where $\mathrm{u}^\circ_i \in \mathbb{R}^{N_t}$ is the set of injection fractions for well $i$ over $N_t$ control periods. Hence, $\mathrm{u}^\circ_i = [u_{i,1}, \ldots, u_{i,t}, \ldots, u_{i,N_t}]$ and $t = 1, \ldots, N_t$. A constraint will be introduced to assure the injection fractions sum to unity at each control period.

Two different objective functions will be considered in this study. The first objective is to minimize the mobile $CO_2$ fraction in the system at the end of the simulation time frame. This acts to facilitate dissolution and residual trapping, which are potentially 'safer' storage mechanisms. This objective function, denoted $f_1$, is computed as

$$f_1 = \left(\frac{m_m}{m_t}\right)_{T_f}, \tag{2}$$

where $m_m$ is the mass (in kg-mol) of mobile $CO_2$ in the full system at final simulation time $T_f$, and $m_t$ is the total mass of $CO_2$ injected into the system. Here we take $T_f = 200$ years.

The second objective is to maximize $CO_2$ storage efficiency. This can be quantified in various ways – here we seek to maximize the ratio between the gas phase $CO_2$ volume and an appropriate reservoir volume. Similar to the treatments of Wen et al. (2021) and Bachu (2015), we define this objective, denoted $f_2$, as

$$f_2 = -\left(\frac{\sum_{i\in D_{sa}} V_i \phi_i S_i}{\sum_{j\in D_{fr}} V_j \phi_j}\right)_{T_f}. \quad (3)$$

In Eq. 3, the numerator is the total pore volume containing $CO_2$ in the gas phase, where $V_i$, $\phi_i$, and $S_i$ represent the bulk volume, porosity, and $CO_2$ gas saturation in block $i \in D_{sa}$ at the end of the simulation run (200 years). Here $D_{sa}$ denotes the storage aquifer and $i$ is any grid block in the storage aquifer with $S_i > 10^{-5}$. The denominator is the reservoir volume associated with the the areal footprint of the plume. The areal footprint is determined by first identifying the plume in each layer. Here the plume is taken to be all cells with $S_i > 10^{-5}$. The rectangle in the $x$-$y$ plane that contains the plume, in all layers, is the areal footprint. This is illustrated schematically for a two-layer system in Fig. 1. Here the blue circle represents the plume in one layer, and the yellow circle represents the plume in the other layer. The red rectangle, which encompasses both circles, defines the areal footprint. The footprint region $D_{fr}$ is then the brick-shaped domain formed by vertically projecting the areal footprint (red rectangle in Fig. 1) through the entire aquifer in the $z$-direction. Quantities $V_j$ and $\phi_j$ are the bulk volume and porosity for block $j$ in the footprint region $D_{fr}$. Since the optimization problem in Eq. 1 is defined as a minimization and we seek to maximize storage efficiency, a negative sign is applied in the definition of $f_2$.

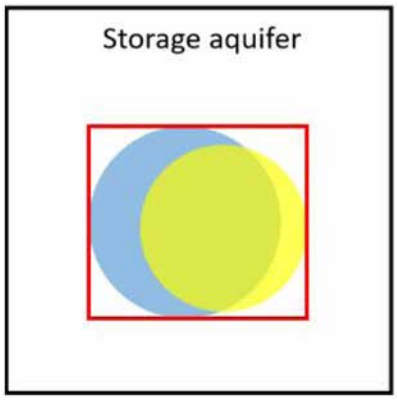



**Figure 1—Areal footprint of the $CO_2$ plume in a two-layer system. Blue and yellow circles indicate the plume in the two layers, and the red rectangle is the areal footprint. The footprint region is formed by vertically projecting the areal footprint.**

## Optimization Constraints

Constraints on the geometry of the well configuration, on the volume/mass of $CO_2$ injected at each time step, and on the maximum bottomhole pressure (BHP) of each injection well must be satisfied in the optimizations. The treatments for these constraints are considered in turn.

The well geometric constraints considered in this study include minimum and maximum well length, minimum interwell distance, and minimum well-to-boundary distance. These constraints depend only on the well location variables $u_l$, so they can be evaluated prior to performing flow simulation. The constraint violations are measured in meters, and we assign a value of zero to the constraint if there is no violation. We will now provide brief descriptions of these geometric constraints. Further details are provided by Zou et al. (2022).

The minimum and maximum allowable well lengths are denoted $L_{wl}^{min}$ and $L_{wl}^{max}$. For each proposed well $\tilde{u}_i$, the well length $L_{i,wl}$ is simply the Euclidean distance between the two endpoints. The minimum and maximum well length constraints, denoted by $C_{i,wl}^{min}$ and $C_{i,wl}^{max}$, are computed as

$$C_{i,wl}^{min} = \begin{cases} L_{wl}^{min} - L_{i,wl} & \text{if } L_{wl}^{min} > L_{i,wl}, \\ 0 & \text{if } L_{wl}^{min} \le L_{i,wl}, \end{cases} \quad (4a)$$

$$C_{i,wl}^{max} = \begin{cases} L_{i,wl} - L_{wl}^{max} & \text{if } L_{i,wl} > L_{wl}^{max}, \\ 0 & \text{if } L_{i,wl} \le L_{wl}^{max}. \end{cases} \tag{4b}$$

The minimum interwell distance constraint prevents wells from intersecting or being placed too close to each other. We apply the segment-to-segment squared-distance algorithm (Schneider and Eberly, 2003) to compute the shortest distance between each pair of wells. Once these pairwise distances are computed, we can identify the minimum interwell distance between well $i$ and all other wells, which we denote $L_{i,iw}$. Given a user-specified minimum interwell distance $L_{iw}^{min}$, the constraint violation $C_{i,iw}^{min}$ is given by

$$C_{i,iw}^{min} = \begin{cases} L_{iw}^{min} - L_{i,iw} & \text{if } L_{iw}^{min} > L_{i,iw}, \\ 0 & \text{if } L_{iw}^{min} \le L_{i,iw}. \end{cases} \tag{5}$$

The minimum well-to-boundary (or well-to-perimeter) distance constraint forces wells to be placed within a specified region. This constraint can be converted to a bound constraint for rectangular-shaped domains (as are considered in this study), though in the general case of nonconvex domains it is nonlinear. Here this constraint is computed using the signed-distance method. With this approach, the set of midpoints on boundary-block faces is denoted as a. Each point in a is associated with its normal vector. For each well $\tilde{u}_i$, we identify two points $a_i^h$ and $a_i^t$ within a that have the shortest distance to $\tilde{u}_i^h$ and $\tilde{u}_i^t$, respectively. Two vectors, $\left(a_i^h - \tilde{u}_i^h\right)$ and $\left(a_i^t - \tilde{u}_i^t\right)$, are constructed. The dot product of each vector with the corresponding normal vector (associated with $a_i^h$ and $a_i^t$) is then computed. The signs of the dot products are assigned to $\| \left(\tilde{u}_i^h - a_i^h\right) \|_2$ and $\| \left(\tilde{u}_i^t - a_i^t\right) \|_2$, respectively. The well-to-boundary distance $L_{i,wp}$ is taken to be the minimum of these two quantities. The constraint violation $C_{i,wp}^{min}$ is then given by

$$C_{i,wp}^{min} = \begin{cases} L_{wp}^{min} - L_{i,wp} & \text{if } L_{wp}^{min} > L_{i,wp}, \\ 0 & \text{if } L_{wp}^{min} \le L_{i,wp}, \end{cases} \tag{6}$$

where $L_{wp}^{min}$ is the specified minimum well-to-boundary distance.

We combine the geometric constraints into a total geometric constraint violation $Q$ (in units of meters) and a normalized total geometric constraint violation $H$ (dimensionless). These are defined as follows. For each well $\tilde{u}_i$, $i = 1, \ldots, N_w$, we construct a constraint violation vector $C_i = [C_{i,1}, \ldots, C_{i,j}, \ldots, C_{i,N_c}]$, where $N_c = 4$ is the number of geometric constraints considered in this study. The total constraint violations $Q$ and $H$, for a full candidate solution $u_l \in \mathbb{R}^{6N_w}$, are then defined as

$$Q = \epsilon + \sum_{i=1}^{N_w} \| \mathbf{C}_i \|_1. \tag{7}$$

$$H = \sum_{i=1}^{N_w} \sum_{j=1}^{N_c} \frac{C_{i,j}}{\max\left(1, L_j^{lim}\right)}. \tag{8}$$

In Eq. 7, $\epsilon = 10^{-4}$ m is an arbitrary small constant introduced to enable us to plot log $Q$. Hence, $Q = 10^{-4}$ m indicates feasibility in terms of the the geometric constraints. The normalized total constraint violation $H$ will be used in the penalty function described later. In Eq. 8, $L_j^{lim}$ denotes the user-specified limit for each constraint. The max(1, $L_j^{min}$) term is used as a normalizing factor to avoid division by zero.

We now describe the constraints on $CO_2$ injection and BHPs. The total injection rate target, specified in terms of surface rate, is prescribed for each control period (here it is constant in time). For each injection fraction in the set of control variables for well $i$, $u^{\circ}{}_i = [u_{i,1}, \ldots, u_{i,t}, \ldots, u_{i,N_t}]$, $t = 1, \ldots, N_t$, we apply minimum and maximum bound constraints of 0 and 1. This means a well could potentially inject anywhere from 0 to 100% of the total injection target during an injection period. We also impose a linear equality constraint that forces the injection rate fractions from all wells, at each control period, to sum to 1, i.e., $\Sigma_{i=1}^{N_w} u_{i,t} = 1$, for $t = 1, \ldots, N_t$. This constraint is independent of the state variables x and can be evaluated prior to flow simulation. The constraint violation is expressed as

$$C_{t,wr} = \left|1 - \sum_{i=1}^{N_w} u_{i,t}\right|, \quad t = 1, \ldots, N_t. \tag{9}$$

This constraint can also be formulated such that only $N_w - 1$ variables need be determined in each control period. The treatment applied here, however, is straightforward and easily implemented with the repair procedure described later.

