Fusion-based Hybrid Many-objective Optimization Algorithm

Amin Ibrahim, IEEE Member
Faculty of Electrical, Computer, and Software Engineering
University of Ontario Institute of Technology
Oshawa, Canada
amin.ibrahim@uoit.ca

Shahryar Rahnamayan, SMIEEE
Faculty of Electrical, Computer, and Software Engineering
University of Ontario Institute of Technology
Oshawa, Canada
shahryar.rahnamayan@uoit.ca

Miguel Vargas Martin, IEEE Member
Faculty of Business and IT
University of Ontario Institute of Technology
Oshawa, Canada
miguel.vargasmartin@uoit.ca

Kalyanmoy Deb, FIEEE
Faculty of Electrical, Computer, and Computer Engineering
Michigan State University
East Lansing, Michigan, USA
kdeb@egr.msu.edu

Abstract*— In the last three decades there have been a number of efficient multi-objective optimization algorithms capable of solving real-world problems. However, due to the complexity of most real-world problems (high-dimensionality of problems, computationally expensive, and unknown function properties) researchers and decision-makers are increasingly facing the challenge of selecting an optimization algorithm capable of solving their hard problems. In this paper, we propose a simple yet efficient hybridization of multi- and many-objective optimization algorithms framework called hybrid many-objective optimization algorithm using fusion of solutions obtained by several many-objective algorithms (fusion) to gain the combined benefits of several algorithms and reducing the challenge of choosing one optimization algorithm to solve complex problems. During the optimization process, the Fusion framework (1) executes all optimization algorithms in parallel, (2) it combines solutions of these algorithms and extracts well-distributed solutions using predefined structured reference points or user-defined reference points, and (3) adaptively selects best-performing algorithm to tackle the problem at different stages of the search process. A case study of the fusion framework by considering GDE3, SMPSO, and SPEA2 as multi-objective optimization algorithms is presented. Experimental results on five unconstrained and four constrained benchmark test problems with three to ten objectives show that the Fusion framework significantly outperforms all algorithms involved in the hybridization process as well as the NSGA-III algorithm in terms of diversity and convergence of obtained solutions. Furthermore, the proposed framework is consistently able to find accurate solutions for all test problems which can be interpreted as its high robustness characteristic.

Keywords— Hybrid optimization; Algorithms fusion; GDE3, SPEA2, SMPSO; NSGA-III, Many-objective optimization; Reference-point-based optimization, Evolutionary computation.

I. INTRODUCTION

There have been numerous multi-objective algorithms that attempt to solve complex problems in the past thirty years. Nonetheless, researchers and decision-makers are increasingly faced with the difficulty of choosing an appropriate algorithm capable of solving their problem in an effective manner due to a well-established “No-Free-Lunch” theorem [1]. This theorem stipulates that an algorithm that may have proven to give good performance on a particular class of problems may not provide the same level of performance on other classes of problems. Consequently, researchers shifted their focus instead to developing powerful algorithms that are more problem-specific or instance-specific. This task can be accomplished via the hybridization of optimization algorithms where new algorithms are developed by combining two optimization algorithms. Another way to accomplish this is by combining mathematical methods with an evolutionary optimization algorithm. Yet, another method incorporates evolutionary operators (selection, mutation, and crossover) into non-evolutionary optimization algorithms [2]. The expectation here is that hybridization combines the desirable properties of different approaches such that the hybrid algorithm exhibits improved exploration and exploitation capabilities.

For instance, a hybrid multi-objective evolutionary algorithm called HMOEA was proposed by Tan et a1. [3]. They utilized specialized genetic operators along with variable-length representation and a local search heuristic to find the Pareto optimal routing solutions for the truck and trailer vehicle routing problem (TTVRP). Results from experiments showed that the HMOEA is effective in solving multi-objective and multi-modal combinatorial optimization problems. Xia and Wu [4] similarly proposed a hybrid multi-objective algorithm by combining the PSO algorithm for its explorative power, and simulated annealing (SA) for its exploitations to solve the flexible job-shop scheduling problem (FJSP). A hybrid multi-objective algorithm based on the features of a biological immune system (IS) and bacterial optimization (BO) to find Pareto optimal solutions for flow shop scheduling problem was proposed by Tavakkoli-Moghaddam et al. [5]. This particular algorithm uses the clonal selection principle in IS with highest affinity antibodies and criterion distinguishing between antigens and antibodies in BO for the Pareto dominance relationship among solutions. Also, Karthikeyan et al. [6] proposed a hybrid discrete firefly algorithm (HDFA) to solve the multi-objective FJSP problem. In this proposal, the discrete firefly algorithm and a local search

978-1-5090-4601-0/17/$31.00 ©2017 IEEE

(LS) method were combined to enhance the searching accuracy and information sharing among fireflies.

Another work by Wang et al. [7] proposed a hybrid evolutionary algorithm that is based on different crossover and mutation strategies along with adaptive constrained-handling technique to deal with numerical and engineering constrained optimization problems. A hybrid and adaptive co-evolutionary optimization method that can efficiently solve a wide range of multi-objective optimization problems was proposed by Zăvoianu et al. [8]. This particular approach combines Pareto-based selection for survival, differential evolution's crossover and mutation operators, and decomposition-based strategies. An ensemble strategy was recently proposed to benefit from both the availability of diverse approaches and to overcome difficulties associated with fine tuning associated parameters. Such work includes ensemble of the ε parameter values and an ensemble of external archives in a multi-objective PSO algorithm [9], constraint handling methods to tackle constrained multi-objective optimization problems [10], and the various neighborhood sizes in multi-objective evolutionary algorithm based on decomposition (MOEA/D) with online self-adaptation [11]. Other notable hybridization of selection mechanisms include selection hyper-heuristics (mixing selection, mutation operators and accepting strategies) [21] and bi-criterion evolution (hybridization of Pareto-based and non-Pareto-based selection criterions) [22].

A hybrid population-based algorithm, called PSOGSA, was proposed by Mirjalili and Hashim [12] by combining Particle Swarm Optimization (PSO) and the Gravitational Search Algorithm (GSA). The aim of the algorithm was the integration of the exploitation ability of PSO with the exploration ability of GSA to synthesize the strengths of both algorithms. Also, El-hossini et al. [13] proposed three hybrid algorithms that are based on the strength Pareto evolutionary algorithm 2 (SPEA2) and the PSO to solve multi-objective optimization problems. These algorithms use strength Pareto fitness assignment to maintain an external archive. The three algorithms are developed by alternating the evolutionary and PSO processes in a different order. Results showed that the proposed hybrid PSO algorithms have a comparable performance to SPEA2. Furthermore, Tang and Wang [14] proposed a novel hybrid multi-objective evolutionary algorithm (HMOEA) for real-valued multi-objective problems via incorporating the concepts of personal best and global best in PSO and evolutionary operators (i.e., multiple crossovers); it was for improving the robustness of evolutionary algorithms to solve different kinds of optimization problems.

Recently, Ibrahim et al. proposed hybridization of population-based metaheuristic algorithms called fusion of non-dominated fronts using reference points (FNFR) [15] to extract well-distributed solutions from a large set of non-dominated solutions collected during several runs of multiple algorithms.. Inspired from the FNFR framework and in the effort of developing a powerful general-purpose hybridization framework, we propose a novel hybridization of population-based multi- or many-objective optimization algorithms called hybrid many-Objective algorithm using Fusion of solutions from multiple algorithms (Fusion) to gain the combined benefits of several multi-objective optimization algorithms (MOOAs) and reduce the challenge of choosing one optimization algorithm to solve complex problems. The main difference between the proposed framework and the FNFR is that, in FNFR, the fusion process occurs after the optimization process, whereas in the proposed framework, the hybridization and fusion of solutions is done during the optimization process. Unlike other hybridization methods discussed above [3 - 15], the main features of the proposed framework are as follows:

1) The Fusion framework allows users to select and include multiple optimization algorithms in the search process with highest flexibility
2) The parallel execution of multiple algorithm using the same population to determine best performing algorithms at every stage of the search process.
3) Since reference-point-based selection mechanism is utilized, Fusion maintains the diversity of solutions.
4) In the serial execution stage of the Fusion framework, best performing algorithms are given the chance to run independently and continue generating improved candidate solutions.
5) In the Fusion framework, several algorithms can be used without the need of extra parameter tuning so several optimization algorithms can be hybridized with minimal effort.

The rest of the paper is organized as follows. Section II outlines the Fusion framework in detail. Section III presents experimental studies conducted to verify the efficacy of the proposed framework on 3- to 10-objective benchmark test problems. Concluding remarks are provided in Section IV.

