# A Multiobjective Memetic Algorithm Based on Particle Swarm Optimization

Dasheng Liu, K. C. Tan, C. K. Goh, and W. K. Ho

***Abstract*—In this paper, a new memetic algorithm (MA) for multiobjective (MO) optimization is proposed, which combines the global search ability of particle swarm optimization with a synchronous local search heuristic for directed local fine-tuning. A new particle updating strategy is proposed based upon the concept of fuzzy global-best to deal with the problem of premature convergence and diversity maintenance within the swarm. The proposed features are examined to show their individual and combined effects in MO optimization. The comparative study shows the effectiveness of the proposed MA, which produces solution sets that are highly competitive in terms of convergence, diversity, and distribution.**

***Index Terms*—Memetic algorithm (MA), multiobjective (MO) optimization, particle swarm optimization (PSO).**

## I. Introduction

MANY real-world optimization problems involve optimizing multiple noncommensurable and often competing criteria that reflect various design specifications and constraints [22]. Ever since the pioneering effort of Schaffer [18], many evolutionary techniques for multiobjective (MO) optimization have been proposed. A few of these algorithms include the nondominated sorting genetic algorithm II (NSGAII) [4], the strength Pareto evolutionary algorithm 2 (SPEA2) [23], the incrementing MO evolutionary algorithm (IMOEA) [20], etc.

Particle swarm optimization (PSO) [10] is a stochastic optimization technique that is inspired by the behavior of bird flocks. Although PSO is relatively new, it has been shown to offer higher convergence speed for MO optimization as compared to canonical MOEAs. In recent years, the field of MOPSO has been steadily gaining attention from the research community [2], [6], [14], [17].

It is known that memetic algorithms (MAs) [15], hybridizing EAs and local search (LS) heuristics can be implemented to maintain a balance between exploration and exploitation, which is often crucial to the success of the search and optimization processes [12]. While many works on such hybrids exist in the context of EAs [1], [8], [9], [13], [16], MA is rarely considered in PSO.

This paper is concerned with the development of a multiobjective memetic algorithm (MOMA) within the context of PSO. In contrast to single-objective (SO) optimization, it is essential to obtain a well-distributed and diverse solution set for finding the final tradeoff in MO optimization. However, the high speed of convergence in PSO often implies a rapid loss of diversity during the optimization process, which inevitably leads to undesirable premature convergence. Thus, the challenge in designing a MOMA based upon PSO is to deal with the premature convergence without compromising the convergence speed.

Apart from hybridizing LS heuristic and PSO, the existing particle updating strategy in PSO is extended to account for the requirements in MO optimization. Two heuristics are proposed to deal with the issue, including a synchronous particle LS (SPLS) and a fuzzy global-best (f-gbest) for the updating of a particle trajectory. The SPLS utilizes swarm information to perform a directed fine-tuning operation, while the second heuristic is based upon the concept of possibility [5] to deal with the problem of maintaining diversity within the swarm as well as to promote exploration in the search.

The remainder of this paper is organized as follows. Some background information is provided in Section II, while details of the proposed features of the MOMA are described in Section III. The proposed MAs performance on benchmarks is shown in Section IV. Section V examines the individual and combined effects of the proposed features, and Section VI presents a comparative study of the proposed MA with well-known MO optimization algorithms on a number of benchmark problems. Conclusions are drawn in Section VII.

## II. Preliminaries

### A. MO Optimization

In general, many real-world applications involve complex optimization problems with various competing specifications and constraints. Without loss of generality, we consider a minimization problem with decision space $\boldsymbol{X}$, which is a subset of real numbers. For the minimization problem, it tends to find a parameter set $\boldsymbol{P}$ for

$$\operatorname*{Min}_{\boldsymbol{P}\in X} \boldsymbol{F}(\boldsymbol{P}), \quad \boldsymbol{P}\in\mathbb{R}^D \tag{1}$$

where $P=\{p_1,p_2,\ldots,p_D\}$ is a vector with $D$ decision variables and $F=\{f_1,f_2,\ldots,f_M\}$ are $M$ objectives to be minimized.

The solution to the MO optimization problem exists in the form of an alternate tradeoff known as a Pareto optimal set. Each objective component of any nondominated solution in the Pareto optimal set can only be improved by degrading at least

Manuscript received May 12, 2005; revised October 28, 2005 and January 5, 2006. This paper was recommended by Associate Editor Y. S. Ong.

The authors are with the Control and Simulation Laboratory, Department of Electrical and Computer Engineering, National University of Singapore, Singapore 117576 (e-mail: 90301345@nus.edu.sg).

Digital Object Identifier 10.1109/TSMCB.2006.883270

one of its other objective components. A vector $\boldsymbol{F}_a$ is said to dominate another vector $\boldsymbol{F}_b$, denoted as

$$\begin{aligned}\boldsymbol{F}_a \prec \boldsymbol{F}_b, \quad &\text{iff } f_{a,i} \leq f_{b,i} \qquad \forall i = \{1, 2, \ldots, M\} \\ &\text{and } \exists j \in \{1, 2, \ldots, M\} \text{ where } f_{a,i} \prec f_{b,i}. \end{aligned} \tag{2}$$

In the total absence of information regarding the preference of objectives, a ranking scheme based upon the Pareto optimality is regarded as an appropriate approach to represent the fitness of each individual in an EA for MO optimization [7].

### B. Particle Swarm Optimization

A standard particle swarm optimizer maintains a swarm of particles that represent the potential solutions to the problem on hand. Each particle $P_i(= x_{i,1} \times x_{i,2} \times \cdots \times x_{i,D})$ embeds the relevant information regarding the $D$ decision variables $\{x_j, j = 1, 2, \ldots, D\}$ and is associated with a fitness that provides an indication of its performance in the objective space $F \in \mathbb{R}^M$. Its equivalence in $F$ is denoted by $F_i(= f_{i,1} \times f_{i,2} \times \cdots \times f_{i,M})$, where $\{f_k, k = 1, 2, \ldots, M\}$ are the objectives to be minimized.

In essence, the trajectory of each particle is updated according to its own flying experience as well as to that of the best particle in the swarm. The basic PSO algorithm can be described as

$$\begin{aligned}\nu_{i,d}^{k+1} &= w \times \nu_{i,d}^{k} + c_1 \times r_1^k \times \left(p_{i,d}^k - x_{i,d}^k\right) \\ &\quad + c_2 \times r_2^k \times \left(p_{g,d}^k - x_{i,d}^k\right) \end{aligned} \tag{3}$$

$$x_{i,d}^{k+1} = x_{i,d}^{k} + \nu_{i,d}^{k} \tag{4}$$

where $\nu_{i,d}^k$ is the $d$th dimension velocity of particle $i$ in cycle $k$; $x_{i,d}^k$ is the $d$th dimension position of particle $i$ in cycle $k$; $p_{i,d}^k$ is the $d$th dimension of personal best (pbest) of particle $i$ in cycle $k$; $p_{g,d}^k$ is the $d$th dimension of the gbest in cycle $k$; $w$ is the inertia weight; $c_1$ is the cognition weight and $c_2$ is the social weight; and $r_1$ and $r_2$ are two random values uniformly distributed in the range of [0, 1].