Various heuristics have been introduced for the maximum allowable injection pressure relative to the initial pressure (Allen et al., 2017; Wang et al., 2013; Zhou et al., 2008). Following suggestions in these studies, we limit the BHP for each well to a maximum of 150% of the initial pressure at the top of the storage aquifer. With this specification, a well can switch from rate control to BHP control during the simulation run. This can lead to field-wide $CO_2$ injection that is below the target, which will result in $C_{t,wr} > 0$.

We also wish to ensure that all injected $CO_2$ is retained within the storage aquifer rather than flowing into the boundary region (these model regions will be defined later). To accomplish this, the constraint quantity $C_{aq}$ is defined as

$$C_{aq} = \frac{m_{tg} - m_{in} + m_{out}}{m_{tg}}. \tag{10}$$

Here $m_{tg}$ is the total mass prescribed to be injected over the full injection period, $m_{in}$ is the mass of injected $CO_2$ in the full system, and $m_{out}$ is the mass of $CO_2$ that has flowed out of the storage aquifer. Satisfying this constraint means we meet the total injection target and all $CO_2$ is retained in the storage aquifer. Because this constraint is dependent on the state variables x, it can only be evaluated after flow simulations are performed. In this study, we set a feasibility threshold for $C_{aq}$ of $10^{-5}$, meaning $C_{aq} > 10^{-5}$ is considered infeasible.

## Integrated Framework for Optimization and Constraint Handling

We begin this section with a description of the integrated optimization framework. Next, the constraint handling methods applied to satisfy the various constraints are presented. The core optimization algorithms (PSO and DE) are then discussed briefly. Finally, we describe the multifidelity optimization treatment implemented in this work.

***Overview of Optimization Framework.*** An overview of the integrated framework is shown in Fig. 2. This is similar to the workflow described by Zou et al. (2022), though there are some differences due to the appearance of additional constraints. During initialization, algorithmic hyperparameters, initial solutions, parameters for constraint treatments and termination criteria are specified. As noted earlier, some of the constraints are independent of the state variables x, and these can be evaluated (and treated) prior to performing any flow simulations. These constraints, which include $C_{i,wl}^{max}$, $C_{i,wl}^{min}$, $C_{i,iw}^{min}$, $C_{i,wp}^{min}$, and $C_{t,wr}$, are computed for each of the proposed solutions. If a solution contains constraint violations, we apply the repair procedure (described later) to eliminate or reduce these violations.

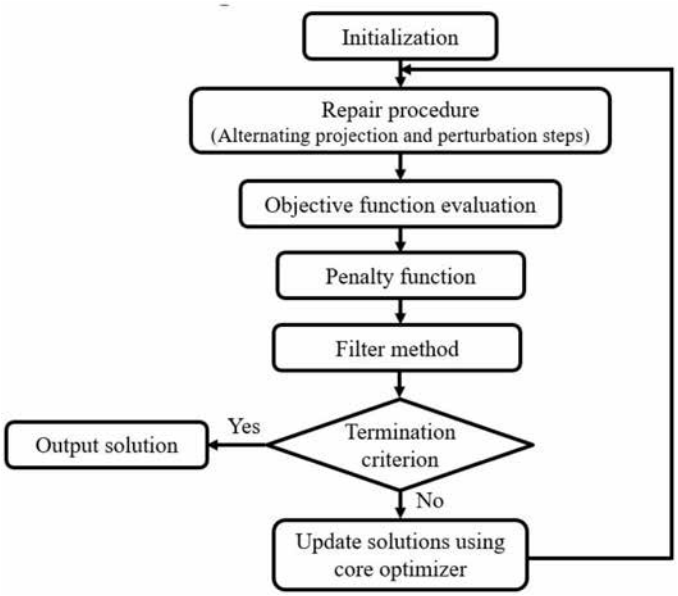



**Figure 2—Overall optimization workflow.**

Next, using the repaired solutions, we perform flow simulations and compute objective function values. A penalty function is applied to penalize solutions that still contain some amount of infeasibility in the geometric constraints. Additional treatment of $C_{t,wr}$ is not required as this is a linear constraint that is always satisfied after application of the repair procedure. The constraint $C_{aq}$, which does depend on x, is then evaluated. Because this is a nonlinear output constraint, a filter method is applied to achieve feasibility. Finally, the repaired solutions are updated using the core optimization algorithm. The workflow proceeds until a termination criterion is satisfied.

***Constraint Handling.*** The repair procedure applied to treat geometric constraints was introduced by Volkov and Bellout (2018) and was later applied in a derivative-free optimization framework by Zou et al. (2022). The repair entails the solution of a bi-objective optimization problem where the goal is to find a 'repaired' version $u_r$ of the original solution u (proposed by the core optimizer) that minimizes both the total geometric constraint violation and the distance between the repaired and original solutions. Sequential quadratic programming (SQP) is applied for this optimization, with each constraint treated independently. Even after repair, the well configuration may still contain a degree of constraint violation, since the allowable distance from the original solution is limited. An additional penalty is thus applied to increase (for a minimization problem) the objective function value. The penalized objective functions, denoted $Y_1$ and $Y_2$, for the two objective functions considered in this study ($f_1$ and $f_2$ in Eqs. 2 and 3) are computed as:

$$y_1 = f_1 \times \left(1 + \frac{H(u_r)}{\zeta}\right), \tag{11a}$$

$$y_2 = f_2 \times \left(1 - \frac{H(u_r)}{\zeta}\right). \tag{11b}$$

Here $\zeta$ represents a penalty parameter that varies depending on the magnitude of $H$. Please refer to Zou et al. (2022) for the specific parameter values and more details on the overall repair and penalty treatments.

The well rate constraint is also treated in the repair procedure. In this case we minimize the quantity $C_{t,wr}$ (Eq. 9). Unlike the nonlinear geometric constraints, this is a linear equality constraint that is satisfied by SQP in a single projection. Hence, an additional penalty function is not needed for this constraint. The nonlinear output constraint $C_{aq}$ (Eq. 10) is treated after objective function evaluation using the filter method described by Isebor et al. (2014). In the filter method, the constraint violation $C_{aq}$ is regarded as an objective

to minimize along with the objective function $f_1$ or $f_2$. If $C_{aq} > 10^{-5}$, minimizing $C_{aq}$ takes precedence over minimizing $f_1$ or $f_2$.

***Core Optimization Algorithms.*** We now describe the two derivative-free core optimization algorithms, particle swarm optimization (PSO) and differential evolution (DE), applied in this study. In PSO, introduced by Kennedy and Eberhart (1995), each solution vector $\mathrm{u} \in \mathbb{R}^n$ ($n = 6N_w + N_w \times N_t$) is treated as a particle (individual) in a swarm (population) containing $N$ solutions. At each PSO iteration $k$, the $N$ solutions ($\mathrm{u}_i^k$, $i = 1, \ldots, N$) are updated through application of

$$\mathrm{u}_i^{k+1} = \mathrm{u}_i^k + \mathrm{v}_i^{k+1}, \tag{12a}$$

$$\mathrm{v}_i^{k+1} = \omega \mathrm{v}_i^k + c_1 \mathrm{D}_1 (\mathrm{u}_i^{Pbest} - \mathrm{u}_i^k) + c_2 \mathrm{D}_2 \left(\mathrm{u}_i^{Gbest} - \mathrm{u}_i^k\right). \tag{12b}$$

Here $\mathrm{v}_i^{k+1} \in \boldsymbol{R}^n$ is a velocity vector composed of so-called inertial, cognitive and social velocity components. The inertial term, $\omega \mathrm{v}_i^k$, involves the velocity at the previous iteration; this term acts to maintain some continuity in the search. The cognitive term, $c_1 \mathrm{D}_1 \left(\mathrm{u}_i^{Pbest} - \mathrm{u}_i^k\right)$, moves the particle toward the best location it has found thus far in the search $\left(\mathrm{u}_i^{Pbest}\right)$. The social term, $c_2 \mathrm{D}_2 \left(\mathrm{u}_i^{Pbest} - \mathrm{u}_i^k\right)$, attracts the solution to the location of the best particle in the overall swarm or in its more limited 'neighborhood' $\left(\mathrm{u}_i^{Gbest}\right)$. Stochasticity is introduced through $\mathrm{D}_1$ and $\mathrm{D}_2$, which are diagonal matrices containing uniformly distributed random values sampled from [0, 1] at each iteration.

The hyperparameters $\omega$, $c_1$, and $c_2$ can be tuned to enhance PSO performance. Zou et al. (2022) conducted a detailed assessment to determine the best-performing set of hyperparameters. Consistent with their results, here we specify $\omega = 0.721$, $c_1 = 1.193$, and $c_2 = 1.193$. In addition, a global topology, where each particle is connected to all particles at each iteration, is used.

DE, introduced by Storn and Price (1997), is also a population-based derivative-free optimization method that shares some similarities with genetic algorithms. Each solution $\mathrm{u}_i^k$, $i = 1, \ldots, N$, is updated to $\mathrm{u}_i^{k+1}$ by applying so-called mutation, crossover, and selection operations. During mutation, a 'mutant' $\mathrm{x}_i^k$ is generated for each solution $\mathrm{u}_i^k$ through application of

$$\mathrm{x}_l^k = \mathrm{u}_{best}^k + w_m \left(\mathrm{u}_{i,r_1}^k - \mathrm{u}_{i,r_2}^k\right), \tag{13}$$

where $\mathrm{u}_{best}^k$ is the global best solution found thus far in the optimization, $\mathrm{w_m} \in [0, 2]$ is the specified mutation factor, and $\mathrm{u}_{i,r_1}^k$ and $\mathrm{u}_{i,r_2}^k$ are two different solutions randomly selected from the currently population. A trial solution $\hat{\mathrm{u}}_i^k$ is then generated by accepting decision variables from either the solution $\mathrm{u}_i^k$ or the mutant $\mathrm{x}_i^k$ via crossover. In binomial crossover (used here), a random value $r_j \in (0, 1)$ is generated and compared with a constant crossover factor $c_f \in (0, 1)$ at the location of each decision variable. If $r_j \le c_f$, $\hat{\mathrm{u}}_i^k$ inherits the associated decision variable from $\mathrm{x}_i^k$, and it inherits from $\mathrm{u}_i^k$ if $r_j > c_f$. Finally, the objective function value for the trial solution $\hat{\mathrm{u}}_i^k$ is evaluated and compared with the original solution $\mathrm{u}_i^k$. The updated solution $\mathrm{u}_i^{k+1}$ is taken to be the solution with the smaller objective function value (for a minimization problem).

