R2-HMEWO: Hybrid multi-objective evolutionary algorithm based on the Equilibrium Optimizer and Whale Optimization Algorithm

1st Farajollah Tahernezhad-Javazm
Intelligent Systems Research Centre
Ulster University
Derry/Londnderry, UK
tahernezhad_javazm-f@ulster.ac.uk

2nd Debbie Rankin
Intelligent Systems Research Centre
Ulster University
Derry/Londnderry, UK
d.rankin1@ulster.ac.uk

3rd Damien Coyle
Intelligent Systems Research Centre
Ulster University
Derry/Londnderry, UK
dh.coyle@ulster.ac.uk

***Abstract*—Multi-objective evolutionary algorithms can be categorized into three basic groups: domination-based, decomposition-based, and indicator-based algorithms. Hybrid multi-objective evolutionary algorithms, which combine algorithms from these groups, are gaining increased popularity in recent years. This is because hybrid algorithms can compensate for the drawbacks of the basic algorithms by adding different operators and structures that complement each other. This paper introduces a hybrid-multi objective evolutionary algorithm (R2-HMEWO) that applies hybridization in the form of structure and operators. R2-HMEWO is based on the whale optimization algorithm (WOA) and equilibrium optimizer (EO). Elite individuals of WOA and EO are selected from a repository based on the R2-indicator and shifted density estimation-based method. In order to improve solutions' diversity, a reference points method is devised to select next-generation individuals. The proposed multi-objective algorithm is evaluated on 19 benchmark test problems (ZDT, DTLZ, and CEC009) and compared with six state-of-the-art (SOTA) algorithms (NSGA-III, NSGA-II, MOEA/D, MOMBI-II, MOEA/IGD-NS, and dMOPSO). Based on the inverted generational distance (IGD) metric (mean of 25 independent runs), R2-HMEWO outperformed other algorithms on 14 out of 19 test problems and revealed a highly competitive performance on the other test problems. Also, R2-HMEWO performed statistically significant better than MOEA/D and dMOPSO in 15/19 and 14/19 test problems, respectively ($p<0.05$), and reached significant performance in 4 test problems (from ZDT and CEC09) compared to other algorithms.**

***Index Terms*—evolutionary algorithm, multi-objective optimization, whale optimization algorithm, equilibrium optimizer, reference directions, R2 indicator, shifted density estimation**

I. Introduction

Dealing with multi-objective optimization problems (MOPs) has recently come under the spotlight of many researchers. Two well-known approaches of mathematical-based and evolutionary-based/swarm-based optimizations are considered to tackle MOPs. Evolutionary algorithms (EAs) and swarm intelligence (SI) are more versatile in solving MOPs (MOPs with non-differentiable and discontinuous characteristics) than mathematical optimizers. The application of multi-objective evolutionary algorithms (MOEAs) is prevalent in real-world problems [1]–[6], and they have received promising results in recent years. MOEAs can be classified into three categories: domination-based, decomposition-based, and indicator-based, and each category has advantages and disadvantages.

The domination-based method is one of the most popular methods in MOEAs. Non-dominated sorting genetic algorithm (NSGA-II) [7] and NSGA-III [8] are two famous algorithms that apply a non-dominated sorting operator (NSO) and a distribution operator for producing Pareto front (PF) solutions. The main modification in NSGA-III compared to NSGA-II is the distribution operator, which helps the algorithm reach well-distributed solutions, specifically in many-objective problems (MaOPs). The crowding distance operator selects the distributed solutions in NSGA-II, whereas NSGA-III improves solutions' diversity based on a reference point method [9]. The main drawback of domination-based methods is poor convergence, particularly in dealing with MaOPs.

On the other hand, decomposition-based methods utilize the concept of a scalarization function to divide MOPs into several single-objective problems. The first algorithm using this concept is a multi-objective evolutionary algorithm based on decomposition (MOEA/D) [10]. Decomposition is performed by the scalarization (aggregation) functions, of which weighted sum (WS), Tchebycheff (TCH), penalty-based intersection (PBI), and more recently, achievement scalarization function (ASF) are the most popular of these. Aggregation functions ease the process of optimization by not utilizing the concept of Pareto dominance. However, on the downside, choosing a proper scalarization function that meets the problem's characteristics adds another parameter to the problem. In decomposition methods, weight vectors should be defined to improve the diversity of solutions. Also, each individual is updated considering the number of its neighbors. Therefore, defining the weight vector and the appropriate number of neighbors increases the complexity of this method.

Indicator-based methods evaluate individuals during the run by measuring convergence and diversity with indicators such as hypervolume (HV) [11], R2 [12], e^+ [13], Δ_p [14], and IGD [15]. Unfortunately, the primary indicator-based algorithms, such as the indicator-based evolutionary algorithm

(IBEA) [13], suffer from high computational cost. This issue is explicitly exacerbated when these algorithms are confronted with MaOPs.

Finding an impeccable algorithm is currently not possible, and each method has its upside and downside. Combining different algorithms, known as hybridization, can present a solution to this problem. In hybridization, algorithms can be modified by other complementary structures/operators. However, every algorithm/method is not compatible to be combined, and sometimes solutions are not converged. Operator hybridization [16]–[18] and method/structure hybridization [19]–[22] are common among EA publications. Generally, structure/method hybridization is common among MOEAs, and operator hybridization is common among single-objective EAs.

This paper applies hybridization in two stages of structure hybridization and operator hybridization. We propose a hybrid MOEA based on the two relatively new algorithms of whale optimization (WOA) and equilibrium optimizer (EO) (operator hybridization). These two algorithms complement each other and accelerate the convergence speed. On the other hand, a reference points method and a shifted density estimation-based method are added to improve the diversity of solutions. Also, an R2 indicator is incorporated into the main structure for converting the final algorithm into a multi-objective algorithm. To accelerate the convergence speed, an elite archive is also designed in which the best WOA and EO solutions are selected among all archive's individuals. Finally, our algorithm (R2-HMEWO) is compared with six well-known MOEAs (NSGA-III, NSGA-II, MOEA/D, MOMBI-II, MOEA/IGD-NS, and dMOPSO) on nineteen test problems (from ZDT, DTLZ, and CEC09). The results reveal the SOTA performance of our algorithm compared with the other algorithms.

