2014 IEEE Congress on Evolutionary Computation (CEC)
July 6-11, 2014, Beijing, China

Sensitivity Analysis of Parallel Cell Coordinate System in Many-Objective Particle Swarm Optimization

Wang Hu, Gary G. Yen, and Xin Zhang

***Abstract*—Parallel Cell Coordinate System (PCCS) was proposed to evaluate the individual fitness in an archive and access the population progress in the evolutionary environment. In a Many-objective Optimization Problem (MaOP), it is much harder to tradeoff the convergence and diversity than in a Multiobjective Optimization Problem. To more effectively tackle the MaOPs, the PCCS and the aggregation-based approach are integrated into a Many-objective Optimization Particle Swarm Optimization (MaOPSO). In this paper, the sensitivity of PCCS is examined with respect to the number of objectives and the maximum size of an archive. The experimental results indicate that the MaOPSO performs better than MOEA/D in terms of IGD and HV metrics on the WFG test suit, and PCCS is not sensitive to the number of objectives and the maximum size of an archive.**

***Keywords*—particle swarm optimization; many-objective optimization problem (MaOP); many-objective optimization particle swarm optimization (MaOPSO); parallel cell coordinate system (PCCS)**

I. Introduction

PARALLEL Cell Coordinate System (PCCS) was originally introduced in [1] to estimate the density of a nondominated solution for maintaining the archive and selecting the gBest in Multiobjective Particle Swarm Optimization (MOPSO). PCCS was further extended to assess the evolutionary environment in [2] and an adaptive MOPSO based on PCCS (pccsAMOPSO) was accordingly proposed. In PCCS, the number of columns is the same as the number of objectives in a given Multiobjective Optimization Problem (MOP), and the number of rows, K, is the current size of the archive. K is not a user-defined parameter. During the evolutionary process, K is changed dynamically while the size of an archive varies. However, an archive in a Multiobjective Optimization Evolutionary Algorithm (MOEA) is commonly bounded to the maximum size, which is a user-chosen parameter. Therefore, it is necessary to thoroughly analyze the sensitivity of PCCS to know its fundamental characteristics.

Recently, Many-Objective Optimization Problem (MaOP), which refers to the simultaneous optimization of four or more objectives, has been gaining an increasing attention in the evolutionary multiobjective optimization (EMO) community. An important reason is due to distinctly different behaviors of MOEAs in a MaOP against in a MOP. A MOP is commonly referred as to those problems with two or three objectives.

W. Hu is with the School of Information and Software Engineering, University of Electronic Science and Technology of China, Chengdu, Sichuan 610054, China (e-mail: scuhuwang@uestc.edu.cn).

G. G. Yen is with the School of Electrical and Computer Engineering, Oklahoma State University, Stillwater, OK 74075 USA (phone: +1-405-744-7743; fax: +1-405-744-9198; e-mail: gyen@okstate.edu).

X. Zhang is with the School of Information and Software Engineering, University of Electronic Science and Technology of China, Chengdu, Sichuan 610054, China (e-mail: 422231937@qq.com).

This work was supported by the Fundamental Research Funds for the Central Universities (Grant No. ZYGX2013J078).

In order to examine the sensitivity of PCCS with respect to the number of objectives and the maximum size of the archive, a Many-Objective Particle Swarm Optimization (MaOPSO) algorithm based on PCCS was developed as the test algorithm in this paper. The sensitive experiments were empirically studied on the test instances with the various numbers of objectives and the varying maximum size of the archive.

The remainder of this paper is organized as follows. The background knowledge is briefly described in Section II. The test algorithm MaOPSO is presented to examine the sensitivity of PCCS in Section III. The experimental results are discussed in Section IV. The conclusions are summarized in Section V.

II. Background

A. MaOEA

The main reason why these MOEAs lose the search capability in a MaOP is largely due to the ineffective definition of Pareto optimality, which cannot effectively discriminate the solutions in a high-dimensional objective space. The proportion of nondominated solutions based on Pareto optimality among all the solutions found by an Evolutionary Algorithm (EA) dramatically grows with the increasing number of objectives, which inadvertently deteriorates the search ability of Pareto dominance-based algorithms. In addition, it is difficult to represent and visualize the complete Pareto front of a MaOP, because a high-dimensional front requires an exponentially large number of nondominated solutions with respect to the number of objectives.

Over the past few years, appreciable efforts have been dedicated to tackle these challenges in MaOPs. Many improvements from the existing MOEAs, such as dimensionality reduction techniques [3]-[6], modifications of Pareto dominance principle [7]-[12], ranking dominance-based [13]-[17], indicator-based [18]-[21], and aggregation-based methods [22]-[23], are proposed for the MaOPs. Nevertheless, the performance in terms of convergence and diversity of a Many-Objective Optimization Evolutionary Algorithm (MaOEA) is still far from meeting the requirements in real-world applications of the MaOPs

