Particle Swarm Optimization With Composite Particles in Dynamic Environments

Lili Liu, Shengxiang Yang, *Member, IEEE*, and Dingwei Wang

Abstract—In recent years, there has been a growing interest in the study of particle swarm optimization (PSO) in dynamic environments. This paper presents a new PSO model, called PSO with composite particles (PSO-CP), to address dynamic optimization problems. PSO-CP partitions the swarm into a set of composite particles based on their similarity using a "worst first" principle. Inspired by the composite particle phenomenon in physics, the elementary members in each composite particle interact via a velocity-anisotropic reflection scheme to integrate valuable information for effectively and rapidly finding the promising optima in the search space. Each composite particle maintains the diversity by a scattering operator. In addition, an integral movement strategy is introduced to promote the swarm diversity. Experiments on a typical dynamic test benchmark problem provide a guideline for setting the involved parameters and show that PSO-CP is efficient in comparison with several state-of-the-art PSO algorithms for dynamic optimization problems.

Index Terms—Composite particle, dynamic optimization problem (DOP), particle swarm optimization (PSO), scattering operator, velocity-anisotropic reflection (VAR).

I. INTRODUCTION

PARTICLE SWARM optimization (PSO) has become an active branch of swarm intelligence (SI) during the last decade [4], [26], [36], [38]. As a population-based technique, PSO was inspired by the emergent motion of swarms and flock behavior [25], [40], [51], [52]. Instead of manipulating new individuals through information recombination as in evolutionary algorithms (EAs), particles in PSO iteratively explore optima in a multidimensional search space by utilizing personal memories and sharing the information within a specific neighborhood. This motion principle of particles makes PSO a natural candidate for solving static optimization problems [20], [37], where the search space remains fixed during the optimization progress. For static problems, the aim is to design PSO algorithms that can quickly and precisely find optima in the search space.

However, most real-world optimization problems are subject to changes in the environment [17], [33]. For these dynamic optimization problems (DOPs), the evaluation function and/or environmental conditions may change over time due to many factors such as the stochastic arrival of new tasks, machine breakdown or maintenance, market fluctuation, and financial variations. Given this, the goal of an optimization algorithm would now be to track the moving optima or to find emerging new optima in the search space, as opposed to seeking a fixed satisfactory solution. This new goal poses a big challenge to classic EAs and SI [24], [39].

In recent years, the study of DOPs has attracted a growing interest from the EA community. EAs are intrinsically inspired by natural or biological evolution, which is always subject to an ever-changing environment. Several approaches have been developed into traditional EAs to address DOPs [24], [47], including diversity schemes [10], [14], [41], [45], [46], memory schemes [5], [50], multipopulation schemes [6], [49], and adaptive schemes [31], [48].

In a similar fashion to EAs, SI has also been used recently to address DOPs with several promising results [3], [15], [16]. In particular, the investigation of PSO in dynamic environments has become one of the most important applications of SI [2], [21]. However, similar to EAs, classic PSO should be modified to deal with the convergence problem for solving DOPs: Once the swarm has converged, particles lose the ability to track new optima due to the low velocity [2]. To address this problem, various approaches have been developed to improve the performance of PSO algorithms for DOPs [1], [18], [34].

Within PSO, the principles that govern the movement of a particle are the interactions among particles and the retrospection of the past experience of the particle. In general, a particle in PSO is driven toward two attractors: the best global location found by the swarm within a specific neighborhood and the best local location found by the particle itself so far. Due to the special features of dynamic interactions and information sharing, PSO turns out to be potentially attractive for solving DOPs. Therefore, PSO with effective interaction strategies, which can combine the technique of fully integrating valuable information of particles to quickly drive particles to exploit more promising regions and diversification methods to improve the exploration capacity, should have a good potential in dynamic environments.

In this paper, we investigate the introduction of one prevalent concept of composite particles from the domain of physics into PSO to elaborate the behavior of particles. Through integrating some principles of this concept with the classic PSO, a PSO algorithm with composite particles, denoted *PSO-CP* in this paper, has recently been proposed in [27] with some promising preliminary results. This paper further introduces a new

L. Liu and D. Wang are with the College of Information Science and Engineering, Northeastern University, Shenyang 110004, China and also with the Key Laboratory of Integrated Automation of Process Industry (Northeastern University), Ministry of Education, Shenyang 110004, China (e-mail: liulili@ise.neu.edu.cn; dwwang@mail.neu.edu.cn).

S. Yang is with the Department of Computer Science, University of Leicester, University Road, LE1 7RH Leicester, U.K. (e-mail: s.yang@mcs.le.ac.uk).

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

2   IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART B: CYBERNETICS

approach to construct composite particles, which first considers worse particles in the swarm according to their similarity. Based on the experimental results in static landscapes, a novel scattering operator is introduced, with the expectation of regaining the local diversity within composite particles, as well as enhancing the efficiency of the velocity-anisotropic reflection (VAR) operator, which was introduced in [27] for a better tracking capacity for PSO in dynamic environments. In addition, an integral movement strategy is introduced to enhance the ability of peak detection in a new environment. In this paper, the effects of crucial parameters and approaches on the behavior of PSO-CP in dynamic environments are investigated. In addition, we consider the performance comparisons of PSO-CP with other state-of-the-art PSO algorithms to address DOPs.

The rest of this paper is outlined as follows. Section II briefly reviews the principles of PSO and relevant work on PSO in dynamic environments. Section III describes the background on composite particles in physics. Section IV provides the proposed PSO-CP in detail. The experimental results and analysis of PSO-CP in comparison with other PSO algorithms for DOPs are given in Section V. Finally, Section VI concludes this paper with discussions on relevant future work.

II. RELATED WORK

A. PSO

In PSO, each particle holds a memory of the best position that it has seen so far and within its neighborhood. Particles update their velocity based on their current velocity and position and these two attractors. The modification of the moving orbit of a particle, for example, particle i, is described as [25]

$$\vec{v}_i(t+1) = \omega\vec{v}_i(t) + c_1\vec{\xi} \times (\vec{p}_i(t) - \vec{x}_i(t)) \\ + c_2\vec{\eta} \times (\vec{p}_g(t) - q\vec{x}_i(t)) \quad (1)$$
$$\vec{x}_i(t+1) = \vec{x}_i(t) + \vec{v}_i(t+1) \quad (2)$$

where ω is the inertia weight, which controls the degree that the velocity of a particle at time t influences the velocity of that particle at time $t+1$. Vectors $\vec{v}_i(t)$ and $\vec{x}_i(t)$ represent the current velocity and position of particle i at time t, respectively, $\vec{p}_i(t)$ and $\vec{p}_g(t)$ represent the position of the best solution discovered so far by particle i and by all particles in the neighborhood of particle i, respectively, c_1 and c_2 are the acceleration constants that determine the influence of the two attractors to particle i, respectively, and $\vec{\xi}$ and $\vec{\eta}$ are random vectors with each constituent drawn with a uniform probability from [0, 1]. The symbol of "$\times$" represents the component-wise product of the corresponding vectors.

In the classical PSO model, according to (1) and (2), particles share information through the swarm attractor $\vec{p}_g$ and evoke memories by particle attractors $\vec{p}_i$.

B. PSO in Dynamic Environments

In recent years, PSO has been increasingly applied to solve DOPs. When addressing DOPs, an efficient PSO algorithm should be able to continuously track the dynamically changing optima [11], [29]. This challenges classic PSO due to the problem of diversity loss in the swarm with the searching progress. In classic PSO, after initialization, particles in the swarm will usually congregate to local or global optima in the search space over successive iterations [42]–[44]. Thereafter, when a change occurs and the optima in the search space move, the slackened velocities and converged particles will deprive PSO of a sufficient exploration ability to track new optima. Therefore, for DOPs, classic PSO needs to be modified to deal with the diversity loss [2], [3].

For DOPs, one simple method is to view an environmental change as the arrival of a new problem and solve it from scratch [7]. However, some knowledge obtained during the interaction of particles inspired by social networks may still be useful in tracking the orbit of the optima in a new environment. Several mechanisms have been introduced to improve the performance of classic PSO in dynamic environments. These mechanisms include adopting the rediversification scheme when a change is detected [8], maintaining the diversity during the run via repulsion [1] or dynamic network topology [21], [29], and multiswarm schemes [28]. These mechanisms are briefly reviewed below.

Hu and Eberhart [18] suggested randomizing part of or the whole swarm when a change is detected. This rediversification scheme may eliminate some valuable information obtained in the history that may be useful for a new environment.

Repulsion mechanisms have been investigated to increase the swarm diversity in dynamic environments. A PSO model that imitates a cloud of charged particles orbiting around the neutral particles was proposed in [1]. The charge could produce a repulsion force for a converging swarm to track the changed optima.

As a major diversity enhancement scheme, adjusting the form of interaction between particles through a proper topology within the swarm of particles was investigated in dynamic environments. Li and Dam [29] applied a gridlike neighborhood structure for DOPs. Janson and Middendorf [21] adopted hierarchical PSO to address static and dynamic problems. They used a local neighborhood instead of the global neighborhood via a hierarchical structure to maintain the diversity of the swarm.

The multiswarm approach is also a considerable technique for maintaining the swarm diversity, with the purpose of tracking different local optima in the dynamic landscape. An example is the charged PSO studied in [3], which works by constructing a set of swarms that interact locally and globally by an exclusion operator and an anticonvergence operator, respectively. Another multiswarm PSO, studied in [34] and [35], uses the perception of speciation to track multiple peaks simultaneously in dynamic environments. In [28], the speciation mechanism was also combined with the idea of quantum particles to sustain diversity within a species and has been shown a great improvement on optima tracking.

Researchers have also hybridized PSO with other optimization algorithms for DOPs. Esquivel and Coello [12] devised a hybrid PSO (Hy_PSO) model, which uses a dynamic macromutation operator to maintain the diversity within the swarm. A collaborative PSO model, named *collaborative evolutionary swarm optimization* (CESO) and proposed by Lung and Dumitrescu [30], uses two collaborative populations: One evolves with the differential evolution algorithm, and another evolves with the classic PSO. These hybrid approaches also indicate that adopting effective schemes for interactions among particles may provide PSO with competitive performance in dynamic environments.

The aforementioned approaches indicate some key considerations for improving the adaptation of PSO in dynamic environments. These are described as follows.

1) Particles should take into account useful information from some other particles besides the best fit one to prevent the crowding of particles, which seriously restricts the ability of PSO to search for new optima.
2) The weak particles should exploit information from better particles in the neighborhood to fly toward a promising region in the search space (where new optima may exist) as quickly as possible to accelerate the optimization process in a new environment.
3) The fitter particles may take advantage of knowledge from the weaker ones to probe promising areas and explore for new optima.

In the following, a modified PSO model that integrates the notion of a composite particle in physics to meet these requirements is proposed.

III. COMPOSITE PARTICLE IN PHYSICS

The notion of the composite particle in PSO-CP is derived from a branch of physics [13], [32]. In physics, composite particles usually refer to a particular kind of particles that are composed of two or more particles via chemical actions. These composite particles possess not only the properties of each elementary particle but also some composite properties [13]. This composition technique enables composite particles to have some superior properties, such as stability and oxidation resistance, and, hence, enables them to have better performance than single particles in complex environments.

Recently, composite systems that are composed of inorganic particles and organic polymer have been extensively studied. These systems have much better performance because they effectively syncretize the properties of inorganic and organic materials. Composite materials can be obtained by assembling effects of chemical bonds. For example, $CaCO_3$ particles can be adsorbed by porous SiO_2 particles to form $CaCO_3/SiO_2$ composite particles [23].

There are three essential principles behind the composite particle phenomenon in physics. They are summarized below.

1) Internal dissimilarity and compatibility. Elementary particles involved in each composite particle have different surfaces and properties. This way, the composite particle may have diverse effects to fit with its surroundings. On the other hand, there are some compatibilities or similarities among the elementary ones to guarantee a strong cohesion.
2) Internal action. Elementary particles in each composite particle interact with each other by chemical bonding. Adopting nonidentical chemical reactions may lead to different properties of particles.
3) Concerted action. Once composite particles are constructed, each composite particle is considered as a whole to react with other particles in the overall system.

IV. COMPOSITE PSO

In this paper, the concept of composite particles in physics is integrated into the classic PSO algorithm. The proposed PSO-CP model has the following three characteristics.

1) Composite particles are constructed based on the similarity of particles.
2) The information-sharing mechanism among different composite particles utilizes the basic framework of the classic PSO model.
3) The information-sharing among elementary particles within each composite particle employs a special reflection method and an integral movement scheme.

In the following, the construction and operations for composite particles are described in detail.