The remainder of the paper is organized as follows: rudiments of MOEA, EO, and WOA are described in Section II. Section III introduces the structure of R2-HMEWO. Experimental settings are explained in Section IV. Finally, results and conclusions are provided in Section V and Section VI, respectively.

II. Backgrounds

A. Multi-Objective Optimization Problem - MOP

A multi-objective problem of $\mathbf{F}(\mathbf{x})$ with the "m" objective functions of $(f_1(\mathbf{x}), ..., f_m(\mathbf{x}))$ is shown in Equation. 1 :

$$\text{minimize } \mathbf{F}(\mathbf{x}) = (f_1(\mathbf{x}), ..., f_m(\mathbf{x}))^T \qquad (1)$$
$$\forall \mathbf{x} \in \Psi \; (\Psi^n \rightarrow \Omega^m)$$

where $\mathbf{x} = (x_1, x_2, ..., x_n)$ is the n-dimensional vector (where n is the number of decision variables), and Ψ^n and Ω^m are the search/decision space and objective space, respectively. An MOEA finds non-dominated solutions, which represent the optimal trade-off between objectives. Based on the domination definition, arbitrary vector $\mathbf{b} = (b_1, b_2, ..., b_m)^T$ is dominated by vector $\mathbf{a} = (a_1, a_2, ..., a_m)^T$ iff $\forall i \in 1, 2, ..., m$, $a_i \leq b_i$ and $\mathbf{a} \neq \mathbf{b}$ and it can be presented as $\mathbf{a} \prec \mathbf{b}$. The Pareto front (PF) is the set of objective vectors of the non-dominated solutions among the variable space and can be shown as: PF = $\{\mathbf{F}(\mathbf{x}) | \mathbf{x} \in \text{PS}\}$. PS is Pareto optimal solutions and the projection of PS in objective space makes PF (PS $= \{\mathbf{x} \in \Psi \mid \nexists \mathbf{y} \in \Psi, \mathbf{F}(\mathbf{y}) \prec \mathbf{F}(\mathbf{x})\}$).

B. Reference Point Method

The main problem associated with NSGA-II is the low diversity of solutions, especially where there are a high number of objectives. NSGA-III solved this issue by replacing the crowding distance operator [7] by adding pre-defined and well-spread points on the normalized hyperplanes in the objective space. Das and Dennis's method [9] is the most popular method for generating these points, but the curse of dimensionality is the main deficiency of this method. Therefore, other methods such as Riesz s-energy [23] are suggested trying to overcome the drawback of Das and Dennis' approach. In NSGA-III, each normalized non-dominated individual is associated with a reference point at the end of each generation. The association criterion is the nearest orthogonal distance between an individual and other reference points. Finally, a niching operator is applied, and the population for the next generation is selected [8].

C. R2 Indicator Method

R2 is a unary indicator that evaluates convergence and diversity [24]. The R family indicator, including R1, R2, and R3, assesses two arbitrary sets' relative quality [25]. Compared to other popular indicators such as HV, the main advantage of this indicator is its low computational cost. It is classified in the indicator categories that need both scalarization function and reference point, and scalarization functions greatly impact on its performance [25]. For example, based on Equation. 2, the R2 indicator for assessing the quality of set ($\mathcal{W}$) with the TCH scalarization function, distributed weight vector of $\Phi = \{\phi_1, \phi_2, ..., \phi_m\}$, and reference point of $\mathbf{z}^* = \{z_1^*, z_2^*, ..., z_m^*\}^T$ is obtained as follows:

$$R2(\mathcal{W}, \Phi, \mathbf{z}^*) = \frac{1}{|\Phi|} \sum_{\phi \in \Phi} \underset{x \in \mathcal{W}}{min} \Big\{ \underset{i=1,2,...,m}{max} \frac{|f_i(\mathbf{x}) - {z_i}^*|}{\phi_i} \Big\} \quad (2)$$

D. Whale Optimization Algorithm - WOA

WOA [26] is a swarm and nature-based optimization algorithm mimicking the hunting approach of humpback whales. This approach is known as the bubble-net strategy, based on the two main sections of encircling prey and bubble-net attacking. Humpback whales try to trap fish by locating themselves below a swarm of fish and guiding them to the surface in a spiral trajectory whilst releasing bubbles (Figure 1).

The whale's new positions are updated based on the two methods: 1) searching and encircling for prey and 2) bubble net attacking. In the first method, to increase exploration (searching for prey), new positions are obtained using Equation. 3.

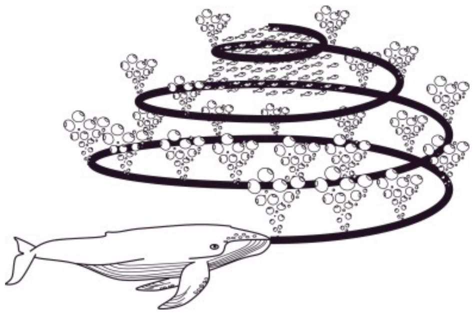

Fig. 1. The bubble-net hunting strategy of the humpback whale [26]

$$\begin{aligned}\overline{W}_{t+1} &= \overline{W}_{rand} - \overline{A}\cdot\overline{D} \\ \overline{D} &= \left|\overline{C}\cdot\overline{W}_{rand} - \overline{W}(t)\right| \\ \overline{A} &= 2\overline{a}\cdot\overline{r} - \overline{a} \\ \overline{C} &= 2\cdot\overline{r}\end{aligned} \tag{3}$$

where $\overline{W}(t)$ reveals the whale position in the current iteration. $\overline{W}_{rand}$ is a random whale. $|\;|$, and $\cdot$ indicate the absolute value and pairwise multiplication, respectively. Also, $\overline{r}$ is a random vector in the $[0,1]$ interval, and a is a parameter, decreasing from 2 to 0. In order to enhance exploitation (regarding the encircling for prey method), the best solution, reached until the current iteration ($\overline{W}^*(t)$), should be chosen (Equation. 4).