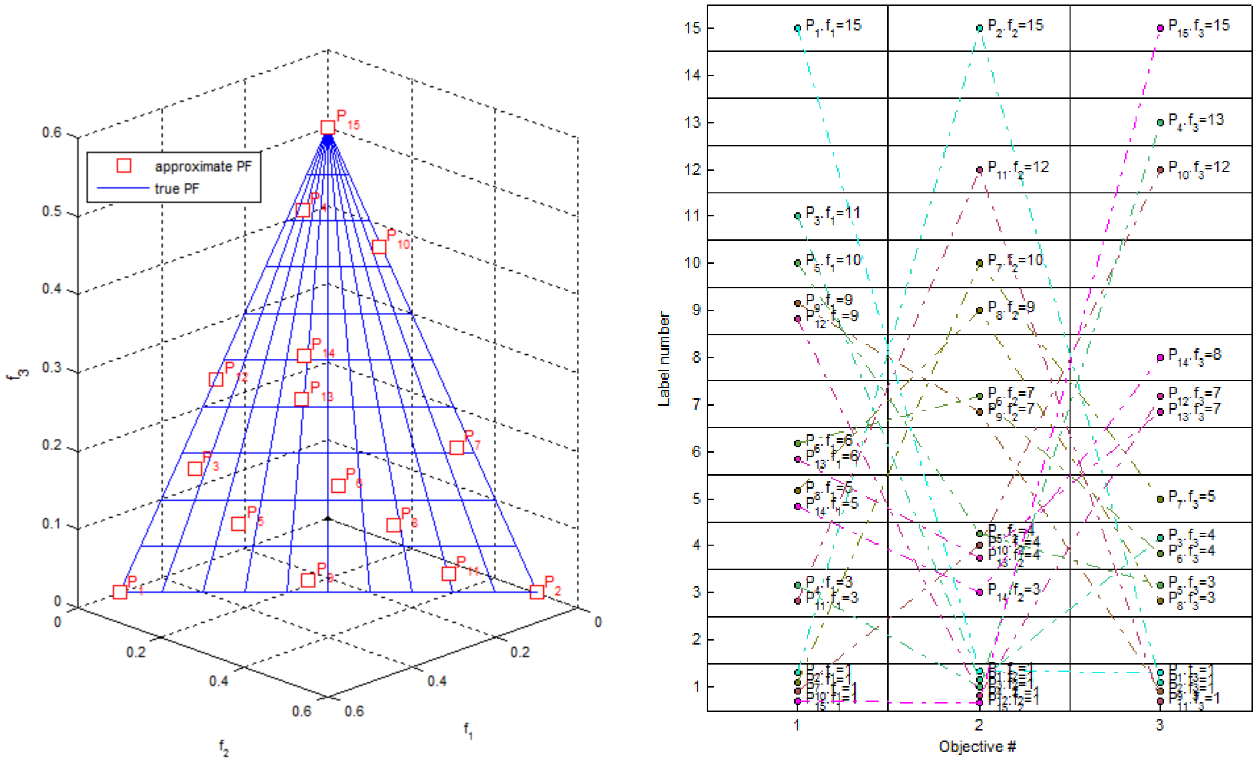

Fig. 1. An example for mapping an archive into the PCCS [2]. Left side: 15 nondominated solutions in the archive obtained by the proposed MaOPSO from DTLZ1 with three objectives. Right side: each objective of a solution in left side is mapped into a unique cell within a 2-D grid with 15 rows and 3 columns corresponding to 15 solutions in the archive of a 3-objective MaOP, respectively. For example, the parallel cell coordinate of P_{14} is (5,3,8).

[24]. The complication for balancing the convergence and diversity is much more severe in a MaOEA than in a MOEA.

Dimensionality reduction is an intuitive approach to address the curse of dimensionality in a MaOP. The linear principal component analysis and the nonlinear dimensionality reduction were proposed for reducing the relevant and irrelevant objectives, respectively [3]-[5]. The Pareto Corner Search Evolutionary Algorithm (PCSEA) was introduced to search for the corners of a Pareto front to identify and reduce the relevant objectives [6]. However, dimensionality reduction is at the cost of information loss of the original problem. Meanwhile, not all MaOPs can be reduced to their respective MOPs due to the inherent complexity. Therefore, the designs of non-reduction approaches are still the inevitable challenges for the MaOPs.

Some modifications of Pareto dominance, such as Dominance Area Control [7], ε-Dominance [8], k-Optimality [9], Grid Dominance [10], Fuzzy-based Pareto Dominance [11] were proposed to remedy the ineffectiveness of Pareto optimality in a MaOP. The number of nondominated solutions is decreased while the selection pressure is increased in these variants. The convergence performances of the new dominance relations were significantly better than that of the original Pareto dominance relation [12].

Ranking-dominance or fitness assignment approaches, such as Weighted Sum, Average Ranking [13], Maximum Ranking [14], Preference Order [15], Favor Relation [15], Global Detriment [16], are the alternatives for Pareto dominance in a MaOP. These approaches increase the selection pressure toward the Pareto front, but decrease the diversity of solutions [16]. In some cases, the population converges to a few solutions or a single solution [17].

Indicator-based Evolutionary Algorithm (IBEA) was proposed in [18] to select the individuals according to their maximal hypervolume contribution. IBEA and S metric selection evolutionary multiobjective optimization algorithm (SMS-EMOA) [19], using the time-consuming exact computation of hypervolume, have been found to perform well in balancing convergence and diversity in MaOPs. Hypervolume Estimation (HypE) [21] algorithm was recently developed to reduce the computational cost by Monte Carlo sampling. These indicator-based MaOEAs are beneficial to improve the diversity. However, they excessively prefer the extreme solutions in an approximate Pareto front, which may hinder the evolutionary process [20].

Aggregation-based approaches, such as Multiple Single Objective Pareto Sampling (MSOPS) [22] and Multiobjective Optimization Evolutionary Algorithm based on Decomposition (MOEA/D) [23], map a MaOP into a series of Single-objective Optimization Problems (SOPs) by the user-predefined weighted vectors. These approaches improve the convergence. However, they are subjected to the pre-defined weight vectors for the diversity. In addition, it is difficult to choose an appropriate weight vector to meet the requirements of the decision makers in a real-world application.

B. PCCS

PCCS proposed in [1]-[2] is a diversity evaluation mechanism for maintaining a well-distributed archive and selecting the gBest for a particle. PCCS is based on Parallel coordinates which is a popular way for visualizing high-dimensional geometry and analyzing multivariate data intuitively.

In PCCS, the m-th objective of the k-th nondominated solution in the archive, $f_{k,m}$, is mapped to an integral label number within a 2-D grid with $K \times M$ cells according to Eq. (1), where K is the current size of the archive and M is the number of objectives of a MaOP at hand,

$$L_{k,m} = \left\lceil K \frac{f_{k,m} - f_m^{\min}}{f_m^{\max} - f_m^{\min}} \right\rceil \tag{1}$$

Here, $\lceil x \rceil$ is a ceiling operator that returns the smallest integer which is not less than x. $f_m^{\max} = \max_k f_{k,m}$ and $f_m^{\min} = \min_k f_{k,m}$ are the maximum and minimum of the m-th objective value in the archive, respectively. $L_{k,m} \in \{1,2,\ldots,K\}$ is an integer number transformed from the real number, $f_{k,m}$, after normalization. $L_{k,m}$ is set to one if $f_{k,m} = f_m^{\min}$ to avoid zero as a denominator in special cases. It is noted that K, changing dynamically with the size of archive, is not a user-defined parameter. In a nondominated archive, each solution is expected to occupy a cell in each dimension alone if all the nondominated solutions are well-distributed perfectly in the approximate Pareto front. Therefore, in PCCS, the length of cell is automatically adjusted once any one of $f_m^{\max}$, $f_m^{\min}$, and the size of archive is changed.

Any set of points in Cartesian Coordinate System can be represented by Parallel Cell Coordinate (PCC) in a 2-D grid which can be visualized intuitively by the style of parallel axes.

