Many-Objective Optimization Algorithm Applied to History Matching

Junko Hutahaean, Vasily Demyanov, Mike Christie
Energy, Geoscience, Infrastructure, and Society, Heriot-Watt University
Edinburgh, EH14 4AS, United Kingdom
Email: {jjh30, v.demyanov, m.christie}@hw.ac.uk

***Abstract*—Reservoir model calibration, called history matching in the petroleum industry, is an important task to make more accurate predictions for better reservoir management. Providing an ensemble of good matched reservoir models from history matching is essential to reproduce the observed production data from a field and to forecast reservoir performance. The nature of history matching is multi-objective because there are multiple match criteria or misfit from different production data, wells and regions in the field. In many cases, these criteria are conflicting and can be handled by the multi-objective approach. Moreover, multi-objective provides faster misfit convergence and more robust towards stochastic nature of optimization algorithms. However, reservoir history matching may feature far too many objectives that can be efficiently handled by conventional multi-objective algorithms, such as multi-objective particle swarm optimizer (MOPSO) and non-dominated sorting genetic algorithm II (NSGA II). Under an increasing number of objectives, the performance of multi-objective history matching by these algorithms deteriorates (lower match quality and slower misfit convergence). In this work, we introduce a recently proposed algorithm for many-objective optimization problem, known as reference vector-guided evolutionary algorithm (RVEA), to history matching. We apply the algorithm to history matching a synthetic reservoir model and a real field case study with more than three objectives. The paper demonstrates the superiority of the proposed RVEA to the state of the art multi-objective history matching algorithms, namely MOPSO and NSGA II.**

Keywords—history matching; multi-objective optimization; many-objective optimization; reservoir simulation

I. INTRODUCTION

History matching is the process of calibration of a reservoir model to obtain an acceptable match between the simulated model response and the observed dynamic data from the fields, wells or regions by adjusting uncertain model parameters. The dynamic data can be pressure, gas to oil ratio, water cut, oil production rate, and water production rate. Model parameters that may be modified include rock properties (porosity, horizontal and vertical permeability), fluid properties (compressibility, oil and water relative permeability, capillary pressure), and geological properties (fault transmissibility, aquifer volume).

The nature of history matching as an ill-posed inverse problem with non-unique solutions makes it one of the most challenging and resource-intensive stages in reservoir management. To overcome this challenge, history matching powered by stochastic population-based sampling algorithms has been used widely to generate an ensemble of matched models [1]–[3]. These algorithms minimize objective function (defined as a measure of discrepancy between simulated and observed data) and thus obtain a reservoir model that best approximates the dynamic variable data recorded during reservoir life. The outcome of this process is an ensemble of matched models due to the non-uniqueness solution of the inverse problem. Providing an ensemble of good matched models is essential to support the decision-making process based on the predictions from these models.

The nature of reservoir history matching is multi-objective because there are multiple wells and match criteria in well and regions in the field. In many cases, these match criteria are conflicting, and no feasible solution optimizes all of them simultaneously. For instance, an improvement in oil rate match in one well causes a deterioration of the gas rate in another well. In this case, we look for acceptable trade-off rather than a unique solution, referred to as Pareto optimality. A set of feasible solution vectors is called Pareto optimal if no other feasible vector would improve some criteria without causing a simultaneous deterioration of at least one other criteria. All Pareto optimal vectors are called non-dominated.

History matching involving multiple and conflicting objectives have been traditionally handled by single-objective optimization [1]–[3]. In single objective history matching, we sum up all the objective components from different wells by the weighted-sum and, next, solving the equivalent single objective optimization problem to obtain the best possible matches. This approach reduces the algorithm's sensitivity to variations in match quality within different parts of the reservoir and across various production data. Moreover, it is difficult to obtain the right set of weighting factor due to the behavior of observed production data from different wells and regions can be strongly uncorrelated. Additionally, the weight also depends on the scaling of each objective which may have a different order of magnitude. The use of this single objective approach without appropriate scaling results in extremely rough response surface of objective functions.

A multi-objective optimization approach can be used to address the issues mentioned above in the single objective

This work is partially funded by the Adrian Todd Golden Key Student Support Fund.

history matching [4]–[9]. It can effectively handle multiple and conflicting objectives in the reservoir history matching. A multi-objective approach uses a number of objective components to guide the algorithm to different areas of objective space that lead to a diverse set of models with high match quality. Moreover, multi-objective history matching is faster and more robust towards stochastic nature of optimization algorithm [5]–[7]. A recent study in [10] demonstrates that ensemble of matched-models from multi-objective history matching, coupled with Bayesian uncertainty quantification framework, provides more reliable reservoir forecasting than single objective history matching under uncertainty in the model parameterization.

However, reservoir history matching may feature far too many objectives that can be efficiently handled by conventional multi-objective algorithms. For example, a medium-size oil field can have more than 100 wells and multiple production data to match in history matching. In this situation, multi-objective history matching handled by Pareto and dominance-based algorithms, such as MOPSO and NSGA II, are less effective as the majority of solutions are non-comparable due to most points becoming non-dominated in high-dimensional objective space. As the results, the performance of multi-objective history matching deteriorates with a high number of objectives, i.e. slower misfit convergence and lower match quality as illustrated in Fig. 1(a). Fig. 1(b) shows the exponential increase of non-dominated solutions with increasing number of objectives from multi-objective history matching a synthetic reservoir model. Similar figures can be found in [11].

This paper applies one of the recent many-objective optimization algorithms, a reference vector-guided evolutionary algorithm (RVEA) [12], to history matching. The algorithm utilizes predefined reference vectors to balance the convergence and diversity. The reference vector-guided selection method helps convergence by successfully addressing the loss of selection pressure in many-objective optimization. The solutions diversity can also be effectively achieved with the aid of reference vectors. A diverse set of matched models is important to generate better reservoir forecasting. Two case studies are presented: history matching of a standard benchmarking reservoir model and a real field case study.