$$\begin{aligned}\overline{W}_{t+1} &= \overline{W}^*(t) - \overline{A}\cdot\overline{D} \\ \overline{D} &= \left|\overline{C}\cdot\overline{W}^*(t) - \overline{W}(t)\right|\end{aligned} \tag{4}$$

Bubble net attacking also includes two phases consisting of the shrinking encircling method and the spiral updating position. These two phases are formulated in Equation. 5, which tries to simulate shrinking and spiral-shaped whale trajectories for chasing the fish.

$$\overline{W}_{t+1} = \begin{cases} \overline{W}^* - \overline{A}.\overline{D} & \text{if} \quad p < 0.5 \\ e^{bk}.\cos(2\pi k).\overline{B} + \overline{W}^*(t) & \text{if} \quad p \geq 0.5 \end{cases} \tag{5}$$

where p and k are random numbers located in the $[0,1]$ and $[-1,1]$ intervals. b is a constant parameter specifying the logarithmic spiral shape, and W^* is the best solution, reached until t (current) iteration.

E. Equilibrium Optimizer - EO

Based on the control volume mass models, Faramarzi et al. [27] proposed the EO, a physics-inspired algorithm. The algorithm simulates the mass conservation equation in volume control and tries to find equilibrium states. EO has four main steps, and at the first step, particles' concentrations (solutions' positions) are generated randomly using Equation. 6:

$$\begin{aligned}P_i^{initial} &= P_{min} + rand_i(P_{max} - P_{min}) \\ i &= 1, 2, ..., n\end{aligned} \tag{6}$$

where n is the number of particles. $P^{initial}$ is the initial position (concentration) and P_{min} and P_{max} are the maximum and minimum dimension's values, respectively. Also, $rand_i$ is a random number in the $[0,1]$ interval. The ultimate goal is to find the equilibrium state, known as the global optimum. However, as the global optimum is not specified, four particles (this number can be changed based on the problem) that are the four best particles among all particles in each generation are selected during the process. These four particles plus another particle (the average position of the four best particles) enhance the algorithm's exploration and exploitation, respectively, and build the equilibrium pool as shown in Equation. 7.

$$\overline{P}_{eq-pool} = \left\{\overline{P}_{eq1}, \overline{P}_{eq2}, \overline{P}_{eq3}, \overline{P}_{eq4}, \overline{P}_{eq-avg}\right\} \tag{7}$$

Particles are updated based on the random pool's members in each generation. Particles' positions are also updated regarding the exponential term of $\overline{F}$ in Equation. 8.

$$\begin{aligned}\overline{E} &= e^{-\overline{\kappa}(t-t_0)} \\ t_0 &= \left(1 - \frac{it}{it_{max}}\right)^{\left(a_2\frac{it}{it_{max}}\right)}\end{aligned} \tag{8}$$

where $\overline{\kappa}$ is a random vector in the $[0,1]$ interval, a_2 is constant, controlling exploitation, and it and it_{max} are the current and maximum algorithm's iteration, respectively. In addition, t_0 (Equation. 9) tries to manage exploration ability using the constant a_1. Higher values of a_1 lead to more exploration and lower exploitation and vice versa.

$$t_0 = \frac{1}{\overline{\kappa}}\ln\left(-a_1 sign\left(\overline{r} - 0.5\right)\left[1 - e^{-\overline{\kappa}t}\right]\right) + t \tag{9}$$

where r is a random vector in the $[0,1]$ interval. Another parameter that influences particles' positions (the most influential) and controls exploitation ability is the generation rate ($\overline{gr}$) shown in Equation. 10.

$$\begin{aligned}\overline{gr} &= \overline{gr}_0 e^{-\overline{\kappa}(t-t_0)} = \overline{gr}_0\overline{E} \\ \overline{gr}_0 &= \overline{gcp}\left(\overline{P}_{eq} - \overline{\kappa}\overline{P}\right) \\ \overline{gcp} &= \begin{cases} 0.5r_1 & \text{if} \quad r_2 \geq gp \\ 0 & \text{if} \quad r_2 < gp \end{cases}\end{aligned} \tag{10}$$

where gcp is known as the control parameter of generation, and r_1 and r_2 are random numbers in the $[0,1]$ interval. gp is known as the generation parameter, defining the amount of exploration and exploitation in each generation ($gp : 0.5$ is suggested for a proper exploration and exploitation). Overall, a particle in EO is updated based on Equation. 11 by considering all the update equations.

$$\overline{P} = \overline{P}_{eq} + \left(\overline{P} - \overline{P}_{eq}\right).\overline{E} + \frac{\overline{gr}}{\overline{\kappa}V}\left(1 - \overline{E}\right) \tag{11}$$

where V is a unit and $\overline{P}_{eq}$, $\overline{E}$, $\overline{gr}$, and $\overline{gr}$ are obtained based on the Equation. 7-Equation. 10.

