Article:

Saxena, D.K., Sinha, A., Duro, J.A. orcid.org/0000-0002-7684-4707 et al. (1 more author) (2016) Entropy-based termination criterion for multiobjective evolutionary algorithms. IEEE Transactions on Evolutionary Computation, 20 (4). pp. 485-498. ISSN 1089-778X

https://doi.org/10.1109/TEVC.2015.2480780

Entropy based Termination Criterion for Multi-objective Evolutionary Algorithms

Dhish Kumar Saxena, Arnab Sinha, João A. Duro, and Qingfu Zhang

***Abstract*—Multi-objective Evolutionary Algorithms evolve a population of solutions through successive generations towards the Pareto-optimal Front. One of the most critical questions faced by the researchers and practitioners in this domain, relates to the number of generations that may be sufficient for an algorithm to offer a good approximation of the Pareto-optimal Front, for a given problem. Ironically, till date this question largely remains unanswered and the number of generations are arbitrarily fixed *a priori*, with potentially punitive implications. If the *a priori* fixed generations are insufficient, then the algorithm reports sub-optimal solutions. In contrast, if the *a priori* fixed generations are far–too-many, it implies waste of computational resources. This paper proposes a novel entropy based *dissimilarity* measure that helps identify *on-the-fly* the number of generations beyond which an algorithm *stabilizes*, implying that either a good approximation has been obtained, or that it can not be obtained due to the stagnation of the algorithm in the search space. Given that, in either case no further improvement in the approximation can be obtained, despite additional computational expense, the proposed *dissimilarity* measure provides a termination criterion and facilitates a termination detection algorithm. The *generality*, *on-the-fly* implementation, *low computational complexity*, and the demonstrated *efficacy* of the proposed termination detection algorithm, on a wide range of multi- and many-objective test problems, define the novel contribution of this paper.**

***Index Terms*—Evolutionary Multi-objective Optimization, Many-objective Optimization, Entropy, Termination Detection Algorithm.**

I. Introduction

AN Optimization problem characterized by M objectives (assuming minimization, without loss of generality), n variables, J-inequality and K-equality constraints, can be stated as:

$$\left.\begin{array}{lll} \text{Minimize} & \mathbf{F}(\mathbf{x}) = (f_1(\mathbf{x}), \ldots, f_M(\mathbf{x}))^\intercal, & \\ \text{subject to} & g_j(\mathbf{x}) \leq 0, & j = 1, \ldots, J; \\ & h_k(\mathbf{x}) = 0, & k = 1, \ldots, K; \\ \text{where} & \mathbf{x} = (x_1, x_2, \ldots, x_n)^\intercal, & \\ & x_i^L \leq x_i \leq x_i^U, & i = 1, \ldots, n. \end{array}\right\} \quad (1)$$

If $M = 1$, it is referred to as a single-objective problem (SOP), and if $M \geq 2$, it is referred to as a multi-objective optimization problem (MOP). If, in particular, $M \geq 4$, then the problem is seen as a special class of MOP, and referred to as a many-objective optimization problem (MaOP) [1]–[3]. In (1), the set of all $\mathbf{x} = (x_1, x_2, \ldots, x_n)^\intercal$ which satisfy each of the inequality and equality constraints define the *feasible decision (variable) space* Ω, and the corresponding $\mathbf{F}(\mathbf{x}) \in \mathbb{R}^M$ define the *attainable objective space*. If the objectives in an MOP are conflicting, then no single solution can simultaneously optimize all the objectives. Hence, the notion of optimality in the context of MOPs is associated with a set of solutions which offer different trade-offs among the objectives.

The best trade-offs among the objectives can be defined in terms of Pareto optimality, as explained below. A solution $\mathbf{x}$ is said to dominate another solution $\mathbf{x}'$, *iff* $\forall i \in \{1, \ldots, M\}$: $f_i(\mathbf{x}) \leq f_i(\mathbf{x}')$ and $\exists i \in \{1, \ldots, M\}$: $f_i(\mathbf{x}) < f_i(\mathbf{x}')$. The subset of Ω which contains the elements that are not dominated by any other element of Ω is referred as the Pareto-optimal set (Ω^*), while its image in the objective space is called the Pareto-optimal front (POF) [4].

For continuous problems, Ω^* and the POF usually contain an infinite number of solutions. However, from a practical perspective, finding only a finite set of solutions that are evenly distributed across the POF is considered sufficient. This is because: (i) generating the entire Pareto-optimal set can be computationally expensive or infeasible, and (ii) in real-world applications, the decision maker may only be interested in a representative set of Pareto-optimal solutions, towards picking the most preferred solution for implementation. Given this, the goal pragmatically reduces to finding a good POF–approximation, implying a set of solutions characterized by good *convergence* (solutions either lying on the true POF or being significantly close to it); and good *diversity* (solutions well distributed across the true POF).

Over the last two decades, Multi-objective Evolutionary Algorithms (MOEAs) have been effective tools for solving real-world MOPs [5]. Ironically, a critical question inevitably faced by every MOEA practitioner, as to how many generations a given MOEA is to be run to ensure a good POF–approximation (for a given problem), has been largely overlooked. In the absence of an adequate answer, the common practice is to arbitrarily fix the number of generations *a priori* or to run an MOEA till the available computational budget is exhausted. However, this approach has several pitfalls, in that:

- an MOEA may report sub-optimal solutions if the *a priori* fixed generations are insufficient, or the available computational budget is low. In contrast, far-too-many *a priori* fixed generations, or very high computational budget may lead to a waste of computational resources.

D. K. Saxena is with the Department of Mechanical & Industrial Engineering, Indian Institute of Technology, Roorkee, India (email: dhishfme@iitr.ac.in).

A. Sinha is with the Singapore University of Technology and Design, Singapore (email: arnab.sinha@gmail.com)

J. A. Duro is with the Department of Computer Science, Bath University, Bath, UK (email: j.a.duro@bath.ac.uk).

Q. Zhang is with the Department of Computer Science, City University of Hong Kong, Hong Kong; School of Computer Science and Electronic Engineering, University of Essex, Colchester, UK (email:qingfu.zhang@cityu.edu.hk; qzhang@essex.ac.uk).

- it is devoid of any inbuilt-intelligence for: (i) an intermittent or final *assessment* of how good is the obtained POF–approximation vis-*à*-vis the true POF, and (ii) prompting *corrective* measures, in case an MOEA gets stuck in a part of the POF or the search space excluding the POF.

Given the above, the distinctive contributions of this paper include:

- ideation of the notion of a *robust* termination criterion for MOEAs: recognizing that a termination criterion should be practically usable for any given MOEA and an optimization problem, further backed by the identified research gap (Section II), this paper ideates that a termination criterion be referred as *robust*, if it is: (i) *generic*, in the sense that it does not require an *a priori* knowledge of the POF, and neither depends on MOEA-specific operators, nor on the MOEA-related performance indicators, (ii) implementable *on-the-fly*, and (iii) *computationally efficient*, allowing *scalability* with the number of objectives.
- proposition of a *robust* termination criterion and a termination detection algorithm: an entropy based *dissimilarity* measure with a computational complexity that is linear in both the number of objectives and population size has been proposed, which helps identify *on-the-fly* the number of generations beyond which an MOEA *stabilizes* and fails to improve the quality of solutions any further. This provides a termination criterion, to implement which, a termination detection algorithm is formalized.
- extensive simulations and results: the efficacy of the proposed termination detection algorithm is demonstrated on MOPs and MaOPs, at a scale that is unprecedented in the existing literature on this issue. Here, the results are based on $1,440$ simulations, performed on 23 MOPs and 21 MaOPs. For each problem, the simulations correspond to three different settings of an algorithmic parameter (n_p, explained later), and 10 different POF–approximations obtained by an MOEA. While NSGA-II is used as the underlying MOEA for each of the 44 test cases, ϵ-MOEA is used for four MaOPs.

The remaining of this paper is organized as follows: Section II reviews the existing literature on termination algorithms for MOEAs. The formative concepts for this paper relating to entropy, relative entropy, and probability distribution estimation are presented in Section III, while their implementation in the context of MOEAs is discussed in Section IV. A *dissimilarity* measure is proposed in Section V, and the resulting MOEA termination detection algorithm is presented in Section VI. The chosen test problems and associated experimental settings are discussed in Section VII. The experimental results are discussed in Section VIII, the potential future directions in Section IX, while the paper concludes with Section X.

II. Past Research on Termination Criterion for Multi-objective Evolutionary Algorithms

The quest for a termination algorithm for MOEAs has gained prominence in the last decade or so, following some earlier research [6]–[9] on the conditions for convergence for MOEAs, and emphasis on the significance of this issue in [10].

The existing termination algorithms are highlighted below.

1) An *online* termination algorithm has been proposed in [11], where NSGA-II [12] is terminated when the variation of the maximal crowding distance mean gets below a user-defined threshold (interpreted as stabilization of the population).
2) The algorithm in [13] terminates an MOEA when the ratio of dominated solutions in two consecutive generations stabilizes, and is demonstrated on NSGA-II and SPEA2 [14], and on PESA [15] in [16]. Its variant in [17] proposes termination when the variance of the normalized objective values stabilizes, and is demonstrated on NSGA-II, SPEA2, and HypE [18].
3) The *offline* convergence detection algorithm in [19], gathers populations from a series of MOEA runs, uses Kolmogorov-Smirnov statistical test [20] to gauge the changes in a set of performance indicators [21] (generational distance, hypervolume, and spread indicator), and infer about convergence and termination. Its *online* implementation in [22] replaces the Kolmogorov-Smirnov statistical test by the χ^2-variance test and t-test [20]. The above *offline* and *online* implementations have also been compared in [23].
4) The criterion in [24] detects convergence and calls for an MOEA's termination based on the adjustment of the indicator values (like hypervolume, ϵ-indicator, and mutual-dominance-rate (ratio of dominated solutions)) to a uniform model, computed through the least squares approximation and slope of the model. Its utility has been demonstrated on NSGA-II, SPEA2 and PESA.
5) The criterion in [25] computes the ratio of the number of members in a preceding archive (of non-dominated solutions) that are retained in the current archive, to the size of the current archive. When the improvement in this ratio over successive generations falls below a threshold, the MOEA's termination is recommended.
6) The criterion in [26] defines a stability score for each solution, as the number of solutions in the neighborhood (defined in the variable space) that dominate it. An MOEA is terminated, if the stability score for the population (average of all solutions) tends to zero.
7) In [27], the Karush-Kuhn-Tucker (KKT) [28] optimality conditions are utilized, a KKT-proximity measure (shown to reduce sequentially to zero as the iterates approach the KKT point) is proposed, which in turn guides MOEA termination.