II. Many-Objective History Matching and Case Studies

In a minimization problem, without loss of generality, the multi-objective optimization can be defined as:

$$\begin{cases} minimize\ \left(f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), \dots, f_M(\boldsymbol{x})\right) \\ \quad subject\ to\ h_k^l \le x_k \le h_k^u \\ \quad \boldsymbol{x} = \{x_1, x_2, \dots, x_k, \dots, x_N\} \end{cases} \tag{1}$$

where $F(\boldsymbol{x}) = \{f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), \dots, f_M(\boldsymbol{x}): \mathbb{R}^N \to \mathbb{R}^M$, $\boldsymbol{x} = \{x_1, x_2, \dots, x_k, \dots, x_N\}$ is the vector of the N model parameters, while M is the number of objective functions, and the variables h_k^l and h_k^u represent the lower and upper boundary for each model parameter. The objective function in history matching measures the discrepancy between reservoir model response and the observation data. It is generally called the *misfit* and is the negative log of likelihood. In history matching, the misfit generally comprises multiple components as a typical petroleum reservoir is characterized by multiple wells and production data.

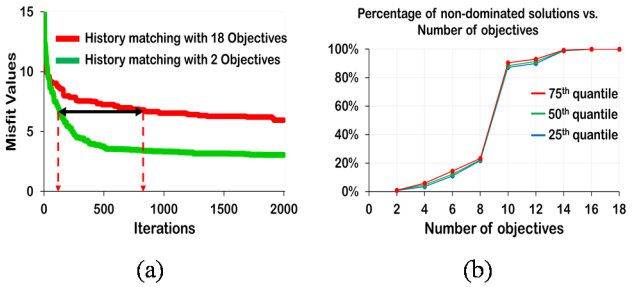


Fig. 1 (a) An example of the deterioration of performance on multi-objective history matching with increasing number of objectives; (b) Percentage of the non-dominated solutions with increasing number of objectives in multi-objective history matching a synthetic reservoir model.

From this, we can see that the number of misfit components to be minimized can easily be more than three, as the reservoir is mostly developed with more than one well and for each well there are multiple observed production data to match. This makes history matching a many-objective optimization problem.

In the petroleum literature, some studies have dealt with more than three objectives in history matching. Han [13] used a multi-objective evolutionary algorithm (MOEA) for history matching with four objective functions. They demonstrated that use of multi-objective history matching improves the predictability of production forecasts in a waterflooding project. Similarly, in [14], non-dominated sorting genetic algorithm II (NSGA II) was applied for history matching with four objective functions to measure the mismatch of the geological and seismic data. In [7], Christie *et al*. applied multi-objective particle swarm optimization (MOPSO) to four-objective history matching on a real field case study. In this study, we use MOPSO and NSGA II as the benchmarks due to the successful application of both algorithms to real field case studies and codes availability.

We use two case studies: a small synthetic reservoir model based on a real field that allows for a reasonable number of flow simulations and another larger, real oil field with more expensive objective function evaluations. Examining these case studies allows us to demonstrate the applicability of the proposed algorithm to real-world problems.

A. PUNQ-S3

1) Problem: PUNQ-S3 (Production forecasting with UNcertainty Quantification variant 3) is an industry standard benchmark for history matching, uncertainty quantification, and optimization. It is a synthetic reservoir model based on a real field operated by Elf Exploration and Production [15]. A dome shape reservoir model has a gas cap and oil rim and is bounded on the east and south by a fault as shown in Fig. 2(a). The reservoir model has 2660 grid blocks (19x28x5), of which 1761 are active. The grid blocks have equal sizes in the *x* and *y* directions of 180 m. There are six production wells, namely PRO-1, PRO-4, PRO-5, PRO-11, PRO-12, and PRO-15, which are located near the initial gas-oil contact (GOC).

The truth case was generated using Gaussian Random Fields for porosity and permeability properties [16]. Pressure,

volume and temperature (PVT) data from the original model was used to complete the model. Reservoir simulation was then used to generate production data (bottom hole pressure BHP, water cut WCT, and gas-oil ratio GOR) and Gaussian noise was added to account for measurement error.

2) Model Parameters and Objective Functions: We use a layer cake model description with 5 vertical layers and 9 lateral zones following the original dataset from the benchmark study [16]. The model parameterization following [10] (Parameterization Set 1) includes 12 regions: 9 sand channels (3 channels each in layer 1, 3, and 5), 1 floodplain, 1 homogeneous layer 2, and 1 homogeneous layer 4. In total, there are 24 parameters to modify in history matching consisting of 12 parameters for porosity (φ) and 12 parameters for horizontal permeability (k_h) multiplier (*Mult*) which are related as in (2) and (3). The ranges of these parameters with the implied permeability ranges are shown in Table I.

$$\log(k_h) = \log(Mult) + (0.77 + 9.03\varphi) \quad (2)$$

$$k_v = 3.124 + 0.306 k_h \quad (3)$$

TABLE I. POROSITY AND MULTIPLIER RANGE WITH THE IMPLIED HORIZONTAL AND VERTICAL PERMEABILITY IN PUNQ-S3 CASE STUDY

Zone	φ	*Mult*	k_h (mD)	k_v (mD)
Layer 1, 3, 5	0.15 – 0.30	0.1 – 10	133 – 3013	44 – 925
Background	0.05 – 0.15	0.1 – 10	16 – 133	8 – 44
Layer 2 and 4	0.05 – 0.15	0.1 – 10	16 – 133	8 – 44

Eight years of production history data including bottom hole pressure (*BHP*), water cut (*WCT*) and the gas-oil ratio (*GOR*) from all wells are used for the history matching. The data are uncorrelated and following the original dataset [16]. The objective function, misfit, to be minimized is defined as:

$$M = \frac{1}{n_w}\sum_i \frac{1}{n_p}\sum_j \frac{1}{n_t}\sum_k \left(w_{ijk} \frac{(obs_{ijk} - sim_{ijk})}{\sigma_{ijk}} \right)^2 \quad (4)$$

where n_w is the number of evaluated wells with i runs over it, n_p is the number of observed production data with j runs over it, n_t is the number of time steps for the j^{th} history data with k runs over it, obs is the observed history, sim is the simulated value, σ^2 is the variance of the measurement errors, and w is the weight factor, with runs over i, j and k. From (4), we, can see that there are 18 misfit components in PUNQ-S3 (misfits from 6 production wells with 3 production data from each well to match).