### C. Performance Metrics

In order to provide a quantitative assessment for the performance of MO optimizer, three issues are often taken into consideration, i.e., the distribution, the spread across the Pareto optimal front, and the ability to attain the global tradeoff [19]. Comparative studies performed by researchers such as Deb *et al.* [4], etc., made use of a suite of unary performance metrics pertinent to the optimization goals of proximity, diversity, and distribution. In this paper, three different qualitative measures are used.

The metric of generational distance (GD) gives a good indication of the gap between the discovered Pareto front ($\mathrm{PF}_{\mathrm{known}}$) and the true Pareto front ($\mathrm{PF}_{\mathrm{true}}$), which is given by

$$\mathrm{GD} = \left(\frac{1}{n_{\mathrm{PF}}} \sum_{i=1}^{n_{\mathrm{PF}}} d_i^2\right)^{1/2} \tag{5}$$

where $n_{\mathrm{PF}}$ is the number of members in $\mathrm{PF}_{\mathrm{known}}$ and $d_1$ is the Euclidean distance between the member $i$ in $\mathrm{PF}_{\mathrm{known}}$ and its nearest member in $\mathrm{PF}_{\mathrm{true}}$.

The metric of maximum spread (MS) measures how "well" the $\mathrm{PF}_{\mathrm{true}}$ is covered by the $\mathrm{PF}_{\mathrm{known}}$ through hyperboxes formed by the extreme function values observed in the $\mathrm{PF}_{\mathrm{true}}$ and $\mathrm{PF}_{\mathrm{known}}$. It is defined as

$$\mathrm{MS} = \sqrt{\frac{1}{M} \sum_{i=1}^{M} \left[\frac{\min(f_i^{\max}, F_i^{\max}) - \max(f_i^{\min}, F_i^{\min})}{F_i^{\max} - F_i^{\min}}\right]^2} \tag{6}$$

where $M$ is the number of objectives, $f_i^{\max}$ and $f_i^{\min}$ are the maximum and minimum of the $i$th objective in $\mathrm{PF}_{\mathrm{known}}$, respectively, and $F_i^{\max}$ and $F_i^{\min}$ are the maximum and minimum of the $i$th objective in $\mathrm{PF}_{\mathrm{true}}$, respectively.

The metric of spacing $(S)$ gives an indication of how evenly the solutions are distributed along the discovered front

$$S = \left[\frac{1}{n_{\mathrm{PF}}} \sum_{i=1}^{n_{\mathrm{PF}}} (d_i' - \bar{d}')^2\right]^{1/2} \Big/ \bar{d}', \quad \bar{d}' = \frac{1}{n_{\mathrm{PF}}} \sum_{i=1}^{n_{\mathrm{PF}}} d_i' \tag{7}$$

where $n_{\mathrm{PF}}$ is the number of members in $\mathrm{PF}_{\mathrm{known}}$ and $d_i'$ is the Euclidean distance (in the objective domain) between the member $i$ in $\mathrm{PF}_{\mathrm{known}}$ and its nearest member in $\mathrm{PF}_{\mathrm{known}}$.

## III. MO Memetic PSO

As described in the Introduction, the proposed MA for MO optimization aims to preserve population diversity for finding the optimal Pareto front and to include the feature of local fine-tuning for good population distribution by filling any gaps or discontinuities along the Pareto front. In this section, the proposed features of SPLS and f-gbest will be described, and the implementation detail of the MA will be presented.

### A. Archiving

In our algorithm, elitism [21] is implemented in the form of a fixed-size archive to prevent the loss of good particles due to the stochastic nature of the optimization process. The size of the archive can be adjusted according to the desired number of particles distributed along the tradeoff in the objective space. The archive is updated at each cycle, e.g., if the candidate solution is not dominated by any members in the archive, it will be added to the archive. Likewise, any archive members dominated by this solution will be removed from the archive. In order to maintain a set of uniformly distributed nondominated particles in the archive, the dynamic niche-sharing scheme [21] is employed. When the predetermined archive size is reached, a recurrent truncation process [11] based on niche count is used to eliminate the most crowded archive member.

### B. Selection of the Gbest

In MOPSO, gbest plays an important role in guiding the entire swarm toward the global Pareto front. Contrary to SO optimization, the gbest for MO optimization exists in the form of a set of nondominated solutions, which inevitably leads to the issue of selecting the gbest. It is known that the selection of an appropriate gbest is critical for the search of a diverse and uniformly distributed solution set in MO optimization.

Adopting a similar approach in [2], each particle in the swarm will be assigned a nondominated solution from the archive as gbest. Specifically, binary tournament selection of nondominated solutions from the archive is carried out independently for each particle in every cycle. Therefore, each particle is likely to be assigned different archived solution as the gbest. This implies that each particle will search along different direction in the decision space, thus aiding the exploration process in the optimization. In order to promote diversity and to encourage exploration of the least populated region in the search space, the selection criterion for reproduction is based on niche count. In the event of a tie, preference will be given to solutions lying at the extreme ends of an arbitrarily selected objective.

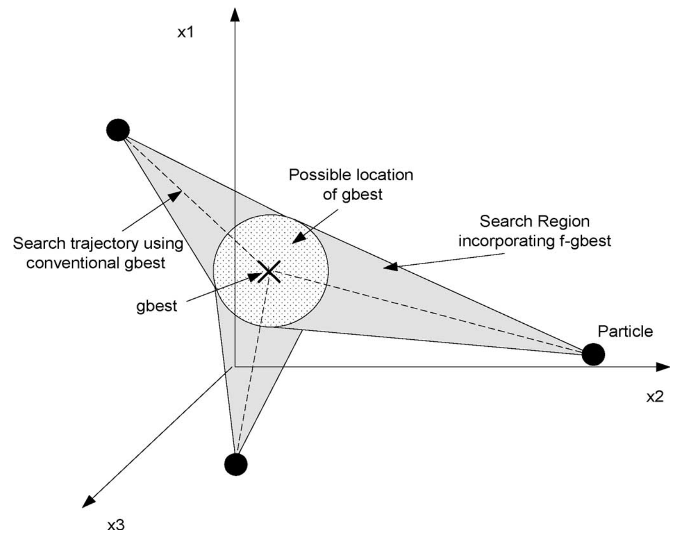



Fig. 1. Search region of f-gbest.

