A Localized High Fidelity Dominance based Many-Objective Evolutionary Algorithm

Dhish Kumar Saxena, Sukrit Mittal, Sarang Kapoor, and Kalyanmoy Deb

COIN Report Number 2021002

Abstract—**Over the last decade or more, the development of Many-objective Evolutionary Algorithms (MaOEAs) capable of dealing with four or more objectives simultaneously has been of dominant interest for the researchers. The associated challenges relate to - the inability of the Pareto-dominance concept to induce selection pressure for convergence, representation of a large-dimensional Pareto front with a limited set of points, and difficulty in visualization and decision making. This has led researchers to recognize the utility of the use of reference vectors to ensure uniformity/diversity in a large-dimensional space even with a reasonably limited number of points. In effect, decomposition-based MaOEAs and even their hybrids which conjunctly rely on Pareto-dominance for convergence, have demonstrated a lot of promise. This paper proposes a novel hybrid MaOEA, namely, HFiDEA, where in diversity is ensured through the use of reference vectors, while convergence in pursued through the newly proposed *localized high fidelity dominance* definition. The latter, referred to as *lhf-dominance*, marks the first attempt to overcome the limitations of Pareto-dominance by factoring - the number of objectives in which a solution is better or worse, than the other; the degree by which a solution is better or worse in the objectives, than the other; and the scope for incorporating decision maker's preferences between objectives. Besides these fundamental contributions, this paper also addresses *on-the-fly* timing for nadir point estimation and self-termination of HFiDEA. While the former positively impacts diversity maintenance, the latter is critically important when the Pareto-front is not known a priori. The efficacy of HFiDEA is demonstrated through its comparison with other MaOEAs, over a wide range of test problems.**

Index Terms—**Many-objective, Localized High Fidelity Dominance, Nadir point update, Self-termination.**

I. Introduction

Multi-objective optimization problems (MOPs) refer to those problems where two or more conflicting objectives are to be simultaneously optimized. An MOP [1] with M objectives in n variables can be given by:

$$\begin{array}{ll} \text{Minimize} & F(X) = (f_1(X), \ldots, f_m(X)) \\ \text{subject to} & X \equiv \{x_1, \ldots, x_n\}^T \in \Omega \end{array} \tag{1}$$

Notably, for each solution $X \in \Omega \subseteq \mathbb{R}^n$, there exists an $F(X) \equiv \{f_1(X), \ldots, f_M(X)\} \in \mathbb{R}^m$. Furthermore:

- given two solutions X and $Y \in \Omega$, X is said to *dominate* Y if X is not worse than Y in any objective, and X is better than Y in at least one objective, implying: (i) $f_m(X) \leq f_m(Y)\ \forall m = 1, \ldots, M$, and (ii) $\exists\ m \in \{1, \ldots, M\} : f_m(X) < f_m(Y)$.
- a solution $X^* \in \Omega$ is called Pareto-optimal if there is no $X \in \Omega$ that dominates X^*. The set of all the Pareto-optimal solutions is called the Pareto Set (PS), and its corresponding representation in the objective space constitutes the *efficient set* or loosely known as the Pareto Front, given by $PF = \{F(X) \in \mathbb{R}^M | X \in PS\}$.

The goal of Multi-objective Evolutionary Algorithms (MOEAs) is to evolve a finite set of random solutions over several iterations, to a set of solutions that approximates well the true PF for a given problem in terms of convergence (proximity of the true PF), and diversity (coverage across the PF with a reasonably uniform distribution) [1]. Broadly, the prominent MOEAs could be classified into: (i) Pareto-Dominance-based (including NSGA-II [2], SPEA2 [3]), (ii) indicator-based (including IBEA [4]), and (iii) decomposition-based methods (including MOEA/D [5]). The efficacy of these MOEAs is well established for two- and three-objective problems. However, problems with more than three objectives, typically referred to as *Many-objective* problems (MaOPs), pose challenges relating to *poor search efficiency; need for a population size that increases exponentially with M, unknown impact of recombination parameters, difficulty in visualization and decision making* [6]–[9]. While these challenges are common to all MOEAs, the mode and intensity of their manifestation, and remedial approaches vary depending on the MOEA category, as highlighted below.

1) Dominance-based MOEAs: An increase in M causes a higher proportion of solutions to become non-dominated [10]. Given this (a) the primary selection criterion of Pareto-dominance ceases to be effective, and results in a diminished selection pressure for convergence (referred

D.K. Saxena and S. Mittal are with the Department of Mechanical and Industrial Engineering, Indian Institute of Technology Roorkee, Uttarakhand-247667, India, e-mail: {dhish.saxena, smittal1}@me.iitr.ac.in.

S. Kapoor is with the Department of Computer Science and Engineering, Indian Institute of Technology Roorkee, Uttarakhand-247667, India, e-mail: skapoor@cs.iitr.ac.in.

K. Deb is with the Beacon NSF Center and the Department of Electrical and Computer Engineering, Michigan State University, East Lansing, MI, 48824, USA, e-mail: kdeb@egr.msu.edu.

as *dominance resistance* in [8]), and (b) the density-based secondary selection criterion ends up favoring the remote and boundary solutions (referred as *active diversity promotion* in [8]). The need to induce stronger selection pressure for convergence, has led to the development of relaxed-dominance based methods, including, ϵ-dominance [11], grid-dominance [12], preference order ranking [13], k-optimality [14], L-optimality [15], and knee-point-bias [16]. Efforts have also been made to counter the adverse impact of active diversity promotion by: controlled activation of crowding distance [17]; assignment of high density to poorly converged solutions [18]; use of a substitute assignment distance measure [19]. Yet, the challenge remains on how to set the new parameters involved.

2) Indicator-based MOEAs: Such MOEAs, in principle, pose a promising alternative since they do not rely on Pareto-dominance. This perhaps explains the rationale for development of hypervolume (HV) based MaOEAs (including SMS-EMOA [20] and HypE [21]), and R2-indicator based MaOEAs (including Mombi [22]). However, their utility is impaired owing to different factors. In that, the running times of HV based MaOEAs increase exponentially with M, while the other indicator metrics are not monotonic with Pareto-dominance, and some need a priori knowledge of the PF.

3) Decomposition-based MOEAs: Such MOEAs, led by MOEA/D, rely on generating reference vectors (RVs), and decomposing a given problem in to a set of subproblems (one per RV), each of which could be optimized using aggregation functions, in collaboration with neighboring subproblems. This architecture does away with the challenge of dominance resistance, and also provides the scope for approximating an M-dimensional PF with a reasonably sized solution set. However, the balance between convergence and diversity is strongly dependent on the choice of aggregation function [12], and the relation between the neighborhood size for local mating and local replacement and the performance [23]. The quest to address such challenges has led to several variants of MOEA/D towards tackling MaOPs, including, the L_p family of scalarizing methods [24], and two independent distance measures to balance convergence and diversity [25]. Lately, another variant, namely, MOEA/D-LWS [26] has been proposed, which exploits the high search efficiency of the weighted-sum approach while aiming to find the most converged solution in the neighbourhood of each reference-vector. Interestingly, MOEA/D-LWS has shown superior/comparable performance on MaOPs, compared to MOEA/D; its variants in MOEA/D-AS [27] and MOEA/D-SS [28]; and several other algorithms including PICEA-g [29], HypE [21], θ-DEA [30], and (iv) SPEA2+SDE [31].

Besides the above, hybrid methods have been developed which rely on integration of:

1) Dominance and decomposition-based approaches, for instance, NSGA-III [32], [33] is based on the use of Pareto dominance for convergence and a set of supplied reference points (decomposition principle) for diversity. Its variants with *on-the-fly* reference point relocation strategy [33], and replacement of Pareto-dominance with θ-dominance [34] also fall in this category besides [35], [36].
2) Dominance and Indicator based approaches, for instance, an MOEA whose non-dominated solutions are divided in to two archives, one for convergence (Indicator-based selection) and another for diversity (Pareto-based selection) has been proposed in [37].

This paper proposes a hybrid MaOEA, namely, HFiDEA, relying on the integrated use of RVs for diversity preservation, and the newly proposed *localized high fidelity dominance* definition to induce enhanced selection pressure for convergence. The latter marks the first attempt to overcome the limitations of Pareto-dominance by simultaneously factoring - the number of objectives in which a solution is better or worse, and the degree by which a solution is better or worse; and the scope for incorporating decision maker's preferences between objectives. In addition, this paper also addresses *on-the-fly* timing for Nadir point estimation which positively influences diversity maintenance, and self-termination of HFiDEA by tracking the stability of solutions. The efficacy of HFiDEA is demonstrated on a wide range of *unconstrained*[1] test problems.

The remaining paper is organized as follows: The motivation for this paper is presented in section II, where the novel *localized high fidelity dominance* has also been introduced. The proposed MaOEA, namely HFiDEA, is presented in Section III. Its management of the convergence-diversity tradeoff is discussed in Section IV. The test suite, experimental settings, results, and aspects such as computational complexity are presented in Section V. The paper concludes with Section VI.

II. Motivation

This paper finds its inspiration in the promise offered by conjunct use of the dominance and decomposition principles, towards tackling MaOPs. While NSGA-III offers an instance of how this could be realized, this paper aims to *overcome the limitations that the underlying notion of Pareto-dominance suffers from (in many-objective context)*, and *eliminate the use of the non-domination ranking* which is known to turn ineffective in inducing sufficient selection pressure for convergence. To this effect, HFiDEA is being proposed here, where in diversity is ensured by use of the RVs, and convergence is pursued by use of the newly proposed *localized high fidelity dominance*, referred to as *lhf-dominance*.

[1]To cater to constrained MaOPs–the HFiDEA proposed here is being adapted/generalized, and is to be presented in a sequel to this paper.

When dealing with MaOPs, the existing concept of Pareto-optimality and Pareto-dominance are inefficient in modeling and simulating human decision making. While comparing any two solutions, the following three aspects which are crucial in human decision making process and from a practical perspective, are not taken into consideration [38], including: (a) the number of objectives in which a solution is better or worse, than the other [38], (b) the degree by which a solution is better or worse in the objectives, than the other, and (c) the decision maker's preferences between objectives (if any). Accounting for these factors may lead to several degrees of dominance and consequently, several different concepts of *optimality* can be introduced [39]. In this spirit, a novel *lhf-dominance* is being proposed here, in that:

- the term *localized* is to assert that its scope is *local* - restricted to the natural cluster of solutions formed around *each* reference vector (detailed in Section III).
- the term *high fidelity* is to assert that deeper human decision making criteria are being employed to distinguish two solutions which may otherwise be adjudged as incomparable by Pareto-dominance
- given two (Pareto) non-dominated solutions, namely X and Y, let n_b^X and let n_b^Y denote the number of objectives in which X is better than Y, and vice-versa, respectively. Then, X is said to *lhf-dominate* Y if both the following conditions are met:
  1) X is better in more or equal number of objectives than Y, i.e., $n_b^X \geq n_b^Y$.
  2) The weighted gain in objectives in which X is better, exceeds the weighted loss in objectives in which it is worse than Y. For a minimization problem it implies: $\Delta F_{X,Y} = \sum_{m=1}^{M} \frac{f_m^X - f_m^Y}{w_m} < 0$, where w_m are the components of a user-defined weight-vector or the nearest reference-vector.
- X Pareto-dominates Y implies that X *lhf-dominates* Y. This holds, since the former implies the following:
  1) $n_b^X > 0;\ n_b^Y = 0 \rightarrow n_b^X > n_b^Y$
  2) $f_m^X \leq f_m^Y \xrightarrow{for\ w_m \geq 0\ \forall\ m=1,\ldots,M} \sum_{m=1}^{M} \frac{f_m^X - f_m^Y}{w_m} < 0$

  The fact that both the conditions for *lhf-dominance* are met, implies that the stated property holds.

The mode in which the notion of *lhf-dominance* is utilized for developing a hybrid MOEA for tackling MaOPs is evident in Section III, while some of its salient features with regard to the management of convergence-diversity tradeoff are highlighted in Section IV.

III. Proposed Algorithm: HFiDEA

The *localized high fidelity dominance* based EA, namely, HFiDEA, being proposed here, relies on random initialization of population; crossover; mutation; and environmental selection aided by the use of RVs and the newly proposed *lhf-dominance*. In any generation, the implementation of environmental selection, where the goal is to select the best N solutions from the $2N$ sized combined parent and child populations, entails the following. First the RV closest to each solution is identified, implying that eventually some RVs may get associated with a cluster of solutions, while others may remain unassociated. For each RV associated with a cluster, one cluster representative is selected based on *lhf-dominance*. For the unassociated RVs, the solutions closest to each of them are selected as their representatives, respectively. However, while doing so, the solutions from the clusters that have already been selected as RV representatives are exempted. Another notable aspect of HFiDEA relates to normalization of the objective space which impacts the distribution of the RVs and eventually the diversity of the population. Normalization requires estimation of Nadir point, and unlike most other RV based EAs where the Nadir point estimation and normalization is done in every iteration, HFiDEA utilizes authors' previous work on population *stabilization* tracking [40] for intermittent estimation/update of Nadir point. This change is inspired by the revelation that it leads to faster and better PF approximation. While intermittent Nadir point update is an integral part of HFiDEA, the choice on whether or not to use the *stabilization* tracking for HFiDEA's termination is left to the user.

Algorithm 1: Generation t of HFiDEA procedure

```
Input: M, N, P_t, Z, Z^I = φ, Z^N = φ, ψ^N, ψ^TM, UF
       and t_TM.
Output: P_{t+1}
begin
    P_mating ← Random Selection (P_t)
    Q_t ← Crossover + Mutation (P_mating)
    R_t ← P_t ⋃ Q_t
    Z_k^I ← min(Z_k^I, min_{x∈R_t} f_k(x)) ∀ k ∈ [1, M]
    F ← All objective vectors in R_t
    if Z^N = φ then
        F ← F − Z^I
    else
        F ← Normalized F using Z^I & Z^N
    end
    P_{t+1} ← Environment Selection(R_t, F, N, Z)
    C^N ← Check Stabilization(P_t, P_{t+1}, Z, ψ^N)
    if C^N is True then
        Z^N ← Update Nadir(P_{t+1}, Z^I)
    if UF = 1 then
        C^TM ← Check Stabilization(P_t, P_{t+1}, Z, ψ^TM)
    if C^TM is True or t_TM = t then
        Terminate at generation t_TM = t
end
```

In the above background, the implementation of any generation t of HFiDEA, is summarized in Algorithm 1.