In this study, we approach the history matching with many-objective optimization of 6 objective functions by decomposing the misfit function in (4) based on production wells as in (5)

$$\begin{cases} minimize\ (M_1(p), M_2(p), \dots, M_6(p)) \\ \boldsymbol{p} = \{BHP, WCT, GOR\} \end{cases} \quad (5)$$

where M_1 to M_6 correspond to the misfits from production wells PRO-1, PRO-4, PRO-5, PRO-11, PRO-12 and PRO-15 respectively, which are summed up on all production variable $\boldsymbol{p}$ over all time steps with unity weights. For instance, $M_1 = (M_{BHP} + M_{WCT} + M_{GOR})_{PRO-1}$.

B. Zagadka

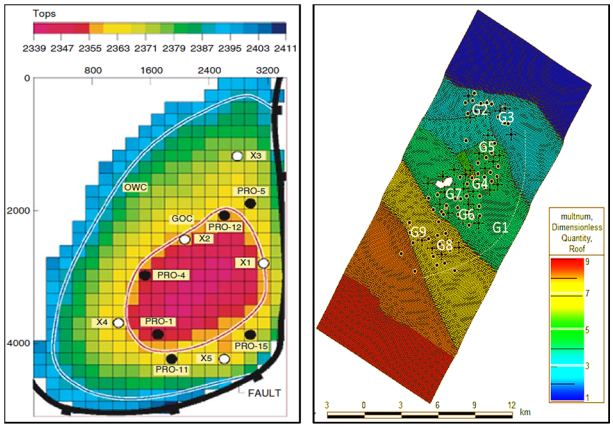
1) Problem: The Zagadka field is located in Western Siberia and is a medium-size oil field. It has 95 wells, some of which have 10-15 years of history. The field is produced by a combination of reservoir aquifer and water injection for pressure support to maintain production rates. The reservoir model has a dimension of 108 x 265 x 7 with around 135,000 active grid blocks. We are supplied with the model consisting of 95 wells in 9 groups. The field is compartmentalized with possible sealing faults creating seven compartments. Fig. 2(b) shows the multiplier region map of the Zagadka reservoir model. More details about the field can be found in [7].

2) Model Parameters and Objective Functions: We use the same model parameterization as in [7] for history matching study. The parameterization depicts geological structure zonation of the field. It has 19 parameters: 7 global k_h multiplier, 1 global k_v multiplier, 3 capillary pressure (P_c) values, 4 oil relative permeability (k_{ro}) values, 1 water relative permeability (k_{rw}) value, 1 fault transmissibility, and 2 aquifer support multipliers. Table II gives the description and range of these parameters.

TABLE II. PARAMETER RANGES IN ZAGADKA CASE STUDY

Parameter	Number	Range
k_h multiplier	7	1 – 30
k_v multiplier	1	0.1 – 0.9
Capillary pressure 1	1	18 – 23
Capillary pressure 2	1	9 – 13
Capillary pressure 3	1	1.01 – 3.0
Oil relative permeability 1	1	0.5 – 1
Oil relative permeability 2	1	0.6 – 0.9
Oil relative permeability 3	1	0.1 – 0.3
Oil relative permeability 4	1	0.02 – 0.095
Water relative permeability	1	0.2 – 1.25
Fault transmissibility	1	0.0 – 1.0
Aquifer support multiplier	2	2.5 – 4.5

In Zagadka, all wells are grouped into 9 groups: G1 to G9 based on the geological structure, fault block in the model, and the time when the wells were drilled. The wells in the group G1 are the exploration wells located in different regions across the



(a) (b)

Fig. 2 (a) PUNQ-S3 and (b) Zagadka reservoir model.

field, whereas the wells in the other groups are clustered in particular regions of the field following the drilling schedule.

We use the historical data of the group oil and water production rates (*GOPR* and *GWPR*) in the history matching. The objective function to be minimized is defined as:

$$M = \sum_i^{n_g} \sum_j^{n_p} w_{ij} \sum_k^{n_t} \frac{(obs_{ijk} - sim_{ijk})^2}{2\sigma_{ijk}^2} \quad (6)$$

where n_g is the number of groups with i runs over it, n_p is the number of observed production data with j runs over it, n_t is the number of time steps for the j^{th} history data with k runs over it, obs is the observed history, sim is the simulated value, σ^2 is the variance of the measurement errors, and w is the weight factor, with runs over i and j. From (6), we can see that there are 18 misfit components in Zagadka (misfits from 9 groups with 2 production data from each group to match).

In this study, we group these 18 misfit components into four objectives as in (7) based on production start time from each group and four predominant clusters where the wells are located:

$$\begin{cases} minimize \left(M_1(p), M_2(p), \dots, M_4(p)\right) \\ \boldsymbol{p} = \{GOPR, GWPR\} \end{cases} \quad (7)$$

where M_1 to M_4 correspond to the misfit from groups (G1), (G2 and G3), (G4 to G7), and (G8 and G9), respectively, which are summed up on all production variable $\boldsymbol{p}$ over all time steps with unity weights. For instance, $M_2 = (M_{GOPR} + M_{GWPR})_{G2} + (M_{GOPR} + M_{GWPR})_{G3}$.

III. Reference Vector-Guided Evolutionary Algorithm

In recent years, methods for improving the performance of Evolutionary Algorithms (EAs) in many-objective optimization problems have gained popularity since problems of this type abound in real world applications of science and engineering [17]. A comprehensive review of these different methods can be found in [18].