DE also contains hyperparameters that impact algorithm performance. Here we use the best-performing DE hyperparameters identified by Zou et al. (2022). The specific settings are $w_m = 0.5$ and $c_f = 0.5$. In addition, consistent with the earlier study, binomial crossover (rather than exponential crossover) is applied.

***Multifidelity Optimization.*** Detailed (fine-scale) flow effects can impact the objective function values computed in the optimizations. The use of highly resolved models in all optimization iterations would,

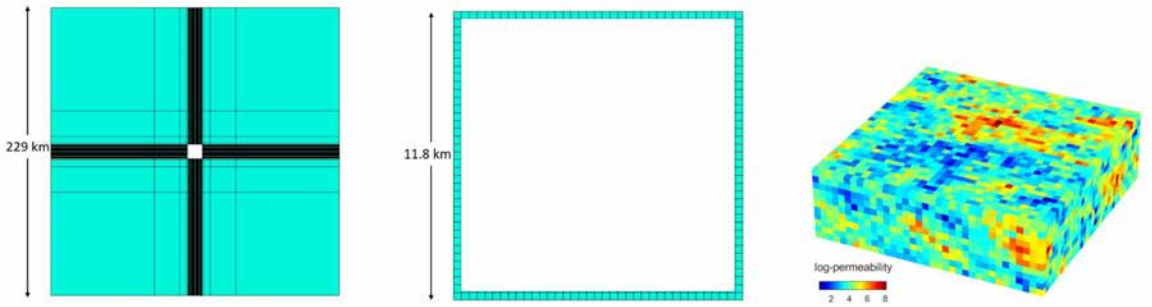
however, lead to excessive computational costs. In this work we therefore introduce a multifidelity optimization treatment, which we now describe.

Our multifidelity approach involves three levels of grid resolution. Details of the gridding at each level are provided in the next section. The coarsest resolution (referred to as level 1) is used at the start of the optimization run. Because the optimization algorithms applied here are stochastic, three separate runs are performed, using different randomly generated swarms/populations. The level 1 optimizations are terminated based on a percent improvement criterion, where we 'look back' a specific number of iterations (20 in this study) and terminate the run if the percent improvement in the objective function is below 1%. Next, optimization is performed using the level 2 grid, which is a 2 × 2 × 2 refinement of the level 1 grid. Only one level 2 run is performed. The initial swarm/population is comprised of the best solutions, from all three runs, found at the end of the level 1 optimizations (thus the swarm is diversified at this stage). The same termination criterion is again applied (less than 1% improvement over 20 iterations). The level 3 optimization involves runs performed on a grid constructed through a 3 × 3 × 3 refinement of the level 1 grid. The initial swarm/population at level 3 is simply the swarm/population at the end of the level 2 optimization. The level 3 optimization is terminated when a 1% improvement is not achieved over 10 iterations.

In the multifidelity treatment applied in this work, the permeability field is defined on the level 1 scale. Finer-scale heterogeneity is not introduced at levels 2 and 3. This means that permeability is uniform over 2 × 2 × 2 'patches' at level 2, and over 3 × 3 × 3 patches at level 3. It would be straightforward to define the permeability field at the finest (level 3) scale and then upscale it to levels 1 and 2, as was done by Aliyev and Durlofsky (2017), though this approach is not considered here. In future work, it will be of interest to use upscaled multiphase flow functions (Mishra et al., 2020; Rabinovich et al., 2015) in this multifidelity procedure.

## Aquifer Model and Base-case Setup

The aquifer model used in this study is a variant of that considered in previous investigations (Cameron and Durlofsky, 2012; Sun and Durlofsky, 2019). The full model, shown in Fig. 3(a), is 229 km × 229 km × 242 m, and the grid dimensions are 43 × 43 × 11. The central region, shown in Fig. 3(c) and referred to as the storage aquifer, is of dimensions 11.2 km×11.2 km×242 m. This portion of the model, at level 1, is represented on a 35 × 35 × 11 uniform grid. The larger outer area surrounding the storage aquifer, which is modeled with a nonuniform grid, enables pressure dissipation. This region is homogeneous, with a permeability of 60.3 mD and a porosity of 0.2.




(a) 43 × 43 × 11 overall model, with the 35 × 35 × 11 storage aquifer shown as the white square in the center

(b) 37 × 37 × 11 overall model with the 35 × 35 × 11 storage aquifer shown as the white square in the center. Pore-volume multiplier is applied to the outer region (in blue)

(c) Log-permeability of the storage aquifer (white square in (a) and (b))

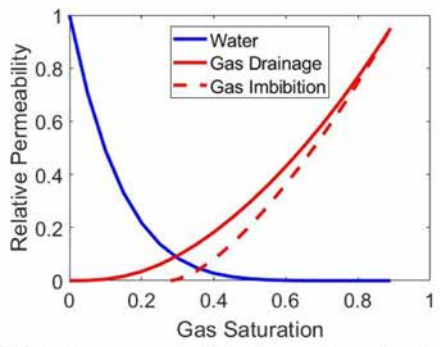
**Figure 3—Overall reservoir models and log-permeability of the storage aquifer (adapted from Sun and Durlofsky (2019)).**

In this study, we replace the outer region in Fig. 3(a) with a single layer of grid blocks of the same size as the aquifer blocks. A pore-volume multiplier is applied such that these blocks have the same total pore

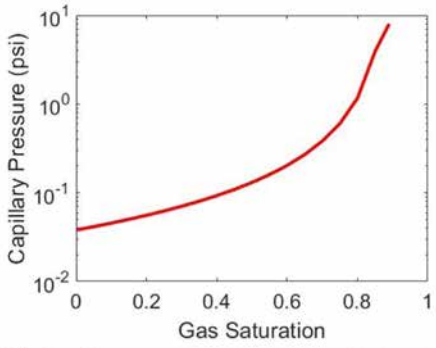
volume as the full outer region in the original model. The resulting model, which represents the level 1 model used in this study, is shown in Fig. 3(b). A comparison of simulation results using the models in Fig. 3(a) and (b) will be presented later.

The storage aquifer shown in Fig. 3(c) is heterogeneous. The mean and standard deviation of log-permeability are 4.03 and 0.99, respectively. Permeability is a diagonal tensor with $k_y = k_x$ and $k_z = 0.1k_x$ ($k_x$, $k_y$ and $k_z$ are permeability components in the coordinate directions). Porosity is correlated with permeability and is, on average, 0.2. The depth of the reservoir is 1524 m, which results in a pressure of 155 bar at the top of the model. The reservoir temperature is 55.2 °C. These conditions ensure that $CO_2$ is supercritical with a density of approximately 666 kg/m$^3$. The model is initially saturated with brine (10,000 ppm NaCl).

In the simulations, we include the effects of relative permeability hysteresis (using the model in Cameron and Durlofsky (2012)), as shown in Fig. 4(a). Heterogeneous capillary pressure curves, which depend on porosity and permeability, are also used (the specific model is that applied in Sun and Durlofsky (2019)). The capillary pressure curve constructed with the Brooks-Corey model for a porosity of 0.2 and permeability of 29 mD is shown in Fig. 4(b). The mutual solubility of $CO_2$ and the aqueous phase, as well as salt precipitation, are included. The model does not include capillary pressure hysteresis, molecular diffusion, or mineral trapping.




(a) Relative permeability curves for $CO_2$–brine system (both drainage and imbibition)

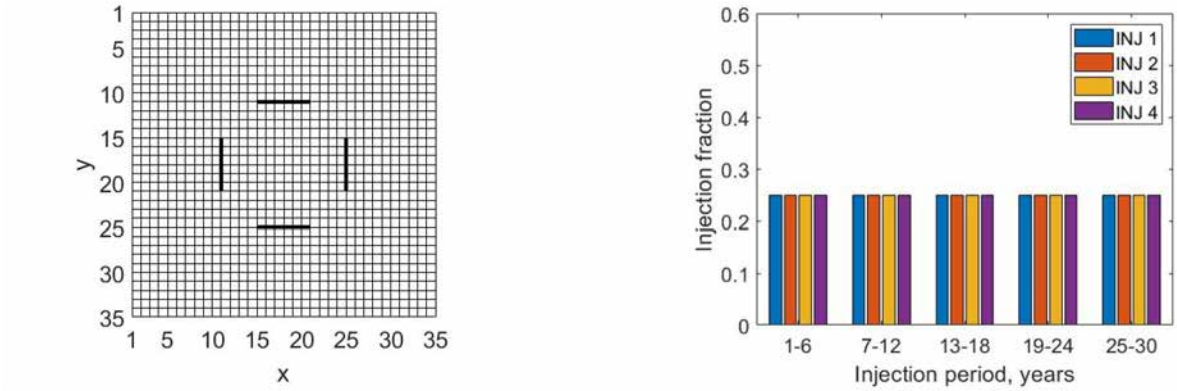



(b) Capillary pressure at porosity of 0.2 and permeability of 29 mD

**Figure 4—$CO_2$–brine two-phase flow functions.**

The total field injection target is 4 megatonnes $CO_2$ per year for 30 years. The simulations continue for an additional 170 years, so the total simulation time frame is 200 years. All simulations are conducted using the CO2STORE option in Eclipse (Schlumberger, 2014). The optimizations are performed within the Stanford Smart Fields Unified Optimization Framework.