The inputs to HFiDEA include: the MOP/MaOP definition including the number of objectives M; population size N; randomly initialized population P_t; reference-vector set Z; crossover and mutation parameters; Ideal Z^I and Nadir Z^N points initialized as null sets; the parameter set influencing the timing of nadir-point estimation (ψ^N); and the user-flag UF (Boolean) on termination. In that, $UF = 0$ implies that HFiDEA will terminate at user pre-specified t_{TM} generations, while $UF = 1$ activates the inbuilt *stabilization* tracking algorithm, leading to *on-the-fly* assessment of t_{TM} and HFiDEA's termination there-at. The latter also necessitates as input the parameter set guiding self-termination, namely, ψ^{TM}.

The procedure starts by creating a mating pool P_{mating} by subjecting P_t members to random selection; creating N offspring solutions Q_t using the crossover and mutation operators; and updating the ideal point Z^I using the merged parent and offspring population, namely, R_t (lines 2-5). Notably, unlike the Z^I update which is done in every generation, the nadir point Z^N updates are done intermittently, corresponding to only those generations where-at the inbuilt *stabilization* tracking algorithm under mild parameter settings in ψ^N recommends it (lines 13-15). Here, its important to note that until the first update of Z^N is executed, the R_t is not normalized. Instead, the objective vectors in R_t are simply translated in a manner that Z^I becomes the origin for R_t (line 7-9). The rationale is that during the early generations, the populations evolve rapidly and the extreme points fluctuate significantly. It implies that the Nadir-point estimates are likely to be too unstable, and if employed for normalization of R_t, the diversity may be adversely impacted [40]. This aspect has also been discussed in [41] and two different normalization schemes have been proposed. In that, the use of sigmoidal function to control the extent of normalization, has been shown to outperform the alternative. However, its use necessitates the knowledge of t_{TM} a priori, knowing which could be challenging in real-world problems or even variants of test problems under different parameter settings. To avoid such dependence, this paper in line with [40] adopts a step-function where no normalization is performed till the population has mildly stabilized. Subsequently, when the Nadir point estimates could be relied on, R_t is normalized using current Z^I and the last updated Z^N (line 10). In contrast to Z^N, Z^I is likely to remain relatively stable, considering that it happens to be the best point found till the current generation. This justifies its usage for translation, across all the generations.

The above procedure is repeated until t_{TM} generations are completed (line 19). In that, t_{TM} may be pre-specified by the user or may be assessed *on-the-fly* by the inbuilt *stabilization* tracking algorithm under strict settings in ψ^{TM} (depending on the user's choice for UF).

Notably, the core of HFiDEA lies in its two main constituents, namely, environmental selection and *stabilization* tracking algorithm which under mild and strict parameter settings triggers Z^N update and HFiDEA's termination, respectively. Hence, the following sections elaborate on these main constituents.

A. Environment Selection

Here, the aim is to select the N best solutions (P_{t+1}) from the existing $2N$ solutions (R_t), the procedure for which is presented in Algorithm 2. The input towards this selection includes the $2N$ objective vectors F of the solutions in R_t, and the N reference vectors. At a particular generation t, the Fs may comprise of translated or normalized values depending on whether until t, the first Z^N update has been triggered or not. Regardless of this fact, the selection criterion for all ts is consistently guided by *lhf-dominance*. First, the perpendicular distance ($d^{\perp}$) [32] is computed for each solution in R_t with each RV in Z, and the values are stored in a matrix, namely, $D^{\perp}$, sized, $[2N \times N]$ (line-3). Each solution in R_t is then associated with its nearest RV (by $d^{\perp}$ computed in previous step, line-4). Next, for each RV $Z_j \in Z$, the solutions associated with Z_j are used to form a cluster $\mathcal{C}$ (line-6). It is naturally possible that some RVs may remain unassociated with any solution, i.e., there may exist some Z_j for which the cluster may be an empty set.

In this background, the task of environmental selection reduces to identifying the best representative solution for each Z_j. The manner in which this is accomplished depends on whether a particular Z_j is associated with a cluster of solutions, or has remained unassociated. For all such Z_j which have an associated cluster $\mathcal{C}$, attention is focused on only the (Pareto) non-dominated solutions in $\mathcal{C}$ (line-7). Of these, the solution nearest (by $d^{\perp}$) to Z_j is tagged as the α-solution (line-8). Furthermore:

- for RVs that coincide with the objective axis vectors, the respective α solutions are chosen as the best RV representatives. This choice is made to allow for a more comprehensive Z^N update, and eventually support as broad the coverage of PF approximation as possible.
- for RVs that do not coincide with the objective axis vectors (line-9), it is evaluated whether any/some of the remaining solutions in $\mathcal{C}$ *lhf-dominate* α (lines 11–12). If no solution meets this test, α is retained as the best representative for the corresponding RV. Alternatively, any/all such solutions which lhf-dominate α are stored in a set g, $g \subset \mathcal{C}$ (line-13). If there is only one solution in g, by default it qualifies as the best representative for the corresponding RV, else, the solution with minimum $d^{\perp}$ value is selected (line-16).

All the solutions selected as the best representatives for the RVs associated with a cluster of solutions (as above), are stored in the *Selected* set S, and simultaneously removed from R_t (line-17). In a scenario, where the

Algorithm 2: Environment Selection(R_t, F, N, Z)

1 **begin**
2 $S \leftarrow \phi$, $Z^L \leftarrow Z$
3 $D^{\perp} : d^{\perp}_{i,j} \leftarrow d^{\perp}(F_i, Z_j)\ \forall\ i \in [1, 2N]\ \&\ j \in [1, N]$
4 Associate each solution R_t with its nearest RV
5 **foreach** $Z_j \in Z$ **do**
6 $\mathcal{C} \leftarrow$ All solutions in neighbourhood of Z_j
7 Remove pareto-dominated solutions in $\mathcal{C}$
8 $\alpha \leftarrow$ Solution in $\mathcal{C}$ with minimum $d^{\perp}$
9 **if** *Z_j is not an axis vector* **then**
10 $g \leftarrow \phi$
11 **for** $\beta \in \mathcal{C}$ **do**
12 **if** *β lhf-dominates α* **then**
13 $g \leftarrow g \bigcup \beta$
14 **end**
15 **if** *count(g) $\geq$ 1* **then**
16 $\alpha \leftarrow$ Solution in g with minimum $d^{\perp}$
17 $S \leftarrow S \bigcup \alpha$; $R_t \leftarrow R_t \setminus \alpha$; $Z^L \leftarrow Z^L \setminus Z_j$
18 **end**
19 $F \leftarrow$ Normalized Objective vectors from R_t
20 **foreach** $z \in Z^L$ **do**
21 $D^{\perp} : d^{\perp}_i \leftarrow d^{\perp}(F_i, z)\ \forall\ F_i \in F$
22 $\alpha \leftarrow$ Solution in R_t with minimum $d^{\perp}$
23 $S \leftarrow S \bigcup \alpha$
24 **end**
25 **return** S
26 **end**

Algorithm 3: Check Stabilization(P, Q, Z, ψ)

1 **begin**
2 $c_1 =$ ***False***, $c_2 =$ ***False***
3 $n_p, n_s \leftarrow \psi$
4 $p_z \leftarrow Associate(P, Z)$, $q_z \leftarrow Associate(Q, Z)$
5 $\mu\mathcal{D}_t \leftarrow 0$; $z^A \leftarrow 0$
6 **foreach** $z \in Z$ **do**
7 **if** *$p_z \neq \phi$ **and** $q_z \neq \phi$* **then**
8 $\bar{p} = mean(p_z)$, $\bar{q} = mean(q_z)$
9 $\mu\mathcal{D}_t \leftarrow \mu\mathcal{D}_t + \mathcal{D}(z)$; $z^A \leftarrow z^A + 1$ Eq 2
10 **else if** *$p_z = \phi$ **or** $q_z = \phi$* **then**
11 $\mu\mathcal{D}_t \leftarrow \mu\mathcal{D}_t + 1.0$; $z^A \leftarrow z^A + 1$
12 **end**
13 **end**
14 $\mu\mathcal{D}_t \leftarrow \mu\mathcal{D}_t / z^A$
15 Compute the mean (μ_t) and standard deviation (σ_t) from $\mu\mathcal{D} = [\mu\mathcal{D}_1, \ldots, \mu\mathcal{D}_t]$
16 $D_t = round(\mu_t, n_p), S_t = round(\sigma_t, n_p)$
17 **if** $[D_t = D_{t-1} = \cdots = D_{t-n_s}]$ **then**
18 $c_1 =$ ***True***
19 **if** $[S_t = S_{t-1} = \cdots = S_{t-n_s}]$ **then**
20 $c_2 =$ ***True***
21 **if** *c_1 is **True and** c_2 is **True*** **then**
22 **return** ***True***
23 **else**
24 **return** ***False***
25 **end**
26 **end**

RVs associated with cluster of solutions are k, the task of environmental selection further reduces to finding, from a pool of $2N - k$ solutions, the best representative solution for each of the unassociated or left-out RVs ($N - k$ in number, denoted as Z^L). Towards it, for each $z \in Z^L$, the nearest solution (by $d^{\perp}$) from R_t is identified, and added to the solution set S (lines 19–23). The environmental selection performed as above, leads to N selected solutions in the set S, one each per RV, and these constitute the next parent population P_{t+1}.

B. Two-pronged Stabilization Tracking

As highlighted earlier, HFiDEA utilizes as its integral constituent, an earlier proposed stabilization tracking algorithm for updating the Nadir point Z^N estimates. HFiDEA also provisions the scope to utilize the same for assessing the appropriate generation for its termination (t_{TM}), *on-the-fly*. If the later option is not to be availed, then t_{TM} may be provided as an input for an HFiDEA run. Regardless of the purpose of its usage, this algorithm, formalized as Algorithm 3 uses the same principles, just that the inputs may vary (discussed later in this section).

The building blocks for *stability* tracking algorithm are as follows. Consider, P and Q as the HFiDEA populations from two successive generations. At first, each solution in P and Q is associated with its closest (by $d^{\perp}$) RV. Let p_z and q_z represent the set of normalized objective vectors of solutions $p \in P$ and $q \in Q$, respectively, associated with any reference-vector $z \in Z$ (line-4). Then, a distance measure $\mathcal{D}(z)$ is proposed, which in principle, computes the normalized distance between the sets of solutions p and q, along z. In that, $\mathcal{D}(z) = 0$ implies that solutions from two successive generations coincide with respect to z (indicating stability). The computation of $\mathcal{D}(z)$ depends on different possible scenarios, as elaborated below.

1) If both p_z and q_z have one or more members each, then first their respective means are computed, namely $\bar{p}$ and $\bar{q}$, and subsequently $\mathcal{D}(z)$ is computed as per Equation 2 (lines 7–9).
2) If p_z has one or more members and q_Z is empty, or vice-versa, then $\mathcal{D}(z)$ is set to 1 (the maximum possible value) to signify that the population has not yet stabilized (lines 10–11).
3) If both p_z and q_z are empty, then the vector z does not contribute to $\mathcal{D}(z)$, in that particular generation.

$$\mathcal{D}(z) = \frac{\left|z^T\bar{p} - z^T\bar{q}\right|}{\max(z^T\bar{p}, z^T\bar{q})}. \tag{2}$$

Once $\mathcal{D}(z)$ is computed for each z, they are averaged over all $z \in Z$ to yield the distance measure for the generation t, namely, $\mu\mathcal{D}_t$ (line-14), where z^A marks the number of active RVs - those for which p_z and q_z are not simultaneously empty. At any generation t, given the availability of $\mu\mathcal{D} = [\mu\mathcal{D}_1, \ldots, \mu\mathcal{D}_t]$, their mean μ_t and standard deviation σ_t are computed, based on which the stability of successive populations is tracked. In that, if both μ_t and σ_t individually conform with their respective values over n_s successive generations, up to n_p decimal places, then the populations are considered to have *stabilized* (lines 16–22).

Clearly, the settings of the input parameter set $\{n_p, n_s\}$ could be adapted in view of the purpose that the above *stability* tracking ought to serve. For instance, a mild setting of $\{n_p, n_s\} \equiv \{2, 20\}$ may suffice to gauge moderate stabilization of populations over successive generations, pointing to an appropriate timing for Z^N update. Intuitively, a stricter setting $\{n_p, n_s\} \equiv \{3, 20\}$ or $\{3, 50\}$ may be more adequate to point to t_{TM}, *on-the-fly*.

Considering that Z^N update is an integral constituent of HFiDEA, its procedure [32] is discussed below. Reference to Algorithm 4 reveals that first the origin is translated to coincide with Z^I. This necessitates that for each solution (indexed i), the j^{th} objective value is reduced by the j^{th} component of Z^I. Next, for each axis vector (indexed k), the minimum ASF [42] solution is identified. Once all the M extremes are determined, an M dimensional linear hyperplane is constructed, whose intercepts (a_j on the j-th objective axis) are computed. Since the origin coincides with Z^I, adding the respective intercepts a_j to Z^I_j would yield Z^N_j, completing the Nadir point update.

Algorithm 4: Update Nadir(P, Z^I)

begin
- $X \leftarrow$ All solutions in P
- Translate Objectives: $f_j(X_i) = f_j(X_i) - Z^I_j$, where $i \in [1, N]$, $j \in [1, M]$
- Find extreme points: $Z^{ext.}_k = X_i : \arg\min_{i \in [1,N]} ASF(F(X_i), w_k)$, where $k \in [1, M]$, $ASF(F(X_i), w_k) = \max_{j=1}^{M} f_j(X_i)/w_{k,j}$, $\{w : w_{k,k} = 1 \ \&\ w_{k,j} = \epsilon \ \forall\ j \neq k;\ j,k \in [1, M]\}$ Compute intercepts a_j for $j = 1, \ldots, M$
- **return** $Z^N \leftarrow a + Z^I$

end

IV. lhf-Dominance Guided Convergence-Diversity Tradeoff: A Prelude to Experimental Results

As evident above, HFiDEA ensures diversity through the use of RVs (decomposition), and convergence through the use of dominance criterion. While this hybrid architecture relying on both decomposition and dominance resembles the one utilized by NSGA-III, there are some significant differences in the manner in which these constituents are implemented in HFiDEA. In that:

- **convergence:** unlike NSGA-III which relies on Pareto-dominance and non-domination ranking of population members, HFiDEA utilizes the newly proposed *lhf-dominance* which unprecedentedly discriminates between the solutions based on - the number of objectives in which they are better/worse; the degree by which they are better/worse in the objectives; and the decision maker preferences between the objectives where these preferences could be altered for different parts of the search space (different preferences across different RVs).
- **diversity:** the performance of RV based MaOEAs is impacted by the efficacy of Z^N update and normalization of the population. Unlike, NSGA-III where the Z^N update and normalization is done in every generation, HFiDEA utilizes authors' previous work [40] for intermittent estimation/update of Nadir point. This change is inspired by the revelation in [40] that intermittently timed Nadir-point estimation leads to faster and better PF approximation owing to more *stable* search.
- **convergence-diversity tradeoff:** within any cluster of (Pareto) non-dominated solutions associated with a particular reference vector, NSGA-III prefers the solution closest to the reference vector (say, α) in any generation, implying that eventually the distribution of solutions in the approximated PF aligns closely with the RVs. However, for the same cluster HFiDEA potentially allows for selection of a solution other than α if it *lhf-dominates* α. In effect, for each solution cluster around a reference vector, HFiDEA gives higher preference to convergence than uniformity/diversity (proximity to the reference vector) while selecting the cluster representative. Hence, NSGA-III and HFiDEA differ in their management of convergence-diversity tradeoff as further discussed in Section IV and reiterated in Section V.

