A random forest-regression-based inverse-modeling evolutionary algorithm using uniform reference points

Pezhman Gholamnezhad1 | Ali Broumandnia2 | Vahid Seydi2

1Faculty of Computer Engineering and Information Technology, Shahid Sattari University of Aeronautical Sciences and Technology, Tehran, Iran
2Department of Computer, South Tehran Branch, Islamic Azad University, Tehran, Iran

Correspondence
Ali Broumandnia, Department of Computer, South Tehran Branch, Islamic Azad University, Tehran, Iran.
Email: broumandnia@gmail.com

Abstract
The model-based evolutionary algorithms are divided into three groups: estimation of distribution algorithms, inverse modeling, and surrogate modeling. Existing inverse modeling is mainly applied to solve multi-objective optimization problems and is not suitable for many-objective optimization problems. Some inverse-model techniques, such as the inverse-model of multi-objective evolutionary algorithm, constructed from the Pareto front (PF) to the Pareto solution on nondominated solutions using a random grouping method and Gaussian process, were introduced. However, some of the most efficient inverse models might be eliminated during this procedure. Also, there are challenges, such as the presence of many local PFs and developing poor solutions when the population has no evident regularity. This paper proposes inverse modeling using random forest regression and uniform reference points that map all nondominated solutions from the objective space to the decision space to solve many-objective optimization problems. The proposed algorithm is evaluated using the benchmark test suite for evolutionary algorithms. The results show an improvement in diversity and convergence performance (quality indicators).

KEYWORDS
inverse modeling, many-objective optimization, model-based evolutionary algorithm, random forest regression, uniform reference points

1 | INTRODUCTION
1.1 | Background
The genetic operator or selection strategy plays a key role in the performance of multiobjective evolutionary algorithms (MOEAs) [1]. As a result, machine learning (ML) models have been used in evolutionary algorithms (EAs) to solve optimization problems [2, 3]. ML techniques are applied instead of heuristic operators, such as regression models (e.g., the Gaussian process), clustering models, or classification models. To map the objective space to the decision space, the model-based evolutionary algorithms (MBEAs) have recently been divided into three groups [2]: estimation of distribution algorithms (EDAs), surrogate-assisted evolutionary algorithms (SAEAs), and inverse-model evolutionary algorithms (IMEAs).

MBEAs face several challenges: In the EDAs, building an ML model is generally more time-consuming. Also,
there are constraints for training data. Furthermore, as the number of decision variables increases, the performance declines sharply. Generally, the EDAs are focused on estimating distributions in the decision space. The SAEAs are effective in selecting a set of passable candidate solutions with restricted computational resources. However, selecting the surrogate models is not straightforward, and the high-dimensional input data increases the computing time in the training model. The inverse model maps nondominated solutions from the Pareto front (PF) in the object space to the Pareto solutions (PS) in the decision space [4]. This algorithm differs from the SAEAs (SAEAs)[5, 6]. SAEAs use a map pattern of distribution of decision space to approximate the candidate solutions in objective space because there are no clear patterns or the fitness functions are expensive to compute. Inverse models evaluate the distribution of the Pareto optimal solutions by mapping the patterns from objective space to decision space. However, the development of many-objective optimization using the MBEAs is still in its infancy, with no proposal method developed so far, particularly in the inverse modeling group.

1.2 | Motivation and contributions

Although the SAEAs have been explored to address the MaOPs, the development of many-objective optimization on the MBEAs is still in its infancy. Particularly, the probability models based on the inverse model through the Gaussian process or other methods, such as linear regression, have not yet been employed on MaOPs. Since the computational order of all inverse models in the MOEA (IM-MOEA) is expensive, the random grouping method has been used to reduce them [7]. However, some potentially efficient inverse models are removed during the random grouping. This is a challenge when there are too many local PFs, which leads to poor solutions. Also, the random grouping procedure eliminates the possibility of applying efficient decision variables to other objective spaces. Furthermore, this procedure cannot specify which decision variables \( x_n \) are the most efficient to assign to the objective functions \( f_m \). Recently, inverse model-based multi-objective EDA using random-forest variable importance methods has been proposed to solve multi-objective optimization problems. [8]. However, this method is not suitable for solving many-objective problems because as the number of objective functions increases, the performance decreases sharply (the conventional multi-objective optimization methods are no longer applicable in many-objective optimization. In fact, their respective mechanism of maintaining diversity differs). Moreover, because uniform reference points can direct the search space toward exceptional solution locations, they can be employed in IM-MaOEAs. Also, Gaussian distributions frequently have a limited sample space, especially when the standard deviation \( \delta \) is small, which would limit an EDA’s exploration ability.