In this section, we describe reference vector-guided evolutionary algorithm (RVEA) [12], a recently proposed algorithm to solve many-objective optimization problems. The general framework of the RVEA algorithm is shown in Algorithm 1. RVEA has a similar elitism strategy to non-dominated sorting genetic algorithm II (NSGA II) [19]. However, compared to NSGA II, there are three main differences with RVEA: (1) a set of predefined reference vectors is required for input in RVEA; (2) reference vectors are used to guide the selection of elitist solutions; (3) there is a reference vector adaptation strategy to cope with different scales between objectives. Traditional genetic operators, i.e. simulated binary crossover (SBX) [20] and polynomial mutation [21], are used to generate the offspring population and then combined with the parent population for elitism selection. Steps 1, 9, and 10 in the Algorithm 1 will be further detailed in sections III. A, B, and C.

Algorithm 1 The general framework of RVEA [12]

1: **Input**: maximum number of generations t_{max}, a set of unit reference vectors $V_0 = \{\boldsymbol{v}_{0,1}, \boldsymbol{v}_{0,2,\dots}, \boldsymbol{v}_{0,N}\}$;
2: **Output**: final population $P_{t_{max}}$;
3: /*Initialization*/
4: **Initialization**: create the initial Population P_0 with N randomized individuals;
5: /* Main Loop*/
6: **while** $t < t_{max}$ **do**
7: Q_t = offspring creation (P_t);
8: $P_t = P_t \cup Q_t$;
9: P_{t+1} = reference vector guided selection (t,P_t,V_t);
10: V_{t+1} = reference vector adaptation (t,P_{t+1},V_t, V_0);
11: $t = t + 1$;
12: **end while**

A. Reference Vector

Reference vectors are used to divide the objective space into smaller subspaces. Its usage has been demonstrated successfully to improve the convergence and diversity of many-objective optimization problem [12], [22].

In this work, without loss of generality, we use unit reference vectors generated uniformly in the first quadrant with the origin as the initial point as illustrated in Fig. 3. The canonical simplex-lattice design method [23] is employed to generate a set of uniformly distributed points on the hyperplane as in (8) and then mapped to the hypersphere by the transformation as in (9) [12]:

$$\begin{cases} \boldsymbol{u}_i = (u_i^1, u_i^2, \dots, u_i^M) \\ u_i^j \in \left\{\frac{0}{H}, \frac{1}{H}, \dots, \frac{H}{H}\right\}, \sum_{j=1}^M u_i^j = 1 \end{cases} \quad (8)$$

$$\boldsymbol{v}_i = \frac{\boldsymbol{u}_i}{\|\boldsymbol{u}_i\|} \quad (9)$$

where $i = 1, \dots, N$ with N being the number of uniformly distributed points, M is the number of objectives, H is a positive integer for the simplex lattice design, and v_i are the unit reference vectors. A total number of $N = \binom{H+M-1}{M-1}$ uniformly distributed reference vectors can be generated, given M and H, according to the property of simplex-lattice design. For instance, there are $N = 10$ reference vectors if we define $H = 3$ in $M = 3$ objective problem.

B. Reference Vector-Guided Selection

In RVEA, selection of the next generation is done separately on each subspace partitioned by the reference vectors. This selection strategy consists of three steps: (1) objective value translation; (2) population partition; and (3) angle penalized distance (APD) calculation, which is followed by the elitism selection [12].

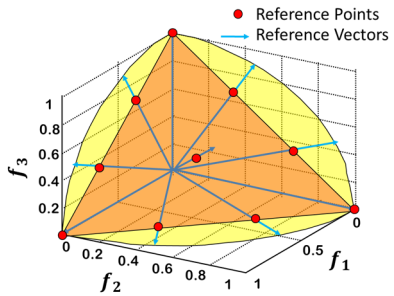


Fig. 3 An example of generating ten uniformly distributed unit reference vectors in a three-objective optimization problem (after [12]).

1) Objective Value Translation: The objective value translation makes sure the initial point of the reference vectors is always the origin, and all the translated objective values are in the first quadrant, as illustrated in Fig. 4 for the two-objective optimization problem. Objective values in population P_t, denoted as $F_t = \{\boldsymbol{f}_{t,1}, \boldsymbol{f}_{t,2}, \dots, \boldsymbol{f}_{t,|P_t|}\}$, where t is the generation index, are translated into F_t' by:

$$\mathbf{f}'_{t,i} = \mathbf{f}_{t,i} - \mathbf{z}_t^{min} \tag{10}$$

where $i = 1, \dots, |P_t|$, $\mathbf{f}_{t,i}$, $\mathbf{f}'_{t,i}$ are the objective vectors of individual i before and after the translation, and $\mathbf{z}_t^{min} = (z_{t,1}^{min}, z_{t,2}^{min}, \dots, z_{t,m}^{min})$ represents the minimal objective values calculated from F_t.

2) Population Partition: The translated objective values in population P_t are partitioned to N subpopulations $P_{t,1}, P_{t,2}, \dots, P_{t,N}$ by associating each individual to the closest reference vector, as illustrated in Fig. 5, where N is the number of reference vectors. The spatial relationship between each translated individual subpopulation to the reference vector is determined by the acute angle, as in (11). Additionally, the closest reference vector to which the individual subpopulation will be allocated is based on the largest cosine value of (11), as in (12):

$$\cos \theta_{t,i,j} = \frac{\mathbf{f}'_{t,i} \cdot \mathbf{v}_{t,j}}{\|f'_{t,i}\|} \tag{11}$$

$$P_{t,k} = \{I_{t,i} | k = \underset{j \in \{1,\dots,N\}}{\operatorname{argmax}} \cos \theta_{t,i,j}\} \tag{12}$$

where $\theta_{t,i,j}$ is the angle between objective vector $\mathbf{f}'_{t,i}$, and the reference vector $\mathbf{v}_{t,j}$, $I_{t,i}$ represents the I-th individual subpopulation in P_t, with $i = 1, \dots, |P_t|$.