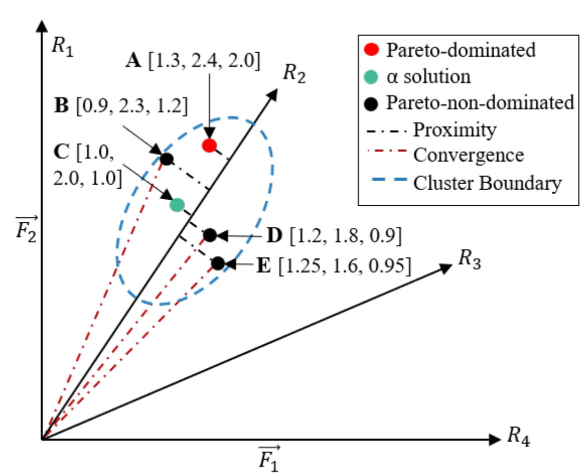


Figure 1: Depicting the convergence-diversity tradeoff management by HFiDEA.

The management of convergence-diversity tradeoff by HFiDEA vis-*à*-vis NSGA-III is explained with reference to Figure 1. In that, the projections of five solutions (A–E) in a three-objective problem are shown on a two-objective sub-space. For a sample illustration, these solutions form a cluster $\mathcal{C}$ associated with the R_2 RV weighted as $[0.3, 0.6, 0.1]$. As the first step, the check for Pareto-dominance finds A to be dominated, hence, only non-dominated solutions B–E qualify for further consideration. Then, the solution nearest to R_2 (by $d^{\perp}$) is identified, here C (α-solution). At this point, NSGA-III would have chosen the C as the cluster representative (or niche). However, HFiDEA invokes the *lhf-dominance* to investigate if any/some of B, D, or E outperform C by being better in more objectives, and also the weighted net gain in the objectives (where the weights coincide with those of R_2). Since $n_b^B < n_b^C$, B does not *lhf-dominate* C. Furthermore, since $n_b^D > n_b^C$, $n_b^E > n_b^C$, $\Delta F_{D,C} = -0.667$, and $\Delta F_{E,C} = -0.333$, both D and E *lhf-dominate* C. Finally, since D is closer to R_2 than E, D is selected as the cluster representative, surviving to the next generation. A more generic consideration of cluster representatives is covered in the supplementary file.

At a higher level, the distinction between NSGA-III and HFiDEA in their management of convergence-diversity tradeoff, could be appreciated, through the following steps and associated emphasis:

1) Pareto-dominance check emphasizing convergence
2) α-solution identification emphasizing diversity
3) *lhf-dominance* check emphasizing convergence
4) identification of the nearest ($d^{\perp}$) *lhf-dominant* solution emphasizing diversity

Notably, NSGA-III manages the convergence-diversity tradeoff by using only the first two steps. In that, inherently, there is superior selection pressure for better uniformity/diversity (as the non-dominated solution nearest to the RV under consideration is niched by default) than convergence, since the limitations of non-domination ranking are well known in the context of MaOPs. In contrast, HFiDEA does away with the use of non-domination ranking, and adds two extra layers of granularity to discriminate between competing solutions. In that, solutions endorsing better convergence from a practical human decision making perspective (n_b and ΔF) are identified, and one among them offering the best uniformity/diversity (minimum $d^{\perp}$ with RV) is finally selected.

One aspect is critical to note with respect to items (3) and (4) above. If for any RV, HFiDEA finds two or more β solutions that *lhf-dominate* α, it implies that each β solution offers better convergence than α. However, to balance the convergence-diversity tradeoff, a judicious choice is made for a β having the minimum $d^{\perp}$. However, if HFiDEA were to greedily evolve in support of only convergence, than a β offering the most negative $\Delta F_{\beta,\alpha}$ would have to be picked - a choice that has been avoided on purpose. To summarize, HFiDEA pursues better convergence without compromising on too much on uniformity/diversity (too much deviation from the RV).

V. Results

In this section, the performance of HFiDEA has been compared with NSGA-III [32] and MOEA/D-LWS [26]. NSGA-III is chosen for comparison to represent the hybrid (decomposition and dominance) EAs, a category that HFiDEA also falls in. Similarly, MOEA/D-LWS has been chosen to represent the decomposition-based EAs, and one which performs at par or better than a wide range of EAs, including MOEA/D, MOEA/D-AS, MOEA/D-SS, PICEA-g, HypE, θ-DEA, and SPEA2+SDE, already demonstrated in [21], [27]–[31].

Notably, unlike HFiDEA and NSGA-III, MOEA/D-LWS uses an offline unbounded external archive (UEA) to store all the non-dominated solutions found from initial to the termination generation, and the corresponding HV is used as the performance measure. We do not find the notion of using the offline archive for performance evaluation of an MaOEA inspiring for a *fundamental* reason besides the aspect of *runtime complexity* and *operational* challenges. In that:

- *fundamentally*, we believe, the requirement of evaluating the efficacy of a newly developed MaOEA needs to be distinguished from the task of solving a real-world problem. In case of the latter, it may be judicious to utilize each and every solution evaluated since the beginning. However, while evaluating the strengths and limitations of a new MaOEA, focus ought to be on how good a PF approximation, sized N, could be obtained by starting with a randomly initialized population of the same fixed size N. An algorithm's ability to recognize which past solutions are potentially useful to create new and better solutions and carrying them over to the next generations with a fixed population size is an intellectually challenging task and should stay as an essential and elegant property of a population-based optimization algorithm. Considering divide-and-conquer strategies with the help of an external archive of a size many times larger than the population size may eventually diminish the need of a population and deprives them from taking advantage of the *implicit parallelism* which provides the population-based algorithms their worth.
- in terms of *computational complexity*, for a given M, the source in [26] cites it as $O(N^2)$. However, for UEA to be updated so it could participate in performance evaluation, the complexity assumes $O(M\tilde{N}^2)$, where, $\tilde{N} = N \times t_{TM}$. Hence, the complexity becomes a quadratic function of number of generations till MaOEA's termination. This perhaps offers a plausible

- reason as to why the experiments in [26] restricted the scope to 7 objectives and $t_{TM} = 250$.
- *operationally* it may not be reasonable to compare different MaOEAs by inducting an offline archive in the end, *if, in principle*, the respective potential for improvement in each, is not equable. The archive approach requires a final selection of N solutions from the archive, a matter which plays an important role in defining the performance of the original algorithm, without directly reflecting its true performance.

Perhaps, recognizing these challenges, [26] endorses the rationale of choosing the nearest solution to each RV from the UEA. This would allow for a PF approximation, sized N (coinciding with the initial population) on which performance metrics could be applied. In this background, two options are availed in this paper:

- adaptation of MOEA/D-LWS: the performance metric is applied to the standalone population, at t_{TM}, sized N (the UEA is not merged) for a fair comparison with NSGA-III and HFiDEA.
- adaptation of NSGA-III and HFiDEA: like MOEA/D-LWS, these are augmented with the UEA; the nearest RV from each solution in the UEA is identified; clusters are formed for each RV; and the respective environmental selection is implemented, leading to a PF approximation sized N.

In this paper, we do not completely neglect the UEA approach, but make two comparisons – with (Tables IV and V) and without (Tables II and III) the UEA approach.

A. Test Problems

In this paper, the following test problems have been used: DTLZ1–DTLZ4 [43] and all WFG [44] except WFG3 (degenerative). In that, their three to 15-objective versions have been used. For the DTLZ problems, the number of variables are kept as $n_{var} = (M-1) + k$, where $k = 20$. Similarly, for WFG problems, the number of variables are $n_{var} = 2 \times (M-1) + k$, where $k = 20$. Notably, the use of $k = 20$ makes these problems quite difficult compared to the generally used $k = 5$.

B. Parameter Settings

Notably, HFiDEA, NSGA-III and MOEA/D-LWS all require a set of reference points, for which the Das and Dennis (DD) method [45] is used. The number of gaps p for DD method and the corresponding population sizes N are highlighted in Table I. Where there are two values of p shown, the first value is used to generate boundary points while the second value is used to generate interior points [32]. The number of reference-vectors chosen is identical to the population size.

For all the experiments across the three MaOEAs, the SBX crossover ($p_c = 0.9$ and $\eta_c = 20$) and polynomial mutation ($p_m = 1/n_{var}$ and $\eta_m = 20$) are used. However, only for DTLZ1 and DTLZ3 problems, $\eta_m = 5$ is used (other parameters remaining the same). This is done, considering that these problems are highly multimodal, and ensuring diversity during the early generations is critical, towards which a lower distribution index for mutation, helps. Apart from these, the other MOEA/D-LWS related parameters include the neighbourhood size (N_N= 10% of N) and the mating restriction probability ($\delta = 0.8$) [26]. Also, the offspring replacement size *nr* for MOEA/D-LWS is not limited, since it reportedly [26] works better for MaOPs.

Table I: Parameter settings for the Das and Dennis method.

M	3	5	8	10	15
N	105	210	156	275	135
p	13	6	3, 2	3, 2	2, 1

The results are presented for two different termination settings, including: (i) the a priori specified $t_{TM} = 500$, and (ii) t_{TM} inferred using the *stability-tracking* algorithm with parameter setting of $\{n_p, n_s\} \equiv \{3, 50\}$. For each of the above categories: (i) Z^N update is performed with parameter setting of $\{n_p, n_s\} \equiv \{2, 20\}$, and (ii) the presented results correspond to 21 runs for each MaOEA, and subjected to a statistical analysis using Wilcoxon rank-sum test.

C. Performance Indicators

This paper employs HV as the primary performance metric, for which the reference point used is $R_{M\times 1} = [\frac{N}{N-1}, \ldots, \frac{N}{N-1}]^T$ [46].

$$\begin{aligned} d^{\|}(F) &= \frac{1}{N}\Sigma_{i=1}^{N}\frac{\left|F(\alpha_i)^T z_{\alpha_i}\right|}{\|z_{\alpha_i}\|}, \\ d^{\perp}(F) &= \frac{1}{N}\Sigma_{i=1}^{N}\left\|F(\alpha_i) - d_i^{\|}\frac{z_{\alpha_i}}{\|z_{\alpha_i}\|}\right\|. \end{aligned} \tag{3}$$

In addition, two metrics $d^{\|}$ and $d^{\perp}$ (as used in Penalty-based Boundary Intersection [5]) are evaluated towards separate measures for convergence and uniformity/diversity, respectively. The only adaptation is that these measures, as in Equation 3, are evaluated for the entire population by summing up their respective values for the representative solution (α_i) of corresponding RV (z_{α_i}). Notably, the objective vector F used here has been normalized using the known PF extremes, for each test problem.

D. Analysis and Trends

To begin with, the performance of NSGA-III, MOEA/D-LWS (without assistance from UEA) and HFiDEA is presented in the Table II, for t_{TM}: (a) set a priori as 500, and (b) as determined for HFiDEA *on-the-fly* by the stability tracking algorithm with $\psi^{TM} = \{n_p, n_s\} = \{3, 50\}$. In case of (b), the results for NSGA-III and MOEA/D-LWS are generated for the same t_{TM} as obtained for

Table II: Performance of NSGA-III, MOEA/D-LWS (no UEA) and HFiDEA, using the mean (μ) and standard deviation (σ) for Hypervolume, over 21 runs each. The results presented as $\mu(\sigma)$ correspond to t_{TM}: (a) set apriori as 500, and (b) determined for HFiDEA on-the-fly by the stability tracking algorithm with $\psi^{TM} = \{n_p, n_s\} = \{3, 50\}$. In case of (b), the t_{TM} obtained for HFiDEA (tabulated below) is used for NSGA-III and MOEA/D-LWS. The symbols "–", "=", or "+" against NSGA-III and MOEA/D-LWS highlight where these are statistically worse than, comparable to or better than HFiDEA, respectively.