In this paper, a new approach is proposed, which is based on an inverse model with Bayesian linear and random forest regression along with uniform reference points: the random forest-regression-based inverse-modeling evolutionary algorithm (RFIEA) for many-objective optimization problems. In this method, the offspring was created using Bayesian linear regression by sampling the reference points and a set of uniformly distributed solutions from the PF in the objective space mapped to PS in the decision space, in which a Gaussian process with random forest variable importance algorithm [9, 10] is applied for mapping rich solutions from the PF to the PS. To summarize, the Gaussian process is used to approximate fitness values, and the random forest features importance method is applied in the inverse model to optimally allocate \( (x_n) \) to \( (f_m) \).

The remainder of this paper is organized as follows: In Section 2, the related work is reviewed. In Section 3, the proposed algorithm is presented. In Section 4, the experimental analysis of algorithms, outcomes, and computational complexities are demonstrated. Finally, in Section 5, conclusions and future work are given.

2 | RELATED WORK

The EAs are a subset of nature-inspired algorithms that are commonly used to solve optimization problems. In most conventional EAs, the operators (crossover and mutation) are used, but they cannot learn the structures of problems. Therefore, the MBEAs have been introduced, which are EAs that can learn. The MBEAs have been divided into three groups [2]: (1) EDAs, (2) SAEAs, and (3) IMEAs.

In the third group, the IM-MOEA was presented, in which nondominated solutions were transformed from the PF in the object space to the PS in the decision space [4]. The SAEAs transform the decision space to estimate the fitness value of solutions that are candidates for objective space when there is no obvious fitness function or calculating functions are expensive. The IMEAs have been introduced to approximate the distribution of Pareto optimal solutions. In IM-MOEA, when the PFs are atypical, some partitions may be ineffective. This is because some of the reference vectors that produced related partitions do not have enough number-rich solutions. Therefore, Cheng and others proposed an adaptive IM-MOEA (A-IM-MOEA) [11]. In this method, the search procedure
is divided into two phases: exploration and exploitation. In the exploration phase, the reference vector associated with the maximal number of candidate solutions was replaced by a randomly generated reference vector, whereas in the exploitation phase, the reference vector associated with the minimal number of candidate solutions was replaced [12]. Lin and others proposed the enhanced IM-MOEA (E-IM-MOEA) [13], in which the uniformly distributed reference vectors were used in the exploration phase to increase the population diversity, and the distribution information of nondominated solutions was used in the exploitation phase to increase the population diversity. The reference vectors are dynamically regulated for better exploitation. Sindhya and Hakanen proposed the interactive IM-MOEA (I-IM-MOEA) to create solutions in the areas where the decision-maker was interested [14]. Recently, a double learning model-based multi-objective EDA has been proposed [12], which is based on a dynamic mixture of sampling in the decision space by the regularity-based learning model and sampling in the objective space using the inverse learning models [12]. Furthermore, a sequence-based deterministic initialization method is introduced to identify the properties of a fitness landscape. Additionally, an inverse model-based multi-objective EDA using random-forest variable importance methods has been proposed to solve multi-objective optimization problems [8].

3 | PROPOSED ALGORITHM

In this section, the proposed method is described as follows. Section 3.1 provides the problem statement. Section 3.2 illustrates the framework of the proposed algorithm. Section 3.3 describes the generation of reference points.