It is useful to define a base-case scenario for the model described above. This setup, shown in Fig. 5(a), involves four horizontal wells arranged in a square pattern in layer 9 of the model (layer 11 is the bottom layer). To meet the yearly injection target of 4 megatonnes $CO_2$, each well is specified to inject at a constant surface rate of $1.467 \times 10^6$ m$^3$/day for 30 years (this is based on a $CO_2$ density of 1.868 kg/m$^3$ at surface conditions). In the optimization runs, we will consider variable rates in each of the injectors over five control periods, with each period lasting six years. The injection rates are expressed in terms of the injection fraction for each well in each period, as shown in Fig. 5(b). Finally, as discussed earlier, BHPs are limited to 150% of the initial aquifer pressure (which is 155 bar at 1524 m). We thus impose a maximum injection BHP of 233 bar for each well.



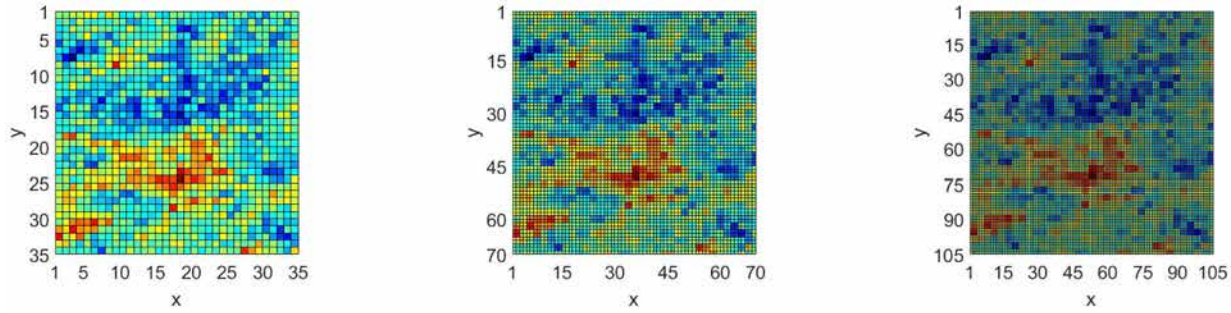

(a) Base-case well configuration (layer 9). Thick black lines indicate the four horizontal wells

(b) Base-case well injection rate fractions

**Figure 5—Well locations and settings for the base-case scenario.**

## Grid Refinement and Sensitivity Analysis

The model shown in the previous section is too coarse to resolve detailed flow effects. As discussed earlier, we apply a multifidelity treatment to (partially) address this issue. The storage aquifer in the level 1 grid contains 35×35×11 cells. The level 2 grid entails a 2×2×2 refinement (70×70×22 storage aquifer cells), and the level 3 grid a 3 × 3 × 3 refinement (105 × 105 × 33 storage aquifer cells). The top layers of the storage aquifer models for these three grid levels are shown in Fig. 6. Prior to simulation, a conforming outer region, as depicted in Fig. 3(b), is added to the model. It is evident in Fig. 6 that permeability refinement is not introduced as we proceed through the levels of grid refinement, as discussed earlier.




(a) Level 1 (35 × 35 × 11)

(b) Level 2 (70 × 70 × 22)

(c) Level 3 (105 × 105 × 33)

**Figure 6—Grid resolution levels considered for the storage aquifer in multifidelity optimization. Outer region not shown. Permeability field is defined on the level 1 grid.**

We now assess results for both objective functions on the three grid levels. The well positions on the level 2 and level 3 grids are adjusted to correspond to those on the level 1 grid. Simulations with the base-case setup are performed both for 'full' models with nonuniform gridding in the outer region (as shown in Fig. 3(a)) and for models with a single ring of cells and pore-volume multipliers (as shown in Fig. 3(b)). We confirmed that the specified (target) mass of $CO_2$ was injected over the 30-year injection period in all cases. Results for mobile $CO_2$ fraction as a function of time are presented in Fig. 7. Our first observation is that the pore-volume multiplier results (dashed curves with circles, labeled with 'PVMULT') agree closely with the corresponding cases simulated with nonuniform outer-boundary grids (solid curves, labeled with 'nonuniform'). This agreement is observed on all three grid levels. Furthermore, the pressure increases by only a very small amount, about 0.6 bar, in the single ring of boundary cells (with pore-volume multipliers) over the simulation time frame. These observations indicate that we can use a single ring of cells and pore-volume multipliers rather than nonuniform grids that extend out a great distance. This is a useful finding, as the pore-volume multiplier approach leads to fewer overall cells and thus faster simulation runs.

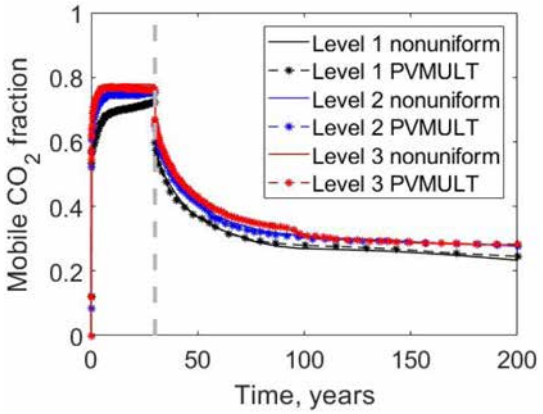



**Figure 7—Impact of grid resolution and outer-region treatment on results for mobile $CO_2$ fraction (base-case setup). Gray dashed line indicates the end of the injection period.**

The results in Fig. 7 do display some variation with grid level, as would be expected. The variation between levels 1 and 2 is larger than that between levels 2 and 3, suggesting that convergence under grid refinement is occurring (though we do not claim the converged solution has been achieved). The end-time mobile $CO_2$ fraction for the level 1 (PVMULT) run is 0.245, while for level 2 it is 0.278 and for level 3 it is 0.283. Thus we see that, although there is some error, the level 1 results are still reasonable and thus appropriate for use in early stages of the optimization.

The impact of grid level on the storage efficiency objective is now considered. These simulations are performed using the pore-volume multiplier approach. Results are presented in Fig. 8, where we show $CO_2$ saturation for the three levels in the top layer of the model (first row) and full 3D $CO_2$ plumes (second row). Although the total $CO_2$ injected is the same at all levels, substantial differences in plume extent and $CO_2$ saturation are evident between the level 1 and level 2 results. Differences also exist between the level 2 and level 3 saturation fields, but they are less than those between the level 1 and level 2 results. This is consistent with our observations for the first objective. In terms of objective function values (computed through application of Eq. 3), storage efficiency is 0.044 at level 1, 0.037 at level 2, and 0.036 at level 3.

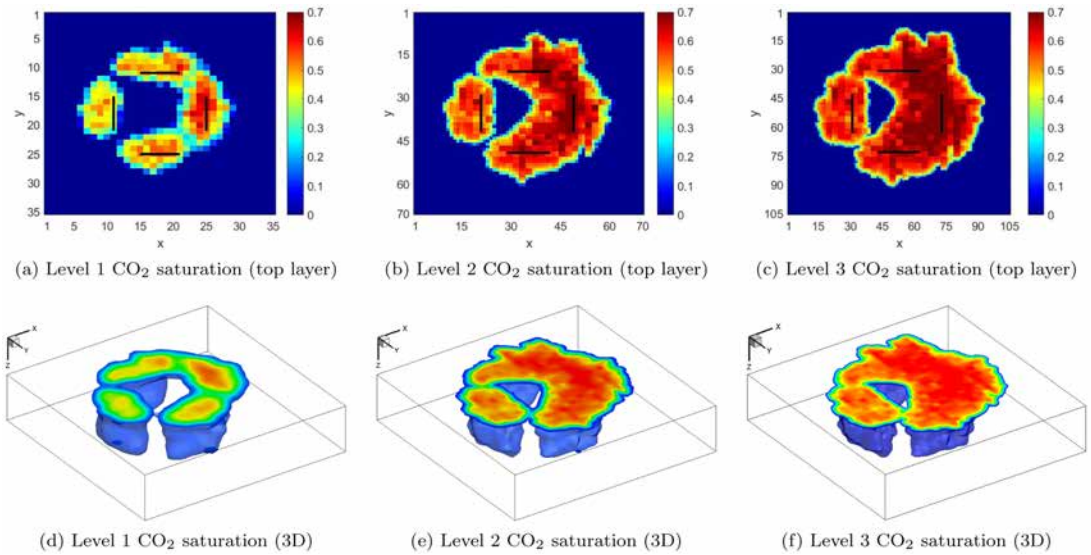



**Figure 8—Impact of grid resolution on $CO_2$ saturation fields (base-case setup). Results shown for top layer of the model (first row) and for full 3D field (second row).**

## Optimization Results

In this section, we present results for the minimization of mobile $CO_2$ fraction and for the maximization of storage efficiency. Both quantities are optimized at the end of simulation time frame (200 years). PSO and DE optimizations, performed within the integrated workflow described in the previous section, are considered. Four deviated/horizontal $CO_2$ injection wells are specified in all cases. The well schedules are divided into five control periods over the 30-year injection period. This results in 11 decision variable per well, and a total of 44 decision variables for the full problem. Following Zou et al. (2022), we use population/swarm size $N = 44$ for both PSO and DE in this study. Geometric constraints on the well configuration, which are applied in all cases, are provided in Table 1. Because of the stochastic nature of the algorithms, PSO and DE are typically run multiple times. Here we perform three optimization runs, though for the multifidelity case this entails three runs only at the coarsest (level 1) grid resolution.

