

Two-Level of Nondominated Solutions Approach to Multiobjective Particle Swarm Optimization

M. A. Abido

Electrical Engineering Department
King Fahd University of Petroleum and Minerals
Dhahran 31261, Saudi Arabia

mabido@kfupm.edu.sa

ABSTRACT

In multiobjective particle swarm optimization (MOPSO) methods, selecting the local best and the global best for each particle of the population has a great impact on the convergence and diversity of solutions, especially when optimizing problems with high number of objectives. This paper presents a two-level of nondominated solutions approach to MOPSO. The ability of the proposed approach to detect the true Pareto optimal solutions and capture the shape of the Pareto front is evaluated through experiments on well-known non-trivial test problems. The diversity of the nondominated solutions obtained is demonstrated through different measures. The proposed approach has been assessed through a comparative study with the reported results in the literature.

Categories and Subject Descriptors

I.2.8 [Artificial Intelligence]: Problem Solving, Control Methods, and Search – *heuristic methods*.

General Terms

Algorithms, Experimentation, Performance, Verification.

Keywords

Particle Swarm Optimization, multiobjective optimization, Pareto-optimal set, nondominated solutions.

1. INTRODUCTION

Evolutionary Algorithms (EA) are good candidate to multiobjective optimization problems due to their abilities, to search simultaneously for multiple Pareto optimal solutions and, perform better global search of the search space [1]. The Particle Swarm Optimization (PSO) is a swarm intelligence method that models social behavior to guide swarms of particles towards the most promising regions of the search space [2-5]. Generally, PSO is characterized as simple in concept, easy to implement, and computationally efficient. Unlike the other heuristic techniques, PSO has a flexible and well-balanced mechanism to enhance and

adapt the global and local exploration abilities. It usually results in faster convergence rates than the Genetic Algorithms [6]. The PSO has been applied to different single objective optimization problems with impressive success [7-11]. Although PSO's performance, in single-objective optimization tasks, has been extensively studied, a little work has been done for multiobjective optimization problems thus far.

Recently, investigators are paying more and more interest on PSO to solve multi-objective problems as discussed in the following section. Changing a PSO to a multi-objective PSO (MOPSO) requires redefinition of global and local best individuals in order to obtain a front of optimal solutions in MOPSO. In multiobjective particle swarm optimization, there is no absolute global best, but rather a set of nondominated solutions. In addition, there may be no single local best individual for each particle of the swarm. Choosing the global best and local best to guide the swarm particles becomes nontrivial task in multiobjective domain. Some attempts have been done in the literature to select the best guides for the particles in MOPSO.

Parsopoulos and Vrahatis [12] presented a first study of the performance of the PSO in multiobjective optimization problems. The performance of the PSO in terms of finding Pareto front in weighted aggregation cases was presented. A vector evaluated PSO (VEPSO) based on the concept of the vector evaluated genetic algorithm (VEGA) [13] was proposed and examined to perform multiobjective optimization. However, selection of individuals that excel in one objective without looking to the other objectives implies a problem of killing the middling performance individuals that can be very useful for compromise solutions [14].

Hu and Eberhart [15] presented a MOPSO that uses dynamic neighborhood strategy to obtain the best local guide for each particle in biobjective optimization problems. However, selecting the best local guides based on one of the objectives degrades the algorithm performance as one-dimensional optimization is used to deal with multiple objectives. In addition, selecting the fixed objective needs a priori knowledge about the objective functions. Furthermore, choosing the number of neighbors considered, the objective function to optimize and the one to be fixed, and extending the algorithm to higher dimensional objective spaces are very much involved questions.

Coello and Lechuga [16] proposed a MOPSO where the objective space is divided to hypercubes before selecting the best local guide for each particle in the population. However, this method biases the selection toward under-represented areas of the

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

GECCO '07, July 7–11, 2007, London, England, United Kingdom.
Copyright 2007 ACM 978-1-59593-697-4/07/0007...\$5.00.

estimated Pareto front and only one local best solution is maintained for each particle. In addition, the random selection of the best local guide affects its quality. However, The proposed method is compared with the Pareto Archived Evolutionary Strategy (PAES) and Nondominated Sorting Genetic Algorithms II (NSGA-II) with promising results.

Fieldsend and Singh [17] proposed a MOPSO which uses a dominated tree for the choice of the best local guide for each particle in the population. The presented method has been tested on four biobjective test problems with promising results in three test problems compared with PAES and a competing MOPSO presented in [16]. However, the performance of the presented MOPSO in [16] and in [17] was very poor in the multifrontal fourth test problem with multimodality [18]. In addition, the algorithm has not been validated on higher dimensional objective spaces and might lose the most suitable solutions.

Mostaghim and Teich [19] proposed a sigma method in which the best local guides for each particle are adopted to improve the convergence and diversity of a PSO approach used for multiobjective optimization. They also use a "turbulence" operator, but applied on decision variable space. The use of the sigma values increases the selection pressure of PSO which was already high. This may cause premature convergence in some cases e.g., in multifrontal problems. Comparisons with the strength Pareto evolutionary algorithm 2 (SPEA2) [20] and the dominated trees of [17] using four test problems and the coverage metric [21].