3) Angle Penalized Distance (APD) Calculation: In RVEA, the ideal point is always the axis origin as the objective values have been translated in (10). Hence, the convergence criterion can be represented by the distance of $\boldsymbol{f}'_{t,i}$ to the origin, i.e., $\|\boldsymbol{f}'_{t,i}\|$, whereas the diversity criterion is naturally represented by the angle $\theta_{t,i,j}$, as in (11). Angle-penalized distance, $d_{t,i,j}$, is introduced to balance these criteria (convergence and diversity), an important feature of RVEA for history matching problem, with a scalarization as in (13):

$$d_{t,i,j} = \left(1 + P(\theta_{t,i,j})\right) \cdot \|\mathbf{f}'_{t,i}\| \tag{13}$$

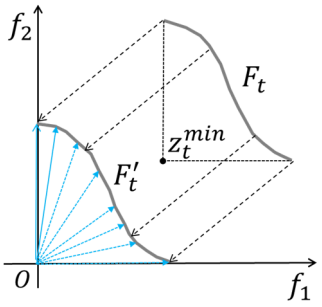


Fig. 4 An illustration of the objective value translation in two objective optimization problem (after [12]).

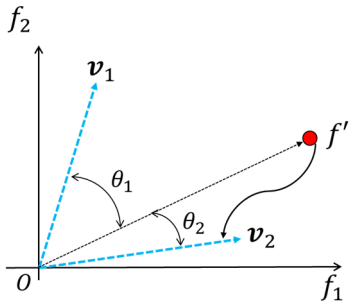


Fig. 5 An illustration of the population partition in objective space for 2 objective optimization problem (after [12]).

with $P(\theta_{t,i,j})$ being a penalty function related to $\theta_{t,i,j}$:

$$P(\theta_{t,i,j}) = M \cdot \left(\frac{t}{t_{max}}\right)^{\alpha} \cdot \frac{\theta_{t,i,j}}{\gamma_{\boldsymbol{v}_{t,j}}} \tag{14}$$

$$\gamma_{\boldsymbol{v}_{t,j}} = \min_{i \in \{1,\dots,N\}, i \neq j} \langle \boldsymbol{v}_{t,i}, \boldsymbol{v}_{t,j} \rangle \tag{15}$$

where M is the number of objectives, N is the number of reference vectors, t_{max} is the maximum number of generations, α is the parameter governing the rate of change of $P(\theta_{t,i,j})$, and $\gamma_{\boldsymbol{v}_{t,j}}$ is the smallest angle value between reference vector $\mathbf{v}_{t,j}$ and the other reference vectors in the current generation. Angle $\gamma_{\boldsymbol{v}_{t,j}}$ is used to normalize the angle specified by $\mathbf{v}_{t,j}$.

Following all three of these steps, the elitist solutions selection is performed based on the smallest APD, as in (13) and (14), for the population in the next generation. Further details on the reference-guided selection can be found in [12].

C. Reference Vector Adaptation

In real world optimization problems, we often encounter an issue with different scales of objective values from different objectives. In this problem, uniformly distributed reference vectors will not produce uniformly distributed solutions as illustrated in Fig. 6(a). A reference vector adaptation strategy is employed in RVEA to cope with this problem as illustrated in Fig. 6(b). It works by adapting the reference vectors based on the ranges of the objective values as in (16)

$$v_{t+1,i} = \frac{v_{0,i} \circ (z_{t+1}^{max} - z_{t+1}^{min})}{\|v_{0,i} \circ (z_{t+1}^{max} - z_{t+1}^{min})\|} \tag{16}$$

where $i = 1, \dots, N$, $v_{t+1,i}$ represents the i-th adapted reference vector for the next generation $t+1$, $v_{0,i}$ represents the i-th

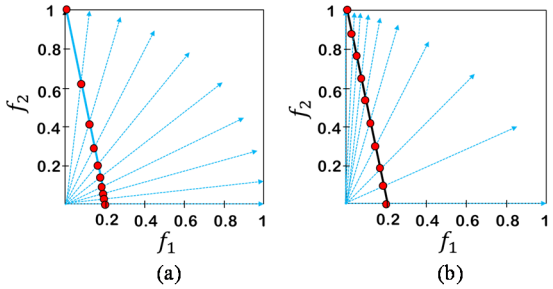


Fig. 6 The Pareto optimal solutions (red dots) with objective values scale of different ranges specified by (a) 10 uniformly distributed reference vectors and (b) 10 adapted reference vectors on a Pareto Front. Objective f_1 has values range 0-0.2 and objective f_2 has scale range 0-1. (after [12]).

initial uniformly distributed reference vectors, z_{t+1}^{max} and z_{t+1}^{min} represent the maximum and minimum values of the objective in the $t+1$ generation respectively, and the o operator represents the Hadamard product. By applying this adaptation strategy, the RVEA can obtain uniformly distributed solutions on the problem with different scale of objective value ranges.

The reference vector adaptation strategy should not be used too frequently during the search process to ensure a stable convergence [24]. A parameter f_r is introduced in RVEA to control the frequency of using the adaptation strategy. Further details on the reference vector adaptation strategy can be found in [12].

IV. Comparative Results

In this section, we compare the many-objective history matching results on both case studies described earlier from RVEA with MOPSO [5] and NSGA II [19]. For each case study, 10 trial history matching runs are performed, where the variation between run is caused by the initial choice of random seed and the subsequently generated random numbers in the stochastic optimizer.

A. Parameter Settings

Table III describes the parameter settings for each algorithm in both case studies. A termination criterion of 1500 and 250 function evaluations is set for PUNQ-S3 and Zagadka case studies, respectively, as there is no significant improvement of misfit value beyond these iterations.

