Information Sciences 193 (2012) 81–103

Contents lists available at SciVerse ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Convergence analysis and improvements of quantum-behaved particle swarm optimization

Jun Sun [a,*], Xiaojun Wu [a,*], Vasile Palade [b], Wei Fang [a], Choi-Hong Lai [c], Wenbo Xu [a]

[a] Key Laboratory of Advanced Control for Light Industry (Ministry of China), No. 1800, Lihu Avenue, Wuxi, Jiangsu 214122, China
[b] Department of Computer Science, University of Oxford, Parks Road, Oxford OX1 3QD, United Kingdom
[c] School of Computing and Mathematical Sciences, University of Greenwich, London SE10 9LS, United Kingdom

ARTICLE INFO

Article history:
Received 14 August 2011
Received in revised form 18 December 2011
Accepted 3 January 2012
Available online 11 January 2012

Keywords:
Global Convergence
Particle swarm
Quantum behavior
Time complexity
Performance evaluation

ABSTRACT

Motivated by concepts in quantum mechanics and particle swarm optimization (PSO), quantum-behaved particle swarm optimization (QPSO) was proposed as a variant of PSO with better global search ability. Although it has been shown to perform well in finding optimal solutions for many optimization problems, there has so far been little theoretical analysis on its convergence and performance. This paper presents a convergence analysis and performance evaluation of the QPSO algorithm and it also proposes two variants of the QPSO algorithm. First, we investigate in detail the convergence of the QPSO algorithm on a probabilistic metric space and prove that the QPSO algorithm is a form of contraction mapping and can converge to the global optimum. This is the first time that the theory of probabilistic metric spaces has been employed to analyze a stochastic optimization algorithm. We provided a new definition for the convergence rate of a stochastic algorithm as well as definitions for three types of convergence according to the correlations between the convergence rate and the objective function values. With these definitions, the effectiveness of the QPSO is evaluated by computing and analyzing the time complexity and the convergence rate of the algorithm. Then, the QPSO with random mean best position (QPSO-RM) and the QPSO with ranking operator (QPSO-RO) are proposed as two improvements of the QPSO algorithm. Finally, some empirical studies on popular benchmark functions are performed in order to make a full performance evaluation and comparison between QPSO, QPSO-RM, QPSO-RO and other variants of PSO.

1. Introduction

Particle swarm optimization (PSO), motivated by the social behavior of bird flocks or fish schooling, was first introduced by Kennedy and Eberhart as a population-based optimization technique [31]. In PSO, the potential solutions, called particles, fly through the problem space by following their own experiences and the current best particle. The PSO algorithm is comparable in performance with the well known Genetic Algorithm (GA) approach [1,21,29,44,50], and has gained increasing popularity during the last decade due to its effectiveness in performing difficult optimization tasks.

In order to gain a deep insight into the mechanism of PSO, many theoretical analyses have been done on the algorithm. Most of these works focused on the behaviour of the single particle in PSO, analyzing the particle's trajectory or its stability by using deterministic or stochastic methods [5,11,18,20,28,30,32,51,69,84]. As for the algorithm itself, Van den Bergh [5]

* Corresponding authors. Address: Key Laboratory of Advanced Control for Light Industry (Ministry of China), Department of Computer Science and Technology, Jiangnan University, No. 1800, Lihu Avenue, Wuxi, Jiangsu 214122, China. Tel./fax: +86 510 85916500.
E-mail addresses: sunjun_wx@hotmail.com (J. Sun), wu_xiaojun@yahoo.com.cn (X. Wu).

0020-0255/$ - see front matter © 2012 Elsevier Inc. All rights reserved.
doi:10.1016/j.ins.2012.01.005

proved that the canonical PSO is not a global search algorithm, even not a local one, according to the convergence criteria provided by Solis and Wets [74].

In addition to the theoretical analyses mentioned above, there has been a considerable amount of work done in developing the original version of PSO, through empirical simulations. Shi and Eberhart [68] introduced the concept of inertia weight to the original PSO, in order to balance the local and global search during the optimization process. Clerc [10] proposed an alternative version of PSO incorporating a parameter called constriction factor which should replace the restriction on velocities. Angeline [2] introduced a tournament selection into PSO based on the particle's current fitness so that the properties that make some solutions superior were transferred directly to some of the less effective particles. This technique improved the performance of the PSO algorithm on some benchmark functions. Suganthan [76] proposed a variant of the algorithm, with another general form of particle swarm optimization referred to as the local best (LBest) model. It divided the swarm into multiple "neighborhoods", where each neighborhood maintained its own local best solution. This approach was less prone to becoming trapped in local minima, but typically had slower convergence. Several researchers investigated other neighborhood topologies or adaptive topologies that may enhance the performance of PSO, in order to improve the exploration ability of the algorithm [6,12,27,33,34,42,43,46,49,54].

Some researchers have attempted to experiment with various ways to simulate the particle trajectories by directly sampling, using a random number generator with a certain probability distribution [35–39,56,64,78,79]. For example, Sun et al., inspired by quantum mechanics and the trajectory analysis of PSO [11], used a strategy based on a quantum δ potential well to sample around the previous best points [78], and later introduced the mean best position into the algorithm and proposed a new version of PSO, quantum-behaved particle swarm optimization (QPSO) [79,80]. The QPSO algorithm essentially falls into the family of bare-bones PSO [35,36], but uses double exponential distribution and an adaptive strategy to sample particle's positions. The iterative equation of QPSO is very different from that of PSO, and leads QPSO to be global convergent, as will be proved mathematically in this paper. Besides, unlike PSO, QPSO needs no velocity vectors for particles, and also has fewer parameters to adjust, making it easier to implement.