II. Proposed Fusion Framework

In this section, we present a novel hybridization technique called hybridization of multi- and many-objective optimization algorithms framework called Hybrid Many-Objective Algorithm Using Fusion of Solutions from Several Many-Objective Algorithms (Fusion) to gain the combined benefits of several algorithms and reducing the challenge of choosing one optimization algorithm to solve complex problems. The Fusion framework contains four modules and Fig. 1 shows an overall flowchart of the framework.

A. Module 1: Initialization and Parameter Settings

In this module, multi-objective algorithms (Alg_1, Alg_2,…,Alg_n) with distinctive characteristics, which are suitable for solving the problem, are selected. For example, one can select an algorithm known for its diversity-preserving mechanism, and another known for its convergence ability, and another which is capable of maintaining good spread. Next, all parameters' settings required by each algorithm are set. Finally, the initial N random individuals for population (P_0) are created and objective functions, constraint functions and overall constraint violation for each solution are evaluated.

B. Module 2: Parallel Execution of all Algorithms

In this module, each algorithm involved in the Fusion framework is provided with the current population (P_0) and in turn generates a new population $\left(P_{Alg_i}\right)$, where $i = 1 \dots n$, according to $Alg_i's$ procedures. Thereafter, each algorithm combines P_0 and P_{Alg_i} and selects the best N candidate solutions according to their selection mechanism. If any

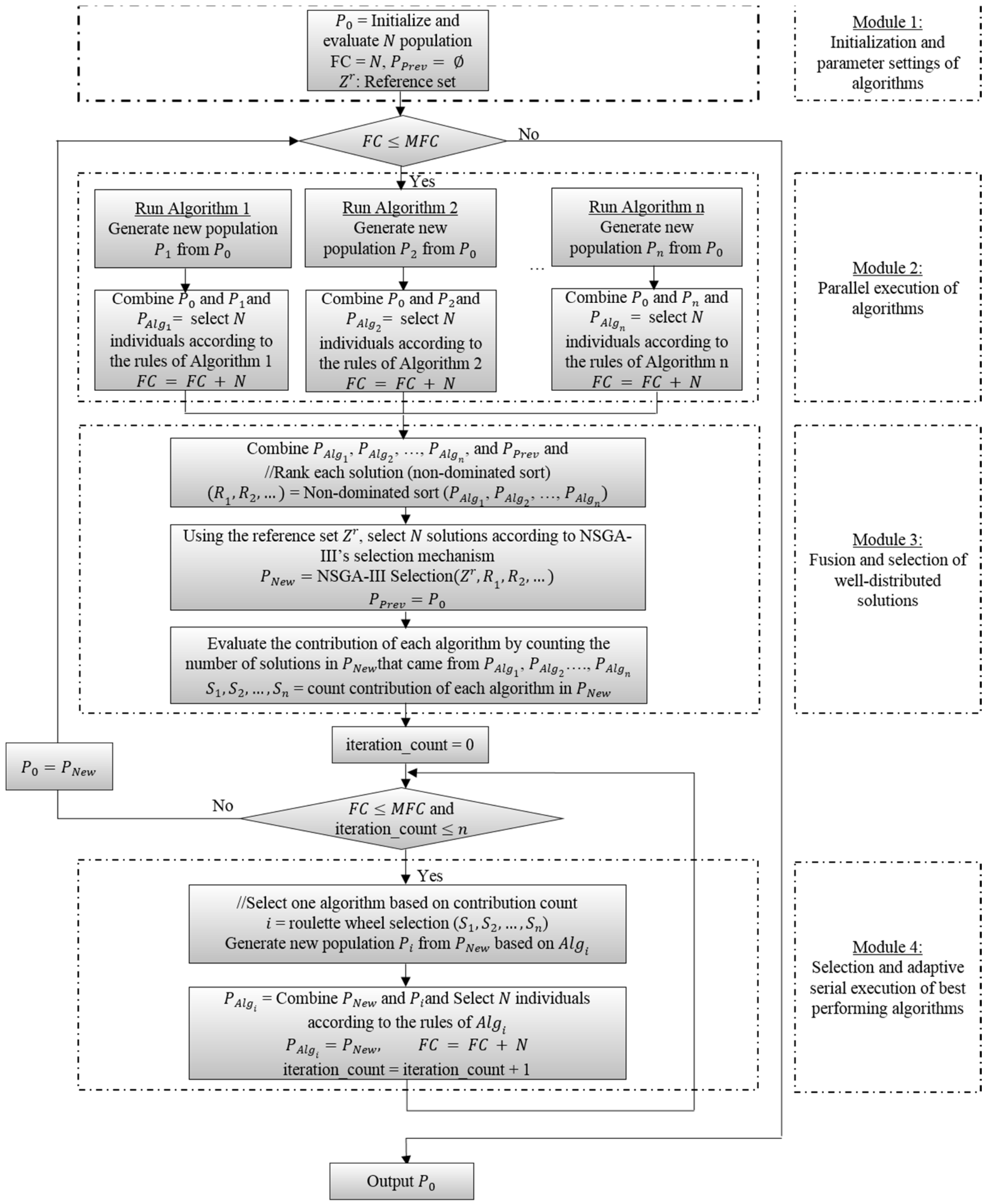

Fig. 1. Flowchart illustrating hybridization of MOEAs using the fusion of solutions of several MOEAs, where FC is the number of function calls and MFC is the maximum number of function calls .

algorithm involved in this framework employs an external archive, then the archive is consolidated based on the new candidate solution generated by the algorithm. Moreover, if the total number of function calls exceed the maximum number of function calls during parallel execution of algorithms, the process terminates and P_0 is reported. It is worth mentioning here that since all algorithms are supplied with the same population to generate and select new population based on their mechanism, promising algorithms can adaptively be selected and used to generate improved candidate solutions in subsequent stages of the search process. In Module 3, we describe the steps used in identifying the best performing algorithms in each stage.

C. Module 3: Fusion and Selection of Best Solutions

In this module, first solutions obtained by $Alg_1$to Alg_n are combined ($P_{Alg_i} \cup P_{Alg_i} \ldots P_{Alg_n}$) and sorted according to different non-domination levels (R_1, R_2 ,..). Second, all the solutions from each domination level are accepted one-by-one to construct elite population (P_{New}) until the size is equal to N. If the number of candidate solutions in the last domination level to be entered is greater than N, then the remaining solutions are selected based on NSGA-III's [16, 17]; which is a reference-point-based non-dominated sorting selection mechanism. Once the new population P_{New} reaches N, then, the contribution of each algorithm is evaluated by counting the number of solutions in P_{New} which came from each algorithm. By this way, the most suitable algorithm at the current search stage is identified. As a result: 1) well-distributed set of solutions at the current stage of the search process are selected, and 2) the contribution of each algorithm at the current stage are determined.

D. Module 4: Adaptive Serial Execution

After determining the contribution of each algorithm, we need to select best performing algorithm to run for the next n iterations, where n is the number of algorithms involved in the Fusion framework. This way a higher chance is given to those algorithms performing well in the current stage to continue the search process independently in hopes that they generate promising candidate solutions during subsequent generations. In this module, we utilize a roulette wheel-based selection mechanism according to each algorithm's contribution count. Then, the selected algorithm is given the chance to generate current population P_{Alg_i} from the population P_{New}. This current population is in turn combined with P_{New} ; N candidate solutions are then selected based on the algorithm's procedures and rules. This process is repeated n times so that highly performing algorithms have greater chance to be selected and to generate improved candidate solutions. Once this step is done, modules 2 to 4 are repeated until the termination criteria is met.

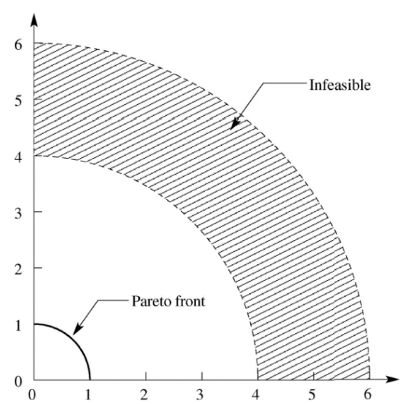

Fig. 2. Two-objective version of the C1-DTLZ3 problem [17].