III. Proposed Algorithm Framework

In this section, we explain our algorithm in detail. R2-HMEWO has applied the synergy of the two algorithms of EO and WOA. EO and WOA are two relatively new algorithms, demonstrating promising performance in recent works. Although algorithms' combination does not always result in a better mix, EO-WOA receives good results on benchmarks regarding convergence and diversity criteria. Structures similarities can be enumerated as one of the reasons for this good performance. To convert this combination to an MOEA, and since EO and WOA are elite-based algorithms, an elite archive is also applied in R2-HMEWO. The archive's members are updated in each generation by considering two criteria of convergence and diversity. Regarding the convergence metric, all members are ranked based on the R2 indicator. As achievement scalarization function (ASF) [28] received increasing attention in recent years, it is devised in the R2 indicator framework. An ASF for arbitrary set $\mathcal{S}$ is described in Equation. 12:

$$ASF(\mathcal{S}|w, z^*) = \max_{1\leq i\leq m} \left\{ \frac{|f_i(\mathbf{x}) - z_i^*|}{w_i} \right\} \tag{12}$$

where m is the number of objectives. $\mathbf{z}^* = (z_1^*, z_2^*, ..., z_m^*)$ is a reference vector and w ($w_i \geq 0$ for $1 \leq i \leq m$ and $\sum_{i=1}^{m} w_i = 1$) is the weight vector. In addition, we apply a method based on shifted density estimation (SDE) [29] to increase the chance of archive members selection with the proper distribution. This method discriminates archive individuals with an equal R2 rank using their normalized diversity distance [30]. The normalized diversity distance of the ith Individual (d) in archive ($\mathcal{AR}$) is calculated as shown in Equation. 13.

$$\begin{aligned} C(d_i, \mathcal{AR}) &= \frac{\text{SDE}(d_i) - \text{SDE}_{min}}{\text{SDE}_{max} - \text{SDE}_{min}} \\ \text{SDE}_{max} &= \max\{\text{SDE}(d)|d \in \mathcal{AR}\} \\ \text{SDE}_{min} &= \min\{\text{SDE}(d)|d \in \mathcal{AR}\} \end{aligned} \tag{13}$$

where $SDE(d_i)$ is the shifted density estimation of individual d_i, and obtained using Equation. 14.

$$\text{SDE}(d_i) = \min_{d_j\in\mathcal{AR}, j\neq i} \sqrt{\sum_{l=1}^{m} \text{sde}\left(f_l(d_i), f_l(d_j)\right)^2} \tag{14}$$

$$\text{sde}\left(f_l(d_i), f_l(d_j)\right) = \begin{cases} f_l(d_j) - f_l(d_i) & f_l(d_j) > f_l(d_i) \\ 0 & f_l(d_j) \leq f_l(d_i) \end{cases}$$

where $f(\cdot)$ is the objective function. The sorting structure of archive individuals is shown in Algorithm. 1, and the sorting priority (in ascending order) is based on the R2 rank and SDE-based output, respectively.

The main structure of R2-HMEWO is presented as a flowchart in Figure 2. At the first step, N random individuals with the same number of reference points (based on the Riesz s-Energy method) are generated. Prior to the first iteration, a NSO is applied to the initial population, and non-dominated individuals are ranked regarding the R2 indicator and normalized diversity distance, respectively. Finally, all the ranked individuals are reserved in the archive. At the next stage, the algorithm enters into the repetitive process of generating individuals, and this process will be continued until it meets the termination criterion. WOA and EO operators are applied in parallel to the population in each generation. Elite individuals are selected from 10% of high-ranked archive members regarding the WOA algorithm. The four best particles in EO (in Equation. 7) are selected based on the 40% of high-ranked members in the pool. So, for instance, $\overline{P}_{eq1}$ is chosen from the 10% of high-ranked particles, $\overline{P}_{eq2}$ is chosen from 10%-20%, and $\overline{P}_{eq3}$ and $\overline{P}_{eq4}$ are determined from 20%-30% and 30%-40% of high-ranked pool members, respectively. Next, NSO separates non-dominated individuals from the concatenated population of WOA and EO. Non-dominated individuals are sorted and ranked by ranking operators and preserved in the archive. Finally, normalization, association, and niche-preservation operators separate N individuals from the concatenated population of WOA and EO. This process continues until the end of optimization.

Algorithm 1: Archive ranking structure

input : Archive Members ($\mathcal{AR}$), and weight vector ($\mathcal{U}$)

output: Ranked Archive's members (Sorted$\mathcal{AR}$)

for $\overline{u} \in \mathcal{U}$ **do**
    for $d \in \mathcal{AR}$ **do**
        $d.Sval \leftarrow ASF(d.\overline{Obj}|0, \overline{u})$
        # $d.Sval$: scalarization value
        #$d.\overline{Obj}$: objectives
        $d.C$ = Compute normalize diversity distance value using Equation. 13 & Equation. 14
    Sorted$\mathcal{AR}$ = Sort($\mathcal{AR}, d.Sval^{1st}, \frac{1}{1+d.C}^{2nd}$)
    #$1st$: first priority #$2nd$: second priority

IV. Experimental Settings

For assessing the R2-HMEWO, three test functions of ZDT (ZDT1-ZDT4 and ZDT6), DTLZ (DTLZ1-DTLZ4), and CEC09 (UF1-UF10) are considered, including problems with two and three objectives. The characteristics of each test problem are shown in Table. I [31]. R2-HMEWO is compared with two domination-based algorithms (NSGA-II and NSGA-III) [7], [8], a decomposition based algorithm (MOEA/D) [10], R2, and IGD indicator-based algorithms (MOMBI-II and MOEA/IGD-NS) [32], [33], and finally, a hybrid swarm-based algorithm based on the decomposition method and particle swarm optimization (dMOPSO) [34].