The QPSO algorithm has been shown to successfully solve a wide range of continuous optimization problems and many efficient strategies have been proposed to improve the algorithm [13–15,19,45,48,86,89]. While empirical evidence has shown that the algorithm works well, there has thus far been little insight into how it works. In this paper, we investigate the convergence issue of the QPSO and propose two improved versions of the algorithm as well. First, the global convergence of the QPSO is analyzed on the probability metric (PM) space established for the algorithm. We prove that the QPSO is a form of contraction mapping on the PM space and its orbit is probabilistic bounded, and, in turn, that the algorithm converges asymptotically to the global optimum, the unique fixed point. It is the first time that the theory of PM-sapces has been used to analyze a stochastic optimization algorithm. Next, the time complexity and its relationship to the behavior of a single particle are addressed, and a new definition of the convergence rate for a stochastic algorithm is presented, followed by semi-theoretically evaluating the time complexity and convergence rate of the QPSO and PSO. Then, we propose two improved versions of QPSO, in order to enhance the search ability of the algorithm. One improved QPSO employs a random mean best position, so that the particle swarm can be diversified during the search and thus its global search ability is enhanced. The other incorporates a ranking operator to select a random particle whose personal best position replaces the global best position in order to guide the particle to escape the local optima. Finally, in order to further evaluate the efficiency of the QPSO algorithms and test the performance of the improved QPSO, we make a performance comparison with other variants of PSO by testing the algorithms on a set of problems from the CEC2005 benchmarks.

The remainder of the paper is structured as follows. In Section 2, the principles of QPSO are introduced. The global convergence analysis of QPSO is given in Section 3. The efficiency evaluation of the algorithms by time complexity and convergence rate is provided in Section 4. Section 5 presents the two proposed improved QPSO algorithms. Section 6 provides the experimental results on benchmark functions. Some concluding remarks are given in Section 7.

2. Quantum-behaved particle swarm optimization

In the PSO with M individuals, each individual is treated as a volume-less particle in the N-dimensional space, with the current position vector and velocity vector of particle i at the nth iteration represented by $X_{i,n} = \left(X_{i,n}^1, X_{i,n}^2, \ldots, X_{i,n}^N\right)$ and $V_{i,n} = \left(V_{i,n}^1, V_{i,n}^2, \ldots, V_{i,n}^N\right)$. The particle updates its position and velocity according to the following equations:

$$V_{i,n+1}^j = V_{i,n}^j + c_1 r_{i,n}^j \left(P_{i,n}^j - X_{i,n}^j\right) + c_2 R_{i,n}^j \left(G_n^j - X_{i,n}^j\right), \quad (1)$$

$$X_{i,n+1}^j = X_{i,n}^j + V_{i,n+1}^j, \quad (2)$$

for $i = 1, 2, \ldots, M$; $j = 1, 2, \ldots, N$, where c_1 and c_2 are called acceleration coefficients. Vector $P_{i,n} = \left(P_{i,n}^1, P_{i,n}^2, \ldots, P_{i,n}^N\right)$ is the best previous position (the position giving the best objective function value or fitness value) of particle i called personal best (*pbest*) position, and vector $G_n = \left(G_n^1, G_n^2, \ldots, G_n^N\right)$ is the position of the best particle among all the particles in the population and called global best (*gbest*) position. The parameters $r_{i,n}^j$ and $R_{i,n}^j$ are sequences of two different random numbers distributed uniformly on (0, 1), that is, $r_{i,n}^j, R_{i,n}^j \sim U(0,1)$. Generally, the value of $V_{i,n}^j$ is restricted within the interval $[-V_{\max}, V_{\max}]$. Without loss of generality, if we consider the following minimization problem:

$$\begin{aligned} &\text{Minimize} \quad f(X), \\ &\text{s.t. } X \in S \subseteq R^N, \end{aligned} \tag{3}$$

where $f(X)$ is an objective function continuous almost everywhere and S is the feasible space, then $P_{i,n}$ can be updated by

$$P_{i,n} = \begin{cases} X_{i,n} & \text{if} \quad f(X_{i,n}) < f(P_{i,n-1}) \\ P_{i,n-1} & \text{if} \quad f(X_{i,n}) \geqslant f(P_{i,n-1}) \end{cases}, \tag{4}$$

and accordingly G_n can be found by $G_n = P_{g,n}$, where $g = \text{argmin}_{1 \leqslant i \leqslant M}[f(P_{i,n})]$.

The trajectory analysis in [11] demonstrated the fact that the convergence of the PSO algorithm may be achieved if each particle converges to its local attractor, $p_{i,n} = \left(p_{i,n}^1, p_{i,n}^2, \ldots p_{i,n}^N\right)$ defined at the coordinates

$$p_{i,n}^j = \frac{c_1 r_{i,n}^j P_{i,n}^j + c_2 R_{i,n}^j G_n^j}{c_1 r_{i,n}^j + c_2 R_{i,n}^j}, \quad 1 \leqslant j \leqslant N, \tag{5}$$

or

$$p_{i,n}^j = \varphi_{i,n}^j P_{i,n}^j + \left(1 - \varphi_{i,n}^j\right) G_n^j, \tag{6}$$

where $\varphi_{i,n}^j = c_1 r_{i,n}^j \Big/ \left(c_1 r_{i,n}^j + c_2 R_{i,n}^j\right)$ with regard to the random numbers $r_{i,n}^j$ and $R_{i,n}^j$ in (1) and (5). In PSO, the acceleration coefficients c_1 and c_2 are generally set to be equal, i.e., $c_1 = c_2$, and thus $\varphi_{i,n}^j$ is a sequence of uniformly distributed random numbers on (0,1). As a result, Eq. (6) can be restated as

$$p_{i,n}^j = \varphi_{i,n}^j P_{i,n}^j + \left(1 - \varphi_{i,n}^j\right) G_n^j, \quad \varphi_{i,n}^j \sim U(0,1). \tag{7}$$

The above equation indicates that $p_{i,n}$, the stochastic attractor of particle i, lies in the hyper-rectangle with $P_{i,n}$ and G_n being the two ends of its diagonal so that it moves following $P_{i,n}$ and G_n. In fact, as the particles are converging to their own local attractors, their current position, personal best positions, local attractors and the global best positions are all converging to one point, leading the PSO algorithm to converge. From the point view of Newtonian dynamics, in the process of convergence, the particle moves around and careens toward point $p_{i,n}$ with its kinetic energy (or velocity) declining to zero, like a returning satellite orbiting the earth. As such, the particle in PSO can be considered as the one flying in an attraction potential field centered at point $p_{i,n}$ in Newtonian space. It has to be in a bound state for the sake of avoiding explosion and guaranteeing convergence. If these conditions are generalized to the case that the particle in PSO moves in quantum space, it is also indispensable that the particle should move in a quantum potential field to ensure the bound state. The bound state in quantum space, however, is entirely different from that in Newtonian space, which may lead to a very different form of PSO. This is the motivation of the QPSO algorithm [78].