Hu *et al.* [22] adopted a secondary population called extended memory and introduced some further improvements to their dynamic neighborhood PSO approach presented in [15]. Nevertheless, it is worth indicating that this approach completely fails in generating the true Pareto front of some problems. In addition, the presented algorithm has been compared to the SPEA [23] using the set coverage metric [21].

Li [24] proposed an approach in which the main mechanisms of the NSGA-II [25] are adopted in a PSO algorithm. The proposed approach showed a very competitive performance with respect to the NSGA-II even outperforming it in some cases.

Lu [26] presented the dynamic population strategy assisted PSO. This method can evolve to an approximately optimal population size while the population is approaching the true Pareto front. However, this algorithm suffers from some difficulties in finding a well-approximated Pareto optimal front [27].

Mostaghim and Teich [28] proposed a new method uses the property of moving particles in MOPSO and divides the population of the covering MOPSO into subswarms. The subswarms try to cover the gaps between the nondominated solutions found in the initial run. The proposed covering method is tested on different test problems.

In most recent work in the field of MOPSO, different approaches have been introduced to identify the local best solution based on memorizing all the nondominated solutions visited by a particle and selecting the local best among them [29]. These approaches have been implemented on some standard test problems where the results show that keeping the particle archive improves significantly the effectiveness of the technique. However, keeping all the nondominated solutions visited by a particle has a

drawback of increasing drastically the computational burden. On the other hand, a comprehensive survey of the state-of-the-art in multiobjective particle swarm optimizers can be found in [30] where different techniques reported in MOPSO development have been categorized and discussed.

In this paper, a new MOPSO technique based on two-level of nondominated solutions is proposed. The ability of the proposed approach to detect the true Pareto optimal solutions and capture the shape of the Pareto front is studied through experiments on well-known non-trivial test problems.

2. MULTIOBJECTIVE OPTIMIZATION

Many real-world problems involve simultaneous optimization of several objective functions. Generally, these functions are non-commensurable and often conflicting objectives. Multiobjective optimization with such conflicting objective functions gives rise to a set of optimal solutions, instead of one optimal solution. The reason for the optimality of many solutions is that no one can be considered to be better than any other with respect to all objective functions. These optimal solutions are known as *Pareto-optimal* solutions.

A general multiobjective optimization problem consists of a number of objectives to be optimized simultaneously and is associated with a number of equality and inequality constraints. It can be formulated as follows:

$$\text{Minimize } f_i(x) \quad i = 1, \dots, N_{obj}, \quad (1)$$

$$\text{Subject to : } \begin{cases} g_j(x) = 0 & j = 1, \dots, M, \\ h_k(x) \leq 0 & k = 1, \dots, K, \end{cases} \quad (2)$$

where f_i is the i^{th} objective function, x is a decision vector that represents a solution, and N_{obj} is the number of objectives.

For a multiobjective optimization problem, any two solutions x_1 and x_2 can have one of two possibilities- one dominates the other or none dominates the other. In a minimization problem, without loss of generality, a solution x_1 dominates x_2 *iff* the following two conditions are satisfied:

$$\forall i \in \{1, 2, \dots, N_{obj}\} : f_i(x_1) \leq f_i(x_2), \quad (3)$$

$$\exists j \in \{1, 2, \dots, N_{obj}\} : f_j(x_1) < f_j(x_2). \quad (4)$$

If any of the above conditions is violated, the solution x_1 does not dominate the solution x_2 . If x_1 dominates the solution x_2 , x_1 is called the nondominated solution within the set $\{x_1, x_2\}$. The solutions that are nondominated within the entire search space are denoted as *Pareto-optimal* and constitute the *Pareto-optimal set* or *Pareto-optimal front*.

3. PROPOSED APPROACH TO MOPSO

3.1 Overview

In multiobjective particle swarm optimization, a set of nondominated solutions must replace the single global best

individual in the standard single objective PSO case. In addition, there may be no single local best individual for each particle of the swarm. Choosing the global best and local best to guide the swarm particles becomes nontrivial task in multiobjective domain. This paper presents two-level of nondominated solutions approach to address these problems. In the proposed approach, elitism is also considered by copying any nondominated solution obtained to an external set in order to keep the new nondominated solutions obtained during generations. The external set is updated regularly to hold only the nondominated solutions. The basic definitions and the major steps of the proposed approach can be explained as follows.

3.2 Proposed MOPSO Algorithm

The major elements of the proposed MOPSO technique are briefly defined as follows: -

Nondominated local set, $S_j^*(t)$: It is a set that stores the nondominated solutions obtained by the j^{th} particle up to the current time. As the j^{th} particle moves through the search space, its new position is added to this set and the set is updated to keep only the nondominated solutions. An average linkage based hierarchical clustering algorithm [31] used by SPEA [32] is employed to reduce the nondominated local set size if it exceeds a certain prespecified value.

Nondominated global set, $S^{}(t)$:** It is a set that stores the nondominated solutions obtained by all particle up to the current time. First, the union of all nondominated local sets is formed. Then, the nondominated solutions out of this union are members in the nondominated global set. An average linkage based hierarchical clustering algorithm is employed to reduce the nondominated global set to a manageable size.