IGD is applied to evaluate the convergence and diversity of the non-dominated solutions. For a reference point of $\mathbf{z}^* =$

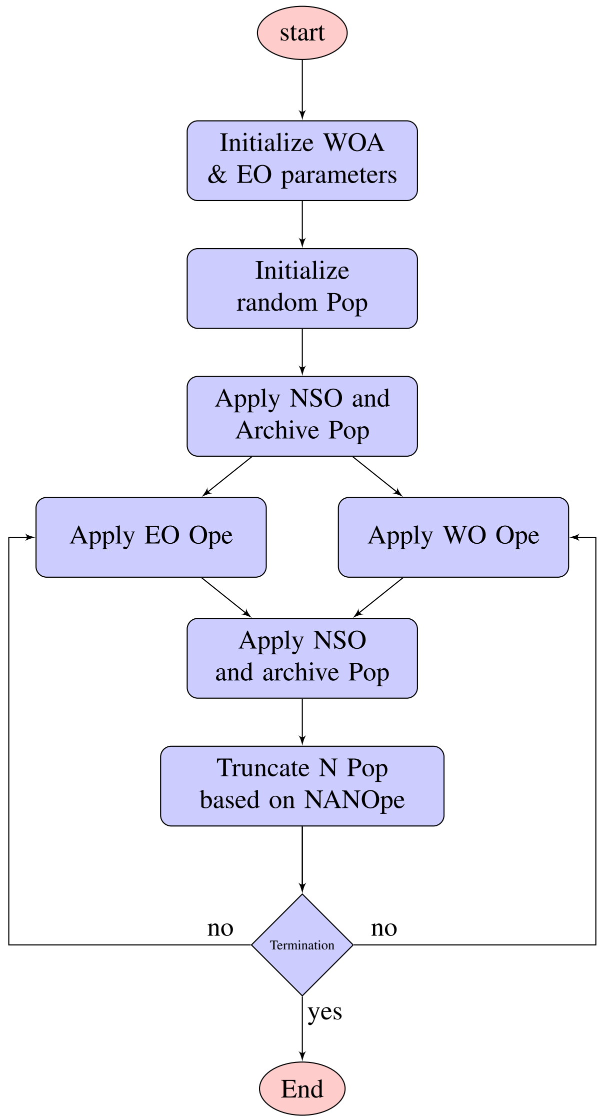

Fig. 2. R2-HMEWO flow chart (Pop: population, Ope: operator, NANOpe: normalization, association, and niching operator)

$(z_1^*, z_2^*, ..., z_m^*)$, IGD computes the average distance of each point in set $\mathcal{S}$ to the reference points (Equation. 15).

$$IGD(\mathcal{S}, z^*) = \frac{\sum_{x \in z^*} ED(x, \mathcal{S})^b}{|\mathcal{S}|} \tag{15}$$

$$ED(s, z^*) = \min_{z \in z^*} \sqrt{\sum_{i=1}^{m} (s_i - z_i)^2}$$

where $ED(.)$ is the Euclidean distance between $s \in \mathcal{S}$ with a nearest point in z^* and b is an arbitrary number ($b > 0$). Each algorithm is run 25 times independently to overcome the problem of randomness. In all the benchmark problems, 100 is considered as the total number of individuals. The number of function evaluations (NFE) is chosen as the termination criterion. NFEs are defined based on recent articles [35]–[38]. Regarding the two-objective test problems of ZDT and CEC09 (UF1-UF7), the maximum NFE is selected at 10k and 60k, respectively. Also, in the three-objective test problems of CEC09 (UF8-UF10) and DTLZ, the termination criterion is set at 100k NFE. In R2-HMEWO the repository is known as the hall of fame in which its members represent the best solution for each generation. To give a chance to the archive to produce better individuals and also prevent the algorithm from trapping in local optimum, simulated binary crossover and polynomial mutation [39] are applied on the archive. These operators cause a disturbance in the repository in each 10% of the total Max NFE. R2-HMTLBO is implemented in Python. The other algorithms being compared are implemented in Matlab (source code available in [40])

TABLE I
ZDT, UF, AND DTLZ CHARACTERISTICS

Name	Obj[a]/Dim[b]	Char[c]
UF1	2/30	Concave PF, Complex PS
UF2	2/30	Concave PF, Complex PS
UF3	2/30	Concave PF, Complex PS
UF4	2/30	Convex PF, Complex PS
UF5	2/30	Discrete PF, Complex PS
UF6	2/30	Discrete PF, Complex PS
UF7	2/30	Complex PS
UF8	3/30	Concave and Parabolic PF, Complex PS
UF9	3/30	Discrete and Planar PF, Complex PS
UF10	3/30	Concave and Parabolic PF
ZDT1	2/30	Convex PF
ZDT2	2/30	Concave PF
ZDT3	2/30	Discrete PS and PF
ZDT4	2/30	Multifrontal PF
ZDT6	2/30	Concave PF
DTLZ1	3/7	Linear and Multimodal PF
DTLZ2	3/12	Concave and Unimodal PF
DTLZ3	3/12	Concave and Multimodal PF
DTLZ4	3/12	Concave and Biased PF

[a] Objectives [b] Dimensions [c] Characteristics.

V. EXPERIMENTAL RESULTS AND DISCUSSION

The average and standard deviation of all 25 runs for each algorithm are presented in Table. II, in which the best results are in bold based on the minimum acquired mean. Also, to demonstrate the algorithms' significant differences, the Tukey HSD test with a 5% significance level is applied. Algorithms with better, worse, and equal performance compared with R2-HMEWO are shown $+$, $-$, and $\approx$ respectively in Table. II.

R2-HMEWO was significantly better than all algorithms when applied to the ZDT4, UF3, UF9, and UF10 test problems. Also, it is shown that R2-HMEWO had significant results' similarities with NSGA-III, NSGA-II, MOEA/IGD-NS, and MOMBI-III, respectively. Regarding the IGD mean, R2-HMEWO had an outstanding performance in bi-objective ZDT and CEC09 benchmark functions. R2-HMEWO outperformed all algorithms in bi-objective ZDT and CEC09 except ZDT6 and UF5 test functions. dMOPSO and MOEA/IGD-NS had a smaller IGD mean in ZDT6 and UF5, respectively. In three objective test problems, although our algorithm performed well in UF8-UF10, in DTLZ tests, it was not successful in reaching the minimum IGD mean. MOEA/IGD-NS performed better in DTLZ test problems, except in DTLZ4, in which R2-HMEWO had the smallest IGD mean. Overall, our algorithm can deal appropriately with discrete problems. However, based on the IGD mean, R2-HMEWO is unsuccessful in dealing with three-objective multi-modal problems such as DTLZ1 and DTLZ3. This deficiency is eclipsed by its good performance in parabolic test problems (UF8 and UF10).