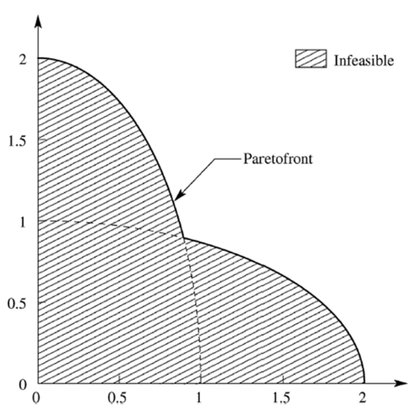

Fig. 3. Two-objective version of the C3-DTLZ4 problem [17].

III. Experimental Setup and Results

In this section, we present the algorithms used in the fusion hybridization framework, parameter settings, and simulation results on 3- to 10- objective benchmark test problems.

A. Utilized Algorithms

In order to assess the search capability of the proposed Fusion framework, we have utilized three MOOAs that have considerable differences in their fitness assignment and diversity mechanism to gain the combined benefits of these algorithms during the search process. These MOOAs are: the Generalized Differential Evolution Generation 3 (GDE3) [18], Speed-constrained Multi-objective Particle Swarm Optimization (SMPSO) [19], and the Strength Pareto Evolutionary Algorithm 2 (SPEA2) [20]. GDE3 uses a growing population and non-dominated sorting with pruning of non-dominated solutions to decrease the population size at the end of each generation. This mechanism improves the diversity of obtained solutions. On the other hand, since SMPSO mimics the social behavior of birds flocking to find food, particles move in the search space in a cooperative manner where movements are performed by the velocity operator. The velocity operator is guided by a local and a social behaviour of swarm. SPEA2 maintains an external archive according to each individual's strength by counting the number of individuals that dominate it as well as the number of individuals by which it is dominated. Moreover, SPEA2 uses a nearest neighbor density estimation method to guide the search process efficiently and it preserves boundary solutions. All these three algorithms are well-known and commonly used ones.

Since our method uses structured reference points similar to the NSGA-III algorithm to maintain the distribution of solutions during the search process, we compare our proposed method with the NSGA-III algorithm.

B. Test Problems

In order to test the quality of the proposed algorithm, we have used five unconstrained and four constrained many-objective benchmark test problems. The first sets of these test problems are the DTLZ (DTLZ1 – DTLZ4, Convex DTLZ2) family of test problems. The number of variables for these test problems are $(M + k - 1)$, where M is the number of objectives and $k = 5$ for DTLZ1, while $k = 10$ for DTLZ2, DTLZ3, DTLZ4, and Convex DTLZ2. The corresponding Pareto-optimal fronts lie in $f_i \in [0, 0.5]$ for the DTLZ1 problem and in $f_i \in [0, 1]$ for other DTLZ problems. The summary of the DTLZ problem characteristics is shown in Table I.

The second set of test problems utilized in this study are the four constrained versions of the DTLZ family test problems: Type-1 and Type-3. The Type-1 (C1-DTLZ1 and C2-DTLZ3) constrained problems contain the original DTLZ1 and DTLZ3 test problems. However, two constraints are added to create a barrier in approaching the original Pareto-optimal front. The C1-DTLZ1 test problem has a narrow feasible region surrounded by the infeasible region and this introduces a minor difficulty for optimizers to converge to the true Pareto-optimal front. The C1-DTLZ3 introduces a band of infeasible region adjacent to the Pareto-optimal front. Therefore, this creates the highest level of difficulty for optimizers to converge to the true Pareto-optimal front as they need to penetrate the band of the infeasible region when they travel from feasible to infeasible then to feasible region. The summary of the Type-1 problem characteristics is shown in Table I. Fig. 2 illustrates the feasible and the Pareto-optimal front of two-objective C1-DTLZ3 problems.

TABLE I. BENCHMARK TEST PROBLEMS

Problem	Characteristics
DTLZ1	Unconstrained, linear
DTLZ2	Unconstrained, concave
Convex DTLZ2	Unconstrained, convex
DTLZ3	Unconstrained, concave, multimodal
DTLZ4	Unconstrained, concave, biased
C1-DTLZ1	Constrained, linear, has barrier created by constraints
C1-DTLZ3	Constrained, concave, multimodal, has large barrier created by constraints
C3-DTLZ1	Constrained, linear, the Pareto-optimal front moved to the added constraint surface
C3-DTLZ4	Constrained, concave, biased, the Pareto-optimal front moved to the added constraint surface

TABLE II. GDE3, SMPSO, SPEA2, AND NSGA-III PARAMETER SETTINGS. n IS THE NUMBER OF VARIABLES AND $|P|$ IS THE POPULATION SIZE

GDE3	
Mutation probability (F)	0.1
Crossover probability (CR)	0.5
SMPSO	
Archive size	$\lvert P \rvert$
Polynomial mutation (p_m)	$1/n$
Mutation Distribution Index (η_m)	20
SPEA2	
Archive size	$\lvert P \rvert$
SBX probability (p_c)	0.9
Polynomial mutation (p_m)	$1/n$
Crossover Distribution Index (η_c)	20
Mutation Distribution Index (η_m)	20
NSGA-III	
SBX probability (p_c)	1.0
Polynomial mutation (p_m)	$1/n$
Crossover Distribution Index (η_c)	30
Mutation Distribution Index (η_m)	20

TABLE III. NUMBER OF REFERENCE POINTS AND POPULATION SIZES USED IN NSGA-III AND ELITENSGA-III ALGORITHMS.

Number of Objectives (M)	Divisions		Reference Points(H)	Population Size (N)
	Outer	*Inner*		
3	12	0	91	92
5	6	0	210	212
8	3	2	156	156
10	3	2	275	276

The Type-3 (C3-DTLZ1 and C3-DTLZ4) constrained problems contain the original DTZL1 and DTLZ4 test problems. However, M constraints are added to original problems so that the original Pareto-optimal front is no longer optimal. Instead, the new Pareto-optimal front is created by portions of constraint surfaces. These problems are deigned to assess the optimizers' ability to stay on the newly created Pareto-optimal surface. The summary of the Type-3 problem characteristics is shown in Table I. Fig. 3 illustrates the feasible and the Pareto-optimal front of two-objective C3-DTLZ4 problems.

C. *Parameter and Experimental Settings*

The GDE3 algorithm has two control parameters: mutation amplification factor (F) and crossover rate (CR). The SMPSO algorithm has three parameters: archive size, polynomial mutation (p_m), and mutation distribution index (η_m). The NSGA-III algorithm has four control parameters: SBX probability, polynomial mutation, crossover distribution index, and mutation distribution index. In addition to the NSGA-III control parameters, SPEA2 has archive size parameter. In order to maintain a consistent and fair comparison, the parameter settings for all algorithms including the Fusion framework are kept the same as the original studies of each algorithm. Table II presents parameter settings used by GDE3, SMPSO, SPEA2, and NSGA-III algorithms. Furthermore, since the Fusion framework as well as the NSGA-III algorithm require predetermined reference points to maintain the diversity of solutions, we have used the same setting reported in the original NSGA-III studies [16, 17]. Table III shows the number of reference points (H), the population size (N), and the number of inner and outer divisions used for different dimensions of test problems.

To evaluate the performance of the proposed algorithm, we have used the inverse generational distance (IGD) metric, which is capable of measuring the convergence and the diversity of the obtained Pareto-optimal solutions at the same time. The IGD measure has been predominantly used to evaluate the performance of evolutionary many-objective problems [16, 17, 19]. The IGD metric measures the distances between each solution composing the Pareto-optimal front and the obtained solution. The IGD metric is defined as follow:

$$IGD = \frac{\sqrt{\sum_{i=1}^{n} d_i^2}}{n},$$

where n refers to the number of solutions in the Pareto-optimal front, and d_i refer to the Euclidean distance (measured in objective space) between each point of the Pareto-optimal front (reference Pareto front) and the nearest member of obtained solution. In this study, the reference Pareto front is constructed by joining all results of all the executions and then selecting the non-dominated solutions. Furthermore, all algorithms were executed 20 times independently and the best, the worst, the median, and the average results of each algorithm is recorded. Additionally, the Wilcoxon's signed rank statistical test is conducted at a 5% significance level in order to evaluate the statistical significance of obtained results.

D. *Experimental Results and Discussion*

To evaluate the performance of the proposed algorithm, we have conducted two sets of experiments on constrained and unconstrained DTLZ family test problems ccontaining three- to ten objectives. The first experiment investigates the performance of the proposed Fusion framework on unconstrained test problems with varying Pareto-optimal front shapes. The second experiment investigates how the proposed framework copes with constrained problems containing barriers in approaching the Pareto-optimal front and its ability tackle problems with their Pareto-optimal surface is created by portions of added constrained surface.