**Table 1—Well geometric constraints applied in all cases**

| | |
|---|---|
| Maximum well length $L_{wl}^{max}$ | 1600 m |
| Minimum well length $L_{wl}^{min}$ | 640 m |
| Minimum interwell distance $L_{iw}^{min}$ | 960 m |
| Minimum well-to-boundary distance $L_{wp}^{min}$ | 960 m |

In order to appropriately quantify computational demands in the multifidelity optimization runs, we apply the concept of effective number of function evaluations. With this approach, which is based on a computational complexity perspective, a single function evaluation corresponds to a fine-scale (level 3) simulation. One effective function evaluation at level 1 entails 27 level 1 simulations (function evaluations), since the level 1 grid contains a factor of 27 fewer grid blocks. Similarly, one effective function evaluation at level 2 corresponds to eight level 2 simulations. Note that this treatment assumes linear scaling in simulation time with the number of grid blocks in the model. Because the scalings observed for the cases considered in this study are slightly super-linear, this approach underestimates the actual reduction in computation resulting from the use of multifidelity optimization. The computational complexity approach is nonetheless useful, however, because it avoids simulator and cluster-specific complications related to detailed simulator settings, simulator overhead, i/o, memory, other demands on the computational cluster, etc. In terms of actual timings, a single (typical) level 3 simulation run requires about 111 minutes on CentOS 7 compute nodes (2 × E5-2660 Intel processor). With the computational complexity treatment, this corresponds to timings of 4 minutes for a level 1 run and 33 minutes for a level 2 run.

For the first objective function, we present comparisons between multifidelity optimization and optimizations performed using only the fine-scale model (level 3 grid). The latter optimizations are run three times with different (randomly generated) initial swarms/populations. In the case of multifidelity optimization, three runs are performed at level 1, but only a single run is performed at levels 2 and 3, as described earlier. The computational demands for the multifidelity optimizations are compared to those for fine-scale optimization on the basis of the effective number of function evaluations.

### Case 1: Minimization of Mobile $CO_2$ Fraction

PSO results are shown in Fig. 9 for level 3 optimizations (all-fine-scale, denoted 'AF') and for multifidelity optimization ('MF'). The evolution of the penalized objective function ($Y_1$ in Eq. 11(a)) with the effective number of function evaluations, for the best solution in the swarm, is presented in Fig. 9(a). The AF optimization runs are terminated after performing a number of function evaluations comparable to the number of effective function evaluations used in the MF run (approximately 1760 in this case). In Fig.

9(a), we observe jumps in the MF results when switching from one grid level to the next. These jumps correspond to the errors associated with the coarser-grid simulations. It is evident, however, that most of the improvement in the objective function value achieved at level 1 is applicable at level 2 (and similarly for levels 2 and 3). We also see that, for the same computational effort, the MF treatment provides a lower optimized mobile $CO_2$ fraction (0.096) compared to the AF results, which range from 0.110 to 0.118. This demonstrates the impact of the multifidelity treatment for this problem.

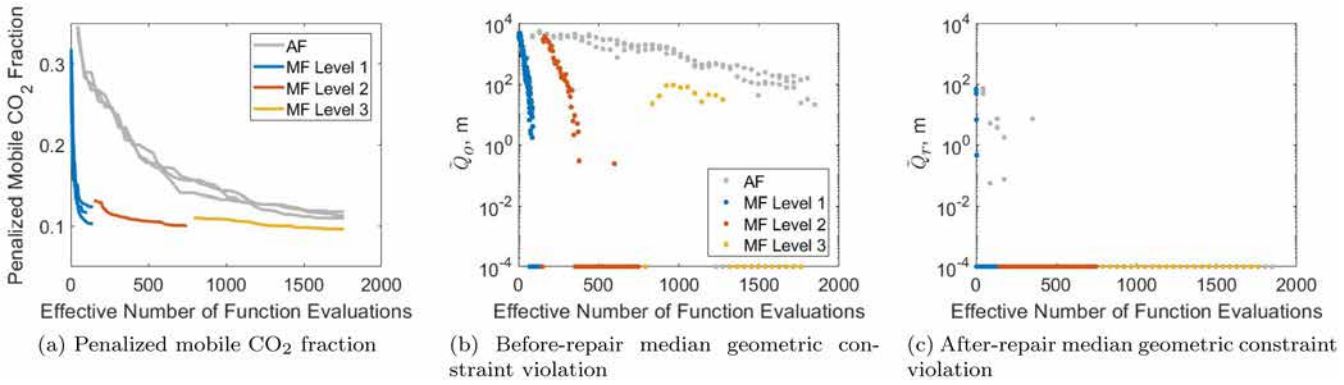



**Figure 9—Evolution of PSO for the minimization of mobile $CO_2$ fraction. Results are shown for optimizations performed with all-fine-scale models (AF) and with multifidelity treatment (MF).**

Fig. 9(b) and (c) show the median geometric constraint violations over the swarm, at each PSO iteration, before ($Q_o$, where 'o' denotes original) and after ($Q_r$, where 'r' denotes repaired) the repair procedure has been applied. In these plots, a value of $10^{-4}$ indicates feasibility. Reduction in $Q_o$ is observed in Fig. 9(b) as the PSO runs proceed for both the AF and MF optimizations. Median before-repair feasibility is, however, only attained in the MF case. After-repair feasibility is achieved for both the AF and MF cases, as is apparent in Fig. 9(c), which illustrates the effectiveness of the repair procedure. Note that all runs (AF and MF) achieve feasibility by the end of the optimization.

Analogous optimization results using DE are presented in Fig. 10. In Fig. 10(a), we again observe substantial benefit from using the multifidelity treatment. The MF solution provides an optimized mobile $CO_2$ fraction of 0.090 after 1662 effective function evaluations, whereas the best AF run provides an optimized mobile $CO_2$ fraction of 0.124 after a comparable number (1672) of effective function evaluations. In addition, consistent with the PSO results, we observe faster reduction in before-repair constraint violations in the MF case, though the repair procedure again provides feasible median solutions in all cases (Fig. 10(b) and (c)).

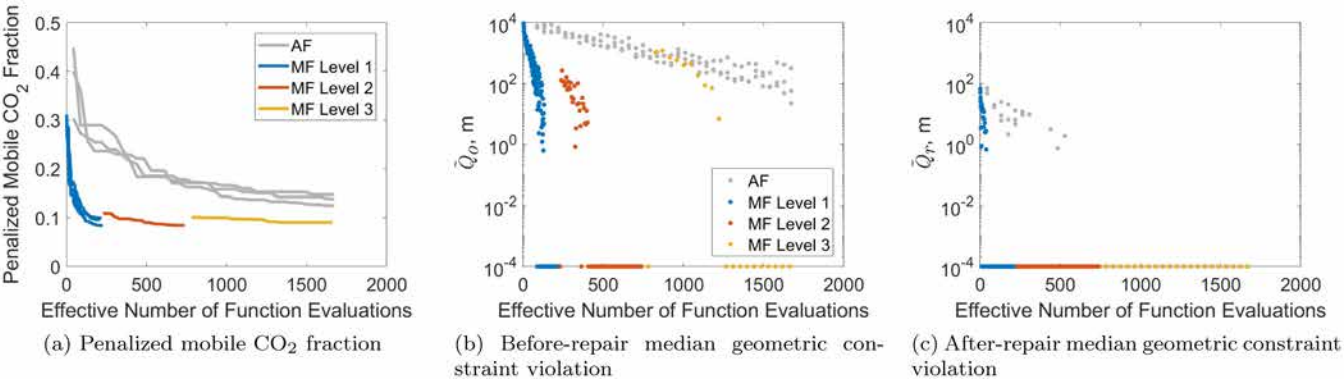



**Figure 10—Evolution of DE for the minimization of mobile $CO_2$ fraction. Results are shown for optimizations performed with all-fine-scale models (AF) and with multifidelity treatment (MF).**

Results from all runs, for both PSO and DE, are summarized in Table 2. The final optimized mobile $CO_2$ fractions are shown in the last column. From these values, it is evident that the multifidelity DE case provides the lowest mobile $CO_2$ fraction (0.090) at 200 years. We also see that the multifidelity PSO run outperforms the AF runs.

**Table 2—All-fine-scale (AF) and multifidelity (MF) optimization results for minimizing mobile $CO_2$ fraction using PSO and DE**

| | After 792 effective FEs | Final result |
|---|---|---|
| PSO AF Run 1 | 0.141 | 0.113 |
| PSO AF Run 2 | 0.156 | 0.118 |
| PSO AF Run 3 | 0.156 | 0.110 |
| PSO MF | 0.110 | 0.096 |
| DE AF Run 1 | 0.172 | 0.147 |
| DE AF Run 2 | 0.170 | 0.138 |
| DE AF Run 3 | 0.162 | 0.124 |
| DE MF | 0.101 | 0.090 |

The multifidelity runs, for both PSO and DE, switch from level 2 to level 3 slightly before 792 effective function evaluations (FEs). This corresponds to 18 iterations for the all-fine-scale runs. In the second column of Table 2, we provide optimization results at this stage for all runs (these values are all computed at level 3). The multifidelity PSO and DE runs significantly outperform the all-fine-scale runs. In fact, the multifidelity results at this stage provide comparable or better objective function values than are attained by the all-fine-scale optimizations at the end of the runs. The main reason for the superior performance of the multifidelity approach is that it enables many more PSO or DE iterations to be performed for a given computational budget, thus allowing for more exploration of the search space than is achieved with all-fine-scale optimization runs.

In terms of timings, based on the 111-minute run-time for a single level 3 simulation, the total serial computation required to complete one DE all-fine-scale optimization run (1672 function evaluations) is about 3093 hours. Thus, about 9280 hours are required for the three all-fine-scale runs. If each run is fully parallelized (which means we have 44 cores and 44 simulator licenses available), the elapsed time for one run would be about 70 hours (210 hours for three runs). The multifidelity treatment, as implemented here, has the advantage of requiring three runs only at level 1. Thus the total serial time for the multifidelity optimization (based on 657 combined effective function evaluations for the three level 1 runs, 512 at level 2, and 924 at level 3) is 2035 hours. The fully parallelized elapsed time is about 118 hours. Thus we achieve speedup factors, relative to all-fine-scale optimization, of 4.6 (serial) and 1.8 (fully parallelized). These values understate actual speedup because they do not account for super-linear scaling.

