A hybrid algorithm based on MOEA/D and local search for multiobjective optimization

Man-Fai Leung
School of Science and Technology
The Open University of Hong Kong
Hong Kong, China
mfleung@ouhk.edu.hk

Sin-Chun Ng
School of Science and Technology
The Open University of Hong Kong
Hong Kong, China
scng@ouhk.edu.hk

***Abstract*—A hybrid algorithm is proposed for multiobjective optimization in this paper. The proposed algorithm consists of multiobjective evolutionary algorithm based on decomposition (MOEA/D) and recurrent neural network, where MOEA/D is for global search while recurrent neural network is for local search. The performance of the proposed algorithm is compared with other three multi-objective algorithms in terms of hypervolume and inverted generational distance. The performance investigation shows that the proposed algorithm generally outperforms the compared algorithms.**

***Index Terms*—multiobjective optimization, evolutionary algorithm, recurrent neural network, Pareto-optimal solution, MOEA/D**

I. Introduction

Multiobjective optimization (MO) aims to optimize a problem with two or more conflicting objectives simultaneously, under a set of constraints. After optimization an MO problem, non-dominated solutions will be obtained. If there is no other solution that dominates the non-dominated solution within the feasible region, then the non-dominated solution is called Pareto-optimal solution (POS).

For solving a multiobjective optimization problem (MOP), classical methods [1], [2] first transform a parameterized single-objective problem, then multiple optimization processes are repeated with different parameters to acquire a set of solutions. Over the past few decades, the use of meta-heuristics (e.g., genetic algorithm [3], simulated annealing [4] and tabu search [5]) has become popular [6] because each of them can generate a set of POSs at a time. Among them, MOEA/D [7] is the most popular for MO. It decomposes an MOP into various single-objective subproblems by using scalarization and the subproblems are solved using an evolutionary algorithm (EA). A set of weight vectors must be assigned properly to get a set of well-distributed non-dominated solutions. For this reason, a substantial amount of work has been carried out related to MOEA/D and various versions were proposed (e.g., [8]–[12]). However, the EA in MOEA/D is deficient in local search. To enhance convergence of the algorithm, one approach is to tune the parameters in EA. Another approach is to incorporate other measures for enhancing the local search ability (e.g., [13]–[19]). Ishibuchi et al. proposed a genetic local search algorithm for MO and a scalarizing function is randomly drawn for parent selection [13]. In [14], a hybrid (memetic) algorithm was proposed with the use of local search (1+1)-PAES for MO. In [15], a local search strategy was proposed for memetic multiobjective algorithms. In [16], a gradient-based sequential quadratic programming was incorporated into multiobjective algorithms. In [17], a memetic algorithm was proposed for combinatorial MOPs.

Since the pioneer work of Hopfield and Tank on a continuous-time optimization approach based on neural networks [20], [21], extensive related works have been conducted: a neural network was proposed to solve the shortest path problem [22]; a projection neural network was proposed for solving constrained optimization problems [23]; an approach based on swarm neural networks was proposed to equalities-constrained nonconvex optimization [24]; a collective neurodynamic approach based on neural networks was proposed to global optimization [25]; a collaborative neurodynamic approach was proposed for solving multiobjective distributed optimization [26]; a discrete-time projection neural network was proposed to nonnegative matrix factorization [27]; a duplex neurodynamic approach was proposed to biconvex optimization [28]; a neurodynamic approach was proposed to nonsmooth constrained pseudoconvex optimization [29]. These studies showed that optimization approaches based on neural networks could conduct precise local search with the nature of parallel and distribution in information processing.

Based on the above discussions, this paper proposes an algorithm for solving MO problems which hybridizes MOEA/D and multiple recurrent neural networks. A bilevel hierarchy is designed so that MOEA/D and recurrent neural networks work in a cooperative manner. For each iteration, MOEA/D generates a set of non-dominated solutions at the upper level, then multiple recurrent neural networks are employed to conduct precise local search in parallel at the lower level. As a result, MOEA/D and recurrent neural networks work collaboratively for a set of POSs.

The remainder of this paper is organized as follows: Section II introduces some preliminary concepts on MO, MOEA/D and recurrent neural network. Section III presents a hybridized algorithm for MO. Section IV presents the performance comparisons which show the proposed algorithm outperformed the

The work described in this paper was fully supported by the Open University of Hong Kong Research Grant (No. R5083).

978-1-7281-6929-3/20/$31.00 ©2020 IEEE

compared algorithms regarding sixteen test problems. Section V is the conclusion.

II. Preliminaries

A. Multi-Objective Optimization

Minimizing an MOP with m objectives can be defined as:

$$\min F(x) = (f_1(x), f_2(x), ..., f_m(x))^T \tag{1}$$

where $x = (x_1, x_2, ..., x_n)^T \in \mathbb{R}^n$ is an n-dimensional set of decision variables.

During the optimization process of an MOP, sets of solutions will be generated. Among these solutions, some of them are non-dominated. A non-dominated solution has the property that none of its objective values can be improved further without worsening its other objective values, and it is called POS if no other solution dominates it. Solution μ dominates solution μ', denoted by $\mu \prec \mu'$, if and only if $f_j(\mu) \leq f_j(\mu')$ for $j = 1, 2, \ldots, m$. All the POSs for a specific MOP form a Pareto front. The aim of optimizing an MOP is to obtain a set of non-dominated solutions that are as close as possible to the Pareto front and as diverse as possible.