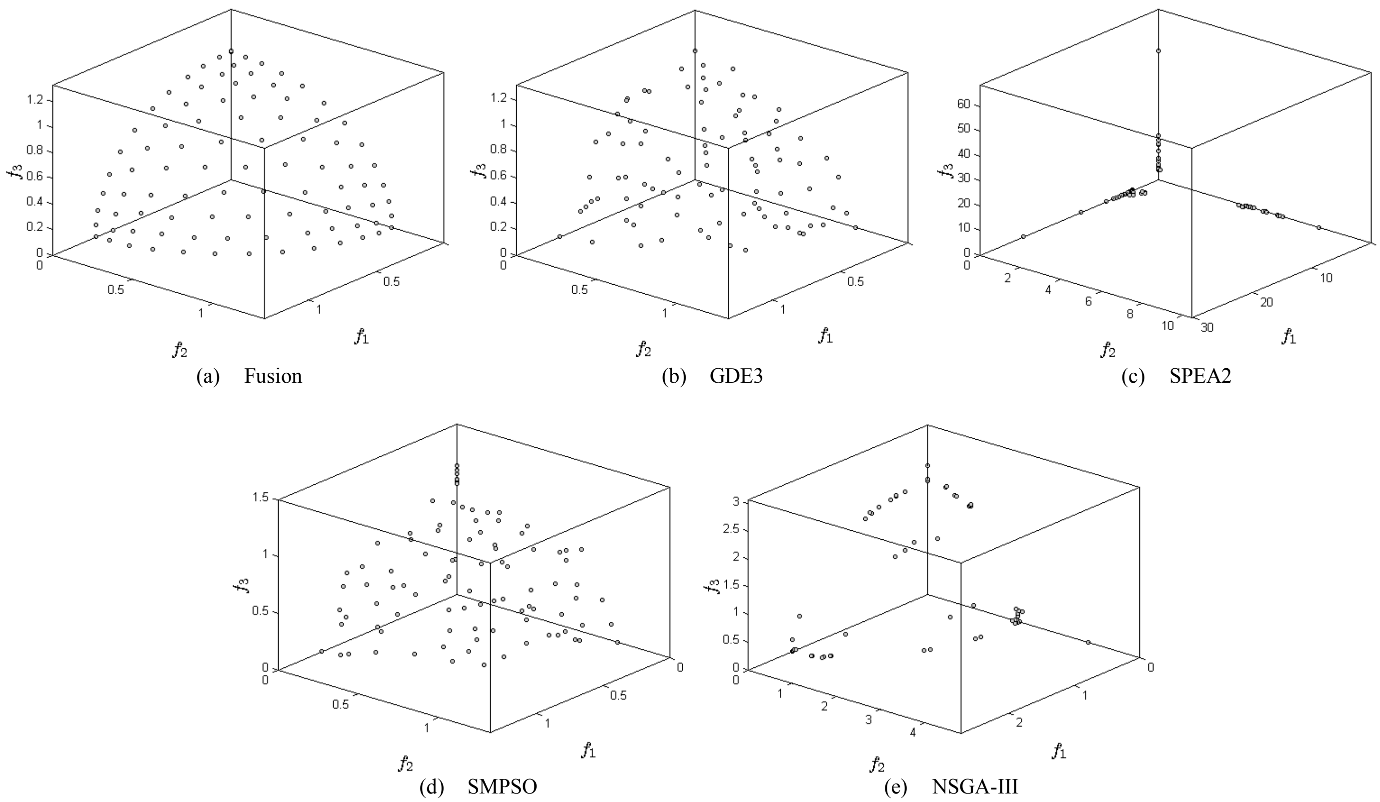

Fig. 4. The trade-off plots of obtained solutions by Fusion, GDE3, SPEA2, SMPSO and NSGA-III algorithms for three-objective DTLZ3 test problem.

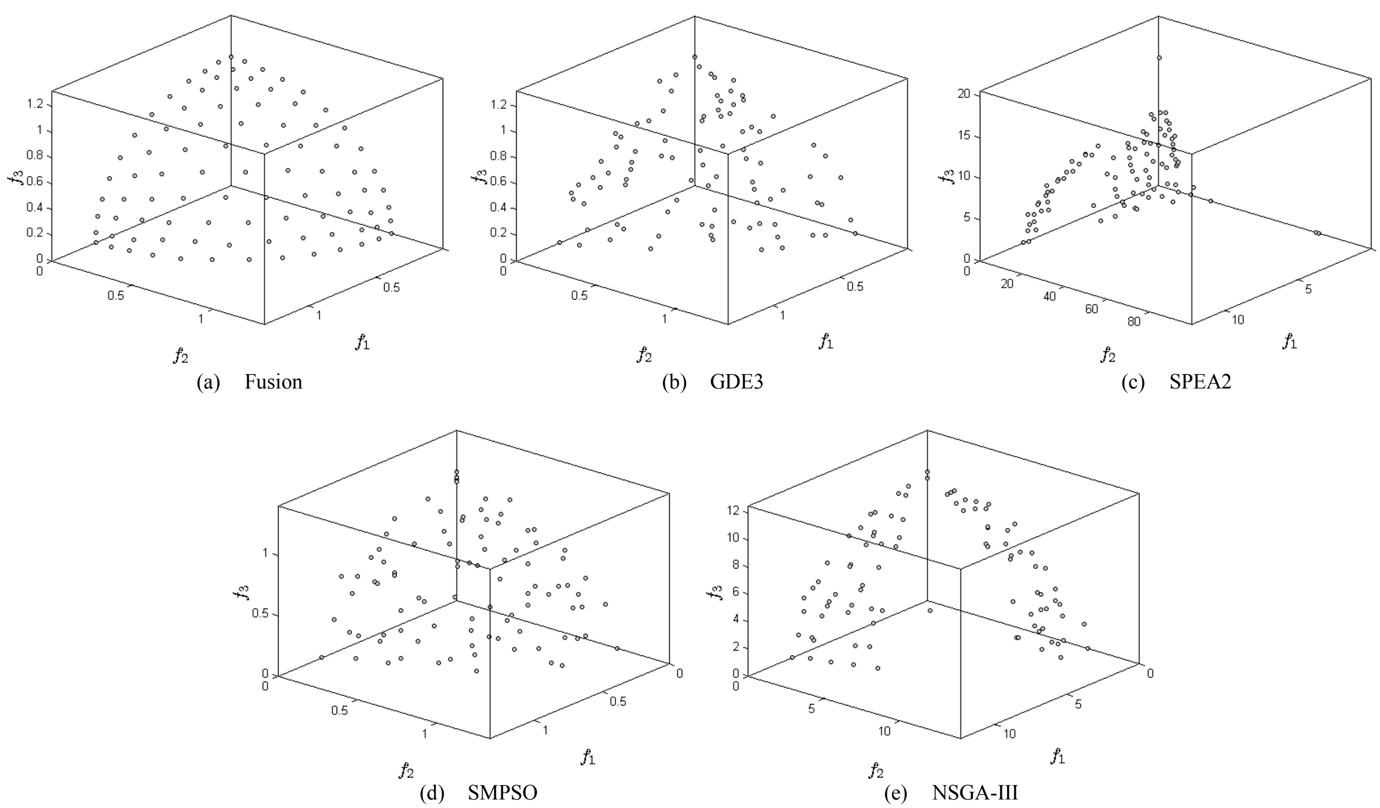

Fig. 5. The trade-off plots of obtained solutions by Fusion, GDE3, SPEA2, SMPSO and NSGA-III algorithms for three-objective C1-DTLZ3 test problem.

TABLE IV. BEST, MEDIAN, WORST, AND AVERAGE IGD VALUES FOR FUSION, GDE3, SPEA2, SMPSO AND NSGA-III ON M-OBJECTIVE DTLZ1, DTLZ3, AND DTLZ4 PROBLEMS. BEST PERFORMED ALGORITHM IS SHOWN IN DARK GRAY AND SECOND BEST IS SHOWN LIGHT GRAY. * INDICATES A SIGNIFICANCE LEVEL OF 0.05 BETWEEN THE TOP TWO ALGORITHMS.