It may be noted that most of the approaches discussed above suffer from one or more of the following limitations:

- reliance on MOEA-specific operators which may adversely affect the *generality* of the termination algorithms: for instance, the *online* termination algorithm proposed in [11] depends on the crowding distance operator which is used in NSGA-II but not in many other MOEAs.
- reliance on MOEA-related performance indicators which could lead to *premature* termination: for instance, the

termination algorithm in [13] relies on utilizing the ratio of dominated solutions in a given population. Notably, in the case of MaOPs, it has been well reported in literature that almost the entire population becomes non-dominated very early in the evolutionary search [29], causing the loss of selection pressure for convergence to the POF. While it may be logical to terminate an MOEA when the ratio of dominated solutions stabilizes (low potential for a good POF–approximation), it could still imply *premature* termination in terms of the POF–representation [30]. In that: (i) an MOEA termination guided by the ratio of dominated solutions may lead to a poor POF–approximation and also poor POF–representation [30], and (ii) if the MOEA is allowed to run beyond the inferred termination, significant improvements in the POF–representation could be achieved, which paves way for effective objective-reduction [30]–[32] in MaOPs.
Similarly, the algorithm based on the variance of normalized objective values [17] could lead to *premature* termination, because: (i) even for solution sets with largely disparate in-range distribution, the variance may become comparable, early in the evolutionary search process, and (ii) if the underlying MOEA is allowed to run further (beyond the inferred termination), even the in-range distribution of solutions may become comparable (besides the variance), for populations in successive generations.
- paradoxical requirement of *a priori* knowledge of the POF: for instance, the termination algorithm in [19] utilizes an indicator like generational distance, which can not be computed unless the POF is known *a priori*.
- poor scalability with the number of objectives: the termination algorithms based on generational distance [19] and hypervolume [19], [24] may not be usable for MaOPs, since the computational cost of computing these indicators grows exponentially with the number of objectives.

Hence, it is fair to infer that the impelling need for a *robust* termination criterion largely remains unfulfilled, implying a critical research gap which this paper aims to bridge.

III. Formative concepts: Entropy, Relative Entropy, and Probability Distribution Estimation

The formative concepts for this paper are presented below.

In information theory, *entropy* as a concept introduced by Shannon [33] measures the uncertainty associated with the prediction of the outcome of a random variable, which is equivalent to its information content (with the opposite sign). Let X be a discrete random variable with cardinality T (each element, given by x_i) and probability distribution given by $p(X)$. Then, entropy can be defined as,

$$\mathcal{H}(X) = -\sum_{i=1}^{T} p(x_i) \log p(x_i), \tag{2}$$

Notably, $\mathcal{H}(X)$ only quantifies the information within the probability distribution $p(x_i)$. For comparing two different distributions, a concept known as *relative entropy* (also known as Kullback–Leibler divergence [34]) quantifies how close a probability distribution $p(x_i)$ is to a model (or candidate) distribution $q(x_i)$. It can be used as a *dissimilarity* measure between two stochastic processes. For a discrete domain, this measure can be expressed as:

$$\mathcal{KL}(p\|q) = -\sum_{i=1}^{T} p(x_i) \log \left\{ \frac{q(x_i)}{p(x_i)} \right\}. \tag{3}$$

Notably, the following characteristics hold true for $\mathcal{KL}(p\|q)$: (i) is always non-negative, i.e., $\mathcal{KL}(p\|q) \geq 0$; (ii) it is not symmetric since $\mathcal{KL}(p\|q) \neq \mathcal{KL}(q\|p)$; and (iii) only in a case when $p(X) = q(X)$ then $\mathcal{KL}(p\|q) = \mathcal{KL}(q\|p)$.

Towards computing entropy and relative entropy, the probability distribution $p(x_i)$ and $q(x_i)$ may be estimated using parametric, semi-parametric, or non-parametric methods [35]. The non-parametric estimation methods are most flexible since they do not make assumptions about the probability distribution *functional form*, and the density estimation is entirely data-driven. Two most commonly used methods in this category are multivariate histograms [36] and kernel density estimation [37], [38]. The former being a simple and fast method, is utilized in this work and is discussed below.

Given a D-dimensional space, the multidimensional histogram method partitions each of the D dimensions into a fixed number of intervals (n_b), defined by an anchor point (often, the origin) and bin widths for each dimension, namely, $h_1, \ldots, h_D$. These partitions result in ${n_b}^D$ number of cells, jointly constituting a hyperrectangle with its hypervolume given by $\prod_{j=1}^{D} h_j$. The probability distribution function associated with cell x_i is given by

$$p(x_i) = \frac{k(x_i)}{\hat{N}}, \tag{4}$$

where $\hat{N}$ represents the total number of data points, and $k(x_i)$ denotes the number of data points that exist in the cell x_i. Clearly, with an increase in the number of dimensions (D), the number of cells grow exponentially (${n_b}^D$), and hence, fixing n_b (on which the smoothing of the probability distribution depends) is a major challenge associated with this method.

IV. Multidimensional Histogram Algorithm for MOEA Populations

This paper, as pointed out earlier, aims to develop a termination criterion for MOEAs, by proposing a *dissimilarity* measure capable of detecting their stabilization. This measure (proposed later) utilizes the concept of *relative entropy*, wherein, the two distributions to-be-compared correspond to MOEA populations (objective vectors of the feasible non-dominated solutions) in two successive generations, say, P and Q. Their probability distributions, say, p and q, respectively, are computed by using the multidimensional histogram method, whose complexity grows exponentially with the dimension D that needs to be partitioned. In the current context, D corresponds to the number of objectives (M), which could be quite large for a given optimization problem. Recognizing the challenge of complexity, this section proposes a novel implementation of the multidimensional histogram method, such that the associated complexity reduces to $O(N \times M)$ (quantification in Section VI-B), where N represents the population size used by an MOEA, i.e., the size of each, P and Q.

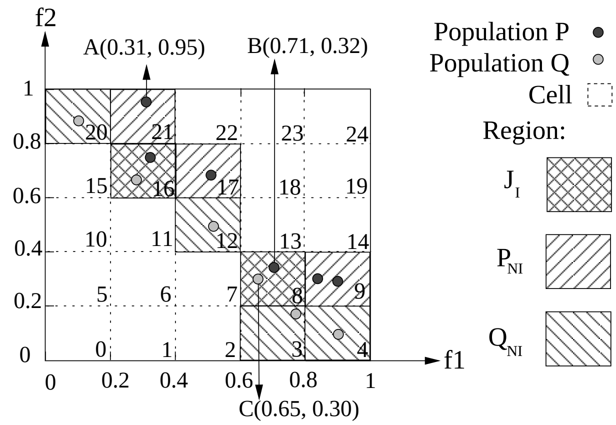


Fig. 1: Highlighting the effect of population dispersion on the number of cells required for partitioning, and assignment of a unique cell identification number to each cell. The numbers 0-24 inside the cells represent the cell numbers.

A. Computationally Efficient Data Structures

Towards a computationally efficient partitioning of the M-dimensional objective space, it is assumed that all the bins have the same width, and the use of following data structures is proposed:

- J_I: a set of cells representing the intersection region for P and Q, where each cell satisfies $p(x_i) > 0$ & $q(x_i) > 0$
- P_{NI}: a set of cells representing the non-intersection region, where each cell satisfies $p(x_i) > 0$ and $q(x_i) = 0$
- Q_{NI}: a set of cells representing the non-intersection region, where each cell satisfies $p(x_i) = 0$ and $q(x_i) > 0$
- C: a vector that stores cells from regions J_I and P_{NI};
- C_q: a vector that stores cells from region Q_{NI}.
- P_c and Q_c: vectors that store the number of solutions in each cell of C, contributed by P and Q, respectively.
- Q_{cq}: a vector that stores the number of solutions in each cell of C_q, contributed by Q.

For a sample illustration, consider P and Q comprising of six solutions each, as shown in Figure 1. Notably:

- $|J_I| = 2$; $|P_{NI}| = 3$; $|Q_{NI}| = 4$; $|C| = 5$; $|C_q| = 4$
- $C \equiv$ {8,9,16,17,21}; $C_q \equiv$ {3,4,12,20}
- $P_c \equiv$ {1,2,1,1,1}; $Q_c \equiv$ {1,0,1,0,0}; $Q_{cq} \equiv$ {1,1,1,1}.

Interestingly, a total of only nine cells are required to partition P and Q combined, as opposed to 25 (n_b^2). This reflects on to the utility of the proposed data structures.

B. Assignment of a Unique Identification Number to a Cell

To enable a cell's storage in C or C_q, it needs to be assigned a unique identification number. Let $s = \{s_j; j = 1, \ldots, M\}$ be a solution either from population P or Q. Then the procedure for assignment of a unique identification number for a cell occupying s, is as follows:

- let $O_{max,j}$ and $O_{min,j}$ define the maximum and minimum values, respectively, for the j^{th} objective, amongst all solutions in the combined population, namely $P \cup Q$
- let a function named $GetCell\_id$ return a value (c), to help identify the cell that a solution belongs to, and then:

1) map the solution into the range $[0, 1]$ by:

$$\bar{s}_j = \frac{s_j - O_{min,j}}{O_{max,j} - O_{min,j}}, \text{ for } j = 1, \ldots, M; \quad (5)$$

2) let a vector $B = \left\{\frac{0}{n_b}, \frac{1}{n_b}, \ldots, \frac{n_b}{n_b}\right\}$ with size $n_b + 1$ define a set of intervals such that:

$$B_{k_j} \leq \bar{s}_j \leq B_{k_j+1}, \; k_j \in [0, \ldots, n_b - 1] \quad (6)$$

3) determine the unique cell identification number, as:

$$c = \sum_{j=1}^{M} k_j \times n_b^{j-1}. \quad (7)$$

For illustration, consider the solutions A, B, and C in Figure 1, where $M = 2$ and $n_b = 5$. The cell identification number for A ($k = \{1, 4\}$), is given by $c = 21$, while for B and C ($k = \{3, 1\}$), $c = 8$, implying that solutions belonging to the same cell have the same cell identification number.

C. Multidimensional Histogram Algorithm: General Steps

The general steps of the multidimensional histogram algorithm (Algorithm 1) are as follows.