Scalarization is a popular approach for multiobjective optimization as an MOP is converted to various subproblems according to a set of predefined weights so that these subproblems can be optimized simultaneously using meta-heuristics such as EAs [30]. For example, MOGLS [13], C-MOGA [31] and MOEA/D [7] are the multiobjective algorithms based on scalarization. Two main decomposition methods are weighted sum (WS) and weighted Chebyshev (CH) [32]:

1) WS: It combines the objectives according to the weight vectors. For the i-th weight vector $\lambda^i = (\lambda^i_1, \lambda^i_2, \ldots, \lambda^i_m)^T$, where $0 \leq \lambda^i_j \leq 1$ and $\sum_{j=1}^m \lambda^i_j = 1$, the POS corresponds to the i-th subproblem is then calculated by:

$$f_{\text{WS}}(x|\lambda^i) = \sum_{j=1}^{m} \lambda^i_j f_j(x). \tag{2}$$

2) CH: Let $r^* = (r^*_1, r^*_2, \ldots, r^*_m)^T$ be a reference vector , and $r^*_j < \min\{f_j(x)|x \in \Omega\}$. The i-th subproblem is optimized by:

$$f_{\text{CH}}(x|\lambda^i) = \max_{1\leq j\leq m} \left\{\lambda^i_j |f_j(x) - r^*_j|\right\}. \tag{3}$$

The WS approach is effective for MOPs with convex Pareto fronts in minimization [30]. On the other hand, the CH approach works well for MOPs with non-convex Pareto fronts. The CH approach is used in this paper. However, f_{CH} is non-differentiable and it can be reformulated as:

$$\begin{aligned} \min\ & \xi \\ \text{s.t. } & \lambda^i_j(f_j(x) - r^*_j) - \xi \leq 0, \\ & \xi \in \mathbb{R} \end{aligned} \tag{4}$$

where $\xi = \max_{1\leq j\leq m} \left\{\lambda^i_j |f_j(x) - r^*_j|\right\}$.

B. MOEA/D

As one of the major works in multiobjective optimization, the MOEA/D provides a formal framework [7] for solving various MOPs efficiently. A set of evenly distributed weight vectors $\{\lambda^1, \lambda^2, \ldots, \lambda^N\}$ is adopted in scalarization for converting N subproblems. Then these subproblems are optimized simultaneously. In the framework, M is the neighborhood number of λ_i for all i, and the neighboring subproblems are considered for the i-th subproblem. A small value of M may weaken the exploration while large value of M may weaken the exploitation of the algorithm. Besides, two solutions of the neighboring subproblems are selected randomly from the i-th subproblem and offspring is generated by genetic operators. Then the offspring is used to update the neighboring solution. After meeting the stopping criteria, a set of solutions are generated.

C. Projection Neural Network

Consider a constrained optimization problem which is single-objective:

$$\begin{aligned} \min\ & f_c(x) \\ \text{s.t. } & g(x) \leq 0, \\ & \underline{x} \leq x \leq \overline{x} \end{aligned} \tag{5}$$

where $f_c(x)$ is the objective function, $g(x)$ is the inequality constraint, $\underline{x}$, $\overline{x}$ are lower and upper bound respectively.

Assume both $f_c(x)$ and $g(x)$ are convex, a two-layer projection neural network (PNN) is adopted to solve (5) and obtain the global optimum [23], [33]:

$$\begin{cases} \epsilon \dfrac{dx}{dt} = -x + P[x - \nabla f_c(x) - \nabla g(x)\beta] \\ \epsilon \dfrac{d\beta}{dt} = -\beta + [\beta + g(x)]^+ \end{cases} \tag{6}$$

where ϵ is a positive time constant, x is the state vector, $\nabla f_c(x)$ and $\nabla g(x)$ are the gradients of $f_c(x)$ and $g(x)$, $[x]^+ = \max\{x, 0\}$ and $P(x)$ is a piecewise linear activation function:

$$P(x) = \begin{cases} \overline{x}, & x > \overline{x} \\ x, & \underline{x} \leq x \leq \overline{x}. \\ \underline{x}, & x < \underline{x} \end{cases}$$

III. Proposed Hybrid Algorithm for Multi-Objective Optimization

In this section, a hybrid algorithm based on MOEA/D and PNN is presented. A bilevel hierarchy is designed for the algorithm so that the global and local search algorithms work in a cooperative manner: MOEA/D carries out global search at the upper level and the tentative solutions are passed to PNN for refinement (local search) at the lower level.

To conduct precise local search using PNNs, (6) is modified as follows:

$$\begin{cases} \epsilon \dfrac{d\alpha}{dt} = -\alpha + P[\alpha - e_{n+1} - \nabla g(\alpha)\beta] \\ \epsilon \dfrac{d\beta}{dt} = -\beta + [\beta + g(\alpha)]^+ \end{cases} \tag{7}$$

where $\alpha = (x^T, \xi)^T \in \mathbb{R}^{n+1}$, $e_{n+1} = (0, 0, \dots, 0, 1)^T \in \mathbb{R}^{n+1}$, $g(\alpha) = (f_j(x) - r_j^* - \xi)^T$ for $j = 1, \dots, m$.

Algorithm 1 shows the pseudocode of the proposed algorithm. During the initialization stage, N weight vectors are generated uniformly [10], [34], the M closest weight vectors for each λ^i are determined, the position of each population is initialized randomly within the boundaries, and the reference vector is also initialized. After that, the algorithm keeps executing the following procedure until meeting the stopping criterion. For each population x^i, $i = 1, \dots, N$, two indexes (denoted as k and l) are selected randomly from $B(i)$ to generate a new solution using genetic operators. Next, a repair heuristic is applied on the newly generated solution. After that, PNN is used to conduct precise local search. Then the reference vector and the neighboring solutions are updated. After optimizing all the subproblems, the corresponding solutions are considered as members of an external archive with a limited capacity. Note that only the non-dominated solutions are kept in the external archive [35]. When meeting the stopping criterion, the members in the archive is regards as the output.