In QPSO, each single particle is treated as a spin-less one moving in quantum space. Thus the state of the particle is characterized by a wave function ψ, where $|\psi|^2$ is the probability density function of its position. Inspired by the convergence analysis of the particle in PSO, we assume that, at the nth iteration, particle i flies in the N-dimensional quantum space with a δ potential well centered at $p_{i,n}^j$ on the jth dimension ($1 \leqslant j \leqslant N$). Let $Y_{i,n+1}^j = \left|X_{i,n+1}^j - p_{i,n}^j\right|$, then we can obtain the normalized wave function at iteration n+1 as;

$$\psi\left(Y_{i,n+1}^j\right) = \frac{1}{\sqrt{L_{i,n}^j}} \exp\left(-Y_{i,n+1}^j / L_{i,n}^j\right), \tag{8}$$

which satisfies the bound condition that $\psi\left(Y_{i,n+1}^j\right) \to 0$ as $Y_{i,n+1}^j \to \infty$. $L_{i,n}^j$ is the characteristic length of the wave function. By the definition of wave function, the probability density function is given by

$$Q\left(Y_{i,n+1}^j\right) = \left|\psi\left(Y_{i,n+1}^j\right)\right|^2 = \frac{1}{L_{i,n}^j} \exp\left(-2Y_{i,n+1}^j / L_{i,n}^j\right), \tag{9}$$

and thus the probability distribution function is

$$F\left(Y_{i,n+1}^j\right) = 1 - \exp\left(-2Y_{i,n+1}^j / L_{i,n}^j\right). \tag{10}$$

Using the Monte Carlo method, we can measure the jth component of position of particle i at the $(n+1)$th iteration by

$$X_{i,n+1}^j = p_{i,n}^j \pm \frac{L_{i,n}^j}{2} \ln\left(1/u_{i,n+1}^j\right), \quad u_{i,n+1}^j \sim U(0,1), \tag{11}$$

where $u^j_{i,n+1}$ is a sequence of random numbers uniformly distributed on (0, 1). Two approaches of determining the value of $L^j_{i,n}$ were proposed in [78,79], respectively:

$$L^j_{i,n} = 2\alpha\left|X^j_{i,n} - p^j_{i,n}\right|, \tag{12}$$

and

$$L^j_{i,n} = 2\alpha\left|X^j_{i,n} - C^j_n\right|, \tag{13}$$

where $C_n = \left(C^1_n, C^2_n, \ldots, C^N_n\right)$ is called mean best (*mbest*) position defined by the average of the *pbest* positions of all particles, i.e., $C^j_n = (1/M)\sum_{i=1}^{M} P^j_{i,n}\ (1 \leqslant j \leqslant N)$. These two strategies result in two versions of the QPSO algorithm. To distinguish them, we denote the QPSO with the former approach as QPSO-Type 1 and the QPSO with the latter one as QPSO-Type 2. Therefore, the position of the particle in either type of QPSO is updated according to the following two equations respectively:

$$X^j_{i,n+1} = p^j_{i,n} \pm \alpha\left|X^j_{i,n} - p^j_{i,n}\right| \ln\left(1/u^j_{i,n+1}\right), \tag{14}$$

or

$$X^j_{i,n+1} = p^j_{i,n} \pm \alpha\left|X^j_{i,n} - C^j_n\right| \ln\left(1/u^j_{i,n+1}\right). \tag{15}$$

The parameter α in Eqs. (13)–(15) is known as the contraction-expansion (CE) coefficient, which can be adjusted to balance the local and global search of the algorithm during the optimization process. The search procedure of the algorithm is outlined in Fig. 1. Note that rand*i*(·), *i* = 1, 2, 3, is used to denote random numbers generated uniformly and distributed on (0, 1).

Procedure of the QPSO algorithm:
Begin
Initialize the current position $X^j_{i,0}$ and the personal best position $P^j_{i,0}$ of each particle, evaluate their fitness values and find the global best position G_0; Set n=0.
While (termination condition = false)
Do
Set n=n+1;
Compute mean best position C_n (for QPSO-Type 2);
Select a suitable value for α;
for (i=1 to M)
for j=1 to N
$\varphi^j_{i,n} = \text{rand1}(\cdot)$;
$p^j_{i,n} = \varphi^j_{i,n} P^j_{i,n} + (1-\varphi^j_{i,n}) G^j_n$;
$u^j_{i,n} = \text{rand2}(\cdot)$;
if (rand3(·) < 0.5)
$X^j_{i,n+1} = p^j_{i,n} + \alpha | X^j_{i,n} - p^j_{i,n} | \ln(1/u^j_{i,n+1})$ (for QPSO-Type 1);
(or $X^j_{i,n+1} = p^j_{i,n} + \alpha | X^j_{i,n} - C^j_n | \ln(1/u^j_{i,n+1})$ (for QPSO-Type 2));
else
$X^j_{i,n+1} = p^j_{i,n} - \alpha | X^j_{i,n} - p^j_{i,n} | \ln(1/u^j_{i,n+1})$ (for QPSO-Type 1);
(or $X^j_{i,n+1} = p^j_{i,n} - \alpha | X^j_{i,n} - C^j_n | \ln(1/u^j_{i,n+1})$ (for QPSO-Type 2));
end if
end for
Evaluate the fitness value of $X_{i,n+1}$, that is, the objective function value $f(X_{i,n+1})$;
Update $P_{i,n}$ and G_n
end for
end do
end

Fig. 1. The procedure of the QPSO algorithm.