Local best, $X_j^*(t)$, and Global best, $X_j^{}(t)$:** The individual distances between members in nondominated local set of the j^{th} particle, $S_j^*(t)$, and members in nondominated global set, $S^{**}(t)$, are measured in the objective space. If $X_j^*(t)$ and $X_j^{**}(t)$ are the members of $S_j^*(t)$ and $S^{**}(t)$ respectively that give the minimum distance, they are selected as the local best and the global best of the j^{th} particle respectively.

In the proposed MOPSO algorithm, the population has n particles and each particle is an m -dimensional vector, where m is the number of optimized parameters. The computational flow of the proposed MOPSO technique can be described in the following steps.

Step 1 (Initialization): Set the time counter $t=0$ and generate randomly n particles, $\{X_j(0), j=1, \dots, n\}$, where $X_j(0)=[x_{j,1}(0), \dots, x_{j,m}(0)]$. $x_{j,k}(0)$ is generated by randomly selecting a value with uniform probability over the k^{th} optimized parameter search space $[x_k^{\min}, x_k^{\max}]$. Similarly, generate randomly initial velocities of all particles, $\{V_j(0), j=1, \dots, n\}$, where $V_j(0)=[v_{j,1}(0), \dots, v_{j,m}(0)]$. $v_{j,k}(0)$ is generated by randomly selecting a value with uniform probability over the k^{th} dimension $[-v_k^{\max}, v_k^{\max}]$ where the particle velocity in the k^{th} dimension is limited by some maximum value, v_k^{\max} . This limit enhances the local exploration of the problem space and it realistically simulates the incremental changes of human learning. To ensure uniform velocity through all dimensions, the maximum velocity in the k^{th} dimension is proposed as:

$$v_k^{\max} = (x_k^{\max} - x_k^{\min}) / N \quad (5)$$

where N is a selected number of intervals.

Each particle in the initial population is evaluated using the objective functions. For each particle, set $S_j^*(0)=\{X_j(0)\}$ and the local best $X_j^*(0)=X_j(0)$, $j=1, \dots, n$. Search for the nondominated solutions and form the nondominated global set $S^{**}(0)$. The nearest member in $S^{**}(0)$ to $X_j^*(0)$ is selected as the global best $X_j^{**}(0)$ of the j^{th} particle. Set the external set equal to $S^{**}(0)$. Set the initial value of the inertia weight $w(0)$.

Step 2 (Time updating): Update the time counter $t=t+1$.

Step 3 (Weight updating): Update the inertia weight.

Step 4 (Velocity updating): Using the local best $X_j^*(t)$ and the global best $X_j^{**}(t)$ of each particle, $j=1, \dots, n$, the j^{th} particle velocity in the k^{th} dimension is updated according to the following equation:

$$v_{j,k}(t) = w(t) v_{j,k}(t-1) + c_1 r_1 (x_{j,k}^*(t-1) - x_{j,k}(t-1)) + c_2 r_2 (x_{j,k}^{**}(t-1) - x_{j,k}(t-1)) \quad (6)$$

where c_1 and c_2 are positive constants and r_1 and r_2 are uniformly distributed random numbers in $[0,1]$. If a particle violates the velocity limits, set its velocity equal to the proper limit.

Step 5 (Position updating): Based on the updated velocities, each particle changes its position according to the following equation.

$$x_{j,k}(t) = v_{j,k}(t) + x_{j,k}(t-1) \quad (7)$$

If a particle violates its position limits in any dimension, set its position at the proper limit.

Step 6 (Nondominated local set updating): The updated position of the j^{th} particle is added to $S_j^*(t)$. The dominated solutions in $S_j^*(t)$ will be truncated and the set will be updated accordingly. If the size of $S_j^*(t)$ exceeds a prespecified value, the hierarchical clustering algorithm will be invoked to reduce the size to its maximum limit.

Step 7 (Nondominated global set updating): The union of all nondominated local sets is formed and the nondominated solutions out of this union are members in the nondominated global set $S^{**}(t)$. The size of this set will be reduced by hierarchical clustering algorithm if it exceeds a prespecified value.

Step 8 (External set updating): The external Pareto-optimal set is updated as follows.

Copy the members of $S^{**}(t)$ to the external Pareto set.

Search the external Pareto set for the nondominated individuals and remove all dominated solutions from the set.

If the number of the individuals externally stored in the Pareto set exceeds the maximum size, reduce the set by means of clustering.

Step 9 (Local best and global best updating): The individual distances between members in $S_j^*(t)$, and members in $S^{**}(t)$, are measured in the objective space. If $X_j^*(t)$ and $X_j^{**}(t)$ are the members of $S_j^*(t)$ and $S^{**}(t)$ respectively that give the minimum distance, they are selected as the local best and the global best of the j^{th} particle respectively.

Step 10 (Stopping criteria): If the number of iterations exceeds the maximum then stop, else go to step 2.

3.3 The Proposed MOPSO Implementation

The proposed MOPSO based approach was implemented using FORTRAN language and the developed software program was executed on a 1.8-GHz Pentium 4 PC. Initially, several runs have been done with different values of the PSO key parameters such as the initial inertia weight and the maximum allowable velocity. Other parameters are selected as: number of particles $n=50$, decrement constant $\alpha=0.99$, $c_1=c_2=2$, and the search will be terminated if the number of iterations reaches 1000.