Algorithm 1 Proposed Hybrid Algorithm for Multiobjective Optimization

```
1:  Generate {λ^1, λ^2, ..., λ^N}.
2:  For each λ^i, set B(i) = {i_1, ..., i_M}, where
    λ^{i_1}, ..., λ^{i_M}, determine the M closest weight vectors.
3:  Initialize the population {x^1, ..., x^N} randomly within
    the search space.
4:  Initialize the reference vector r*.
5:  for each x^i do
6:      Select k and l randomly from B(i) to generate a new
        solution by using genetic operators.
7:      Apply a repair heuristic on the newly generated
        solution.
8:      Conduct precise local search using (7).
9:      Update r*.
10:     Update the neighboring solutions.
11: end for
12: Update external archive.
13: if the stopping criteria is not met do
14:     go to 5.
15: else
16:     Output the solutions in the external archive.
17: end if
```

IV. Performance Comparisons

This section introduces the performance measures used in the performance comparison, and compares the proposed algorithm with other selected algorithms.

A. Performance Measures

In this study, inverted generational distance (IGD) and hypervolume (HV) are considered to measure the quality of the generated solutions. They are the two most widely used indicators in the literature. Both HV and IGD values measure the diversity and convergence of sets of approximated Pareto solutions. The larger of the HV value obtained, the better quality of the obtained solutions. On the other hand, a smaller IGD value means a better set of the obtained solutions.

1) IGD [36]: Let PF^* be a set of uniformly distributed points sampled from the true Pareto front and AP is the set of approximated Pareto solutions. The metric is defined as:
$$\text{IGD}(AP, PF^*) = \frac{\sum_{pf \in PF^*} dist(pf, AP)}{|PF^*|} \tag{8}$$
where $dist(pf, AP)$ returns the minimum Euclidean distance between pf and points in AP.

2) HV [37]: Let $r = (r_1, ..., r_m)^T$ be a reference point dominated by other Pareto solutions. The HV value of AP is the non-overlapped region of all the hypercubes formed by the reference point and each stored solution a in AP. The metric is defined as:
$$\text{HV}(AP) = L\left(\bigcup_{a \in AP} [f_1(a), r_1] \times \ldots \times [f_m(a), r_m] \right) \tag{9}$$
where L is the Lebesgue measure [38].

B. Experimental Settings

Three popular multi-objective algorithms are chosen for performance comparison. The algorithms are MOEA/D [7], MOEA/D-DE [8] and HMOEA/D [16]. The implementations of the algorithms are coded in Matlab. Five bi-objective and eleven tri-objective benchmarks are used. The bi-objective benchmarks are ZDT1 to ZDT4 and ZDT6 [39] while the tri-objective benchmarks are DTLZ1 to DTLZ6 [40], and MaF1 to MaF5 [41]. 51 runs are conducted for each algorithm on every optimization problem. For bi-objective benchmarks, each of the algorithms stops at 30000 fitness evaluation. For tri-objective benchmarks, each of the algorithms stops at 200000 fitness evaluation. For ZDT optimization problems, $r = (2, 2)^T$. For DTLZ1 to DTLZ6, and MaF1 to MaF3, $r = (2, 2, 2)^T$. For MaF4, $r = (3, 6, 12)^T$. For MaF5, $r = (12, 6, 3)^T$. The population size of each algorithm is set to 100. For MOEA/D, HMOEA/D and the proposed algorithm, the crossover rate is set to 1, the distribution index is set to 20, and the mutation rate is set to $1/n$. For MOEA/D-DE, the distribution index and the mutation rate of the polynomial mutation operator are set to 20 and $1/n$, the control parameters CR and F in DE are set to 1.0 and 0.5. Besides, M is set to 20 for all the compared algorithms.

C. Experimental Results

Figures 1 to 3 show the non-dominated solutions of the median run produced by the compared algorithms. It is found that the proposed algorithm can generate sets of solutions with better convergence and diversity generally. Tables I and II show the mean HV and IGD values of the approximated solutions by each of the compared algorithms respectively, where the best mean is shown in **boldface**. Besides, the standard deviations are in round brackets.