3. Convergence of the QPSO algorithm

In this section, the global convergence of the QPSO is analyzed by establishing a probabilistic metric space (PM-space) for the algorithm, in which we prove that the algorithm is a form of contraction mapping and its orbit is probabilistic bounded. Before the convergence analysis, we give an introduction to the PM-space and some related work, and provide basic concepts of a PM-space and the fixed point theorem on the PM-space that are used in the analysis.

3.1. An introduction to PM-spaces and some related work

The idea of a PM-space was first introduced by Menger as a generalization of ordinary metric spaces [47]. In this theory, the notion of distance has a probabilistic nature, that is, the distance between two points x and y is represented by a distribution function $F_{x,y}$, and for any positive number t, the value $F_{x,y}(t)$ is interpreted as the probability that the distance from x to y is less than t. Such a probabilistic generalization is well adapted for the investigation of physical events, and has also important applications in nonlinear analysis [9].

The theory of PM-spaces was brought to its present state by Schweizer and Sklar [59–62], Šeerstnev [67], Tardiff [82] and Thorp [83]. There are also many others studying on PM-spaces [16,58,66]. For a clear and detailed history, as well as for the motivations behind the introduction of PM-spaces, the reader should refer to the book by Schweizer and Sklar [63].

The convergence theorems for obtaining the stable points, i.e., the fixed point theorems for contraction mappings, have been always an active area of research since 1922, with the celebrated Banach contraction fixed point theorem. Seghal [65] initiated the study of the fixed point theorems in PM-spaces. Subsequently, some other fixed point theorems for contraction mappings have been proved in PM-spaces [8,25,26,40,53,71–73]. In [52], the authors established global output convergence for a recurrent neural network (RNN) with continuous and monotone non-decreasing activation functions, by using the fixed point theorems in PM-spaces. They provided the sufficient conditions to guarantee the global output convergence of this class of neural networks, which are very useful in the design of RNNs. However, since the output of a RNN is not probabilistic of nature, they practically employed the fixed point theorems in PM-spaces to analyze the convergence of the sequence of non-random variables. That is, they essentially specialized the fixed point theorems in PM-spaces into the ones in ordinary metric spaces.

In the remaining part of this section, we analyze the convergence of the QPSO algorithm, whose output, i.e. the fitness value of the global best position, is a sequence of random variables. Therefore the theory of PM-spaces is very suitable for the analysis of the algorithm. To our knowledge, it is the first time that the fix point theorem of a stochastic algorithm has ever been proved. Although this section focuses on the QPSO algorithm, the established theoretical framework can be used as a general–purpose analysis tool for the convergence of any stochastic optimization algorithm.

3.2. Preliminaries

Definition 1. Denote the set of all real numbers as R, and the set of all non-negative real numbers as R^+. The mapping $f: R \to R^+$ is a called distribution function if it is non-decreasing, left continuous and $\inf_{t\in R} f(t) = 0$, $\sup_{t\in R} f(t) = 1$.

We denote the set of all distribution functions by $\mathcal{D}$, and $H(t)$ is the specific distribution function defined by

$$H(t) = \begin{cases} 1 & t > 0 \\ 0 & t \leqslant 0 \end{cases}. \tag{16}$$

Definition 2. A probabilistic metric space (briefly, a PM-space) is an ordered pair (E, F), where E is a nonempty set and F is a mapping of $E \times E$ into $\mathcal{D}$. The value of F at $(x, y) \in E \times E$ is denoted by $F_{x,y}$, and $F_{x,y}(t)$ represents the value of $F_{x,y}$ at t. The functions $F_{x,y}$ $(x,y \in E)$ are assumed to satisfy the following conditions:

(PM-1) $F_{x,y}(0) = 0$;
(PM-2) $F_{x,y}(t) = H(t)$ for all $t > 0$ if and only if $x = y$;
(PM-3) $F_{x,y} = F_{y,x}$;
(PM-4) $F_{x,y}(t_1) = 1$ and $F_{y,z}(t_2) = 1$ imply $F_{x,z}(t_1 + t_2) = 1$, $\forall x, y, z \in E$.

The value $F_{x,y}(t)$ of $F_{x,y}$ at $t \in R$ can be interpreted as the probability that the distance between x and y is less than t.

Definition 3. A mapping Δ: $[0, 1] \times [0, 1] \to [0, 1]$ is a triangle norm (briefly t-norm) if it satisfies: for every $a, b, c, d \in [0, 1]$,

(Δ-1) $\Delta(a, 1) = a$, $\Delta(0, 0) = 0$;
(Δ-2) $\Delta(a, b) = \Delta(b, a)$;
(Δ-3) $\Delta(c, d) \geqslant \Delta(a, b)$ for $c \geqslant a$, $d \geqslant b$;
(Δ-4) $\Delta(\Delta(a, b), c) = \Delta(a, \Delta(b, c))$.

It can be easily verified that $\Delta_1(a, b) = \max\{a + b - 1, 0\}$ is a t-norm.

Definition 4. A Menger probabilistic metric space (briefly a Menger space) is a triplet (E, F, Δ), where (E, F) is a PM-space and t-norm Δ is such that Menger's triangle inequality

(PM-4)' $F_{x,z}(t_1 + t_2) \geqslant \Delta(F_{x,y}(t_1), F_{y,z}(t_2))$

holds for all $x, y, z \in E$ and for all $t_1 \geqslant 0$, $t_2 \geqslant 0$.

If (E,F,Δ) is a Menger space with a continuous t-norm, then it is a Hausdoff Space with the topology $\mathcal{T}$ introduced by the family $\{U_y(\varepsilon, \lambda): y \in E, \lambda > 0\}$, where

$$U_y(\varepsilon, \lambda) = \{x \in E, F_{x,y}(\varepsilon) > 1 - \lambda, \varepsilon > 0, \lambda > 0\} \tag{17}$$

is called an (ε, λ)-neighborhood of $y \in E$. Hence we can introduce the following concepts into (E, F, Δ).