1) Find a cell corresponding to each solution in P, by using $GetCell\_id$ function (described in Section IV-B). If the cell already exists in vector C, increment by one the element in the vector P_c that corresponds to the identified cell. Otherwise, proceed as follows:
   a) add the cell to vector C, to keep a track of all found cells for population P;
   b) update the vector P_c for the solution found, by initializing the corresponding cell position as 1;
   c) for the same position, initialize vector Q_c with the value 0 (no solution from population Q exists).

   Here (Steps 2-12), the number of solutions from P that fall into each cell are counted. Each cell is stored in C, while the number of solutions are stored in P_c. The corresponding position in Q_c is initialized as 0.
2) Find a cell corresponding to each solution in Q, by using $GetCell\_id$ function. If the cell already exists in vector C, increment the corresponding position in vector Q_c. Otherwise, proceed as follows:
   a) If the cell exists in vector C_q, increment by one the element in the vector Q_{cq}.
   b) Else: (i) add the cell to vector C_q to keep a track of all found cells for population Q that were not found for population P and (ii) initialize with value 1 the corresponding position in vector Q_{cq}.

   Here (Steps 13-27), the number of solutions from Q that fall into each cell are counted, by distinguishing between: (i) the cells already occupied by population P (corresponding to vector C and counted by vector Q_c) and (ii) cells only occupied by population Q (corresponding to vector C_q and counted by vector Q_{cq}).

To summarize, Algorithm 1 ensures that for P and Q, the number of solutions that exists in each cell are stored in vectors P_c, Q_c and Q_{cq}. This allows for computation of the probability distribution associated with each cell (Equation 4), a step that is directly implemented in Algorithm 2 (Section VI).

Algorithm 1: Multidimensional Histogram Algorithm for two MOEA populations.

Input:
P: feasible and non-dominated population corresponding to instant t
Q: feasible and non-dominated population corresponding to instant $t+1$
n_b: number of bins used to partition the search space equally among all objectives
Output:
C: vector that stores cells from regions J_I and P_{NI}
C_q: vector that stores cells from region Q_{NI}
P_c: vector that stores the number of solutions within each cell of C for population P
Q_c: vector that stores the number of solutions within each cell of C for population Q
Q_{cq}: vector that stores the number of solutions within each cell of C_q for population Q

begin
    for *each* s *in* P **do**
        $c \leftarrow GetCell\_id(s, n_b)$
        if c *exist in* C **then**
            k = {index position of c in vector C}
            $P_{c,k} = P_{c,k} + 1$ /* increment $P_{c,k}$ by 1 */
        else
            $C = \{C, c\}$ /* concatenate c at the end of C */
            $P_c = \{P_c, 1\}$ /* concatenate 1 at the end of P_c */
            $Q_c = \{Q_c, 0\}$ /* concatenate 0 at the end of Q_c */
        end
    end
    for *each* s *in* Q **do**
        $c \leftarrow GetCell\_id(s, n_b)$
        if c *exist in* C **then**
            k = {index position of c in vector C}
            $Q_{c,k} = Q_{c,k} + 1$ /* increment $Q_{c,k}$ by 1 */
        else
            if c *exist in* C_q **then**
                k = {index position of c in vector C_q}
                $Q_{cq,k} = Q_{cq,k} + 1$ /* increment $Q_{cq,k}$ by 1 */
            else
                $C_q = \{C_q, c\}$ /* concatenate c at the end of C_q */
                $Q_{cq} = \{Q_{cq}, 1\}$ /* concatenate 1 at the end of Q_{cq} */
            end
        end
    end
end

V. Proposed Dissimilarity Measure

This section proposes a *dissimilarity* measure that helps identify *on-the-fly* the number of generations beyond which an MOEA could be considered to have stabilized, thereby, providing a criterion for its termination.

For a cell $x_i \in J_I$ (intersection set), solutions from both P and Q exist, and the *dissimilarity* measure, namely $\mathcal{D}(p,q)_{\mathcal{I}}$, can be given by

$$\mathcal{D}(p,q)_{\mathcal{I}} = \mathcal{KL}(p\|q) + \mathcal{KL}(q\|p), \text{ where} \tag{8}$$

$$\mathcal{KL}(p\|q) = -\sum_{x_i \in J_I} \frac{p(x_i)}{2} \log \left\{ \frac{q(x_i)}{p(x_i)} \right\}, \tag{9}$$

$$\mathcal{KL}(q\|p) = -\sum_{x_i \in J_I} \frac{q(x_i)}{2} \log \left\{ \frac{p(x_i)}{q(x_i)} \right\} \tag{10}$$

Furthermore, if $\mathcal{D}(p,q)_{\mathcal{Y}_P}$ and $\mathcal{D}(p,q)_{\mathcal{Y}_Q}$ are to represent the *dissimilarity* measures for the cells $x_i \in P_{NI}$ and $x_i \in Q_{NI}$, respectively, the combined *dissimilarity* measure for the non-intersection sets, namely $\mathcal{D}(p,q)_{\mathcal{Y}}$, can be given by

$$\mathcal{D}(p,q)_{\mathcal{Y}} = \mathcal{D}(p,q)_{\mathcal{Y}_P} + \mathcal{D}(p,q)_{\mathcal{Y}_Q}, \text{ where} \tag{11}$$

$$\mathcal{D}(p,q)_{\mathcal{Y}_P} = -\sum_{x_i \in P_{NI}} \frac{p(x_i)}{2} \log p(x_i), \tag{12}$$

$$\mathcal{D}(p,q)_{\mathcal{Y}_Q} = -\sum_{x_i \in Q_{NI}} \frac{q(x_i)}{2} \log q(x_i) \tag{13}$$

Finally, the *dissimilarity* measure between two MOEA populations, denoted by $\mathcal{D}(p,q)$, can be defined by Equation 14.

$$\mathcal{D}(p,q) = \mathcal{D}(p,q)_{\mathcal{I}} + \mathcal{D}(p,q)_{\mathcal{Y}}. \tag{14}$$

Notably, $\mathcal{D}(p,q)$ has the following characteristics:

- it is non-negative: $\mathcal{D}(p,q) \geq 0$
- it is symmetric: $\mathcal{D}(p,q) = \mathcal{D}(q,p)$
- if $p(X) = q(X)\ \forall X$ (implying $p(x_i) = q(x_i)\ \forall x_i \in J_I$, and $P_{NI} = Q_{NI} = \emptyset$), then $\mathcal{D}(p,q) = 0$.
- if $p(X) \neq q(X)\ \forall X$, then the magnitude of $\mathcal{D}(p,q)$ can not be gauged, except that it will grow as more and more data points fall into non-intersection regions.

In the current context, where the data sets P and Q represent MOEA populations in successive generations, the $\mathcal{D}(p,q)$ measure may be high during the initial generations, due to significantly dissimilar populations. Subsequently, it may:

(I) retain high and largely varying values, if MOEA populations keep traversing in the non-POF objective space
(II) retain high but almost similar values, if MOEA populations stagnate in the non-POF objective space
(III) achieve a zero or near-zero value when MOEA populations over successive generations become largely similar. This may occur in scenarios where an MOEA:
  a) offers a good POF–approximation (good convergence to & good distribution across the POF)
  b) stagnates in a part of the true POF (good convergence to & poor distribution across the POF)

These characteristics of the *dissimilarity* measure, promise to counter some of the pitfalls in fixing of the number of MOEA-generations *a priori* or running it till the available computational budget is exhausted. In that:

- an MOEA termination based on population stabilization is not arbitrary, and the likelihood of sub-optimal solutions or wastage of computational resources, reduces.
- a zero or near-zero value of $\mathcal{D}(p,q)$ could be treated as a necessary condition for a good POF–approximation. Clearly, if the necessary condition is:
  – violated (Items (I) and (II) above), then the lack of good POF–approximation could be guaranteed.
  – met (Item (III) above), then it *provides clues* to two distinct possibilities on the quality of POF–approximation, following which either the MOEA could be terminated, or it could be treated as an indicator of the right timing for MOEA parameters to be *explored/corrected* (more in Section IX).

Given the state-of-the-art, where most existing MOEAs:

- provide a good POF–approximation in the case of two- and three-objective problems, the proposed *dissimilarity* measure may achieve a zero or near-zero value.
- fail in providing a good POF–approximation in the case of MaOPs, the proposed *dissimilarity* measure may retain high values.

In either case, the underlying MOEA could be terminated because the quality of POF–approximation can not be further improved, despite additional computational expense.

VI. MOEA Termination Detection Algorithm

This section proposes an MOEA termination detection algorithm (Algorithm 2), by integrating: (a) probability density estimation by a multidimensional histogram algorithm (Algorithm 1), (b) computation of the *dissimilarity* measure (Equation 14), and (c) a termination criterion based on the mean and standard deviation of the *dissimilarity* measures. While the items (a) and (b) have previously been discussed in detail, the termination criterion (item (c)) is introduced below.

A. MOEA Termination Criterion

MOEAs evolve a randomly initialized population through several iterations (generations) of the *variation* (such as crossover and mutation) and *selection* operators, aiming to arrive at the POF. MOEAs are stochastic in nature, and hence, it is fair to expect variations in the *dissimilarity* measures in successive MOEA generations. In general, for a given problem (difficulty influenced by the number and nature of objectives, constraints, and variables), and pre-specified MOEA-parameter settings, the extent of variation may depend on the advancement of MOEA population (articulated in Items (I)–(III), Section V) vis-$\grave{a}$-vis the true POF.

Let i denote the generation-counter, t denote the current generation, $\mathcal{D}_i$ denote the $\mathcal{D}(p,q)$ value at i^{th} generation; and M_t & S_t denote the mean & standard deviation of $\mathcal{D}(p,q)$ measures from the first to the t^{th} generation, as given by Equations 15 and 16[1], respectively.

$$M_1 = \mathcal{D}_1 \text{ and } M_t = \frac{1}{t}\sum_{i=1}^{t} \mathcal{D}_i, \text{ where } t \geq 2 \tag{15}$$

$$S_t = \frac{1}{t}\sum_{i=1}^{t} (\mathcal{D}_i - M_t)^2 \tag{16}$$

Towards a robust algorithm for termination detection (Algorithm 2), it is proposed that when the mean and standard deviation of the *dissimilarity* measures in a pre-specified number of successive generations (n_s) of the MOEA may coincide–up to a pre-specified number of decimal places (n_p), then the underlying MOEA be terminated, and the last generation be reported as N_{gt}. Its rationale is that, if n_p and n_s are reasonably large numbers (contextually, say, $n_p \geq 2$ and $n_s \geq 20$), then the n_p–conformance between n_s successive means and standard deviations, respectively, becomes a stringent criterion which:

- may not be met by a set of randomly generated solutions, or even MOEA populations successively traversing in the non-POF objective space (Item (I), Section V)
- may be met in cases where the MOEA stabilizes (Items (II) and (III), Section V), and the perturbations in $\mathcal{D}(p,q)$ values over different generations may not be strong enough to violate the requirement of n_p conformance.