3.1 | Problem statement

The EDAs (probabilistic model-building genetic algorithms) guides the search through creating and sampling an explicit probabilistic model of possible solutions. Figure 1 shows the framework of IM-MOEA. In each generation of IM-MOEA, parent and offspring (combined population) are partitioned into some subpopulations (partition), and the selection operator is performed on each subpopulation to identify the parents of the subpopulation. The distribution pattern (probabilistic model) of parents ($X^P$ and $Y^P$) is acquired and estimated by the conditional probability distribution of $P(X^P|Y^P)$ through the inverse modeling. To create offspring following this pattern, some samples are produced in the objective space based on the distribution of parents ($Y^o$). Then, using $P(X^o|Y^o)$, $Y^o$ are transformed back to the decision space using Bayes’ theorem:

\[
P(X^o) = \frac{P(X^P|Y^P)P(Y^o)}{P(Y^P|X^P)}
\]

where $P(X^P|Y^P)$ is a priori knowledge [4]. The estimation of an m-input and n-output inverse model $P(X^P|Y^P)$, where m and n are the numbers of objectives and decision variables, respectively, will be time-consuming. This method applies random grouping strategy because all decision variables are independent of one another. Therefore, some decision variables are grouped together, to be derived from the same objective using inverse models. In fact, inverse modeling can be approximated as follows:

\[
P(X|Y) \approx \prod_{i=1}^{N} (P(x_i|f_j) + \epsilon_{ij}),
\]

where $i = 1,...,n$ and $j = 1, 2, ..., m, m > 2$, and it is assumed that $\epsilon_{ij} \sim N(0, (\delta_{ij})^2)$, is a Gaussian noise. Thus, the Gaussian process is used in inverse modeling:

\[
P(x_i|f_j) = N(0, C + (\sigma_n)^2I).
\]

Also, the polynomial mutation operator is applied to grow the population diversity [15].

However, some challenges still remain in the IM-MOEA:

When the PFs are atypical, some partitions may be ineffective. This is because some of the reference vectors that produced related partitions do not contain a sufficient number of rich solutions.
Also, during this process, some of the efficient inverse modes may be removed, or the chances of assigning $x_n$ to $f_m$ disappear, or the optimal assignment of $x_n$ to $f_m$ cannot be specified [8].

Moreover, in IM-MOEA, when the dimensionality of objectives increases, the performance decreases sharply. Also, the samples in the objective space ($Y_o$) are generated only based on the objective values of the current parent population. Generally, the Gaussian distribution has a limited sample space, especially when the standard deviation $\delta$ is small, which would limit an EDA’s exploration ability. Therefore, this method cannot be used to solve many-objective optimization problems.

### 3.2 Framework of proposed algorithm

In this paper, the inverse-model from the MBEAs is described, and a new approach, based on an inverse model with Bayesian linear and random forest regression along with uniformly reference points, is proposed: the RFIEA for many-objective optimization problems. In IM-MOEA, when the objective functions increase, the performance decreases sharply (the conventional multi-objective optimization methods are no longer applicable in many-objective optimization. In fact, the mechanism of maintaining diversity is different on them). Thus, this paper uses the uniform reference points to solve many-objective problems through the inverse model.

In this method, the reference points-based approach produces some points with high performance in diversity and convergence, and these points are used to boost the selection pressure toward the PF while maintaining an extensive and uniform distribution across solutions [16]. The offspring were created by sampling alongside the reference points in a Bayesian linear regression, and a set of uniformly distributed solutions from the PF in the objective space is mapped to PS in the decision space, in which a Gaussian process and random forest regression (random forest variable importance) algorithm [9, 10] is used for mapping rich solutions from PF to the PS.

To summarize, the Gaussian process is used to approximate fitness values, and the random forest features importance method is applied in the inverse model to find the optimal allocation of ($x_n$) to ($f_m$). In fact, random forest feature importance determines the importance of a variable ($x_n$) in the decision space for predicting ($f_m$), and the selected variables are the most important.

The proposed algorithm has the following potential advantages over the existing model-based EA: it is a suitable method based on an inverse model with Bayesian linear and random forest regression, as well as uniform reference points, for solving many-objective optimization problems in MBEAs. It can determine the best assignment of $x_n$ to $f_m$ and the best distributions of $x_n$ to $f_m$ in inverse-modeling. Also, the effective inverse models should not be removed. The proposed algorithm outperforms the compared algorithm in terms of rate of convergence, spread, and uniformity.

Figure 2 depicts the diagram representation of the proposed algorithm.