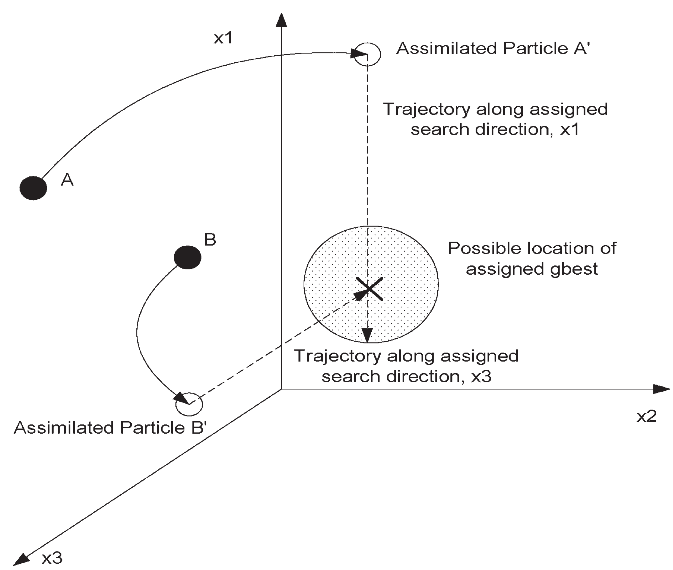



Fig. 2. SPLS of assimilated particle along $x1$ and $x3$.

### C. Fuzzy Global Best (f-gbest)

In PSO, the swarm converges rapidly within the intermediate vicinity of the gbest. However, such a high convergence speed often results in: 1) the lost of diversity required to maintain a diverse Pareto front and 2) premature convergence if the gbest corresponds to a local optima. This motivates the development of f-gbest, which is based on the concept of possibility measure to model the lack of information about the true optimality of the gbest. In contrast to conventional approaches, the gbest is denoted as "possibly at $(d_1 \times d_2 \times d_3 \cdots \times d_D)$," instead of a crisp location.

Consequently, the calculation of particle velocity can be rewritten as

$$p_{c,d}^k = N\left(p_{g,d}^k, \sigma\right) \tag{8}$$

$$\sigma = f(k) \tag{9}$$

$$\begin{aligned} \nu_{i,d}^{k+1} &= w \times \nu_{i,d}^k + c_1 \times r_1^k \times \left(p_{i,d}^k - x_{i,d}^k\right) \\ &\quad + c_2 \times r_2^k \times \left(p_{c,d}^k - x_{i,d}^k\right) \end{aligned} \tag{10}$$

where $p_{c,d}^k$ is the $d$th dimension of f-gbest in cycle $k$. From (8), it can be observed that the f-gbest is characterized by a normal distribution $N(p_{g,d}^k, \sigma)$, where $\sigma$ represents the degree of uncertainty about the optimality of the gbest. In order to account for the information received over time that reduces uncertainty about the gbest position, $\sigma$ is modeled as some nonincreasing function of the number of cycles as given in (9). For simplicity, $f(k)$ is defined as

$$f(k) = \begin{cases} \sigma_{\max}, & \text{cycles} < \alpha \cdot \text{max\_cycles} \\ \sigma_{\min}, & \text{otherwise} \end{cases} \tag{11}$$

where $\sigma_{\max}$, $\sigma_{\min}$, and $\alpha$ are set as 0.15, 0.0001, and 0.4, respectively.

The f-gbest should be distinguished from conventional turbulence or mutation operator, which applies a random perturbation to the particles. The function of f-gbest is to encourage the particles to explore a region beyond that defined by the search trajectory, as illustrated in Fig. 1. By considering the uncertainty associated with each gbest as a function of time, f-gbest provides a simple and efficient exploration at the early stage when $\sigma$ is large and encourages local fine-tuning at the latter stage when $\sigma$ is small. Subsequently, this approach helps to reduce the likelihood of premature convergence and guides the search toward filling any gaps or discontinuities along the Pareto front for better tradeoff representation.

### D. Synchronous Particle Local Search (SPLS)

The MOPSO is hybridized with an SPLS, which performs directed local fine-tuning to improve the distribution of nondominated solutions. The issues considered in the design of the LS operator include: 1) the selection of appropriate search direction; 2) the selection of appropriate particles for local optimization; and 3) the allocation of computational budget for LS.

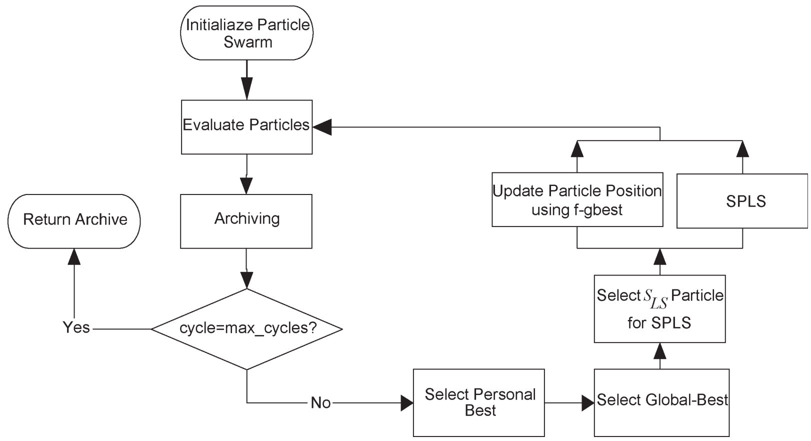



Fig. 3. Flowchart of FMOPSO.

SPLS is performed in the vicinity of the particles, and the procedure is outlined in the following pseudocode.

Step 1) Select $S_{\mathrm{LS}}$ particles randomly from particle swarm.
Step 2) Select $N_{\mathrm{LS}}$ nondominated particles with the best niche count from the archive to a selection pool.
Step 3) Assign an arbitrary nondominated solution from the selection pool to each of the $S_{\mathrm{LS}}$ particles as gbest.
Step 4) Assign an arbitrary search space dimension for each of the $S_{\mathrm{LS}}$ particles.
Step 5) Assimilation: With the exception of the assigned dimension, update the position of $S_{\mathrm{LS}}$ particles in the decision space with the selected gbest position.
Step 6) Update the position of all $S_{\mathrm{LS}}$ assimilated particles using (8)–(11) along the preassigned dimension only.

The operation of SPLS is illustrated in Fig. 2. The rationale of selecting the least crowded particles from the archive in Step 2) is to encourage the discovery of better solutions that fill the gaps or discontinuities along the discovered Pareto front. Instead of selecting particles directly from the archive for fine-tuning purposes, the assimilation process in Step 5) provides a means of integrating swarm and archive information. In contrast to random variation, Step 6) exploits knowledge about the possible location of gbest to probe the feasibility of the gbest position along the assigned direction.

From the pseudocode, it is clear that the allocation of computational resource for LS is determined by $S_{\mathrm{LS}}$. A high setting of $S_{\mathrm{LS}}$ allows the swarm to perform LS from different starting points, i.e., it enables the swarm to exploit along different directions. Likewise, a small setting of $S_{\mathrm{LS}}$ will restrict the LS operation. In our algorithm, the value of $S_{\mathrm{LS}}$ and $N_{\mathrm{LS}}$ is chosen as 20 and 2, respectively.