A. Construction

To simplify the operations, the structure of a composite particle is designed to be a simple geometrical shape, namely, a triangle, which consists of three elementary particles. In our previous work [27], a composite particle is created by first randomly selecting one particle from the swarm and then randomly generating two particles that form a triangle with the selected particle with the length of the interconnecting edges being L. Thereafter, the elementary members of a composite particle are fixed during the whole computation process. This original approach requires a user-specified parameter L, which may not be properly set without some prior knowledge on the problem being solved. On the other hand, a group of particles with a dynamically adjusted set of members according to the feedback obtained from the current fitness landscape may be more competitive, particularly in dynamic environments [35].

As discussed in Section I, a partition method that can appropriately create subpopulations, in which the member particles can fully and effectively absorb the information from others, is vital for achieving better performance of the overall optimization. As a promising niching technique, the species-based mechanism in [35] has been shown to outperform other approaches for locating and tracking multiple optima. To identify multiple species to search different peaks in parallel, it applies a "fit-first" principle to select a species seed and a prespecified radius parameter to determine the member particles for each subpopulation.

In this paper, a similar method with the idea of "worse first" is presented in PSO-CP. A composite particle can be defined as three members with a similar feature, according to the Euclidean distance. Different from the species-based approach in [35], a composite particle is constructed with a priority of first considering the "weak" particles in the swarm. The algorithm (as shown in Fig. 1) of constructing composite particles is performed at each iteration. It starts from a list L_I, in which all particles are sorted in the order of increasing fitness values. Composite particles are constructed one by one by removing particles from L_I. When constructing a composite particle C_i, the worst particle among the remaining particles in L_I is first selected, and then two particles in L_I that are similar (i.e., they are the closest particles to the first selected one) will be selected to form the composite particle.

We adopt such a construction scheme for three factors. First, it is expected that composite particles may adjust their elementary particles dynamically based on the feedback information, which may reflect the current environment, and, hence, encouraging particles (whether fitter ones or not) rapidly and

```
Procedure Construct composite particles (C, N_c):
begin
    L_{I_Size} := swarm_size
    L_I := sort all particles in the increasing fitness order
    for i := 1 to N_c do
        C_i := ∅
    end for
    for i := 1 to N_c do
        x_{i_worst} := select Worst Solution (L_I)
        C_i := C_i ∪ {x_{i_worst}}
        L_{I_Size} := L_{I_Size} − 1
        L_I := L_I − {x_{i_worst}}
        for each x_j ∈ L_I (x_j ≠ x_{i_worst}) do
            calculate Euclidian Distance (x_j, x_{i_worst})
        end for
        for k := 1 to 2 do
            x_i^k := select Closest Solution (L_I, x_{i_worst})
            C_i := C_i ∪ {x_i^k}
            L_{I_Size} := L_{I_Size} − 1
            L_I := L_I − {x_{i_worst}}
        end for
    end for
end
```

Fig. 1.   Procedures for constructing composite particles, where N_c is the size of the set of composite particles C, which is set to $[(swarm\_size - 1)/3]$ in this paper, and C_i is the ith composite particle.

continuously tracking the moving optima. Furthermore, by means of self-adjustment within each composite particle, which will be introduced in the following section, nonpioneering elementary particles with useful information are expected to be sufficiently developed, which helps PSO-CP in tracking the moving optima. It is hoped that particles should exploit information from their near neighbors that hold valuable knowledge, which has been proven to be an effective interaction mode in dealing with multimodal optimization problems both in static and dynamic environments [28], [35].

The particles in the swarm that do not belong to any composite particle are referred to as "independent particles." With the above composite particle construction scheme, the size of independent particles in the swarm is $N_I = swarm\_size - 3 * [(swarm\_size - 1)/3]$. In the case that the value of $swarm\_size$ is a multiple of three, the whole swarm will contain $N_I = 3$ independent particles. With the above construction scheme, it can be seen that independent particles are usually the fittest particles. This allows a divergence of fitter particles, which may be beneficial for a rapid response to changes.

B. Self-Adjustment With an Adaptive Parameter

As discussed in Section III, composite particles in physics may behave better through the interaction of their elementary particles. Similarly, each composite particle in PSO-CP should adjust its internal structure to enhance the efficiency of search in dynamic environments.

For each composite particle, there are two primary concerns. The first one is how to design an interaction method among elementary particles, which may drive the whole particle to explore a promising search space more extensively. A VAR scheme is proposed with the idea of replacing the worst particle with another reflection point toward the "better" direction in the search space. By using this method, it is expected that the new composite particle helps track promising peaks in a new environment. The second concern is how to select a
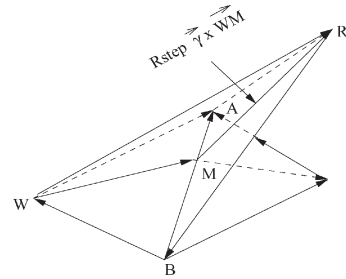

Fig. 2.   Construction of a new composite particle through the VAR scheme.

"pioneer particle" that may synthesize valuable information of the whole composite particle for sharing information with other composite and independent particles in the swarm. This pioneer particle is also responsible for transferring knowledge back to the other two elementary particles within the composite particle. The procedures are described in detail as follows.

1) Substep 1 (VAR With an Adaptive Step Size): The construction of a new composite particle via the VAR scheme is illustrated in Fig. 2. The position of the worst particle in a composite particle is denoted as W. A point, denoted by M, is generated randomly on the straight line with the two endpoints representing the other two elementary particles. If the reflection point is fitter than the worst one, the composite particle is reflected in accordance with the point W to a point R (there is no reflection performed if this is not the case). The new composite particle then consists of points A, R, and B. The reflection point R is calculated as follows:

$$\overrightarrow{WR} = \overrightarrow{WM} + R_{\text{step}}\vec{\gamma} \times \overrightarrow{WM} \tag{3}$$

where R_{step} is the reflection step size that controls the degree that the worst particle moves from point M in the direction of $\vec{\gamma} \times \overrightarrow{WM}$, and $\vec{\gamma}$ is the VAR vector. R_{step} prevents the reflection point R from moving too close to M and, hence, avoids a rapid convergence within the composite particle.

It can be seen that, if $\vec{\gamma}$ is a scalar value, particles will only be able to explore the 2-D search space. To ensure that particles can explore the D-dimensional space comprehensively, a VAR scheme is used in the relevant vector.

Definition: A D-dimensional vector $\vec{\gamma} = (\gamma_1, \gamma_2, \ldots, \gamma_D)$ is a VAR vector if it satisfies

$$|\gamma_i - \gamma_j| \leq d \qquad i, j \in (1, 2, \ldots, D) \tag{4}$$
$$\exists i, j, \quad \text{such that } \gamma_i \neq \gamma_j \tag{5}$$

where d is the maximum difference between reflection velocities for each dimension, which determines the degree of departure from the initial direction $\overrightarrow{WM}$. A larger value of d could produce a more comprehensive reflection space for composite particles.

It is clear that, with the VAR vector shown in (4) and (5), the current composite particle cannot be restricted to the same space in which the former composite particle resides. That is, $\overrightarrow{WR}$ cannot be linearly represented by $\overrightarrow{WA}$ and $\overrightarrow{WB}$ in any case [22]. A brief illustration to support this conclusion is given below.

Denote $\overrightarrow{WA} = (a_1, \ldots, a_D)$ and $\overrightarrow{WB} = (b_1, \ldots, b_D)$. Then, $\overrightarrow{WR} = (\gamma_1(b_1 + (1-z)a_1), \ldots, \gamma_D(b_D + (1-z)a_D))$, where z is generated randomly in the range of $[0.0, 1.0]$. Assume that

$\vec{WR}$ can be linearly represented by $\vec{WA}$ and $\vec{WB}$. Then, there exist k_1 and k_2 that meet the following:

$$(\gamma_1(b_1 + (1-z)a_1), \ldots, \gamma_D(b_D + (1-z)a_D) \\ = k_1(a_1, \ldots, a_N) + k_2(b_1, \ldots, b_N). \quad (6)$$

Then, we have $\gamma_1 = \gamma_2 = \cdots = \gamma_D = k_1/(1-z) = k_2$. This conclusion does not conform to (4). Hence, the primary assumption is false, and the VAR vector can drive composite particles to explore in the D-dimension search space. In this paper, each component in the VAR vector for the ith composite particle is generated by

$$\gamma_{ij} = rand(0, e^{-|v_{ij}/v_{\max}|}), \ i \in \{1, \ldots, N_c\}, \ j \in \{1, \ldots, N\} \quad (7)$$

where v_{ij} and γ_{ij} are the velocity and the reflection factor of the ith composite particle in the jth dimension, respectively, and $N_c = [(swarm\_size - 1)/3]$ is the total number of composite particles in the swarm.

It is noticeable that the reflection factor is designed to be relevant to the velocity of the composite particle. In a sense, the VAR vector works like the chemical bond inside the composite particle in physics as described in Section III. Different values of the elements in the VAR vector can be regarded as different chemical bonds being adopted.

We adopt such a rule for two reasons. First, when the velocities have the tendency to shrink to a small value, particularly when the population converges, the numerical range of the reflection velocity tends to be larger. Hence, the exploration ability will be enhanced adaptively because the degree of departure from the original direction is enlarged. To some extent, the reflection forces the worst elementary particle of a composite particle to depart from the two other ones and strive for better tracking. Second, the difference degree d between the reflection velocities in each dimension will be restricted to a moderate degree in case the reflection direction deviates too much from a "better" direction.

In addition, given that swarm diversification is a critical issue for PSO in dynamic environments, an adaptive reflection step size is also introduced into PSO-CP according to an entropy-based diversity measurement, which is calculated as follows [9]:

$$D_{\text{entropy}}(P) = \left[\frac{1}{\pi}\left(\arctan AD(P) + \frac{\pi}{2}\right)\right]\frac{E(P)}{\log(M)} \quad (8)$$

where

$$E(P) = -\sum_{j=1}^{Q}(p_j \log(p_j)) \quad (9)$$

$$p_j = \frac{|S_j|}{M} \quad (10)$$

where P is the set of individuals with the size of M, $E(P)$ is the entropy of the set P, $S_1, S_2, \ldots, S_Q$ are the subsets of P with the size of $|S_1|, |S_2|, \ldots, |S_Q|$, respectively, and $\forall p, q \in 1, 2, \ldots, Q$ and $p \neq q$, $S_p \cap S_q = \emptyset$.

In (8), $AD(P)$ is the average Euclidean distance between individuals in P, which is defined as follows:

$$AD(P) = \frac{\sum_{i=1}^{M-1}\sum_{k=i+1}^{M} ED(\vec{X}_i, \vec{X}_j)}{M-1} \quad (11)$$

where $ED(\vec{X}_i, \vec{X}_j)$ is the Euclidean distance between individuals $\vec{X}_i$ and $\vec{X}_j$.

With the above definition of the entropy-based diversity measurement, the reflection step size described in (3) can be designed as an adaptive parameter $R_{\text{step\_adp}}$, which can be calculated as follows:

$$R_{\text{step\_adp}} = R_{\text{step}}(1 - D_{\text{entropy}}(CS)) \quad (12)$$

where CS is the current swarm. In this paper, the reflection step size is adaptively adjusted according to the diversity degree of the population measured by $D_{\text{entropy}}(P)$ within the range of [0.0, 1.0]. It is expected that this technique can encourage the detection of new peaks according to the convergence level of the whole swarm.

2) Substep 2 (Pioneer Particle Selection): A pioneer particle, which integrates the valuable information of elementary particles in a composite particle, should be selected to participate in the information-sharing in the whole swarm, similar to what particles in the classic PSO algorithm do. It is practical to select the best elementary particle in a composite particle, which allows elementary particles to move to even fitter positions and, hence, have a better tracking capacity in dynamic environments.

C. Integral Movement

In classic PSO, each particle is driven by the swarm and particle attractors. Nevertheless, some useful information may be erased in the sense that some potential regions in the search space may be skipped over. This is undesirable since it may result in diversity loss, which is a great obstacle for PSO in addressing both static and dynamic problems. Therefore, an integral movement scheme is employed in PSO-CP after the update of pioneer particles. This integral movement scheme conveys the velocity of a pioneer particle to the other two elementary particles in each composite particle.

The reason for introducing this scheme lies in the fact that all elementary particles may move under a right guidance, and that, meanwhile, some valuable information may be preserved for creating new particles for the next iteration. It utilizes the "concerted action" principle of the composite particle in physics, which was described in Section III.

To summarize, Fig. 3 is a schematic diagram that describes the motion of different sorts of particles in PSO-CP.

