

An Investigation on Evolutionary Gradient Search for Multi-objective Optimization

C. K. Goh, Y. S. Ong and K. C. Tan

Abstract—Evolutionary gradient search is a hybrid algorithm that exploits the complementary features of gradient search and evolutionary algorithm to achieve a level of efficiency and robustness that cannot be attained by either techniques alone. Unlike the conventional coupling of local search operators and evolutionary algorithm, this algorithm follows a trajectory based on the gradient information that is obtain via the evolutionary process. In this paper, we consider how gradient information can be obtained and used in the context of multi-objective optimization problems. The different types of gradient information are used to guide the evolutionary gradient search to solve multi-objective problems. Experimental studies are conducted to analyze and compare the effectiveness of various implementations.

Index Terms—Evolutionary algorithm, gradient search, multi-objective optimization

I. INTRODUCTION

EVOLUTIONARY algorithms (EAs) are a class of stochastic optimization techniques that have been successfully applied to real world optimization problems involving multiple non-commensurable and often competing design specifications and constraints. Nonetheless, it is well known that EAs, while effective as global optimizers, require considerable time to find the local optimal. A number of works are thus focused on the hybridization of EAs and local search operators to cope with the increasing complexity of real-world environments in the recent years. In a wider context, hybrid algorithms are also known as Memetic algorithms.

At present, researchers are beginning to appreciate the strengths of hybrid algorithms for solving multi-objective (MO) optimization problems. The few well-known examples of multi-objective Memetic algorithms (MOMAs) includes the multi-objective genetic local search (MOGLS) [8], [12] and the Memetic Pareto achieved evolutionary strategy (M-PAES) [11]. The MOGLS is probably the first hybrid algorithm proposed for multi-objective (MO) optimization which is an aggregation-based MO evolutionary algorithm (MOEA). In this approach, a number of neighbors are generated using mutation and the best neighbor replaces the original solution. The M-PAES works by incorporating a population of solution that undergoes crossover and applying PAES as the local search operator [2]. Apart from the development of new MOMAs, issues such as the balance between exploration and

exploitation is also explored in [7], [18]. More recently, Shukla [15] investigated the effectiveness of two different gradient based stochastic search methods as mutation operators in the non-dominated sorting genetic algorithm II (NSGAI) [3].

Evolutionary gradient search (EGS) [1], [13] represents a hybrid algorithm that combines the features of gradient search and evolutionary strategies. Unlike the conventional coupling of local search operators and evolutionary algorithm, the basic idea of EGS is to follow a trajectory based on the gradient information obtained via a evolutionary process. It has been shown that EGS is able to find good solutions with the efficiency of gradient search techniques while retaining the robustness of evolutionary approaches.

This paper considers the potential of EGS for MO optimization. Successful implementation of a multi-objective EGS (MO-EGS) requires modification to the existing algorithm to account for issues unique to MO optimization. Firstly, it is desirable to find a set of uniformly distributed and diverse solutions that are as close as possible to the Pareto-optimal front. Secondly, appropriate fitness assignment schemes are required to provide the gradient information necessary to guide the optimization process. We consider how elitism can be applied and used to guide the MO-EGS to sample the entire Pareto front. For the latter issue, we discuss three different fitness assignment approaches namely, 1) random weights aggregation, 2) goal programming, and 3) performance indicator. Finally, we compare the performance of MO-EGS against the non-dominated sorting genetic algorithm II (NSGAI) [3] and demonstrate its effectiveness over four well-known MO test functions.

The remainder of this paper is organized as follows: Some background information is provided in Section II while details of the MO-EGS are described in Section III.

II. MO OPTIMIZATION

The concepts of Pareto dominance and Pareto optimality are fundamental in MO optimization, with Pareto dominance forming the basis of solution quality. Unlike SO optimization where a complete order exist (i.e, $f_1 \leq f_2$ or $f_1 \geq f_2$), \vec{X}^{n_x} is partially-ordered when multiple objectives are involved. In fact, there are three possible relationships between the solutions that is defined by Pareto dominance.