An example for mapping an archive with 15 nondominated solutions into the PCCS is illustrated in Fig. 1. On the left subfigure, there are 15 nondominated solutions in the archive obtained by the proposed MaOPSO from DTLZ1 with three objectives. After the archive is mapped into the PCCS, there are 15 rows and three columns in the grid corresponding to 15 solutions in the archive of a three-objective MaOP, respectively. Each objective of a solution in the left subfigure is mapped into a unique cell coordinate in the right subfigure. All components of a nondominated solution, represented by the label number of the corresponding "parallel cell coordinates", are linked by a dash dotted line to display clearly in the right subfigure. For example, the parallel cell coordinate of P_{14} is (5,3,8).

The distance between two nondominated solutions in the unit of cell, named Parallel Cell Distance (*PCD*), is measured by the sum of the differences of cell coordinates over all objectives. The Parallel Cell Distance of two nondominated solutions P_i and P_j, $PCD(P_i,P_j)$, can be calculated according to Eq. (2) after they are respectively mapped to $L_{i,m}$ and $L_{j,m}$:

$$PCD(P_i, P_j) = \begin{cases} 0.5 & \text{if } \forall m \;\; L_{i,m} = L_{j,m} \\ \sum_{m=1}^{M} \left| L_{i,m} - L_{j,m} \right| & \text{otherwise} \end{cases} \tag{2}$$

If P_i and P_j are mapped into the same cells for all M dimensions in PCCS, the *PCD* value is set to 0.5 to avoid division by zero in Eq. (3).

The density of P_i, in the hyper-space formed by the archive in objective space can be measured by the *PCD* between P_i and all other members, P_j (j=1,2,…,K, $j \neq i$), in the archive according to Eq. (3).

$$Density(P_i) = \sum_{\substack{j=1 \\ j \neq i}}^{K} \frac{1}{PCD(P_i, P_j)^2} \tag{3}$$

From Eq. (3), the density of P_i is affected by all its neighbors. The nearer the neighbor is close to P_i, the larger density is contributed to P_i by the neighbor.

The density based on *PCD* will be used in both updating the archive and selecting the diversity-gBest.

The Potential of a nondominated solution P_i in the archive is defined by Eq. (4):

$$Potential(P_i) = \sum_{m=1}^{M} L_{i,m} \tag{4}$$

The Potential quantifies a nondominated solution among its competitors in the archive by combining the order relation along the optimization direction and the degree in the unit of cell in PCCS. Here, "Potential" is referring to "potential energy" in physics. It is expected that the potential energy of every individual in the archive gradually decreases to a steady state for improving the convergence through evolution process.

III. The Test Algorithm – MaOPSO

In order to examine the sensitivity of PCCS in terms of the number of objectives and the maximum size of an archive, a test algorithm based on PCCS is needed to optimize the MaOPs in this paper. As a case study, PSO is selected to serve as the nature-inspired meta-heuristic algorithm in the test MaOEA due to its fast convergence and relative simplicity.

It is interesting that some single-objective optimization methods are recalled to solve the MaOPs or MOPs to emphasis the convergence. Aggregation-based approaches, such as MSOPS and MOEA/D, map a MaOP into a series of Single-objective Optimization Problems (SOPs) by the user-predefined weighted vectors. These approaches improve the convergence. However, they are subjected to the pre-defined weight vectors for the diversity. In addition, it is difficult to choose an appropriate weight vector to meet the requirements of the decision makers in a real-world application.

In this test algorithm, named Many-objective Optimization Particle Swarm Optimization (MaOPSO), several extreme nondominated solutions are sought for at first by a Single-Objective Optimization Particle Swarm Optimization (SOPSO) through aggregating the objective functions with the pre-defined weight vectors. Then, the approximate Pareto front is extended from these extreme solutions by a PSO-based many-objective optimizer. The tested MaOPSO is described as follows.

Step 1 (Seeking M extreme nondominated solutions)

Step 1.1 Aggregate the M objective function of a MaOP into M SOPs through a pre-defined weight matrix. For a given MaOP, M is the number of objective functions; $\mathbf{x}=[x_1,x_2,\ldots,x_D]^T$, D is the number of decision variables. The aggregated SOPs can be expressed as:

$$SOP_i(\mathbf{x})=\sum_{j=1}^{M} w_{ij}\times f_j(\mathbf{x}).$$

Here, $[w_{ij}]=\begin{bmatrix} \varepsilon & \frac{1-\varepsilon}{M-1} & \cdots & \frac{1-\varepsilon}{M-1} \\ \frac{1-\varepsilon}{M-1} & \varepsilon & \cdots & \frac{1-\varepsilon}{M-1} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1-\varepsilon}{M-1} & \frac{1-\varepsilon}{M-1} & \cdots & \varepsilon \end{bmatrix}$, ε is a very small positive number. The sum of the weight components w_{ij} in each row i is equal to one, namely, $\sum_{j=1}^{M} w_{ij}=1$. i=1,2,…,M; w_{ij} is the weight component at the i-th row and the j-th column; $f_j(\mathbf{x})$ is the j-th objective function value of an individual $\mathbf{x}$.

Step 1.2 Optimize the M aggregated SOPs to generate M extreme nondominated solutions at the approximate Pareto front by a SOPSO through M separate runs. Here, a perturbation operator (e.g., Elitism Learning Strategy [17]) is beneficial to deal with the multi-modal problems.

Step 2 (Initialize the population of MaOPSO)

Step 2.1 Initialize randomly the population around the M extreme nondominated solutions within a region formed by the upper and lower boundaries of decision variables. For example, the particle i can be initialized as $\mathbf{x}_i = \mathbf{x}_{ES} + (\mathbf{U} - \mathbf{L}) \times rand()$, here, $\mathbf{x}_{ES}$ is the randomly selected extreme nondominated solution; $\mathbf{U}$ and $\mathbf{L}$ are the upper and lower boundaries. The M extreme nondominated solutions obtained at Step 1 are beneficial to improve the convergence of the MaOPSO.

Step 2.2 The objective function values are evaluated for particle i in the initial population as $\mathbf{F}(\mathbf{x}_i)=[f_{i1}(\mathbf{x}_i),\ldots f_{im}(\mathbf{x}_i),\ldots f_{iM}(\mathbf{x}_i)]^T$.

Step 2.3 Set the personal best solution, pBest, to its initial solution. For particle i, the position $\mathbf{x}_i$ and objective values $\mathbf{F}(\mathbf{x}_i)$ are set to $\mathbf{pBest}_i$ as its initial personal best solution.

Step 2.4 Initialize an external archive to store the nondominated solutions. Firstly, set the archive $\mathbf{A}=\mathbf{\varnothing}$. Secondly, for each particle P_i, set the position $\mathbf{x}_i$ and objective values $\mathbf{F}(\mathbf{x}_i)$ to a new solution s, if $\mathbf{F}(\mathbf{x}_i)$ is not dominated by any member of $\mathbf{A}$, then $\mathbf{A}=\mathbf{A}-\{a_j\}\cup\{s\}$, here, $\{a_j\}$ is the members in $\mathbf{A}$ who is dominated by s, if any.