D. Enhancing Diversification Within a Composite Particle

Although it is expected that the VAR operator and the integral movement help avoid collision and velocity-slackening of elementary particles within a composite particle, the shrinking of the composite particle may still happen if the spatial size is lower than a certain degree. For example, as shown in (3), when $|\vec{WM}|$ is subject to a small value, the diversity cannot be enhanced through the VAR operator, and the VAR operator may even lose the effect of improving the composite particle. When elementary particles are very close to each other, while R_{step} can be set to a large value, the collision of the better two particles still cannot be avoided. This is undesirable since the particle diversity around promising regions is critical for optima tracking [3].
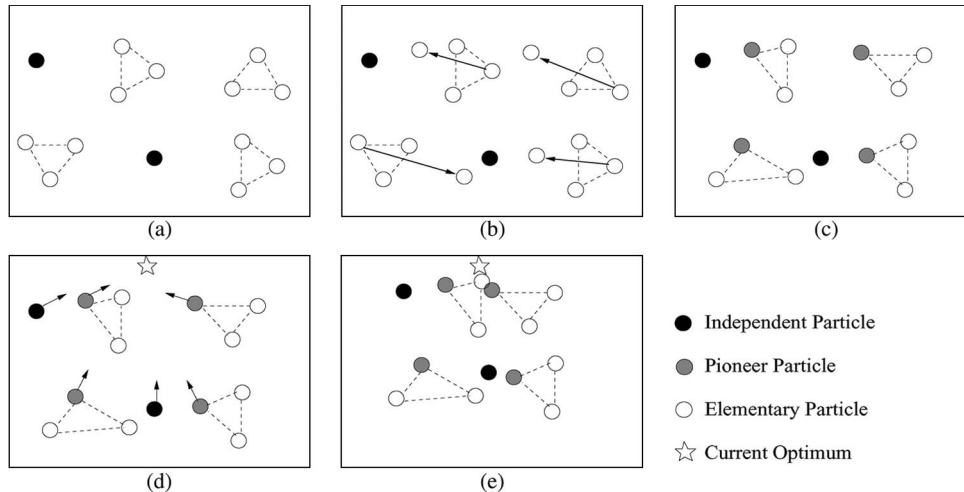
Fig. 3. Illustration of the motion of different sorts of particles. (a) Initialization. (b) Performing the VAR scheme for composite particles. (c) Identifying pioneer particles. (d) Updating the swarm that contains independent and pioneer particles. (e) Performing the integral movement within composite particles.
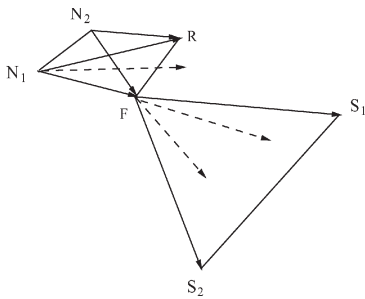

Fig. 4. Generating a new composite particle through the scattering operation.

To prevent the above problem, it is practical to introduce a repulsion operator that guarantees a sufficient "local" diversity level and provides a possible direction to cover peaks in dynamic environments. A new scattering operation is presented here with the above considerations. It is triggered when the composite particle is converged. In a similar fashion to the convergence measurement for species in [28], a composite particle is considered to be converged when the local diversity, defined as the Euclidean distance of the worst member to the furthermost member within the composite particle, is smaller than a threshold θ.

The scattering operation aims to improve the fittest elementary particles to a possible promising direction, according to the information derived from the worse ones, as illustrated in Fig. 4. The position of the fittest elementary particle in a composite particle is denoted by F. The two points, denoted by N_1 and N_2, are the "nonpioneer" particles. The fittest one will be scattered into two new particles along the direction of $\vec{N_1F}$ and $\vec{N_2F}$ to substitute the points of N_1 and N_2. The new composite particle consists of points F, S_1, and S_2, with the scattering points S_1 and S_2 calculated as follows:

$$\vec{FS_1} = \vec{\phi} \times \vec{N_1F} \qquad (13)$$
$$\vec{FS_2} = \vec{\phi} \times \vec{N_2F} \qquad (14)$$

where $\vec{\phi}$ is a random vector with each element generated in the range of $[S_{\text{step\_min}}, S_{\text{step\_max}}]$. Here, $S_{\text{step\_min}}$ and $S_{\text{step\_max}}$ are the minimum and maximum values of the stretching step size, respectively. They control the degree of di-

versity that this operation may promote. In this paper, $S_{\text{step\_min}}$ and $S_{\text{step\_max}}$ are set to 2 and 3, respectively.

There are two reasons we adopt such a principle. First, it is expected that this operation could work effectively when the local diversity of a composite particle has decreased to a certain degree. Once a composite particle converges, the elementary particles may be clustered in a promising area. The directions of $\vec{N_1F}$ and $\vec{N_2F}$, with fitness increasing from the point N_1 and N_2 to F, respectively, are also the right directions toward the fitter regions. Second, due to the special triangular structure of the composite particle, the two scattering particles may be naturally subject to a repulsion pressure since the distance between S_1 and S_2 is larger than that between N_1 and N_2 (see Fig. 4). By applying this scattering operation, it is expected that the population diversity increases, which lead to a better exploration capacity, and that the elementary particles are driven to cover more promising areas.

It is practical to introduce an adaptive method for adjusting the diversity threshold according to the convergence level of composite particles. In this study, (8) is applied to describe the local diversity degree, and the local diversity threshold is adaptively adjusted as follows:

$$\theta_{\text{adp}} = \theta\left(1 - D_{\text{entropy}}(C)\right) \qquad (15)$$

where C is a composite particle. This way, the requirement for the local diversification can be adaptively adjusted according to the convergence level of composite particles.

E. Detection and Response to Changes

To avoid misleading the tracking behavior of particles in dynamic environments, the shared information among composite particles with respect to the best knowledge obtained and the stored information corresponding to each elementary particle's local best solution should be corrected when an environmental change occurs. This requires PSO-CP to detect the changes.

In PSO-CP, the change of the environment can be detected by reevaluating the position $\vec{p_g}$ at each iteration. That is, when the fitness value of $\vec{p_g}$ does not match the stored best value at the last iteration, an environmental change is detected. This detection strategy will work under instances in which the changes

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

LIU *et al.*: PSO WITH COMPOSITE PARTICLES IN DYNAMIC ENVIRONMENTS 7

of the fitness landscape affect the region where $\vec{p_g}$ is located. Further research could focus on a more effective detection method. Whenever an environmental change is detected, a response method is invoked, and each particle's memory, i.e., $\vec{p_i}$, is reevaluated. By using this response method, it is expected that some valuable information in previous environments can be conserved to help the swarm track changes.

In summary, Fig. 5 shows the pseudocode of PSO-CP proposed in this paper to address DOPs. PSO-CP differs from some prior PSO algorithms, such as the niching technique [35], [39] and multiswarm approaches [3], [28], in two aspects. First, the construction of composite particles starts from nonfitter particles to create a subset of the whole swarm without user-specified parameters (i.e., the niching radius). Second, the interaction mechanism among particles within a composite particle does not follow the standard PSO particle motion principle but is implemented by the VAR scheme and the integral movement scheme.

F. Complexity Analysis

The complexity of the algorithm for constructing a composite particle can be estimated according to the number of evaluations of Euclidean distances between particles. Assuming that there are N particles sorted in the list of L_I. Therefore, the number of Euclidean distance calculations required for this procedure, i.e., $T(N)$, can be calculated as follows:

$$T(N) = \sum_{i=1}^{\lfloor \frac{N}{3} \rfloor} (N - 3i + 2). \qquad (16)$$

That is, the worst time complexity of the procedure is $O(N^2)$.

The multiquantum swarm optimization (MQSO) algorithm [3] also has a complexity of $O(N^2)$, which is at the expense of calculating Euclidean distances between particles. However, the partition method that integrally considers the distance and fitness could produce an effective scheme to contribute a full utilization of the involved particles.

The CESO algorithm [30], which hybridizes PSO with other optimization algorithms for addressing dynamic environments, has a complexity of $O(N)$. CESO focuses on using differential evolution to explore the motion principle of the original PSO model, but always performs weaker than PSO-CP according to the experiments reported below. Hence, it is worthy to enhance the adaptivity of PSO by integrating other effective metaheuristic methods for solving DOPs.

V. EXPERIMENTAL STUDY

A. Dynamic Test Environments

The moving peak benchmark (MPB) problem proposed by Branke [5] has been widely used as a dynamic benchmark problem in the literature. Within the MPB, the optima can be varied by three features, i.e., the location, height, and width of the peaks. Hence, it has the flexibility and universality to generate a series of dynamic landscapes with different dynamic characteristics. For the D-dimensional landscape, the problem is defined as follows:

$$F(\vec{x}, t) = \max_{i=1,\dots,N} \frac{H_i(t)}{1 + W_i(t) \sum_{j=1}^{D} (x_j(t) - X_{ij}(t))^2} \qquad (17)$$

```
Procedure Proposed PSO-CP for DOPs:
begin
    t := 0
    // Initialize swarm
    for i := 1 to swarm_size do
        for j := 1 to D do
            v_ij := rand[V_min, V_max]; x_ij := rand[X_min, X_max]
        end for
        Evaluate f(x_i)
        p_i := x_i; f(p_i) := f(x_i)
    end for
    p_g := arg max{f(p_i)|i = 1, ···, swarm_size}
              p_i
    f(p_g) := max(f(p_i)); f_Record := f(p_g)
    repeat
        // Detection and response for change
        Evaluate f(p_g)
        if f(p_g) ≠ f_Record then
            for i := 1 to swarm_size do
                Re-evaluate f(p_i)
            end for
        end if
        Construct composite particles (C, N_c)
        for k := 1 to N_c do
            if D_k < θ then      // Scattering operation
                x_k^1 := (1 + φ)x_{k_fittest} − φx_k^1
                x_k^2 := (1 + φ)x_{k_fittest} − φx_k^2
            end if
            // VAR operation
            x_k^R := x_k^M − R_step γ⃗ × (x_{k_worst} − x_k^M)
            Evaluate f(x_k^R)
            if f(x_k^R) > f(x_{k_worst}) then x_k^worst := x_k^R end if
            x_{k_pioneer} := selectBestElementary(C_k)
        end for
        // Update p_i and p_g
        for i := 1 to swarm_size do
            if f(x_i) > f(p_i) then p_i := x_i; f(p_i) := f(x_i) end if
        end for
        S_PI = {x_{k_pioneer}|k = 1, 2, ..., N_C}∪
                {x_{n_indep}|n = 1, 2, ..., swarm_size − 3N_C}
        for each particle i ∈ S_PI do
            v_i(t + 1) := ωv_i(t) + c_1 ξ⃗ × (p_i(t) − x_i(t))
                         +c_2 η⃗ × (p_g(t) − x_i(t))
            x_i(t + 1) := x_i(t) + v_i(t + 1)
        end for
        for each composite particle in C do
            Calculate the distance that the pioneer particle has moved
                after update
            Move the other two elementary particles by the same
                distance regarding each dimension
        end for
        Update p_i and p_g in S
    end repeat
end
```

Fig. 5. Pseudocode of the proposed PSO-CP for DOPs, where S is the whole swarm with size $swarm\_size$, and C is the set of composite particles with size N_C.

where $H_i(t)$ and $W_i(t)$ are the height and the width of peak i at time t, respectively, and $X_{ij}(t)$ is the jth element of the location of peak i at time t. The N independently specified peaks are blended together by the $\max$ function.