Definition 1: Weak Dominance: $\vec{f}_1 \in \vec{F}^M$ weakly dominates $\vec{f}_2 \in \vec{F}^M$, denoted by $\vec{f}_1 \preceq \vec{f}_2$ iff $f_{1,i} \leq f_{2,i} \forall i \in \{1, 2, \dots, M\}$

Definition 2: Strong Dominance: $\vec{f}_1 \in \vec{F}^M$ strongly dominates $\vec{f}_2 \in \vec{F}^M$, denoted by $\vec{f}_1 \prec \vec{f}_2$ iff $f_{1,i} \leq$

C. K. Goh is with the Data Storage Institute, Agency for Science, Technology and Research, Singapore (e-mail: ckgo@nus.edu.sg)

K. C. Tan is with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore 117576 (e-mail: el-tank@nus.edu.sg)

Y. S. Ong is with the School of Computer Engineering, Nanyang Technological University, Singapore 639798 (e-mail: asysong@ntu.edu.sg)

$f_{2,i} \forall i \in \{1, 2, \dots, M\}$ and $f_{1,j} < f_{2,j} \exists j \in \{1, 2, \dots, M\}$

Definition 3: Incomparable: $\vec{f}_1 \in \vec{F}^M$ is incomparable with $\vec{f}_2 \in \vec{F}^M$, denoted by $\vec{f}_1 \sim \vec{f}_2$ iff $f_{1,i} > f_{2,i} \exists i \in \{1, 2, \dots, M\}$ and $f_{1,j} < f_{2,j} \exists j \in \{1, 2, \dots, M\}$

With the definition of Pareto dominance, we are now in the position to consider the set of desirable solutions in MO optimization.

Definition 4: Pareto Optimal Front: The Pareto optimal front, denoted as PF^* , is the set of nondominated solutions with respect to the objective space such that $\text{PF}^* = \{\vec{f}_i^* | \nexists \vec{f}_j^* \prec \vec{f}_i^*, \vec{f}_j^* \in \vec{F}^M\}$

Definition 5: Pareto Optimal Set: The Pareto optimal set, denoted as PS^* , is the set of solutions that are nondominated in the objective space such that $\text{PS}^* = \{\vec{x}_i^* | \nexists \vec{F}(\vec{x}_j) \prec \vec{F}(\vec{x}_i^*), \vec{F}(\vec{x}_j) \in \vec{F}^M\}$

The set of tradeoff solutions is known as the Pareto optimal set and these solutions are also termed “noninferior”, “admissible” or “efficient” solutions. The corresponding objective vectors of these solutions are termed “non-dominated” and each objective component of any non-dominated solution in the Pareto optimal set can only be improved by degrading at least one of its other objective components [14].

III. MO EVOLUTIONARY GRADIENT SEARCH

This section describes the MO-EGS framework for MO optimization. The first part describes the main modifications made to the canonical EGS described in [1], [13]. The second part describes the different fitness assignment approaches that are necessary in the estimation of the gradient used in the optimization process.

A. Basic Procedure

The basic EGS procedure consists of two steps, 1) the estimation of the gradient by an evolutionary approach, and 2) the update of the solution in a steepest descent manner. This process is iterated until the predetermined stopping criteria is satisfied.

In contrast to single objective optimization, the following two issues must be addressed before gradient information can be beneficial in the context of multi-objective optimization:

- the trajectory specified by the gradient will lead the EGS to a single point on the Pareto front,
- a solution can be incomparable as well and this does not translate well in a steepest descent approach.

Therefore the basic EGS do not apply to MO optimization directly. The MO-EGS algorithm must be capable of finding a set of nondominated solutions. A simple approach is to introduce the concept of population by conducting multi-directional search simultaneously. However, this approach does not make good use of the information regarding the gaps in the nondominated solutions found.

Unlike canonical EGS that works with a single solution, this proposed framework places no restriction on the population size. After the random initialization of the population, all individuals are evaluated and nondominated solutions are

updated into the fixed-size archive. The archive is an important component of evolutionary MO techniques and it has been shown to be crucial to ensure the convergence. A candidate solution is added to the archive if it is not dominated by any members in the archive, while any archive members dominated by this solution will be removed. When the predetermined archive size is reached, a recurrent truncation process based on niche count is used to eliminate the most crowded archive member.

As in the case of EGS, N trial solutions are generated randomly using normal distribution mutation. The i -th trial solution of the parent \vec{f} is given as

$$\vec{f}_i = \vec{f} + \vec{z} \quad (1)$$

where $z_j = N(0, \sigma_t)$ and σ_t controls the mutation strength. These trial solutions will then undergo evaluation and nondominated trial solutions are subsequently added into the archive. The different fitness assignment schemes investigated in this work will be discussed in the next section.