Algorithm 2: MOEA Termination Detection Algorithm

Input:
n_s: the number of successive generations of an MOEA for which the mean and standard deviation of the *dissimilarity* measures are to be compared
n_p: the number of decimal places to which the mean and standard deviation of the *dissimilarity* measures are to be compared
n_b: number of bins for the multidimensional histogram
$t = 1$, $c_1 = false$ and $c_2 = false$.

1 **begin**
2 Generate a population of feasible non-dominated solutions randomly and let this population be denoted by P.
3 Run an MOEA for one generation, using P as initial, and generate a new feasible non-dominated population. Let this new population be denoted by Q.
4 $(C, C_q, P_c, Q_c, Q_{cq}) \leftarrow MultiHistogram(P, Q, n_b)$ /* (Algorithm 1) */
5 $\mathcal{D}_t = 0$ /* Initialize the *dissimilarity* measure at instant t */
6 **for** *each* i *in* C **do**
7 $p = P_{c,i}/|P|$ /* $p(x_i)$, $x_i \in J_I \cup P_{NI}$ */
8 $q = Q_{c,i}/|Q|$ /* $q(x_i)$, $x_i \in J_I \cup P_{NI}$ */
9 **if** $q > 0$ **then** /* Relative entropy */
10 $\mathcal{D}_t = \mathcal{D}_t - \left[\left(\frac{p}{2}\log\frac{p}{q}\right) + \left(\frac{q}{2}\log\frac{p}{q}\right)\right]$ /* (Equation 8) */
11 **else if** $q = 0$ **then** /* Entropy of P */
12 $\mathcal{D}_t = \mathcal{D}_t - p\log p$ /* (Equation 12) */
13 **end**
14 **end**
15 **for** *each* i *in* C_q **do** /* Entropy of Q */
16 $q = Q_{cq,i}/|Q|$ /* $q(x_i)$, $x_i \in Q_{NI}$ */
17 $\mathcal{D}_t = \mathcal{D}_t - q\log q$ /* (Equation 13) */
18 **end**
19 Determine M_t and S_t /* Equations 15 and 16 */
20 $\hat{M}_t = Round(M_t, n_p)$ /* Round M_t to the ${n_p}^{th}$ decimal place */
21 $\hat{S}_t = Round(S_t, n_p)$ /* Round S_t to the ${n_p}^{th}$ decimal place */
22 **if** $t > n_s$ **then**
23 **if** $[\hat{M}_t = \hat{M}_{t-1} = \ldots = \hat{M}_{t-n_s}]$ **then** $c_1 = true$
24 **if** $[\hat{S}_t = \hat{S}_{t-1} = \ldots = \hat{S}_{t-n_s}]$ **then** $c_2 = true$
25 **end**
26 **if** $c_1 = true$ *and* $c_2 = true$ **then**
27 Report Q as the final population and set $N_{gt} = t$
28 Terminate the run.
29 **else**
30 Set $t = t + 1$, $c_1 = false$, $c_2 = false$, $P = Q$ and go to step 3.
31 **end**
32 **end**

The steps of the Algorithm 2, are as described below.

1) (step 2) Generate a population (P) of feasible non-dominated solutions randomly.
2) (step 3) With P as the initial population, run an MOEA for one generation to get a new population (Q) of feasible non-dominated solutions.
3) (step 4) Using n_b and populations P and Q as input,

[1]For the randomly initialized population ($i = 0$), $\mathcal{D}(p,q)$ can not be computed because there is no other population available. For $i = 1$, two populations are available (the randomly generated (p) and one evolved by an MOEA (q)), hence $\mathcal{D}(p,q)$ can be computed, but that being the only *dissimilarity* measure available, implies $M_1 = \mathcal{D}_1$ and $S_1 = 0$.

run Algorithm 1 and obtain the vectors, namely, C, C_q, P_c, Q_c, and Q_{cq}.

4) (step 5) Initialize the *dissimilarity measure* value as zero, $\mathcal{D}_t = 0$ (for each generation, t).
5) Based on the computed vectors C, C_q, P_c, Q_c, and Q_{cq}:
   a) (steps 6-13) For each cell in C: compute the probability distributions associated with P (p, step 7) and Q (q, step 8), and the *dissimilarity* measure using Equation 8 or 12 depending on a cell's membership in J_I or P_{NI}, and update $\mathcal{D}_t$.
   b) (steps 15-18) For each cell in C_q: compute the probability distribution associated with Q (q, step 16), and the *dissimilarity* measure using Equation 13, and update $\mathcal{D}_t$.
6) (steps 19-21) Compute the mean and standard deviation (M_t and S_t, Equation 15 and 16) of $\mathcal{D}_t$, and round-off both to the ${n_p}^{th}$ decimal place to determine $\hat{M}_t$ and $\hat{S}_t$.
7) (steps 22-31) If the number of generations (t) exceed n_s, terminate the MOEA and declare $N_{gt} = t$, if: (a) the values between $\hat{M}_t$ and $\hat{M}_{t-n_s}$ are equal ($c_1 = true$), *and* (b) the values between $\hat{S}_t$ and $\hat{S}_{t-n_s}$ are equal ($c_2 = true$). Else, set $P = Q$ and return to Step 2.

B. Computational Complexity of the Proposed Algorithm

The computational complexity of the proposed MOEA termination detection Algorithm 2 is *linear* in the number of objectives (M) and the population size (N). This claim can be justified, as follows. The computational complexity of Algorithm 2 is linked to the following:

- evaluation of function *GetCell_id* (Section IV-B): this function has a computational complexity of $O(M)$, independent of n_b. This is because: (i) the integer rounding of the term $n_b \times \bar{s}_j$ will simply provide k_j satisfying Equation 6, and (ii) subsequent evaluation of the unique cell id is based only on multiplication and addition operations over the variable $j = 1 \dots M$ (Equation 7).
- multidimensional histogram for two population sets: here, the Algorithm 1 calls the function *GetCell_id* for each and every population member, implying the computational complexity of $O(N \times M)$, by this step.
- the three stages in Algorithm 2, including:
  - a call to Algorithm 1 ($O(N \times M)$).
  - evaluation of the dissimilarity measure (Equation 8): this has a dynamic computational complexity of $O(K)$, where K is the sum of cardinality of the sets C and C_q. Notably, K cannot be greater than $2 \times N$ (combined size of the parent and child populations). Hence, the worst case complexity could be $O(N)$.
  - computation of M_t and S_t (Equations 15 and 16): these computations have a computational complexity of $O(1)$, hence, can be neglected.

To summarize, the total worst case computational complexity of Algorithm 2 is $O(N \times M + N) = O(N \times M)$.

VII. Test Problems and Experimental Settings

The effectiveness of the Algorithm 2 is investigated against a wide range of test problems with varying difficulty levels, including, CTP1–8 [39]; ZDT1–6 [39]; SCH1–2, POL, KUR, FON, BNH, SRN, TNK, OSY [39]; DTLZ1–4 [40]; DTLZ5(I, M) [30] (more details on these problems are provided in the supplementary file attached with this paper). While NSGA-II has been chosen as the underlying MOEA for all the test problems, ϵ-MOEA [41] has been used only for a few sample cases (as justified in Section VIII-B). These MOEAs share a common set of parameters, which are set as below. With a population size of 200; the probability of crossover and mutation used is 0.9 and 0.1, respectively; while the distribution index for crossover and mutation is chosen as 5 and 20, respectively. An additional parameter associated with ϵ-MOEA is ϵ, which is set as $\epsilon_i = 0.3,\ \forall\ i = 1, \dots, M$.

Notably, besides the two population sets, three other inputs are required by the Algorithm 2, including, n_b, n_p and n_s. As argued in Section VI-A, n_p and n_s need to be reasonably large numbers, where their largeness is to be gauged in the context of the role they play towards termination detection. With regard to n_b, efforts have been made to determine its suitable value [42]–[45] in the general context of probability density estimation. However, in this work such approaches may not be best suited, for the following reasons:

- estimation of an optimal n_b value is a computationally expensive task, more-so, in the current context where either or both the number of objectives and population size may be large.
- in general, the data points may be scattered all over the search space. However, in the current context, the data points are non-dominated solutions capturing the trade-off between the objectives, and hence, may occupy only a small fraction of the total number of cells determined by n_b (as in Figure 1). Furthermore, an optimal value of n_b with regard to the probability density estimation may not necessarily be optimal with regard to estimation of the difference in the density functions of different data sets, as is the case with the proposed *dissimilarity* measure.

Given the above considerations, simulations for each problem are performed corresponding to 10 different solution sets obtained from an MOEA (NSGA-II or ϵ-MOEA), and the parameters being $n_b = 10$; $n_s = 20$; and $n_p = 2, 3$ and 4.

VIII. Experimental Results

This section reports the performance of the Algorithm 2 on the considered test-suite, as below:

(a) the N_{gt} determined by Algorithm 2 (against $n_p = 2, 3, 4$), and the corresponding hypervolume[2] measures have been tabled.

(b) the POF–approximations (by the underlying MOEA) at N_{gt} against $n_p = 2, 3, 4$ are simultaneously plotted to facilitate a visual comparison between them. It may be

[2]The hypervolume computation is done by a dimension-sweep algorithm [46], with $O(N^{M-2} \log N)$ time and linear space complexity (M and N denoting the number of objectives and population size, respectively). Towards it, the source codes at: http://iridia.ulb.ac.be/∼manuel/hypervolume, are used. The reference points used, are based on the maximum values of the objective functions across the population, and are reported in the supplementary file attached with this paper.

TABLE I: NSGA-II based N_{gt} (against $n_p = 2, 3, 4$, respectively) & $\mathcal{D}(p, q)$ measures by Algorithm 2 for a range of multi-objective problems, and the corresponding hypervolume measures.

The N_{gt} and $\mathcal{D}(p, q)$ results are formatted as $\mu \pm \sigma$, where μ and σ represent the mean and standard deviation of the respective entry, over 10 different NSGA-II runs. The reported hypervolume measures correspond to only one (fifth of the 10) NSGA-II run.