Step 3 (Iterate the population of MaOPSO)

Step 3.1 Select candidates for the leaders of the population. Firstly, for each member a_k, k=1, 2,…K, K is the size of the archive, in $\mathbf{A}$, calculate its density according to Eq. (3). Secondly, Sort $\mathbf{A}$ in ascending order by the densities of its members. Thirdly, the set of candidate leader $\mathbf{CL}$ consists of the M extreme solutions and the top M (the number of objectives) solutions with largest densities.

Step 3.2 Select randomly a member from $\mathbf{CL}$ as the global best solution $\mathbf{gBest}$ of the particle i.

Step 3.3 Update the velocity $\mathbf{v}_i$ and position $\mathbf{x}_i$ of the particle i according to the PSO motion equation:

$$\begin{cases} \mathbf{v}_i(t+1)=\omega\mathbf{v}_i(t)+c_1r_1(\mathbf{pBest}_i-\mathbf{x}_i(t)) \\ \qquad\qquad +c_2r_2(\mathbf{gBest}-\mathbf{x}_i(t)) \\ \mathbf{x}_i(t+1)=\mathbf{x}_i(t)+\mathbf{v}_i(t+1) \end{cases}.$$

Step 3.4 Randomly select a nondominated solution from the archive as elitist to perturb the particle with a Gaussian mutation of variable range at a random dimension by Elitism Learning Strategy [17] if a random value is less than the linear decreasing learning rate from 1.0 to 0.

Step 3.5 Evaluate the objective function values of the particle i as $\mathbf{F}(\mathbf{x}_i)=[f_{i1}(\mathbf{x}_i),\ldots,f_{im}(\mathbf{x}_i),\ldots,f_{iM}(\mathbf{x}_i)]^T$.

Step 3.6 Update the pBest of a particle. Replace $\mathbf{F}(\mathbf{pBest}_i)$ and $\mathbf{pBest}_i$ with $\mathbf{F}(\mathbf{x}_i)$ and $\mathbf{x}_i$, respectively, if $\mathbf{F}(\mathbf{x}_i)$ dominates $\mathbf{F}(\mathbf{pBest}_i)$, or $\mathbf{F}(\mathbf{x}_i)$ and $\mathbf{F}(\mathbf{pBest}_i)$ are nondominated with each other.

Step 3.6 Update the external archive. Set $\mathbf{x}_i$ and $\mathbf{F}(\mathbf{x}_i)$ to a new solution s, if $\mathbf{F}(\mathbf{x}_i)$ is not dominated by any member of $\mathbf{A}$, then $\mathbf{A}=\mathbf{A}-\{a_j\}\cup\{s\}$, here, $\{a_j\}$ is those members in $\mathbf{A}$ who is dominated by s. Then discard the member in $\mathbf{A}$ with the maximal density calculated by Eq. (3) if the size of current $\mathbf{A}$ is larger than the pre-defined maximal size of $\mathbf{A}$.

Step 4 Check the terminal condition. Report the contents in $\mathbf{A}$ if t is larger than the pre-defined maximal generation T, otherwise, go to Step 3.

IV. Experiments

In this research, two experiments are performed on the MaOPs. One experiment is designed to compare the proposed algorithm MaOPSO with the state-of-the-art algorithm MOEA/D [23] on the WFG test suit. The other one is devoted to examine the sensitivity of PCCS in terms of the maximal archive size.

A. Benchmark Problems

To validate the test algorithm MaOPSO whether can work on MaOPs or not, the WFG [25] test suit is chosen as the benchmark MaOPs due to its rich characteristics. It includes non-separable problems, deceptive problems, a truly

TABLE I THE PROPERTIES OF WFG TEST SUITE

Problem	Objective	Separability	Modality	Bias	Geometry
WFG1	$f1{:}M$	separable	unimodal	polynomial, flat	convex, mixed
WFG2	$f1{:}M$-1	non-separable	unimodal	----	convex, disconnected
	fM	non-separable	multimodal	----	
WFG3	$f1{:}M$	non-separable	unimodal	----	linear, degenerate
WFG4	$f1{:}M$	separable	multimodal	----	concave
WFG5	$f1{:}M$	separable	deceptive		concave
WFG6	$f1{:}M$	non-separable	unimodal	----	concave
WFG7	$f1{:}M$	separable	unimodal	parameter dependent	concave
WFG8	$f1{:}M$	non-separable	unimodal	parameter dependent	concave
WFG9	$f1{:}M$	non-separable	multimodal, deceptive	parameter dependent	concave

degenerate problem, a mixed shape Pareto front problem, problems scalable in the number of position-related parameters, and problems with dependencies between position- and distance-related parameters. The WFG test suite provides a means of assessing the performance of optimization algorithms on a wide range of different problems. Each test instance in this suit is written as WFGk(M), k=1,2,…,9, k is the sequence number of the member in this test suit; M=3,4,5,7,10, M is the number of objectives in the resulted scalable MaOP. The properties of WFG1-9 are listed in Table I, [25].

The WFG2(3,4,5,7) is selected to analyze the sensitivity of PCCS with respect to the maximal archive size.

B. The Test Algorithm and Simulation Settings

The MaOPSO described in Section III is used to analyze the sensitivity of PCCS. Namely, the four test instances WFG2(3,4,5,7) are simulated by MaOPSO over the maximal archive size from 60 to 200 with an interval of 10. At the same time, the MOEA/D is chosen as the peer algorithm to compare the performance of MaOPSO, because MOEA/D is a popular and excellent algorithm in solving MOPs and MaOPs.

In MaOPSO, the size of population at Step 1 (SOPSO) is set to 20 and the maximal generation is set to 400, so that the total number of function evaluations is 10,000 for all test instances. At Step 3, the population size is 50 and the maximal generation is 100, so that the total number of function evaluations is 5,000. So the total number of function evaluations is $8{,}000\times M + 5{,}000$ for a test instance with M objectives. For simplicity, the inertia weight ω is adjusted from 0.9 down to 0.4 according to the linear decreasing strategy [26] in proportion to the generation loop variable. The personal acceleration factor c_1 and the social acceleration factor c_2 are set to the constant 1.429 [27]. Furthermore, the Elite Learn Strategy (ELS) with the learning rate (lr) [28], linearly decreased from 1.0 down to 0.1 with respect to the generation variable.