Definition 5. Let $\{x_n\}$ a sequence in a Menger space (E, F, Δ), where Δ is continuous. The sequence $\{x_n\}$ converges to $x_* \in E$ in $\mathcal{T}$ $(x_n \xrightarrow{\mathcal{T}} x_*)$, if for every $\varepsilon > 0$ and $\lambda > 0$, there exists a positive integer $K = K(\varepsilon, \lambda)$ such that $F_{x_n, x_*}(\varepsilon) > 1 - \lambda$, whenever $n \geqslant K$.

The sequence $\{x_n\}$ is called a $\mathcal{T}$-Cauchy Sequence in E, if for every $\varepsilon > 0$ and $\lambda > 0$, there exists a positive integer $K = K(\varepsilon, \lambda)$ such that $F_{x_n, x_m}(\varepsilon) > 1 - \lambda$, whenever $m, n \geqslant K$. A Menger space (E, F, Δ) is called $\mathcal{T}$-Complete if every $\mathcal{T}$-Cauchy Sequence in E converges in $\mathcal{T}$ to a point in E. In [63], it was proved that every Menger space with continuous t-norm is $\mathcal{T}$-Complete.

Definition 6. Let (E, F, Δ) be a Menger space where Δ is continuous. The self-mapping T that maps E into itself is $\mathcal{T}$-continuous on E, if for every sequence $\{x_n\}$ in E, $Tx_n \xrightarrow{\mathcal{T}} Tx_*$ whenever $x_n \xrightarrow{\mathcal{T}} x_* \in E$.

3.3. The fixed point theorem in PM-space

Definition 7. Let T be a self-mapping of a Menger space(E, F, Δ). T is a contraction mapping, if there exists a constant $k \in (0, 1)$ and for every $x \in E$ there exists a positive integer $n(x)$ such that for every $y \in E$,

$$F_{T^{n(x)}x, T^{n(x)}y}(t) \geqslant F_{x,y}\left(\frac{t}{k}\right), \quad \forall t \geqslant 0. \tag{18}$$

A set $A \subset (E, F, \Delta)$ is called probabilistic bounded if $\sup_{t>0}\inf_{x,y\in A} F_{x,y}(t) = 1$. Denote the orbit generated by T at $x \in E$ by $O_T(x; 0, \infty)$, i.e., $O_T(x; 0, \infty) = \{x_n = T^n x\}_{n=0}^{\infty}$. For the contraction mapping in Definition 7, we have the following fixed point theorem.

Theorem 1. *Let a self-mapping $T: (E, F, \Delta) \to (E, F, \Delta)$ be the contraction mapping in Definition 7. If for every $x \in E$, $O_T(x; 0, \infty)$ is probabilistic bounded, then there exists a unique common fixed point x_* in E for T, and for every $x_0 \in E$, the iterative sequence $\{T^n x_0\}$ converges to x_* in $\mathcal{T}$ (see the proof in Appendix A).*

3.4. Global convergence of the QPSO algorithm

3.4.1. Construction of the PM-space for QPSO

Consider the minimization problem defined by (3), which is rewritten as follows:

$$\begin{aligned} &\text{Minimize} \quad f(X), \\ &\text{s.t. } X \in S \subseteq R^N, \end{aligned} \tag{19}$$

where f is a real-valued function defined over region S and continuous almost everywhere, and S is a compact subset of a N-dimensional Euclidean space R^N. Let V represent the range of f over S, and thus $V \subset R$. Denote $f_* = \min_{X\in S}\{f(X)\}$.

Theorem 2. *Consider the ordered pair (V, F), where F is a mapping of $V \times V$ into D. For every $x, y \in V$, if the distribution function $F_{x,y}$ is defined by $F_{x,y}(t) = P\{|x - y| < t\}$, $\forall t \in R$, then (V, F) is a PM-space (see the proof in Appendix A).*

Theorem 3. *The triplet (V, F, Δ) is a Menger space, where $\Delta = \Delta_1$ (see the proof in Appendix A).*

(V, F, Δ) is a Menger space with continuous t-norm Δ_1, and also is a Hausdoff space of the topology T introduced by the family $\{U_y(\varepsilon, \lambda): y \in V, \lambda > 0\}$, where $U_y(\varepsilon, \lambda) = \{x \in V, F_{x,y}(\varepsilon) > 1 - \lambda, \varepsilon, \lambda > 0\}$, and consequently is $\mathcal{T}$-Complete.

3.4.2. The fixed point theorem of the QPSO algorithm

We regard the QPSO as a mapping denoted by T, and therefore T is a self-mapping of the Menger space (V, F, Δ). When $n = 0$, the global best position G_0 is generated by the initialization of QPSO. Let $f_0 = f(G_0)$, and thus $f_0 \in V$. By a series of iterations of the algorithm, we can obtain a sequence of global best position $\{G_n, n \geqslant 1\}$, and a sequence of the corresponding non-increasing function values $\{f_n, n \geqslant 1\}$, where $f_n = f(G_n)$. We can consider $\{f_n, n \geqslant 1\}$ as a sequence of points generated by T in (V, F, Δ), i.e. $f_n = T^n f_0$ and $f_n \in V$. Denoting the orbit generated by T at $f_0 \in V$ by $O_T(f_0; 0, \infty)$, we have $O_T(f_0; 0, \infty) = \{f_n = T^n f_0\}_{n=0}^{\infty}$. The following theorem proves that T is a contraction mapping of (V, F, Δ).