### E. Implementation

The proposed MA incorporating f-gbest and SPLS is named as FMOPSO. The flowchart of FMOPSO is shown in Fig. 3. In every cycle, archiving is performed after the evaluation process of particles. Then, the pbest and gbest of each particle in the swarm are updated. Note that, the pbest of each particle will be updated only if a better pbest is found, and the f-gbest is implemented in place of conventional deterministic gbest. The update of particle position using f-gbest is performed concurrently with the SPLS. To maintain a balance between the exploration of "fly" and the exploitation of SPLS, the total number of evaluations is kept at the size of the particle swarm $N_{\mathrm{pop}}$ for every cycle. Specifically, $S_{\mathrm{LS}}$ particles will undergo the SPLS, while the rest of the $N_{\mathrm{pop}} - S_{\mathrm{LS}}$ particles in the swarm will be updated with the velocity calculated by (10).

## IV. Benchmark Problem and FMOPSO Performance

In the context of MO optimization, the benchmark problems must pose sufficient difficulty to impede the ability of MOEA in searching for the Pareto optimal solutions. Deb [3] has identified several characteristics such as multimodality, convexity, discontinuity, and nonuniformity, which may challenge the algorithm's ability to converge or to maintain good population diversity in MO optimization. In this paper, six benchmark problems ZDT1, ZDT4, ZDT6, FON, KUR, and POL are selected to examine the performance of the proposed MOMA. Many researchers such as the authors in [4], [17], [22], and [23] have applied these problems to examine their proposed algorithms. The definition of these test functions is summarized in Table I.

The tradeoffs generated by FMOPSO for the different benchmarks in one arbitrary run are shown in Fig. 4 (10 000 evaluations for ZDT1, FON, KUR, and POL, 50 000 evaluations for ZDT4, and 20 000 evaluations for ZDT6). It can be seen that FMOPSO can converge to the true Pareto front and evolve a diverse solution set for all benchmark problems, although the solutions for ZDT1 and FON are not evenly distributed along the true Pareto front. Since FMOPSO has found the near-optimal Pareto front and covered the full extent for ZDT1 and FON, the SPLS will help find more nondominated solutions on the gaps or discontinuities along the Pareto front, and the dynamic niche sharing scheme will improve spacing metric if given more cycles. Fig. 4 shows that FMOPSO can overcome the difficulties presented by these benchmarks.

TABLE I
DEFINITION OF THE TEST PROBLEMS

| Test Problem | Definition |
|---|---|
| ZDT1 | $f_1(x_1) = x_1$ <br> $g(x_2,\cdots,x_m) = 1 + 9\cdot\sum_{i=2}^{m} x_i/(m-1)$ <br> $h(f_1, g) = 1 - \sqrt{f_1/g}$ <br> $\textit{where } m = 30, \textit{ and } x_i \in [0,1].$ |
| ZDT4 | $f_1(x_1) = x_1$ <br> $g(x_2,\cdots,x_m) = 1 + 10(m-1) + \sum_{i=2}^{m}(x_i^2 - 10\cos(4\pi x_i))$ <br> $h(f_1, g) = 1 - \sqrt{f_1/g}$ <br> $\textit{where } m = 10,\ x_1 \in [0,1] \textit{ and } x_2,\ldots,x_m \in [-5,5].$ |
| ZDT6 | $f_1(x_1) = 1 - \exp(-4x_1)\sin^6(6\pi x_1)$ <br> $g(x_2,\cdots,x_m) = 1 + 9((\sum_{i=2}^{m} x_i)/(m-1))^{0.25}$ <br> $h(f_1, g) = 1 - (f_1/g)^2$ <br> $\textit{where } m = 10,\ x_i \in [0,1]$ |
| FON | $\textit{Minimize}\left(f_1, f_2\right)$ <br> $f_1(x_1,\ldots,x_8) = 1 - \exp[-\sum_{i=1}^{8}(x_i - 1/\sqrt{8})^2]$ <br> $f_2(x_1,\ldots,x_8) = 1 - \exp[-\sum_{i=1}^{8}(x_i + 1/\sqrt{8})^2]$ <br> $\textit{where } -2 \le x_i < 2, \forall i = 1,2,\ldots,8$ |
| KUR | $\textit{Minimize}(f_1, f_2)$ <br> $\begin{cases} f_1(x) = \sum_{i=1}^{2}[-10\exp(-0.2\sqrt{x_i^2 + x_{i+1}^2})] \\ f_2(x) = \sum_{i=1}^{3}[\lvert x_i\rvert^{0.8} + 5\sin(x_i^3)] \end{cases}$ <br> $\textit{where } -5 \le x_i < 5, \forall i = 1,2,3$ |
| POL | $\textit{Minimize}\left(f_1, f_2\right)$ <br> $f_1(x) = 1 + (A_1 - B_1)^2 + (A_2 - B_2)^2$ <br> $f_2(x) = (x_1 + 3)^2 + (x_2 + 1)^2$ <br> $A_1 - 0.5\sin 1 - 2\cos 1 + \sin 2 - 1.5\cos 2$ <br> $A_2 = 1.5\sin 1 - \cos 1 + 2\sin 2 - 0.5\cos 2$ <br> $B_1 = 0.5\sin x_1 - 2\cos x_1 + \sin x_2 - 1.5\cos x_2$ <br> $B_2 = 1.5\sin x_1 - \cos x_1 + 2\sin x_2 - 0.5\cos x_2$ <br> $\textit{where } -\pi \le x_i \le \pi, \forall i = 1,2$ |

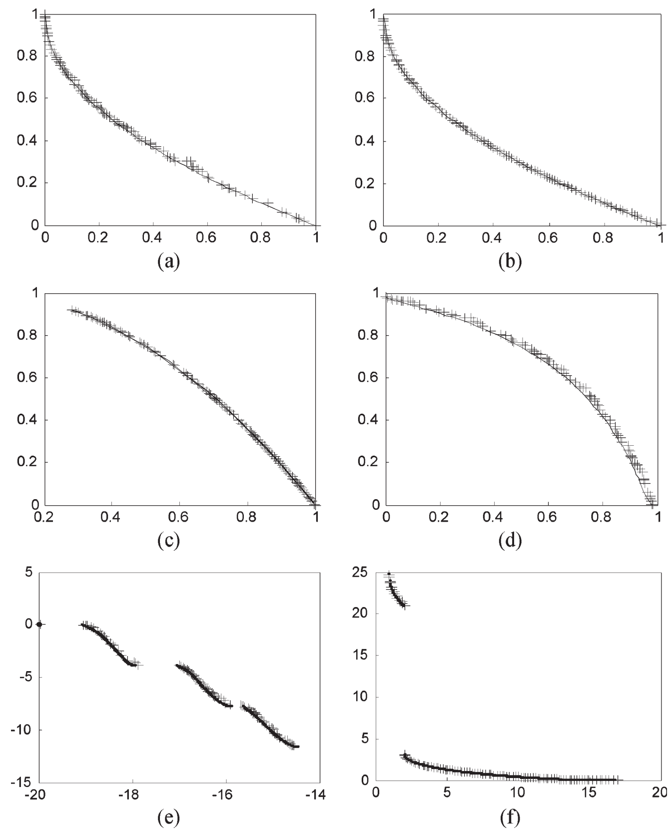


Fig. 4. Evolved tradeoffs by FMOPSO for (a) ZDT1, (b) ZDT4, (c) ZDT6, (d) FON, (e) KUR, and (f) POL.