Instead of updating the \vec{f} directly, a copy of it (denoted as \vec{c}) is made and updated in the following manner,

$$\vec{c}' = \vec{c} - \frac{\sqrt{N}}{\kappa} \frac{\vec{\delta} \cdot \vec{z}}{\|\vec{\delta} \cdot \vec{z}\|} \quad (2)$$

$$\vec{\delta} = \sum_{i=1}^N (F_i - F) \quad (3)$$

where $\vec{\delta}$ is the gradient vector. F and F_i are the fitness of the parent and the i -th trial solution, respectively. The updated solution \vec{c}' is compared against the parent solution and accepted only if it dominates the parent. If \vec{c}' is dominated, the parent is replaced by a nondominated solution selected from the archive by means of tournament selection. The criterion for selection is based on niche count to promote the sampling of the entire Pareto front. In the situation where both \vec{f} and \vec{c}' are nondominated, the parent is either replaced by \vec{c}' or by a nondominated solution tournament selected from the archive.

After the updating of the solution, the step size σ is adapted as follows:

$$\sigma_{t+1} \begin{cases} \sigma_t \zeta, & \text{if } \vec{c}' \prec \vec{f} \\ \sigma_t / \zeta, & \text{if } \vec{f} \prec \vec{c}' \end{cases} \quad (4)$$

However, it is well-known that $\sigma_t \rightarrow 0$ if left unchecked and the norm is to specify some lower bound. In this work, we specified both lower and upper thresholds, and allow σ_t to “oscillate” between the specified bounds. When σ_t reaches the lower threshold, the update equation becomes $\sigma_{t+1} = \sigma_t \zeta$. Vice versa, when σ_t reaches the upper threshold, the update equation becomes $\sigma_{t+1} = \sigma_t / \zeta$.

B. Incorporating MO Gradient Information

Since EGS for single-objective (SO) optimization uses the objective value to estimate the gradient, it is necessary to select appropriate means of representing the fitness of the solutions.

1) *The random weight approach*: The weighted approach aggregates the set of objectives into a single fitness function by multiplying each objective with some weight. Thus the fitness assigned to the i -th trial solution \vec{f}'_i is given as,

$$F_i = \sum_{j=1}^M w_j \cdot f'_{i,j} \quad (5)$$

where $w_j \in [0, 1]$ is the weight assigned to the j -th objective. The weights are selected such that $\sum_{j=1}^M w_j = 1$.

One problem associated with this method is the need to specify the weights to guide the evolutionary process to sample the entire Pareto front. Murata and Ishibuchi [12] suggested the use of normalized random weights to generate different search trajectories during the evolution. In this approach, the weight vector assigned to each solution is generated randomly before the evaluation process. Despite initial criticism that random weights have the same limitation of sensitivity to the shape of PF*, Jin *et al* [10] have shown that the weighted approach can transverse the entire Pareto front regardless of the shape. The algorithm applying the random weight approach to estimate the gradient is denoted here as MO-EGS-RW.

2) *The goal programming approach*: In goal programming, each of the objective is assigned a goal or target value to be achieved. The fitness of the i -th trial solution \vec{f}'_i of \vec{f} is given as,

$$F_i = \sum_{j=1}^M \|f'_{i,j} - g_j\| \quad (6)$$

where g_j is the goal assigned to the j -th objective and $\|\cdot\|$ refers to Euclidean Norm.

It is difficult to define the goals without *a priori* knowledge of the problem. In order to apply the goal programming approach in MOGLS, it is necessary to know how the trial solution performs relative to the parent. This work adopts a simple method of generating the goals randomly in the following manner

$$g_i = f_i - z_i \quad (7)$$

where f_i is the i -th objective of solution \vec{f} and $z_i \in [0, 1]$ is generated at random. F is used directly as the gradient estimate, i.e. $\delta_i = F$. The algorithm applying goal programming to estimate the gradient is denoted here as MO-EGS-GP.

3) *The performance indicator approach*: The use of performance indicators as the fitness function is a very recent development and Fleischer [5] is probably the first to suggest that MO performance indicators can be used to guide the evolutionary process. Indicators such as hypervolume [4] and binary indicators [20] have been applied to guide the optimization process. While no clear guidelines on the choice of metrics exist at this time, it is clear that the selected measure must be able to provide an indication of solution quality in the aspects of diversity and convergence in order to exert the selection pressure.