	Generations N_{gt}			Dissimilarity Measure $\mathcal{D}(p, q)$			Hypervolume		
Problem	$n_p = 2$	$n_p = 3$	$n_p = 4$	$n_p = 2$	$n_p = 3$	$n_p = 4$	$n_p = 2$	$n_p = 3$	$n_p = 4$
CTP1	150±27	595±148	2399±691	0.035±0.001	0.019±0.001	0.016±0.000	1.306054	1.306284	1.306262
CTP2	133±20	539±71	2316±430	0.036±0.004	0.013±0.001	0.007±0.000	1.346923	1.347202	1.347365
CTP3	176±33	1025±151	6376±830	0.118±0.020	0.049±0.006	0.024±0.001	1.317466	1.327290	1.330874
CTP4	232±38	1314±119	8505±1159	0.144±0.023	0.066±0.010	0.030±0.003	1.217796	1.281876	1.314137
CTP5	160±18	578±40	2672±174	0.044±0.005	0.018±0.001	0.008±0.000	1.311780	1.317208	1.324180
CTP6	149±22	592±122	2259±419	0.033±0.003	0.017±0.001	0.013±0.000	8.702711	8.702202	8.703030
CTP7	196±21	820±74	3625±405	0.045±0.006	0.019±0.001	0.013±0.000	10.61766	10.61772	10.61773
CTP8	144±40	567±213	2473±969	0.026±0.006	0.011±0.001	0.007±0.000	8.555072	8.560030	8.561277
ZDT1	393±26	1771±122	8224±550	0.112±0.007	0.029±0.001	0.010±0.000	6.662722	6.663545	6.663480
ZDT2	462±27	2115±133	9880±582	0.146±0.008	0.035±0.001	0.011±0.000	7.330062	7.330301	7.330399
ZDT3	320±26	1411±104	6480±466	0.092±0.008	0.023±0.001	0.007±0.000	7.041472	7.042860	7.042790
ZDT4	502±26	2167±102	10244±480	0.218±0.010	0.057±0.002	0.019±0.000	265.6638	265.6636	265.6638
ZDT5	246±24	1023±53	4688±284	0.064±0.006	0.016±0.001	0.004±0.000	726.7726	727.6003	730.1003
ZDT6	597±61	2714±41	12658±240	0.248±0.019	0.074±0.001	0.035±0.000	5.816499	5.816490	5.816570
SCH1	73±12	226±48	1164±128	0.035±0.003	0.031±0.001	0.029±0.000	12875.30	12875.30	12875.29
SCH2	53±19	178±52	860±135	0.024±0.002	0.021±0.001	0.019±0.000	659.2688	659.2766	659.2829
POL	139±19	542±125	2448±581	0.028±0.004	0.014±0.001	0.010±0.000	2916.155	2916.146	2916.148
KUR	282±24	1255±102	5666±456	0.052±0.004	0.019±0.001	0.011±0.000	633.9241	633.9534	633.9651
FON	330±42	1456±178	6664±840	0.059±0.004	0.018±0.001	0.009±0.000	0.337600	0.337189	0.337386
BNH	56±13	148±43	975±185	0.021±0.002	0.020±0.002	0.020±0.000	5082.153	5082.767	5082.387
SRN	130±26	530±107	2326±557	0.035±0.003	0.023±0.001	0.020±0.000	278248.1	278314.4	278320.4
TNK	206±37	866±147	3937±695	0.043±0.002	0.013±0.001	0.006±0.000	15.22497	15.22506	15.22529
OSY	323±39	1478±157	6772±798	0.084±0.010	0.021±0.002	0.008±0.000	69852.53	69906.43	69906.63

noted that for all the two-objective test problems considered in this paper: (i) the true POFs are known, and are reported in their respective references, and (ii) the POF–approximations obtained by the underlying MOEA (NSGA-II) conforms with the true POFs. However, as the proposed *dissimilarity* measure primarily investigates the stabilization of the MOEA (a necessary but not sufficient condition for conformance with the true POF), the discussions in the subsequent section are focused on MOEA stabilization and termination, rather than the quality of the POF–approximation offered by them.

(c) the $\mathcal{D}(p, q)$ measure at each MOEA generation is plotted, along with its mean and standard deviation over as many generations as deemed necessary to mark (done by dotted vertical lines) the N_{gt} against $n_p = 2, 3, 4$.

For the sake of brevity, only a few sample plots under items (b) and (c) above are presented here, while the remaining ones are presented in the supplementary file attached with this paper.

A. Experimental Results for Multi-objective Problems

The N_{gt} determined by Algorithm 2, and the corresponding hypervolume measures are presented in Table I.

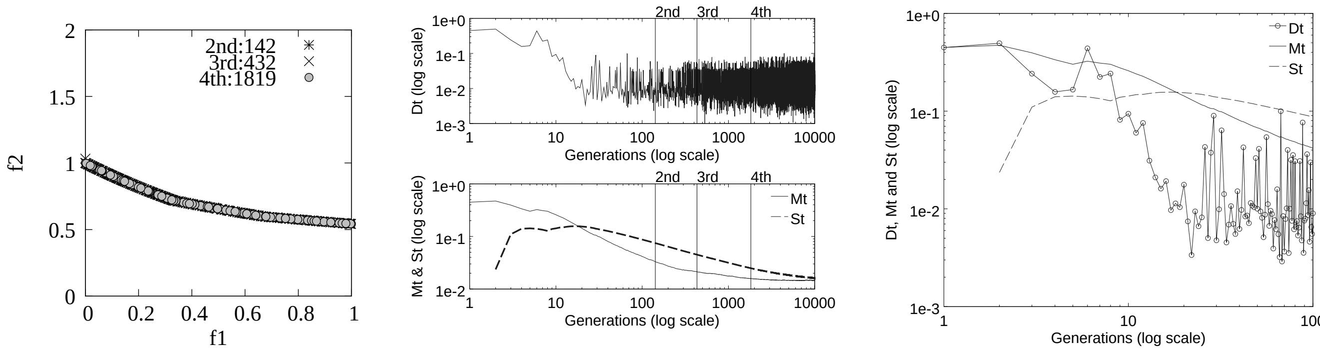
Before interpreting the entire set of results, CTP1 is chosen for a sample discussion, for which:

- Figure 2a reveals that the POF–approximation offered by NSGA-II, at all N_{gt} against $n_p = 2, 3, 4$, is significantly similar. Hence, it could be inferred that the NSGA-II population has largely stabilized by the N_{gt} against $n_p = 2$, and running NSGA-II any further does not offer any significant improvements. These observations are also supported by the significantly low $\mathcal{D}(p, q)$ measures (Figure 2b). The minor variations in $\mathcal{D}(p, q)$ at any given generation, compared to the immediately previous ones, could be attributed to the *variation* operator like mutation. Furthermore, while the Table I presents N_{gt} averaged over 10 different NSGA-II runs, Figure 2b marks the N_{gt} corresponding to one of the 10 runs.
- Figure 2c facilitates the visualization of how $\mathcal{D}(p, q)$ varies from the initial generations (to the first 100 generations, here). The variations therein could be explained by the facts that: (i) M_t and S_t are set to zero at $t = 0$, and (ii) like any other MOEA, NSGA-II makes significant advances toward the POF in the early phase of evolution, while relatively smaller advances are made subsequently. Hence, while M_t rapidly shoots up during the first few generations, it gradually subsides.

Figure 2b and 2c also affirm the rationale for basing the termination on the n_p–conformance of both the mean and standard-deviation of $\mathcal{D}(p, q)$, over n_s generations of an MOEA. It can be seen that variations in the $\mathcal{D}(p, q)$ values at different generations do not cause an abrupt change in the corresponding values of M_t and S_t. This suggests that while an N_{gt} based on $n_p = 2$ may be reliable, recourse to $n_p = 3$ or $n_p = 4$ will add to the reliability of reported N_{gt}, at additional computational cost.

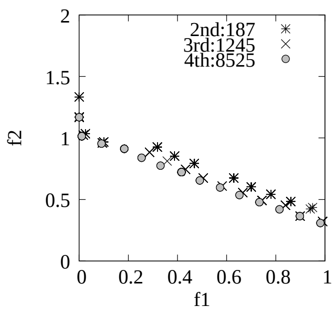
The results for all other MOPs (Table I) reveal that the $\mathcal{D}(p, q)$ values become smaller (tend to zero) as the value of n_p is raised from 2 to 4. While it is impractical to relate the reported N_{gt} with the underlying features of all the problems, some interesting cases are being discussed, in that:

- among the CTP problems, the N_{gt} and $\mathcal{D}(p, q)$ values for CTP4 are the highest, suggesting that the latter posed

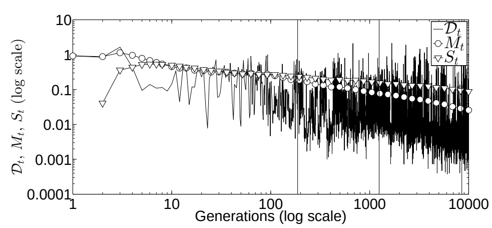


(a) POF–approximation (b) $\mathcal{D}(p, q)$ values (c) $\mathcal{D}(p, q)$ values over Initial Generations

Fig. 2: CTP1: NSGA-II based POF–approximations (at N_{gt} against $n_p = 2, 3, 4$), and corresponding $\mathcal{D}(p, q)$ measures (with mean & standard deviation) by Algorithm 2. The evolution of $\mathcal{D}(p, q)$ measures over first 100 generations is also highlighted.