To demonstrate the effectiveness of the proposed approach, different cases with various objectives are considered in this study.

4. RESULTS AND DISCUSSIONS

To compare the results and to assess the effectiveness of the proposed approach to MOPSO, a group of benchmark test problems to compare the multiobjective techniques have been examined. The test for each function has been done for 100 runs. For the first two test problems, TP1 and TP2, the SPEA [32] implemented in [33-34] has been employed for comparison purposes. However, the results of SPEA for the last two test problems, TP3 and TP4, have been downloaded from [35].

4.1 Test Problem TP1

This test problem can be defined as follows [34].

$$f_1(x) = \begin{cases} -x & \text{if } x \leq 1 \\ -2+x & \text{if } 1 < x \leq 3 \\ 4-x & \text{if } 3 < x \leq 4 \\ -4+x & \text{if } x > 4 \end{cases} \quad (8)$$

$$f_2(x) = (x-5)^2 \quad (9)$$

$$-100 \leq x \leq 100 \quad (10)$$

This test problem has a discontinuous Pareto front. The Pareto fronts of the proposed MOPSO and SPEA are shown in Fig. 1.

4.2 Test Problem TP2

This test problem can be defined as follows.

$$f_1(x_1, x_2) = 2 + (x_1 - 2)^2 + (x_2 - 1)^2 - 10(225 - x_1^2 - x_2^2) \quad (11)$$

$$f_2(x_1, x_2) = 9x_1 + (x_2 - 1)^2 - 10(3x_2 - x_1 - 10) \quad (12)$$

$$c_1 = 225 - x_1^2 - x_2^2 \quad (13)$$

$$c_2 = 3x_2 - x_1 - 10 \quad (14)$$

$$c_1 = \begin{cases} c_1 & \text{if } c_1 \leq 0 \\ 0 & \text{if } c_1 > 0 \end{cases} \quad (15)$$

$$c_2 = \begin{cases} c_2 & \text{if } c_2 \leq 0 \\ 0 & \text{if } c_2 > 0 \end{cases} \quad (16)$$

$$-20 \leq x_1, x_2 \leq 20 \quad (17)$$

The Pareto fronts of the proposed MOPSO and SPEA are shown in Fig. 2.

4.3 Test Problem TP3

This test problem can be defined as follows [18].

$$f_1(x_1) = x_1 \quad (18)$$

$$f_2(\vec{x}) = g(x_2, \dots, x_n) \cdot h(f_1, g) \quad (19)$$

$$g(x_2, \dots, x_n) = 1 + 9 \cdot \left(\sum_{i=2}^n x_i \right) / (n-1) \quad (20)$$

$$h(f_1, g) = 1 - \sqrt{\frac{f_1}{g}} \quad (21)$$

$$\vec{x} = (x_1, x_2, \dots, x_n) \quad (22)$$

$$x_i \in [0, 1], \quad i = 1, 2, \dots, 30 \quad (23)$$

This test problem has a convex Pareto optimal front formed with

$$g(x_2, \dots, x_n) = 1 \quad (24)$$

The Pareto fronts of the proposed MOPSO and SPEA are shown in Fig. 3.

4.4 Test Problem TP4

This test problem can be defined as follows [18].

$$f_1(x_1) = x_1 \quad (25)$$

$$f_2(\vec{x}) = g(x_2, \dots, x_n) \cdot h(f_1, g) \quad (26)$$

$$g(x_2, \dots, x_n) = 1 + 9 \cdot \left(\sum_{i=2}^n x_i \right) / (n-1) \quad (27)$$

$$h(f_1, g) = 1 - \sqrt{\frac{f_1}{g}} - \frac{f_1}{g} \sin(10\pi f_1) \quad (28)$$

$$\vec{x} = (x_1, x_2, \dots, x_n) \quad (29)$$

$$x_i \in [0, 1], \quad i = 1, 2, \dots, 30 \quad (30)$$

This test problem represents the discreteness feature. Its Pareto optimal front consists of several non-contiguous formed with

$$g(x_2, \dots, x_n) = 1 \quad (31)$$

The Pareto fronts of the proposed MOPSO and SPEA are shown in Fig. 4.