Theorem 4. *The mapping T is a contraction mapping of the Menger space (V, F, Δ) (see the proof in Appendix A).*

Theorem 5. *f_* is the unique fixed point in V such that for every $f_0 \in V$, the iterative sequence $\{T^n f_0\}$ converges to f_* (see the proof in Appendix A).*

Define the optimality region of problem (3) by $V_\varepsilon = V(\varepsilon) = \{f: f \in V, f - f_* < \varepsilon\}$. For QPSO in$(V, F, \Delta)$, the theorem below shows the equivalence between the convergence in $\mathcal{T}$ and the convergence in probability.

Theorem 6. *The sequence of function values $\{f_n, n \geqslant 0\}$ generated by the QPSO converges to f_* in probability (see the proof in Appendix A).*

The above convergence analysis on a PM-space can be used to analyze other random optimization algorithms. Essentially, any global convergent algorithm is a contraction mapping defined by Definition 7, whose orbit is probabilistic bounded. The PSO algorithm is not global convergent, since it does not satisfy the contractive condition of Definition 7, even though its orbit is probabilistic bounded.

4. Time complexity and convergence rate of the QPSO algorithm

4.1. Measure of time complexity

Convergence is an important characteristic of a stochastic optimization algorithm. Nevertheless, it is not sufficient to evaluate the efficiency of the algorithm by comparison to others. The most promising and tractable approach is the study of the distribution of the number of steps required to reach the optimality region $V(\varepsilon)$, more specifically by comparing the expected number of steps and higher moments of this distribution. The number of steps required to reach $V(\varepsilon)$ is defined by $K(\varepsilon) = \inf\{n | f_n \in V_\varepsilon\}$. The expected value (time complexity) and the variance of $K(\varepsilon)$, if they exist, can be computed by

$$E[K(\varepsilon)] = \sum_{n=0}^{\infty} n a_n, \tag{20}$$

$$Var[K(\varepsilon)] = E[K^2(\varepsilon)] - \{E[K(\varepsilon)]\}^2 = \sum_{n=0}^{\infty} n^2 a_n - \left(\sum_{n=0}^{\infty} n a_n\right)^2, \tag{21}$$

where $a_n = a_n(\varepsilon)$ with $a_n(t)$ defined by (A13), that is

$$a_n = P\{K(\varepsilon) = n\} = P\{f_0 \in V_\varepsilon^c, f_1 \in V_\varepsilon^c, f_2 \in V_\varepsilon^c, \ldots, f_{n-1} \in V_\varepsilon^c, f_n \in V_\varepsilon\} = a_n(\varepsilon). \tag{22}$$

Referring to (A14), we can also find that

$$F_{f_n f_*}(\varepsilon) = P\{f_n \in V_\varepsilon\} = \sum_{i=0}^{n} a_n. \tag{23}$$

Thus, it is required that $\sum_{i=0}^{\infty} a_n = 1$ so that the algorithm can be global convergent. It is evident that the existence of $E[K(\varepsilon)]$ relies on the convergence of $\sum_{n=0}^{\infty} n a_n$. Generally, for most of the stochastic optimization algorithms, particularly population-based random search techniques, it is far more difficult to compute all a_n analytically than to prove the global convergence of the algorithm. To evaluate $E[K(\varepsilon)]$ and $Var[K(\varepsilon)]$, researchers have either undertaken theoretical analysis relying on specific situations or provided the numerical results on some specific functions [3,22–24,75,85,87].

Now we focus our attention on the problem of how the behavior of the individual particle influences the convergence of QPSO, from the perspective of time complexity. It has been shown that for both types of QPSO, setting $\alpha \leqslant e^\gamma \approx 1.781$ (where $\gamma \approx 0.577215665$ is called Euler constant) can prevent the particle from exploding [81]. As can be seen, however, the proof of global convergence of QPSO does not involve the behavior of the individual particle. It is true that the algorithm is global convergent even when the particle diverges (i.e. when $\alpha > e^\gamma$). The global convergence of QPSO only requires that $\sum_{i=0}^{\infty} a_n = 1$ or $g_n(\varepsilon) > 0$ for all n, according to (A14). When the particle diverges, for every $\varepsilon > 0$, although $g_n(\varepsilon)$ declines constantly, $F_{f_n f_*}(\varepsilon) = \sum_{i=0}^{n} a_n(\varepsilon) = 1 - \prod_{i=1}^{n}[1 - g_i(\varepsilon)]$ may also converge to 1 since $0 < g_n(\varepsilon) < 1$ for all $n < \infty$. The series $\sum_{n=0}^{\infty} n a_n$, however, may diverge in such a case. Therefore, we find that the divergence of the particle may also guarantee the global convergence of the algorithm, but can result in infinite complexity in general. On the other hand, when the particle converges or is bounded, $g_n(\varepsilon)$ does not decline constantly but may even increase during the search. As a result, for certain

$\varepsilon > 0$, $\sum_{n=0}^{\infty} na_n$ can converge, which implies that the algorithm has finite time complexity. Thus, to make QPSO converge globally with finite time complexity, we have to set $\alpha \leqslant e^{\gamma}$ to ensure the convergence or boundedness of the particle.

4.2. The convergence rate

Another measure used to evaluate the efficiency of the algorithm is its convergence rate. The mathematical complexity of analyzing the convergence rate of a population-based random optimization algorithm, however, is no less significant than that of computing the expected value or variance of $K(\varepsilon)$. Although some work has been done on the issue of the convergence rate [7,41,55,57,74,85], it is apparently an open problem for arbitrary objective functions. In these literatures, the convergence rate of an algorithm is defined as the rate of change of Euclidean distance from the current solution to the optimal point. This definition can indeed measure the convergence speed of the algorithm intuitively and effectively when the optimization problem is unimodal. However, if the problem is multimodal, it may fail to evaluate the efficiency of the algorithm properly when the current solution flies away from the optimal point but its fitness value improves. Particularly, when the objective function has many optimal solutions in the given search domain, such a definition for the convergence rate is unfeasible since we cannot determine which optimal point is used as the reference point, to which the distance from current solution should be computed.