(a) CTP4: POF–approximation



(b) CTP4: $\mathcal{D}(p, q)$ values

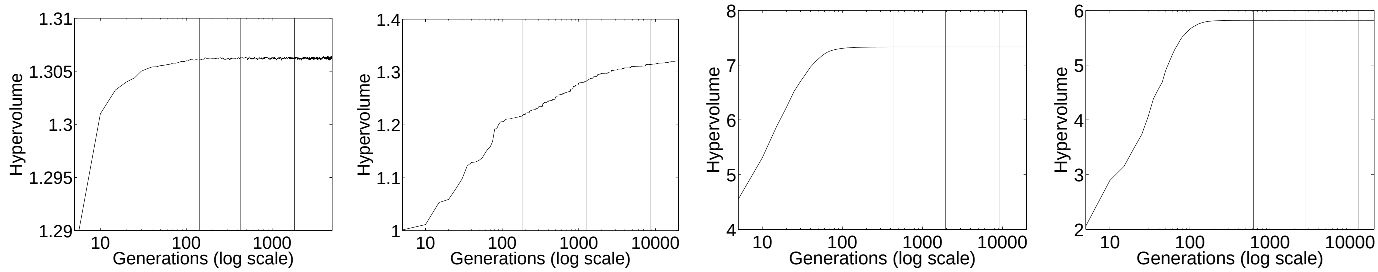
Fig. 3: CTP4: NSGA-II based POF–approximations (at N_{gt} against $n_p = 2, 3, 4$), and corresponding $\mathcal{D}(p, q)$ measures (with mean & standard deviation) by Algorithm 2.

most difficulties for NSGA-II. This inference could be justified by the fact that the Pareto-optimal solutions for CTP4 lie at the end of long narrow *tunnel*s [47], posing difficulties for an MOEA to converge. This also explains why in the case of CTP4[3]: (a) an increase in n_p is accompanied by improvements in the POF-approximation (Figure 3a), and (b) the variation in $\mathcal{D}(p, q)$ values is more significant compared to other CTP problems (Figure 3b).

- within the ZDT problems, the N_{gt} and $\mathcal{D}(p, q)$ values corresponding to ZDT6 are the highest, followed by ZDT4 (regardless of n_p). These results could be explained by the fact that compared to other problems:
  (a) ZDT6 poses more difficulties to an MOEA owing to the non-uniformity in its search space, in that: (i) the Pareto-optimal solutions are not uniformly distributed across the POF, and (ii) the density of solutions changes with respect to their proximity from the POF, such that the density closer to the POF is lower.
  (b) ZDT4 poses more difficulties to an MOEA owing to the presence of 21^9 local Pareto-optimal solutions, and 100 distinct POFs of which only one is global.
- in the case of problems with low or moderate difficulty levels, some interesting and plausible patterns are observed (though, no hard generalizations can be drawn, given the stochastic nature of MOEAs). For example:
  (a) when $M = 2$, $J = 0$ and $n = 1$: the mean N_{gt} is lower when the nature of objective(s) is simpler, as in SCH2 (linear and quadratic objectives) vis-*à*-vis SCH1 (both quadratic objectives).
  (b) when $M = 2$, $J = 2$ and $n = 2$: the N_{gt} are in the order BNH $<$ SRN $<$ TNK, and can be explained by difficulty levels guided by the nature of POF and constraints. Both BNH and SRN have a convex POF, but in BNH the constraints do not make any part of the unconstrained POF infeasible, unlike SRN where they eliminate some part of it. In TNK, the POF is discontinuous and lies over a nonlinear constraint surface.
  (c) the N_{gt} for OSY are relatively higher, and can be attributed to higher difficulty levels guided by more constraints ($J = 6$), more variables ($n = 6$), and a POF which is a concatenation of five regions and maintaining subpopulations at the intersection of constraint boundaries is a challenge.

Besides the fact that: (i) the reported N_{gt} correlate well with the difficulty levels of the underlying problems, and (ii) the corresponding POF approximations obtained conform with the true POFs available in literature, the accuracy of Algorithm 2 could be further validated through the corresponding hypervolume measures (Table I). Once an MOEA achieves a good POF–approximation, the hypervolume should not significantly change if the MOEA were to run any longer. Hence, for the reported N_{gt} to be inferred as correct:

[3]Notably, all the plots for MOPs presented in the supplementary file attached with this paper are alike CTP1, in that, no significant visual difference can be observed in the POF–approximations against increasing N_{gt} corresponding to an increase in n_p from 2 to 4.



(a) CTP1 (b) CTP4 (c) ZDT2 (d) ZDT6

Fig. 4: Hypervolume measure across MOEA generations, for a few sample multi-objective problems. The vertical lines in each plot mark the N_{gt} deduced by Algorithm 2, corresponding to $n_p = 2, 3, 4$, respectively.

- the hypervolume measure preceding the N_{gt} corresponding to $n_p = 2$ should undergo significant change. This is indeed the case, as sample plots (for problems with varying degree of difficulty) in Figure 4 reveal.
- the hypervolume measure across the N_{gt} corresponding to $n_p = 2, 3$ and 4 should not vary significantly. This is indeed the case, barring the CTP4 problems, the plausible reasons for which have already been discussed above.

B. Experimental Results for Many-objective Test Problems

Most existing MOEAs are known to fail in providing a good POF–approximation in the case of MaOPs with reasonable computational effort [29]. This paper neither aims to demonstrate the efficacy of the termination detection algorithm on multiple MOEAs, nor aims at a comparative evaluation of MOEAs on MaOPs. Instead, it aims to illustrate the *generality* and *scalability* of the Algorithm 2. Towards it:

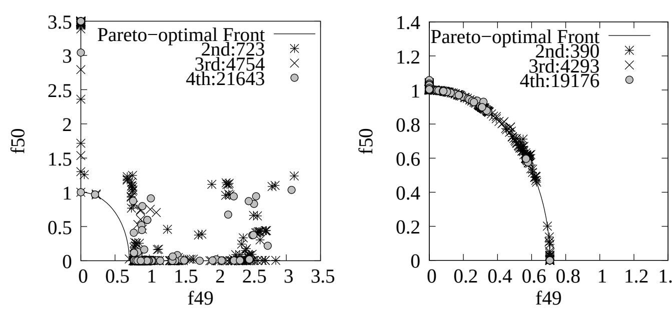
(a) the performance of Algorithm 2 on MaOPs, corresponding to NSGA-II (a generational MOEA), is studied.
(b) the scalability of Algorithm 2 is assessed by evaluating the accuracy of its deductions (N_{gt}) for some MaOPs, with reference to the hypervolume measure.
(c) the *generality* of Algorithm 2 is assessed by evaluating its performance for some MaOPs, corresponding to (another broad category of MOEAs, besides the generational) a steady-state MOEA, namely, ϵ-MOEA.

The item (a) articulated above can be realized with respect to the two categories of MaOPs considered: (i) the DTLZ problems, where the dimension of the POF is the same as M, and (ii) the DTLZ5(I, M) problems, where the dimension of the POF (I) may be less than M. Notably, for both categories, the N_{gt} can be seen (Table II) to increase with an increase in the desired degree of accuracy controlled by n_p (as expected). In the particular case of DTLZ problems:

- the general trend is that the $\mathcal{D}(p, q)$ values increase with an increase in M.
- for all the considered variants of the DTLZ problems, the $\mathcal{D}(p, q)$ values are far-off from zero, implying stagnation of NSGA-II far-off from the true POF.
- in many instances NSGA-II can be seen to stagnate faster when the number of objectives are higher, for a particular DTLZ problem. For instance, corresponding to the case of $n_p = 2$: (i) the N_{gt} for DTLZ1(5), DTLZ1(15), and DTLZ1(25) are 357 ± 158, 318 ± 75, 322 ± 54, respectively, and (ii) the N_{gt} for DTLZ3(5) and DTLZ3(15) are 336 ± 100 and 282 ± 40, respectively. This could be related to the fact that as M increases, more and more solutions tend to be non-dominated from the early generations, given which the dominance based primary selection becomes ineffective, resulting in early stagnation.

Furthermore, in the particular case of DTLZ5(I, M) problems:

- the general trend of an increase in the $\mathcal{D}(p, q)$ values with an increase in M holds, but with some exceptions. For example, the $\mathcal{D}(p, q)$ values for DTLZ5$(2, 20)$ on an average are better (lower) than those for DTLZ5$(5, 10)$. Though the POF–approximation is likely to worsen with an increase in M, the above could be explained by the fact that the search efficiency of an MOEA is not just governed by M, but also by the dimension of the POF.
- for all the variants of the DTLZ5(I, M) problem, barring DTLZ5$(2, 5)$, the $\mathcal{D}(p, q)$ values are far-off from zero, implying stagnation of NSGA-II far-off from the true POF. This is verified by the POF–approximations for a sample problem, namely, DTLZ5$(2, 50)$ (Figure 5a).



(a) NSGA-II (b) ϵ-MOEA

Fig. 5: DTLZ5$(2, 50)$: POF–approximations by NSGA-II and ϵ-MOEA, at N_{gt} (against $n_p = 2, 3, 4$).

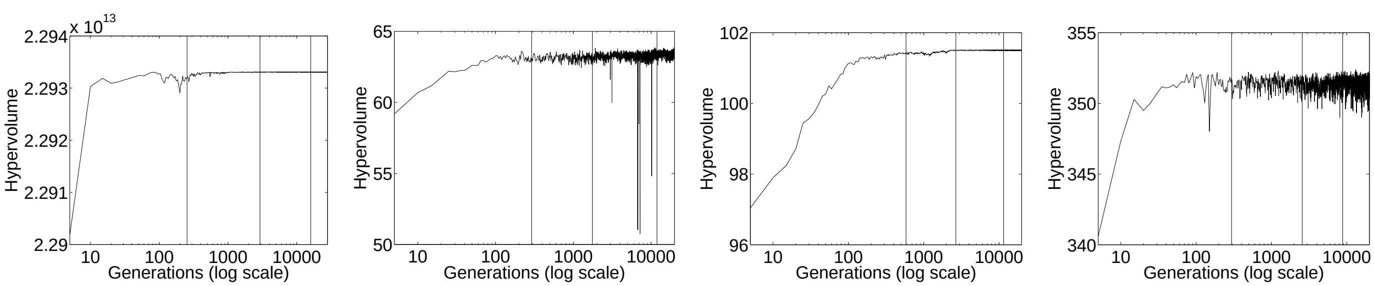
Towards realizing the item (b) articulated above, the hypervolume measures for problems with $M = 5$ have been reported[4] in Table II. These measures, some of which have

[4]Beyond $M = 5$, it becomes difficult to compute the hypervolumes in practically feasible time limit, hence, the scope is limited to $M = 5$.

TABLE II: NSGA-II based N_{gt} (against $n_p = 2, 3, 4$) & $\mathcal{D}(p, q)$ measures for many-objective versions of the DTLZ and DTLZ5(I, M) problems, and the corresponding hypervolume measures (only when $M = 5$). In addition, the ϵ-MOEA based N_{gt} and $\mathcal{D}(p, q)$ measures for problems with two- or three-dimensional POFs are reported, in rows prefixed by: Problem$^{\epsilon}$.