Problem	M	Termination at $t_{TM} = 500$			Termination at $t_{TM} \equiv \psi^{TM} = \{n_p, n_s\} = \{3, 50\}$			
		NSGA-III	MOEA/D-LWS	HFiDEA	t_{TM}	NSGA-III	MOEA/D-LWS	HFiDEA
DTLZ1	3	0.998944 (0.0319) =	1.026910 (0.0022) +	0.988666 (0.0395)	588	1.020600 (0.0114) =	1.028520 (0.0007) +	1.019780 (0.0111)
DTLZ1	5	0.992759 (0.0290) =	1.015600 (0.0296) +	0.993639 (0.0334)	723	1.023660 (0.0009) –	1.024150 (0.0000) +	1.024050 (0.0002)
DTLZ1	10	1.008870 (0.0653) –	1.036960 (0.0003) +	1.032360 (0.0132)	736	1.036790 (0.0006) –	1.037100 (0.0000) +	1.037080 (0.0001)
DTLZ1	15	0.826673 (0.3626) –	1.117980 (0.0000) +	1.117820 (0.0004)	822	1.112780 (0.0126) –	1.117980 (0.0000) +	1.117980 (0.0000)
DTLZ2	3	0.447062 (0.0001) +	0.416168 (0.0030) –	0.437760 (0.0013)	495	0.447062 (0.0001) +	0.416168 (0.0030) –	0.437765 (0.0010)
DTLZ2	5	0.716643 (0.0004) +	0.635698 (0.0057) –	0.704219 (0.0019)	583	0.718462 (0.0004) +	0.636591 (0.0072) –	0.706045 (0.0017)
DTLZ2	10	0.933044 (0.0026) –	0.731680 (0.0186) –	0.965990 (0.0011)	657	0.942679 (0.0012) –	0.732268 (0.0139) –	0.967372 (0.0008)
DTLZ2	15	1.052750 (0.0043) –	0.145856 (0.0587) –	1.068480 (0.0094)	627	1.060790 (0.0027) –	0.133175 (0.0353) –	1.073130 (0.0010)
DTLZ3	3	0.576417 (0.3249) =	0.961045 (0.0710) +	0.406738 (0.2804)	577	0.896788 (0.1429) +	1.016770 (0.0138) +	0.745719 (0.2336)
DTLZ3	5	0.164223 (0.2026) +	0.715552 (0.3500) +	0.078360 (0.2294)	658	0.858553 (0.2755) +	1.017310 (0.0219) +	0.618641 (0.3687)
DTLZ3	10	0.114696 (0.2413) –	0.784889 (0.3237) +	0.503582 (0.4457)	706	0.734789 (0.3925) –	1.036970 (0.0004) +	0.834707 (0.3614)
DTLZ3	15	0.004833 (0.0216) –	1.112110 (0.0178) =	0.860741 (0.4340)	722	0.173350 (0.3347) –	1.117980 (0.0000) +	0.976462 (0.3424)
DTLZ4	3	0.404723 (0.1281) –	0.418083 (0.0027) –	0.428066 (0.0453)	876	0.405761 (0.1285) –	0.416513 (0.0032) –	0.428578 (0.0455)
DTLZ4	5	0.719306 (0.0004) +	0.634740 (0.0051) –	0.704309 (0.0016)	863	0.721832 (0.0002) +	0.634782 (0.0038) –	0.705150 (0.0016)
DTLZ4	10	0.952149 (0.0007) –	0.779248 (0.0152) –	0.961627 (0.0010)	791	0.957186 (0.0002) –	0.772922 (0.0180) –	0.962055 (0.0009)
DTLZ4	15	1.076430 (0.0004) +	0.114777 (0.0044) –	1.059770 (0.0096)	687	1.078100 (0.0002) +	0.114928 (0.0043) –	1.058980 (0.0106)
WFG1	3	0.317611 (0.0213) –	0.372395 (0.0521) =	0.383059 (0.0053)	967	0.387141 (0.0101) –	0.448678 (0.0164) +	0.437327 (0.0097)
WFG1	5	0.275125 (0.0168) –	0.432143 (0.0156) +	0.347157 (0.0164)	1126	0.346199 (0.0049) –	0.494750 (0.0191) +	0.415917 (0.0128)
WFG1	10	0.201711 (0.0126) –	0.338757 (0.0332) =	0.336033 (0.0255)	1427	0.288000 (0.0317) –	0.388086 (0.0374) =	0.407037 (0.1138)
WFG1	15	0.243980 (0.0077) –	0.213329 (0.0211) –	0.265519 (0.0491)	1582	0.404965 (0.0281) +	0.286794 (0.0128) –	0.325030 (0.0470)
WFG2	3	0.858936 (0.0731) –	0.881887 (0.0738) =	0.899879 (0.0631)	809	0.864339 (0.0739) –	0.884379 (0.0740) =	0.904581 (0.0641)
WFG2	5	0.976256 (0.0558) =	0.957222 (0.0883) =	0.938986 (0.0952)	948	0.989638 (0.0577) =	0.961480 (0.0901) =	0.943141 (0.0967)
WFG2	10	0.964223 (0.0857) +	0.775687 (0.1540) –	0.916526 (0.0847)	1017	0.974247 (0.0862) –	0.827908 (0.1547) =	0.977989 (0.0858)
WFG2	15	1.002830 (0.0930) +	0.452118 (0.1732) –	0.805264 (0.1057)	1019	1.017930 (0.0963) +	0.499804 (0.1853) –	0.873672 (0.1058)
WFG4	3	0.419985 (0.0029) –	0.419896 (0.0017) –	0.426934 (0.0024)	669	0.426694 (0.0022) –	0.422351 (0.0015) –	0.429894 (0.0021)
WFG4	5	0.634187 (0.0052) –	0.679595 (0.0067) –	0.700382 (0.0022)	795	0.660965 (0.0038) –	0.669019 (0.0072) –	0.709503 (0.0014)
WFG4	10	0.745126 (0.0196) –	0.761154 (0.1866) –	0.895028 (0.0121)	807	0.811236 (0.0150) –	0.826073 (0.1459) –	0.927991 (0.0065)
WFG4	15	0.785680 (0.0210) –	0.140336 (0.0633) –	0.917555 (0.0218)	774	0.878644 (0.0216) –	0.171353 (0.0411) –	0.978849 (0.0139)
WFG5	3	0.387768 (0.0024) –	0.376352 (0.0023) –	0.390869 (0.0015)	639	0.391540 (0.0023) =	0.378236 (0.0027) –	0.392145 (0.0019)
WFG5	5	0.616598 (0.0026) –	0.614232 (0.0069) –	0.655188 (0.0041)	726	0.631940 (0.0026) –	0.602072 (0.0096) –	0.659094 (0.0026)
WFG5	10	0.775600 (0.0056) –	0.821280 (0.0074) –	0.878403 (0.0038)	785	0.814960 (0.0035) –	0.814058 (0.0110) –	0.890071 (0.0023)
WFG5	15	0.828357 (0.0105) –	0.203867 (0.0647) –	0.897989 (0.0176)	753	0.886560 (0.0074) –	0.202966 (0.0542) –	0.951327 (0.0059)
WFG6	3	0.389201 (0.0059) –	0.380980 (0.0084) –	0.394201 (0.0061)	611	0.393438 (0.0057) =	0.380763 (0.0079) –	0.396324 (0.0066)
WFG6	5	0.619087 (0.0093) –	0.603518 (0.0119) –	0.660255 (0.0100)	690	0.633133 (0.0081) –	0.601466 (0.0108) –	0.665926 (0.0092)
WFG6	10	0.805152 (0.0159) –	0.719414 (0.0209) –	0.888103 (0.0205)	734	0.837573 (0.0150) –	0.723404 (0.0245) –	0.896305 (0.0202)
WFG6	15	0.900471 (0.0204) –	0.146991 (0.0566) –	0.961147 (0.0234)	703	0.939184 (0.0148) –	0.154967 (0.0674) –	0.978436 (0.0202)
WFG7	3	0.434254 (0.0011) –	0.422109 (0.0017) –	0.435651 (0.0011)	646	0.438142 (0.0008) +	0.422607 (0.0017) –	0.436492 (0.0011)
WFG7	5	0.681711 (0.0028) –	0.662800 (0.0062) –	0.713054 (0.0013)	735	0.696981 (0.0017) –	0.654153 (0.0059) –	0.715865 (0.0013)
WFG7	10	0.827437 (0.0062) –	0.881032 (0.0066) –	0.953400 (0.0015)	747	0.878718 (0.0033) –	0.871332 (0.0080) –	0.961769 (0.0010)
WFG7	15	0.921953 (0.0165) –	0.221715 (0.1177) –	0.994021 (0.0188)	706	0.981661 (0.0105) –	0.200136 (0.1226) –	1.040280 (0.0116)
WFG8	3	0.343937 (0.0024) +	0.318901 (0.0036) –	0.333206 (0.0029)	612	0.348410 (0.0023) +	0.320584 (0.0039) –	0.336320 (0.0024)
WFG8	5	0.533688 (0.0049) –	0.551750 (0.0064) =	0.554216 (0.0111)	666	0.549585 (0.0035) –	0.556014 (0.0056) –	0.564756 (0.0118)
WFG8	10	0.663097 (0.0104) +	0.683677 (0.0067) +	0.595736 (0.0792)	730	0.716990 (0.0132) +	0.688342 (0.0073) +	0.630823 (0.0828)
WFG8	15	0.732501 (0.0424) +	0.089775 (0.0596) –	0.457578 (0.0662)	693	0.795928 (0.0226) +	0.092349 (0.0581) –	0.502575 (0.0833)
WFG9	3	0.359819 (0.0210) =	0.330193 (0.0015) –	0.374304 (0.0287)	588	0.361099 (0.0215) =	0.330334 (0.0018) –	0.375411 (0.0293)
WFG9	5	0.545913 (0.0052) –	0.551171 (0.0053) –	0.621529 (0.0201)	645	0.550041 (0.0047) –	0.553531 (0.0044) –	0.630744 (0.0179)
WFG9	10	0.642468 (0.0146) –	0.598902 (0.0333) –	0.714022 (0.0765)	704	0.665271 (0.0172) –	0.610613 (0.0334) –	0.742805 (0.0675)
WFG9	15	0.638410 (0.0324) +	0.153646 (0.0394) –	0.548876 (0.1205)	651	0.669560 (0.0316) +	0.157284 (0.0388) –	0.584677 (0.1277)
# –/=/+ →		**32/05/11**	**33/06/09**	**of 48 problems**		**30/05/13**	**33/04/11**	**of 48 problems**

HFiDEA. These results capture the mean (μ) and standard deviation (σ) for HV, over 21 runs for each MaOEA. Notably, the symbols "–", "=", or "+" against NSGA-III and MOEA/D-LWS highlight where these MaOEAs are statistically worse than, comparable to or better than HFiDEA, respectively. The results reveal:

- NSGA-III performed worse than HFiDEA for: (i) 32 test cases out of 48 with (a) for t_{TM} setting, and (ii) 30 test cases with setting (b) for t_{TM}.
- MOEA/D-LWS performed worse than HFiDEA for: 33 test cases out of 48 with both t_{TM} settings
- Compared to the DTLZ problems, HFiDEA's performance is relatively better for the WFG problems where the scales of the objective functions are more disparate.

More insights into the functioning of HFiDEA are made possible by Table III. In that, across all test cases, the mean values over 21 HFiDEA runs, are shown for the following:

- percentage association of RVs with single- or multi-solution clusters. The latter case is further bifurcated into instances where the solution α closest to an RV (ensuring uniformity/diversity) is chosen as the cluster representative vis-*à*-vis those where a solution β that *lhf-dominates* α (ensuring better convergence) is chosen. Since NSGA-III and HFiDEA are structurally similar, the α-β percentages have a direct bearing on their

Table III: Insights into functioning of HFiDEA w.r.t. $t_{TM} = 500$: quantifying the occurrences of RV associated clusters with single and multiple solutions, and highlighting the convergence-diversity tradeoff based on the chosen solution (α/β). The tabulated values show the mean over 21 HFiDEA runs. For a single run, the percentage is computed by dividing the total occurrences over t_{TM} generations by the maximum possible count of $N \times t_{TM}$. Also shown are the mean values of $d^{\parallel}$- and $d^{\perp}$-metric. The symbol "–", "=", or "+" means that the concerned algorithm is statistically worse than, comparable to or better than HFiDEA.

Problem	M	Single Solution Clusters (%)	Multi-solution Clusters			Performance Metrics					
			Total (%)	Selected solution		$d^{\parallel}$			$d^{\perp}$		
				α (%)	β (%)	NSGA-III	MOEA/D-LWS	HFiDEA	NSGA-III	MOEA/D-LWS	HFiDEA
DTLZ1	3	25.025	72.317	70.174	2.143	7.7765+	8.5453=	10.2451	0.0443–	0.0586–	0.0390
DTLZ1	5	30.170	67.115	62.361	4.754	13.6569=	17.0464=	16.1876	0.1059=	0.1352–	0.0982
DTLZ1	10	9.366	17.789	16.352	1.436	13.5670=	17.1640–	10.7700	0.1618–	0.2429–	0.1125
DTLZ1	15	27.631	54.984	50.305	4.680	23.6738–	7.2950=	6.4299	0.1523–	0.4247–	0.0978
DTLZ2	3	25.404	74.087	57.596	16.490	1.0003+	1.0013–	1.0006	0.0004+	0.0422–	0.0289
DTLZ2	5	30.198	69.218	53.862	15.355	1.0042–	0.9980+	0.9991	0.0014+	0.1233–	0.0545
DTLZ2	10	9.826	18.834	13.559	5.275	1.0312–	0.9981–	0.9877	0.0120+	0.2513–	0.0936
DTLZ2	15	29.233	61.005	52.790	8.215	1.0352–	1.0006+	1.0037	0.0185+	0.4070–	0.0436
DTLZ3	3	25.145	72.287	70.302	1.985	10.6286+	13.6334=	15.5649	0.0632–	0.0557–	0.0385
DTLZ3	5	30.008	67.367	62.825	4.542	23.1489+	24.2626+	31.4474	0.1108–	0.1310–	0.0997
DTLZ3	10	9.318	18.097	16.727	1.370	33.6846–	22.5155=	19.0273	0.1614–	0.2435–	0.1223
DTLZ3	15	25.534	56.265	51.932	4.333	47.1399–	7.6866=	9.6370	0.1191–	0.4259–	0.0858
DTLZ4	3	26.341	68.155	52.244	15.911	1.0005=	1.0013–	1.0003	0.0543–	0.0409+	0.0432
DTLZ4	5	34.519	64.104	49.603	14.501	1.0024–	0.9974=	0.9972	0.0014+	0.1229–	0.0577
DTLZ4	10	11.977	16.886	13.045	3.841	1.0111–	0.9975–	0.9791	0.0120+	0.2433–	0.1049
DTLZ4	15	36.134	56.780	49.539	7.241	1.0031–	1.0000–	0.9807	0.0127+	0.4073–	0.1194
WFG1	3	27.778	59.867	42.645	17.222	0.9348–	0.9429–	0.8847	0.2158–	0.1649=	0.1252
WFG1	5	30.161	53.713	39.989	13.724	0.8233–	0.7768–	0.7128	0.4049–	0.2727+	0.2925
WFG1	10	3.275	5.915	3.638	2.276	0.7220–	0.8121–	0.5377	0.5477–	0.4867+	0.5182
WFG1	15	20.339	25.997	21.883	4.114	0.5637–	0.7737–	0.5020	0.6768=	0.6549+	0.6753
WFG2	3	32.688	65.085	49.272	15.813	0.5516=	0.5856–	0.5561	0.0613=	0.0737–	0.0489
WFG2	5	34.236	63.126	46.489	16.637	0.3978–	0.5353–	0.3945	0.0536+	0.1498–	0.1191
WFG2	10	9.486	15.169	11.027	4.142	0.2529+	0.5587–	0.2818	0.1752+	0.4007–	0.2158
WFG2	15	26.906	42.520	34.881	7.640	0.2022=	0.8693–	0.2324	0.4580=	0.6697–	0.5321
WFG4	3	34.268	64.003	40.029	23.974	1.0127–	1.0018+	1.0056	0.0073+	0.0407–	0.0284
WFG4	5	34.354	63.026	45.948	17.078	1.0299–	0.9922+	0.9988	0.0260+	0.1151–	0.0617
WFG4	10	9.923	17.485	13.245	4.240	1.0273–	0.9838+	0.9950	0.0841+	0.2860–	0.0959
WFG4	15	24.897	58.310	51.971	6.338	0.9948+	0.9925+	1.0044	0.1385–	0.6238–	0.0982
WFG5	3	33.370	64.304	39.908	24.396	1.0259–	1.0201+	1.0213	0.0201+	0.0424–	0.0296
WFG5	5	35.342	63.008	44.653	18.355	1.0302–	1.0094+	1.0122	0.0339+	0.1157–	0.0635
WFG5	10	10.188	17.673	12.959	4.714	1.0277–	0.9909+	0.9950	0.0653+	0.2150–	0.0980
WFG5	15	28.454	61.976	55.487	6.489	1.0149–	0.9969+	1.0048	0.0882=	0.4588–	0.0888
WFG6	3	30.510	68.099	47.104	20.995	1.0274–	1.0192=	1.0208	0.0152+	0.0426–	0.0281
WFG6	5	32.350	65.831	48.037	17.794	1.0344–	1.0078+	1.0115	0.0279+	0.1224–	0.0584
WFG6	10	9.705	18.024	12.802	5.222	1.0299–	0.9989–	0.9941	0.0498+	0.2427–	0.0947
WFG6	15	27.871	60.251	51.344	8.907	1.0154–	1.0005–	0.9901	0.0572+	0.4171–	0.1070
WFG7	3	32.046	66.311	39.756	26.555	1.0061–	1.0003+	1.0016	0.0035+	0.0407–	0.0285
WFG7	5	33.802	64.132	43.426	20.706	1.0139–	0.9941+	0.9954	0.0131+	0.1217–	0.0574
WFG7	10	9.840	17.348	11.751	5.597	1.0311–	0.9753+	0.9825	0.0525+	0.2310–	0.0974
WFG7	15	25.031	56.948	48.207	8.740	0.9974=	0.9928=	0.9955	0.1057+	0.5161–	0.1104
WFG8	3	29.387	64.068	46.656	17.412	1.0590+	1.0550+	1.0605	0.0448–	0.0482–	0.0412
WFG8	5	33.472	63.335	48.795	14.540	1.0915–	1.0433+	1.0754	0.0746+	0.1189–	0.0813
WFG8	10	9.924	17.269	12.576	4.693	1.1021–	1.0195+	1.0625	0.1223+	0.2594–	0.1302
WFG8	15	25.604	59.079	52.500	6.579	1.0350+	0.9950+	1.0785	0.2067=	0.6193–	0.2066
WFG9	3	33.708	61.973	34.145	27.827	1.0375–	1.0395=	1.0256	0.0302+	0.0488–	0.0335
WFG9	5	38.029	58.588	42.778	15.810	1.0444–	1.0221–	1.0097	0.0629+	0.1153–	0.0726
WFG9	10	12.975	12.714	7.850	4.865	1.0245–	1.0054=	1.0099	0.1131+	0.2364–	0.1279
WFG9	15	38.762	36.802	27.605	9.198	0.9955+	0.9939+	1.0081	0.1721+	0.4264–	0.2443
# –/=/+ $\rightarrow$						**33/06/09**	**17/11/20**	**of 48 probs.**	**13/06/29**	**43/01/04**	**of 48 probs.**