In MOEA/D, in order to set the population size approximate to that of MaOPSO, the parameter T (the number of neighboring sub-problems) is set to 20, and the parameter H (the number of total sub-problems) is set to be 13, 6, 5, 3, and 2 for three-, four-, five-, seven-, and ten-objective problems, respectively, so as to make the population size among the possible values equal to the closest integer to 100. The maximal generation is set to 400. So the total number of function evaluations is 40,000. The crossover rate is 1.00, while the mutation rate is $1/n$, where n is the number of decision variables. The weight vectors in MOEA/D are uniformly chosen according to their original paper, [23], so as to make the obtained approximate Pareto front well-distributed.

The performance metrics on each test instance are obtained from 30 independent runs to arrive at statistical significance. In order to provide the statistical quantifications on performance metrics, the non-parametric statistical hypothesis test, Mann-Whitney-Wilcoxon rank-sum test [29] (also called U-test) is applied to quantify whether one of two approximation fronts from independent observations tends to have a better performance in a statistical meaningful sense, especially when the performance metric values of the two approximation fronts obtained by two competing algorithms are very close to each other or even indiscriminative. All simulations in this experiment are performed on a 64-bit notebook PC with 1.2 GHz dual core CPU and 4GB memory.

C. Performance Metrics

Two performance metrics, Inverted Generational Distance (IGD) [23] and Hyper-Volume (HV) [30]-[31], are chosen to quantify the performance in terms of convergence and diversity. For the IGD metric, the true Pareto fronts of the test instances are required as the referenced fronts to measure the performance metric. The more the samples of a true Pareto front are available, the better the IGD metric will be for a MaOP, yet the higher the incurred computational cost will be. For balancing the accuracy and complexity, the size of the samples in a true Pareto front is set to around $(M\text{-}2)\times 5{,}000$, M is the number of objectives. Namely, about 5,000, 10,000, 15,000, 25,000, and 40,000 for three-, four-, five-, seven-, and ten-objective problems, respectively. For the HV metric, the relative HV with an acceptable tolerance is approximately calculated through the Monte Carlo simulation with 100,000 sampling points [31]. Each objective value of the reference points for evaluating HV is chosen as 2*(1:M) + 5, M is the number of objectives.

D. Results of Comparative Performance

The comparative experiment results in terms of IGD and HV are listed in Table II. The three items, mean, standard deviation (*Std.*), and significance symbol of U-test over all test instances by all peer algorithms, are filled in the form of "**mean (Std.) #**" in each data cell of Table II. Here, "#" stands for one of the significance symbol ("+," "-," or "=") of U-test, which respectively implies that the IGD of MaOPSO is better, worse, or same, than/as that of MOEA/D on WFGi(M) by the U-test at the significance level of α=5% for a two-tailed test.