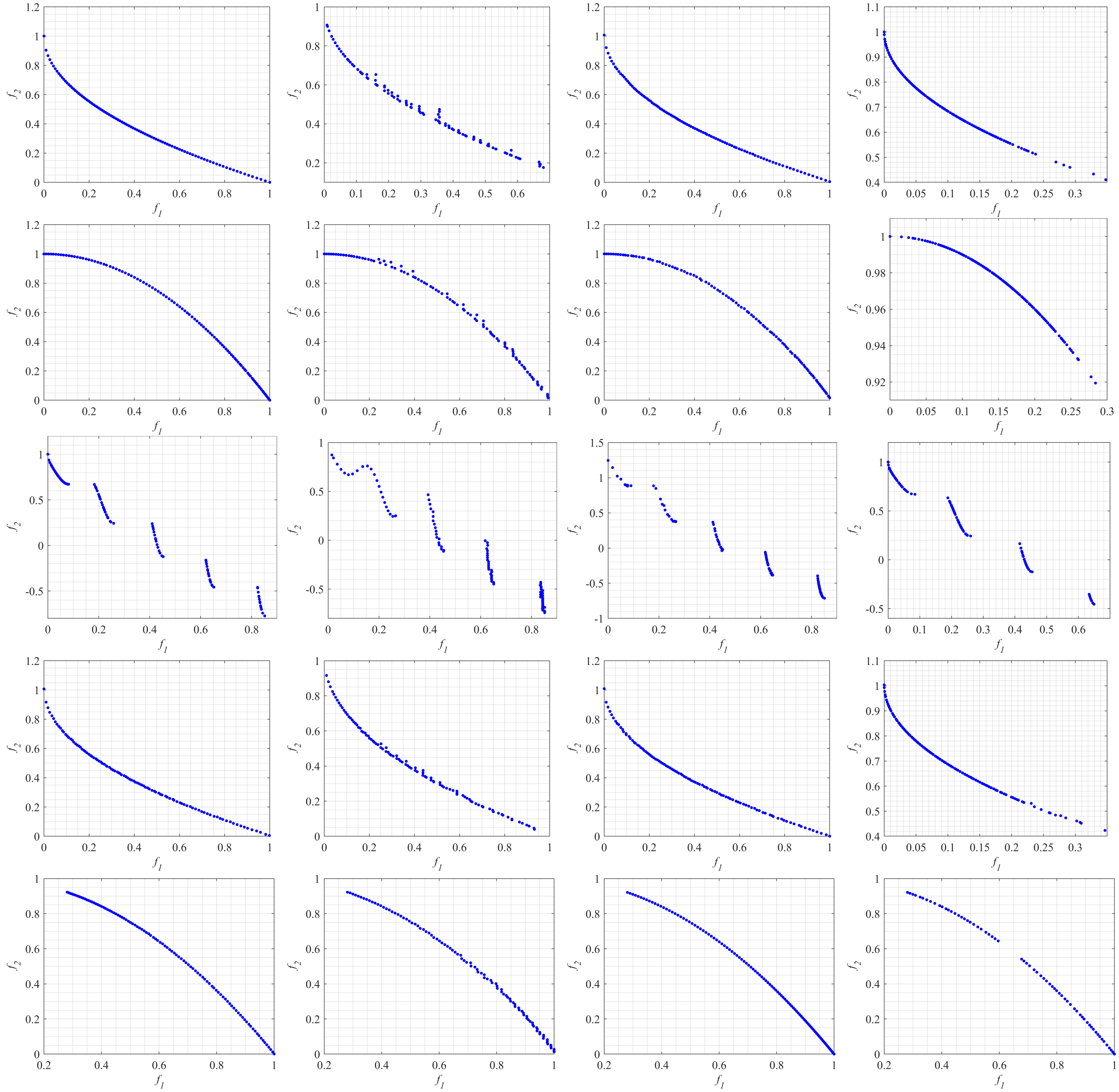

Fig. 1. Solutions (the median run) generated by the proposed algorithm (subplots of the 1st column), MOEA/D (subplots of the 2nd column), MOEA/D-DE (subplots of the 3rd column) and HMOEA/D (subplots of the last column) on ZDT1 (subplots of the 1st row), ZDT2 (subplots of the 2nd row), ZDT3 (subplots of the 3rd row), ZDT4 (subplots of the 4th row) and ZDT6 (subplots of the last row).

1) Results of ZDT test problems: For ZDT1 and ZDT4, the corresponding Pareto fronts are convex. For ZDT2 and ZDT6, the Pareto fronts are concave. Besides, the Pareto front of ZDT3 is disconnected. Tables I and II show the proposed approach outperforms the others in the ZDT benchmarks.

2) Results of DTLZ test problems: In view of DTLZ benchmarks, the proposed algorithm obtains 10 best mean results (out of 12), while MOEA/D obtains 2. Regarding the HV, the proposed algorithm outperforms others in all the test problems among other compared algorithms except DTLZ1. Regarding the IGD, the proposed algorithm obtained better mean results in all the test problems among other compared algorithms except DTLZ3.

DTLZ1 has the property of multiple local Pareto fronts which makes it as a hard-to-converge optimization problem. Regarding the HV, MOEA/D obtains the largest mean value, while the proposed algorithm outperforms HMOEA/D and MOEA/D-DE. Besides, the proposed algorithm obtains the