The framework of RFIEA is illustrated in algorithm 1:

1. The population is randomly initialized. In fact, the initial population consists of observations.
   Then, in each generation of RFIEA:
2. The parent is generated based on the nondominated sorting (the elitist nondominated sorting proposed in NSGA-II) and selection operation. In fact, the individual is sorted into a number of nondominated fronts, a crowding distance is computed on the same front, selection on the current population is performed, and parent population is created.
3. The reference points are generated, as mentioned in Sections 3 and 4. In IM-MOEA, when the PFs are atypical, some partitions may be ineffective. This is because some of the reference vectors that produced related partitions do not have a sufficient number of rich solutions. The reference points are evenly distributed in the decision space. Therefore, the proposed algorithm uses them to build models, and then new expected solutions are sampled based on the built models. Then, an inverse model is applied to the selected parents, and the Gaussian likelihood function and random forest regression are applied to the parent population. Then, a Gaussian process along with random forest regression is performed on the test dataset with reference points using Bayesian linear regression and random forest variable importance, where the test data are offspring in objective space, $Y_{\text{offspring}}(t) \cup R$.

![Figure 2](image-url)
4. New candidate solutions are then reproduced on the decision space, $X^{\text{offspring}}(t)$.

5. The combined population, $(P(t) \cup O(t))$, is then updated for the next generation.

**Algorithm 1** The overall algorithm framework of the proposed method, based on random forest regression

**Initialization:** The population is randomly initialized.

/*main loop*/

while The condition of termination has not accrued do

Create parent population: $P(t)$; (observations)

Reference Points: the reference points are generated from parent population, RP.

Inverse Modeling:
Gaussian likelihood function
random forest regression (variable importance of random forest)
training a Gaussian process: on test set data $(Y^O(t) \cup RP)$;

Reproduction: $O(t)$;

Update the combined population: $P(t + 1) = \bigcup (P(t) \cup O(t))$;

$t = t + 1$;

end while

The IM-MOEA uses the random grouping method in inverse modeling without prior knowledge of the property of problem indivisibility. This algorithm uses random forest. However, since this algorithm is related to many-objective optimization problems and the number of objective functions is large, reference points with $Y^O$ are used in the Gaussian process to avoid eliminating some effective inverse models and losing the chance of randomly allocating decision variables to other objective spaces. Figure 2 depicts the diagram representation of the proposed algorithm.

### 3.3 Reference points

The reference points in the objective space guide evolution and are classified into two groups: ideal points and nadir points. In the ideal points, the objective values are not lower than the best objective values of the given solutions, whereas in the nadir points, the objective values are not higher than the worst objective values of the given solutions. Thus, the individual who is closer to the ideal points and farther away from the nadir points has better convergence. If the objective values are not lower than those of a subset of the known solutions, they are referred to as local ideal points, and if the objective values are not lower than those of all known solutions, they are referred to as global ideal points:

\[
\begin{align*}
  r_{\text{Local}} &= (f_1(x), \ldots, f_m(x) - \epsilon_m, \ldots, f_M(x)), \\
  r_{\text{Global}} &= (f_1(x) - \epsilon_1, \ldots, f_m(x) - \epsilon_m, \ldots, f_M(x) - \epsilon_M),
\end{align*}
\]

where $\epsilon_m$ is either zero or an arbitrarily small positive number. Figure 3 illustrates an example of local and global ideal points: where $r_{\text{Local}}$ is a local ideal point for the subset $\{x_2, x_3, x_4\}$, and $r_{\text{Global}}$ is the global ideal point for all points $\{x_1, x_2, x_3, x_4, x_5\}$.

To generate the reference points, all nondominated individuals are sorted based on crowding distances in each dimensional objective space, and then, the individuals with the greatest crowding distances are selected to generate reference points by reducing the corresponding objective values using the formula. Finally, nondominated reference points with good distribution performance are selected based on the crowding distances in each objective.
the original high-dimensional objective space. Figure 4 illustrates an example of creating reference points using \( \{x_1, x_2, x_3, x_4, x_5, x_6\} \). The first dominated individual is \( x_3 \), so the other five solutions will be used to generate the reference points. Then, \( x_1, x_2, x_4, x_5, x_6 \) are selected for their large crowding distances, and the reference points \( r_2, r_3, r_5, r_8 \) are created. Also, \( r_1, r_4, r_6, r_7 \) are created for \( f_2 \). Then, \( r_3 \) is removed because it is dominated by the other reference points. At the end, the reference points \( r_1, r_2, r_4, r_5, r_6, r_8 \) are selected, assuming they need four reference points.