In this work, the hypervolume metric is applied to evaluate the volume of objective space dominated by a solution. It should be noted that we have effectively recasted the MO problem as a SO problem that maximizes the hypervolume

covered by the solutions and the fitness of the trial solution \vec{f}'_i is given as

$$F_i = \cup_{j=1}^M (r_j - f'_{i,j}) \quad (8)$$

where r_j is the reference point of the j -th objective. Since the relative quality of the trial solution to the parent is the only information required, the contribution of other solutions in the population is not considered to reduce the computational effort required to compute the hypervolume. The reference point r_j is defined in terms of \vec{f} ,

$$r_j = f_j + 1. \quad (9)$$

As in the case of the goal programming approach, F is used directly as the gradient estimate. The algorithm applying performance indicator for estimating gradient is denoted here as MO-EGS-PI.

IV. EXPERIMENTAL STUDIES

This section starts with the description of the four MO test problems used in this work. Then three performance metrics are introduced and defined in Section IV-B. In Section IV-C, a performance comparison between the different MO-EGS implementations and NSGAI1 will be made.

A. MO Test Problems

Four benchmark problems, FON, KUR, DTLZ2 and DTLZ3, are used to examine the effectiveness of the different MO-EGS implementations. The test problems of FON and KUR are selected to challenge the ability of the MO-EGS in handling nonconvexity and complex Pareto front shapes, while DTLZ2 and DTLZ3 are formulated as a five-objective optimization problem to investigate the scalability of the algorithms. The definition of these test functions is summarized in Table III.

B. Performance Metrics

Many comparative studies [9], [17], [19] made use of a suite of unary performance metrics [21] pertinent to the MO optimization goals of proximity, diversity and distribution. Three metrics including two unary metrics and a n -ary domination ratio are applied in this work.

1) *Proximity Indicator*: The metric of generational distance (GD) gives a good indication of the gap between the PF* and the evolved PF. Mathematically, the metric is a function of individual distance given as,

$$GD = \frac{1}{n_{PF}} \cdot \left(n_{PF} \sum_{i=1}^{n_{PF}} d_i^2 \right)^{\frac{1}{2}} \quad (10)$$

where $n_{PF} = |PF|$, d_i is the Euclidean distance (in objective space) between the i -th member of PF and the nearest member of PF*. A low value of GD is desirable, which reflects a small deviation between the evolved and the true Pareto front.

TABLE I
DEFINITION OF STATIC TEST FUNCTIONS

Test function	Definition
1 FON	$f_1(x_1, \dots, x_8) = 1 - \exp[-\sum_{i=1}^8 (x_i - \frac{1}{\sqrt{8}})^2],$ $f_2(x_1, \dots, x_8) = 1 + \exp[-\sum_{i=1}^8 (x_i - \frac{1}{\sqrt{8}})^2],$ where $-2 \leq x_i < 2, \forall i = 1, 2, \dots, 8$
2 KUR	$f_1(x_2, x_3) = \sum_{i=1}^2 [-10 \exp(-0.2 \sqrt{x_i^2 + x_{i+1}^2})],$ $f_1(x_2, x_3) = \sum_{i=1}^3 [x_i ^{0.8} + 5 \cdot \sin(x_i^3)]$ $x_i \in [-5, 5]$
3 DTLZ2	$f_1(\vec{x}) = (1 + g(\vec{x}_M)) \cdot \cos(0.5\pi x_1) \cdots \cos(0.5\pi x_{M-1}),$ $f_2(\vec{x}) = (1 + g(\vec{x}_M)) \cdot \cos(0.5\pi x_1) \cdots \sin(0.5\pi x_{M-1})$ \vdots $f_M(\vec{x}) = (1 + g(\vec{x}_M)) \cdot \sin(0.5\pi x_1)$ $g(\vec{x}_M) = \sum_{x_i \in \vec{x}_M} (x_i - 0.5)^2$ where $M = 5, \vec{x}_M = \{x_M, \dots, x_{M+9}\}, x_i \in [0, 1]$
4 DTLZ3	$f_1(\vec{x}) = (1 + g(\vec{x}_M)) \cdot \cos(0.5\pi x_1) \cdots \cos(0.5\pi x_{M-1}),$ $f_2(\vec{x}) = (1 + g(\vec{x}_M)) \cdot \cos(0.5\pi x_1) \cdots \sin(0.5\pi x_{M-1})$ \vdots $f_M(\vec{x}) = (1 + g(\vec{x}_M)) \cdot \sin(0.5\pi x_1)$ $g(\vec{x}_M) = 100 \left\{ \vec{x}_M + \sum_{x_i \in \vec{x}_M} (x_i - 0.5)^2 - \cos(20\pi(x_i - 0.5)) \right\}$ where $M = 5, \vec{x}_M = \{x_M, \dots, x_{M+9}\}, x_i \in [0, 1]$