TABLE III. Parameter Settings for each Algorithm

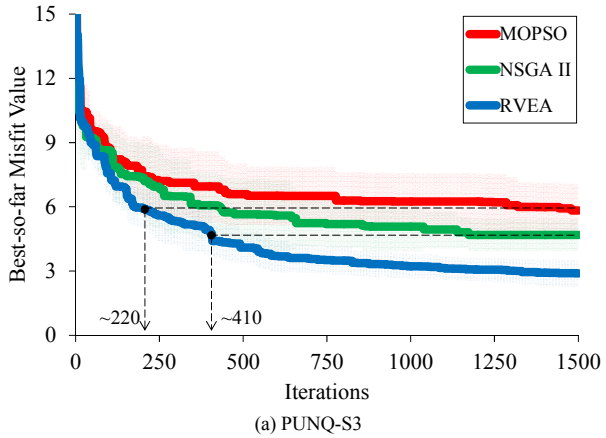
Algorithm	Parameters	PUNQ-S3 (6 Objectives)	Zagadka (4 Objectives)
RVEA	(H_1, H_2)	(2, 0)	(2, 0)
	(α, f_r)	(2, 0.1)	(2, 0.1)
	$(\eta_c, p_c, \eta_m, p_m)$[a]	(30, 1, 20, 1/24)	(30, 1, 20, 1/19)
	Population size	21	20
	Ref. vectors	21	10
MOPSO	(ω, c_1, c_2)[b]	(0.729, 1.494, 1.494)	(0.729, 1.494, 1.494)
	Population size	20	20
NSGA II	$(\eta_c, p_c, \eta_m, p_m)$[a]	(30, 1, 20, 1/24)	(30, 1, 20, 1/19)
	Population size	20	20

[a] SBX distribution index (η_c), crossover probability (p_c), polynomial mutation distribution index (η_m) and probability (p_m).
[b] Inertia weight (ω), cognitive (c_1) and social (c_2) parameters.

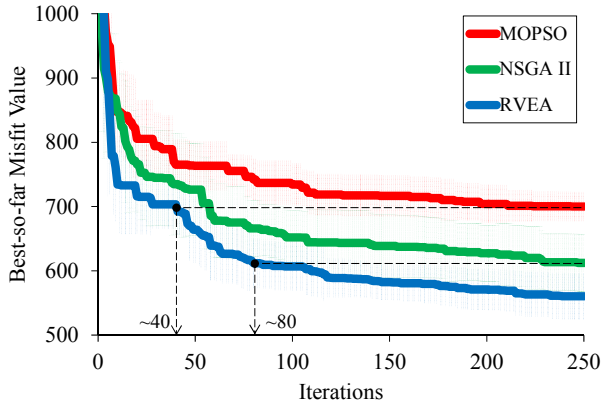
B. Performance Measures

Maintaining a good balance between misfit convergence and diversity of the matched-models is particularly crucial to the history matching powered by stochastic algorithms. The goal in history matching is to obtain high quality of matched-models fast and an ensemble of diverse matched-models for robust and reliable forecasting [10].

1) Misfit Value Evaluations and Convergence Speed: We compute the global sum of misfits after many-objective history matching runs from all three algorithms to evaluate the quality of matched-models. Fig. 7(a) and Fig. 7(b) show the average and standard deviation of best-so-far misfit values evolutions



(a) PUNQ-S3



(b) Zagadka

Fig. 7 Average and standard deviation of best-so-far misfit value over 10 trial history matching runs from three different algorithms on (a) PUNQ-S3 and (b) Zagadka case studies.

from 10 trial history matching runs from each algorithm for PUNQ-S3 and Zagadka case studies, respectively. In both case studies, the average of best-so-far misfit values of history matching with RVEA is predominantly below the lower bound of best-so-far misfit values from history matching with MOPSO and NSGA II. These figures also show that the upper bound of best-so-far misfit values from RVEA are always below the average of best-so-far misfit values from MOPSO and NSGA II. These results demonstrate not only the robustness of history matching with RVEA but also its better performance than MOPSO and NSGA II for obtaining high quality matched models given the same number of flow simulations

Significant improvements of history matching with RVEA from MOPSO and NSGA II are summarized in Fig. 8. Fig. 8 shows the boxplot of the final best-so-far misfit values from 10 trial history matching runs from each algorithm for PUNQ-S3 (a) and Zagadka (b). We use the Wilcoxon rank sum test for the statistical significant test as it is more sensitive than the *t*-test, safer since it does not assume a normal distribution and the outliers have less effect [25]. In both case studies, the improvements of history matching with RVEA on the final misfit values from MOPSO and NSGA II are highly significant

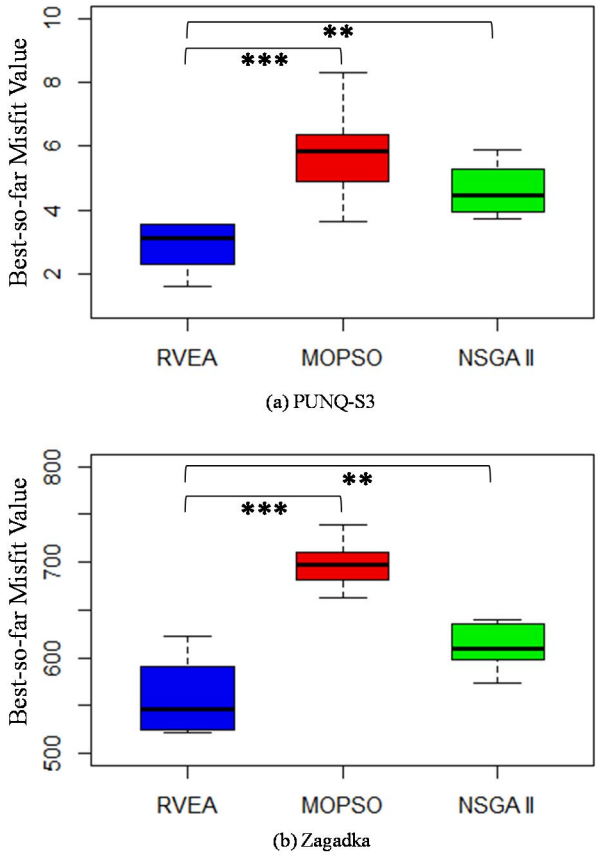


Fig. 8 Box plot of best-so-far misfit value at the end iterations over 10 trial history matching runs from three different algorithms with statistical significance test comparing RVEA with MOPSO and NSGA II on (a) PUNQ-S3 and (b) Zagadka. (**:p-value < 0.01; ***:p-value < 0.001).