results in Table II. In that, if the same set of RVs and its associated clusters are subjected to the respective environmental selection in NSGA-III and HFiDEA, the cluster representatives picked by both would be exactly the same for single-solution clusters (no option to pick another) and multi-solution clusters where α remains the cluster representative (no solution is able to *lhf-dominates* α). In essence, the better HV reported by HFiDEA compared to NSGA-III could be attributed to the instances where β offering better convergence emerges as the cluster representative. This is further endorsed by the fact that as the β% noticeably improves for the WFG problems compared to DTLZs, the gains in HV offered by HFiDEA are correspondingly better for the WFG problems.

- $d^{\parallel}$ and $d^{\perp}$, which mark the projections of the selected cluster representatives, along and perpendicular to the corresponding RVs. In that smaller mean for $d^{\parallel}$ signifies overall better convergence, while smaller mean for $d^{\perp}$ signifies better proximity from RVs, hence, more uniform distribution or better diversity. Notably, much along the expected trend, HFiDEA offers: (i) better convergence than NSGA-III in 33 out of 48 test instances, while poorer uniformity/diversity in 29 test instances, and (ii) poorer convergence than MOEA/D-LWS in 20 test instances but better uniformity/diversity in 43 instances (since MOEA/D-LWS gives unilateral preference to convergence).

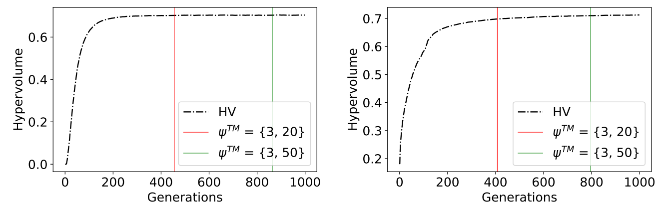
The above results interpreted in conjunction with the DD method for generating the RVs, offer important insights on the impact of the convergence-diversity tradeoff on the HV metric. Notably, the DD method provides a set of equidistant reference-points, that are joined with the origin to obtain a set of uniformly distributed RVs. Since HV accounts for both convergence and uniformity of solutions, HFiDEA's selection of β over α affects HV in two ways: (i) it promises improved HV on account of better convergence, and (ii) it allows for HV deterioration on account of poorer uniformity/diversity. The fact that despite the contradictory possibilities, HFiDEA finally achieves better HV than NSGA-III for 32 of 48 test cases, testifies that the improvement in convergence offered by it overpowered the deterioration in uniformity/diversity.

While Table II presented the results for t_{TM} - set a priori and determined *on-the-fly*, a detailed discussion of these results above, is being restricted to the former case, *due to paucity of space*. However, two aspects are worth highlighting regarding the t_{TM} determination *on-the-fly*:

- for how long to run an MaOEA for unknown scenarios (test problems with modified parameters or real-world problems) is a non-trivial question, often hard to respond to, by pure experience or judgement. In such a situation, having an MaOEA equipped with demonstrated capability to reliably determine t_{TM} *on-the-fly*, should reinforce a lot of confidence for researchers and practitioners, alike. In this context, experiments had been performed for two settings of $\psi^{TM} \equiv \{n_p, n_s\} = \{3,20\}$ and $\{3,50\}$. Clearly, as the degree of *stability* tracked with the $\{3,50\}$ setting ought to be *stricter* than that corresponding to $\{3,20\}$, it is fair to expect more deferred and reliable termination with the former setting.[2]. Its evidence lies in Figure 2, where the HV corresponding to $\{3,50\}$ can be seen to be more stable than that corresponding to $\{3,20\}$.
- the HV, $d^{\parallel}$ and $d^{\perp}$ trends around the relative performance of the three MaOEAs are consistent with those reported for $t_{TM} = 500$. While these trends for both the $\{3,20\}$ and $\{3,50\}$ settings are detailed in the supplementary file, the summary for the latter is highlighted in Table IV.

Finally, the performance of the three MaOEAs, each partially augmented with the UEA is presented in terms of HV, in the Table V, for $t_{TM} = 500$, and 36 test instances. While the corresponding $d^{\parallel}$ and $d^{\perp}$ measures are detailed in the supplementary file, their summary can be found in Table IV. Notably, $M = 8$ is included to compensate for elimination of M=10 and 15 owing to the associated complexity. It is evident that the number

[2]This explains why under space considerations, the HV results corresponding to $\{n_p, n_s\} = \{3,50\}$ had been presented in Table II



(a) DTLZ4 ($M = 5$) **(b)** WFG4 ($M = 5$)

Figure 2: On reliability of t_{TM} determined *on-the-fly*, by tracking the stability of HFiDEA's population over the generations

Table IV: Count of instances in which HFiDEA performed better than/comparable to/worse than NSGA-III and MOEA/D-LWS, in terms of HV, $d^{\parallel}$- and $d^{\perp}$-metrics.

Case		$\psi^{TM} = \{3,50\}$	$t_{TM} = 500$ with UEA
HV	NSGA-III	**30**/05/13 of 48	**25**/04/07 of 36
	MOEA/D-LWS	**33**/04/11 of 48	**23**/04/09 of 36
$d^{\parallel}$	NSGA-III	**31**/08/09 of 48	**24**/06/06 of 36
	MOEA/D-LWS	**20**/11/17 of 48	08/02/**26** of 36
$d^{\perp}$	NSGA-III	11/06/**31** of 48	11/08/**17** of 36
	MOEA/D-LWS	**43**/01/04 of 48	**32**/02/02 of 36

of instances where: (i) MOEA/D-LWS is worse than HFiDEA, falls from 33 of 48 test cases (68.75%) to 23 of 36 test cases (63.88%), and (ii) HFiDEA is better than NSGA-III increases from 32 of 48 test cases (66.67%) to 25 of 36 test cases (69.44%). The plausible reasons for the above trends are, as below:

- the UEA carries the scope of improving the uniformity/diversity in the case of MOEA/D-LWS and HFiDEA, since they accept digression from the RVs in pursuit of better convergence. Notably, while MOEA/D-LWS does not restrict the digression from the RVs at all, HFiDEA attempts to limit it by picking the β solution with minimum $d^{\perp}$ from the RV, among the possible alternatives. This explains as to why the degree of improvement is higher for MOEA/D-LWS.
- in the case of NSGA-III, the UEA does not provide the scope for improvement either in convergence or uniformity/diversity. The later holds as the α solution is always preferred, while the former is true since previous generation solutions inducted as part of the UEA cannot dominate the later ones. This is manifested in the results too, where the relative performance of NSGA-III deteriorates.

E. Computational complexity

In the proposed HFiDEA algorithm, the worst-case complexity is $O(N^2M)$ corresponding to the computation of $d^{\perp}$ in line-3, Algorithm 2. In this, there are $N \times N$ computations of $d^{\perp}$ (whose complexity is $O(M)$). Similarly, MOEA/D-LWS (without UEA) also has the worst-case complexity of $O(N^2M)$ corresponding to the angle calculation (instead of $d^{\perp}$ in HFiDEA), which requires $N \times N$ computations of angle function which is

Table V: Performance of partially UEA-assisted NSGA-III, MOEA/D-LWS, and HFiDEA, using the mean (μ) and standard deviation (σ) for Hypervolume. The results presented as $\mu(\sigma)$ correspond to $t_{TM} = 500$. The symbols "–", "=", or "+" highlight where the concerned algorithm is statistically worse than, comparable to, or better than HFiDEA, respectively.

Problems	M	NSGA-III	MOEA/D-LWS	HFiDEA
DTLZ1	3	0.969252 (0.0649) =	1.010720 (0.0248) +	0.988405 (0.0396)
DTLZ1	5	1.010540 (0.0232) +	1.012810 (0.0208) +	0.991666 (0.0350)
DTLZ1	8	0.895505 (0.2528) =	1.022000 (0.0955) +	0.959304 (0.1230)
DTLZ2	3	0.447025 (0.0001) +	0.421763 (0.0008) –	0.443114 (0.0004)
DTLZ2	5	0.715932 (0.0006) +	0.623055 (0.0036) –	0.709372 (0.0014)
DTLZ2	8	0.859128 (0.0034) –	0.718863 (0.0123) –	0.898382 (0.0021)
DTLZ3	3	0.211491 (0.2024) –	0.713855 (0.3338) +	0.411990 (0.2820)
DTLZ3	5	0.182282 (0.2115) +	0.394125 (0.3866) +	0.079257 (0.2315)
DTLZ3	8	0.020596 (0.0881) =	0.472405 (0.4080) +	0.003536 (0.0116)
DTLZ4	3	0.404757 (0.1282) –	0.403231 (0.0088) –	0.432392 (0.0485)
DTLZ4	5	0.717250 (0.0026) +	0.597390 (0.0154) –	0.710309 (0.0009)
DTLZ4	8	0.878040 (0.0026) –	0.652158 (0.0175) –	0.889507 (0.0186)
WFG1	3	0.320035 (0.0199) –	0.374203 (0.0505) =	0.382411 (0.0053)
WFG1	5	0.276166 (0.0161) –	0.432448 (0.0154) +	0.346689 (0.0164)
WFG1	8	0.178490 (0.0085) –	0.359160 (0.0323) +	0.317090 (0.0254)
WFG2	3	0.858565 (0.0728) –	0.883035 (0.0740) –	0.899257 (0.0629)
WFG2	5	0.975746 (0.0556) =	0.957336 (0.0883) =	0.938521 (0.0951)
WFG2	8	0.949544 (0.0927) +	0.880788 (0.0866) =	0.880819 (0.0995)
WFG4	3	0.419455 (0.0030) –	0.420603 (0.0018) –	0.429901 (0.0022)
WFG4	5	0.633546 (0.0051) –	0.679421 (0.0071) –	0.699241 (0.0026)
WFG4	8	0.698776 (0.0200) –	0.788539 (0.0108) –	0.833313 (0.0098)
WFG5	3	0.386748 (0.0029) –	0.376683 (0.0018) –	0.394372 (0.0022)
WFG5	5	0.615226 (0.0028) –	0.613744 (0.0071) –	0.655903 (0.0040)
WFG5	8	0.723671 (0.0069) –	0.746894 (0.0119) –	0.810169 (0.0046)
WFG6	3	0.388960 (0.0059) –	0.382918 (0.0071) –	0.397296 (0.0066)
WFG6	5	0.618555 (0.0092) –	0.602648 (0.0125) –	0.661215 (0.0098)
WFG6	8	0.742182 (0.0122) –	0.678312 (0.0206) –	0.825937 (0.0144)
WFG7	3	0.434282 (0.0012) –	0.423643 (0.0009) –	0.439548 (0.0005)
WFG7	5	0.681151 (0.0029) –	0.662172 (0.0060) –	0.714750 (0.0009)
WFG7	8	0.780477 (0.0084) –	0.795141 (0.0092) –	0.884225 (0.0031)
WFG8	3	0.342799 (0.0020) +	0.330872 (0.0039) –	0.337880 (0.0022)
WFG8	5	0.527161 (0.0057) –	0.562197 (0.0044) +	0.549812 (0.0113)
WFG8	8	0.560721 (0.0106) –	0.595490 (0.0072) =	0.582388 (0.0441)
WFG9	3	0.359172 (0.0210) –	0.330958 (0.0008) –	0.377717 (0.0275)
WFG9	5	0.544587 (0.0055) –	0.553819 (0.0055) –	0.618318 (0.0200)
WFG9	8	0.587144 (0.0231) –	0.563481 (0.0261) –	0.686490 (0.0358)
# –/=/+ →		**25/04/07**	**23/04/09**	**of 36 problems**

dependent on M. This indicates that both HFiDEA and MOEA/D-LWS (without UEA) are equivalent in context of computational complexity. However, if UEA is to be inducted as an integral part of MOEA/D-LWS, then its complexity practically turns intractable with increasing generations, as already cited in Section V.