In this paper, the dynamism of changes is designed based on the MPB scenario 2 [5], with the height of peaks shifting randomly in the range [30, 70] and the width of peaks shifting randomly within [1, 12]. The position of each peak is shifted by a vector $\vec{v_i}$ of a distance S, and the change of a single peak can be described as follows:

$$\vec{v_i}(t) = \frac{S}{|\vec{r} + \vec{v_i}(t-1)|} ((1-\lambda)\vec{r} + \lambda\vec{v_i}(t-1)) \qquad (18)$$

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

8   IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART B: CYBERNETICS

TABLE I
OFFLINE ERROR AND STANDARD ERROR OF PSO-CP WITH DIFFERENT $swarm\_size$ ON DOPs
WITH $E = 5000$ AND DIFFERENT ENVIRONMENTAL DYNAMICS AND COMPLEXITIES

D, N	$S = 1.0$		$S = 2.0$		$S = 5.0$	
	$swarm\_size = 50$	$swarm\_size = 100$	$swarm\_size = 50$	$swarm\_size = 100$	$swarm\_size = 50$	$swarm\_size = 100$
2, 1	**0.21±0.03**	0.31±0.02	**0.34±0.04**	0.53±0.03	**1.01±0.04**	1.24±0.03
2, 10	**0.62±0.04**	0.67±0.03	0.92±0.03	**0.72±0.02**	1.61±0.06	**1.32±0.04**
2, 100	0.58±0.04	**0.56±0.04**	1.73±0.05	**1.01±0.03**	1.71±0.07	**1.07±0.04**
5, 1	**3.02±0.07**	3.41±0.06	5.02±0.06	**4.12±0.04**	**5.04±0.11**	5.32±0.09
5, 10	**1.03±0.10**	1.31±0.06	2.43±0.07	**1.98±0.06**	3.42±0.08	**3.20±0.13**
5, 100	2.72±0.11	**2.04±0.07**	3.08±0.09	**2.08±0.08**	3.13±0.06	**3.06±0.05**
10, 1	**4.01±0.11**	4.04±0.07	5.62±0.10	**5.12±0.07**	7.21±0.12	**6.13±0.08**
10, 10	3.52±0.13	**2.12±0.08**	**3.05±0.09**	3.37±0.08	4.52±0.07	4.62±0.06
10, 100	3.17±0.11	**2.62±0.09**	3.42±0.09	**2.94±0.06**	4.09±0.06	**3.72±0.06**

where the shift vector $\vec{v}_i(t)$ is a linear combination of a random vector $\vec{r}$ and the previous shift vector $\vec{v}_i(t - 1)$ and is normalized to the length S, which determines the severity of the problem dynamics. The correlated parameter λ is set to 0, which implies that the peak movements are uncorrelated.

B. Experimental Design

Three sets of experiments were carried out to investigate the performance of PSO-CP in dynamic environments. In the first preliminary set of experiments, the effects of key parameters and critical approaches on the performance of PSO-CP were analyzed on different dynamic functions. In addition, a guideline for setting the new introduced parameters (i.e., R_{step}, θ) was provided, with an experimental verification for them to be robust within a range for different dynamic instances. Second, experiments were carried out to compare the performance of PSO-CPs, including the variant proposed in our previous work [27], called PSO-CP$_r$, where the composite particle is being created with a fixed length of L ($L = 0.01 \times (X_{\max} - X_{\min})$) without the scattering operation. We also studied PSO-CP that keeps composite particles for more than one iteration, called PSO-CP$_c$, with a parameter of c representing the construction period of composite particles. In the major set of experiments, PSO-CP was compared with four other PSO models in dynamic environments with different complexities and dynamics of the fitness landscape.

The four contender PSO algorithms are described as follows. The first PSO model is the restart PSO (RPSO), which reinitializes all particles when a change is detected. We consider RPSO to be a straightforward method to response to dynamic environments [18]. The second algorithm is the species-based with particle diversity method (SPSO-PD) studied in [28], which has been proved to be an effective niching PSO model. The third contender PSO model is the MQSO model presented in [3], which can produce competitive results in dynamic environments. Within MQSO, the configuration of $10(5 + 5^q)$ that performs the best was used. The fourth PSO model is the CESO algorithm proposed in [30], which is another state-of-the-art PSO model for DOPs.

For each PSO variant, the learning factors $c_1 = c_2 = 2.05$ and the inertia weight $\omega = 0.729844$ were applied as suggested in [35]. The initial positions of particles were generated randomly in the range of $[X_{\min}, X_{\max}] = [0, 100]$. The total swarm size was set to 100. For PSO-CP, the adaptive method with the initial reflection step size R_{step} in (12) set to 6 and the initial diversity threshold θ in (15) set to 3 was adopted.

For the contender PSO models, other relevant parameters that were recommended by corresponding authors were also applied here.

For each experiment of an algorithm on a test problem, 30 independent runs were executed with the same set of random seeds. For each run of an algorithm on a DOP, ten environmental changes were allowed. For the purpose of evaluating the ability of algorithms to track the optima, the offline error, which was defined by Branke [7] and has been extensively used as a major criterion for DOPs in various studies [21], [35], [46], was used as the primary performance measure. It is the average over, at every point in time, the error of the best solution found since the last change of the environment. This measure is always greater than or equal to zero, and it would be zero if the tracking is perfect.

C. Parameter Analysis for PSO-CP in Dynamic Environments

1) Effect of the Swarm Size: The aim of this set of experiments is to test the effect of the swarm size on the performance of PSO-CP in dynamic environments with different shift severities and complexities. Experiments were carried out with the value of the swarm size set to 50 and 100. The complexities of the fitness landscape were the combination of the dimensionality $D \in \{2, 5, 10\}$ and the number of peaks $N \in \{1, 10, 100\}$. The dynamics of the environment were specified by the value of $S \in \{1.0, 2.0, 5.0\}$ and the number of evaluations between changes, which was set to $E = 5000$. The experimental results are given in Table I.

From Table I, it can be seen that the swarm size 100 gives a better result in most cases. For $S = 1.0$ with a single peak, the better results are obtained when the swarm size is set to 50. The standard errors are smaller for the swarm size 100 than for the swarm size 50 in all investigated functions.

2) Effect of the VAR Scheme: As mentioned in Section IV, the VAR scheme has a potential to drive composite particles to exploit solutions in a promising region. Hence, it is expected that the VAR scheme could lead to a better tracking behavior in dynamic environments. To examine the effect of the VAR scheme on the behavior of PSO-CP under different severities, experiments were carried out with four PSO-CP variants, namely, PSO-CP with the VAR scheme with a fixed value of $R_{\text{step}} = 6.0$, PSO-CP with the adaptive VAR scheme with an initial value of $R_{\text{step}} = 6.0$, PSO-CP with a reflection vector that is unrelated to the velocity in each dimension (i.e., all components in the VAR vector are randomly generated in the range [0.0, 1.0]), and PSO-CP without any reflection scheme. The experimental results are given in Table II.

TABLE II
OFFLINE ERROR AND STANDARD ERROR OF PSO-CP WITH DIFFERENT VAR STRATEGIES AND WITHOUT VAR ON DOPs WITH $E = 5000$ AND DIFFERENT ENVIRONMENTAL DYNAMICS AND COMPLEXITIES

D, N	adaptive VAR	with fixed parameter	random vector	no reflection
	$S = 1.0$			
5, 1	**3.41±0.06**	3.82±0.06	4.74±0.05	19.94±0.18
5, 10	**1.31±0.06**	1.46±0.08	2.36±0.09	10.84±0.22
5,100	**2.04±0.07**	2.41±0.08	2.47±0.11	11.25±0.31
	$S = 2.0$			
5, 1	**4.12±0.06**	4.25±0.06	4.95±0.07	23.43±0.35
5, 10	**1.98±0.06**	2.05±0.07	2.49±0.08	14.92±0.41
5, 100	**2.08±0.08**	2.94±0.08	2.98±0.08	15.75±0.47
	$S = 5.0$			
5, 1	**5.32±0.09**	5.95±0.09	5.87±0.12	30.76±0.49
5, 10	**3.20±0.13**	3.46±0.15	3.21±0.14	18.92±0.51
5, 100	**3.06±0.05**	3.14±0.17	3.09±0.12	28.77±0.67

TABLE III
OFFLINE ERROR AND STANDARD ERROR OF PSO-CP WITH DIFFERENT R_{step} ON DOPs WITH DIFFERENT LANDSCAPES AND DIFFERENT ENVIRONMENTAL DYNAMICS

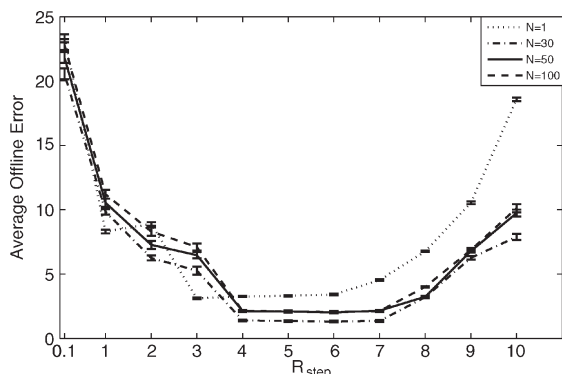| R_{step}	4.0	5.0	5.5	6.0	6.5	7.0
D, N			$S = 1.0$			
2,1	**0.17±0.05**	0.19±0.02	0.20±0.02	0.31±0.02	0.34±0.03	0.35±0.02
2,10	0.75±0.04	0.74±0.04	0.69±0.03	**0.67±0.03**	0.71±0.04	0.73±0.04
2,100	0.67±0.04	0.63±0.05	0.59±0.03	**0.56±0.04**	0.61±0.05	0.64±0.06
5,1	**3.26±0.05**	3.31±0.06	3.35±0.06	3.41±0.06	3.47±0.05	3.53±0.06
5,10	1.40±0.06	1.35±0.04	1.32±0.05	**1.31±0.06**	1.34±0.04	1.36±0.05
5,100	2.14±0.05	2.11±0.04	2.08±0.04	**2.04±0.07**	2.07±0.04	2.13±0.05
10,1	**3.78±0.04**	3.84±0.05	3.92±0.05	4.04±0.07	4.25±0.05	4.36±0.06
10,10	2.19±0.05	2.17±0.06	2.14±0.05	**2.12±0.08**	2.14±0.08	2.16±0.08
10,100	2.68±0.07	2.65±0.07	2.63±0.06	**2.62±0.09**	2.65±0.06	2.67±0.08
D, N			$S = 2.0$			
2,1	0.49±0.02	**0.48±0.02**	0.51±0.03	0.53±0.03	0.54±0.04	0.59±0.03
2,10	0.78±0.05	0.76±0.04	0.74±0.03	**0.72±0.02**	0.75±0.04	0.77±0.04
2,100	1.07±0.04	1.04±0.04	1.02±0.03	**1.01±0.03**	1.02±0.05	1.03±0.04
5,1	4.11±0.05	**4.08±0.05**	4.10±0.05	4.12±0.04	4.14±0.06	4.15±0.06
5,10	2.11±0.06	2.04±0.05	1.99±0.05	**1.98±0.06**	2.03±0.07	2.07±0.08
5,100	2.05±0.09	2.03±0.08	**2.01±0.08**	2.08±0.08	2.04±0.07	2.07±0.09
10,1	5.11±0.08	**5.08±0.06**	5.09±0.08	5.12±0.07	5.16±0.07	5.18±0.09
10,10	3.48±0.08	3.44±0.08	3.42±0.07	**3.37±0.08**	3.41±0.07	3.43±0.08
10,100	3.11±0.09	3.09±0.08	3.04±0.07	**2.94±0.06**	2.97±0.07	3.01±0.08
D, N			$S = 5.0$			
2,1	1.34±0.05	1.32±0.03	1.26±0.04	**1.24±0.03**	1.28±0.04	1.31±0.05
2,10	1.39±0.05	1.35±0.04	1.33±0.05	**1.32±0.04**	1.37±0.06	1.42±0.07
2,100	1.12±0.04	1.11±0.03	1.09±0.18	**1.07±0.04**	1.14±0.03	1.21±0.04
5,1	5.25±0.07	**5.23±0.06**	5.28±0.07	5.32±0.09	5.34±0.07	5.36±0.07
5,10	3.31±0.10	3.25±0.09	3.24±0.11	**3.20±0.13**	3.27±0.13	3.32±0.12
5,100	3.18±0.08	3.14±0.06	3.09±0.08	**3.06±0.05**	3.12±0.08	3. 16±0.09
10,1	6.33±0.11	6.23±0.09	6.18±0.09	**6.13±0.08**	6.24±0.09	6.32±0.10
10,10	4.59±0.07	4.57±0.08	4.56±0.11	**4.52±0.06**	4.64±0.09	4.66±0.10
10,100	3.85±0.08	3.83±0.08	3.79±0.18	**3.72±0.06**	3.78±0.09	3.82±0.09

Fig. 6. Offline error of PSO-CP algorithms with different R_{step} on DOPs with $E = 5000$, $S = 1.0$, $D = 5$, and a different number of peaks.

From Table II, it can be seen that the adaptive VAR scheme applied in this paper gives a better result for all cases. This indicates that the adaptive method can enhance the ability of PSO-CP for DOPs. Comparing with the other two variants (i.e., without any reflection and reflection with a random vector), PSO-CP with a fixed VAR parameter performs better for moderate severity cases ($S = 1.0$ and $S = 2.0$), and the reflection scheme without referring to velocities performs better than the VAR scheme when the dynamism is severe ($S = 5.0$). The reason lies in the fact that, when the change is smooth, utilizing the velocity information of particles, which can reflect the degree of swarm convergence to some extent, may lead to a better exploitation for optima tracking. However, the ignorance of the convergence degree under this environment may weaken the track ability when the new optima are close to the ones in the last environment. When the environment suffers severe changes (for example, if $S = 5.0$), it seems that particles evenly distributed in the search space are more helpful for a better exploration of the landscape. This stresses again that the adaptive VAR scheme with proper parameters can contribute in accelerating the searching speed in dynamic environments.