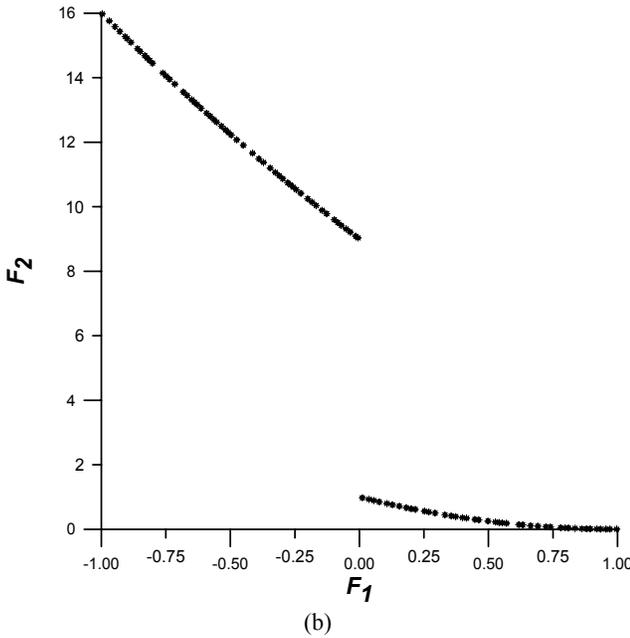
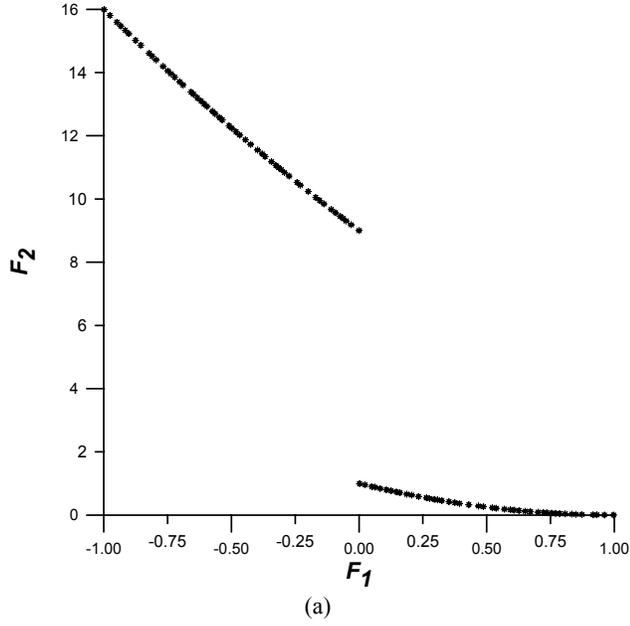


Figure 1: Pareto front of test problem 1 produced by
(a) Proposed MOPSO (b) SPEA

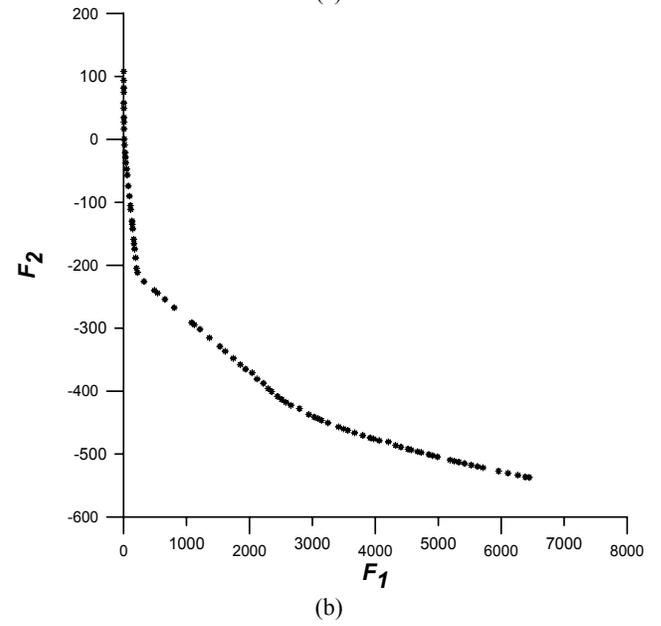
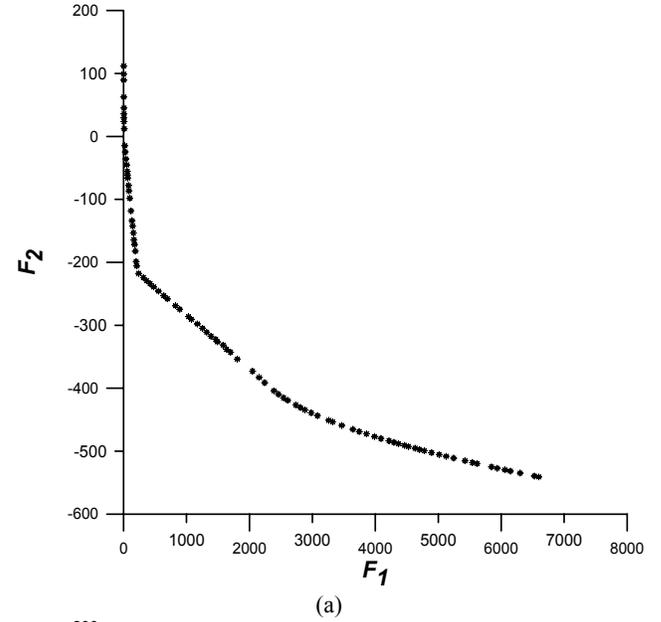