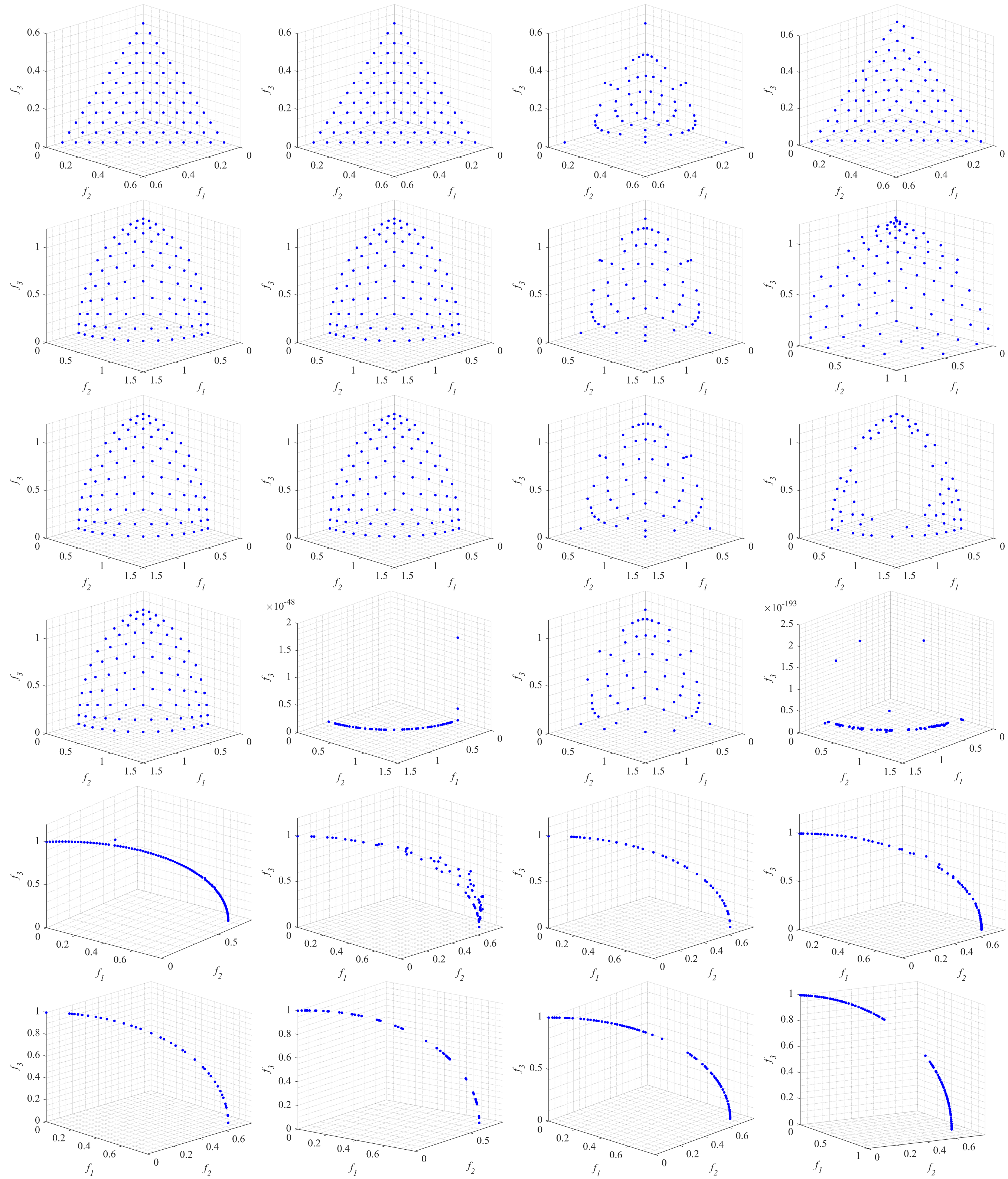

Fig. 2. Solutions (the median run) generated by the proposed algorithm (subplots of the 1st column), MOEA/D (subplots of the 2nd column), MOEA/D-DE (subplots of the 3rd column) and HMOEA/D (subplots of the last column) on DTLZ1 (subplots of the 1st row), DTLZ2 (subplots of the 2nd row), DTLZ3 (subplots of the 3rd row), DTLZ4 (subplots of the 4th row), DTLZ5 (subplots of the 5th row) and DTLZ6 (subplots of the last row).

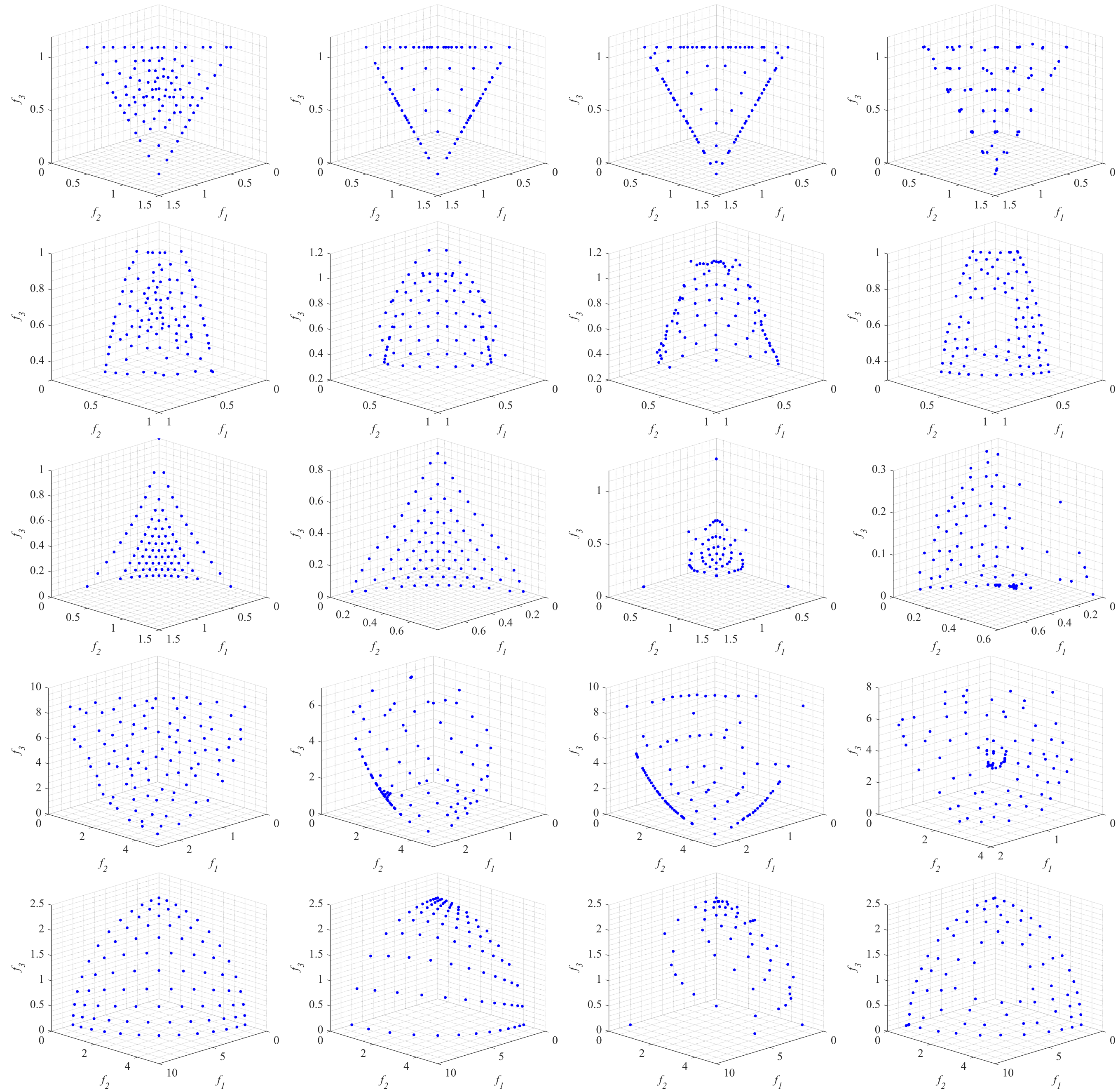

Fig. 3. Solutions (the median run) generated by the proposed algorithm (subplots of the 1st column), MOEA/D (subplots of the 2nd column), MOEA/D-DE (subplots of the 3rd column) and HMOEA/D (subplots of the last column) on MaF1 (subplots of the 1st row), MaF2 (subplots of the 2nd row), MaF3 (subplots of the 3rd row), MaF4 (subplots of the 4th row) and MaF5 (subplots of the last row).