For NSGA-III, the reported computational complexity is $O(N \log^{M-2} N)$ or $O(N^2 M)$, whichever is larger [32]. In that, the first component is for non-dominated sorting, while the second component is for $d^{\perp}$ computations. Here, for the N sizes conventionally employed for different Ms (as also in this paper), it is empirically established in Table VI, that the first component exceeds the second component. Hence, the worst-case complexity of NSGA-III could be treated as $O(N \log^{M-2} N)$.

In the wake of the above, it is fair to infer that the worst-case computational complexity of HFiDEA is lower than that of NSGA-III and MOEA/D-LWS.

Table VI: Comparison of two computational complexities for different combinations (used in this study) of M and N.

M, N	5, 210	8, 156	10, 275	15, 135
$N^2 M$	**2.20E+05**	1.94E+05	7.56E+05	2.73E+05
$N \log^{M-2} N$	9.64E+04	**2.33E+07**	**5.11E+09**	**1.51E+13**

VI. Conclusions

We have proposed a computationally efficient MaOEA, namely, HFiDEA, which ensures diversity through association of population members with reference vectors (RVs), and pursues convergence through the newly proposed *lhf-dominance*. While the limitations of Pareto-dominance and non-domination ranking in the context of MaOPS are well documented for nearly two decades, the proposition of *lhf-dominance* marks an effort to overcome these limitations in a comprehensive and practical manner. The solutions are discriminated by conjunct consideration of (i) the number of objectives in which they are better/worse, (ii) the degree by which they are better/worse in the objectives, and (iii) the scope to induce relative preferences between the objectives that could be altered for different segments of the objective space. The relative preferences between the objectives are aligned with the different weights for different RVs to avoid *subjectivity*. However, these preferences could possibly be fetched as inputs from the decision maker(s) - an aspect which may be interesting from the multi-criterion decision making perspective. While in the later context, more research is required, it is fair to infer that these preferences may alter the distribution of solutions in the approximated PF. Besides these fundamental contributions, this paper has also dealt with issues of practical relevance, including the timing for nadir point estimation/update, and *stability* tracking enabling HFiDEA's termination *on-the-fly*. Motivated by these contributions, the next step is to integrate the constraint handling and automated redundancy determination mechanisms within HFiDEA, so its utility could be fully demonstrated to real-world problems.

References

[1] K. Deb, *Multi-Objective Optimization Using Evolutionary Algorithms*. New York, NY, USA: John Wiley & Sons, Inc., 2001.

[2] 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, pp. 182–197, Apr 2002.

[3] E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: Improving the strength pareto evolutionary algorithm for multiobjective optimization," in *Evolutionary Methods for Design Optimization and Control with Applications to Industrial Problems* (K. C. Giannakoglou, D. T. Tsahalis, J. Périaux, K. D. Papailiou, and T. Fogarty, eds.), (Athens, Greece), pp. 95–100, International Center for Numerical Methods in Engineering, 2001.

[4] E. Zitzler and S. Künzli, "Indicator-based selection in multiobjective search," in *Parallel Problem Solving from Nature - PPSN VIII* (X. Yao, E. K. Burke, J. A. Lozano, J. Smith, J. J. Merelo-Guervós, J. A. Bullinaria, J. E. Rowe, P. Tiňo, A. Kabán, and H.-P. Schwefel, eds.), (Berlin, Heidelberg), pp. 832–842, Springer Berlin Heidelberg, 2004.

[5] 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.

[6] E. J. Hughes, "Evolutionary many-objective optimisation: many once or one many?," in *IEEE Congress on Evolutionary Computation*, pp. 222–227, 2005.

[7] K. Deb and D. K. Saxena, "Searching for pareto-optimal solutions through dimensionality reduction for certain large-dimensional multi-objective optimization problems," in *IEEE Congress on Evolutionary Computation*, pp. 3353–3360, 2006.

[8] R. C. Purshouse and P. J. Fleming, “On the evolutionary optimization of many conflicting objectives,” *IEEE Transactions on Evolutionary Computation*, vol. 11, pp. 770–784, Dec 2007.

[9] D. J. Walker, R. Everson, and J. E. Fieldsend, “Visualizing mutually nondominating solution sets in many-objective optimization,” *IEEE Transactions on Evolutionary Computation*, vol. 17, no. 2, pp. 165–184, 2013.

[10] K. Deb, *Multi-objective optimization using evolutionary algorithms*. Chichester, UK: Wiley, 2001.

[11] M. Laumanns, L. Thiele, K. Deb, and E. Zitzler, “Combining convergence and diversity in evolutionary multiobjective optimization,” *Evolutionary Computation*, vol. 10, pp. 263–282, Sept 2002.

[12] S. Yang, M. Li, X. Liu, and J. Zheng, “A grid-based evolutionary algorithm for many-objective optimization,” *IEEE Transactions on Evolutionary Computation*, vol. 17, no. 5, pp. 721–736, 2013.

[13] F. di Pierro, S. T. Khu, and D. A. Savic, “An investigation on preference order ranking scheme for multiobjective evolutionary optimization,” *IEEE Transactions on Evolutionary Computation*, vol. 11, pp. 17–45, Feb 2007.

[14] M. Farina and P. Amato, “A Fuzzy Definition of “Optimality” for Many-Criteria Optimization Problems,” *IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans*, vol. 34, no. 3, pp. 315–326, 2004.

[15] X. Zou, Y. Chen, M. Liu, and L. Kang, “A new evolutionary algorithm for solving many-objective optimization problems,” *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 38, pp. 1402–1412, Oct 2008.

[16] X. Zhang, Y. Tian, and Y. Jin, “A knee point-driven evolutionary algorithm for many-objective optimization,” *IEEE Transactions on Evolutionary Computation*, vol. 19, no. 6, pp. 761–776, 2015.

[17] S. F. Adra and P. J. Fleming, “Diversity management in evolutionary many-objective optimization,” *IEEE Transactions on Evolutionary Computation*, vol. 15, no. 2, pp. 183–195, 2011.

[18] 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, 2014.

[19] M. Köppen and K. Yoshida, “Substitute distance assignments in nsga-ii for handling many-objective optimization problems,” in *Evolutionary Multi-Criterion Optimization* (S. Obayashi, K. Deb, C. Poloni, T. Hiroyasu, and T. Murata, eds.), (Berlin, Heidelberg), pp. 727–741, Springer Berlin Heidelberg, 2007.

[20] N. Beume, B. Naujoks, and M. Emmerich, “Sms-emoa: Multiobjective selection based on dominated hypervolume,” *European Journal of Operational Research*, vol. 181, no. 3, pp. 1653–1669, 2007.

[21] J. Bader and E. Zitzler, “Hype: An algorithm for fast hypervolume-based many-objective optimization,” *Evolutionary Computation*, vol. 19, pp. 45–76, March 2011.

[22] R. H. Gómez and C. A. C. Coello, “Mombi: A new metaheuristic for many-objective optimization based on the r2 indicator,” in *2013 IEEE Congress on Evolutionary Computation*, pp. 2488–2495, June 2013.

[23] H. Ishibuchi, N. Akedo, and Y. Nojima, “Relation between neighborhood size and moea/d performance on many-objective problems,” pp. 459–474, Springer International Publishing, 2013.

[24] R. Wang, Q. Zhang, and T. Zhang, “Decomposition-based algorithms using pareto adaptive scalarizing methods,” *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 6, pp. 821–837, 2016.

[25] M. Asafuddoula, T. Ray, and R. Sarker, “A decomposition-based evolutionary algorithm for many objective optimization,” *IEEE Transactions on Evolutionary Computation*, vol. 19, no. 3, pp. 445–460, 2015.

[26] R. Wang, Z. Zhou, H. Ishibuchi, T. Liao, and T. Zhang, “Localized weighted sum method for many-objective optimization,” *IEEE Transactions on Evolutionary Computation*, vol. 22, no. 1, pp. 3–18, 2018.

[27] H. Ishibuchi, Y. Sakane, N. Tsukamoto, and Y. Nojima, “Adaptation of scalarizing functions in moea/d: An adaptive scalarizing function-based multiobjective evolutionary algorithm,” in *Evolutionary Multi-Criterion Optimization* (M. Ehrgott, C. M. Fonseca, X. Gandibleux, J.-K. Hao, and M. Sevaux, eds.), (Berlin, Heidelberg), pp. 438–452, Springer Berlin Heidelberg, 2009.

[28] H. Ishibuchi, Y. Sakane, N. Tsukamoto, and Y. Nojima, “Simultaneous use of different scalarizing functions in moea/d,” in *Proceedings of the 12th Annual Conference on Genetic and Evolutionary Computation*, GECCO ’10, (New York, NY, USA), p. 519–526, Association for Computing Machinery, 2010.

[29] R. Wang, R. C. Purshouse, and P. J. Fleming, “Preference-inspired coevolutionary algorithms for many-objective optimization,” *IEEE Transactions on Evolutionary Computation*, vol. 17, no. 4, pp. 474–494, 2013.

[30] Y. Yuan, H. Xu, B. Wang, and X. Yao, “A new dominance relation-based evolutionary algorithm for many-objective optimization,” *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 1, pp. 16–37, 2016.

[31] 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, 2014.

[32] 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,” *Evolutionary Computation, IEEE Transactions on*, vol. 18, pp. 577–601, Aug 2014.

[33] H. Jain and K. Deb, “An evolutionary many-objective optimization algorithm using reference-point based nondominated sorting approach, part ii: Handling constraints and extending to an adaptive approach,” *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 4, pp. 602–622, 2014.

[34] Y. Yuan, H. Xu, and B. Wang, “An improved nsga-iii procedure for evolutionary many-objective optimization,” in *Proceedings of the 2014 Annual Conference on Genetic and Evolutionary Computation*, GECCO ’14, (New York, NY, USA), pp. 661–668, ACM, 2014.

[35] 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, 2015.

[36] H. Seada and K. Deb, “A unified evolutionary optimization procedure for single, multiple, and many objectives,” *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 3, pp. 358–369, 2016.

[37] H. Wang, L. Jiao, and X. Yao, “Two_arch2: An improved two-archive algorithm for many-objective optimization,” *IEEE Transactions on Evolutionary Computation*, vol. 19, no. 4, pp. 524–541, 2015.

[38] M. Farina and P. Amato, “A fuzzy definition of "optimality" for many-criteria optimization problems,” *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, vol. 34, no. 3, pp. 315–326, 2004.

[39] M. Zeleny, “Optimal system design with multiple criteria: De novo programming approach,” *Engineering Costs and Production Economics*, vol. 10, no. 2, pp. 89 – 94, 1986.

[40] D. K. Saxena and S. Kapoor, “On timing the nadir-point estimation and/or termination of reference-based multi- and many-objective evolutionary algorithms,” in *Evolutionary Multi-Criterion Optimization* (K. Deb, E. Goodman, C. A. Coello Coello, K. Klamroth, K. Miettinen, S. Mostaghim, and P. Reed, eds.), (Cham), pp. 191–202, Springer International Publishing, 2019.

[41] L. He, H. Ishibuchi, A. Trivedi, and D. Srinivasan, “Dynamic normalization in moea/d for multiobjective optimization,” in *2020 IEEE Congress on Evolutionary Computation (CEC)*, pp. 1–8, 2020.

[42] A. P. Wierzbicki, “The use of reference objectives in multiobjective optimization,” in *Multiple Criteria Decision Making Theory and Applications* (G. Fandel and T. Gal, eds.), pp. 468–486, Berlin: Springer-Verlag, 1980.

[43] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler, *Scalable Test Problems for Evolutionary Multiobjective Optimization*, pp. 105–145. London: Springer London, 2005.

[44] S. Huband, P. Hingston, L. Barone, and L. While, “A review of multiobjective test problems and a scalable test problem toolkit,” *IEEE Transactions on Evolutionary Computation*, vol. 10, no. 5, pp. 477–506, 2006.

[45] I. Das and J. Dennis, “Normal-boundary intersection: A new method for generating the Pareto surface in nonlinear multicriteria optimization problems,” *SIAM Journal of Optimization*, vol. 8, no. 3, pp. 631–657, 1998.

[46] H. Ishibuchi, R. Imada, Y. Setoguchi, and Y. Nojima, "How to specify a reference point in hypervolume calculation for fair performance comparison," *Evol. Comput.*, vol. 26, p. 411–440, Sept. 2018.

A Supplementary File for "A Localized-high-fidelity Dominance based Many-Objective Evolutionary Algorithm"

Dhish Kumar Saxena, Sukrit Mittal, Sarang Kapoor, and Kalyanmoy Deb

This supplementary file provides the results to further clarify and support the claims of the original paper, including the following:

- first, Figure S1 facilitates insights in to the potential choice of a solution by the *environmental selection* operator in HFiDEA, if it were applied to a cluster of non-dominated solutions around each reference vector, for different PF shapes.
- Table S.1 presents the performance comparison at t_{TM} determined by HFiDEA using $\psi^{TM} = \{n_p, n_s\} = \{3, 20\}$, in context of hypervolume, $d^{\parallel}$ and $d^{\perp}$ metrics.
- Table S.2 presents the performance comparison at t_{TM} determined by HFiDEA using $\psi^{TM} = \{n_p, n_s\} = \{3, 50\}$, in context of $d^{\parallel}$ and $d^{\perp}$ metrics.
- Table S.3 presents the performance comparison at $t_{TM} = 500$ (with UEA), in context of $d^{\parallel}$ and $d^{\perp}$ metrics.

As discussed in the main paper, at any given generation, HFiDEA clusters the (combined parent and child) solutions around each reference-vector, and chooses the cluster representative based on *lhf-dominance*. As an example, Figure 1 (Section IV) in the main paper, presents a sample cluster, and discusses the application of the *lhf-dominance* based environmental selection, leading to the determination of the cluster representative. Here, the aim is to demonstrate at a more *generic* level that the chosen cluster representatives follow certain patterns. These patterns are a result of how the constituents of *lhf-dominance*, including, n_b (number of objectives in which a solution is better/worse than another) and ΔF (weighted degree of betterment in the objectives) favor certain solutions more than others.