Figure 3: Pareto front of test problem 2 produced by
(a) Proposed MOPSO (b) SPEA

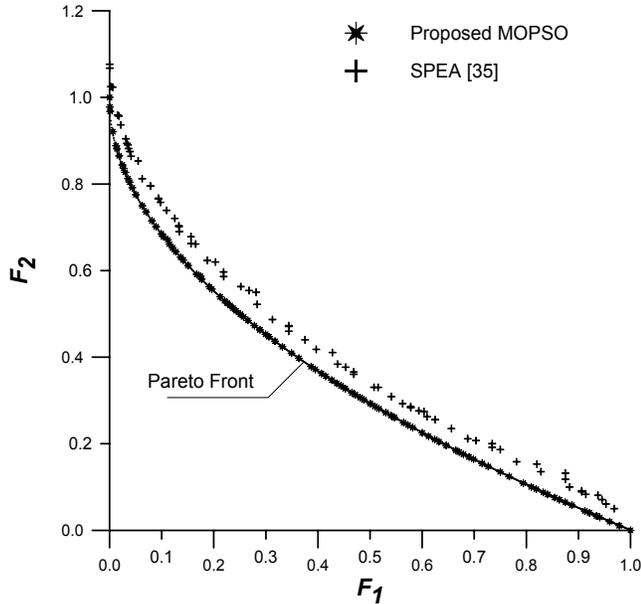


Figure 3: Pareto front of test problem 3

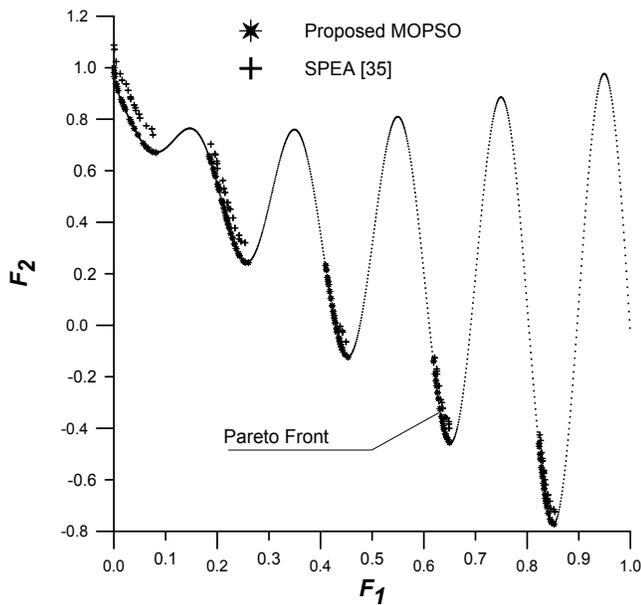


Figure 4: Pareto front of test problem 4

It can be shown from Figs. 1, 2, 3, and 4 that the proposed MOPSO has satisfactory diversity and distribution characteristics for these test problems. The superiority of the proposed MOPSO compared to SPEA is more pronounced in test problems TP3 and TP4 as the proposed MOPSO captures the true Pareto optimal front in these test problems with satisfactory diversity and distribution.

5. A COMPARATIVE STUDY

In this section, a comparative study for quality measures has been carried out to assess the proposed MOPSO. Generally, the definition of quality in the case of multiobjective optimization is more complex than for single objective optimization problems since the optimization goal itself consists of multiple objectives [18, 21, 23]. Therefore, the above results have been compiled and compared in view of the following objectives.

- The obtained nondominated set should be as close as possible to the Pareto-optimal front.
- The obtained nondominated solutions should be as diverse as possible.
- The obtained nondominated solutions should have good distribution over the nondominated front.

A performance measure of the extent of the nondominated solutions is presented in [18]. The measure estimates the range to which the fronts spread out. In other words, it measures the normalized distance of the two outer nondominated solutions. The average values of the normalized distance measure over 100 different optimization runs are given in Table 1. The results show that the proposed MOPSO has larger extent of the nondominated solutions in all test problems considered except TP1 where SPEA has similar result with respect to this measure.

On the other hand, the set coverage metric measure [23] for comparing the performance of the proposed MOPSO and SPEA has been examined in this study. The average values of this measure over 100 different optimization runs are given in Table 2. It can be shown that the performance of the proposed MOPSO is comparable with that of SPEA in TP1 as a percent of 1% of the nondominated solutions of each technique is covered by those of the other. It can be also seen from Table 2 that the proposed MOPSO has better performance in TP2. The better performance of the proposed MOPSO is more pronounced in case of test problems TP3 and TP4 where the nondominated solutions obtained by the proposed MOPSO completely cover those of SPEA and none of SPEA covers any solution of the proposed MOPSO. This confirms the fronts shown in Figs. 3 and 4.

In this study, the quality measure presented in [34] is also implemented. For each test problem, individual nondominated sets of the proposed MOPSO and SPEA are combined to form a pool. Then, the dominance conditions are applied to all solutions in the pool. The nondominated solutions are extracted from the pool to form an elite set of Pareto-optimal solutions obtained by both techniques. The average over 100 different runs for each test problem is given in Table 3. It can be observed that the proposed MOPSO has a 100% share to the elite set while SPEA has no contribution in test problems TP3 and TP4. The better performance of MOPSO is observed also in case of TP2 while a comparable performance has been experienced in TP1.