For brevity, the DTLZ5(I, M) problems are abbreviated as D5(I, M). The N_{gt} and $\mathcal{D}(p, q)$ results are formatted as $\mu \pm \sigma$, where μ and σ represent the mean and standard deviation of the respective entry, over 10 different NSGA-II runs. The hypervolume measures for one NSGA-II run, at N_{gt} corresponding to $n_p = 2, 3, 4$ are formatted as (α; β; γ) $\times$ 10^p. For instance, for $n_p = 2$, the hypervolume is $\alpha \times 10^p$.

Problem	Generations N_{gt}			Dissimilarity Measure $\mathcal{D}(p, q)$			Hypervolume (formatted as above)
	$n_p = 2$	$n_p = 3$	$n_p = 4$	$n_p = 2$	$n_p = 3$	$n_p = 4$	$n_p = 2$; $n_p = 3$; $n_p = 4$
D1(05)	357±158	2544±349	17266±4999	1.193±0.242	1.614±0.173	1.617±0.242	(2.2932; 2.2933; 2.29333) $\times 10^{13}$
D1(15)	318±75	2236±400	23587±6688	3.031±0.105	3.310±0.121	3.403±0.067	–
D1(25)	322±54	1947±507	19358±4530	3.114±0.066	3.248±0.089	3.215±0.096	–
D2(05)	216±44	1490±230	10592±1132	3.093±0.038	3.113±0.019	3.161±0.016	(0.6285; 0.6296; 0.6333) $\times 10^{2}$
D2(15)	276±33	1557±386	10579±1208	3.517±0.041	3.438±0.052	3.274±0.055	–
D2(25)	308±58	1792±278	12912±2224	3.561±0.064	3.401±0.085	3.228±0.036	–
D3(05)	336±100	1886±235	11543±1496	1.835±0.086	1.664±0.120	1.633±0.109	(1.6458; 1.6456; 1.6462) $\times 10^{16}$
D3(15)	282±40	1893±516	14477±1366	3.298±0.023	3.208±0.108	2.971±0.127	–
D3(25)	354±79	2470±933	17671±2946	3.136±0.066	2.992±0.131	2.653±0.213	–
D4(05)	420±36	1766±183	10667±1097	3.019±0.032	3.120±0.011	3.135±0.003	(0.6773; 0.6767; 0.6796) $\times 10^{2}$
D4(15)	268±40	1299±141	10388±889	3.599±0.020	3.670±0.010	3.621±0.007	–
D4(25)	235±65	1422±171	10480±914	3.520±0.043	3.652±0.007	3.652±0.009	–
D5(2,05)	464±59	2015±292	8776±1414	0.436±0.043	0.271±0.077	0.225±0.112	(1.0141; 1.0149; 1.0149) $\times 10^{2}$
D5(2,05)$^{\epsilon}$	308±17	1271±142	5400±634	0.208±0.008	0.180±0.005	0.145±0.007	–
D5(2,20)	706±367	3209±982	15244±1944	2.549±0.476	2.092±0.463	1.982±0.486	–
D5(2,20)$^{\epsilon}$	414±32	2677±261	11272±872	0.533±0.023	0.239±0.024	0.145±0.010	–
D5(2,50)	638±101	5681±789	22363±2620	3.168±0.083	1.943±0.263	1.630±0.252	–
D5(2,50)$^{\epsilon}$	450±67	3748±573	16232±2501	0.888±0.196	0.311±0.031	0.173±0.011	–
D5(3,05)	329±50	2444±220	9813±674	1.479±0.048	0.986±0.036	0.893±0.029	(1.0130; 1.0150; 1.0150) $\times 10^{2}$
D5(3,05)$^{\epsilon}$	304±36	1444±212	10010±897	1.278±0.054	1.214±0.037	1.237±0.039	–
D5(3,20)	238±35	3020±416	14784±1090	3.580±0.056	2.902±0.169	2.793±0.147	–
D5(5,10)	219±27	2191±560	12435±1166	3.430±0.054	3.246±0.112	3.040±0.085	–
D5(5,20)	290±56	3735±357	14366±983	3.703±0.039	3.142±0.052	2.953±0.039	–
D5(7,10)	250±48	1636±387	11919±1324	3.475±0.038	3.387±0.050	3.141±0.083	–
D5(7,20)	306±57	1877±986	15687±1482	3.699±0.039	3.569±0.147	3.030±0.058	–

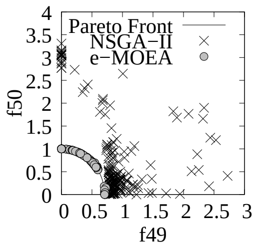


(a) DTLZ1(5) (b) DTLZ2(5) (c) DTLZ5(2,5) (d) DTLZ5(3,5)

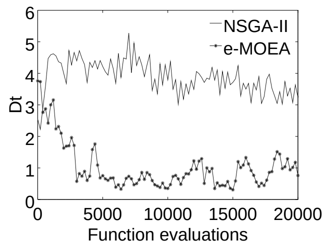
Fig. 6: Hypervolume measure across MOEA generations, for a few sample MaOPs. The vertical lines in each plot mark the N_{gt} deduced by Algorithm 2, against $n_p = 2, 3, 4$, respectively.

also been plotted in Figure 6 suggest that the hypervolume measure: (i) preceding the N_{gt} corresponding to $n_p = 2$ undergoes significant change, and (ii) across the N_{gt} corresponding to $n_p = 2, 3$ and 4 does not vary significantly. This conforms with the expected pattern (Section VIII-A), indicating that the reported N_{gt} are correct. The fact that the accuracy of Algorithm 2 is retained even in the case of MaOPs, affirms its scalability with the number of objectives.

Finally, towards realizing the item (c) articulated above, the response of Algorithm 2 to the POF–approximations offered by ϵ-MOEA, for some problems with two- and three-dimensional POFs, is explored. Notably, both the N_{gt} and $\mathcal{D}(p, q)$ values for ϵ-MOEA are lower than those for NSGA-II (Table II). Interestingly, in contrast to NSGA-II (Figure 5a), ϵ-MOEA seems to have converged to the POF (Figure 5b) and yet reports non-zero $\mathcal{D}(p, q)$ values. This could be explained by the fact that while the ϵ-MOEA population has converged to the POF, it has not stabilized in terms of diversity (evident by the sparsely populated regions on the POF). Given the fact, that the $\mathcal{D}(p, q)$ values basically reflect on to the stabilization of the population, the reported results are justified.



(a) POF–approximations



(b) $\mathcal{D}(p, q)$ measures

Fig. 7: DTLZ5(2, 50): comparison of the POF–approximations by NSGA-II and ϵ-MOEA after 20,000 function evaluations (or 100 generations), and corresponding $\mathcal{D}(p, q)$ measures.

IX. Potential Future Directions

The future research directions, inspired by the proposed $\mathcal{D}(p, q)$ measure may relate to exploring its utility towards:

- performance comparison of different MOEAs: a lower $\mathcal{D}(p, q)$ measure which fundamentally implies higher conformance between MOEA populations in successive generations (Section V), may reflect on to better convergence to the POF in entirety or in part. Figure 7 illustrates that after 20,000 function evaluations[5], the POF–approximation obtained by ϵ-MOEA is much better than NSGA-II's, and this is accompanied by lower $\mathcal{D}(p, q)$ measures for the former. Future work may explore in detail *if and how* the $\mathcal{D}(p, q)$ measures at their respective N_{gt} (or after a fixed number of function evaluations) could be used in conjunction with (if not in isolation) the existing performance indicators that help compare MOEAs. Such a possibility in the case of MaOPs, where an indicator like hypervolume becomes computationally punitively expensive, may mark a significant contribution.
- unraveling the timing for change of MOEA parameters: the parameters associated with the variation operators are fixed *a priori*, and the chosen settings may not favor a distributed exploration of the entire search space or the POF [48]. This is often manifested in problems with disconnected POF, where an MOEA gets stuck in one segment and struggles for several generations before getting a solution in the other segments. Such a situation could possibly be favorably altered by changing the parameter settings *on-the-fly*. For example, in the MOEAs using real-parameter SBX recombination and polynomial mutation operator (as in NSGA-II), if the distribution index of crossover and mutation operators is reduced, the span for the new solutions could be wider, promising faster exploration of the search space, including the POF. Hence, when a zero or near-zero (constant) values of $\mathcal{D}(p, q)$ are observed over significant number of generations, while one option could be to terminate the MOEA, another option could be to treat it as the right timing (otherwise not known) for *exploratory/corrective* option of changing the parameter settings associated with the MOEA's variation operators. Further research in this direction could provide interesting insights and results.
- unraveling the timing of objective-reduction [30], [31] based decision-support for MaOPs: the importance of applying objective reduction techniques to derive a decision support characterized by *objectivity*, *repeatability*, *consistency*, and *coherence* [32] is being recognized. Notably, the propositions of the decision-support depend on when along an MOEA run (in terms of generations) are the objective-reduction techniques applied, an answer to which is ironically not available. In future, it'll be interesting to explore on real-world MaOPs, as to how effectively the stabilization of the proposed *dissimilarity* measure can indicate the stabilization of the MOEA population and an apt timing for application of objective reduction techniques for decision support.

[5]For simplicity, evaluation of one solution is referred as one function evaluation. For $N = 200$, $20,000$ function evaluations $\equiv$ 100 generations, a pre-stabilized case, since 100 is less than the N_{gt} at $n_p = 2$.

X. Conclusion

This paper has proposed a novel entropy based *dissimilarity* measure that provides a termination criterion, and facilitates a *generic*, *scalable*, and *computationally efficient* termination detection algorithm. Based on experiments on a wide range of test MOPs and MaOPs, it is established that the calls for MOEA termination are timed differently for different problems based on their inherent difficulty levels in terms of the nature of objective functions & the (variable) search space, number of objectives involved, and the dimensionality of the true POF. The accuracy of the calls for termination is validated through comparison of the POF-approximations at different stages of an MOEA's run, and also through the hypervolume measures. For some sample problems, the performance of the termination detection algorithm is evaluated against two different categories of MOEAs, namely, NSGA-II (a generational MOEA) and ϵ-MOEA (a steady state MOEA). For a class of MaOPs, where ϵ-MOEA obtained faster and (qualitatively) better POF–approximations, the termination detection algorithm correspondingly reported lower *dissimilarity* measures and called for earlier termination. The authors hope that by facilitating an apt *on-the-fly* MOEA termination, this work will be significantly useful towards preventing either a pre-matured MOEA termination or wastage of computational resources, a critical feature that has ironically remained largely unaddressed so far.

References

[1] 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.

[2] H. Ishibuchi, N. Akedo, and Y. Nojima, "Behavior of multiobjective evolutionary algorithms on many-objective knapsack problems," *Evolutionary Computation, IEEE Transactions on*, vol. 19, pp. 264–283, April 2015.