TABLE II
PERFORMANCE OF DIFFERENT FEATURES FOR ZDT1

| | | Standard | f-gbest only | SPLS only | FMOPSO |
|---|---|---|---|---|---|
| GD | Mean | 0.3179 | 0.0144 | 0.1260 | 0.0046 |
| | Median | 0.2947 | 0.0107 | 0.1137 | 0.0032 |
| | Min | 0.1239 | 0.0031 | 0.0114 | 0.0014 |
| | Max | 0.7973 | 0.0428 | 0.2842 | 0.0180 |
| | Std | 0.1406 | 0.0106 | 0.0597 | 0.0041 |
| S | Mean | 1.0582 | 0.7700 | 0.9740 | 0.8032 |
| | Median | 0.9357 | 0.7459 | 0.8827 | 0.7018 |
| | Min | 0.6542 | 0.4723 | 0.5856 | 0.3543 |
| | Max | 2.4528 | 1.2692 | 1.7858 | 1.9060 |
| | Std | 0.3842 | 0.1996 | 0.2775 | 0.3237 |
| MS | Mean | 0.6703 | 0.9775 | 0.8258 | 0.9977 |
| | Median | 0.6677 | 0.9815 | 0.8437 | 1.0000 |
| | Min | 0.4465 | 0.8950 | 0.5938 | 0.9829 |
| | Max | 0.8961 | 0.9959 | 0.9579 | 1.0000 |
| | Std | 0.1278 | 0.0184 | 0.1003 | 0.0045 |

## V. EXAMINATION OF NEW FEATURES

In this section, four versions of the algorithm (standard MOPSO, MOPSO with f-gbest only, MOPSO with SPLS only, and FMOPSO) are compared to illustrate the individual and combined effects of the proposed features. The results, with respect to the different performance metric of ZDT1 (10 000 evaluations), ZDT4 (50 000 evaluations), and ZDT6 (20 000 evaluations), are summarized in Tables II, III, and IV, respectively. When there are less than three solutions in the archive, NaN is shown as the computation of spacing metric, which requires more than three solutions.

It can be observed from Tables III and IV that the incorporation of SPLS alone can greatly improve the performance of a standard MOPSO in terms of proximity, diversity, and distribution for ZDT4 and ZDT6. The good performance of spacing also shows that the local search ability of SPLS helps the discovery of nondominated solutions on the gaps or discontinuities along the Pareto front. In Table II, even though SPLS only cannot find the true Pareto front for ZDT1, it can still maintain a relatively good spacing metric.

TABLE III
PERFORMANCE OF DIFFERENT FEATURES FOR ZDT4

| | | Standard | f-gbest only | SPLS only | FMOPSO |
|---|---|---|---|---|---|
| GD | Mean | 6.8211 | 7.8011 | 0.2326 | 0.0009 |
| | Median | 6.6020 | 5.3704 | 0.1245 | 0.0008 |
| | Min | 2.7099 | 1.2445 | 0.00073 | 0.00066 |
| | Max | 14.6854 | 26.7440 | 1.1683 | 0.0023 |
| | Std | 2.7790 | 6.8388 | 0.3090 | 0.0003 |
| S | Mean | NaN | NaN | 1.1347 | 0.3255 |
| | Median | NaN | NaN | 1.1103 | 0.3313 |
| | Min | 0.6684 | 0.6032 | 0.3012 | 0.2264 |
| | Max | 2.6588 | 3.1383 | 1.7716 | 0.4541 |
| | Std | NaN | NaN | 0.3304 | 0.0530 |
| MS | Mean | 0.0049 | 0.0038 | 0.7499 | 0.9999 |
| | Median | 0.0006 | 0.0000 | 0.9030 | 1.0000 |
| | Min | 0 | 0 | 0.1815 | 0.9989 |
| | Max | 0.1057 | 0.0727 | 1.0000 | 1.0000 |
| | Std | 0.0193 | 0.0142 | 0.2732 | 0.0002 |

TABLE IV
PERFORMANCE OF DIFFERENT FEATURES FOR ZDT6

| | | Standard | f-gbest only | SPLS only | FMOPSO |
|---|---|---|---|---|---|
| GD | Mean | 2.0468 | 0.0008234 | 0.0671 | 0.0008121 |
| | Median | 2.6877 | 0.0008213 | 0.0008120 | 0.0008171 |
| | Min | 0.0007459 | 0.0006607 | 0.0005993 | 0.0006203 |
| | Max | 4.5743 | 0.0010 | 1.9997 | 0.0009 |
| | Std | 1.8385 | 0.00008347 | 0.3633 | 0.00008171 |
| S | Mean | NaN | 4.4864 | NaN | 0.4406 |
| | Median | NaN | 4.5363 | NaN | 0.4394 |
| | Min | 0.6435 | 1.6784 | 0.2692 | 0.2636 |
| | Max | 6.3745 | 7.6267 | 6.0548 | 0.6302 |
| | Std | NaN | 1.2937 | NaN | 0.0853 |
| MS | Mean | 0.5327 | 0.9138 | 0.9393 | 0.9992 |
| | Median | 0.5893 | 0.9995 | 0.9992 | 0.9992 |
| | Min | 0 | 0.0271 | 0 | 0.9992 |
| | Max | 0.9995 | 0.9995 | 0.9992 | 0.9992 |
| | Std | 0.3726 | 0.2352 | 0.2055 | 0.0000 |

From Tables II and IV, it can be seen that the implementation of f-gbest alone helps MOPSO to discover the near-optimal Pareto front for ZDT1 and ZDT6. However, the f-gbest alone cannot guarantee a good performance on spacing metric. This is expected since the enhancement of global search capability provided by f-gbest is not aimed for the uniform distribution of solutions along the Pareto front. On the other hand, the combination of f-gbest and SPLS allows the discovery of a well-distributed and diverse solution set for ZDT1, ZDT4, and ZDT6 without compromising the convergence speed of the algorithm.

Table III shows that f-gbest only cannot find the true Pareto front of ZDT4 in 50 000 evaluations. The evolutionary traces shown in Fig. 5 reinforce such observation. From the evolutionary traces, it can be observed that f-gbest only can also lead the search to come closer and closer to the true Pareto front, as shown by the decreasing GD, but the improvement is too slow for ZDT4 without the guide of SPLS. Without the balance between the exploration of fuzzy update and the exploitation of SPLS, f-gbest only cannot deal with the multimodality nature of ZDT4 effectively. The discontinuity in the spacing metric of f-gbest only is because in those cycles there are less than three nondominated solutions in the archive. Fig. 5 also ascertains the exploitative nature of SPLS as reflected by the fast convergence speed relative to that without LS (f-gbest only).