Given the above discussion, it is evident that the initial value of the reflection step size plays a vital role in the performance of PSO-CP in dynamic environments. To validate this, experiments were further carried out with the value of R_{step} set in the range [0.1, 10]. The results are shown in Fig. 6.

From Fig. 6, it can be seen that the initial value of R_{step} around 4.0 gives a better result when tracking a single peak.

This indicates that a local optimizer that can produce a small-scale search is beneficial for tracking mild shifts. However, when there are more than ten peaks, a step size around 6.0 becomes more suitable, which indicates that it is important to sustain the local diversity in composite particles for a rapid response to changes in multimodal landscapes. PSO-CP performs the worst in extreme cases of $R_{step} = 0.1$ and $R_{step} = 10$. When $R_{step} = 0.1$, PSO-CP suffers from the diversity loss, which leaves the entire swarm ill-adapted to a new environment. When $R_{step} = 10$, many invalid reflections that are produced do not contribute to optima tracking but lead to a rapid convergence.

Furthermore, we are interested in investigating the range in which the parameter of R_{step} is robust on different dynamic instances. Based on the above analyses, the ability of PSO-CP with R_{step} set in the range of [4.0, 7.0] was provided in Table III. Table III shows that the offline error of PSO-CP is less affected by R_{step} in this range in the investigated dynamic environments. Again, the value around 6.0 produces the best performance for dynamic multimodal optimization problems. One reason lies in the fact that the VAR scheme with R_{step} around 6.0 helps maintaining the diversity at a reasonable level, which contributes to more efficient tracking as compared with the other settings. Another observation is that PSO-CP with $R_{step} = 7.0$ shows a weaker adaptation due to the invalid reflections. R_{step} around 5.0 provides better tracking performance in the dynamic environments with relatively mild changes (i.e., $S = 1.0$ and $S = 2.0$).

3) Effect of the Local Diversity Threshold (θ): The parameter θ determines when the scattering operation is invoked. It has a significant influence on the ability of PSO-CP to address dynamic environments. With the adaptive method, a large initial value of θ may weaken the exploitation ability of composite particles to track the nearby optima. In contrast, a small initial

TABLE IV
OFFLINE ERROR AND STANDARD ERROR OF PSO-CP WITH DIFFERENT θ WITH AND WITHOUT ADAPTIVE METHODS (DENOTED AS ADA AND FIX) ON DOPs WITH $D = 10$, $N = 5$, AND DIFFERENT ENVIRONMENTAL DYNAMICS

θ		0.1	1	2	3	5	10^3
E				$S = 1.0$			
3000	Ada	2.77±0.08	2.14±0.05	**2.12±0.07**	2.17±0.06	2.29±0.07	2.32±0.08
	Fix	2.98±0.09	2.31±0.04	2.46±0.03	2.48±0.06	2.49±0.06	2.57±0.06
5000	Ada	1.99±0.06	1.42±0.04	1.40±0.03	**1.31±0.06**	1.59±0.06	1.84±0.06
	Fix	2.09±0.06	1.51±0.05	1.46±0.05	1.48±0.07	1.52±0.06	1.76±0.08
E				$S = 2.0$			
3000	Ada	3.83±0.11	2.29±0.06	**2.23±0.08**	2.19±0.08	2.20±0.11	3.67±0.09
	Fix	3.95±0.06	3.01±0.05	3.17±0.04	3.21±0.06	3.25±0.05	3.92±0.04
5000	Ada	3.72±0.09	3.04±0.07	2.09±0.08	**1.98±0.06**	2.14±0.06	2.24±0.07
	Fix	3.55±0.07	2.19±0.05	2.05±0.04	2.12±0.06	2.23±0.06	2.37±0.05
E				$S = 5.0$			
3000	Ada	6.21±0.16	4.05±0.15	**3.63±0.18**	3.77±0.17	3.68±0.12	4.82±0.16
	Fix	6.81±0.08	4.03±0.05	3.82±0.11	3.85±0.12	3.89±0.10	4.95±0.12
5000	Ada	5.83±0.14	3.45±0.12	3.42±0.11	**3.20±0.13**	3.62±0.14	3.97±0.13
	Fix	5.98±0.13	3.45±0.10	3.46±0.12	3.55±0.12	3.73±0.11	4.04±0.11

value of θ may weaken the exploration capacity of particles to cover promising peaks after a change. To test the influence of θ on the performance of PSO-CP, experiments were carried out with θ set to different values in the set $\{0.1, 1, 2, 3, 5, 10^3\}$ with and without the adaptive technique. Since the setting of θ concerns the balance between the exploration and exploitation abilities of particles, we focus on the environmental dynamics with $E = \{3000, 5000\}$, which represents a different degree of swarm convergence before a change occurs, and $S = \{1.0, 2.0, 5.0\}$ in a landscape of ten peaks in a 5-D space. The experimental results are given in Table IV, where "Ada" and "Fix" denote the adaptive and fixed schemes for the reflection step size, respectively.

The results in Table IV show that the adaptive diversity threshold θ can produce a better tracking behavior than the fixed θ in all investigated cases. In addition, the parameter θ should be set in the range of [1.0, 5.0] for better performance of PSO-CP on the test DOPs. It can be observed that, when the update interval is small ($E = 3000$), setting θ around 2.0 gives a better result than other settings. The poor offline error for $\theta = 0.1$ indicates the necessity for the scattering operation. When the update interval increases to $E = 5000$, the swarm becomes more converged before a change occurs. Hence, diversification becomes even more important for tracking peaks in these cases, and it is more suitable to set θ in the range of [3.0, 5.0]. However, when $\theta = 10^3$, which gives the highest local diversity degree among all the settings of θ, PSO-CP performs worse than when θ is set to 3.0 at most cases. This happens because too much diversification is harmful to the convergence process required to search for the optima. This result indicates that a good tradeoff between convergence and diversification is vital for tracking the moving peaks and, hence, is vital for the overall performance of PSO-CP for DOPs.

To investigate the ability of PSO-CP with different θ to track the optima and maintain the diversity level during the whole solving process, different from the entropy-based diversity measurement applied in the adaptive technique, the "distance-to-average-point" measurement for the swarm diversity, as defined in [19], was introduced as follows:

$$\text{diversity}(CS) = \frac{1}{M \times L} \sum_{i=1}^{M} \sqrt{\sum_{d=1}^{D}(x_{id} - \overline{x}_d)^2} \quad (19)$$

where CS is the current swarm with the swarm size M, L is the length of the longest diagonal in the search place, D is the problem dimensionality, and $\overline{x}_d$ is the dth dimensional value of the average point.

The dynamic performance of PSO-CP with $\theta = 3.0$ and two extreme cases of $\theta = 0.1$ and 10^3 regarding the offline error is plotted in Fig. 7. The swarm diversity is plotted against the number of evaluations in Fig. 8 with $E = 5000$. Some observations can be drawn from Figs. 7 and 8. For the static period, the performance is similar for the three settings. With the progress of the optimization process, PSO-CP with $\theta = 0.1$ is likely to be trapped into the local optima, and the offline error increases due to the diversity loss. The offline error for $\theta = 3.0$ is considerably better, demonstrating that a sufficient diversity for efficiently exploring the search space can be gained in this case (see Fig. 8). The performance of PSO-CP with $\theta = 10^3$ is always weaker than that of PSO-CP with $\theta = 3.0$ at most cases (except for the case when $S = 5.0$). This is because it focuses on diversification without exploiting peaks in the promising area. This indicates again that the diversity level of the swarm is a major concern but not the singular one that affects the performance of PSO algorithms in dynamic environments [49]. Other factors, such as the speed of exploiting promising regions in the search space of the current environment, should also be considered to achieve a stronger adaptation for PSO algorithms in dynamic environments.

It is noticeable that setting θ to a large value gives a better result in some environmental periods when $S = 5.0$. This indicates that a high diversity degree of the swarm can enable a better tracking behavior when the dynamic problems are subject to a greater level of dynamism. After several iterations, the offline error of PSO-CP with $\theta = 3.0$ becomes the smallest again. This happens because performing the VAR scheme on composite particles that have a reasonable local diversity degree can promote the tracking ability of particles. Another interesting result is that PSO-CP with $\theta = 0.1$ tends to suffer from the diversity loss particularly when $S = 5.0$. This result again indicates that increasing the particle diversity is more important for high-dynamism environments [3], [27].

It is interesting to compare the diversity level between PSO-CP and RPSO during the running process. Note that RPSO can regain a high-level diversity when a change is detected. Fig. 9 shows the swarm diversity of PSO-CP and RPSO on the DOP with $E = 5000$, $D = 5$, and $S = 1.0$. From Fig. 9, it can be seen that in comparison with RPSO, the diversity level of PSO-CP fluctuates within a range. This happens due to the combination of the VAR scheme and diversification strategies. The former approach utilizes good information to probe promising solutions and results in a faster searching speed in dynamic environments, and the latter approach aims at distributing particles in a more comprehensive area, leading to a better exploration of the search space. Furthermore, considering the comparison between RPSO and PSO-CP in the following section, it seems that, in some environments, the idea of maintaining the diversity during the whole process is more beneficial for rapidly responding to changes than the method of rediversification after a change.

Based on the above discussions, the effect of the adaptive parameter θ on the performance of PSO-CP was tested with the initial value of θ in the range of [1.0, 5.0]. To further see
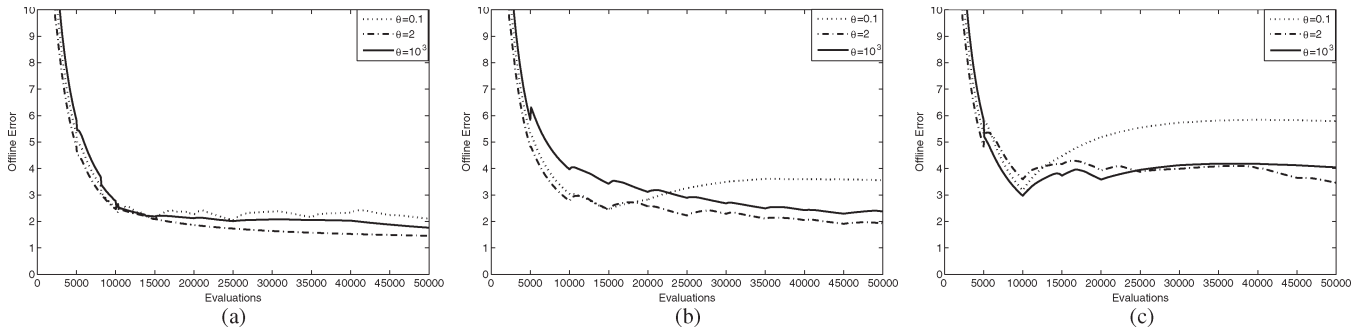
Fig. 7. Dynamic performance of PSO-CP with different θ on the DOPs with $E = 5000$ and different severities of environmental changes: (a) $S = 1.0$, (b) $S = 2.0$, and (c) $S = 5.0$.
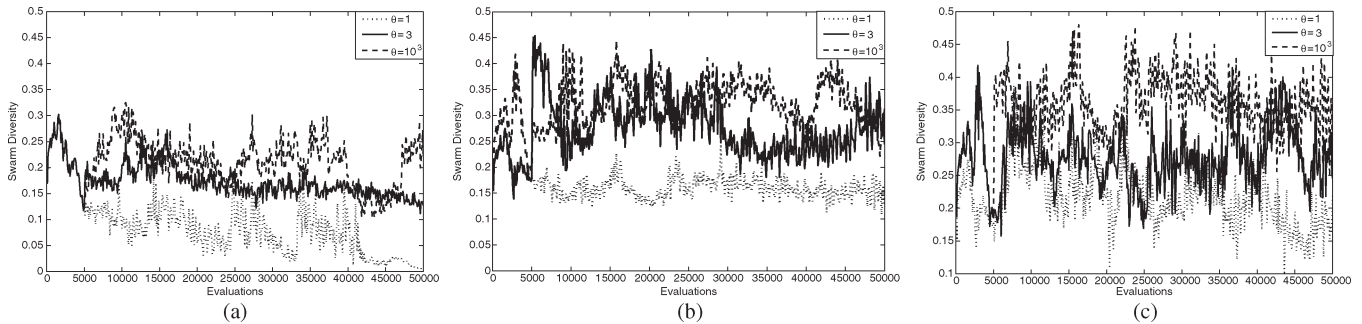

Fig. 8. Swarm diversity of PSO-CP with different θ on the DOPs with $E = 5000$ and different severities of environmental changes: (a) $S = 1.0$, (b) $S = 2.0$, and (c) $S = 5.0$.
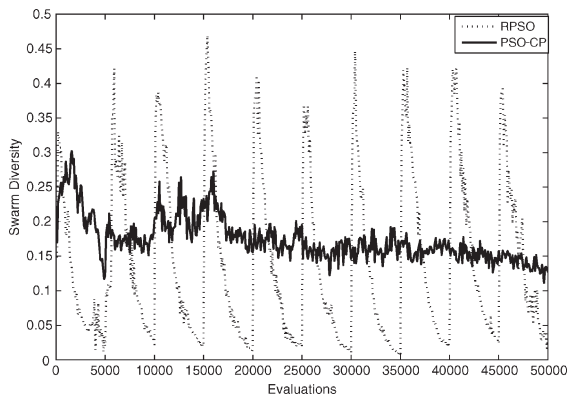

Fig. 9. Swarm diversity for PSO-CP and RPSO on the DOP with $E = 5000$, $S = 1.0$, and $D = 5$.