2) *Diversity Indicator*: A modified maximum spread (MS')[6] is applied to measure how well the PF* is covered by the PF. Specifically, the modified metric takes into account the proximity to PF*, e.g., a higher value of MS' reflects that a larger area of the PF* is covered by the PF. The metric is given as,

$$MS' = \sqrt{\frac{1}{M} \sum_{i=1}^M \left[\frac{\min[\overline{PF}_i, \overline{PF}_i^*] - \max[\underline{PF}_i, \underline{PF}_i^*]}{\overline{PF}_i^* - \underline{PF}_i^*} \right]^2} \quad (11)$$

where \overline{PF}_i and \underline{PF}_i is the maximum and minimum of the i -th objective in PF respectively; \overline{PF}_i^* and \underline{PF}_i^* is the maximum and minimum of the i -th objective in PF*, respectively.

Pareto Dominance Indicator: An n -ary Pareto dominance indicator is proposed in [16] to measure relative solution set quality among n solution sets in the Pareto dominance sense. Considering the different PF, A_1, A_2, \dots, A_n evolved by n algorithms, this metric measures the ratio of nondominated solutions that is contributed by a particular solution set A_i to the nondominated solution set provided by all solution sets. Mathematically, the nondominance ratio (NR) is given by,

$$NR(A_1, A_2, \dots, A_n) = \frac{|B \cap A_1|}{|B|} \quad (12)$$

$$B = \{b_i | \forall b_i \exists a_j \in (A_1 \cup A_2 \dots \cup A_n) \prec b_i\}$$

where A_1 is the solution set under evaluation.

C. Comparative Study

A comparative study including NSGAI is carried out based upon the four benchmark problems listed in Table III to examine the effectiveness of the different MO-EGS implementations. The simulations are implemented in C++ on an Intel

TABLE II
INDEX OF THE DIFFERENT ALGORITHMS

Index	1	2	3	4
Algorithm	MO-EGS-RW	MO-EGS-GP	MO-EGS-PI	NSGAI

Pentium 4 2.8 GHz personal computer. Thirty independent runs are performed for each of the test functions to obtain the statistical information, such as consistency and robustness of the algorithms. For all experiments, a stopping criteria of 20,000 evaluations is used for both NSGAI and MO-EGS. The indices of the five algorithms are listed in Table V.

1) *FON*: FON is a nonconvex problem and it is well-known that aggregation-based MOEAs have problems finding a diverse pareto front for this class of problems. The PFs obtained from the different algorithms using the same random seed are showed in Fig. 1(a)-(d), while the box plots representing the distribution of the different performance metrics are shown in Fig. 2(a)-(c). Fig. 1 shows clearly that the different MO-EGS implementations are capable of finding a uniformly distributed and diverse Pareto front as compared to NSGAI. The advantage of exploiting gradient information in MO-EGS is also clear in Fig. 2 where it is observed that MO-EGS outperforms NSGAI in the aspects of GD, MS and NR.

2) *KUR*: The KUR is characterized by an PF* that is non-convex and disconnected, which contains three distinct and disconnected regions on the final tradeoff. The PFs obtained from the different algorithms are showed in Fig. 3(a)-(d), while the box plots representing the distribution of the different performance metrics are shown in Fig. 4(a)-(c). The main difficulty stemming from the high parameter interactions in this problem is the finding of all the four disconnected regions of PF*. Although NSGAI is capable of evolving a diverse PF,

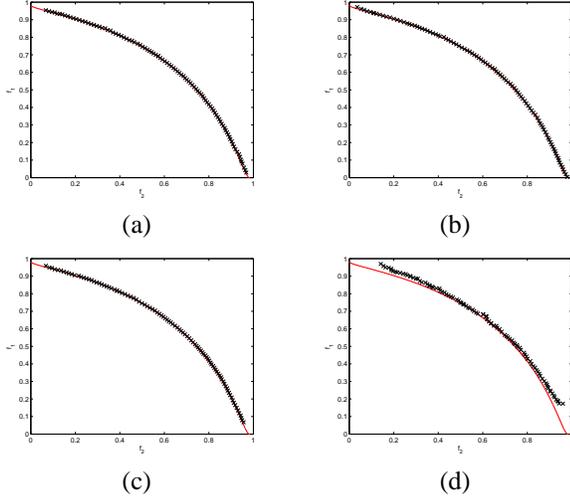


Fig. 1. The evolved Pareto front from (a) MO-EGS-RW, (b) MO-EGS-GP, (c) MO-EGS-PI and (d) NSGAI for FON.