Optimal well configurations and well settings, and the corresponding $CO_2$ saturation fields at 200 years, are presented in Fig. 11. Results are shown for the best solution at each level of the DE multifidelity run. In Fig. 11(a) – (c) the $CO_2$ saturation profiles are shown for the top layer of the model, with the well configuration projected onto this layer. The lengths of the wells cannot be determined from these images unless the well is horizontal, though all wells are at least 1500 m in length (the maximum allowable well length is 1600 m). The full 3D saturation fields are depicted in Fig. 11(d) – (f), and the well control schedules appear in Fig. 11(g) – (i). Two of the wells are (slightly) deviated and the other two wells are horizontal. All wells are placed near the bottom of the reservoir (layers 31-33). Comparing Fig. 11(a), (b) and (c), it is evident that the best well configuration changes relatively little from level to level. The well

controls, however, shift more significantly (Fig. 11(g), (h) and (i)), with the injection schedule becoming more variable as we proceed from level 1 to level 3.

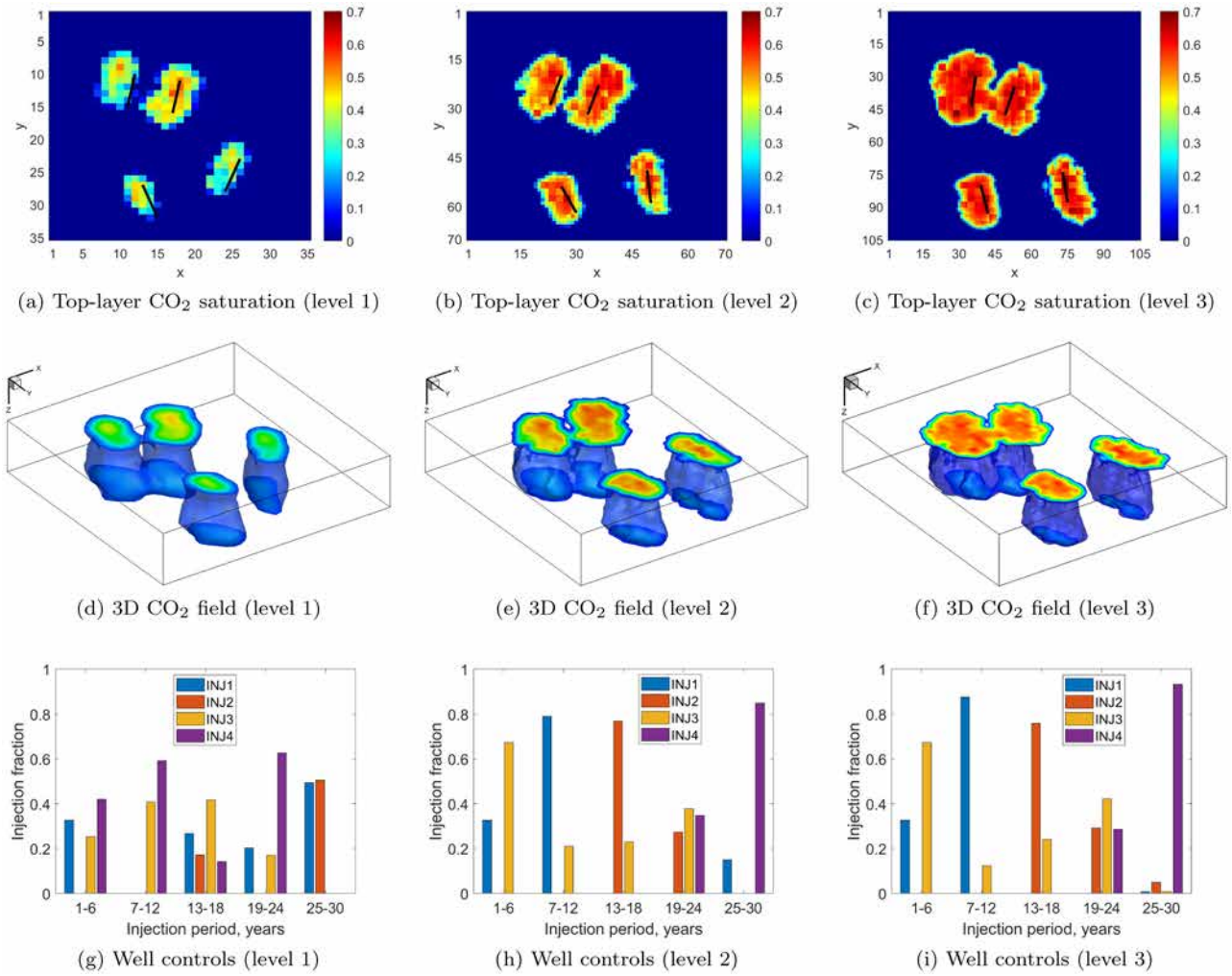



**Figure 11—Top-layer and 3D $CO_2$ fields at 200 years, and well controls for the best solution at each level using multifidelity DE to minimize mobile $CO_2$ fraction.**

The optimal level 3 solution (corresponding to the third column of Fig. 11) entails wells that are separated from one another and subject to highly variable injection rates. In fact, from Fig. 11(i), we see that only two of the wells (INJ1 and INJ3) are active through the first two injection periods, and INJ4 does not start injecting until the fourth period. Thus, all wells need not be drilled at the start of the storage operation. The well configuration in Fig. 11(c) and the injection schedule in Fig. 11(i) provide distinct plumes that promote dissolution and residual trapping.

In Fig. 12, we compare (multifidelity DE) optimization and base-case results for mobile, dissolved, and residual $CO_2$ fractions as a function of time (these results are at level 3). The end of the injection period is indicated by the gray dashed line. Significant reduction in the mobile $CO_2$ fraction is evident in the optimized results over nearly all of the simulation time frame. At 200 years, the mobile $CO_2$ fraction is 0.090 for the optimized case versus 0.283 for the base-case.

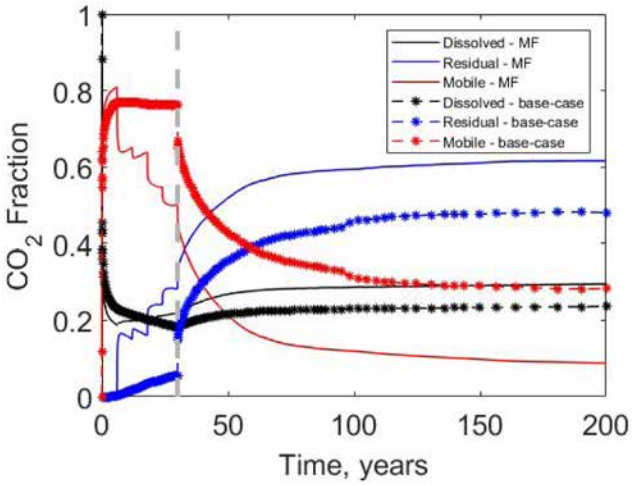



**Figure 12—Comparison of mobile, dissolved, and residual $CO_2$ fractions for multifidelity run and the base-case setup. Gray dashed line indicates the end of the injection period. Objective is minimization of mobile $CO_2$ fraction.**

Next, we compare the results between multifidelity and all-fine-scale DE optimizations. These comparisons appear in Fig. 13; the AF results are for the best DE AF run (run 3). We observe clear differences between the two solutions in the $CO_2$ fractions during the 30-year injection period. This is due to the differences in the optimal well configurations and controls found in the two optimization runs. The differences are relatively small at later times, though it is evident that the multifidelity run provides a lower final value for mobile $CO_2$ fraction than the best all-fine-scale run (0.090 versus 0.124).

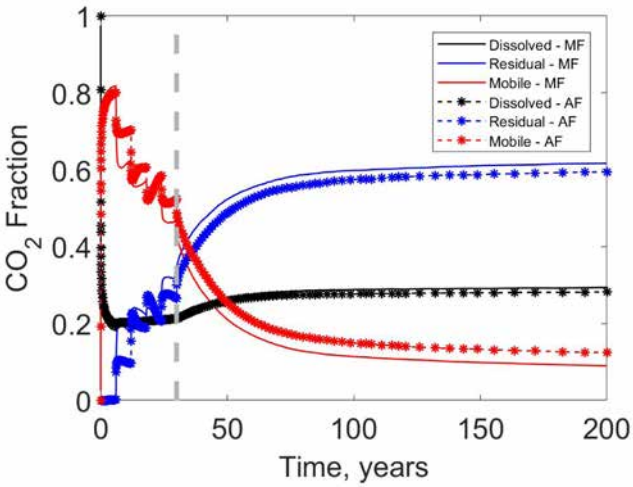



**Figure 13—Comparison of mobile, dissolved, and residual $CO_2$ fractions for multifidelity run and (best) all-fine-scale DE run. Gray dashed line indicates the end of the injection period. Objective is minimization of mobile $CO_2$ fraction.**

Finally, in Fig. 14, we show the injection-well BHPs during the injection period. These results are again for the multifidelity DE solution. The red dashed line indicates the maximum allowable BHP (233 bar) for each injector, and the black dashed line depicts the initial pressure at the top of the reservoir (155 bar). The injector BHPs shown here reflect the injection rate fractions shown in Fig. 11(i). For example, the peaks in INJ1 and INJ3 BHPs appear early in the run, while those for INJ2 and INJ4 occur later. Importantly, the BHP constraints are satisfied by all wells at all times. At a few time steps, INJ2 and INJ4 switch from surface rate control to BHP control, which acts to limit total $CO_2$ injection. The filter treatment, however, reduces this amount to less than the feasibility criterion of $10^{-5}$ for $C_{aq}$ (Eq. 10).