Table 1. Normalized distance measure

Test Problem	TP1	TP2	TP3	TP4
Proposed MOPSO	1.412	1.409	0.965	0.976
SPEA	1.402	1.400	0.961	0.987

Table 2. Percentage of nondominated solutions of set B covered by those in set A

<i>Set A</i>	<i>Set B</i>	<i>TP1</i>	<i>TP2</i>	<i>TP3</i>	<i>TP4</i>
<i>Proposed MOPSO</i>	<i>SPEA</i>	1.0	12.0	100.0	100.0
<i>SPEA</i>	<i>Proposed MOPSO</i>	1.0	6.0	0.0	0.0

Table 3. Average number of Pareto-optimal solutions in elite set of nondominated solutions

<i>Test Problem</i>	<i>TP1</i>	<i>TP2</i>	<i>TP3</i>	<i>TP4</i>
<i>Proposed MOPSO</i>	99	94	100	100
<i>SPEA</i>	99	88	0	0
<i>Elite Set Size</i>	198	182	100	100

6. ACKNOWLEDGMENT

The author acknowledges the support and encouragement of King Fahd University of Petroleum & Minerals through Funded Project # FT/2006-13.

7. CONCLUSIONS

A new approach to multiobjective particle swarm optimization technique is presented in this paper. The proposed approach is based on two-level, local and global, of nondominated solutions to select the local and global guides for each particle in the swarm. The capability of the proposed MOPSO to obtain the true Pareto optimal solutions and capture the shape of the Pareto front is evaluated and tested on well-known non-trivial test problems. The diversity of the nondominated solutions obtained by the proposed MOPSO is demonstrated through different measures. The proposed approach has been assessed through a comparative study with the reported results in the literature. The results show the superiority of the proposed MOPSO approach in terms of capturing the shape of the Pareto front and obtaining nondominated solutions with satisfactory diversity characteristics for the test problems considered.

8. REFERENCES

[1] E. Zitzler, *Evolutionary Algorithms for Multiobjective Optimization: Methods and Applications*, Ph.D. Thesis, Swiss Federal Institute of Technology, Zurich, 1999.

[2] J. Kennedy, "The Particle Swarm: Social Adaptation of Knowledge," *Proceedings of the 1997 IEEE international Conference on Evolutionary Computation ICEC'97*, Indianapolis, Indiana, USA, 1997, pp. 303-308.

[3] P. Angeline, "Evolutionary Optimization versus Particle Swarm Optimization: Philosophy and Performance Differences," *Proceedings of the 7th Annual Conference on Evolutionary Programming*, March 1998, pp. 601-610.

[4] Y. Shi and R. Eberhart, "Parameter Selection in Particle Swarm Optimization," *Proceedings of the 7th Annual*

Conference on Evolutionary Programming, March 1998, pp. 591-600.

[5] E. Ozcan and C. Mohan, "Analysis of a Simple Particle Swarm Optimization System," *Intelligent Engineering Systems Through Artificial Neural Networks*, Vol. 8, 1998, pp. 253-258.

[6] J. Kennedy and R. Eberhart, *Swarm Intelligence*, Morgan Kaufmann Publishers, 2001.

[7] M. A. Abido, "Optimal Design of Power System Stabilizers Using Particle Swarm Optimization," *IEEE Trans. on Energy Conversion*, Vol. 17, No. 3, September 2002, pp. 406-413.

[8] M. A. Abido, "Optimal Power Flow Using Particle Swarm Optimization" *International Journal of Electrical Power and Energy Systems*, Vol. 24, No. 7, October 2002, pp. 563-571.

[9] M. P. Wachowiak, R. Smoliková, Y. Zheng, J. M. Zurada, and A. S. Elmaghraby, "An Approach to Multimodal Biomedical Image Registration Utilizing Particle Swarm Optimization," *IEEE Transactions on Evolutionary Computation*, Vol. 8, No. 3, June 2004, pp. 289-301.

[10] M. F. Tasgetiren, M. Sevkli, Y. C. Liang, and G. Gencyilmaz, "Particle Swarm Optimization Algorithm for Permutation Flowshop Sequencing Problem," *Proceedings of the 4th International Workshop on Ant Colony Optimization and Swarm Intelligence, ANTS2004, LNCS 3172* by Springer-Verlag, Brussels, Belgium, September 5-8, 2004, pp.382-390.

[11] S. Mishra, "A Hybrid Least Square-Fuzzy Bacterial Foraging Strategy for Harmonic Estimation," *IEEE Transactions on Evolutionary Computation*, Vol. 9, No. 1, February 2005, pp. 61-73.

[12] K. E. Parsopoulos and M. N. Vrahatis, "Particle Swarm Optimization Method in Multiobjective Problems," *Proceedings of the ACM 2002 Symposium on Applied Computing (SAC'2002)*, 2002, pp. 603-607.

[13] J. D. Schaffer, *Multiobjective Optimization with Vector Evaluated Genetic Algorithms*, PhD Thesis, Vanderbilt University, Nashville, USA, 1984.

[14] C. A. C. Coello, "A Comprehensive Survey of Evolutionary-Based Multiobjective Optimization Techniques," *Knowledge and Information Systems*, Vol. 1, No. 3, 1999, pp. 269-308.

[15] X. Hu and R. Eberhart, "Multiobjective Optimization Using Dynamic Neighborhood Particle Swarm Optimization," *Congress on Evolutionary Computation (CEC'2002)*, Vol. 2, pp. 1677-1681, IEEE Service Center, Piscataway, New Jersey, May 2002.