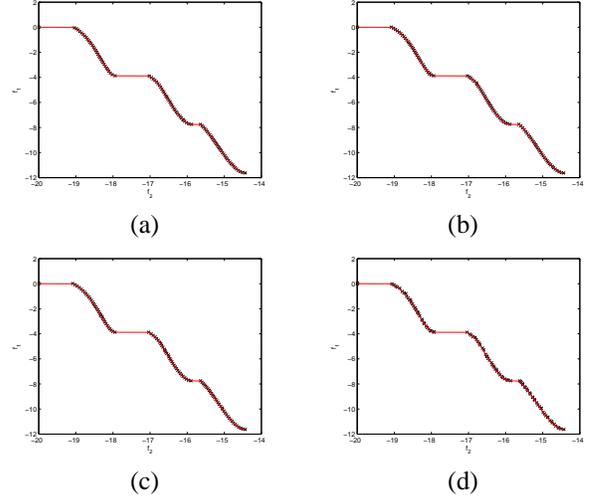


Fig. 3. The evolved Pareto front from (a) MO-EGS-RW, (b) MO-EGS-GP, (c) MO-EGS-PI and (d) NSGAI for KUR.

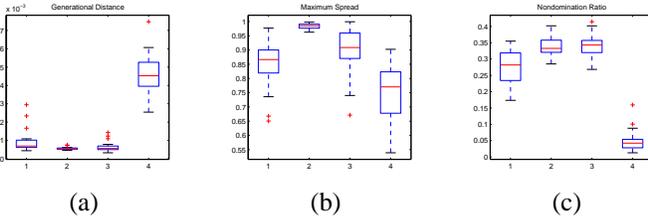


Fig. 2. Performance metric of (a) GD, (b) MS, and (c) NR for FON.

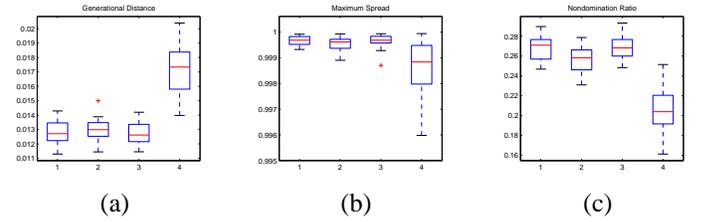


Fig. 4. Performance metric of (a) GD, (b) MS, and (c) NR for KUR.

it faced difficulty in finding it consistently. On the other hand, the various implementation of MO-EGS behaves similarly with each other and are able to find a near-optimal, diverse and uniformly distributed Pareto front.

3) *DTLZ2 and DTLZ3*: DTLZ2 and DTLZ3 is used to challenge the algorithm's capability to produce adequate pressure in driving individuals towards the high-dimensional PF*. Moreover, the DTLZ3 is also characterized by the challenge of multi-modality. The box plots representing the distribution of the different performance metrics for DTLZ2 and DTLZ3 are shown in Fig. 5 and Fig. 6, respectively. From the results, it is observed that MO-EGS scales better with increasing objectives as compared to NSGAI in all aspects of GD, MS, and NR. This is probably because the use of gradient information to drive the optimization process produces a larger selection pressure as compared to the nondominance ranking adopted in NSGAI.

D. Further Analysis

This section examines the effects of the oscillating σ_t , population sizing and the number of trial solutions on the optimization process. The problem of FON is used in this section since it is known to pose difficulties to aggregation-based MOEAs. As before, 30 simulation runs are conducted for each experimental setup.

1) *Effect of Oscillating σ_t* : The performance of the various MO-EGS implementations with and without the feature of oscillating σ_t is tabulated in Table. III. From the table, it is

clear that this feature allow the MO-EGS to find a near-optimal and more diversified Pareto front. This is particularly in the case for MO-EGS-GP.