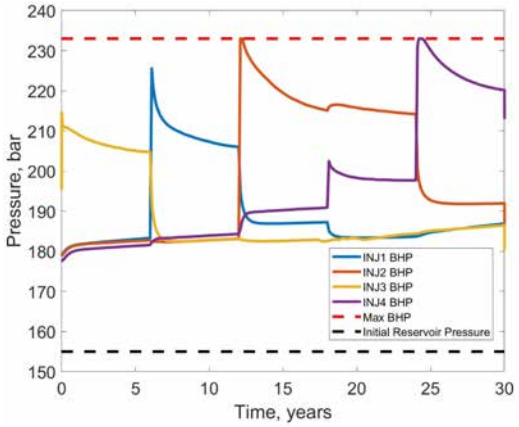



**Figure 14—Injection-well BHPs for multifidelity DE solution minimizing mobile $CO_2$ fraction.**

## Case 2: Maximization of Storage Efficiency

We now present PSO and DE results for the second objective function – maximization of $CO_2$ storage efficiency. Because, for the previous case, we achieved better results with less overall computational effort using the multifidelity treatment, we do not perform any all-fine-scale (AF) optimizations for this case. Figs. 15 and 16 display the multifidelity optimization results for PSO and DE, respectively. Comparing Figs. 15(a) and 16(a) with Figs. 9(a) and 10(a), we observe larger jumps between grid levels for this objective function, consistent with our observations for the base-case. PSO and DE provide comparable final results, with PSO giving an optimized storage efficiency of 0.074, and DE giving 0.073. These results represent a factor-of-two improvement relative to the base-case value of 0.036.

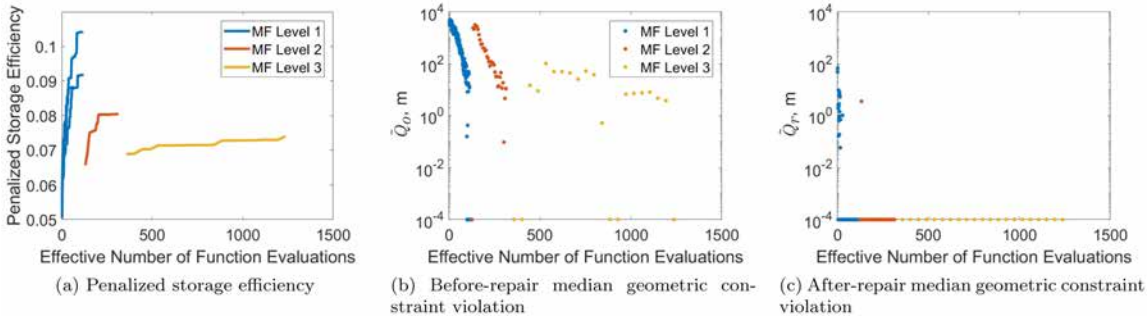



**Figure 15—Evolution of multifidelity PSO for the maximization of storage efficiency.**

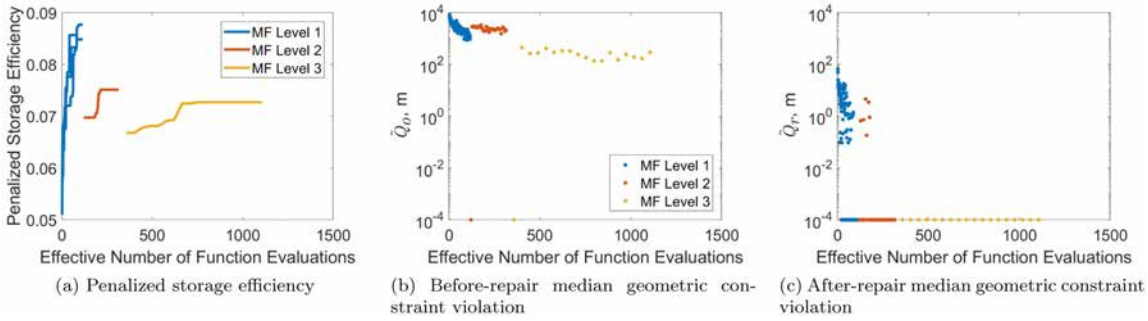



**Figure 16—Evolution of multifidelity DE for the maximization of storage efficiency.**

Geometric constraints for this objective function are more difficult to satisfy because storage efficiency is improved by placing wells close together, which can trigger constraint violations. The median before-repair geometric constraint violations are shown in Figs. 15(b) and 16(b). As expected, these constraint

violations are larger, particularly for DE, than in the previous case (compare Figs. 16(b) and 10(b)). The repair procedure is again effective in providing geometric feasibility, as is evident in Figs. 15(c) and 16(c).

Results for the best PSO solutions at the three grid levels are presented in Fig. 17. Fig. 17(a) – (c) show the 2D $CO_2$ saturation profiles and well configurations projected onto the top layer, (d) – (f) display the plumes in 3D, and (g) – (i) show the well controls. As was the case with the first objective function, the optimized solutions entail both horizontal and deviated wells placed near the bottom of the formation. Compared to the previous optimized configuration (Fig. 11(c)), the well configurations in this case (Fig. 17(c)) display much tighter well spacings. This is of course consistent with the goal of maximizing storage efficiency. In addition, although still quite variable, the well control settings for this case (Fig. 17(i)) correspond to less extreme injection fraction values than in the first case (Fig. 11(i)). BHPs for the four injectors, for the PSO solution, are presented in Fig. 18. It is apparent that the BHP constraints are satisfied at all times. In addition, the overall field injection target is met.

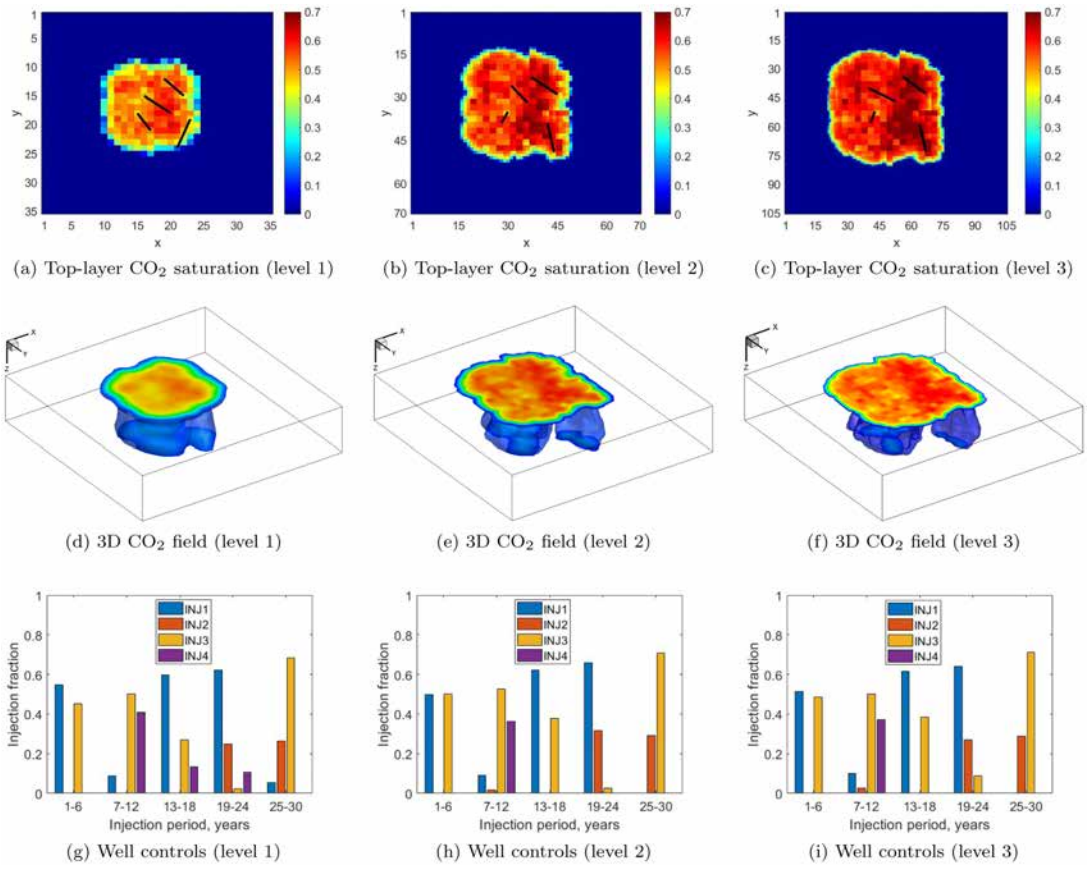



**Figure 17—Top-layer and 3D $CO_2$ fields at 200 years, and well controls for the best solution at each level using multifidelity PSO to maximize storage efficiency.**

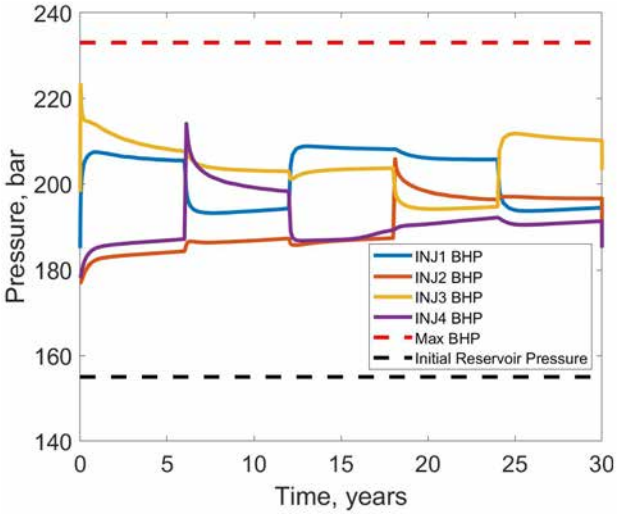



**Figure 18—Injection-well BHPs for multifidelity PSO solution maximizing storage efficiency.**

## Concluding Remarks

In this work, we developed a general framework for optimizing the locations and time-varying rates of multiple horizontal/deviated $CO_2$ injection wells. Two objective functions were considered – the minimization of mobile $CO_2$ fraction and the maximization of $CO_2$ storage efficiency – along with multiple realistic constraints. The latter include geometric constraints on the well configuration, injection-rate constraints, and plume-location constraints. BHPs were also limited to a specified maximum. The constraints were handled with a variety of treatments, which included a preprocessing repair procedure, a filter method, and a penalty method. Two derivative-free core optimization algorithms, PSO and DE, were applied. A multifidelity approach involving three levels of grid resolution was introduced.