[16] C. A. C. Coello and M. S. Lechuga, "MOPSO: A Proposal for Multiple Objective Particle Swarm Optimization," *Congress on Evolutionary Computation (CEC'2002)*, Vol. 2, pp. 1051-1056, IEEE Service Center, Piscataway, New Jersey, May 2002.

[17] J. E. Fieldsend and S. Singh, "A Multi-Objective Algorithm based upon Particle Swarm Optimization, an Efficient Data Structure and Turbulence," *Proceedings of the 2002 U.K. Workshop on Computational Intelligence*, pp. 37-44, Birmingham, UK, 2-4 September 2002.

- [18] E. Zitzler, K. Deb, and L. Thiele, "Comparison of Multiobjective Evolutionary Algorithms: Empirical Results," *Evolutionary Computation*, Vol. 8, No. 2, 2000, pp. 173-195.
- [19] S. Mostaghim and J. Teich, "Strategies for Finding Good Local Guides in Multiobjective Particle Swarm Optimization (MOPSO)," *Proceedings of 2003 IEEE Swarm Intelligence Symposium*, Indianapolis, IN, USA, April 2003, pp. 26-33.
- [20] E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: Improving the Strength Pareto Evolutionary Algorithm," *Proceedings of EUROGEN 2001*, Athens, Greece, September 2001.
- [21] E. Zitzler, and L. Thiele, "Multiobjective Optimization Using Evolutionary Algorithms – A Comparative Case Study," *Parallel Problem Solving from Nature V*, pages 292-301, Amsterdam, September 1998, Springer-Verlag.
- [22] X. Hu, R. Eberhart, and Y. Shi, "Particle Swarm with Extended Memory for Multiobjective Optimization," *Proceedings of 2003 IEEE Swarm Intelligence Symposium*, Indianapolis, IN, USA, April 2003, pp. 193-197.
- [23] E. Zitzler and L. Thiele, "Multiobjective Evolutionary Algorithms: A Comparative Case Study and the Strength Pareto Approach," *IEEE Trans. on Evolutionary Computation*, Vol. 3, No. 4, 1999, pp. 257-271.
- [24] Xiaodong Li, "A Nondominated Sorting Particle Swarm Optimizer for Multiobjective Optimization," *Lecture Notes in Computer Science, Proceedings of Genetic and Evolutionary Computation GECCO 2003*, Vol. 2723, Part I, Berlin, Germany, July 2003, pp. 37-48.
- [25] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A Fast and Elitist Multiobjective Genetic Algorithms: NSGA-II," *IEEE Trans. on Evolutionary Computation*, Vol. 6, No. 2, 2002, pp. 182-197.
- [26] H. Lu, "Dynamic Population Strategy Assisted Particle Swarm Optimization in Multiobjective Evolutionary Algorithm design," *IEEE Neural Network Society, IEEE NNS Student Research Grants 2002*, Final reports 2003.
- [27] M. P. Song and G. C. Gu, "Research on Particle Swarm Optimization: A Review," *Proceedings of the 3rd International Conference on Machine Learning and Cybernetics*, Shanghai, China, 26-29 August 2004, pp. 2236-2241.
- [28] S. Mostaghim and J. Teich, "Covering Pareto-Optimal Fronts by Subswarms in Multiobjective Particle Swarm Optimization," *Proceedings of IEEE Congress on Evolutionary Computation CEC'2004*, Portland, Oregon, USA, June 19-23, 2004, pp. 1404-1411.
- [29] Jürgen Branke and Sanaz Mostaghim, "About Selecting the Personal Best in Multi-Objective Particle Swarm Optimization," *Proceedings of the 9th International Conference on Parallel Problem Solving from Nature - PPSN IX*, Reykjavik, Iceland, September 9-13, 2006, pp. 523-532.
- [30] M. Reyes-Sierra and C. A. C. Coello, "Multi-Objective Particle Swarm Optimizers: A Survey of the State-of-the-Art," *International Journal of Computational Intelligence Research*, Vol. 2, No. 3, 2006, pp. 287-308.
- [31] J. N. Morse, "Reducing the Size of Nondominated Set: Pruning by Clustering," *Computers and Operations Research*, Vol. 7, No. 1-2, 1980, pp. 55-66.
- [32] E. Zitzler and L. Thiele, "An Evolutionary Algorithm for Multiobjective optimization: The Strength Pareto Approach," *TIK-Report*, No. 43, 1998.
- [33] M. A. Abido, "Environmental/Economic Power Dispatch Using Multiobjective Evolutionary Algorithms," *IEEE Trans. on Power Systems*, Vol. 18, No. 4, November 2003, pp. 1529-1537.
- [34] M. A. Abido, "Multiobjective Evolutionary Algorithms for Electric Power Dispatch Problem" *IEEE Trans. on Evolutionary Computation*, Vol. 10, No. 3, June 2006, pp. 315-329.
- [35] [Website] [<http://www.tik.ee.ethz.ch/~zitzler/testdata.html>]
- [36] J. D. Schaffer, *Multiple Objective Optimization with Vector Evaluated Genetic Algorithms*, PhD Thesis, Vanderbilt University, 1984.
- [37] K. Deb, "Multi-Objective Genetic Algorithms: Problem Difficulties and Construction of Test Problems," *Evolutionary Computation*, Vol. 7, No. 3, 1999, pp. 205-230.