TABLE II COMPARISONS IN TERMS OF IGD AND HV BETWEEN THE PROPOSED MAOPSO AND MOEA/D ON WFG TEST SUITE

		IGD		HV	
Function	M	MaOPSO	MOEA/D	MaOPSO	MOEA/D
WFG1	3	**2.14E-1 (6.5E-2)**	2.70E-1 (8.6E-3) +	**8.20E-1 (3.9E-2)**	7.72E-1 (6.3E-3) +
	4	3.29E-1 (2.7E-2)	**2.27E-1 (5.3E-2) -**	7.49E-1 (1.6E-2)	**8.11E-1 (5.1E-2) -**
	5	3.90E-1 (1.2E-2)	**3.94E-1 (3.0E-2) =**	7.10E-1 (8.9E-3)	**7.18E-1 (2.9E-2) =**
	7	**3.76E-1 (2.2E-2)**	4.12E-1 (3.9E-2) +	**6.49E-1 (7.3E-3)**	6.37E-1 (1.9E-2) +
	10	**4.37E-1 (1.2E-2)**	5.25E-1 (6.3E-2) +	**5.84E-1 (4.8E-3)**	5.65E-1 (2.1E-2) +
WFG2	3	**4.93E-2 (9.2E-3)**	1.20E-1 (7.3E-3) +	9.68E-1 (1.2E-2)	9.90E-1 (6.5E-4) -
	4	**1.06E-1 (1.6E-2)**	1.97E-1 (6.3E-3) +	9.44E-1 (1.6E-2)	9.97E-1 (2.9E-4) -
	5	**1.46E-1 (2.2E-2)**	4.79E-1 (1.5E-1) +	**9.30E-1 (2.0E-2)**	9.19E-1 (8.9E-2) =
	7	2.20E-1 (1.9E-2)	**2.06E-1 (4.4E-2) -**	8.95E-1 (2.5E-2)	**9.94E-1 (5.3E-3) -**
	10	2.56E-1 (1.3E-2)	**2.29E-1 (3.5E-2) -**	8.61E-1 (2.0E-2)	**9.33E-1 (5.8E-2) -**
WFG3	3	7.46E-2 (2.3E-2)	**2.59E-2 (4.6E-4) -**	8.85E-1 (1.0E-2)	**9.07E-1 (4.3E-4) -**
	4	2.48E-1 (4.4E-2)	**1.43E-1 (1.7E-2) -**	8.46E-1 (9.8E-3)	**8.69E-1 (1.9E-3) -**
	5	6.02E-1 (1.2E-1)	**5.10E-1 (2.6E-2) -**	8.04E-1 (1.3E-2)	**8.16E-1 (4.1E-3) -**
	7	3.03E+0 (8.5E-1)	**9.75E-1 (3.7E-1) -**	7.29E-1 (1.7E-2)	**7.41E-1 (1.7E-2) -**
	10	3.07E+1 (6.0E+0)	**1.07E+1 (4.8E+0) -**	**6.79E-1 (1.5E-2)**	6.19E-1 (1.8E-2) +
WFG4	3	**5.92E-2 (2.9E-3)**	1.25E-1 (8.0E-3) +	**9.58E-1 (4.5E-4)**	9.54E-1 (7.7E-4) +
	4	**1.35E-1 (7.9E-3)**	3.16E-1 (4.2E-2) +	**9.79E-1 (3.9E-4)**	9.69E-1 (1.2E-3) +
	5	**2.03E-1 (1.3E-2)**	7.45E-1 (1.7E-1) +	**9.88E-1 (3.8E-4)**	9.67E-1 (4.9E-3) +
	7	**2.69E-1 (1.8E-2)**	2.74E-1 (6.6E-2) =	**9.94E-1 (2.6E-4)**	9.14E-1 (4.4E-2) +
	10	2.80E-1 (1.9E-2)	**2.01E-1 (3.1E-2) -**	**9.97E-1 (1.2E-4)**	7.14E-1 (2.2E-2) +
WFG5	3	**6.82E-2 (5.6E-3)**	9.40E-2 (1.4E-3) +	**9.30E-1 (2.2E-3)**	9.29E-1 (6.0E-4) +
	4	**1.50E-1 (6.5E-3)**	3.16E-1 (7.9E-3) +	**9.45E-1 (2.3E-3)**	9.42E-1 (8.3E-4) +
	5	**2.21E-1 (1.5E-2)**	3.89E-1 (3.7E-2) +	**9.51E-1 (2.6E-3)**	9.50E-1 (1.6E-3) =
	7	3.09E-1 (2.8E-2)	**2.91E-1 (1.4E-2) -**	**9.53E-1 (3.1E-3)**	8.26E-1 (2.1E-2) +
	10	3.10E-1 (3.0E-2)	**1.44E-1 (1.2E-2) -**	**9.51E-1 (2.8E-3)**	6.80E-1 (1.7E-2) +
WFG6	3	1.04E-1 (2.3E-2)	**1.00E-1 (2.3E-3) =**	9.11E-1 (1.8E-2)	**9.54E-1 (1.8E-3) -**
	4	**1.97E-1 (2.1E-2)**	3.08E-1 (2.2E-2) +	**9.17E-1 (1.7E-2)**	9.03E-1 (4.5E-2) =
	5	**2.79E-1 (2.2E-2)**	5.97E-1 (6.6E-2) +	9.22E-1 (1.8E-2)	**9.51E-1 (1.3E-2) -**
	7	**3.86E-1 (2.3E-2)**	5.32E-1 (8.2E-2) +	**9.22E-1 (2.1E-2)**	9.00E-1 (3.7E-2) +
	10	**4.91E-1 (2.0E-2)**	6.62E-1 (1.6E-1) +	**9.31E-1 (1.7E-2)**	7.38E-1 (9.8E-2) +
WFG7	3	**5.63E-2 (1.3E-3)**	8.99E-2 (8.5E-4) +	**9.59E-1 (2.2E-4)**	9.56E-1 (2.8E-4) +
	4	**1.32E-1 (3.1E-3)**	3.13E-1 (4.7E-3) +	**9.79E-1 (3.6E-3)**	9.72E-1 (2.7E-4) +
	5	**2.04E-1 (5.5E-3)**	4.84E-1 (5.0E-3) +	**9.85E-1 (5.2E-3)**	9.75E-1 (3.3E-4) +
	7	**3.25E-1 (1.0E-2)**	6.48E-1 (6.3E-2) +	**9.67E-1 (2.0E-2)**	9.59E-1 (1.2E-2) +
	10	**4.66E-1 (2.2E-2)**	8.95E-1 (1.5E-1) +	**9.33E-1 (3.3E-2)**	8.84E-1 (5.4E-2) +
WFG8	3	**1.28E-1 (1.4E-2)**	1.42E-1 (1.6E-2) +	**9.33E-1 (1.9E-2)**	9.28E-1 (1.5E-2) =
	4	**2.05E-1 (1.4E-2)**	3.65E-1 (1.6E-2) +	**9.43E-1 (2.2E-2)**	9.23E-1 (2.0E-2) +
	5	**2.71E-1 (2.1E-2)**	5.44E-1 (6.5E-2) +	**9.24E-1 (3.2E-2)**	9.14E-1 (1.2E-2) =
	7	**3.69E-1 (3.0E-2)**	5.74E-1 (3.8E-2) +	**8.19E-1 (5.1E-2)**	7.66E-1 (2.2E-2) +
	10	**5.07E-1 (4.1E-2)**	6.32E-1 (3.5E-2) +	**6.89E-1 (3.8E-2)**	5.74E-1 (3.7E-2) +
WFG9	3	**6.61E-2 (5.3E-3)**	9.78E-2 (1.7E-3) +	9.03E-1 (1.1E-2)	**9.27E-1 (2.5E-3) -**
	4	**1.64E-1 (2.7E-2)**	3.22E-1 (1.8E-2) +	8.60E-1 (3.6E-2)	**8.73E-1 (4.4E-2) =**
	5	**2.68E-1 (2.1E-2)**	3.38E-1 (3.6E-2) +	8.25E-1 (2.7E-2)	**8.80E-1 (4.6E-2) -**
	7	7.76E-1 (1.3E-1)	**5.57E-1 (9.7E-2) -**	**8.27E-1 (2.0E-2)**	7.49E-1 (3.1E-2) +
	10	**2.61E+1 (4.3E+0)**	2.70E+1 (1.1E-2) +	**8.22E-1 (1.2E-2)**	5.94E-1 (2.8E-2) +
Better(+)			30		25
Same(=)			3		7
Worse(-)			12		13

The three items, mean, standard deviation (*Std.*), and significance symbol of U-test, are filled in the form of "**mean (Std.) #**" in each data cell. Here, "#" stands for one of the significance symbol ("+,""-," or "=") of U-test.

The numbers of the same significance symbols ("+," "-," and "=") are respectively summed up from each column and filled in the last three rows with the titles, "Better(+)," "Worse(-)," and "Same(=)," respectively. Meanwhile, the best value of IGD among all algorithms on each test instance is highlighted by **boldface** at each data row in Table II.

From Table II, as a whole, the MaOPSO outperforms MOEA/D, because MaOPSO obtains 30 and 25 statistically better values of IGD and HV, respectively, out of the 45 WFG test instances, while MOEA/D acquires 12 and 13 better values of IGD and HV, respectively, out of the same test instances.

In this table, the MaOPSO underperforms MOEA/D on WFG3 (with the degenerated Pareto front). The reason for this poor performance of MaOPSO is that the candidate leaders, which are selected from the archive according to the

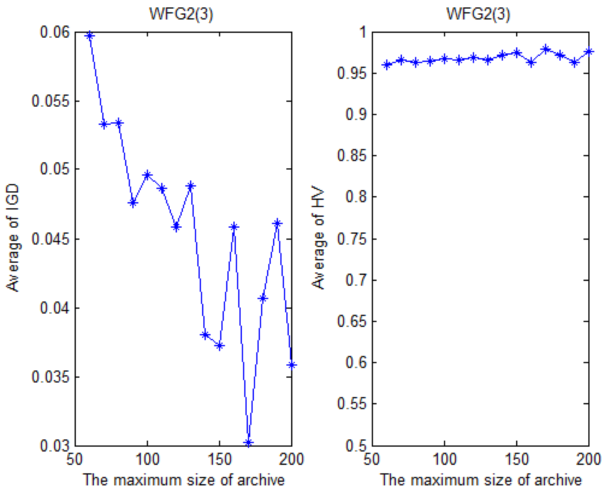

Fig. 2. The average IGD and HV of WFG2(3)