TABLE V
OFFLINE ERROR AND STANDARD ERROR OF PSO-CP WITH DIFFERENT θ ON DOPs WITH $E = 5000$ AND DIFFERENT FITNESS LANDSCAPES WITH DIFFERENT DIMENSIONALITIES (D) AND NUMBER OF PEAKS (N)

θ	1.0	2.0	2.5	3.0	4.0	5.0
D, N			$S = 1.0$			
5,1	3.43±0.07	3.36±0.06	**3.38±0.05**	3.41±0.06	3.43±0.07	3.45±0.08
5,10	1.41±0.05	1.40±0.06	1.40±0.07	1.31±0.06	**1.30±0.06**	**1.30±0.07**
5,100	2.08±0.06	2.07±0.07	2.06±0.05	**2.04±0.07**	2.05±0.06	2.07±0.07
D, N			$S = 2.0$			
5,1	4.11±0.09	**4.11±0.08**	4.12±0.06	4.12±0.04	4.24±0.08	4.25±0.06
5,10	2.01±0.07	2.00±0.08	2.00±0.08	**1.98±0.06**	1.99±0.08	2.02±0.06
5,100	2.15±0.07	2.12±0.08	2.10±0.08	**2.08±0.08**	2.11±0.06	2.14±0.07
D, N			$S = 5.0$			
5,1	5.39±0.12	5.37±0.11	5.34±0.08	**5.32±0.07**	5.36±0.08	5.38±0.07
5,10	3.25±0.10	3.24±0.12	3.23±0.12	**3.20±0.13**	3.21±0.12	3.26±0.13
5,100	3.15±0.08	3.14±0.08	3.11±0.07	**3.06±0.05**	3.08±0.08	3.12±0.07

the effect of θ on the performance of PSO-CP, we carried out some experiments on PSO-CP on DOPs with different fitness landscape features. Table V presents the experimental results regarding the performance of PSO-CP with different θ on DOPs with $E = 5000$ and different fitness landscapes with different dimensionalities and number of peaks. From Table V, it can be seen that setting θ in the range of [2.0, 5.0] does not greatly affect the performance of PSO-CP in the investigated cases. Hence, we recommend setting θ within this range, which can provide a proper diversity level and a sufficient search power with the adaptive technique.

4) Effect of the Integral Movement: Another strategy introduced in our approach is the integral movement. To study the effect of the integral movement on the performance of PSO-CP in dynamic environments, experiments were carried out on PSO-CP with and without the integral movement. The experimental results regarding the offline error and the swarm diver-

sity against the number of evaluations are plotted in Figs. 10 and 11, respectively.

From Figs. 10 and 11, it can be seen that the integral movement scheme can maintain a sufficient diversity level and, hence, can produce a significant advantage to the performance of PSO-CP. This confirms our expectation that this technique can promote the swarm diversity and, hence, can provide PSO-CP with a better tracking behavior in dynamic environments.

D. Comparing PSO-CP Variants in Dynamic Environments

1) Comparison With $PSO\text{-}CP_r$: The experimental results of comparison between PSO-CP$_r$ [27] and PSO-CP proposed in this study on dynamic problems with different shift severities are provided in Table VI, where the experiments were specified by $E = 5000$ and $S \in \{0, 1.0, 3.0, 6.0\}$ in a landscape of ten peaks in a 5-D solution space. In addition, the statistical test results of comparing PSO-CP with the contender

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

12   IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART B: CYBERNETICS
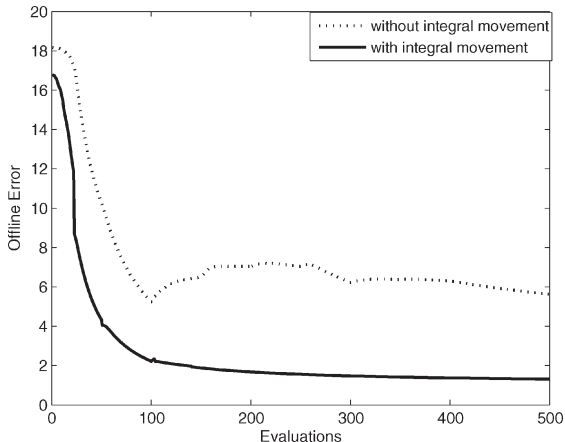

Fig. 10. Dynamic performance of PSO-CP with and without the integral movement scheme and $\theta = 2$ on the DOP with $E = 5000$, $S = 1.0$, $D = 5$, and $N = 10$.
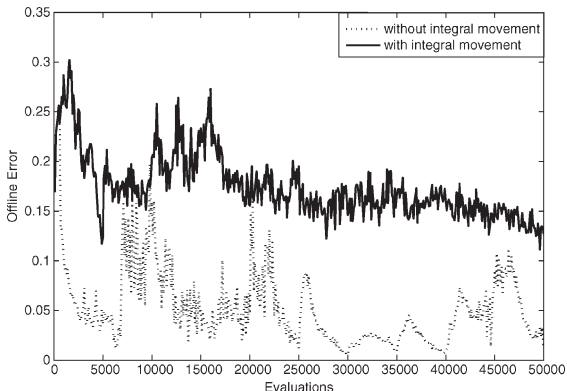

Fig. 11. Swarm diversity of PSO-CP with and without the integral movement scheme and $\theta = 2$ on the DOP with $E = 5000$, $S = 1.0$, $D = 5$, and $N = 10$.

TABLE VI
OFFLINE ERROR AND STANDARD ERROR OF ALGORITHMS ON DOPs WITH $E = 5000$, $D = 5$, AND $N = 10$. THE t-TEST RESULTS OF COMPARING PSO-CP WITH PSO-CP$_r$ ARE SHOWN IN BRACKETS

S	0	1.0	3.0	6.0
PSO-CP	**0.87±0.07**	**1.31±0.06**	**2.21±0.06**	**3.22±0.14**
PSO-CP$_r$	1.15±0.07 (+)	1.65±0.08 (+)	2.72±0.08(+)	4.73±0.11(+)

PSO algorithms in terms of the offline error by the one-tailed t-test with 58 degrees of freedom at a 0.05 level of significance are given in the brackets in Table VI, where the t-test result is shown as "+", "−", or "∼" when PSO-CP is significantly better than, significantly worse than, or statistically equivalent to the corresponding algorithm, respectively.

It can be seen from Table VI that for each fixed S, PSO-CP outperforms PSO-CP$_r$ in all investigated cases. This happens because the construction scheme in PSO-CP improves its adaptivity due to the prompt adjustment based on the combination of fitness and Euclidean distance information obtained from the current fitness landscape. The enhancements such as the detection method and the adaptive scattering operation also give advantages for PSO-CP in dynamic environments.

2) Comparison on DOPs With PSO-CP$_c$: One major issue for PSO-CP is the complexity problem as analyzed above. One straightforward method is to increase the construction period of the composite particles. The experiments were carried out

TABLE VII
OFFLINE ERROR AND STANDARD ERROR OF ALGORITHMS ON DOPs WITH $E = 5000$, $D = 5$, AND $N = 10$. THE t-TEST RESULTS OF COMPARING PSO-CP$_C$ WITH $c = 1$ AND OTHER VALUES ARE SHOWN IN BRACKETS

S	0	1.0	3.0	6.0
c = 1	**0.87±0.07**	**1.31±0.06**	**2.21±0.06**	**3.93±0.14**
c = 2	0.97±0.08(∼)	1.48±0.10(∼)	2.36±0.07(+)	4.42±0.09(+)
c = 5	1.02±0.10(+)	1.51±0.11(+)	2.51±0.07(+)	4.61±0.10(+)
c = 10	1.12±0.09(+)	1.54±0.11(+)	2.60±0.07(+)	4.68±0.11(+)

TABLE VIII
OFFLINE ERROR AND STANDARD ERROR OF ALGORITHMS ON DOPs WITH $E = 5000$, $D = 5$, $N = 10$, AND DIFFERENT SHIFT SEVERITIES. THE t-TEST RESULTS OF COMPARING PSO-CP WITH OTHER PSO ALGORITHMS ARE SHOWN IN BRACKETS

S	RPSO	SPSO-PD	MQSO	CESO	PSO-CP
0	31.84±0.48(+)	1.63±0.05(+)	1.18±0.07(+)	0.95±0.04(+)	**0.87±0.07**
1	32.48±0.46(+)	1.93±0.08(+)	1.75±0.06(+)	1.68±0.05(+)	**1.31±0.06**
2	33.69±0.48(+)	2.25±0.09(+)	2.40±0.06(+)	2.27±0.05(+)	**1.98±0.06**
3	33.97±0.51(+)	2.74±0.09(+)	3.00±0.06(+)	2.64±0.07(+)	**2.21±0.06**
4	34.19±0.52(+)	3.05±0.10(+)	3.59±0.10(+)	3.23±0.07(+)	**2.61±0.11**
5	34.33±0.53(+)	3.24±0.11(+)	4.24±0.10(+)	3.35±0.09(+)	**3.20±0.13**
6	34.96±0.54(+)	4.95±0.13(+)	4.79±0.10(+)	3.94±0.11(∼)	**3.93±0.14**

with c setting to different values in the set $\{1, 2, 5, 10\}$, where c is the construction period of the composite particles, and dynamic environments were consistent with the above settings. The experimental results are shown in Table VII.

It can be seen from Table VII that for PSO-CP$_c$ on all investigated environments, the performance drops heavier and heavier when the value of c increases from 1 to 10. Although the PSO-CP recommended in this paper has the additional overhead of computing, it does present some significant advantages, in particular, when the environmental dynamics are more severe (i.e., $S = 3.0$ and $S = 6.0$). This is because the construction executed for each iteration can rapidly respond to the current fitness landscape. Further investigation is needed on how to balance the complexity and the effectiveness of PSO-CP.

E. Major Experimental Results and Analysis

In this major set of experiments, the performance of PSO-CP in dynamic environments was compared with the four contender PSO algorithms, i.e., RPSO, SPSO-PD, MQSO, and CESO. All corresponding parameters were fixed to be the same values for investigated DOPs.

1) Comparison on DOPs With Different Dynamics: These experiments investigate the performance comparison between the four PSO algorithms on DOPs with different dynamics. When considering the effect of different shift severities on the performance of PSO algorithms, experiments were specified by $E = 5000$ and $S \in \{0, 1.0, 2.0, 3.0, 4.0, 5.0, 6.0\}$. When considering the effect of a different number of evaluations between changes on the performance of PSO algorithms, experiments were specified by $S = 1.0$ and $E \in \{3000, 5000, 10000\}$. For both cases, the landscape dimensionality is set to $D = 5$, and the number of peaks is set to $N = 10$. The results are given in Tables VIII and IX, respectively.