VI. Conclusion and future works

In this work, a hybrid multi-objective evolutionary algorithm is proposed, applying operators from equilibrium optimizer and whale optimization algorithm. Furthermore, to convert the algorithm for multi-objective problems and improve the algorithm's convergence, an external archive is devised for selecting elite individuals. To meet convergence and diversity criteria, the archive selection priority is based on the R2 indicator and a shifted density estimation-based method, respectively. A reference point-based method is used to truncate excessive individuals in each generation and this method also enhances the population diversity. Our method, R2-HMEWO, was compared with five state-of-the-art multi-objective EAs, which include reference point-based (NSGA-III), decomposition-based (MOEA/D), indicator-based (MOMBI-II and MOEA/IGD-NS), and a hybrid swarm-based algorithm (dMOPSO). All algorithms were evaluated on three well-known benchmarks with 19 test functions of ZDT (ZDT1-ZDT4 and ZDT6), CEC09 (UF1-UF10), and DTLZ (DTLZ1-DTLZ4). Each algorithm was executed 25 times independently, and the IGD mean was applied as the comparison criterion. Except in DTLZ test functions, R2-HMEWO performed considerably well on ZDT and CEC09 test functions. Although R2-HMEWO obtained competitive results for three objective test problems, the performance is more notable in two objective problems. R2-HMEWO achieved good performance in discreet and parabolic problems but did not performance as well in multi-modal problems. In future work, we will investigate the incorporation of more efficient operators and structures. For instance, MOEA/IGD-NS revealed promising results in this paper. Therefore, it is worthwhile to search for applying better indicators in elite-based MOEAs. Due to increasing interest in solving problems with more than three objectives, it is also necessary to investigate the ability of hybrid algorithms to solve many-objective problems. Also, evaluating the algorithm performance in dealing with different performance indicators and constrained problems is one of our goals for future research.

Acknowledgment

We are grateful for access to the Tier 2 High Performance Computing resources provided by the Northern Ireland High-Performance Computing (NI-HPC) facility funded by the UK Engineering and Physical Sciences Research Council (EPSRC), Grant Nos. EP/T022175/ and EP/W03204X/1. DC is grateful for the UKRI Turing AI Fellowship 2021-2025 funded by the EPSRC (grant number EP/V025724/1).

Farajollah Tahernezhad-Javazm is supported by a Dr George Moore Endowment PhD scholarship in Data Analytics.

References

[1] A. Zhou, B.-Y. Qu, H. Li, S.-Z. Zhao, P. N. Suganthan, and Q. Zhang, "Multiobjective evolutionary algorithms: A survey of the state of the art," *Swarm and evolutionary computation*, vol. 1, no. 1, pp. 32–49, 2011.

[2] S. Shamshirband, B. Khoshnevisan, M. Yousefi, E. Bolandnazar, N. B. Anuar, A. W. A. Wahab, and S. U. R. Khan, "A multi-objective evolutionary algorithm for energy management of agricultural systems—a case study in iran," *Renewable and Sustainable Energy Reviews*, vol. 44, pp. 457–465, 2015.

[3] F. Tahernezhad-Javazm, V. Azimirad, and M. Shoaran, "A review and experimental study on the application of classifiers and evolutionary algorithms in EEG-based brain–machine interface systems," *Journal of neural engineering*, vol. 15, no. 2, p. 021007, 2018.

[4] G. Ismayilov and H. R. Topcuoglu, "Neural network based multi-objective evolutionary algorithm for dynamic workflow scheduling in cloud computing," *uture Generation computer systems, volume=102, pages=307–322, year=2020, publisher=Elsevier*.

[5] P. P. Biswas, P. Suganthan, and G. A. Amaratunga, "Decomposition based multi-objective evolutionary algorithm for windfarm layout optimization," *Renewable energy*, vol. 115, pp. 326–337, 2018.

[6] M. Abouhawwash and A. M. Alessio, "Multi-objective evolutionary algorithm for pet image reconstruction: Concept," *IEEE Transactions on Medical Imaging*, vol. 40, no. 8, pp. 2142–2151, 2021.

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

[8] 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 Transactions on Evolutionary Computation*, vol. 18, no. 4, pp. 577–601, 2013.

[9] I. Das and J. E. Dennis, "Normal-boundary intersection: A new method for generating the pareto surface in nonlinear multicriteria optimization problems," *SIAM journal on optimization*, vol. 8, no. 3, pp. 631–657, 1998.

[10] Q. Zhang and H. Li, "MOEA/D: A multiobjective evolutionary algorithm based on decomposition," *IEEE Transactions on Evolutionary Computation*, vol. 11, no. 6, pp. 712–731, 2007.

[11] M. Fleischer, "The measure of Pareto optima applications to multi-objective metaheuristics," in *International Conference on Evolutionary Multi-Criterion Optimization*, pp. 519–533, Springer, 2003.

[12] D. Brockhoff, T. Wagner, and H. Trautmann, "On the properties of the R2 indicator," in *Proceedings of the 14th annual conference on Genetic and evolutionary computation*, pp. 465–472, 2012.

[13] E. Zitzler and S. Künzli, "Indicator-based selection in multiobjective search," in *International conference on parallel problem solving from nature*, pp. 832–842, Springer, 2004.

[14] O. Schutze, X. Esquivel, A. Lara, and C. A. C. Coello, "Using the averaged hausdorff distance as a performance measure in evolutionary multiobjective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 16, no. 4, pp. 504–522, 2012.

[15] C. A. C. Coello and M. R. Sierra, "A study of the parallelization of a coevolutionary multi-objective evolutionary algorithm," in *Mexican international conference on artificial intelligence*, pp. 688–697, Springer, 2004.