smallest mean IGD value, while both the mean IGD values of the generated solutions by MOEA/D and the proposed algorithm are very close. Result shows that MOEA/D is competitive for solving DLTZ1. The Pareto front of DTLZ2 is concave. Regarding both HV and IGD, the proposed algorithm can generate solutions with the best mean values. The Pareto front of DTLZ3 is concave and it is multi-modal in nature. The proposed algorithm obtains the largest mean HV value while MOEA/D obtains the smallest mean IGD value. Indicating that MOEA/D has the advantage of handling optimization problems with multiple local Pareto fronts. Besides, the local search of the proposed algorithm may weaken its global search ability. The Pareto front of DTLZ4 is concave and it is uni-modal in nature. The proposed algorithm outperforms the compared algorithms regarding the HV and IGD values of the generated solutions. Both DTLZ5 and DTLZ6 have degenerated Pareto optimal fronts, the proposed algorithm also outperforms the compared algorithms.

TABLE I
PERFORMANCE COMPARISON IN TERMS OF HYPERVOLUME

Problem	m	Proposed algorithm	MOEA/D	MOEA/D-DE	HMOEA/D
ZDT1	2	**9.1430e-1** (6.69e-4)	8.9672e-1 (2.22e-2)	9.1211e-1 (1.65e-3)	8.9420e-1 (5.01e-2)
ZDT2	2	**8.3185e-1** (1.06e-3)	7.9168e-1 (6.81e-2)	8.2387e-1 (2.94e-3)	7.9488e-1 (4.48e-3)
ZDT3	2	**9.4923e-1** (3.52e-4)	9.4460e-1 (1.28e-3)	9.4241e-1 (7.42e-3)	9.4851e-1 (3.83e-4)
ZDT4	2	**9.1128e-1** (2.50e-3)	8.1696e-1 (4.63e-3)	8.1149e-1 (9.48e-2)	8.6027e-1 (1.32e-2)
ZDT6	2	**7.6054e-1** (6.54e-4)	7.5559e-1 (1.03e-3)	7.5963e-1 (2.01e-3)	7.5800e-1 (1.42e-3)
DTLZ1	3	9.9676e-1 (1.15e-4)	**9.9682e-1** (1.17e-4)	9.9590e-1 (2.08e-4)	9.4521e-1 (4.60e-2)
DTLZ2	3	**9.2648e-1** (8.45e-4)	9.2485e-1 (7.32e-4)	9.2107e-1 (6.61e-4)	9.2462e-1 (9.68e-4)
DTLZ3	3	**9.2632e-1** (9.59e-4)	9.2512e-1 (6.56e-4)	9.2059e-1 (5.09e-4)	9.1858e-1 (3.86e-3)
DTLZ4	3	**9.3030e-1** (6.73e-3)	8.3758e-1 (1.39e-1)	9.1878e-1 (4.65e-3)	9.1393e-1 (7.28e-4)
DTLZ5	3	**7.9012e-1** (1.08e-3)	7.2394e-1 (1.47e-3)	7.6031e-1 (1.37e-3)	7.5885e-1 (2.27e-3)
DTLZ6	3	**7.8823e-1** (1.00e-3)	7.5422e-1 (1.83e-3)	7.6023e-1 (1.15e-3)	7.5667e-1 (6.27e-3)
MaF1	3	**6.9851e-1** (8.90e-4)	6.8700e-1 (7.65e-4)	6.9023e-1 (1.06e-3)	6.3086e-1 (6.49e-3)
MaF2	3	**7.8251e-1** (2.70e-3)	6.4833e-1 (1.30e-3)	6.4484e-1 (1.55e-3)	7.2741e-1 (1.19e-3)
MaF3	3	**9.9851e-1** (1.45e-4)	9.8712e-1 (2.68e-4)	9.8791e-1 (5.44e-4)	9.1030e-1 (6.81e-3)
MaF4	3	**7.5393e-1** (3.50e-3)	6.8317e-1 (2.78e-3)	6.9851e-1 (2.48e-3)	6.7710e-1 (1.05e-2)
MaF5	3	**8.2184e-1** (8.75e-2)	7.0534e-1 (2.04e-1)	8.0306e-1 (3.99e-3)	7.7135e-1 (1.54e-1)