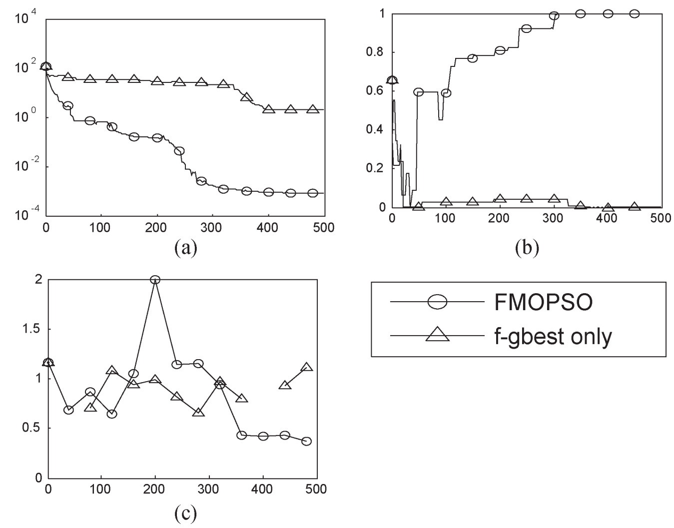



Fig. 5. Evolutionary trajectories in (a) GD, (b) MS, and (c) $S$ for ZDT4.

It can be noted from Table II that SPLS alone cannot overcome the difficulty of ZDT1. Further information can also be extracted by comparing the explored objective space for the algorithms at different timeline, as shown in Fig. 6. From Fig. 6, it is observed that the explored objective space for FMOPSO is more extensive. By taking into account the uncertainty of the information gathered about the gbest position, the fuzzy updating strategy actually prevents the evolving particles to converge upon similar regions in the search space. Therefore, in the evolution process, the explored space is extended. Standard update does not provide enough diversity for ZDT1. The fast convergence ability of SPLS leads the SPLS only algorithm to stagnate at different local Pareto front.

## VI. COMPARATIVE STUDY

In this section, the performance of FMOPSO is compared to five existing MO algorithms, including CMOPSO [2], SMOPSO [14], IMOEA [20], NSGAII [4], and SPEA2 [23]. All the algorithms were implemented in C++, and the simulations were performed on an Intel Pentium IV 2.8-GHz personal computer. Thirty simulation runs were performed for each algorithm on each test problem in order to study the statistical performance. A random initial population was created for each of the 30 runs on each test problem. Note that, the number of evaluations is set to be relatively small to examine the convergence of FMOPSO. The parameter settings and indexes of the different algorithms are shown in Tables V and VI, respectively. Fig. 7 summarizes the statistical performance of the different algorithms.

### A. ZDT1

From Fig. 7(a)–(c), it can be observed that, except FMOPSO, other algorithms still have many solutions that are located far from the true Pareto front. It can be seen that the average performance of FMOPSO is the best among the six algorithms adopted. In addition, FMOPSO is also able to evolve a diverse solution set, as evident from Fig. 7(b) and (c). At the same time, we note that the solutions are not evenly distributed along

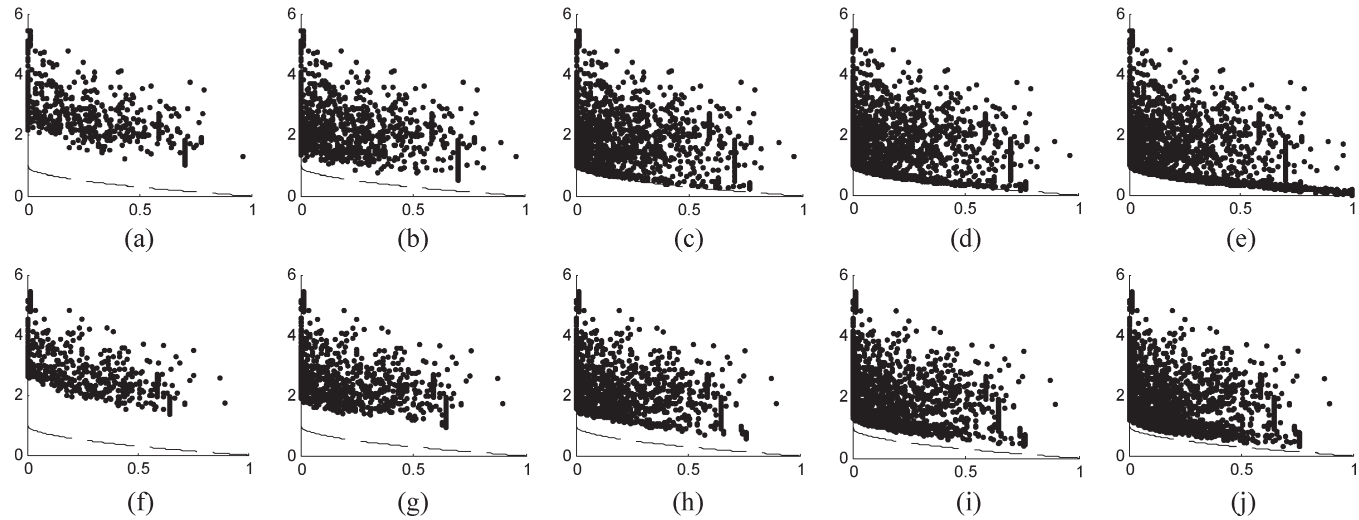


Fig. 6. (Top) Explored objective space FMOPSO at generation (a) 20, (b) 40, (c) 60, (d) 80, and (e) 100 and (bottom) SPLS only at generation (f) 20, (g) 40, (h) 60, (i) 80, and (j) 100 for ZDT1.

TABLE V
PARAMETER SETTINGS OF THE DIFFERENT ALGORITHMS

| | |
|---|---|
| Populations | Population size 100 in FMOPSO, CMOPSO, SMOPSO, NSGAII, and SPEA2.<br>Population size 20~100 in IMOEA.<br>Archive (or secondary population) size 100. |
| Representation | 15 bits for each variable in IMOEA, NSGAII and SPEA2.<br>Real number representation in FMOPSO, CMOPSO, and SMOPSO. |
| Selection | Binary tournament selection |
| Crossover rate | 0.8 in IMOEA, NSGAII, and SPEA2 |
| Crossover method | Uniform crossover in IMOEA, NSGAII, and SPEA2 |
| Mutation rate | *1/chromosome_length* for ZDT1, ZDT4 and ZDT6.<br>*1/bit_number_per_variable* for FON, KUR and POL. |
| Mutation method | Bit-flip mutation for IMOEA, NSGAII and SPEA2.<br>Adaptive mutation for CMOPSO.<br>Turbulence operator for SMOPSO. |
| Grid Division | 30 for CMOPSO |
| Evaluations | 10,000 for ZDT1, FON, KUR and POL; 50,000 for ZDT4; 20,000 for ZDT6. |

TABLE VI
INDEXES OF THE DIFFERENT ALGORITHMS

| Index | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| Algorithm | FMOPSO | CMOPSO | SMOPSO | IMOEA | NSGA | SPEA2 |

the true Pareto front, as illustrated by a relatively large spacing metric. Since FMOPSO has covered the full extent of the true Pareto front, as illustrated by the high value of MS, and the evaluation number is set to be relatively small for ZDT1 (only 10 000 evaluations), it will improve spacing metric if given more evaluations for the incorporated SPLS and dynamic niche sharing scheme.