PSO and DE multifidelity optimization results were compared with all-fine-scale optimizations for the first objective function (minimization of mobile $CO_2$ fraction). The multifidelity approach was shown to be the method of choice with both optimizers, as it provided better optima, with less overall computation, than the use of all-fine-scale simulations. Among all the optimization runs, multifidelity DE resulted in the lowest mobile $CO_2$ fraction (0.090), which represents a 68% improvement over the base-case result (0.283). With all-fine-scale simulations, DE provided a mobile $CO_2$ fraction of 0.124, which exceeds the multifidelity DE result by 38%. The optimal configuration involved well-separated horizontal and deviated wells placed towards the bottom of the storage aquifer. This design, along with a highly variable injection-rate schedule, led to separated plumes and enhanced dissolution and residual trapping.

For the maximization of storage efficiency, results were generated using only the multifidelity treatment. The optimized objective function values from PSO and DE were very similar (0.074 and 0.073, respectively). These storage efficiencies are about twice the base-case value. For this objective function the geometric constraints are more challenging to satisfy because storage efficiency increases when wells are closely spaced. The repair procedure was nonetheless able to provide feasible well configurations, and all other constraints were also satisfied. The optimized configuration and well settings in this case led to a single ‘merged’ plume that extended over much less of the aquifer than the plumes for the first objective function.

There are many areas that should be considered in future work. In this study, only horizontal and deviated wells were optimized. Vertical wells, commonly considered in $CO_2$ operations, did not appear in our solutions because of the minimum-well-length constraint. By reducing this value, we could also consider vertical wells. The level 2 and level 3 models used in the multifidelity approach were generated by performing grid refinement on the level 1 model, without introducing finer-scale geological heterogeneity. It will be of interest to define the geomodel on a very fine scale and then populate the geomodels at

each level through application of proper upscaling procedures. The resolution at different levels, switching criteria, and initialization of the swarm/population at the next level in multifidelity optimization should also be further explored. The two objective functions considered in this study are somewhat conflicting in nature. Thus, multiobjective optimization could be utilized to identify a set of Pareto-optimal solutions, which would be useful for decision making. Drilling and operating costs could also be incorporated into the optimizations. Geomechanical effects are often important, and these could be included in the simulations. Finally, treatments for handling geological uncertainty will be required in practice and should be pursued. These could entail optimizing over multiple geomodels (i.e., performing robust optimization) and applying closed-loop procedures in which the geomodels are updated via history matching over the course of the storage operation.

## Acknowledgments

We are grateful to GeoCquest II, a BHP-sponsored collaborative project involving University of Melbourne (Australia), University of Cambridge (UK) and Stanford University, for funding. We thank the Stanford Smart Fields Consortium and the Stanford Doerr School of Sustainability for additional funding. We also thank Oleg Volkov for software support and useful discussions, and Su Jiang and Dylan Crain for providing simulation models. We acknowledge the Stanford Center for Computational Earth & Environmental Sciences (CEES) for computational resources.

## References

Al-Khdheeawi E.A., Vialle S., Barifcani A., et al. 2017. Influence of injection well configuration and rock wettability on $CO_2$ plume behaviour and $CO_2$ trapping capacity in heterogeneous reservoirs. *Journal of Natural Gas Science and Engineering* **43**: 190 – 206.

Aliyev E. and Durlofsky L.J. 2017. Multilevel field development optimization under uncertainty using a sequence of upscaled models. *Mathematical Geosciences* **49**: 307 – 339.

Allen R., Nilsen H.M., Andersen O., et al. 2017. On obtaining optimal well rates and placement for $CO_2$ storage. *Computational Geosciences* **21**: 1403 – 1422.

Bachu S. 2015. Review of $CO_2$ storage efficiency in deep saline aquifers. *International Journal of Greenhouse Gas Control* **40**: 188 – 202.

Cameron D.A. and Durlofsky L.J. 2012. Optimization of well placement, $CO_2$ injection rates, and brine cycling for geological carbon sequestration. *International Journal of Greenhouse Gas Control* **10**: 100 – 112.

Chen B. and Pawar R.J. 2019. Capacity assessment and co-optimization of $CO_2$ storage and enhanced oil recovery in residual oil zones. *Journal of Petroleum Science and Engineering* **182**: 106342.

Cihan A., Birkholzer J.T., and Bianchi M. 2015. Optimal well placement and brine extraction for pressure management during $CO_2$ sequestration. *International Journal of Greenhouse Gas Control* **42**: 175 – 187.

Goda T. and Sato K. 2013. Global optimization of injection well placement toward higher safety of $CO_2$ geological storage. *Energy Procedia* **37**: 4583 – 4590.

International Energy Agency. 2022. Legal and regulatory frameworks for CCUS. Technical report, International Energy Agency.

Isebor O.J., Durlofsky L.J., and Echeverría Ciaurri D. 2014. A derivative-free methodology with local and global search for the constrained joint optimization of well locations and controls. *Computational Geosciences* **18**: 463 – 482.

Jin Z.L. and Durlofsky L.J. 2018. Reduced-order modeling of $CO_2$ storage operations. *International Journal of Greenhouse Gas Control* **68**: 49 – 67.

Jun C., Kim M., and Shin H. 2019. Optimization of well placement and operating conditions for various well patterns in $CO_2$ sequestration in the Pohang Basin, Korea. *International Journal of Greenhouse Gas Control* **90**: 102810.

Kennedy J. and Eberhart R. 1995. Particle swarm optimization. Proceedings of ICNN'95 – International Conference on Neural Networks, 1942 – 1948.

Li B., Tchelepi H.A., and Benson S.M. 2013. Influence of capillary-pressure models on $CO_2$ solubility trapping. *Advances in Water Resources* **62**: 488 – 498.

Mishra A., Kurtev K.D., and Haese R.R. 2020. Composite rock types as part of a workflow for the integration of mm-to cm-scale lithological heterogeneity in static reservoir models. *Marine and Petroleum Geology* **114**: 104240.

Miyagi A., Akimoto Y., and Yamamoto H. 2018. Well placement optimization for carbon dioxide capture and storage via CMA-ES with mixed integer support. Proceedings of the Genetic and Evolutionary Computation Conference Companion, 1696 — 1703.

Park C., Oh J., Jo S., et al. 2021. Multi-objective optimization of $CO_2$ sequestration in heterogeneous saline aquifers under geological uncertainty. *Applied Sciences* **11**: 9759.

Rabinovich A., Itthisawatpan K., and Durlofsky L.J. 2015. Upscaling of $CO_2$ injection into brine with capillary heterogeneity effects. *Journal of Petroleum Science and Engineering* **134**: 60 – 75.

Schlumberger 2014. Eclipse Reference Manual.

Schneider P.J. and Eberly D.H. 2003. *Geometric Tools for Computer Graphics*. Elsevier Science.

Shamshiri H. and Jafarpour B. 2012. Controlled $CO_2$ injection into heterogeneous geologic formations for improved solubility and residual trapping. *Water Resources Research* **48**: W02530.

Shao Q., Boon M., Youssef A., et al. 2022. Modelling $CO_2$ plume spreading in highly heterogeneous rocks with anisotropic, rate-dependent saturation functions: A field-data based numeric simulation study of Otway. *International Journal of Greenhouse Gas Control* **119**: 103699.

Smit B., Reimer J.A., Oldenburg C.M., et al. 2014. *Introduction to Carbon Capture and Sequestration*. Imperial College Press.

Stopa J., Janiga D., Wojnarowski P., et al. 2016. Optimization of well placement and control to maximize $CO_2$ trapping during geologic sequestration. *Akademia Górniczo-Hutnicza Drilling, Oil, Gas* **33**: 93 — 104.

Storn R. and Price K. 1997. Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization* **11**: 341 – 359.

Sun W. and Durlofsky L.J. 2019. Data-space approaches for uncertainty quantification of $CO_2$ plume location in geological carbon storage. *Advances in Water Resources* **123**: 234 – 255.

Volkov O. and Bellout M. 2018. Gradient-based constrained well placement optimization. *Journal of Petroleum Science and Engineering* **171**: 1052 – 1066.

Wang Y., Zhang K., and Wu N. 2013. Numerical investigation of the storage efficiency factor for $CO_2$ geological sequestration in saline formations. *Energy Procedia* **37**: 5267 – 5274.

Wen G., Hay C., and Benson S.M. 2021. CCSNet: A deep learning modeling suite for $CO_2$ storage. *Advances in Water Resources* **155**: 104009.

Yamamoto H. and Doughty C. 2011. Investigation of gridding effects for numerical simulations of $CO_2$ geologic sequestration. *International Journal of Greenhouse Gas Control* **5**: 975 – 985.

Youssef A.A., Tran L., and Matth¨ai S. 2021. Impact of the vertical resolution of corner-point grids on $CO_2$ plume migration predictions for layered aquifers. *International Journal of Greenhouse Gas Control* **106**: 103249.

Zhang Z. and Agarwal R. 2012. Numerical simulation and optimization of $CO_2$ sequestration in saline aquifers for vertical and horizontal well injection. *Computational Geosciences* **16**: 891 – 899.

Zheng F., Jahandideh A., Jha B., et al. 2021. Geologic $CO_2$ storage optimization under geomechanical risk using coupled-physics models. *International Journal of Greenhouse Gas Control* **110**: 103385.

Zhou Q., Birkholzer J.T., Tsang C.F., et al. 2008. A method for quick assessment of $CO_2$ storage capacity in closed and semi-closed saline formations. *International Journal of Greenhouse Gas Control* **2**: 626 – 639.

Zou A., Ye T., Volkov O., et al. 2022. Effective treatment of geometric constraints in derivative-free well placement optimization. *Journal of Petroleum Science and Engineering* **215**: 110635.