The definition proposed in this work differs from the above one in that it measures the convergence rate by the rate of change of the difference between the current best fitness value and the optimal value, not by that of the difference between the current best point and the optimal solution. More precisely, the convergence rate at the nth step $c_n \in (0, 1)$ by the conditional expectation of the change rate of the difference between the current best fitness value and the minimum fitness value, that is

$$c_n = E\left[\frac{|f_n - f_*|}{|f_{n-1} - f_*|}\middle|f_{n-1}\right] = E\left[\frac{f_n - f_*}{f_{n-1} - f_*}\middle|f_{n-1}\right]. \tag{24}$$

Thus we have

$$E[(f_n - f_*)|f_{n-1}] = c_n(f_{n-1} - f_*). \tag{25}$$

The advantage of this kind of definition for the convergence rate lies in that is can be applied to arbitrary objective functions. It can be observed from (24) that with given f_{n-1}, smaller $c_n \in (0, 1)$ results from smaller f_n or $|f_n - f_*|$, implying rapider decreasing of the f_n, i.e. the faster convergence speed of the algorithm. Let $\varepsilon_n = E(f_n - f_*)$ be the expected error at iteration $n \geqslant 0$. If there exists a constant $c \in (0, 1)$ called expected convergence rate, then $\varepsilon_n = c^n\varepsilon_0$ for every $n \geqslant 0$, which, by elementary transformation, leads to

$$n = \frac{\log_{10}(\varepsilon_n/\varepsilon_0)}{\log_{10}(c)} = -\frac{\Theta}{\log_{10}(c)}, \tag{26}$$

where $\Theta > 0$ denotes the orders of magnitude the error is to be decreased. If Θ is fixed, then the time n that is required to decrease the error by Θ orders of magnitude decreases as c decreases toward zero. Since the expected error after $K(\varepsilon)$ iterations is approximately ε, we therefore have $\varepsilon_{K(\varepsilon)} = c^{K(\varepsilon)}\varepsilon_0 \approx \varepsilon$, from which we can evaluate $K(\varepsilon)$ approximately by

$$K(\varepsilon) \approx \frac{\log_{10}(\varepsilon/\varepsilon_0)}{\log_{10}(c)} = \frac{\log_{10}(\varepsilon/E[f_0 - f_*])}{\log_{10}(c)} = \frac{\log_{10}\{\varepsilon/[E(f_0) - f_*]\}}{\log(c)} = \frac{-\Theta_0}{\log(c)}, \tag{27}$$

where $\Theta_0 = \log_{10}((E[f_0] - f_*)/\varepsilon) > 0$. The following theorem states the relationship between c and c_n.

Theorem 7. *Let $\bar{c} = \left(\prod_{i=1}^{n}\bar{c}_i\right)^{1/n}$, where $\bar{c}_i = E(c_i)$. If $\{c_n, n > 0\}$ and $\{f_n - f_*, n > 0\}$ are two negatively correlated (or positively correlated or uncorrelated) sequences of random variables, then $c < \bar{c}$ (or $c > \bar{c}$ or $c = \bar{c}$) (see the proof in Appendix A).*

Since the sequence $\{f_n - f_*, n > 0\}$ decreases with n, the negatively correlation between $\{c_n, n > 0\}$ and $\{f_n - f_*, n > 0\}$ implies that c_n increases or the convergence velocity decreases as f_n decreases. In this case, the convergence of f_n is called sub-linear. When $\{c_n, n > 0\}$ and $\{f_n - f_*, n > 0\}$ are positively correlated, c_n decreases as f_n decreases, which means that the convergence accelerates as f_n decreases, and thus we call that f_n is of super-linear convergence. When $\{c_n, n > 0\}$ and $\{f_n - f_*, n > 0\}$ are uncorrelated, $c = \bar{c}$ for all $n > 0$, implying that $c_n = c$, and the convergence of f_n is known as linear.

Linear convergence may occur in some idealized situations. For Pure Adaptive Search (PAS) [88], if the objective function is the Sphere function $f(X) = X^T \cdot X$, it may achieve linear convergence. Taking a 2-dimensional problem for instance, we have $E = [f_n - f_* \; |f_{n-1}] = E[f_n|f_{n-1}] = E[\|X_n\|^2|\|X_{n-1}\|^2]$. Denoting $r_{n-1}^2 = \|X_{n-1}\|^2$ and considering that X_n is distributed uniformly over $S_n = \left\{X : \|X\|^2 \leqslant r_{n-1}^2\right\}$, we obtain

$$E = [f_n - f_*|f_{n-1}] = \frac{1}{\pi r_{n-1}^2}\int\int_{S_n} r^2\,dX = \frac{1}{\pi r_{n-1}^2}\int_0^{2\pi} d\theta \int_0^{r_{n-1}} r^2 \cdot r\,dr = \frac{r_{n-1}^2}{2} = \frac{\|X_{n-1}\|^2}{2} = \frac{f_{n-1} - 0}{2} = \frac{f_{n-1} - f_*}{2}.$$

Thus it can be find that $c_n = 0.5$ for all $n > 0$ and $c = c_n = 0.5$, implying that linear convergence is achieved by the PAS.

Most of practical stochastic algorithms, however, run in sub-linear convergence in general. If we are to evaluate an algorithm in terms of the convergence rate, we can calculate the value of $\bar{c} = \left(\prod_{i=1}^{n} \bar{c}_i\right)^{1/n}$ and compare it with that of other algorithms, and besides, we may also compute the value of $|\bar{c} - c|$ to measure the "linearity" of its convergence. However, for most of the stochastic algorithms including QPSO, analytical calculation of their convergence rates is no less difficult than that of their time complexities. Therefore, in the remainder part of this section we turn our attention to the numerical results of convergence rate and time complexity on some specific problems.