### B. ZDT4

It can be observed from Fig. 7(d) that CMOPSO, SMOPSO, IMOEA, NSGAII, and SPEA2 have a relatively large GD for ZDT4 at the end of 50 000 evaluations. Besides, Fig. 7(e) shows that CMOPSO and SMOPSO are unable to evolve a diverse set of solutions consistently. On the other hand, FMOPSO is able to escape the local optima of ZDT4 consistently, as reflected by its low value of GD. The FMOPSO is also able to evolve a diverse and well-distributed solution set within 50 000 evaluations, resulting in a high value of MS and a low value of $S$.

### C. ZDT6

From Fig. 7(g), it can be seen that CMOPSO, NSGAII, and SPEA2 failed to find the true Pareto front for ZDT6 within 20 000 evaluations. Although the performance of SMOPSO and IMOEA on GD are better than the aforementioned three algorithms, they failed to evolve a diverse and well-distributed solution set, as shown in Fig. 7(h) and (i). Overall, the FMOPSO is able to evolve a diverse and well-distributed nearly optimal Pareto front for ZDT6 within 20 000 evaluations.

### D. FON

It can be observed from Fig. 7(j) that most of the algorithms are able to find at least part of the true Pareto front for FON. Besides, the PSO paradigm appears to have a slight edge in dealing with the nonlinear tradeoff curve of FON, i.e., the three PSO-based algorithms outperformed other algorithms consistently on GD. It can be seen from Fig. 7(k) that CMOPSO and SMOPSO failed to evolve a diverse set of solutions as compared to FMOPSO. This could be due to the fast convergence of CMOPSO and SMOPSO, which may result in the loss of diversity required for covering the entire final Pareto front.

### E. KUR

The disconnection of the Pareto front of KUR seems not a very big problem for the algorithms adopted here, as shown in Fig. 7(m)–(o). All algorithms are able to find the nearly optimal Pareto front. However, FMOPSO has the best performance in MS. In addition, FMOPSO also showed competitive performance in terms of convergence and distribution.

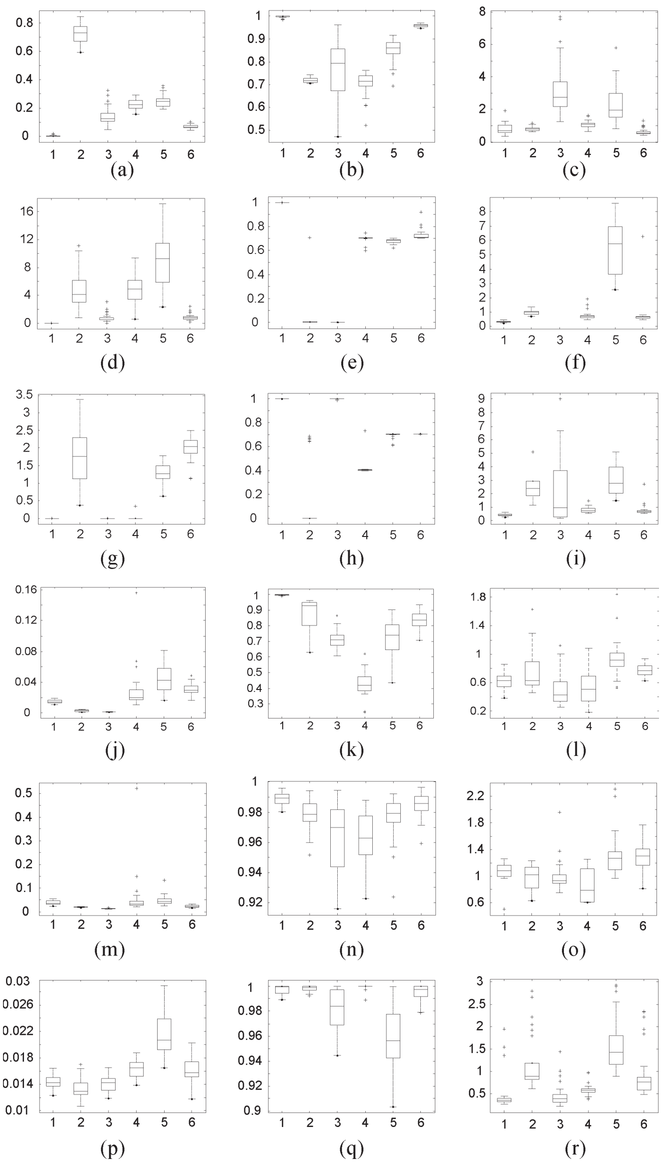


Fig. 7. Statistical performance of the different algorithms. (a) GD, (b) MS, and (c) $S$ for ZDT1. (d) GD, (e) MS, and (f) $S$ for ZDT4. (g) GD, (h) MS, and (i) $S$ for ZDT6. (j) GD, (k) MS, and (l) $S$ for FON. (m) GD, (n) MS, and (o) $S$ for KUR. (p) GD, (q) MS, and (r) $S$ for POL.

### *F. POL*

From Fig. 7(p)–(r), it can be noted that NSGAII has the worst results in terms of convergence, diversity, and distribution. On the other hand, FMOPSO, CMOPSO, SMOPSO, and IMOEA showed competitive performance. PSO paradigm has a slight edge in the aspect of convergence. However, it should be noted that the performance of CMOPSO and SMOPSO in the aspect of $S$ and MS, respectively, are not as good as IMOEA and FMOPSO.

## VII. Conclusion

A new MOMA has been designed within the context of PSO. In particular, two new features in the form of f-gbest and SPLS have been proposed. The SPLS performs directed local fine-tuning, which helps to discover a well-distributed Pareto front. The f-gbest models the uncertainty associated with the optimality of gbest, thus helping the algorithm to avoid undesirable premature convergence. The proposed features have been examined to show their individual and combined effects in MO optimization. The comparative study showed that the proposed FMOPSO produced solution sets that are highly competitive in terms of convergence, diversity, and distribution.

## References


[1] E. K. Burke, P. Cowling, and P. De Causmaecker, "A memetic approach to the nurse rostering problem," *Appl. Intell.*, vol. 15, no. 3, pp. 199–214, Nov. 2001.