Towards it, Figures S1a, S1c, and S1e, show the clusters of solutions around different reference vectors, for linear, convex and concave PF shapes, respectively. In that, the points where the reference-vectors intersect the PF surface are highlighted (black crosses). For simplicity, it is assumed that one solution in each cluster coincides with the marked (black cross) point, so it serves as the α-solution for each reference-vector (since $d^{\perp} = 0$ for such a point). Then for each cluster, its representative solution chosen (as per Algorithm II, main paper), is as below:

- for reference vectors which coincide with the objective axis, *lhf-dominance* is not performed and α is picked by default. Hence, in the Figures S1b, S1d and S1f, the cluster representatives for such reference vectors remain as the black crosses.
- for all other reference vectors, the chosen solution is one, that: (i) *lhf-dominates* the α-solution and (ii) is nearest to the reference-vector. The resulting cluster representatives are shown by red dots in the Figures S1b, S1d and S1f, respectively.

Notably, the chosen cluster representative reveal certain patterns, in that:

- each representative seems to be tending towards the extreme point. This could be attributed to the fact that for any two non-dominated solutions, namely, X and Y, the maximum possible value of n_b^X or n_b^Y is $M-1$ which points to the extreme points.
- the clusters for the reference vectors passing through the edges find a representative on the edge itself. This could be attributed to the ΔF (weighted degree of betterment in the objectives) computation.

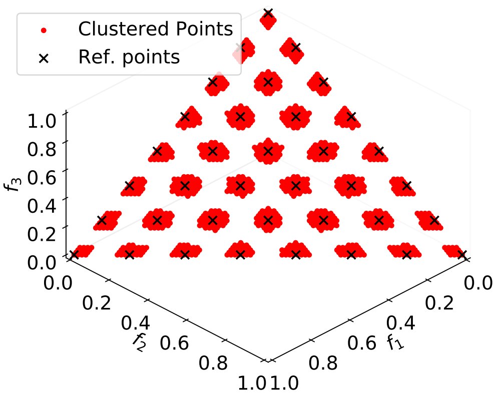
$$\Delta F(X, Y) = \Sigma_{i=1}^{M} \frac{f_i^X - f_i^Y}{w_i} \tag{1}$$

For any reference vector passing through an edge, at least one component of w is 0, say w_k. Hence, while comparing two solutions, the ΔF criterion gives infinite importance to the corresponding objective, f_k. Given this, if an edge and non-edge solution is compared, the former will be selected owing to better value in f_k. For instance, if $M = 3$, a reference-vector passing through the f_1–f_2 plane, implies $w_3 = 0$ and infinite importance to f_3. Hence, ΔF will select the solution belonging to the f_1–f_2 plane.

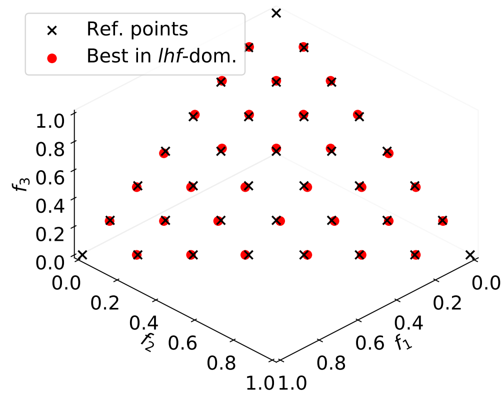
D.K. Saxena and S. Mittal are with the Department of Mechanical and Industrial Engineering, Indian Institute of Technology Roorkee, Uttarakhand-247667, India, e-mail: {dhish.saxena, smittal1}@me.iitr.ac.in.

S. Kapoor is with the Department of Computer Science and Engineering, Indian Institute of Technology Roorkee, Uttarakhand-247667, India, e-mail: skapoor@cs.iitr.ac.in.

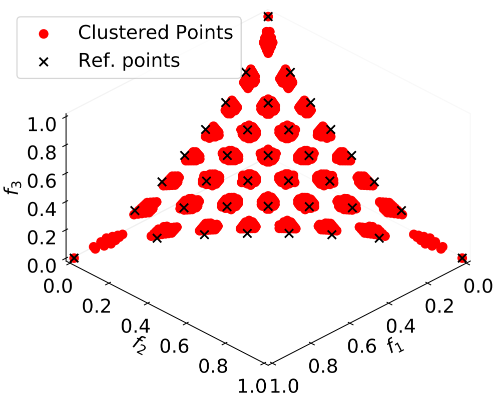
K. Deb is with the Beacon NSF Center and the Department of Electrical and Computer Engineering, Michigan State University, East Lansing, MI, 48824, USA, e-mail: kdeb@egr.msu.edu.



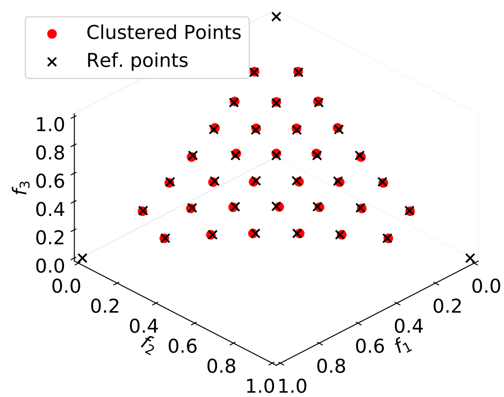
(a) Linear Front (All Clustered Solutions)



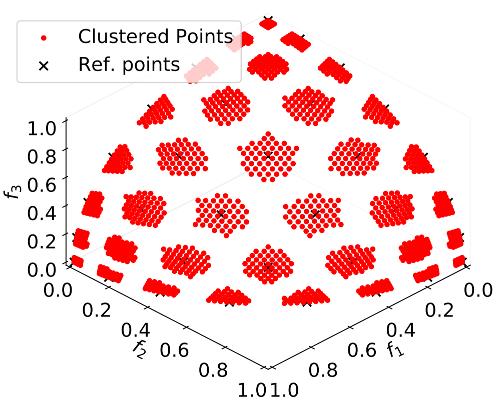
(b) Linear Front (Selected Best Solution)



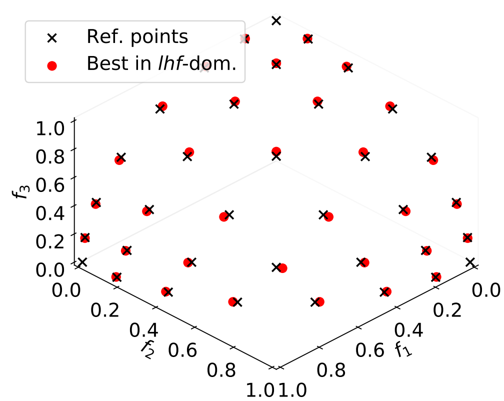
(c) Convex Front (All Clustered Solutions)



(d) Convex Front (Selected Best Solution)



(e) Concave Front (All Clustered Solutions)



(f) Concave Front (Selected Best Solution)

Figure S1: Clustered solutions (Pareto-non-dominated) initialized in the neighbourhood of each reference-vector ((a), (c), (e)), and best solution (identified by Environment Selection of HFiDEA) in the neighbourhood of each reference-vector ((b), (d), (f)) for different PF shapes.

Table S.1: Performance of NSGA-III, MOEA/D-LWS (no UEA) and HFiDEA, using the mean (μ) and standard deviation (σ) of Hypervolume, over 21 runs each (of 48 problems). The results are presented as $\mu(\sigma)$, corresponding to t_{TM} determined on-the-fly by the stability tracking algorithm with $\psi^{TM} = \{n_p, n_s\} = \{3, 20\}$. The mean of $d^{\parallel}$- and $d^{\perp}$-metrics are also shown. The symbols "–", "=", or "+" against NSGA-III and MOEA/D-LWS (here referred to as M-LWS for brevity) highlight where these are statistically worse than, comparable to or better than HFiDEA, respectively.

Problems	M	t_{TM}	Hypervolume			$d^{\parallel}$-metric			$d^{\perp}$-metric		
			NSGA-III	M-LWS	HFiDEA	NSGA-III	M-LWS	HFiDEA	NSGA-III	M-LWS	HFiDEA
DTLZ1	3	302	0.2493 (0.2590) =	0.9018 (0.1086) +	0.2651 (0.2437)	43.0694 =	20.5112 +	42.3749	0.0608 –	0.0587 –	0.0416
DTLZ1	5	356	0.4974 (0.2544) =	0.8690 (0.2031) +	0.4791 (0.2489)	36.3226 =	29.8049 +	39.7287	0.1011 =	0.1353 –	0.0956
DTLZ1	10	387	0.7872 (0.2470) –	1.0100 (0.0488) =	0.9904 (0.1274)	29.6788 –	30.9548 –	14.9723	0.1805 –	0.2415 –	0.1121
DTLZ1	15	407	0.4685 (0.3929) –	1.1174 (0.0013) +	1.1168 (0.0043)	37.4686 –	10.6157 –	7.3355	0.1906 –	0.4241 –	0.1067
DTLZ2	3	248	0.4438 (0.0007) +	0.4133 (0.0046) –	0.4349 (0.0012)	1.0021 =	1.0025 –	1.0020	0.0016 +	0.0422 –	0.0285
DTLZ2	5	295	0.7042 (0.0018) +	0.6326 (0.0076) –	0.7008 (0.0019)	1.0123 –	0.9987 +	1.0027	0.0040 +	0.1237 –	0.0557
DTLZ2	10	334	0.9127 (0.0051) –	0.7381 (0.0172) –	0.9617 (0.0013)	1.0491 –	0.9987 –	0.9902	0.0192 +	0.2502 –	0.0985
DTLZ2	15	348	1.0342 (0.0081) –	0.1579 (0.0589) –	1.0562 (0.0375)	1.0539 –	1.0013 +	1.0065	0.0240 +	0.4082 –	0.0593
DTLZ3	3	325	0.0019 (0.0087) =	0.2513 (0.2519) +	0.0000 (0.0000)	40.1151 =	21.5213 +	46.5098	0.0965 –	0.0561 –	0.0384
DTLZ3	5	349	0.0000 (0.0000) =	0.0086 (0.0183) =	0.0000 (0.0000)	67.2305 =	49.2727 +	77.7079	0.1104 –	0.1308 –	0.0956
DTLZ3	10	349	0.0000 (0.0000) –	0.0360 (0.0804) =	0.1205 (0.2777)	83.1324 –	46.1104 –	33.4502	0.1536 –	0.2441 –	0.1183
DTLZ3	15	352	0.0000 (0.0000) –	0.5186 (0.4344) =	0.7351 (0.4837)	91.3595 –	17.3831 –	13.1225	0.1269 –	0.4233 –	0.1017
DTLZ4	3	465	0.3938 (0.1330) –	0.4168 (0.0039) –	0.4285 (0.0453)	1.0004 =	1.0013 –	1.0003	0.0685 –	0.0410 +	0.0430
DTLZ4	5	455	0.7185 (0.0006) +	0.6358 (0.0033) –	0.7044 (0.0016)	1.0030 –	0.9974 +	0.9978	0.0016 +	0.1226 –	0.0575
DTLZ4	10	417	0.9478 (0.0013) –	0.7793 (0.0156) –	0.9608 (0.0012)	1.0163 –	0.9974 –	0.9800	0.0151 +	0.2432 –	0.1066
DTLZ4	15	362	1.0703 (0.0017) +	0.1148 (0.0044) –	1.0583 (0.0081)	1.0126 –	1.0000 –	0.9822	0.0159 +	0.4073 –	0.1198
WFG1	3	520	0.3216 (0.0212) –	0.3793 (0.0484) =	0.3867 (0.0054)	0.9311 –	0.9351 –	0.8822	0.2103 –	0.1607 =	0.1240
WFG1	5	570	0.2908 (0.0173) –	0.4462 (0.0165) +	0.3571 (0.0170)	0.7980 –	0.7645 –	0.7049	0.3881 –	0.2621 +	0.2869
WFG1	10	690	0.2241 (0.0193) –	0.3542 (0.0383) =	0.3553 (0.0576)	0.6896 –	0.8352 –	0.5279	0.5445 –	0.4721 +	0.5108
WFG1	15	849	0.3018 (0.0128) =	0.2635 (0.0111) –	0.2899 (0.0557)	0.4960 –	0.7018 –	0.4807	0.6751 –	0.6255 +	0.6688
WFG2	3	419	0.8562 (0.0728) –	0.8735 (0.0752) =	0.8972 (0.0627)	0.5532 =	0.5797 –	0.5564	0.0623 =	0.0769 –	0.0494
WFG2	5	488	0.9759 (0.0558) =	0.9571 (0.0882) =	0.9387 (0.0952)	0.3977 –	0.5347 –	0.3938	0.0542 +	0.1496 –	0.1190
WFG2	10	510	0.9645 (0.0857) +	0.7761 (0.1537) –	0.9186 (0.0852)	0.2530 +	0.5548 –	0.2835	0.1743 +	0.3985 –	0.2151
WFG2	15	528	1.0056 (0.0933) +	0.4568 (0.1710) –	0.8100 (0.1053)	0.2017 =	0.8746 –	0.2350	0.4588 =	0.6691 –	0.5288
WFG4	3	351	0.4110 (0.0030) –	0.4167 (0.0019) –	0.4210 (0.0032)	1.0168 –	1.0035 +	1.0084	0.0100 +	0.0400 –	0.0285
WFG4	5	407	0.6220 (0.0058) –	0.6754 (0.0066) –	0.6941 (0.0029)	1.0338 –	0.9924 +	1.0003	0.0306 +	0.1154 –	0.0631
WFG4	10	436	0.7263 (0.0201) –	0.7469 (0.2015) –	0.8830 (0.0139)	1.0265 –	0.9837 +	0.9968	0.0942 =	0.2914 –	0.0982
WFG4	15	416	0.7478 (0.0203) –	0.1517 (0.0612) –	0.8872 (0.0251)	0.9847 +	0.9925 +	1.0036	0.1633 –	0.6321 –	0.1160
WFG5	3	327	0.3792 (0.0030) –	0.3738 (0.0027) –	0.3869 (0.0025)	1.0298 –	1.0205 +	1.0227	0.0227 +	0.0432 –	0.0300
WFG5	5	376	0.6024 (0.0029) –	0.6223 (0.0058) –	0.6495 (0.0044)	1.0343 –	1.0093 +	1.0133	0.0389 +	0.1126 –	0.0649
WFG5	10	435	0.7618 (0.0058) –	0.8200 (0.0079) –	0.8726 (0.0045)	1.0296 –	0.9906 +	0.9956	0.0696 +	0.2145 –	0.0992
WFG5	15	419	0.7964 (0.0126) –	0.2000 (0.0492) –	0.8648 (0.0218)	1.0152 –	0.9961 +	1.0075	0.1001 =	0.4602 –	0.0970
WFG6	3	304	0.3764 (0.0061) –	0.3760 (0.0088) –	0.3858 (0.0069)	1.0336 –	1.0211 +	1.0248	0.0184 +	0.0429 –	0.0285
WFG6	5	342	0.5981 (0.0096) –	0.6027 (0.0111) –	0.6509 (0.0098)	1.0415 –	1.0077 +	1.0139	0.0343 +	0.1229 –	0.0612
WFG6	10	395	0.7787 (0.0157) –	0.7219 (0.0211) –	0.8804 (0.0205)	1.0344 –	0.9991 –	0.9946	0.0588 +	0.2426 –	0.0975
WFG6	15	361	0.8435 (0.0321) –	0.1812 (0.0655) –	0.9322 (0.0268)	1.0183 –	0.9995 –	0.9862	0.0780 +	0.4187 –	0.1305
WFG7	3	341	0.4266 (0.0020) –	0.4209 (0.0013) –	0.4326 (0.0017)	1.0095 –	1.0008 +	1.0030	0.0059 +	0.0408 –	0.0284
WFG7	5	381	0.6672 (0.0043) –	0.6635 (0.0082) –	0.7099 (0.0010)	1.0184 –	0.9933 +	0.9962	0.0180 +	0.1213 –	0.0582
WFG7	10	421	0.8032 (0.0075) –	0.8855 (0.0070) –	0.9473 (0.0022)	1.0347 –	0.9751 +	0.9832	0.0616 +	0.2300 –	0.0994
WFG7	15	361	0.8574 (0.0221) –	0.2195 (0.1240) –	0.9107 (0.0402)	0.9787 +	0.9892 +	1.0051	0.1526 =	0.5243 –	0.1518
WFG8	3	318	0.3322 (0.0027) +	0.3173 (0.0055) –	0.3262 (0.0031)	1.0649 =	1.0567 +	1.0644	0.0454 –	0.0480 –	0.0420
WFG8	5	341	0.5079 (0.0059) –	0.5464 (0.0043) +	0.5356 (0.0104)	1.1013 –	1.0455 +	1.0810	0.0781 +	0.1192 –	0.0848
WFG8	10	413	0.6343 (0.0117) +	0.6801 (0.0061) +	0.5806 (0.0764)	1.1107 –	1.0205 +	1.0653	0.1341 =	0.2607 –	0.1340
WFG8	15	376	0.6742 (0.0598) +	0.0993 (0.0599) –	0.4206 (0.0499)	1.0351 +	0.9940 +	1.0814	0.2303 =	0.6197 –	0.2245
WFG9	3	317	0.3554 (0.0183) =	0.3282 (0.0025) –	0.3702 (0.0254)	1.0397 –	1.0400 =	1.0278	0.0315 =	0.0491 –	0.0338
WFG9	5	343	0.5385 (0.0058) –	0.5483 (0.0069) –	0.6009 (0.0247)	1.0461 –	1.0225 –	1.0166	0.0678 +	0.1167 –	0.0752
WFG9	10	373	0.6223 (0.0168) –	0.5929 (0.0249) –	0.6818 (0.0853)	1.0237 –	1.0055 =	1.0182	0.1272 =	0.2380 –	0.1317
WFG9	15	342	0.5920 (0.0326) +	0.1616 (0.0318) –	0.4930 (0.1115)	0.9817 +	0.9907 +	1.0094	0.2056 +	0.4346 –	0.2720
# –/=/+ →			**31/07/10**	**33/08/07**		**34/09/05**	**20/02/26**		**14/10/24**	**43/01/04**	