From Tables VIII and IX, several results can be observed and are analyzed as follows. First, it can be seen that increasing the value of S increases the difficulty for all PSO algorithms in tracking the global optima, and an increasing interval between changes can encourage better tracking in

TABLE IX
OFFLINE ERROR AND STANDARD ERROR OF ALGORITHMS ON DOPs WITH $S = 1.0$, $D = 5$, $N = 10$, AND A DIFFERENT NUMBER OF EVALUATIONS BETWEEN CHANGES. THE t-TEST RESULTS OF COMPARING PSO-CP WITH CONTENDER PSO ALGORITHMS ARE SHOWN IN BRACKETS

E	RPSO	SPSO-PD	MQSO	CESO	PSO-CP
3000	35.25±0.54(+)	2.48±0.09(+)	2.26±0.09(+)	2.18±0.08(∼)	**2.17±0.08**
5000	33.69±0.48(+)	1.63±0.08(+)	1.75±0.06(+)	1.68±0.05(+)	**1.31±0.06**
10000	31.54±0.33(+)	1.12±0.03(+)	1.07±0.03 (+)	1.09±0.02(+)	**0.97±0.03**

TABLE X
OFFLINE ERROR AND STANDARD ERROR OF ALGORITHMS ON DOPs WITH $E = 5000$, $S = 1.0$, $D = 5$, AND A DIFFERENT NUMBER OF PEAKS. THE t-TEST RESULTS OF COMPARING PSO-CP WITH CONTENDER PSO ALGORITHMS ARE SHOWN IN BRACKETS

N	RPSO	SPSO-PD	MQSO	CESO	PSO-CP
1	31.42±0.02(+)	5.02±0.08(+)	5.07±0.17(+)	4.93±0.07(+)	**3.41±0.06**
10	33.69±0.48(+)	1.63±0.08(+)	1.75±0.06(+)	1.68±0.05(+)	**1.31±0.06**
30	34.32±0.52(+)	2.23±0.08(−)	2.48±0.07(+)	2.15±0.06(+)	**2.02±0.07**
50	34.93±0.49(+)	2.48±0.07(−)	2.50±0.06(+)	2.26±0.06(+)	**2.14±0.08**
100	32.83±0.21(+)	2.54±0.06(+)	2.36±0.04(+)	2.18±0.08(+)	**2.04±0.07**

TABLE XI
OFFLINE ERROR AND STANDARD ERROR OF ALGORITHMS ON DOPs WITH $E = 5000$, $S = 1.0$, $N = 10$, AND DIFFERENT DIMENSIONALITIES. THE t-TEST RESULTS OF COMPARING PSO-CP WITH CONTENDER PSO ALGORITHMS ARE SHOWN IN BRACKETS

D	RPSO	SPSO-PD	MQSO	CESO	PSO-CP
5	32.48±0.46(+)	1.93±0.08(+)	1.75±0.06(+)	1.68±0.05(+)	**1.31±0.06**
10	62.73±0.92(+)	2.36±0.07(+)	2.38±0.08(+)	2.92±0.04(+)	**2.12±0.08**
50	66.75±0.98(+)	20.61±0.32(+)	20.84±0.35(+)	24.80±0.25(+)	**18.25±0.32**

TABLE XII
COMPARISON OFFLINE ERROR AND STANDARD ERROR OF ALGORITHMS ON DOPs WITH $E = 5000$, $S = 1.0$, $N = 10$, AND DIFFERENT CORRELATION OF THE SHIFT PARAMETER. THE t-TEST RESULTS OF COMPARING PSO-CP WITH CONTENDER PSO ALGORITHMS ARE SHOWN IN BRACKETS

λ	RPSO	SPSO-PD	MQSO	CESO	PSO-CP
			$S = 1.0$		
0	32.48±0.46(+)	1.63±0.08(+)	1.75±0.06(+)	1.68±0.05(+)	**1.31±0.06**
0.1	30.67±0.48(+)	1.95±0.08(+)	1.86±0.05(+)	1.73±0.06(+)	**1.32±0.05**
0.3	30.85±0.59(+)	1.96±0.09(+)	1.93±0.06(+)	1.75±0.06(+)	**1.35±0.05**
0.5	31.05±0.53(+)	1.98±0.06(+)	1.96±0.05(+)	1.78±0.07(+)	**1.47±0.06**
0.7	32.74±0.54(+)	2.02±0.07(+)	2.01±0.07(+)	1.83±0.06(+)	**1.42±0.04**
1.0	33.86±0.55(+)	2.03±0.09(+)	2.09±0.06(+)	1.85±0.07(+)	**1.44±0.06**
λ			$S = 3.0$		
0	33.97±0.51(+)	2.74±0.09(+)	3.00±0.06(+)	2.64±0.07(+)	**2.21±0.06**
0.1	34.02±0.58(+)	2.79±0.08(+)	3.03±0.06(+)	2.75±0.08(+)	**2.23±0.06**
0.3	34.18±0.59(+)	2.82±0.07(+)	3.05±0.06(+)	2.79±0.08(+)	**2.27±0.07**
0.5	34.12±0.61(+)	2.85±0.09(+)	3.06±0.07(+)	2.89±0.09(+)	**2.28±0.08**
0.7	34.35±0.64(+)	2.89±0.09(+)	3.10±0.07(+)	2.92±0.09(+)	**2.29±0.06**
1.0	34.47±0.65(+)	2.92±0.10(+)	3.13±0.08(+)	2.95±0.09(+)	**2.30±0.07**
λ			$S = 6.0$		
0	34.96±0.54(+)	4.95±0.13(+)	4.79±0.10(+)	3.94±0.11(∼)	**3.93±0.14**
0.1	35.04±0.59(+)	5.05±0.16(+)	4.85±0.11(+)	4.61±0.12(+)	**4.22±0.14**
0.3	36.31±0.63(+)	5.12±0.15(+)	4.97±0.12(+)	4.53±0.16(+)	**4.41±0.13**
0.5	36.89±0.68(+)	5.13±0.16(+)	5.03±0.12(+)	4.57±0.16(+)	**4.47±0.14**
0.7	37.95±0.73(+)	5.19±0.17(+)	5.09±0.13(+)	4.62±0.18(+)	**4.50±0.14**
1.0	38.64±0.79(+)	5.24±0.17(+)	5.13±0.14(+)	4.67±0.17(+)	**4.51±0.15**

dynamic environments. All SPSO-PD, MQSO, CESO, and PSO-CP significantly outperform RPSO on the DOPs with different environmental dynamics. This indicates that the idea of altering some old solutions with specific interaction mechanisms is helpful in tracking optima in dynamic environments.

Second, another significant result is that PSO-CP outperforms other PSO algorithms on most DOPs, and PSO-CP shows significantly better performance in most cases. This result validates the efficiency of introducing composite particles along with relevant strategies into PSO-CP. The movement strategy can integrate valuable information efficiently, and the VAR scheme has an intensive exploitation ability, which helps composite particles search for optima continuously rather than converging into a solution ahead. The advantage of these schemes results in significantly better results, particularly when the environment suffers from moderate changes.

Third, PSO-CP is statically equivalent to CESO in some cases (for example, when $E = 3000$ and $S = 1.0$, or $E = 5000$ and $S = 6.0$). The reason lies in the fact that the VAR scheme does not fully exhibit its effect when the environment is subject to intensive changes.

2) Comparison on DOPs With a Different Number of Peaks: The performance of PSO algorithms was also investigated on DOPs with a different number of peaks $N \in \{1, 10, 30, 50, 100\}$ and other dynamic and complexity parameters fixed to $E = 5000$, $S = 1.0$, and $D = 5$. The experimental results, including the t-test results of comparing PSO-CP with the contender PSO algorithms, are shown in Table X.

From Table X, it can be seen that PSO-CP outperforms other PSO variants in all investigated cases. This is because the VAR scheme offers a significant advantage in tracking optima, and the integral movement strategy can slow down the process of being trapped and push the searching toward the global optima. The superiority of PSO-CP on DOPs with a large number of peaks indicates that although it is not specially designed for locating and tracking multiple optima simultaneously, the adaptive method also helps in rapidly responding to changes when the distance between peaks is small.

3) Comparison on DOPs With Different Dimensionalities: The performance of the five PSO algorithms was investigated

on the DOPs with different dimensionalities $D \in \{5, 10, 50\}$ and other dynamic and complexity parameters set as follows: $E = 5000$, $S = 1.0$, and $N = 10$. The experimental results regarding the offline error and the t-test of comparing PSO-CP with the contender PSO algorithms are shown in Table XI.

From Table XI it can be seen that PSO-CP performs better than other algorithms on dynamic landscapes with a fixed number of peaks. Based on above discussions, one major reason lies in that the adaptive VAR scheme can encourage the swarm to extensively explore the multidimensional search space.

4) Comparison on DOPs With Different Correlations of the Shift Parameter: Experiments were also carried out to investigate the performance of the five PSO algorithms with different settings of the correlated parameter λ, which represents for the correlation between consecutive movements of a single peak [5]. The effect of setting the correlation parameter $\lambda \in \{0, 0.1, 0.3, 0.5, 0.7, 1\}$ with different shift severities $S \in \{1.0, 3.0, 6.0\}$ on the performance of algorithms is provided in Table XII.

It can be observed that PSO-CP significantly outperforms the other three contender algorithms on all dynamic instances. For $\lambda = 0$, in which the movement is completely uncorrelated in more severe cases, PSO-CP performs statically equivalent to CESO (i.e., $S = 6.0$). However, with the increase of the value of λ, PSO-CP always beats other contender algorithms. This happens because under dynamic environments with correlated changes, the adaptive VAR scheme in PSO-CP contributes to its adaptivity over cases with correlated changes. The VAR scheme

among the near neighbors brings in an effective integration of valuable information in the previous and current environments. This benefit leads to the better performance of PSO-CP over other algorithms.

VI. CONCLUSION AND FUTURE WORK

This paper has presented a new PSO model, called PSO-CP, based on the phenomenon of composite particles in physics for addressing DOPs. The basic motivation behind PSO-CP is to utilize an efficient integrated and interactive mechanism to track promising peaks in dynamic environments. A new kind of composite particles is introduced into PSO. These composite particles have a simple geometrical shape: The elementary particles form a triangle. This model uses a "worst first" principle and spatial information to construct composite particles for highly efficient information sharing. Composite particles are operated through the following operators.

- VAR scheme. It is important to enable particles to continuously track the moving optima over time to address DOPs. Hence, we focus on how to fully exhibit valuable information from all the particles, whether they are fit or not. The VAR scheme with an adaptive method improves the less-fit elementary particles by fitter ones that are close to it in a specialized way. It ensures an exploitative ability for promising regions, thereby driving particles to track optima in dynamic environments.
- Scattering operation. The diversity within each composite particle can encourage elementary particles to track new peaks. The self-adaptive scattering operation helps maintain the local diversity and, hence, guarantees the swarm diversity to watch over new optima, as well as to explore potentially more promising regions.
- Integral movement strategy. The swarm diversity is crucial for a better exploration to cover new peaks in the search space. The integral movement strategy aims to prevent the collision of composite particles, which improves less-fit elementary particles toward the fittest one that is updated according to the standard rule of the classic PSO algorithm.

To justify the proposed PSO-CP, experiments were carried out to compare the performance of PSO-CP with a number of state-of-the-art PSO algorithms on dynamic test problems. From the experimental results, several conclusions can be drawn on the dynamic test environments. First, the introduction of composite particles with their specific interaction form is beneficial for the performance of PSO in dynamic environments. Second, the proposed VAR scheme is efficient to improve the performance of PSO-CP, particularly for multidimensional DOPs. Third, the scattering operator works well for PSO-CP to adapt to a new environment. Fourth, the strategy of extracting information from other particles besides the best solutions as in the classic PSO algorithm is a good choice for preserving valuable information and avoiding convergence to improve the performance of PSO in dynamic environments.

Generally speaking, the experimental results indicate that the proposed PSO-CP can be a good optimizer in dynamic environments.

For future work, it would be valuable to try to simplify the PSO-CP model to release the effect of setting the involved parameters and enhance the robustness of PSO-CP for more complex DOPs. A deep investigation for decreasing the complexity of the algorithm needs to be considered. It is also worthy discussing the proposed PSO-CP from the view of theory, including analyzing the VAR scheme by probability. Finally, the comparison with EAs, such as differential evolution methods, and genetic algorithms is under investigation.