TABLE II
PERFORMANCE COMPARISON IN TERMS OF IGD

Problem	m	Proposed algorithm	MOEA/D	MOEA/D-DE	HMOEA/D
ZDT1	2	**4.3994e-3** (5.79e-4)	1.9643e-2 (2.51e-2)	6.2213e-3 (1.80e-3)	6.1047e-3 (1.18e-3)
ZDT2	2	**4.0248e-3** (2.00e-4)	2.6802e-2 (4.29e-2)	8.9614e-3 (1.84e-3)	8.3455e-3 (1.67e-3)
ZDT3	2	**1.1591e-2** (1.29e-2)	2.2162e-2 (1.31e-2)	2.4432e-2 (2.12e-2)	1.9359e-2 (4.95e-3)
ZDT4	2	**6.6369e-3** (1.68e-3)	2.5653e-2 (1.66e-2)	1.3638e-1 (1.20e-1)	3.7207e-2 (3.43e-3)
ZDT6	2	**5.2907e-3** (2.31e-4)	8.0105e-3 (4.27e-4)	6.1110e-3 (1.58e-5)	8.6582e-3 (2.82e-3)
DTLZ1	3	**2.0566e-2** (8.16e-6)	2.0570e-2 (1.38e-5)	3.0734e-2 (1.03e-4)	6.3352e-2 (1.02e-3)
DTLZ2	3	**5.3464e-2** (1.09e-7)	5.4464e-2 (8.27e-8)	7.5394e-2 (3.33e-4)	6.9179e-2 (3.51e-3)
DTLZ3	3	5.4511e-2 (4.18e-5)	**5.4496e-2** (3.64e-5)	7.5371e-2 (3.08e-4)	9.7009e-2 (1.07e-2)
DTLZ4	3	**1.7093e-1** (2.57e-2)	3.1587e-1 (3.34e-1)	2.3989e-1 (2.05e-1)	2.7910e-1 (1.90e-2)
DTLZ5	3	**1.2836e-2** (1.75e-3)	3.3922e-2 (3.05e-6)	1.4583e-2 (3.65e-5)	1.3404e-2 (4.29e-3)
DTLZ6	3	**1.3656e-2** (2.85e-3)	3.3928e-2 (6.10e-7)	1.4617e-2 (6.58e-6)	1.8192e-2 (1.54e-2)
MaF1	3	**6.0335e-2** (1.74e-3)	7.0475e-2 (1.17e-7)	6.6213e-2 (1.13e-4)	6.3186e-2 (2.58e-3)
MaF2	3	**3.4817e-2** (5.32e-4)	3.8493e-2 (3.95e-4)	4.5222e-2 (5.85e-4)	3.6457e-2 (1.28e-3)
MaF3	3	**3.6689e-2** (4.86e-4)	5.4419e-2 (2.70e-4)	1.0821e-1 (8.40e-4)	1.4408e-1 (7.47e-3)
MaF4	3	**3.3912e-1** (9.74e-3)	6.5661e-1 (1.38e-2)	5.8808e-1 (1.25e-2)	5.7941e-1 (6.69e-2)
MaF5	3	**3.8467e-1** (6.99e-2)	1.3358e+0 (1.91e+0)	4.7640e-1 (1.44e-2)	7.6715e-1 (1.45e+0)

3) Results of MaF test problems: With regard to MaF benchmarks, it is found that the proposed algorithm outperforms the compared algorithms.

The Pareto front of MaF1 is inverted compared to the one of DTLZ1. Figure 3 shows that the proposed algorithm can generated well-distributed solutions. MaF2 has different scaling functions based on DTLZ2. Regarding both HV and IGD, the proposed algorithm can generate solutions with the best mean values in MaF2. MaF3 is used to verify the whether a multiobjective algorithm can handle convex Pareto fronts. Although the proposed algorithm outperforms other algorithms, results show that MOEA/D is competitive in solving MaF3. Similar to DTLZ3, MaF4 consists of multiple local Pareto fronts while they are inverted. Results show that the proposed algorithm can approximate to the Pareto front. MaF5 consists of a badly scaled Pareto front based on DTLZ4. Results show that the proposed algorithm can generate well-distributed solutions with high convergence.

V. CONCLUSIONS

In this paper, a hybrid algorithm is proposed for solving multi-objective optimization problems. The algorithm consists of MOEA/D and projection neural networks which combines the advantages of the two optimizers. Besides, the algorithm is designed in a bilevel hierarchy, where MOEA/D aims for global search at the upper level and multiple projection neural networks conduct precise local search at the lower level. Experimental results show that the proposed algorithm outperforms the compared algorithms in terms of and hypervolume and IGD. Future work may focus on extending the proposed algorithm to many-objective optimization.

References

[1] J. L. Cohon, *Multiobjective Programming and Planning*. Courier Corporation, 2013.

[2] R. E. Steuer, *Multiple Criteria Optimization: Theory, Computation, and Applications*. Wiley, 1986.

[3] G. R. Harik, F. G. Lobo, and D. E. Goldberg, "The compact genetic algorithm," *IEEE Trans. Evol. Comput.*, vol. 3, no. 4, pp. 287–297, 1999.

[4] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, no. 4598, pp. 671–680, 1983.

[5] F. Glover and E. Taillard, "A user's guide to tabu search," *Ann. Oper. Res.*, vol. 41, no. 1, pp. 1–28, 1993.

[6] C. C. Coello, "Evolutionary multi-objective optimization: a historical view of the field," *IEEE Comput. Intell. Mag.*, vol. 1, no. 1, pp. 28–36, 2006.

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

[8] H. Li and Q. Zhang, "Multiobjective optimization problems with complicated pareto sets, MOEA/D and NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 13, no. 2, pp. 284–302, 2009.

[9] H.-L. Liu, F. Gu, and Q. Zhang, "Decomposition of a multiobjective optimization problem into a number of simple multiobjective subproblems," *IEEE Trans. Evol. Comput.*, vol. 18, no. 3, pp. 450–455, 2013.

[10] Z. Wang, Q. Zhang, A. Zhou, M. Gong, and L. Jiao, "Adaptive replacement strategies for MOEA/D," *IEEE Trans. Cybern.*, vol. 46, no. 2, pp. 474–486, 2016.

[11] Z. Wang, Q. Zhang, H. Li, H. Ishibuchi, and L. Jiao, "On the use of two reference points in decomposition based multiobjective evolutionary algorithms," *Swarm and evolutionary computation*, vol. 34, pp. 89–102, 2017.