4 | EXPERIMENTAL ANALYSIS

In this section, the proposed method is evaluated on benchmark tests suits and compared with a series of similar methods, and the experimental results are analyzed using special performance metrics.

4.1 | Benchmark test problems

The proposed method is evaluated on the Walking Fish Group (WFG), containing nine test problems (WFG1 to WFG9) [17]. Table 1 illustrates the features of the test problems.

4.2 | Performance metrics

Quality indicators (QIs) are used to evaluate the quality of solution sets and are classified into six categories [18]: (1) QIs for convergence, (2) QIs for spread, (3) QIs for uniformity, (4) QIs for cardinality, (5) QIs for both spread and uniformity, and (6) QIs for incorporating quality of the four quality aspects.

In experiments, the inverted generational distance (IGD) metric is used to evaluate the performance of EAs in solving MaOPs, which measures the convergence and the spread of the resulted PFs. It is assumed that \( F^\ast \) is a set of uniformly distributed solutions sampled from the PF, and \( P \) marks the approximated solutions created by a given algorithm. Then, the IGD value is estimated by

\[
f_f(x) = \frac{\sum_{f^\ast \in F^\ast} \text{dist}(f^\ast, P)}{|F^\ast|},
\]

where \( |F^\ast| \) is the cardinality of \( F^\ast \) and \( \text{dist}(f^\ast, P) \) marks the shortest Euclidean distance from \( f^\ast \) to the elements in \( P \). The smaller the IGD value, the better the quality of \( P \).

4.3 | Parameter setting

In this paper, QIs of IGD are applied to measure the performance on test instances, and 500 uniformly distributed points are selected from the PF of each test instance to be \( P^\ast \). The number of objectives in experiments is equal to 5, 8, and 10 because of the emphasis on solving maps. According to \( N = C_m^{-1}H_{m-1} \) [19], the population size is equal to 126, 156, and 275, respectively, where \( m \) is the number of objectives and \( H \) is the scale of weight vectors by Das and Dennis’s systematic method. Also, simulated binary crossover (SBX) and polynomial mutation are used as operators if necessary, with SBX having a probability of 0.8 and polynomial mutation having a probability of \( D/1 \), where \( D \) is the number of decision variables. Each test instance consists of 20 independent runs. The parameters of the random forest variable importance method are \( f_m \) and \( X \), where \( f_m \) is the \( m \)th objective function and \( X \) is an individual of the population.

To evaluate the results of experiments involving four compared algorithms, the Wilcoxon rank sum test is used. In this method, a \( + \) labeled illustrates that the proposed algorithm is better than the comparison algorithm, and a \( - \) labeled illustrates that the comparison algorithm is better than the proposed algorithm, and a \( \approx \) labeled illustrates that the proposed algorithm is almost equal to the comparison algorithm.

4.4 | Evaluation and comparison of algorithms

Because the proposed algorithm is related to MBEAs, it is compared with the CSEA and IMMOEA-RF methods,
and because it is related to MOEAs and is tested on many-objective optimization problems, it is compared with the many-objective Evolutionary Optimization Based on Reference Points (RPEA) and MCEDA methods. Therefore, in this section, the performance of the proposed method is compared with IMMOEA-RF [8], in which a Gaussian process and random forest variable importance algorithm is used for mapping nondominated solutions from the PF in the objective space to the PS in the decision space, a CSEA, in which an artificial neural network determines the dominant individuals, a model and clustering-based EDA (MCEDA) [20], which is broken down into several subproblems through the regular model, are presented. Then, to classify the population for each of these subproblems, a clustering method is used, and in each cluster, the EDA is applied to generate the new solutions and the RPEA, in which the reference points-based approach produces some points with high diversity and convergence performance, and these points