[2] C. A. C. Coello, G. T. Pulido, and M. S. Lechuga, "Handling multiple objectives with particle swarm optimization," *IEEE Trans. Evol. Comput.*, vol. 3, no. 3, pp. 256–279, Jun. 2004.

[3] K. Deb, "Multi-objective genetic algorithms: Problem difficulties and construction of test problem," *Evol. Comput.*, vol. 7, no. 3, pp. 205–230, 1999.

[4] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, Apr. 2002.

[5] D. Dubois and H. Prade, *Possibility Theory: An Approach to Computerized Processing and Uncertainty*. New York: Plenum, 1988.

[6] J. E. Fieldsend and S. Singh, "A multi-objective algorithm based upon particle swarm optimization, an efficient data structure and turbulence," in *Proc. U.K. Workshop Comput. Intell.*, Birmingham, U.K., Sep. 2002, pp. 37–44.

[7] C. M. Fonseca and P. J. Fleming, "Genetic algorithm for multiobjective optimization, formulation, discussion and generalization," in *Proc. 5th Int. Conf. Genetic Algorithms*, 1993, pp. 416–423.

[8] P. M. Franca, A. Mendes, and P. Moscato, "A memetic algorithm for the total tardiness single machine scheduling problem," *Eur. J. Oper. Res.*, vol. 132, no. 1, pp. 224–242, 2001.

[9] H. Ishibuchi, T. Yoshida, and T. Murata, "Balance between genetic search and local search in memetic algorithms for multiobjective permutation flowshop scheduling," *IEEE Trans. Evol. Comput.*, vol. 7, no. 2, pp. 204–223, Apr. 2003.

[10] J. Kennedy and R. C. Eberhart, "Particle swarm optimization," in *Proc. IEEE Int. Conf. Neural Netw.*, Perth, Australia, 1995, pp. 1942–1948.

[11] E. F. Khor, K. C. Tan, T. H. Lee, and C. K. Goh, "A study on distribution preservation mechanism in evolutionary multi-objective optimization," *Artif. Intell. Rev.*, vol. 23, no. 1, pp. 31–56, Mar. 2005.

[12] N. Krasnogor and J. E. Smith, "A tutorial for competent memetic algorithms: Model, taxonomy and design issues," *IEEE Trans. Evol. Comput.*, vol. 9, no. 5, pp. 474–488, Oct. 2005.

[13] P. Merz and B. Freisleben, "Fitness landscape analysis and memetic algorithms for the quadratic assignment problem," *IEEE Trans. Evol. Comput.*, vol. 4, no. 4, pp. 337–352, Nov. 2000.

[14] S. Mostaghim and J. Teich, "Strategies for finding good local guides in multi-objective particle swarm optimization (MOPSO)," in *Proc. IEEE Swarm Intell. Symp.*, Indianapolis, IN, Apr. 2003, pp. 26–33.

[15] P. Moscato, "On evolution, search, optimization, GAs and martial arts: Toward memetic algorithm," Caltech Concurrent Comput. Prog., California Inst. Technol., Pasadena, Tech. Rep. 826, 1989.

[16] Y. S. Ong and A. J. Keane, "Meta-Lamarckian learning in memetic algorithms," *IEEE Trans. Evol. Comput.*, vol. 8, no. 2, pp. 99–110, Apr. 2004.

[17] K. E. Parsopoulous and M. N. Vrahatis, "Particle swarm optimization method in multimobjective problems," in *Proc. ACM SAC*, Madrid, Spain, 2002, pp. 603–607.

[18] J. D. Schaffer, "Multiple objective optimization with vector evaluated genetic algorithms," in *Proc. 1st Int. Conf. Genetic Algorithms: Genetic Algorithms and Their Appl.*. Mahwah, NJ: Lawrence Erlbaum, 1985, pp. 93–100.

[19] K. J. Shaw, A. L. Notcliffe, M. Thompson, J. Love, C. M. Fonseca, and P. J. Fleming, "Assessing the performance of multiobjective genetic algorithms for optimization of batch process scheduling problem," in *Proc. Conf. Evol. Comput.*, 1999, vol. 1, pp. 37–45.

[20] K. C. Tan, T. H. Lee, and E. F. Khor, "Evolutionary algorithms with dynamic population size and local exploration for multiobjective optimization," *IEEE Trans. Evol. Comput.*, vol. 5, no. 8, pp. 565–588, 2002.

[21] K. C. Tan, E. F. Khor, T. H. Lee, and R. Sathikannan, "An evolutionary algorithm with advanced goal and priority specification for multiobjective optimization," *J. Artif. Intell. Res.*, vol. 18, pp. 183–215, 2003.

[22] K. C. Tan, E. F. Khor, and T. H. Lee, *Evolutionary Multiobjective Optimization: Algorithms and Applications*. New York: Springer-Verlag, 2005.

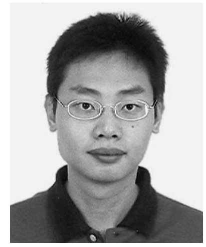
[23] E. Zitzler, M. Laumanns, and L. Thiele, “SPEA2: Improving the strength Pareto evolutionary algorithm,” Comput. Eng. Networks Lab. (TIK), Swiss Fed. Inst. Technol. (ETH), Zurich, Switzerland, Tech. Rep. 103, May 2001.

**Dasheng Liu** received the bachelor and master degrees in electrical engineering from Tianjin University, Tianjin, China, in 2000 and 2003, respectively. He is currently working toward the Ph.D. degree at the National University of Singapore, Singapore.

His research interests include evolutionary computation, fuzzy logic, and neural networks, currently focusing on the application of particle swarm optimization.

**K. C. Tan** received the B.Eng. degree (first class honors) in electronics and electrical engineering and the Ph.D. degree from the University of Glasgow, Glasgow, U.K., in 1994 and 1997, respectively.

He is actively pursuing research in computational intelligence and evolutionary multiobjective optimization, with applications to scheduling, control, data mining, and robotics.



**C. K. Goh** received the B.Eng. degree in electrical engineering from the National University of Singapore, Singapore, in 2003. He is currently working toward the Ph.D. degree at the Centre for Intelligent Control, National University of Singapore.

His current research interests include evolutionary computation and neural networks, specifically in the application of evolutionary techniques to multiobjective optimization.

**W. K. Ho** received the B.Eng. and Ph.D. degrees in electrical engineering from the National University of Singapore, Singapore, in 1987 and 1992, respectively.

Currently, he is an Associate Professor with the Department of Electrical Engineering, National University of Singapore and the Director of the Centre for Intelligent Control, Institute of Engineering Sciences. His research interests include control and signal processing in semiconductor manufacturing and process control. He is the coauthor of the book *Adaptive Control* (Instrument Society of America, 1993). He was a Visiting Scholar at Stanford University from 1998 to 2001.