[12] J. Shi, Q. Zhang, and J. Sun, "PPLS/D: Parallel pareto local search based on decomposition," *IEEE Trans. Cybern.*, 2018.

[13] H. Ishibuchi and T. Murata, "A multi-objective genetic local search algorithm and its application to flowshop scheduling," *IEEE Trans. Syst., Man, Cybern. C*, vol. 28, no. 3, pp. 392–403, 1998.

[14] J. D. Knowles and D. W. Corne, "M-PAES: A memetic algorithm for multiobjective optimization," in *Proc. IEEE Congr. Evol. Comput.*, vol. 1. IEEE, 2000, pp. 325–332.

[15] A. Lara, G. Sanchez, C. A. C. Coello, and O. Schutze, "Hcs: A new local search strategy for memetic multiobjective evolutionary algorithms," *IEEE Trans. Evol. Comput.*, vol. 14, no. 1, pp. 112–132, 2009.

[16] K. Sindhya, K. Miettinen, and K. Deb, "A hybrid framework for evolutionary multi-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 17, no. 4, pp. 495–511, 2012.

[17] L. Ke, Q. Zhang, and R. Battiti, "Hybridization of decomposition and local search for multiobjective optimization," *IEEE Trans. Cybern.*, vol. 44, no. 10, pp. 1808–1820, 2014.

[18] Y. Lei, J. Shi, and Z. Yan, "A memetic algorithm based on MOEA/D for the examination timetabling problem," *Soft Comput.*, vol. 22, no. 5, pp. 1511–1523, 2018.

[19] H. Che and J. Wang, "A collaborative neurodynamic approach to global and combinatorial optimization," *Neural Netw.*, vol. 114, pp. 15–27, 2019.

[20] J. J. Hopfield, D. W. Tank *et al.*, "Computing with neural circuits- a model," *Science*, vol. 233, no. 4764, pp. 625–633, 1986.

[21] D. Tank and J. Hopfield, "Simple'neural'optimization networks: An A/D converter, signal decision circuit, and a linear programming circuit," *IEEE Trans. Circuits Syst.*, vol. 33, no. 5, pp. 533–541, 1986.

[22] J. Wang, "A recurrent neural network for solving the shortest path problem," *IEEE Trans. Circuits Syst. I*, vol. 43, no. 6, pp. 482–486, 1996.

[23] Y. Xia, H. Leung, and J. Wang, "A projection neural network and its application to constrained optimization problems," *IEEE Trans. Circuits Syst. I*, vol. 49, no. 4, pp. 447–458, 2002.

[24] H. Che, C. Li, X. He, and T. Huang, "An intelligent method of swarm neural networks for equalities-constrained nonconvex optimization," *Neurocomputing*, vol. 167, pp. 569–577, 2015.

[25] Z. Yan, J. Fan, and J. Wang, "A collective neurodynamic approach to constrained global optimization," *IEEE Trans. Neural Netw. Learn. Syst.*, 2016.

[26] S. Yang, Q. Liu, and J. Wang, "A collaborative neurodynamic approach to multiple-objective distributed optimization," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 4, pp. 981–992, 2017.

[27] H. Che and J. Wang, "A nonnegative matrix factorization algorithm based on a discrete-time projection neural network," *Neural Netw.*, vol. 103, pp. 63–71, 2018.

[28] ——, "A two-timescale duplex neurodynamic approach to biconvex optimization," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 8, pp. 2503–2514, 2018.

[29] C. Xu, Y. Chai, S. Qin, Z. Wang, and J. Feng, "A neurodynamic approach to nonsmooth constrained pseudoconvex optimization problem," *Neural Netw.*, vol. 124, pp. 180–192, 2020.

[30] A. Trivedi, D. Srinivasan, K. Sanyal, and A. Ghosh, "A survey of multiobjective evolutionary algorithms based on decomposition," *IEEE Trans. Evol. Comput.*, 2016.

[31] T. Murata and M. Gen, "Cellular genetic algorithm for multi-objective optimization," in *Proc. of the 4th Asian Fuzzy System Symposium*. Citeseer, 2002, pp. 538–542.

[32] K. Miettinen, *Nonlinear Multiobjective Optimization*. Kluwer Academic Publishers, Boston, 1999.

[33] Y. Xia and J. Wang, "On the stability of globally projected dynamical systems," *J. Optim. Theory Appl.*, vol. 106, no. 1, pp. 129–150, 2000.

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

[35] N. Padhye, J. Branke, and S. Mostaghim, "Empirical comparison of mopso methods-guide selection and diversity preservation," in *Proc. IEEE Congr. Evol. Comput.* IEEE, 2009, pp. 2516–2523.

[36] P. A. Bosman and D. Thierens, "The balance between proximity and diversity in multiobjective evolutionary algorithms," *IEEE Trans. Evol. Comput.*, vol. 7, no. 2, pp. 174–188, 2003.

[37] M. Emmerich, N. Beume, and B. Naujoks, "An EMO algorithm using the hypervolume measure as selection criterion," in *Proc. Conf. Evol. Multi-Criterion Optimization*. Springer, 2005, pp. 62–76.

[38] M. Fleischer, "The measure of pareto optima: Applications to multi-objective metaheuristics," in *Proc. 2nd Int. Conf. Evol. Multi-Criterion Optimization*. Springer, 2003, pp. 519–533.

[39] E. Zitzler, K. Deb, and L. Thiele, "Comparison of multiobjective evolutionary algorithms: Empirical results," *Evol. Comput.*, vol. 8, no. 2, pp. 173–195, 2000.

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

[41] R. Cheng, M. Li, Y. Tian, X. Zhang, S. Yang, Y. Jin, and X. Yao, "A benchmark test suite for evolutionary many-objective optimization," *Complex Intell. Syst.*, vol. 3, no. 1, pp. 67–81, 2017.