(p-value = 0.00098 and p-value = 0.002, respectively) as shown in Fig. 8(a) and Fig. 8(b).

History matching with RVEA also provides faster misfit convergence than MOPSO and NSGA II in both the PUNQ-S3 and Zagadka case studies, as shown in Fig. 7. In PUNQ-S3, RVEA is 6.8 and 3.6 times faster than MOPSO and NSGA II, respectively, as demonstrated in Fig. 7(a). In Zagadka, history matching with RVEA is 6.25 and 3.1 times faster than MOPSO and NSGA II, as shown in Fig. 7(b).

2) Diversity of the Matched-Models: We use radial coordinate visualization (RadViz) [26] to evaluate the diversity of the matched-models in the objective space. RadViz is a multi-dimensional visualization technique that can display data of three or more objectives in a 2-dimensional projection. The objectives (called dimensional anchors) are distributed evenly along the perimeter of the unit circle after normalization. Each objective vector is held with springs that are attached to the anchors, and the spring force is proportional to the value of the corresponding objective or anchor. The objective vector is depicted as a point in the circle and located in which the spring forces are in equilibrium. For example, a point that is located close to an anchor of one objective have a higher value in that objective than in any other objective. Objective vector with all equal objective values is located exactly in the center of the circle.

A diverse set of models can be identified by the distribution of the points to different areas in the circle. The points where are concentrated in one particular area in the circle indicate less diverse models than the ones where are more dispersed to different areas in the circle.

Fig. 9(a) and Fig. 9(b) show the RadViz plots of all the matched-models from one run in PUNQ-S3 and Zagadka case studies, respectively, with three different algorithms and color-coded with their total misfit values. A similar trend is observed with the rest of the nine runs. In PUNQ-S3, history matching with RVEA provides more diverse set and higher quality of matched-models than MOPSO as in Fig. 9(a.3, a.1). The diversity of the matched models from NSGA II is comparable with RVEA as showed in Fig. 9(a.2, a.3). However, the quality of most matched-models from NSGA II are lower (higher misfit value) than RVEA. History matching with MOPSO results in the worst performance amongst others (low match quality and the least diverse set of matched models) as showed in Fig. 9(a.1). In Zagadka, history matching from all algorithms shares a similar diversity of the matched-models as showed in Fig. 9(b). However, history matching from RVEA results in models with higher quality than NSGA II and MOPSO. These results demonstrate that, from the three compared algorithms and presented case studies; RVEA is the best algorithm to provide high quality and diverse matched-models followed by NSGA II and MOPSO consecutively.

V. Conclusion and Future Research

We demonstrated one of the challenges of multi-objective history matching with a high number of objectives (more than three) by dominance-based multi-objective algorithms. As the number of objectives increases, the performance of these

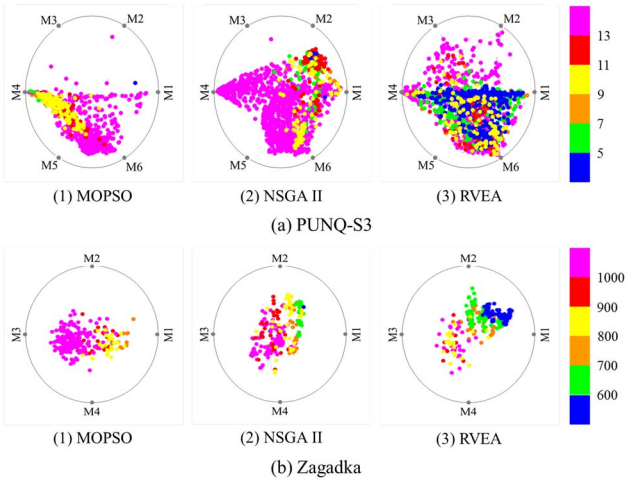


Fig. 9 Radial visualizations of all the models from history matching of (a) PUNQ-S3 and (b) Zagadka, with three different algorithms and color-coded with their total misfit values. Lower misfit values indicates higher quality models, and vice versa. M_i is the objective as defined in Eq. (5) and Eq. (7) for PUNQ-S3 (6 objectives) and Zagadka (4 objectives), respectively, with their values are normalized in this plot.

algorithms, e.g. MOPSO and NSGA II, deteriorates, producing slower misfit convergence and lower match quality.

We introduced a recently proposed many-objective algorithm, RVEA, to solve history matching problems with high numbers of objectives. We applied the algorithm to history matching of the PUNQ-S3 synthetic reservoir model, with six objectives, and a real field Zagadka, with four objectives. In both case studies, history matching powered by RVEA can achieve fast misfit convergence, high match quality, and a diverse set of matched models.

We compared history matching results by RVEA with MOPSO and NSGA II. We observed a significant improvement in history matching performances by RVEA from the ones with MOPSO and NSGA II based on statistical test over 10 history matching runs. History matching with RVEA results in more robust misfit evaluation (smaller variation between 10 runs) and better match quality than MOPSO and NSGA II. In PUNQ-S3, history matching with RVEA can achieve up to 6.8 and 3.6 times faster misfit convergence than history matching with MOPSO and NSGA II, respectively. In the real field case study, Zagadka, the misfit convergence from history matching by RVEA can be up to 6.25 and 3.1 times faster than MOPSO and NSGA II, respectively.

RVEA also successfully maintains the diversity of the matched models in both case studies which are essential for better forecasting. In general, history matching with RVEA provides a more diverse set of models than MOPSO and NSGA II. The diversity on the ensemble of these models ensures good uncertainty quantification in the forecasting period from these models [5]–[7], [10].