[16] N. Zeng, D. Song, H. Li, Y. You, Y. Liu, and F. E. Alsaadi, "A competitive mechanism integrated multi-objective whale optimization algorithm with differential evolution," *Neurocomputing*, vol. 432, pp. 170–182, 2021.

[17] H. Zhang, J. Zhou, Y. Zhang, Y. Lu, and Y. Wang, "Culture belief based multi-objective hybrid differential evolutionary algorithm in short term hydrothermal scheduling," *Energy conversion and management*, vol. 65, pp. 173–184, 2013.

[18] A. Korashy, S. Kamel, F. Jurado, and A.-R. Youssef, "Hybrid whale optimization algorithm and grey wolf optimizer algorithm for optimal coordination of direction overcurrent relays," *Electric Power Components and Systems*, vol. 47, no. 6-7, pp. 644–658, 2019.

TABLE II
ZDT, UF, AND DTLZ TEST PROBLEM RESULTS BASED ON THE IGD METRIC

Test Problem	R2-HMEWO	NSGA-III	MOEAD	MOMBI-II	MOEA/IGD-NS	**dMOPSO**	NSGA-II
ZDT1	**0.00367 ±0.00012**	0.02092 ±0.00982 ≈	0.15408 ±0.05703 −	0.05947 ±0.04653 −	0.01856 ±0.00444 ≈	0.0431 ±0.00012 −	0.01267 ±0.00211 ≈
ZDT2	**0.00351 ±2.02E-04**	0.03746 ±0.05724 ≈	0.51880 ±0.07509 −	0.07758 ±0.10234 −	0.05822 ±0.07111 ≈	0.04759 ±0.10836 ≈	0.02835 ±0.02587 ≈
ZDT3	**0.00300 ±0.00013**	0.01856 ±0.00929 ≈	0.1549 ±0.07811 −	0.01497 ±0.01105 ≈	0.01722 ±0.00754 ≈	0.02773 ±0.00507 ≈	0.01273 ±0.00809 ≈
ZDT4	**0.00352 ±0.00008**	0.60783 ±0.30003 −	0.55397 ±0.14692 −	0.29632 ±0.20431 −	0.49190 ±0.20269 −	0.61876 ±0.14084 −	0.21078 ±0.12459 −
ZDT6	0.20737 ±0.00012	0.18673 ±0.07310 ≈	0.07779 ±0.03147 +	0.05056 ±0.02068 +	0.21770 ±0.10124 ≈	**0.00389 ±0.00145** +	0.06609 ±0.02485 +
UF1	**0.07955 ±0.01305**	0.11159 ±0.01999 ≈	0.25970 ±0.11664 −	0.10606 ±0.02796 ≈	0.11853 ±0.02811 ≈	0.40519 ±0.08848 −	0.11254 ±0.03763 ≈
UF2	**0.02858 ±0.00201**	0.04286 ±0.00500 ≈	0.15654 ±0.06880 −	0.05256 ±0.01724 ≈	0.04938 ±0.01250 ≈	0.09286 ±0.00789 −	0.04681 ±0.0176 ≈
UF3	**0.08795 ±0.01990**	0.24786 ±0.06446 −	0.32757 ±0.02534 −	0.26489 ±0.05578 −	0.25401 ±0.05818 −	0.31710 ±0.00861 −	0.20346 ±0.05394 −
UF4	**0.04770 ±0.00104**	0.04995 ±0.00258 ≈	0.07984 ±0.00355 −	0.04841 ±0.00278 ≈	0.05008 ±0.00190 ≈	0.11601 ±0.00667 −	0.04913 ±0.00135 ≈
UF5	0.29620 ±0.13689	0.29750 ±0.08837 ≈	0.53242 ±0.12086 −	0.29679 ±0.08107 ≈	**0.29039 ±0.09159** ≈	2.37592 ±0.42169 −	0.33893 ±0.10010 ≈
UF6	**0.11122 ±0.01301**	0.13461 ±0.07950 ≈	0.44397 ±0.14235 −	0.15628 ±0.09807 ≈	0.168908 ±0.08743 ≈	1.280816 ±0.334508 −	0.14569 ±0.0667 ≈
UF7	**0.04229 ±0.00359**	0.14322 ±0.13490 ≈	0.37025 ±0.20401 −	0.14081 ±0.14079 ≈	0.15175 ±0.14681 −	0.31136 ±0.08038 −	0.17905 ±0.14926 −
UF8	**0.17618 ±0.07699**	0.50848 ±0.07959 −	0.39997 ±0.28667 −	0.44995 ±0.11483 −	0.25117 ±0.00207 ≈	0.27953 ±0.01450 ≈	0.2867 ±0.06313 ≈
UF9	**0.07949 ±0.00166**	0.31043 ±0.09932 −	0.31804 ±0.02568 −	0.38163 ±0.10399 −	0.25193 ±0.09608 −	0.61495 ±0.05559 −	0.36684 ±0.11377 −
UF10	**0.24729 ±0.00066**	0.43472 ±0.10382 −	0.73607 ±0.15308 −	0.49328 ±0.15841 −	0.38353 ±0.10019 −	0.94026 ±0.02735 −	0.52945 ±0.15683 −
DTLZ1	0.03989 ±0.00026	0.02057 ±0.00002 ≈	0.02056 ±0.00001 ≈	0.02060 ±0.00005 ≈	**0.01936 ±0.00010** ≈	2.67260 ±2.64945 −	0.02747 ±0.00107 ≈
DTLZ2	0.05219 ±0.00010	0.05446 ±0.00000 ≈	0.05446 ±0.00000 ≈	0.05450 ±0.00002 ≈	**0.05141 ±0.00046** ≈	0.13639 ±0.00914 −	0.06949 ±0.00274 ≈
DTLZ3	1.40537 ±1.73147	0.05465 ±0.00029 ≈	0.05475 ±0.00035 ≈	0.05481 ±0.00031 ≈	**0.05146 ±0.00078** ≈	6.57983 ±8.00769 −	0.06824 ±0.00293 ≈
DTLZ4	**0.05248 ±0.00014**	0.13259 ±0.18217 ≈	0.15109 ±0.23057 ≈	0.09018 ±0.17827 ≈	0.27627 ±0.32746 −	0.22119 ±0.04843 −	0.10224 ±0.17577 ≈
+/−/≈		0/5/14	1/14/4	1/7/11	0/6/13	1/15/3	1/5/13