<table>
<thead>
<tr>
<th>Problem</th>
<th>M</th>
<th>RFIEA</th>
<th>IMMOEA-RF</th>
<th>RPEA</th>
<th>CSEA</th>
<th>MCEDA</th>
</tr>
</thead>
<tbody>
<tr>
<td>WFG1</td>
<td>5</td>
<td>6.7391e−1</td>
<td>2.8551e+0(+)</td>
<td>7.1343e−1(+)</td>
<td>8.3925e−1(+)</td>
<td>7.5521e−1(+)</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>4.3154e−1</td>
<td>4.5673e+0(+)</td>
<td>4.9196e−1(+)</td>
<td>6.7895e−1(+)</td>
<td>5.2956e−1(+)</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>4.1357e−1</td>
<td>3.1379e+0(+)</td>
<td>4.8558e−1(+)</td>
<td>7.2113e−1(+)</td>
<td>6.9863e−1(+)</td>
</tr>
<tr>
<td>WFG2</td>
<td>5</td>
<td>5.1755e−1</td>
<td>4.9710e+1(+)</td>
<td>4.6389e−1(+)</td>
<td>5.1901e−1(+)</td>
<td>6.1657e−1(+)</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>5.8924e−1</td>
<td>3.2561e+1(+)</td>
<td>4.8949e−1(+)</td>
<td>5.2728e−1(+)</td>
<td>7.9613e−1(+)</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>6.8273e−1</td>
<td>5.1287e+1(+)</td>
<td>4.9760e−1(+)</td>
<td>6.1013e−1(+)</td>
<td>7.5937e−1(+)</td>
</tr>
<tr>
<td>WFG3</td>
<td>5</td>
<td>4.5620e−1</td>
<td>4.5926e+0(+)</td>
<td>3.9830e−1(+)</td>
<td>5.6421e−1(+)</td>
<td>6.7612e−1(+)</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>5.5348e−1</td>
<td>7.1289e+0(+)</td>
<td>4.2392e−1(+)</td>
<td>4.8953e−1(+)</td>
<td>7.2496e−1(+)</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>5.1578e−1</td>
<td>6.8167e+0(+)</td>
<td>2.8124e−1(+)</td>
<td>4.2366e−1(+)</td>
<td>6.9811e−1(+)</td>
</tr>
<tr>
<td>WFG4</td>
<td>5</td>
<td>5.2531e−1</td>
<td>2.4454e+1(+)</td>
<td>6.1167e−1(+)</td>
<td>7.4230e−1(+)</td>
<td>8.5128e−1(+)</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>3.2572e−1</td>
<td>3.8196e+1(+)</td>
<td>3.3235e−1(+)</td>
<td>4.9074e−1(+)</td>
<td>5.8912e−1(+)</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>4.5175e−1</td>
<td>6.8845e+1(+)</td>
<td>4.7638e−1(+)</td>
<td>5.3031e−1(+)</td>
<td>6.6934e−1(+)</td>
</tr>
<tr>
<td>WFG5</td>
<td>5</td>
<td>4.9811e−1</td>
<td>2.6820e+1(+)</td>
<td>5.3437e−1(+)</td>
<td>7.5251e−1(+)</td>
<td>8.6218e−1(+)</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>4.4541e−1</td>
<td>5.1842e+1(+)</td>
<td>5.1967e−1(+)</td>
<td>6.3954e−1(+)</td>
<td>8.2562e−1(+)</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>5.1577e−1</td>
<td>1.9861e+1(+)</td>
<td>6.5582e−1(+)</td>
<td>8.0132e−1(+)</td>
<td>9.3286e−1(+)</td>
</tr>
<tr>
<td>WFG6</td>
<td>5</td>
<td>3.2192e−1</td>
<td>2.5242e+1(+)</td>
<td>4.6226e−1(+)</td>
<td>6.4391e−1(+)</td>
<td>5.1684e−1(+)</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>4.9242e−1</td>
<td>6.1273e+1(+)</td>
<td>5.1943e−1(+)</td>
<td>7.2185e−1(+)</td>
<td>8.5134e−1(+)</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>5.1734e−1</td>
<td>8.1295e+1(+)</td>
<td>5.3761e−1(+)</td>
<td>6.9133e−1(+)</td>
<td>8.9371e−1(+)</td>
</tr>
<tr>
<td>WFG7</td>
<td>5</td>
<td>3.4562e−1</td>
<td>2.3279e+1(+)</td>
<td>4.1332e−1(+)</td>
<td>8.1427e−1(+)</td>
<td>6.1372e−1(+)</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>3.4811e−1</td>
<td>4.1957e+1(+)</td>
<td>4.4392e−1(+)</td>
<td>6.9152e−1(+)</td>
<td>5.1493e−1(+)</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>4.1573e−1</td>
<td>3.9165e+1(+)</td>
<td>6.0124e−1(+)</td>
<td>7.2363e−1(+)</td>
<td>9.8113e−1(+)</td>
</tr>
<tr>
<td>WFG8</td>
<td>5</td>
<td>7.7531e−1</td>
<td>2.1029e+1(+)</td>
<td>6.9426e−1(+)</td>
<td>9.4383e−1(+)</td>
<td>8.0628e−1(+)</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>7.0457e−1</td>
<td>4.8239e+1(+)</td>
<td>3.2325e−1(+)</td>
<td>5.1071e−1(+)</td>
<td>7.5913e−1(+)</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>5.9173e−1</td>
<td>6.1249e+1(+)</td>
<td>5.2633e−1(+)</td>
<td>7.9032e−1(+)</td>
<td>9.1934e−1(+)</td>
</tr>
<tr>
<td>WFG9</td>
<td>5</td>
<td>3.0532e−1</td>
<td>2.9519e+1(+)</td>
<td>3.9261e−1(+)</td>
<td>4.3831e−1(+)</td>
<td>6.1281e−1(+)</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>6.4572e−1</td>
<td>8.0671e+1(+)</td>
<td>7.3251e−1(+)</td>
<td>9.1372e−1(+)</td>
<td>8.5412e−1(+)</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>3.9173e−1</td>
<td>4.9167e+1(+)</td>
<td>4.7632e−1(+)</td>
<td>6.9032e−1(+)</td>
<td>5.1932e−1(+)</td>
</tr>
</tbody>
</table>