2) *Effect of population sizing σ_t* : Simulations are conducted for population sizes of $|P| \in \{1, 5, 20, 50\}$ and the results are tabulated in Table. IV. The stopping criterion is 20,000, implying a tradeoff between the number of iterations and the number of possible search directions. From the table, it is observed that increasing $|P|$ initially improves the MS at the expense of GD. This is expected since solution improvement is traded off for exploration.

3) *Effect of N* : Simulations are conducted for $N \in \{1, 5, 20, 50\}$ to examine the impact of the number of trials on the gradient estimation. Once again, there is a tradeoff between the effort to achieve better gradient estimation and the computational effort to improve the solutions. The results are tabulated in Table. V, and it can be seen that increasing the number of trials has the same effect as increasing population

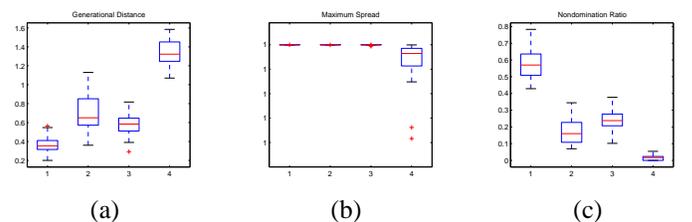


Fig. 5. Performance metric of (a) GD, (b) MS, and (c) NR for DTLZ2.

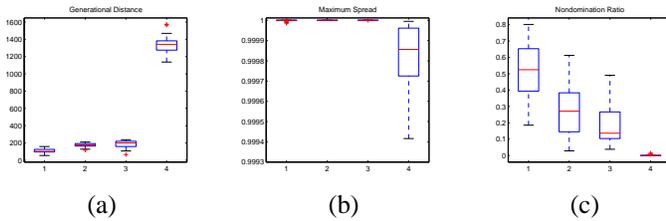


Fig. 6. Performance metric of (a) GD, (b) MS, and (c) NR for DTLZ2.

TABLE III

PERFORMANCE OF MO-EGS FOR FON WITH/WITHOUT OSCILLATING σ_t .

		normal	Oscillating
GD	MO-EGS-RW	0.0018	0.0009
	MO-EGS-GP	0.3201	0.0006
	MO-EGS-PI	0.0013	0.0007
MS	MO-EGS-RW	0.8528	0.9189
	MO-EGS-GP	0.0497	0.9669
	MO-EGS-PI	0.8982	0.9399

sizes.

V. CONCLUSION

In this paper, we present an EGS for multi-objective optimization. Unlike canonical EGS, the proposed MO-EGS incorporates an archive, an elitist scheme that guides the optimization process to sample the entire Pareto front, and a new approach for updating the mutation step size. We also consider three different fitness assignment schemes namely, 1) random weights aggregation, 2) goal programming, and 3) performance indicator. The effectiveness of the various MO-EGS implementation on nonconvex, discontinuous and high dimensional problem is validated and compared against NSGAII. The effects of population sizing, number of trial solutions and the new step-size updating scheme are also investigated.

TABLE IV

PERFORMANCE FOR FON WITH DIFFERENT POPULATION SIZING.

		1	5	20	50
GD	MO-EGS-RW	0.0007	0.0008	0.0011	0.0017
	MO-EGS-GP	0.0005	0.0006	0.0006	0.0009
	MO-EGS-PI	0.0006	0.0007	0.0007	0.0010
MS	MO-EGS-RW	0.8663	0.9147	0.9411	0.9378
	MO-EGS-GP	0.9852	0.9853	0.9818	0.9785
	MO-EGS-PI	0.9087	0.9467	0.9701	0.9762

TABLE V

PERFORMANCE FOR FON WITH DIFFERENT TRIAL SIZING.

		1	5	20	50
GD	MO-EGS-RW	0.0004	0.0006	0.0016	0.0045
	MO-EGS-GP	0.0005	0.0005	0.0008	0.0015
	MO-EGS-PI	0.0005	0.0006	0.0011	0.0023
MS	MO-EGS-RW	0.9383	0.9265	0.8975	0.8206
	MO-EGS-GP	0.9407	0.9700	0.9530	0.9326
	MO-EGS-PI	0.9388	0.9435	0.9261	0.8959