As for future research, the user preference articulation to define the reference vectors in RVEA should also be explored in real-world optimization problems. In this case, the search process to find optimal solutions can be guided by the set of reference vectors defined by the user preferential.

ACKNOWLEDGMENT

We thank BG, E.ON, JOGMEC, and RockFlowDynamics for financial support, Epistemy for providing RAVEN history matching software, RockFlowDynamics for providing tNavigator reservoir simulator and Schlumberger GeoQuest for providing ECLIPSE 100 reservoir simulator.

REFERENCES

[1] L. Mohamed, M. A. Christie, and V. Demyanov, "Reservoir Model History Matching with Particle Swarms: Variants Study," in *SPE-129152-MS*, Mumbai, India, 2010.

[2] Y. Hajizadeh, M. A. Christie, and V. Demyanov, "History matching with differential evolution approach; a look at new search strategies," in *SPE-130253-MS*, Barcelona, Spain, 2010.

[3] A. Abdollahzadeh, A. Reynolds, M. Christie, D. W. Corne, G. J. J. Williams, and B. J. Davies, "Estimation of Distribution Algorithms Applied to History Matching," *SPE J*, vol. 18, no. 3, pp. 508–517, Apr. 2013.

[4] R. W. Schulze-Riegert, M. Krosche, A. Fahimuddin, and S. G. Ghedan, "Multi-Objective Optimization with Application to Model Validation and Uncertainty Quantification," in *SPE-105313-MS*, Manama, Bahrain, 2007.

[5] L. Mohamed, M. A. Christie, and V. Demyanov, "History Matching and Uncertainty Quantification: Multiobjective Particle Swarm Optimisation Approach," in *SPE-143067-MS*, Vienna, Austria, 2011.

[6] Y. Hajizadeh, M. A. Christie, and V. Demyanov, "Towards Multiobjective History Matching: Faster Convergence and Uncertainty Quantification," in *SPE-141111-MS*, The Woodlands, Texas, USA, 2011.

[7] M. Christie *et al.*, "Use of multi-objective algorithms in history matching of a real field," in *SPE-163580-MS*, The Woodlands, Texas, USA, 2013.

[8] H.-Y. Park, A. Datta-Gupta, and M. J. King, "Handling Conflicting Multiple Objectives Using Pareto-Based Evolutionary Algorithm for History Matching of Reservoir Performance," in *SPE-163623-MS*, The Woodlands, Texas, USA, 2013.

[9] M. S. Kanfar and C. R. Clarkson, "Reconciling flowback and production data: A novel history matching approach for liquid rich shale wells," *Journal of Natural Gas Science and Engineering*, 2016.

[10] J. J. Hutahaean, V. Demyanow, and M. A. Christie, "Impact of Model Parameterisation and Objective Choices on Assisted History Matching and Reservoir Forecasting," in *SPE-176389-MS*, Nusa Dua, Bali, Indonesia, 2015.

[11] A. R. R. de Freitas, P. J. Fleming, and F. G. Guimarães, "Aggregation Trees for visualization and dimension reduction in many-objective optimization," *Information Sciences*, vol. 298, pp. 288–314, Mar. 2015.

[12] R. Cheng, Y. Jin, M. Olhofer, and B. Sendhoff, "A Reference Vector Guided Evolutionary Algorithm for Many-Objective Optimization," *IEEE Transactions on Evolutionary Computation*, pp. 1–19, 2016.

[13] Y. Han, C. Park, and J. M. Kang, "Estimation of Future Production Performance Based on Multi-objective History Matching in a Waterflooding Project," in *SPE-130500-MS*, Barcelona, Spain, 2010.

[14] M. E. Niri and D. E. Lumley, "A Multi-Objective Optimization Method for Creating Reservoir Models That Simultaneously Match Seismic and Geologic Data," in *SEG-2014-0065*, Denver, Colorado, USA, 2014.

[15] C. Bos, "Production forecasting with uncertainty quantification," *Final report of EC project, NITG-TNO report NITG*, pp. 99–255, 2000.

[16] *Imperial College London, PUNQ-S3*.

[17] P. J. Fleming, R. C. Purshouse, and R. J. Lygoe, "Many-objective optimization: An engineering design perspective," in *Evolutionary Multi-Criterion Optimization*, Guanajuato, Mexico, 2005, vol. 3410, pp. 14–32.

[18] C. von Lücken, B. Barán, and C. Brizuela, "A survey on multi-objective evolutionary algorithms for many-objective problems," *Comput Optim Appl*, vol. 58, no. 3, pp. 707–756, 2014.

[19] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans on Evol Comp*, vol. 6, no. 2, pp. 182–197, 2002.

[20] K. Deb and R. B. Agrawal, "Simulated binary crossover for continuous search space," *Complex Systems*, vol. 9, no. 2, pp. 115–148, 1995.

[21] K. Deb and M. Goyal, "A combined genetic adaptive search (GeneAS) for engineering design," *Computer Science and Informatics*, vol. 26, pp. 30–45, 1996.

[22] K. Deb and H. Jain, "An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: Solving problems with box constraints," *IEEE Trans on Evol Comp*, vol. 18, no. 4, pp. 577–601, 2014.

[23] J. A. Cornell, *Experiments with mixtures: designs, models, and the analysis of mixture data*, 3rd ed. Wiley, 2002.

[24] I. Giagkiozis, R. C. Purshouse, and P. J. Fleming, "Towards understanding the cost of adaptation in decomposition-based optimization algorithms," presented at the IEEE International Conference on Systems, Man, and Cybernetics, Manchester, 2013, pp. 615–620.

[25] J. Derrac, S. García, D. Molina, and F. Herrera, "A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms," *Swarm and Evolutionary Computation*, vol. 1, no. 1, pp. 3–18, 2011.

[26] P. Hoffman, G. Grinstein, K. Marx, I. Grosse, and E. Stanley, "DNA visual and analytic data mining," presented at the Visualization'97., Proceedings, 1997, pp. 437–441.