FIGURE 5 Statistical results of inverted generational distance (IGD) values obtained by each algorithm on WFG4 on 8 objectives test problems
FIGURE 6  The Pareto optimal solutions and optimization results of (A, D, G, J, and M) 5, (B, E, H, K, and N) 8, and (C, F, I, L, and O) 10 objective test problems acquired by each algorithm on WFG5
are used to strengthen the selection pressure toward the PF while maintaining an extensive and uniform distribution among solutions [16]. Section 4.5 illustrates the experimental results of this group.

### 4.5 Comparisons with MBEAs and MaOEAs

The QIs have been used to analyze the convergence, spread, and cardinality of the proposed algorithm and the quality of solutions. As a consequence, the experimental results were analyzed with IGD and HV metrics. Also, the proposed algorithm is compared with IMMOEA-RF, where this algorithm belongs to the MBEAs and is applied to solve multi-objective optimization problems. CSEA, which belongs to the MBEAs is applied to solve many-objective optimization problems. MCEDA and RPEA are applied to solve many-objective optimization problems.

Table 2 illustrates the IGD result of the proposed algorithm compared to the IMMOEA-RF, CSEA, MCEDA, and RPEA with 5, 8, and 10 objectives.

Table 2 shows that IMMOEA-RF has worse IGD metrics than other compared and proposed methods (RPEA, CSEA, and MCEDA), demonstrating that IMMOEA-RF has worse convergence and spread than other compared and proposed methods. Therefore, it can be argued that the IMMOEA-RF method is not convenient for many-objective optimization problems. Also, it can be seen that the IGD metrics of the proposed algorithm (RFIEA) on test functions of WFG1 to WFG9 are better than those of the RPEA, CSEA, and MCEDA. The RPEA outperforms only in WFG2, WFG3, and WFG8, which might be because of non-separable test problems. However, the WFG9 is non-separable and performs better.

As a result, it can be demonstrated that the convergence and the spread of the proposed algorithm (RFIEA) on test functions of WFG1 to WFG9 are better than that of the other compared methods, and the proposed method outperforms the other compared methods in 5, 8, and 10 objective test problems of WFG1 to WFG9. Figure 5 shows the IGD values of WFG4 for proposed and compared algorithms.

As a result, it can be demonstrated that the proposed algorithm (RFIEA) has greater uniformity on test functions of WFG1 to WFG9 than the other compared methods, and the proposed method outperforms the other compared methods in 5, 8, and 10 objective test problems of WFG1 to WFG9. Figure 6 shows the Pareto optimal solutions and optimization results of 5, 8, and 10 objective test problems acquired by each algorithm on WFG5.