REFERENCES

- [1] D. V. Arnold and R. Salomon, "Evolutionary Gradient Search Revisited," *IEEE Transactions on Evolutionary Computation*, vol. 11, no. 4, pp. 480-495, 2007.
- [2] D. W. Corne, J. D. Knowles, and M. J. Oates, "The Pareto Envelope-based Selection Algorithm for Multiobjective Optimization," in *Proceedings of the Sixth International Conference on Parallel Problem Solving from Nature*, pp. 839-848, 2000.
- [3] K. Deb, S. Agrawal, A. Pratap, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182-197, 2002.
- [4] M. Emmerich, N. Beume, and B. Naujoks, "An EMO Algorithm Using the Hypervolume Measure as Selection Criterion," in *Proceedings of the Third Conference on Evolutionary Multi-Criterion Optimization*, pp. 62-76, 2005.
- [5] M. Fleischer, "The Measure of Pareto Optima. Applications to Multi-objective Metaheuristics," in *Proceedings of the Second International Conference on Evolutionary Multi-Criterion Optimization*, vol. 2632, pp. 519-533, 2003.
- [6] C. K. Goh and K. C. Tan, "An investigation on noisy environments in evolutionary multiobjective optimization," *IEEE Transactions on Evolutionary Computation*, in press.
- [7] H. Ishibuchi, T. Yoshida, and T. Murata, "Balance between Genetic Search and Local Search in Memetic Algorithms for Multiobjective Permutation Flowshop," *IEEE Transactions on Evolutionary Computation*, vol. 7, no. 2, pp. 204-223, 2003.
- [8] A. Jaszkiewicz, "On the Performance of Multiple-Objective Genetic Local Search on the 0/1 Knapsack Problem-A Comparative Experiment," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 4, pp. 402-412, 2002.
- [9] A. Jaszkiewicz, "Do multi-objective metaheuristics deliver on their promises? A computational experiment on the set-covering problem," *IEEE Transactions on Evolutionary Computation*, vol. 7, no. 2, pp. 133-143, 2003.
- [10] Y. Jin, T. Okabe and B. Sendhoff, "Adapting Weighted Aggregation for Multiobjective Evolution Strategies," in *Proceedings of the First Conference on Evolutionary Multi-Criterion Optimization*, pp. 96-110, 2001.
- [11] J. D. Knowles and D. W. Corne, "M-PAES: A Memetic algorithm for multi-objective optimization," in *Proceedings of the 2000 Congress on Evolutionary Computation*, pp. 325-332, 2000.
- [12] T. Murata and H. Ishibuchi, "MOGA: Multi-objective genetic algorithms," in *Proceedings of the 1995 Congress on Evolutionary Computation*, pp. 289-294, 1995.
- [13] R. Solomon, "Evolutionary Algorithms and Gradient Search: Similarities and Differences," *IEEE Transactions on Evolutionary Computation*, vol. 2, no. 2, pp. 45-55, 1998.
- [14] N. Srinivas, and K. Deb, "Multiobjective optimization using non-dominated sorting in genetic algorithms," *Evolutionary Computation*, vol. 2, no. 3, pp. 221-248, 1994.
- [15] P. K. Shukla, "On gradient based local search methods in unconstrained evolutionary multi-objective optimization," in *Proceedings of the fourth International Conference on Evolutionary Multi-Criterion Optimization*, pp. 96-110, 2007.
- [16] K. C. Tan and C. K. Goh, "A Competitive-Cooperation Coevolutionary Paradigm for Dynamic Multi-objective Optimization," *IEEE Transactions on Evolutionary Computation*, accepted.
- [17] K. C. Tan, Y. J. Yang, and C. K. Goh, "A distributed cooperative coevolutionary algorithm for multiobjective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 10, no. 5, pp. 527- 549, 2006.
- [18] K. C. Tan, C. K. Goh, Y. J. Yang, and T. H. Lee, "Evolving better population distribution and exploration in evolutionary multi-objective optimization," *European Journal of Operational Research*, vol. 171, no. 2, pp. 463-495, 2006.
- [19] D. A. Van. Veldhuizen and G. B. Lamont, "On measuring multiobjective evolutionary algorithm performance," in *Proceedings of the 2000 Congress on Evolutionary Computation*, vol. 1, pp. 204-211, 2000.
- [20] E. Zitzler and S. Kunzli, "Indicator-Based Selection in Multiobjective Search," in *Proceedings of the Eighth International Conference on Parallel Problem Solving from Nature*, pp. 832-842, 2004.
- [21] E. Zitzler, L. Thiele, M. Laumanns, C. M. Fonseca and V. G. Fonseca, "Performance assessment of multiobjective optimizers: An analysis and review," *IEEE Transactions on Evolutionary Computation*, vol. 7, no. 2, pp. 117-132, 2003.