Problem	M	MFC	Fusion	GDE3	SPEA2	SMPSO	NSGA-III
DTLZ1	3	250 x 92	4.99E-04	8.01E-04	7.14E-04	9.08E-04	5.14E-04
			5.02E-04	8.31E-04	8.80E-04	9.89E-04	6.82E-04
			5.13E-04	8.64E-03	2.18E-03	1.10E-03	8.64E-03
			5.03E-04*	1.62E-03	1.02E-03	9.91E-04	1.17E-03
	5	450 x 212	6.36E-04	1.14E-03	1.67E-01	2.08E-03	6.39E-04
			6.42E-04	1.16E-03	4.23E-01	2.26E-03	6.89E-04
			6.55E-04	1.22E-03	5.84E-01	2.70E-03	1.25E-03
			6.43E-04*	1.17E-03	4.00E-01	2.32E-03	7.71E-04
	8	700 x 156	1.32E-03	5.72E-03	1.76E+00	5.56E-03	1.23E-03
			1.36E-03	6.72E-03	4.06E+00	8.55E-03	1.56E-03
			1.52E-03	9.49E-03	5.99E+00	2.04E-01	2.46E-03
			1.37E-03*	6.81E-03	4.07E+00	2.24E-02	1.69E-03
	10	1000 x 276	9.71E-04	5.24E-03	2.20E+00	4.66E-03	8.21E-04
			1.05E-03	5.90E-03	3.53E+00	9.75E-03	9.68E-04
			1.23E-03	7.57E-03	4.36E+00	1.70E-01	1.81E-03
			1.06E-03	6.00E-03	3.40E+00	2.81E-02	1.11E-03
DTLZ3	3	250 x 92	3.47E-04	1.71E-03	5.03E-02	1.03E-03	7.95E-03
			5.18E-04	1.77E-03	1.91E-01	1.45E-03	5.09E-02
			6.83E-04	3.47E-02	3.06E-01	7.75E-03	1.55E-01
			4.95E-04*	7.05E-03	1.19E-01	3.13E-03	5.14E-02
	5	450 x 212	8.25E-04	9.13E-04	1.11E-02	2.36E-03	8.39E-04
			8.34E-04	9.72E-04	3.07E-02	3.29E-03	1.08E-03
			8.49E-04	1.02E-03	5.42E-02	4.18E-03	1.61E-03
			8.34E-04*	9.69E-04	3.06E-02	3.31E-03	1.10E-03
	8	700 x 156	1.19E-03	4.94E-03	2.47E-02	7.26E-03	1.81E-03
			1.54E-03	7.01E-03	4.02E-02	9.28E-03	1.88E-03
			1.80E-03	8.81E-03	2.86E-01	9.38E-03	4.04E-03
			1.55E-03*	7.11E-03	7.18E-02	9.09E-03	2.10E-03
	10	1000 x 276	1.10E-03	4.39E-03	1.54E-02	3.87E-03	1.68E-03
			1.63E-03	5.40E-03	2.69E-02	5.65E-03	1.71E-03
			2.14E-03	5.53E-03	2.20E-01	5.69E-03	1.74E-03
			1.61E-03	5.23E-03	4.10E-02	5.46E-03	1.71E-03
DTLZ4	3	400 x 92	5.58E-04	8.90E-04	7.32E-04	9.91E-04	5.57E-04
			5.60E-04	9.80E-04	7.59E-04	1.04E-03	5.61E-04
			5.63E-04	8.42E-03	8.42E-03	1.54E-03	1.27E-02
			5.60E-04*	1.35E-03	3.36E-03	1.08E-03	1.17E-03
	5	700 x 212	9.15E-04	1.42E-03	1.79E-03	1.77E-03	9.12E-04
			9.19E-04	1.46E-03	2.00E-03	2.29E-03	9.15E-04
			1.10E-03	1.50E-03	3.43E-03	2.67E-03	9.23E-04
			9.28E-04	1.46E-03	2.08E-03	2.30E-03	9.16E-04
	8	1100 x 156	1.89E-03	2.65E-03	8.97E-03	3.39E-03	1.86E-03
			1.91E-03	2.80E-03	9.19E-03	3.67E-03	1.89E-03
			1.94E-03	2.95E-03	9.38E-03	4.11E-03	2.02E-03
			1.91E-03	2.81E-03	9.18E-03	3.69E-03	1.90E-03
	10	1500 x 276	1.72E-03	2.09E-03	6.24E-03	2.30E-03	1.82E-03
			1.83E-03	2.23E-03	6.33E-03	2.54E-03	1.83E-03
			1.85E-03	2.36E-03	6.42E-03	2.85E-03	1.89E-03
			1.81E-03*	2.24E-03	6.33E-03	2.58E-03	1.84E-03
Number of statistically significant wins			8	0	0	0	0

1) Unconstrained Problems

The first experiment investigates the performance of Fusion on problems with linear or concave Pareto- optimal fronts for three- to ten- objectives DTLZ1, DTLZ3 and DTLZ4 problems. Fig. 4 shows the obtained Pareto fronts by Fusion, GDE3, SPEA2, SMPSO, and NSGA-III for the three-objective DTLZ3 test problem after 250 generations (250 * 92 function calls). It is evident from these diagrams that Fusion is able to find well-distributed solutions on the Pareto-optimal front. Table IV provides the best, median, worst, and average IGD values of all algorithms for the above-mentioned test problems. From this, we can see that the performance of Fusion is significantly better than not only the algorithms involved in the hybridization process but also NSGA-III for almost all experiments conducted in this section.

.

TABLE V. BEST, MEDIAN, WORST, AND AVERAGE IGD VALUES FOR FUSION, GDE3, SPEA2, SMPSO AND NSGA-III ON M-OBJECTIVE DTLZ2 AND CONVEX DTLZ2 PROBLEMS. BEST PERFORMED ALGORITHM IS SHOWN IN DARK GRAY AND SECOND BEST IS SHOWN LIGHT GRAY. * INDICATES A SIGNIFICANCE LEVEL OF 0.05 BETWEEN THE TOP TWO ALGORITHMS.

Problem	M	MFC	Fusion	GDE3	SPEA2	SMPSO	NSGA-III
DTLZ2	3	250 x 92	6.31E-04	9.23E-04	7.07E-04	1.07E-03	6.28E-04
			6.36E-04	9.85E-04	7.98E-04	1.13E-03	6.34E-04
			6.43E-04	1.05E-03	8.27E-04	1.24E-03	6.56E-04
			6.36E-04	9.87E-04	8.00E-04	1.13E-03	6.36E-04
	5	450 x 212	9.90E-04	1.41E-03	1.64E-03	2.43E-03	9.83E-04
			9.93E-04	1.44E-03	1.78E-03	2.71E-03	9.86E-04
			1.00E-03	1.51E-03	1.92E-03	2.91E-03	1.02E-03
			9.94E-04	1.45E-03	1.79E-03	2.68E-03	9.89E-04*
	8	700 x 156	1.76E-03	3.92E-03	1.09E-02	5.30E-03	1.80E-03
			1.91E-03	4.23E-03	1.11E-02	5.81E-03	1.85E-03
			1.98E-03	4.86E-03	1.14E-02	6.57E-03	2.01E-03
			1.90E-03	4.25E-03	1.11E-02	5.80E-03	1.86E-03
	10	1000 x 276	1.44E-03	4.49E-03	8.27E-03	4.92E-03	1.49E-03
			1.55E-03	4.71E-03	8.40E-03	5.69E-03	1.52E-03
			1.70E-03	4.88E-03	8.64E-03	6.22E-03	1.70E-03
			1.55E-03	4.69E-03	8.41E-03	5.63E-03	1.55E-03
Convex DTLZ2	3	300 x 92	4.77E-04	6.18E-04	4.81E-04	7.93E-04	4.94E-04
			4.98E-04	6.78E-04	5.00E-04	8.46E-04	5.55E-04
			5.11E-04	7.65E-04	5.34E-04	9.64E-04	6.07E-04
			4.96E-04*	6.89E-04	5.05E-04	8.64E-04	5.55E-04
	5	500 x 212	4.11E-04	5.23E-04	8.00E-03	1.27E-03	4.49E-04
			4.33E-04	5.49E-04	1.88E-02	1.61E-03	5.13E-04
			4.96E-04	5.67E-04	2.34E-02	2.13E-03	7.45E-04
			4.36E-04*	5.50E-04	1.78E-02	1.59E-03	5.35E-04
	8	800 x 156	8.14E-04	1.10E-03	3.28E-03	1.54E-03	1.49E-03
			1.34E-03	1.16E-03	3.72E-03	2.19E-03	1.50E-03
			1.47E-03	1.28E-03	3.90E-03	3.02E-03	1.52E-03
			1.31E-03	1.16E-03*	3.69E-03	2.27E-03	1.50E-03
	10	1000 x 276	1.26E-03	5.68E-04	1.59E-03	7.75E-04	1.96E-03
			1.71E-03	6.41E-04	1.85E-03	1.18E-03	1.97E-03
			1.96E-03	7.83E-04	2.02E-03	1.55E-03	2.00E-03
			1.69E-03	6.53E-04*	1.85E-03	1.16E-03	1.97E-03
Number of statistically significant wins			2	2	0	0	1

The second experiment investigates the performance of Fusion on DTLZ2 and Convex DTLZ2 for three- to ten-objectives problems. From Table V we see that even though Fusion's IGD values are not statistically significant than NSGA-III, they are significantly better than algorithms involved in the hybridization process for almost all instances of the test problems. From the above two experiments, we see that none of the algorithms involved in Fusion experiment are not able to find well-distributed and well-converged solutions consistently. However, since Fusion uses predefined structured reference points to guide and preserve the diversity of obtained solutions and adaptively select best performing algorithms in every stage of the search process, it consistently able to find well-distributed solutions that may not be possible using one optimization algorithm.