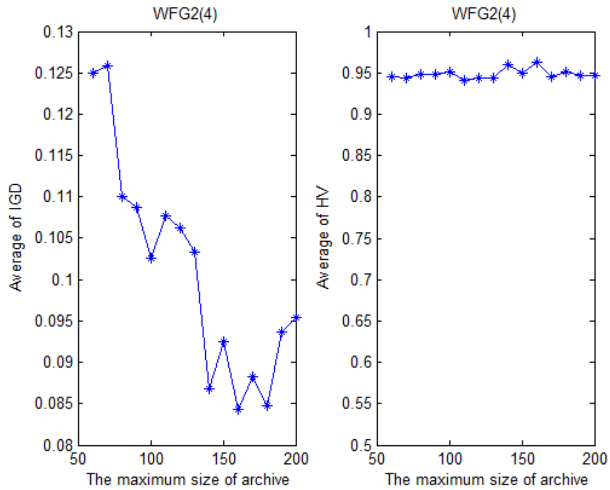

Fig. 3. The average IGD and HV of WFG2(4)

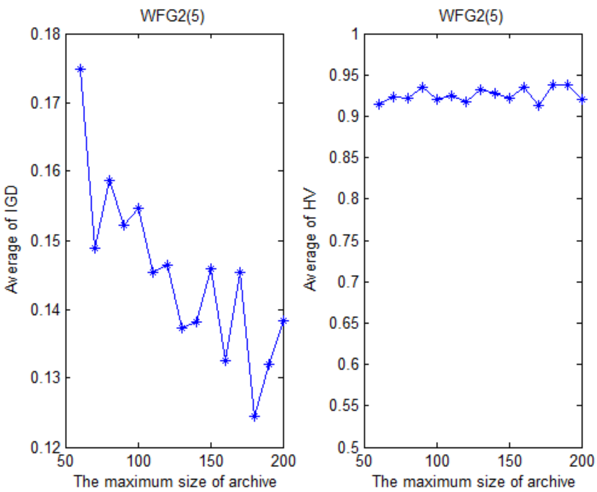

Fig. 4. The average IGD and HV of WFG2(5)

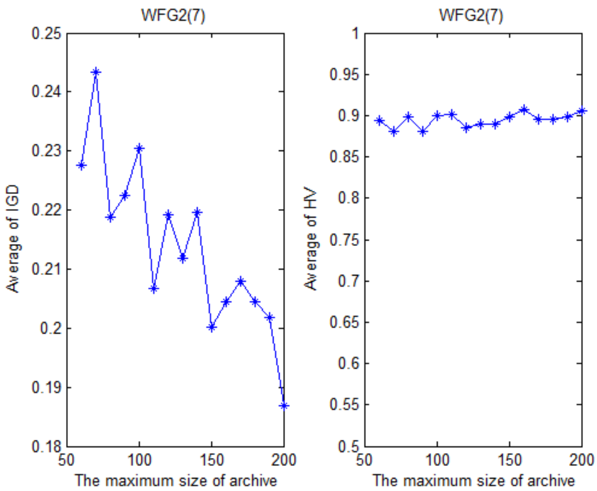

Fig. 5. The average IGD and HV of WFG2(7)

large density, may incidentally attract some individuals to fly away from a degenerated Pareto front to increase the diversity.

From the comparative experiment, MaOPSO with the PCCS can effectively works on the MaOPs. Even more, the comprehensive performance in terms of convergence and diversity, which are measured by IGD and HV, is better than MOEA/D.

E. Results of PCCS Sensitivity

In Eq. (3), the number of columns in PCCS is the same as the number of objectives in a given MaOP. And the number of rows in PCCS, K, is the current size of the archive. It is noted that K, changing dynamically with the size of archive, is not a user-defined parameter. However, its maximum value which is the archive size is a user-chosen parameter. To examine the sensitivity to the granularity of the PCCS, namely, the maximum size of archive, a special experiment is performed to analyze the sensitivity in terms of the maximum archive size in the MaOPs.

In this experiment, the MaOPSO is used to optimize the WFG2 with three-, four-, five-, and seven-objectives, labelled as WFG2(3), WFG2(4), WFG2(5), and WFG2(7), respectively. The performance metrics, IGD and HV, are evaluated over the maximum archive size varying from 60 to 200 with an interval of ten. The average IGD and the average HV over ten independent runs are plotted in the Figs. (2)-(5).

From Figs. (2)-(5), the average IGD in the left subfigures is decreased with the increase of the maximum archive size. According to the definition of IGD, the IGD value, calculated from the same true Pareto front, will decrease when the sample size in an approximate Pareto front increases. The experimental result, as expected, is not sufficient to draw a conclusion regarding the sensitivity on the maximum archive size in terms of IGD. On the other hand, the average HV in the right subfigures almost stays at the same value when the maximum archive size increases from 60 to 200. This result illustrates that the performance of PCCS-based MaOPSO in solving MaOPs is not sensitive to the maximum archive size.

From another viewpoint, the IGD and HV values become worse as the number of objectives increases. For example, the IGD average values of WFG2(3,4,5,7) are respectively about 0.05, 0.1, 0.145, and 0.205 when the maximal archive size is 100 in Figs. (2)-(5). And the HV average values of WFG2(3,4,5,7) are respectively about 0.96, 0.95, 0.93, and 0.88 in the same case. It indicates that the difficulty of a MaOEA increases when the number of objectives grows.

V. Conclusion

PCCS was proposed to evaluate the individual fitness in the archive and access the population progress in the evolutionary environment. A MaOPSO is proposed to tackle the MaOPs in this paper. The PCCS and aggregation-based approach are seamlessly integrated in the designed MaOPSO to dynamically balance the convergence and diversity in the evolutionary process. In order to characterize the application prospect of PCCS, the sensitivity of PCCS is examined on the number of objectives and the maximum size of an archive in MaOPs. Our experimental result shows that the MaOPSO outperforms MOEA/D in terms of IGD and HV over the WFG test suit with a large scale of objectives from three to ten. Another experimental result indicates that PCCS is not sensitive to the maximum archive size in the MaOPs.

Acknowledgment

The first author as a visiting scholar at Oklahoma State University would like to thank the China Scholarship Council for financial support.

References

[1] W. Hu and G. G. Yen, “Density estimation for selecting leaders and maintaining archive in MOSPO,” in *Proc. IEEE Congr. Evol. Comput.*, 2013, pp. 181-188.