Table S.2: Performance of NSGA-III, MOEA/D-LWS (no UEA) and HFiDEA, using the mean (μ) of $d^{\parallel}$- and $d^{\perp}$-metrics, over 21 runs each. The results are corresponding to t_{TM} determined on-the-fly by the stability tracking algorithm with $\psi^{TM} = \{n_p, n_s\} = \{3, 50\}$. The symbols "–", "=", or "+" against NSGA-III and MOEA/D-LWS (here referred to as M-LWS for brevity) highlight where these are statistically worse than, comparable to or better than HFiDEA, respectively.

Problems	M	t_{TM}	$d^{\parallel}$-metric $[\mu(\sigma)]$			$d^{\perp}$-metric $[\mu(\sigma)]$		
			NSGA-III	MOEA/D-LWS	HFiDEA	NSGA-III	MOEA/D-LWS	HFiDEA
DTLZ1	3	588	4.8260 +	7.8504 =	6.2523	0.0441 –	0.0591 –	0.0382
DTLZ1	5	723	4.8164 =	10.5502 –	4.8159	0.0973 =	0.1344 –	0.0966
DTLZ1	10	736	4.8325 =	5.7263 =	5.6193	0.1273 =	0.2449 –	0.1241
DTLZ1	15	822	11.1991 –	3.1017 =	3.5548	0.0747 +	0.4309 –	0.0855
DTLZ2	3	495	1.0003 +	1.0013 –	1.0006	0.0004 +	0.0422 –	0.0288
DTLZ2	5	583	1.0029 –	0.9976 +	0.9984	0.0011 +	0.1225 –	0.0540
DTLZ2	10	657	1.0211 –	0.9978 –	0.9867	0.0081 +	0.2509 –	0.0924
DTLZ2	15	627	1.0254 –	1.0001 +	1.0027	0.0157 +	0.4065 –	0.0350
DTLZ3	3	577	6.3877 +	9.6251 =	10.6916	0.0642 –	0.0569 –	0.0376
DTLZ3	5	658	8.5759 +	17.5630 =	16.1149	0.1100 –	0.1340 –	0.1020
DTLZ3	10	706	11.6113 =	8.3821 =	9.7269	0.1520 –	0.2460 –	0.1296
DTLZ3	15	722	23.6650 –	3.5710 =	6.2156	0.0900 =	0.4290 –	0.0803
DTLZ4	3	876	1.0001 =	1.0012 –	1.0001	0.0530 –	0.0413 +	0.0431
DTLZ4	5	863	1.0005 –	0.9975 –	0.9965	0.0004 +	0.1224 –	0.0590
DTLZ4	10	791	1.0040 –	0.9977 –	0.9771	0.0067 +	0.2438 –	0.1062
DTLZ4	15	687	1.0006 –	1.0000 –	0.9788	0.0107 +	0.4073 –	0.1223
WFG1	3	967	0.8779 –	0.8622 –	0.8378	0.1443 –	0.1104 =	0.1100
WFG1	5	1126	0.7323 –	0.7250 –	0.6576	0.3069 –	0.2283 +	0.2476
WFG1	10	1427	0.6134 =	0.8631 –	0.6219	0.5287 –	0.4290 +	0.4988
WFG1	15	1582	0.4123 +	0.7070 –	0.4593	0.6720 –	0.5846 +	0.6554
WFG2	3	809	0.5497 =	0.5847 –	0.5526	0.0599 =	0.0733 –	0.0486
WFG2	5	948	0.3983 =	0.4801 –	0.3958	0.0389 +	0.1425 –	0.1155
WFG2	10	1017	0.2841 +	0.5131 –	0.3144	0.1281 +	0.3346 –	0.1585
WFG2	15	1019	0.1985 +	0.8750 –	0.2475	0.4716 =	0.6605 –	0.4518
WFG4	3	669	1.0096 –	1.0010 +	1.0041	0.0054 +	0.0401 –	0.0285
WFG4	5	795	1.0213 –	0.9937 +	0.9962	0.0175 +	0.1187 –	0.0603
WFG4	10	807	1.0262 –	0.9827 +	0.9896	0.0537 +	0.2518 –	0.0923
WFG4	15	774	1.0071 –	0.9923 +	1.0032	0.0879 –	0.5980 –	0.0692
WFG5	3	639	1.0242 –	1.0200 +	1.0210	0.0190 +	0.0419 –	0.0295
WFG5	5	726	1.0260 –	1.0097 +	1.0118	0.0285 +	0.1198 –	0.0615
WFG5	10	785	1.0222 –	0.9916 +	0.9942	0.0517 +	0.2175 –	0.0943
WFG5	15	753	1.0119 –	0.9970 +	1.0016	0.0686 +	0.4518 –	0.0718
WFG6	3	611	1.0254 –	1.0189 =	1.0198	0.0141 +	0.0423 –	0.0281
WFG6	5	690	1.0296 –	1.0084 =	1.0099	0.0238 +	0.1219 –	0.0584
WFG6	10	734	1.0245 –	0.9987 –	0.9940	0.0377 +	0.2433 –	0.0891
WFG6	15	703	1.0125 –	1.0013 –	0.9926	0.0440 +	0.4136 –	0.0929
WFG7	3	646	1.0044 –	1.0003 +	1.0012	0.0024 +	0.0410 –	0.0280
WFG7	5	735	1.0090 –	0.9951 –	0.9947	0.0084 +	0.1245 –	0.0567
WFG7	10	747	1.0212 –	0.9773 +	0.9812	0.0348 +	0.2325 –	0.0965
WFG7	15	706	1.0055 –	0.9939 =	0.9933	0.0624 +	0.5064 –	0.0676
WFG8	3	612	1.0565 +	1.0540 +	1.0590	0.0444 –	0.0484 –	0.0408
WFG8	5	666	1.0858 –	1.0420 +	1.0723	0.0727 +	0.1186 –	0.0798
WFG8	10	730	1.0874 –	1.0213 +	1.0564	0.1033 +	0.2569 –	0.1223
WFG8	15	693	1.0323 +	0.9962 +	1.0735	0.1835 =	0.6178 –	0.1888
WFG9	3	588	1.0367 –	1.0393 =	1.0251	0.0300 +	0.0490 –	0.0335
WFG9	5	645	1.0433 –	1.0219 –	1.0068	0.0599 +	0.1147 –	0.0716
WFG9	10	704	1.0251 –	1.0042 –	1.0003	0.1007 +	0.2357 –	0.1250
WFG9	15	651	1.0008 =	0.9949 +	1.0072	0.1524 +	0.4165 –	0.2266
# –/=/+ →			**31/08/09**	**20/11/17**		**11/06/31**	**43/01/04**	**of 48 probs.**

Table S.3: Performance of NSGA-III, MOEA/D-LWS and HFiDEA (incorporated with UEA), using the mean (μ) of $d^{\parallel}$- and $d^{\perp}$-metrics, over 21 runs each. The results are corresponding to $t_{TM} = 500$. The symbols "–", "=", or "+" against NSGA-III and MOEA/D-LWS (here referred to as M-LWS for brevity) highlight where these are statistically worse than, comparable to or better than HFiDEA, respectively.

Problems	M	$d^{\parallel}$-metric			$d^{\perp}$-metric		
		NSGA-III	MOEA/D-LWS	HFiDEA	NSGA-III	MOEA/D-LWS	HFiDEA
DTLZ1	3	10.8521 –	7.0685 +	8.5714	0.0427 =	0.0623 –	0.0445
DTLZ1	5	9.8476 +	11.2152 +	14.6389	0.0977 =	0.1446 –	0.0972
DTLZ1	8	19.5222 =	12.5645 +	19.9470	0.1236 =	0.2455 –	0.1155
DTLZ2	3	1.0003 +	1.0006 +	1.0014	0.0004 +	0.0348 –	0.0098
DTLZ2	5	1.0047 +	0.9978 +	1.0054	0.0013 +	0.1247 –	0.0278
DTLZ2	8	1.0251 –	0.9950 +	1.0005	0.0073 +	0.2268 –	0.0649
DTLZ3	3	14.0685 –	9.4161 +	11.8648	0.0909 –	0.0763 –	0.0519
DTLZ3	5	18.2880 +	36.9814 –	29.2571	0.1095 =	0.1568 –	0.1047
DTLZ3	8	37.0689 =	15.8981 +	32.0451	0.1215 =	0.2590 –	0.1188
DTLZ4	3	1.0005 +	1.0005 +	1.0012	0.0542 –	0.0540 –	0.0262
DTLZ4	5	1.0049 =	0.9972 +	1.0038	0.0014 +	0.1314 –	0.0290
DTLZ4	8	1.0154 –	0.9972 =	0.9972	0.0083 +	0.2392 –	0.0667
WFG1	3	0.9054 –	0.9329 –	0.8850	0.2139 –	0.1644 =	0.1252
WFG1	5	0.7837 –	0.7626 –	0.7189	0.4033 –	0.2723 +	0.2915
WFG1	8	0.7846 –	0.7122 –	0.5867	0.5401 –	0.4406 +	0.4866
WFG2	3	0.5535 =	0.5731 –	0.5535	0.0607 =	0.0731 –	0.0438
WFG2	5	0.4004 =	0.5325 –	0.4021	0.0521 +	0.1493 –	0.1054
WFG2	8	0.3057 +	0.5394 –	0.3431	0.1623 =	0.2666 =	0.2103
WFG4	3	1.0130 –	1.0015 +	1.0071	0.0074 +	0.0397 –	0.0148
WFG4	5	1.0303 –	0.9921 +	1.0053	0.0261 +	0.1154 –	0.0396
WFG4	8	1.0372 –	0.9799 +	1.0065	0.0687 +	0.2141 –	0.0761
WFG5	3	1.0260 –	1.0199 +	1.0227	0.0200 –	0.0412 –	0.0167
WFG5	5	1.0306 –	1.0094 +	1.0178	0.0341 +	0.1158 –	0.0401
WFG5	8	1.0315 –	0.9895 +	1.0078	0.0572 +	0.2028 –	0.0775
WFG6	3	1.0276 –	1.0178 +	1.0222	0.0152 +	0.0402 –	0.0163
WFG6	5	1.0347 –	1.0077 +	1.0173	0.0279 +	0.1225 –	0.0379
WFG6	8	1.0359 –	0.9925 +	1.0058	0.0439 +	0.2210 –	0.0733
WFG7	3	1.0061 –	0.9999 +	1.0029	0.0033 +	0.0395 –	0.0120
WFG7	5	1.0141 –	0.9940 +	1.0012	0.0130 +	0.1220 –	0.0307
WFG7	8	1.0301 –	0.9776 +	0.9951	0.0403 +	0.2153 –	0.0707
WFG8	3	1.0590 =	1.0487 +	1.0583	0.0441 –	0.0464 –	0.0429
WFG8	5	1.0979 –	1.0391 +	1.0804	0.0788 =	0.1182 –	0.0780
WFG8	8	1.1645 –	1.0255 +	1.0749	0.1522 –	0.2409 –	0.1222
WFG9	3	1.0374 –	1.0391 =	1.0266	0.0297 –	0.0477 –	0.0259
WFG9	5	1.0450 –	1.0219 –	1.0160	0.0627 –	0.1138 –	0.0585
WFG9	8	1.0363 –	1.0059 +	1.0158	0.1259 –	0.2171 –	0.1025
# –/=/+ →		**24/06/06**	**08/02/26**		**11/08/17**	**32/02/02**	**of 36 probs.**