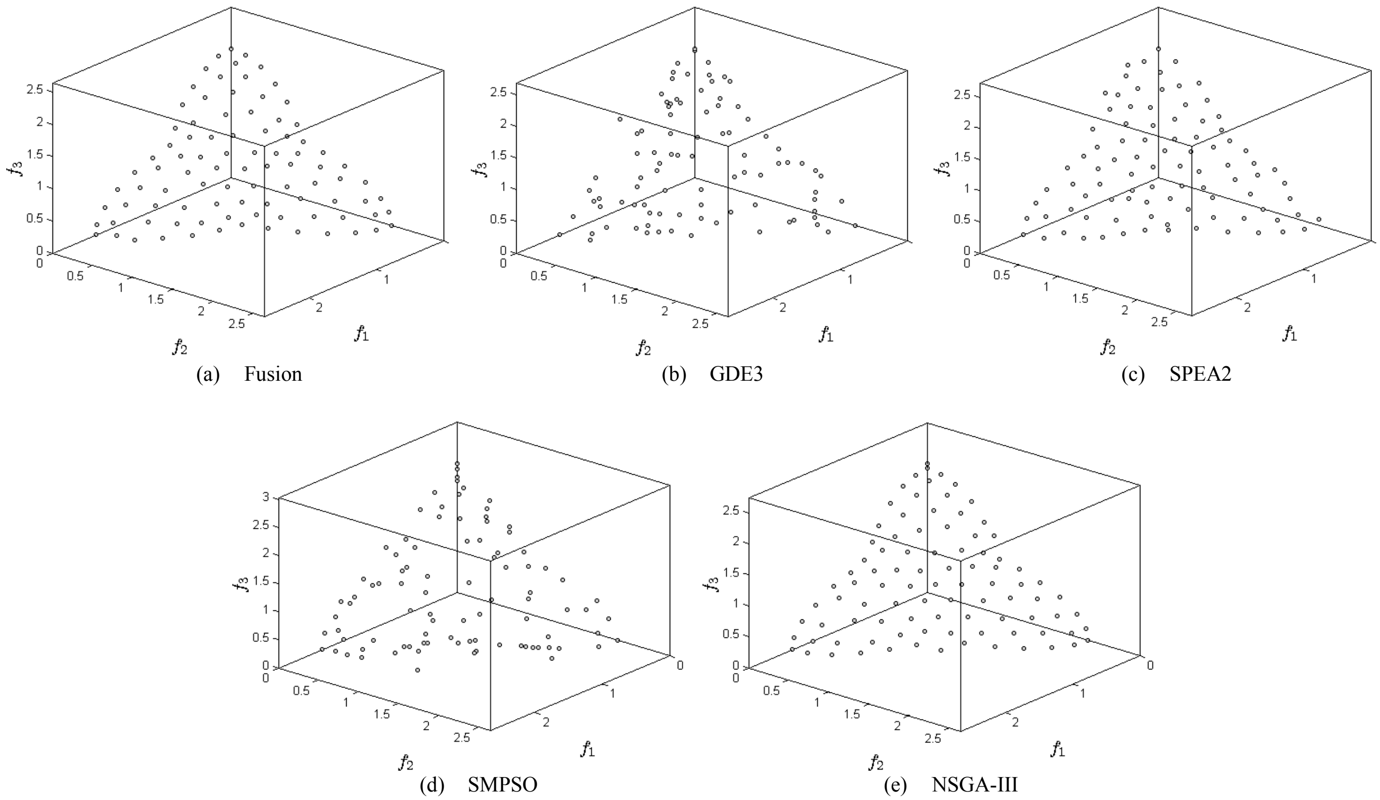

Fig. 6. The trade-off plots of obtained solutions by Fusion, GDE3, SPEA2, SMPSO and NSGA-III algorithms for three-objective C3-DTLZ4 test problem.

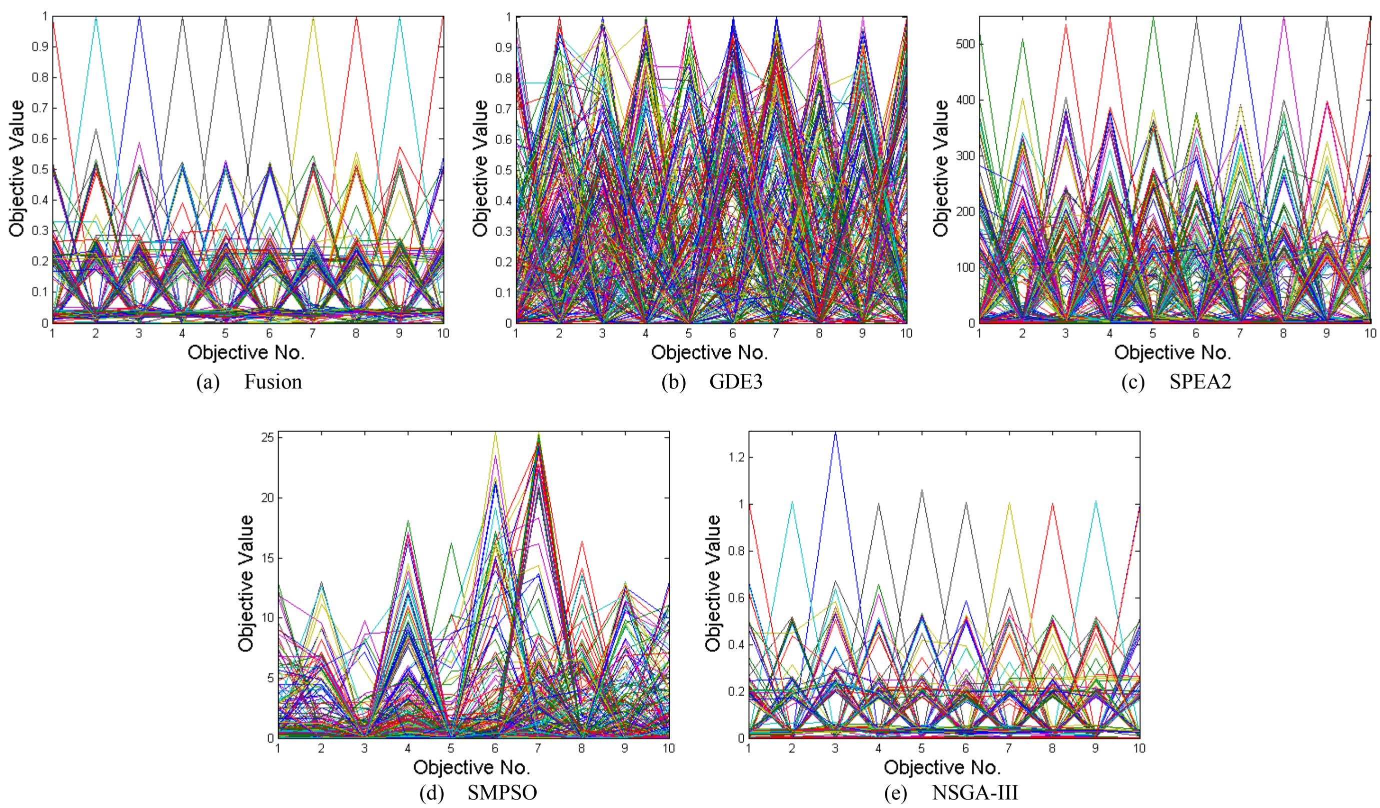

Fig. 7. Value path comparison of obtained solutions by Fusion, GDE3, SPEA2, SMPSO and NSGA-III algorithms for ten-objective C3-DTLZ1 test problem.