[19] Y. Tian, R. Cheng, X. Zhang, F. Cheng, and Y. Jin, "An indicator-based multiobjective evolutionary algorithm with reference point adaptation for better versatility," *IEEE Transactions on Evolutionary Computation*, vol. 22, no. 4, pp. 609–622, 2017.

[20] K. Li, K. Deb, Q. Zhang, and S. Kwong, "An evolutionary many-objective optimization algorithm based on dominance and decomposition," *IEEE Transactions on Evolutionary Computation*, vol. 19, no. 5, pp. 694–716, 2014.

[21] G. Dai, C. Zhou, M. Wang, and X. Li, "Indicator and reference points co-guided evolutionary algorithm for many-objective optimization problems," *Knowledge-Based Systems*, vol. 140, pp. 50–63, 2018.

[22] J. Sun, J. Deng, and Y. Li, "Indicator & crowding distance-based evolutionary algorithm for combined heat and power economic emission dispatch," *Applied Soft Computing*, vol. 90, p. 106158, 2020.

[23] D. Hardin and E. Saff, "Minimal riesz energy point configurations for rectifiable d-dimensional manifolds," *Advances in Mathematics*, vol. 193, no. 1, pp. 174–204, 2005.

[24] J. G. Falcón-Cardona and C. A. C. Coello, "Indicator-based multi-objective evolutionary algorithms: A comprehensive survey," *ACM Computing Surveys (CSUR)*, vol. 53, no. 2, pp. 1–35, 2020.

[25] M. P. Hansen and A. Jaszkiewicz, *Evaluating the quality of approximations to the non-dominated set*. Inst. Math. Modeling, Tech. Univ. Denmark, IMM Tech. Rep. IMM-REP-1998-7, 1998.

[26] S. Mirjalili and A. Lewis, "The whale optimization algorithm," *Advances in engineering software*, vol. 95, pp. 51–67, 2016.

[27] A. Faramarzi, M. Heidarinejad, B. Stephens, and S. Mirjalili, "Equilibrium optimizer: A novel optimization algorithm," *Knowledge-Based Systems*, vol. 191, p. 105190, 2020.

[28] A. P. Wierzbicki, "The use of reference objectives in multiobjective optimization," in *Multiple criteria decision making theory and application*, pp. 468–486, Springer, 1980.

[29] M. Li, S. Yang, and X. Liu, "Shift-based density estimation for pareto-based algorithms in many-objective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 3, pp. 348–365, 2013.

[30] Q. Lin, S. Liu, Q. Zhu, C. Tang, R. Song, J. Chen, C. A. C. Coello, K.-C. Wong, and J. Zhang, "Particle swarm optimization with a balanceable fitness estimation for many-objective optimization problems," *IEEE Transactions on Evolutionary Computation*, vol. 22, no. 1, pp. 32–46, 2016.

[31] W. Yang, L. Chen, Y. Wang, and M. Zhang, "Multi/Many-Objective Particle Swarm Optimization Algorithm Based on Competition Mechanism," *Computational intelligence and neuroscience*, vol. 2020, 2020.

[32] R. Hernández Gómez and C. A. Coello Coello, "Improved metaheuristic based on the R2 indicator for many-objective optimization," in *Proceedings of the 2015 annual conference on genetic and evolutionary computation*, pp. 679–686, 2015.

[33] Y. Tian, X. Zhang, R. Cheng, and Y. Jin, "A multi-objective evolutionary algorithm based on an enhanced inverted generational distance metric," in *2016 IEEE congress on evolutionary computation (CEC)*, pp. 5222–5229, IEEE, 2016.

[34] S. Zapotecas Martínez and C. A. Coello Coello, "A multi-objective particle swarm optimizer based on decomposition," in *Proceedings of*

the 13th annual conference on Genetic and evolutionary computation, pp. 69–76, 2011.

[35] L. Wei, X. Li, and R. Fan, "A new multi-objective particle swarm optimisation algorithm based on R2 indicator selection mechanism," *International Journal of Systems Science*, vol. 50, no. 10, pp. 1920–1932, 2019.

[36] L.-X. Wei, X. Li, R. Fan, H. Sun, and Z.-Y. Hu, "A hybrid multiobjective particle swarm optimization algorithm based on R2 indicator," *IEEE Access*, vol. 6, pp. 14710–14721, 2018.

[37] H. Yu, Y. Wang, and S. Xiao, "Multi-objective particle swarm optimization based on cooperative hybrid strategy," *Applied Intelligence*, vol. 50, no. 1, pp. 256–269, 2020.

[38] Q. Lin, J. Li, Z. Du, J. Chen, and Z. Ming, "A novel multi-objective particle swarm optimization with multiple search strategies," *European Journal of Operational Research*, vol. 247, no. 3, pp. 732–744, 2015.

[39] K. Deb, R. B. Agarwal, *et al.*, "Simulated binary crossover for continuous search space," *Complex systems*, vol. 9, no. 2, pp. 115–148, 1995.

[40] Y. Tian, R. Cheng, X. Zhang, and Y. Jin, "PlatEMO: A MATLAB platform for evolutionary multi-objective optimization [educational forum]," *IEEE Transactions on Evolutionary Computation*, vol. 12, no. 4, pp. 73–87, 2017.