[3] M. Asafuddoula, T. Ray, and R. Sarker, "A decomposition-based evolutionary algorithm for many objective optimization," *Evolutionary Computation, IEEE Transactions on*, vol. 19, pp. 445–460, June 2015.

[4] K. Miettinen, *Nonlinear Multiobjective Optimisation*. Norwell, MA: Kluwer Academic Publishers, 1998.

[5] C. A. C. Coello and G. B. Lamont, *Applications of Multi-Objective Evolutionary Algorithms*. Singapure: World Scientific, 2004.

[6] D. A. V. Veldhuizen and G. B. Lamont, "Evolutionary computation and convergence to a pareto front," in *Stanford University, California*, pp. 221–228, Morgan Kaufmann, 1998.

[7] T. Hanne, "On the convergence of multiobjective evolutionary algorithms," *European Journal of Operational Research*, vol. 117, no. 3, pp. 553 – 564, 1999.

[8] G. Rudolph and A. Agapie, "Convergence properties of some multi-objective evolutionary algorithms," in *In Congress on Evolutionary Computation (CEC 2000)*, pp. 1010–1016, IEEE Press, 2000.

[9] G. Rudolph, "Self-adaptive mutations may lead to premature convergence," *IEEE Transactions on Evolutionary Computation*, vol. 5, pp. 410–414, Aug. 2001.

[10] F. Carlos, X. Gandibleux, P. Korhonen, L. Marti, B. Naujoks, L. Thiele, W. Jyrki, and E. Zitzler, "09041 working group on EMO for interactive multiobjective optimization (1st round)," in *Hybrid and Robust Approaches to Multiobjective Optimization* (K. Deb, S. Greco, K. Miettinen, and E. Zitzler, eds.), no. 09041 in Dagstuhl Seminar Proceedings, (Dagstuhl, Germany), Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, Germany, 2009.

[11] O. Roudenko and M. Schoenauer, "A steady performance stopping criterion for Pareto-based evolutionary algorithms," in *Proceedings of the 6th International Multi-Objective Programming and Goal Programming Conference*, 2004.

[12] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A Fast Elitist Multi-Objective Genetic Algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2000.

[13] L. Martí, J. García, A. Berlanga, and J. M. Molina, "A cumulative evidential stopping criterion for multiobjective optimization evolutionary algorithms," in *Proceedings of the 9th annual conference on Genetic and evolutionary computation*, GECCO '07, (New York, NY, USA), pp. 2835–2842, ACM, 2007.

[14] E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: Improving the strength pareto evolutionary algorithm," Tech. Rep. 103, Computer Engineering and Networks Laboratory (TIK), Department of Electrical Engineering, Swiss Federal Institute of Technology (ETH) Zurich, May 2001.

[15] J. D. Knowles and D. W. Corne, "Approximating the nondominated front using the pareto archived evolution strategy," in *Evolutionary Computation*, vol. 8, pp. 149–172, 2000.

[16] L. Martí, J. García, A. Berlanga, and J. M. Molina, "An approach to stopping criteria for multi-objective optimization evolutionary algorithms: The MGBM criterion," in *IEEE Congress on Evolutionary Computation*, pp. 1263–1270, May 2009.

[17] L. Martí, J. García, A. Berlanga, and J. M. Molina, "A progress indicator for detecting success and failure in evolutionary multi-objective optimization," in *IEEE Congress on Evolutionary Computation*, pp. 1–8, July 2010.

[18] J. Bader and E. Zitzler, "Hype: An algorithm for fast hypervolume-based many-objective optimization," tech. rep., Computer Engineering and Networks Laboratory, ETH Zurich, 8092 Zurich, Switzerland, November 2008.

[19] H. Trautmann, U. Ligges, J. Mehnen, and M. Preuss, "A convergence criterion for multiobjective evolutionary algorithms based on systematic statistical testing," in *Parallel Problem Solving from Nature – PPSN X* (G. Rudolph, T. Jansen, S. Lucas, C. Poloni, and N. Beume, eds.), vol. 5199 of *Lecture Notes in Computer Science*, pp. 825–836, Springer Berlin / Heidelberg, 2008.

[20] D. J. Sheskin, *Handbook of Parametric and Nonparametric Statistical Procedures*. Chapman & Hall/CRC, 4 ed., 2007.

[21] G. Yen and Z. He, "Performance metric ensemble for multiobjective evolutionary algorithms," *Evolutionary Computation, IEEE Transactions on*, vol. 18, pp. 131–144, Feb 2014.

[22] T. Wagner, H. Trautmann, and B. Naujoks, "OCD: Online Convergence Detection for Evolutionary Multi-Objective Algorithms Based on Statistical Testing," in *Evolutionary Multi-Criterion Optimization* (M. Ehrgott, C. Fonseca, X. Gandibleux, J.-K. Hao, and M. Sevaux, eds.), vol. 5467 of *Lecture Notes in Computer Science*, pp. 198–215, Springer Berlin / Heidelberg, 2009.

[23] H. Trautmann, T. Wagner, B. Naujoks, M. Preuss, and J. Mehnen, "Statistical Methods for Convergence Detection of Multi-Objective Evolutionary Algorithms," *Evolutionary Computation*, vol. 17, no. 4, pp. 493–509, 2009.

[24] J. L. Guerrero, L. Martí, A. Berlanga, J. Garcí, and J. M. Molina, "Introducing a robust and efficient stopping criterion for MOEAs," in *IEEE Congress on Evolutionary Computation*, pp. 1–8, July 2010.

[25] T. Goel and N. Stander, "A non-dominance-based online stopping criterion for multi-objective evolutionary algorithms," *International Journal for Numerical Methods in Engineering*, vol. 84, no. 6, pp. 661–684, 2010.

[26] L. Bui, S. Wesolkowski, A. Bender, H. Abbass, and M. Barlow, "A dominance-based stability measure for multi-objective evolutionary algorithms," in *Evolutionary Computation, 2009. CEC '09. IEEE Congress on*, pp. 749–756, May 2009.

[27] J. Dutta, K. Deb, R. Tulshyan, and R. Arora, "Approximate KKT points and a proximity measure for termination," *Journal of Global Optimization*, vol. 56, pp. 1463–1499, May 2012.

[28] R. T. Rockafellar, *Convex Analysis*. Princeton University Press, 1996.

[29] H. Ishibuchi, N. Tsukamoto, and Y. Nojima, "Evolutionary many-objective optimization: A short review," in *IEEE Congress on Evolutionary Computation*, pp. 2419–2426, June 2008.

[30] D. Saxena, J. Duro, A. Tiwari, K. Deb, and Q. Zhang, "Objective reduction in many-objective optimization: Linear and nonlinear algorithms," *IEEE Transactions on Evolutionary Computation*, vol. 17, no. 1, pp. 77–99, 2013.

[31] D. Brockhoff, D. K. Saxena, K. Deb, and E. Zitzler, "On handling a large number of objectives a posteriori and during optimization," in *Multiobjective Problem Solving from Nature*, Natural Computing Series PartIV, (Berlin, Germany), pp. 377–403, Springer, 2007.

[32] J. A. Duro, D. K. Saxena, K. Deb, and Q. Zhang, "Machine Learning based Decision Support for Many-objective Optimization Problems," *Neurocomputing*, vol. 146, no. 0, pp. 30–47, 2014.

[33] C. E. Shannon, "A mathematical theory of communication," *SIGMOBILE Mobile Computing and Communications Review*, vol. 5, pp. 3–55, January 2001.

[34] S. Kullback and R. A. Leibler, "On information and sufficiency," *The Annals of Mathematical Statistics*, vol. 22, pp. 79–86, March 1951.

[35] W. K. Härdle, M. Müller, S. Sperlich, and A. Werwatz, *Nonparametric and Semiparametric Models*. Springer, 2004.

[36] D. W. Scott, *Multivariate Density Estimation: Theory, Practice, and Visualization*. John Wiley & Sons, 1992.

[37] E. Parzen, "On estimation of a probability density function and mode," *The Annals of Mathematical Statistics*, vol. 33, no. 3, pp. 1065–1076, 1962.

[38] M. Rosenblatt, "Remarks on some nonparametric estimates of a density function," *The Annals of Mathematical Statistics*, vol. 27, no. 3, pp. 832–837, 1956.

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

[40] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler, "Scalable Test Problems for Evolutionary Multi-Objective Optimization," in *Evolutionary Multiobjective Optimization: Theoretical Advances and Applications* (A. Abraham, R. Jain, and R. Goldberg, eds.), pp. 105–145, Springer, 2005.

[41] K. Deb, M. Mohan, and S. Mishra, "Evaluating the e-Domination Based Multi-Objective Evolutionary Algorithm for a Quick Computation of Pareto-Optimal Solutions," *Evolutionary Computation*, vol. 13, no. 4, pp. 501–525, 2005.

[42] D. Freedman and P. Diaconis, "On the histogram as a density estimator: L2 theory," *Probability Theory and Related Fields*, vol. 57, no. 4, pp. 453–476, 1981.

[43] D. W. Scott, "On optimal and data-based histograms," *Biometrika*, vol. 66, no. 3, pp. 605–610, 1979.

[44] C. Stone, "An asymptotically histogram selection rule," in *Proceedings of Second Berkeley Symposium* (J. Neyman, ed.), pp. 513–520, Berkeley: Univ. California Press, 1984.

[45] M. Rudemo, "Empirical choice of histograms and kernel density estimators," *Scandinavian Journal of Statistics*, vol. 9, no. 2, pp. 65–78, 1982.

[46] C. M. Fonseca, L. Paquete, and M. L. I. nez, "An Improved Dimension-Sweep Algorithm for the Hypervolume Indicator," in *IEEE Congress on Evolutionary Computation*, (Sheraton Vancouver Wall Centre Hotel, Vancouver, BC, Canada), pp. 16–21, July 2006.

[47] K. Deb, A. Pratap, and T. Meyarivan, "Constrained test problems for multi-objective evolutionary optimization," in *Evolutionary Multi-Criterion Optimization*, vol. 1993 of *Lecture Notes in Computer Science*, pp. 284–298, Springer Berlin / Heidelberg, 2001.

[48] G. Karafotias, M. Hoogendoorn, and A. Eiben, "Parameter control in evolutionary algorithms: Trends and challenges," *Evolutionary Computation, IEEE Transactions on*, vol. 19, pp. 167–187, April 2015.