### 5 CONCLUSIONS

In this paper, a new method, based on inverse modeling using Bayesian linear and random forest regression along with uniformly reference points, has been proposed: the RFIEA for many-objective optimization problems. In this method, the reference points-based approach produces some points with high diversity and convergence performance, and these points are used to strengthen the selection pressure toward the PF while maintaining an extensive and uniform distribution across solutions [16]. The offspring were created using a Bayesian linear regression by sampling the reference points and a set of uniformly distributed solutions from the PF in objective
space mapped to PS in decision space, in which a Gaussian process and random forest regression (random forest variable importance) algorithm [9, 10] are applied for mapping rich solutions from PF to the PS. To summarize, in the inverse modeling process, the Gaussian process is used to estimate fitness values, and the random forest feature importance procedure is used to determine the best assignment of decision variables \((x_n)\) to objective functions \((f_m)\).

The proposed algorithm has been compared with the CSEA and IMMOEA-RF methods, which belong to MBEAs, and the RPEA and MCEDA methods, which belong to many-objective EAs. The results show the improvement of diversity and convergence performance (quality indicators). Moreover, because uniform reference points can direct the search space toward the exceptional solution locations, they can employ IM-MaOEA. Also, the Gaussian distribution commonly has a limited sample space, especially when the standard deviation \(\delta\) is small, which would limit an EDA’s exploration capability. The proposed algorithm outperforms other representative MaOEs on various test examples, as long as the shapes of the Pareto set are not too complex. The rate of the uniformity and performance metric of the proposed algorithm are not better than those of the compared algorithms because the Gaussian distribution is more suitable for the exploitation stage due to its short sampling range. Therefore, in future work, other distributions, such as the Cauchy distribution, in addition to the Gaussian distribution, can be used to improve the uniformity performance. The proposed algorithm can be used in the interdisciplinary fields for future research directions.

CONFLICT OF INTEREST
The authors declare that there are no conflicts of interest.

ORCID
Ali Broumandnia https://orcid.org/0000-0001-5145-2013

REFERENCES
Pezhman Gholamnezhad received his BS degree in software engineering from the Shahid Sattari University of Aerospace Sciences and Technology, Tehran, Iran, in 2004, and his MS degree in AI from the Department of Computer Science at Azad University, Science and Research Branch, Tehran, Iran, in 2011, and his PhD degree in AI from the Department of Computer Science at Azad University, South Tehran Branch, Tehran, Iran, in 2020. His main research interests are Machine Learning, Artificial Intelligence, and Data Science.

Ali Broumandnia received the BS degree from Isfahan University of Technology, Tehran, Iran, in 1991, and his MS degree from Iran University of Science and Technology in 1995, both in hardware engineering and PhD degree of computer engineering from Tehran Islamic Azad University-Science and Research Branch in 2006. From 1993 through 1995, he worked on intelligent transportation control with image processing and designed the Automatic License Plate Recognition for Tehran Control Traffic Company. He has published over 120 computer books, journal, and conference papers. He is interested in Persian/Arabic character recognition and segmentation, Persian/Arabic document segmentation, medical imaging, signal and image processing, and wavelet analysis. He is reviewer of some international journals and conferences.

Vahid Seydi is a Research Fellow in the School of Ocean Science at Bangor University in Data Science and Machine Learning. Prior to Bangor, he was an Assistant Professor at the Department of AI at Azad University South Tehran Branch. He received a BS degree in software engineering in 2005, MS in 2007, and PhD in 2014 in AI, from the Department of Computer Science at Azad University, Science and Research Branch, Tehran, Iran. He has been awarded his current research fellowship in 2020, a merit-based scholarship for attending the school of AI, Rome, Italy, in 2019, Also, He has achieved a full scholarship Award from Azad University(2010–2014), KNTU ISLAB Research Fellowship (2007–2010). He secured the first rank among the graduates in the year 2004–2005. His current research focuses on dedicating machine learning methods to analyze data associated with the ocean.

How to cite this article: P. Gholamnezhad, A. Broumandnia, and V. Seydi, A random forest-regression-based inverse-modeling evolutionary algorithm using uniform reference points, ETRI Journal 44 (2022), 805–815. https://doi.org/10.4218/etrij.2021-0245