[2] W. Hu and G. G. Yen, “Adaptive multiobjective particle swarm optimization based on parallel cell coordinate system,” *IEEE Trans. Evol. Comput.*, early access: http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=6692894

[3] K. Deb and D. K. Saxena, “Searching for Pareto-optimal solutions through dimensionality reduction for certain large-dimensional multiobjective optimization problems,” in *Proc. IEEE Congr. Evol. Comput.*, 2006, pp. 3353-3360.

[4] D. K. Saxena and K. Deb, “Non-linear dimensionality reduction procedures for certain large-dimensional multiobjective optimization problems: employing correntropy and a novel maximum variance unfolding,” in *Proc. Evol. Multi-Criterion Optimization*, 2007, pp. 772-787.

[5] D. K. Saxena, J. Duro, A. Tiwari, K. Deb, and Q. Zhang, “Objective reduction in many-objective optimization: linear and nonlinear algorithms,” *IEEE Trans. Evol. Comput.*, vol. 17, no. 1, pp. 77-99, Feb. 2013.

[6] H. K. Singh, A. Isaacs, and T. Ray, “A Pareto corner search evolutionary algorithm and dimensionality reduction in many-objective optimization problems,” *IEEE Trans. Evol. Comput.*, vol. 15, no. 4, pp. 539-556, Aug. 2011.

[7] H. Sato, H. E. Aguirre, and K. Tanaka, “Controlling dominance area of solutions and its impact on the performance of MOEAs,” in *Proc. Evol. Multi-Criterion Optimization*, 2007, pp. 5-20.

[8] M. Laumanns, L. Thiele, K. Deb, and E. Zitzler, “Combining convergence and diversity in evolutionary multiobjective optimization,” *Evol. Comput.*, vol. 10, no. 3, pp. 263-282, Sep. 2002.

[9] M. Farina and P. Amato, “A fuzzy definition of ‘optimality’ for many-criteria optimization problems,” *IEEE Tran. Syst. Man, Cybern., Part A: Syst. Hum.*, vol. 34, no. 3, pp. 315–326, May 2004.

[10] S. X. Yang, M. Q. Li, and J. H. Zheng, “A grid-based evolutionary algorithm for many-objective optimization,” *IEEE Trans. Evol. Comput.*, early access.

[11] Z. He, G. G. Yen, and J. Zhang, “Fuzzy-based Pareto optimality for many-objective evolutionary algorithms,” *IEEE Trans. Evol. Comput.*, early access.

[12] L. Batista, F. Campelo, F. Guimaraes, and J. Ramirez, “A comparison of dominance criteria in many-objective optimization problems,” in *Proc. IEEE Congr. Evol. Comput.*, 2010, pp. 2359-2366.

[13] P. J. Bentley and J. P. Wakefield, “Finding acceptable solutions in the Pareto-optimal range using multiobjective genetic algorithms,” in *Online World Conf. on Soft Computing in Design and Manufacturing*, 1997, pp. 231-240.

[14] F. di Pierro, S. T. Khu, and D. A. Savić, “An investigation on preference order ranking scheme for multiobjective evolutionary optimization,” *IEEE Trans. Evol. Comput*, vol. 11, no. 1, pp. 17-45, Feb. 2007.

[15] N. Drechsler, R. Drechsler, B. Becker, “Multi-objective optimization based on relation favour,” in *Proc. Evol. Multi-Criterion Optimization*, 2001, pp. 154-166.

[16] M. Garza-Fabre1, G. T. Pulido1, and C. A. Coello Coello, “Ranking methods for many-objective optimization,” in *Proc. Mexican International Conf. on Artificial Intelligence*, 2009, pp. 633-645.

[17] D. Corne and J. Knowles, “Techniques for highly multiobjective optimization: Some non-dominated points are better than others,” in *Proc. Genetic and Evolutionary Computation Conference*, 2007, pp.773-780.

[18] E. Zitzler, and S. Künzli, “Indicator-based selection in multiobjective search,” in *Parallel Problem Solving from Nature*, 2004, pp. 832-842.

[19] N. Beume, B. Naujoks, and M. Emmerich, “SMS-EMOA: multiobjective selection based on dominated hypervolume,” *Eur. J. Oper. Res.*, vol. 181, no. 3, pp. 1653-1669, Sep. 2007.

[20] T. Wagner, N. Beume, and B. Naujoks, “Pareto-, aggregation-, and indicator-based methods in many-objective optimization,” in *Proc. Evol. Multi-Criterion Optimization*, 2007, pp. 742-756.

[21] J. Bader and E. Zitzler, “HypE: an algorithm for fast hypervolume-based many-objective optimization,” *Evol. Comput.*, vol. 19, no. 1, pp. 45-76, Jul. 2011.

[22] E. J. Hughes, "Multiple single objective Pareto sampling," in *Proc. IEEE Congr. Evol. Comput*, 2003, pp. 2678-2684.

[23] 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, Dec. 2007.

[24] R. C. Purshouse, C. Jalbă, and P. J. Fleming, “Preference-driven coevolutionary algorithms show promise for many-objective optimization,” in *Proc. Evol. Multi-Criterion Optimization*, 2011, pp. 136-150.

[25] S. Huband, P. Hingston, L. Barone, R. L. While, “A review of multiobjective test problems and a scalable test problem toolkit,” *IEEE Trans. Evol. Comput.*, vol. 10, no. 5, pp. 447-506, Oct. 2006.

[26] Y. Shi and R. C. Eberhart, “Empirical study of particle swarm optimization,” in *Proc. IEEE Congr. Evol. Comput.*, 1999, pp. 1945-1950.

[27] M. Clerc, “The swarm and the queen: Toward a deterministic and adaptive particle swarm optimization,” in *Proc. IEEE Congr. Evol. Comput.*, 1999, pp. 1951–1957.

[28] Z. H. Zhan, J. Zhang, Y. Li, and H. S. Chung, “Adaptive particle swarm optimization,” *IEEE Trans. Syst. Man, Cybern. B, Cybern.*, vol. 39, no. 6, pp.1362-1381, Dec. 2009.

[29] S. Robert, J. Torrie, and D. Dickey, *Principles and Procedures of Statistics: A Biometrical Approach*, McGraw-Hill: New York, NY, 1997.

[30] E. Zitzler, “Evolutionary algorithms for multiobjective optimization: methods and applications,” *Ph.D. dissertation*, Swiss Federal Institute of Technology, Zurich, Germany, 1999.

[31] L. While, P. Hingston, L. Barone, and S. Huband, “A faster algorithm for calculating hypervolume,” *IEEE Trans. Evol. Comput.*, vol. 10, no. 1, pp. 29-38, Feb. 2006.