REFERENCES

[1] T. M. Blackwell and P. J. Bentley, "Dynamic search with charged swarms," in *Proc. Genetic Evol. Comput. Conf.*, 2002, pp. 19–26.
[2] T. M. Blackwell, "Particle swarms and population diversity I: Analysis," in *Proc. GECCO Workshop Evol. Algorithms Dyn. Optim. Probl.*, 2003, pp. 9–13.
[3] T. M. Blackwell and J. Branke, "Multiswarms, exclusion, and anticonvergence in dynamic environments," *IEEE Trans. Evol. Comput.*, vol. 10, no. 4, pp. 459–472, Aug. 2006.
[4] E. Bonabeau, M. Dorigo, and G. Theraulaz, *Swarm Intelligence: From Natural to Artificial Systems*. New York: Oxford Univ. Press, 1999.
[5] J. Branke, "Memory enhanced evolutionary algorithms for changing optimization problems," in *Proc. Congr. Evol. Comput.*, 1999, vol. 3, pp. 1875–1882.
[6] J. Branke, T. Kauler, C. Schmidth, and H. Schmeck, "A multi-population approach to dynamic optimization problems," in *Proc. 4th Int. Conf. Adapt. Comput. Des. Manuf.*, 2000, pp. 299–308.
[7] J. Branke, *Evolutionary Optimization in Dynamic Environments*. Norwell, MA: Kluwer, 2002.
[8] A. Carlisle and G. Dozier, "Adapting particle swarm optimization to dynamic environments," in *Proc. Int. Conf. Artif. Intell.*, 2000, pp. 429–434.
[9] D. B. Chen and C. X. Zhao, "Particle swarm optimization with adaptive population size and its application," *Appl. Soft Comput.*, vol. 9, no. 1, pp. 39–48, Jan. 2009.
[10] H. G. Cobb and J. J. Grefenstette, "Genetic algorithms for tracking changing environments," in *Proc. 5th Int. Conf. Genetic Algorithms*, 1993, pp. 523–530.
[11] R. C. Eberhart and Y. Shi, "Tracking and optimizing dynamic systems with particle swarms," in *Proc. Congr. Evol. Comput.*, 2001, pp. 94–100.
[12] S. C. Esquivel and C. A. Coello, "Hybrid particle swarm optimizer for a class of dynamic fitness landscape," *Eng. Optim.*, vol. 38, no. 8, pp. 873–888, Dec. 2006.
[13] F. S. Galasso, *Structure and Properties of Inorganic Solids*. New York: Pergamon, 1970.
[14] J. J. Grefenstette, "Genetic algorithms for changing environments," in *Proc. 2nd Int. Conf. Parallel Problem Solving Nature*, 1992, pp. 137–144.
[15] M. Greeff and A. P. Engelbrecht, "Solving dynamic multi-objective problems with vector evaluated particle swarm optimisation," in *Proc. Congr. Evol. Comput.*, 2008, pp. 2922–2929.
[16] M. Guntsch and M. Middendorf, "Applying population based ACO to dynamic optimization problems," in *Proc. 3rd Int. Workshop ANTS*, vol. 2463, *Lecture Notes in Computer Science*, 2002, pp. 111–122.
[17] H. Handa, L. Chapman, and X. Yao, "Robust route optimisation for gritting/salting trucks: A CERCIA experience," *IEEE Comput. Intell. Mag.*, vol. 1, no. 1, pp. 6–9, Feb. 2006.
[18] X. Hu and R. C. Eberhart, "Adaptive particle swarm optimization: Detection and response to dynamic systems," in *Proc. Congr. Evol. Comput.*, 2002, pp. 1666–1670.
[19] J. Hu, J. Zeng, and Y. Tan, "A diversity-guided particle swarm optimizer for dynamic environments," in *Bio-Inspired Computational Intelligence and Applications*. Berlin, Germany: Springer-Verlag, 2007, pp. 239–247.
[20] S. Janson and M. Middendorf, "A hierarchical particle swarm optimizer and its adaptive variant," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 35, no. 6, pp. 1272–1282, Dec. 2005.

LIU *et al.*: PSO WITH COMPOSITE PARTICLES IN DYNAMIC ENVIRONMENTS 15

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

[21] S. Janson and M. Middendorf, "A hierarchical particle swarm optimizer for noisy and dynamic environments," *Genetic Program. Evolvable Mach.*, vol. 7, no. 4, pp. 329–354, Dec. 2006.

[22] S. K. Jain, *Linear Algebra*, 4th ed. Belmont, CA: Thomson, 1994, pp. 12–20.

[23] L. Jiang and Y. Dan, "Preparation of poly(methyl methacrylate)/ $CaCO_3/SiO_2$ composite particles via emulsion polymerization," *Colloid Polym. Sci.*, vol. 282, no. 12, pp. 1374–1380, Oct. 2004.

[24] Y. Jin and J. Branke, "Evolutionary optimization in uncertain environments: A survey," *IEEE Trans. Evol. Comput.*, vol. 9, no. 3, pp. 303–317, Jun. 2005.

[25] J. Kennedy and R. C. Eberhart, "Particle swarm optimization," in *Proc. IEEE Int. Conf. Neural Netw.*, 1995, vol. 4, pp. 1942–1948.

[26] J. Kennedy, R. C. Eberhart, and Y. Shi, *Swarm Intelligence*. San Francisco, CA: Morgan Kaufmann, 2001.

[27] L. Liu, D. Wang, and S. Yang, "Compound particle swarm optimization in dynamic environments," in *Proc. EvoWorkshops, Appl. Evol. Comput.*, vol. 4974, *Lecture Notes in Computer Science*, 2008, pp. 617–626.

[28] X. Li, J. Branke, and T. M. Blackwell, "Particle swarm with speciation and adaptation in a dynamic environment," in *Proc. Genetic Evol. Comput. Conf.*, 2006, pp. 51–58.

[29] X. Li and K. H. Dam, "Comparing particle swarms for tracking extrema in dynamic environments," in *Proc. Congr. Evol. Comput.*, 2003, pp. 1772–1779.

[30] R. I. Lung and D. Dumitrescu, "A collaborative model for tracking optima in dynamic environments," in *Proc. Congr. Evol. Comput.*, 2007, pp. 564–567.

[31] R. W. Morrison and K. A. De Jong, "Triggered hypermutation revisited," in *Proc. Congr. Evol. Comput.*, 2000, pp. 1025–1032.

[32] S. Nakahata, K. Sogabe, and T. Matsuura, "One role of titanium compound particles in aluminium nitride sintered body," *J. Mater. Sci.*, vol. 32, no. 7, pp. 1873–1876, Apr. 1997.

[33] Y. S. Ong, P. B. Nair, and K. Y. Lum, "Max-min surrogate-assisted evolutionary algorithm for robust design," *IEEE Trans. Evol. Comput.*, vol. 10, no. 4, pp. 392–404, Aug. 2006.

[34] D. Parrott and X. Li, "A particle swarm model for tacking multiple peaks in a dynamic environment using speciation," in *Proc. Genetic Evol. Comput. Conf.*, 2004, vol. 1, pp. 98–103.

[35] D. Parrott and X. Li, "Locating and tracking multiple dynamic optima by a particle swarm model using speciation," *IEEE Trans. Evol. Comput.*, vol. 10, no. 4, pp. 440–458, Aug. 2006.

[36] K. E. Parsopoulos and M. N. Vrahatis, "Recent approaches to global optimization problems through particle swarm optimization," *Natural Comput.*, vol. 8, no. 2/3, pp. 235–306, Jun. 2002.

[37] K. E. Parsopoulos and M. N. Vrahatis, "On the computation of all global minimizers through particle swarm optimization," *IEEE Trans. Evol. Comput.*, vol. 8, no. 3, pp. 211–224, Jun. 2004.

[38] R. Poli, J. Kennedy, and T. M. Blackwell, "Particle swarm optimization," *Swarm Intell.*, vol. 1, no. 1, pp. 33–57, 2007.

[39] L. Schoeman and A. P. Engelbrecht, "Niching for dynamic environments using particle swarm optimization," in *Proc. 6th Int. Conf. Simul. Evol. Learn.*, 2006, pp. 134–141.

[40] Y. Tan and Z. M. Xiao, "Clonal particle swarm optimization and its applications," in *Proc. Congr. Evol. Comput.*, 2007, pp. 2303–2309.

[41] R. Tinos and S. Yang, "A self-organizing random immigrants genetic algorithm for dynamic optimization problems," *Genetic Program. Evolvable Mach.*, vol. 8, no. 3, pp. 255–286, Sep. 2007.

[42] I. Trelea, "The particle swarm optimization algorithm: Convergence analysis and parameter selection," *Inf. Process. Lett.*, vol. 85, no. 6, pp. 317–325, Mar. 2003.

[43] F. van den Bergh, "Analysis of particle swarm optimizers," Ph.D. dissertation, Dept. Comput. Sci., Univ. Pretoria, Pretoria, South Africa, 2002.

[44] F. van den Bergh and A. P. Engelbrecht, "Study of particle swarm optimization particle trajectories," *Inf. Sci.*, vol. 176, no. 8, pp. 937–971, Apr. 2006.

[45] S. Yang, "Genetic algorithms with elitism-based immigrants for changing optimization problems," in *Proc. EvoWorkshops, Appl. Evol. Comput.*, vol. 4448, *Lecture Notes in Computer Science*, 2007, pp. 627–636.

[46] S. Yang, "Genetic algorithms with memory- and elitism-based immigrants in dynamic environments," *Evol. Comput.*, vol. 16, no. 3, pp. 385–416, Fall 2008.

[47] S. Yang, Y.S. Ong, and Y. Jin, Eds., *Evolutionary Computation in Dynamic and Uncertain Environments*. Berlin, Germany: Springer-Verlag, 2007.

[48] S. Yang and R. Tinos, "Hyper-selection in dynamic environments," in *Proc. Congr. Evol. Comput.*, 2008, pp. 3185–3192.

[49] S. Yang and X. Yao, "Experimental study on population-based incremental learning algorithms for dynamic optimization problems," *Soft Comput.*, vol. 9, no. 11, pp. 815–834, Nov. 2005.

[50] S. Yang and X. Yao, "Population-based incremental learning with associative memory for dynamic environments," *IEEE Trans. Evol. Comput.*, vol. 12, no. 5, pp. 542–561, Oct. 2008.

[51] J. Q. Zhang, Y. Tan, and X. G. He, "Random black hole particle swarm optimization and its application," in *Proc. IEEE Int. Conf. Neural Netw. Signal Process.*, 2008, pp. 359–365.

[52] J. Q. Zhang, Y. Tan, and X. G. He, "Concentric spatial extension based particle swarm optimization inspired by brood sorting in ant colonies," in *Proc. IEEE Swarm Intell. Symp.*, 2009, pp. 9–15.

Lili Liu received the B.Sc. degree in automatic control and the M.Sc. degree in systems engineering in 2003 and 2007, respectively, from Northeastern University, Shenyang, China, where she is currently working toward the Ph.D. degree in systems engineering in the College of Information Science and Engineering.

She is currently a Teaching Assistant with the College of Information Science and Engineering, Northeastern University. Her current research interests include evolutionary computation for dynamic optimization problems, i.e., dynamic problems within the contexts of scheduling, mobile *ad hoc* networks, and radio frequency identification. She has authored or coauthored more than ten technical papers.

Shengxiang Yang (M'00) received the B.Sc. and M.Sc. degrees in automatic control and the Ph.D. degree in systems engineering from Northeastern University, Shenyang, China, in 1993, 1996, and 1999, respectively.

From October 1999 to October 2000, he was a Postdoctoral Research Associate with the Algorithm Design Group, Department of Computer Science, King's College London, London, U.K. He is currently a Lecturer with the Department of Computer Science, University of Leicester, Leicester, U.K. He has over 100 publications. He has given invited keynote speeches in several international conferences and co-organized several workshops and special sessions in conferences. He serves as an area editor, an associate editor, or an editorial board member for four international journals. He has coedited several books and conference proceedings, and coguest-edited several journal special issues. His major research interests include evolutionary algorithms, swarm intelligence, metaheuristics and hyperheuristics, artificial neural networks, computational intelligence in dynamic and uncertain environments, scheduling, network flow problems and algorithms, and real-world applications.

Dr. Yang is a member of the Association of Computing Machinery Special Interest Group on Genetic and Evolutionary Computation. He is a member of the Task Force on Evolutionary Computation in Dynamic and Uncertain Environments, Evolutionary Computation Technical Committee, IEEE Computational Intelligence Society.

Dingwei Wang received the M.Sc. degree in systems engineering from Huazhong University of Science and Technology, Wuhan, China, in 1984 and the Ph.D. degree in control theory and application from Northeastern University, Shenyang, China, in 1993.

From 1994 to 1995, he was a Postdoctoral Fellow with the North Carolina State University, Raleigh. He is currently a Professor with the Institute of Systems Engineering, Northeastern University, Shenyang, China. He has published more than 300 papers in international or domestic journals, including IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS, IEEE TRANSACTIONS ON NEURAL NETWORKS, *Sciences in China*, etc. His research interests include the modeling of optimization of complex systems, evolutionary computation, and soft computing.

Dr. Wang is a member of the Chinese Association Automation and the Chinese Association of Operations Research. He serves on the editorial board of *Fuzzy Optimization* and *Decision Making* and several Chinese journals.