TABLE VI. BEST, MEDIAN, WORST, AND AVERAGE IGD VALUES FOR FUSION, GDE3, SPEA2, SMPSO AND NSGA-III ON M-OBJECTIVE CONVEX C1-DTLZ1, C1-DTLZ3, C3-DTLZ1, AND C3-DTLZ4 PROBLEMS. BEST PERFORMED ALGORITHM IS SHOWN IN DARK GRAY AND SECOND BEST IS SHOWN LIGHT GRAY. * INDICATES A SIGNIFICANCE LEVEL OF 0.05 BETWEEN THE TOP TWO ALGORITHMS.

Problem	M	NFC	Fusion	GDE3	SPEA2	SMPSO	NSGA-III
C1-DTLZ1	3	250 x 92	4.15E-04	6.22E-04	4.93E-04	6.80E-04	4.16E-04
			4.17E-04	6.71E-04	5.80E-04	7.52E-04	6.30E-04
			4.24E-04	1.07E-02	8.48E-04	2.53E-03	1.03E-02
			4.17E-04*	4.47E-03	6.07E-04	8.54E-04	2.29E-03
	5	450 x 212	5.23E-04	8.19E-04	6.27E-04	9.98E-04	5.10E-04
			5.26E-04	8.52E-04	6.56E-04	1.10E-03	5.17E-04
			5.29E-04	6.85E-03	7.58E-04	1.34E-03	5.24E-04
			5.26E-04	2.35E-03	6.68E-04	1.11E-03	5.17E-04*
	8	700 x 156	1.09E-03	1.74E-03	1.39E-03	2.17E-03	1.06E-03
			1.10E-03	1.78E-03	1.44E-03	2.41E-03	1.07E-03
			1.10E-03	1.70E-02	1.53E-03	2.59E-03	1.09E-03
			1.10E-03	4.15E-03	1.44E-03	2.40E-03	1.07E-03*
	10	1000 x 276	8.20E-04	1.36E-03	1.08E-03	1.64E-03	8.11E-04
			8.23E-04	1.37E-03	1.11E-03	1.84E-03	8.15E-04
			8.27E-04	1.27E-02	1.15E-03	2.10E-03	8.20E-04
			8.23E-04	2.44E-03	1.11E-03	1.85E-03	8.15E-04*
C1-DTLZ3	3	300 x 92	7.56E-04	1.43E-03	1.48E-01	1.46E-03	1.48E-01
			7.62E-04	4.84E-02	1.50E-01	1.76E-03	1.48E-01
			7.67E-04	1.48E-01	2.01E-01	9.25E-03	1.55E-01
			7.62E-04*	6.97E-02	1.53E-01	3.43E-03	1.49E-01
	5	500 x 212	9.06E-04	1.85E-03	4.37E-01	3.85E-03	5.13E-02
			9.16E-04	5.17E-02	8.94E-01	5.12E-03	5.18E-02
			9.28E-04	5.19E-02	1.56E+00	1.56E-01	5.35E-02
			9.16E-04*	3.49E-02	8.78E-01	1.30E-02	5.19E-02
	8	800 x 156	1.58E-03	6.58E-03	1.53E-02	9.29E-03	4.62E-03
			2.32E-03	8.94E-03	6.91E-02	1.06E-02	6.26E-03
			2.90E-03	1.03E-02	8.37E-01	1.08E-02	7.14E-03
			2.23E-03*	8.78E-03	1.27E-01	1.05E-02	6.17E-03
	10	1000 x 276	2.89E-03	4.42E-03	2.66E-02	4.66E-03	6.48E-03
			4.10E-03	5.49E-03	3.68E-02	5.20E-03	7.06E-03
			5.85E-03	5.89E-03	1.96E-01	5.39E-03	7.18E-03
			4.13E-03*	5.39E-03	5.69E-02	5.16E-03	6.97E-03
C3-DTLZ1	3	400 x 92	5.84E-03	3.20E-03	2.65E-01	5.38E-03	6.84E-03
			7.23E-03	3.61E-03	7.99E-01	1.17E-02	8.22E-03
			8.03E-03	5.33E-03	1.31E+00	1.35E-01	9.26E-03
			6.98E-03	3.70E-03*	8.01E-01	2.19E-02	8.26E-03
	5	700 x 212	8.92E-04	1.94E-03	2.23E-01	2.61E-03	1.00E-03
			9.14E-04	2.29E-03	3.52E-01	3.21E-03	1.25E-03
			1.13E-03	2.44E-03	4.97E-01	4.35E-03	1.54E-03
			9.23E-04*	2.25E-03	3.63E-01	3.25E-03	1.25E-03
	8	1100 x 156	1.49E-03	5.89E-03	1.32E+00	5.65E-03	1.76E-03
			1.64E-03	7.95E-03	2.33E+00	3.31E-02	2.15E-03
			1.83E-03	9.37E-03	3.09E+00	1.84E-01	3.82E-03
			1.64E-03*	7.79E-03	2.25E+00	5.30E-02	2.24E-03
	10	1500 x 276	1.21E-03	4.28E-03	1.17E+00	4.47E-03	1.19E-03
			1.31E-03	6.49E-03	1.79E+00	6.23E-02	1.53E-03
			1.40E-03	7.86E-03	2.22E+00	1.39E-01	2.77E-03
			1.30E-03*	6.54E-03	1.78E+00	6.18E-02	1.66E-03
C3-DTLZ4	3	400 x 92	5.74E-04	9.28E-04	7.34E-04	1.05E-03	5.78E-04
			5.89E-04	9.57E-04	7.97E-04	1.17E-03	5.99E-04
			6.23E-04	1.03E-03	7.65E-03	1.99E-03	6.67E-04
			5.90E-04*	9.66E-04	2.15E-03	1.31E-03	6.09E-04
	5	700 x 212	7.71E-04	1.48E-03	1.74E-03	1.82E-03	7.68E-04
			7.80E-04	1.56E-03	1.87E-03	2.05E-03	7.87E-04
			8.03E-04	1.67E-03	2.08E-03	2.18E-03	8.41E-04
			7.82E-04	1.56E-03	1.87E-03	2.03E-03	7.90E-04
	8	1100 x 156	1.83E-03	3.19E-03	7.67E-03	3.42E-03	1.80E-03
			1.85E-03	3.34E-03	7.96E-03	3.58E-03	1.83E-03
			1.88E-03	3.56E-03	8.16E-03	3.96E-03	1.87E-03
			1.85E-03	3.34E-03	7.91E-03	3.59E-03	1.83E-03*
	10	1500 x 276	1.50E-03	2.76E-03	5.89E-03	2.69E-03	1.50E-03
			1.55E-03	2.96E-03	5.98E-03	2.87E-03	1.53E-03
			1.56E-03	3.08E-03	6.13E-03	3.12E-03	1.58E-03
			1.55E-03	2.95E-03	5.99E-03	2.87E-03	1.54E-03
Number of statistically significant wins			9	1	0	0	4

2) *Unconstrained Problems*

The first experiment investigates the performance of Fusion on problems with linear or concave Pareto-optimal fronts for three- to ten- objectives DTLZ1, DTLZ3 and DTLZ4 problems. Fig. 4 shows the obtained Pareto fronts by Fusion, GDE3, SPEA2, SMPSO, and NSGA-III for the three-objective DTLZ3 test problem after 250 generations (250 * 92 function calls). It is evident from these diagrams that Fusion is able to find well-distributed solutions on the Pareto-optimal front. Table IV provides the best, median, worst, and average IGD values of all algorithms for the above-mentioned test problems. From this we can see that the performance of Fusion is significantly better than not only the algorithms involved in the hybridization process but also NSGA-III for almost all experiments conducted in this section.

The second experiment investigates the performance of Fusion on DTLZ2 and Convex DTLZ2 for three- to ten-objective problems. From Table V, we see that even though Fusion's IGD values are not statistically significant than NSGA-III, they are significantly better than algorithms involved in the hybridization process for almost all instances of the test problems. From the above two experiments we see that none of the algorithms involved in Fusion experiment are able to consistently find well-distributed and well-converged solutions. However, since Fusion uses predefined structured reference points to guide and preserve the diversity of obtained solutions and adaptively select best performing algorithms in every stage of the search process, it is consistently able to find a well-distributed solution which may not be possible using one optimization algorithm.

3) *Constrained Problems*

The Type-1 constrained test problems challenge optimizers' ability to penetrate the barrier created by the constraints in order to reach the global Pareto-optimal front. Fig. 5 shows obtained solutions by the Fusion, GDE3, SPEA2, SMPSO, and NSGA-III three-objective C1-DTLZ3 test problem after 300 generations (300 * 92 function calls). From this diagram, we can see that only the Fusion method is able to obtain well-distributed and converged solutions on the Pareto-optimal surface. However, when we look at Fig. 5 (b), (c), and (d), none of the algorithms involved in the hybridization process individually are able to obtain well-distributed and converged solutions with the number of function calls. Furthermore, from Fig. 5 (c) and (e) we observe that the SPEA2 and NSGA-III algorithms are not only unable to penetrate the barrier created by the constraints of the test problem but also failed to find well-distributed on the newly created barrier surface. Table VI shows that the proposed framework significantly outperformed the three algorithms in the hybridization process while showing comparable results with NSGA-III on C1-DTLZ1 and significantly better solution problems on C1-DTLZ3 with three- to ten- objectives.

On the other hand, Type-3 constrained problems are designed to test the ability of an optimizer to stay on the Pareto-optimal front created by portions of constraint surface. Fig. 6 depicts obtained solutions by the Fusion, GDE3, SPEA2, SMPSO, and NSGA-III three-objective C3-DTLZ4 test problem after 400 generations (400 * 92 function calls). From this figure, we can see that Fusion, SPEA2 and NSGA-III are able to obtain comparable distribution of solutions on the Pareto-optimal front. However, from Fig. 6 (b) and (d) we see that GDE3 and SMPSO failed to find well-distributed solution over $f_i \in [0, 2]$. In Table VI we see that Fusion outperformed every algorithm in almost all instances of C3-DTLZ1 and C3-DTLZ4 problems in terms of IGD metric, followed by NSGA-III. Also, from Fig. 7 we see that Fusion is able to obtain well-distributed solutions for C3-DTLZ1 over $f_i \in [0, 1]$ for all ten objectives and trade-offs among them. However, Fig. 7 (c) and (d) illustrate that SPEA2 and SMPSO individually are not able to converge their solution on $f_i \in [0, 1]$.

IV. CONCLUSION

In this paper, we proposed a novel hybridization of multi- and many-objective optimization algorithms framework called fusion-based hybrid many-objective optimization algorithm; which utilizes several many-objective algorithms to gain the combined benefits of several algorithms and reduce the challenge of choosing one optimization algorithm to solve complex problems. In the Fusion framework, several algorithms can be used without the need of extra parameter tuning so that several optimization algorithms can be hybridized with minimal effort. Furthermore, since Fusion uses predefined structured reference points to guide and preserve the diversity of obtained solutions and adaptively select best performing algorithms in every stage of the search process, it can consistently find a well-distributed solution that may not be possible to find using only one optimization algorithm.

The efficacy of the proposed Fusion framework was investigated using three widely used optimization algorithms; GDE3, SMPSO, and SPEA2. Experimental results on five unconstrained and four constrained benchmark test problems with three to ten objectives showed that the Fusion framework significantly outperformed all algorithms involved in the hybridization process as well as the NSGA-III algorithm in terms of diversity and convergence of obtained solutions. Furthermore, the numerical results also show that the proposed Fusion framework is able to consistently show good performance. In the future, we would like to investigate the performance of Fusion in practical many-objective problems.

REFERENCES

[1] D. H. Wolpert and W. G. Macready, "No free lunch theorems for optimization," IEEE transactions on evolutionary computation, vol. 1, pp. 67-82, 1997.

[2] E. Nwankwor, A. K. Nagar, and D. Reid, "Hybrid differential evolution and particle swarm optimization for optimal well placement," Computational Geosciences, vol. 17, pp. 249-268, 2013.

[3] K. C. Tan, Y. Chew, and L. H. Lee, "A hybrid multi-objective evolutionary algorithm for solving truck and trailer vehicle routing problems," European Journal of Operational Research, vol. 172, pp. 855-885, 2006.

[4] W. Xia and Z. Wu, "An effective hybrid optimization approach for multi-objective flexible job-shop scheduling problems," Computers & Industrial Engineering, vol. 48, pp. 409-425, 2005.

[5] R. Tavakkoli-Moghaddam, A. Rahimi-Vahed, and A. H. Mirzaei, "A hybrid multi-objective immune algorithm for a flow shop scheduling problem with bi-objectives: weighted mean completion time and weighted mean tardiness," Information Sciences, vol. 177, pp. 5072-5090, 2007.

[6] S. Karthikeyan, P. Asokan, S. Nickolas, and T. Page, "A hybrid discrete firefly algorithm for solving multi-objective flexible job shop scheduling problems," International Journal of Bio-Inspired Computation, vol. 7, pp. 386-401, 2015.

[7] Y. Wang, Z. Cai, Y. Zhou, and Z. Fan, "Constrained optimization based on hybrid evolutionary algorithm and adaptive constraint-handling technique," Structural and Multidisciplinary Optimization, vol. 37, pp. 395-413, 2009.

[8] A.-C. Zăvoianu, E. Lughofer, G. Bramerdorfer, W. Amrhein, and E. P. Klement, "DECMO2: a robust hybrid and adaptive multi-objective evolutionary algorithm," Soft Computing, vol. 19, pp. 3551-3569, 2015.

[9] S.-Z. Zhao and P. N. Suganthan, "Multi-objective evolutionary algorithm with ensemble of external archives," International Journal of Innovative Computing, Information and Control, vol. 6, pp. 1713-1726, 2010.

[10] B. Y. Qu and P. N. Suganthan, "Constrained multi-objective optimization algorithm with an ensemble of constraint handling methods," Engineering Optimization, vol. 43, pp. 403-416, 2011.

[11] S.-Z. Zhao, P. N. Suganthan, and Q. Zhang, "Decomposition-based multiobjective evolutionary algorithm with an ensemble of neighborhood sizes," IEEE Transactions on Evolutionary Computation, vol. 16, pp. 442-446, 2012.

[12] S. Mirjalili and S. Z. M. Hashim, "A new hybrid PSOGSA algorithm for function optimization," in Computer and information application (ICCIA), 2010 international conference on, 2010, pp. 374-377.

[13] A. Elhossini, S. Areibi, and R. Dony, "Strength Pareto particle swarm optimization and hybrid EA-PSO for multi-objective optimization," Evolutionary Computation, vol. 18, pp. 127-156, 2010.

[14] L. Tang and X. Wang, "A hybrid multiobjective evolutionary algorithm for multiobjective optimization problems," IEEE Transactions on Evolutionary Computation, vol. 17, pp. 20-45, 2013.

[15] A. Ibrahim, S. Rahnamayan, M. Vargas Martin, and K. Deb, "Fusion of Many-Objective Non-dominated Solutions Using Reference Points," in International Conference on Evolutionary Multi-Criterion Optimization, Münster, Germany, 2017, p. In presss.

[16] 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, 2014.

[17] H. Jain and K. Deb, "An evolutionary many-objective optimization algorithm using reference-point based nondominated sorting approach, part II: handling constraints and extending to an adaptive approach," Evolutionary Computation, IEEE Transactions on, vol. 18, pp. 602-622, 2014.

[18] S. Kukkonen and J. Lampinen, "GDE3: The third evolution step of generalized differential evolution," in Evolutionary Computation, 2005. The 2005 IEEE Congress on, 2005, pp. 443-450.

[19] A. J. Nebro, J. J. Durillo, J. Garcia-Nieto, C. C. Coello, F. Luna, and E. Alba, "Smpso: A new pso-based metaheuristic for multi-objective optimization," in Computational intelligence in miulti-criteria decision-making, 2009. mcdm'09. ieee symposium on, 2009, pp. 66-73.

[20] E. Zitzler, M. Laumanns, L. Thiele, E. Zitzler, E. Zitzler, L. Thiele, et al., "SPEA2: Improving the strength Pareto evolutionary algorithm," ed: Eidgenössische Technische Hochschule Zürich (ETH), Institut für Technische Informatik und Kommunikationsnetze (TIK), 2001.

[21] C. Qian, K. Tang, and Z. H. Zhou. "Selection Hyper-heuristics Can Provably Be Helpful in Evolutionary Multi-objective Optimization." International Conference on Parallel Problem Solving from Nature. Springer International Publishing, 2016.M.

[22] Li, S. Yang, and X. Liu. "Pareto or Non-Pareto: Bi-Criterion Evolution in Multiobjective Optimization." IEEE Transactions on Evolutionary Computation 20.5 (2016): 645-665.