4.3. Testing the time complexity and convergence rate of the QSPO algorithm

4.3.1. Evaluation of time complexity

Now, as it is hard to compute analytically the time complexity and convergence rate of QPSO. In this subsection, we test these convergence properties of QPSO empirically, using the Sphere function $f(X) = X^T \cdot X$, which has minimum value at zero. The Sphere function is unimodal and is a special instance from the class of quadratic functions with positive definite Hessian matrix. It has been generally used to test the convergence properties of a random search algorithm [74]. In our experiment, the initialization scope used by each algorithm for the function is $[-10, 10]^N$, where N is the dimension of the problem.

For evaluating the convergence of the algorithms, the first thing we require is a fair time measurement. The number of iterations cannot be accepted as a time measure since the algorithms perform different amount of work in their inner loops and also they have different population sizes. We have used the number of fitness function (objective function) evaluations as a measure of time. The advantage of measuring complexity by counting the function evaluations is that there is a strong relationship between this measure and the processor time as the function complexity increases. Therefore, the subscript m is used to denote the number of fitness function values, and the relationship $m = (n - 1)M + i$ holds, where n is the number of iterations, M is the population size and i is the particle's index.

We performed two sets of experiments to evaluate the time complexity and convergence rate of the QPSO algorithm, respectively. The QPSO-Type 1, QPSO-Type 2 and PSO with constriction factor [10] were tested for the comparison. To test the time complexity, we set the optimality region as $V(\varepsilon) = V(10^{-4}) = \{f: f \in V, f - f_* < 10^{-4}\}$ and recorded $K(\varepsilon)$, the number of function evaluations when the tested algorithm first reached the region. Each algorithm ran 50 times on the Sphere function with a certain dimension. We figured out some statistical results including the mean number of function evaluations $(\overline{K}(\varepsilon))$, the standard deviation of $K(\varepsilon)$ $(\sigma_{K(\varepsilon)})$, the standard error $(\sigma_{K(\varepsilon)}/\sqrt{50})$, the ratio of the standard error and the dimension $(\sigma_{K(\varepsilon)}/(N\sqrt{50}))$, and the ratio of $\overline{K}(\varepsilon)$ and the dimension $(\overline{K}(\varepsilon)/N)$. Tables 1–3 list, respectively, the results generated by QPSO-Type 1 with $\alpha = 1.00$, QPSO-Type 2 with $\alpha = 0.75$ and PSO with constriction factor $\chi = 0.73$ and acceleration coefficients $c_1 = c_2 = 2.05$. Each algorithm used 20 particles. The settings of α for both types of QPSO may lead to good performance in general, which has been demonstrated in our preliminary studies on a set of widely used benchmark functions.

The numerical results of $\overline{K}(\varepsilon)/N$ for QPSO-Type 2, listed in the last columns in Table 2, show to have a fairly stable values, with the maximum and minimum values being 154.8450 and 146.0333 respectively. Moreover, the correlation coefficient

Table 1
Statistical results of the time complexity test for QPSO-Type 1.

N	$\overline{K}(\varepsilon)$	$\sigma_{K(\varepsilon)}$	$\sigma_{K(\varepsilon)}/\sqrt{50}$	$\sigma_{K(\varepsilon)}/(N\sqrt{50})$	$\overline{K}(\varepsilon)/N$
2	232.16	49.24713	6.9646	3.4823	116.0800
3	382.78	67.11832	9.4920	3.1640	127.5933
4	577.92	92.87886	13.1351	3.2838	144.4800
5	741.08	104.8719	14.8311	2.9662	148.2160
6	921.92	128.5678	18.1822	3.0304	153.6533
7	1124.7	115.4569	16.3281	2.3326	160.6714
8	1396.2	170.9966	24.1826	3.0228	174.5250
9	1586.36	159.0606	22.4946	2.4994	176.2622
10	1852.86	185.6496	26.2548	2.6255	185.2860

Table 2
Statistical results of the time complexity test for QPSO-Type 2.

N	$\overline{K}(\varepsilon)$	$\sigma_{K(\varepsilon)}$	$\sigma_{K(\varepsilon)}/\sqrt{50}$	$\sigma_{K(\varepsilon)}/(N\sqrt{50})$	$\overline{K}(\varepsilon)/N$
2	306.26	69.41529	9.8168	4.9084	153.1300
3	455.06	71.15301	10.0626	3.3542	151.6867
4	619.38	60.03057	8.4896	2.1224	154.8450
5	748.92	69.87156	9.8813	1.9763	149.7840
6	883.92	94.45458	13.3579	2.2263	147.3200
7	1043.26	101.0267	14.2873	2.0410	149.0371
8	1171.68	102.8985	14.5520	1.8190	146.4600
9	1314.3	117.4271	16.6067	1.8452	146.0333
10	1477.2	112.9856	15.9786	1.5979	147.7200

Table 3
Statistical results of the time complexity test for PSO.

N	$\overline{K}(\varepsilon)$	$\sigma_{K(\varepsilon)}$	$\sigma_{K(\varepsilon)}/\sqrt{50}$	$\sigma_{K(\varepsilon)}/(N\sqrt{50})$	$\overline{K}(\varepsilon)/N$
2	679.2	126.4995	96.0534	48.0267	339.6000
3	967.8	146.133	136.8676	45.6225	322.6000
4	1235.3	137.2839	174.6978	43.6745	308.8250
5	1417.3	157.8286	200.4365	40.0873	283.4600
6	1686	174.2886	238.4364	39.7394	281.0000
7	1914.5	220.8512	270.7512	38.6787	273.5000
8	2083.3	186.5505	294.6231	36.8279	260.4125
9	2346.5	162.1101	331.8452	36.8717	260.7222
10	2525.2	200.4582	357.1172	35.7117	252.5200

between $\overline{K}(\varepsilon)$ and N in Table 2 was found to be 0.9997. The fact indicates that there is a strong linear correlation between $\overline{K}(\varepsilon)$ and the dimension, i.e. $\overline{K}(\varepsilon) = H \cdot N$, for the QPSO-Type 2. The constant H is function of the algorithm used and appears to be near 150 in Table 2. For QPSO-Type 1 and PSO, the correlation coefficients between $\overline{K}(\varepsilon)$ and N are 0.9967 and 0.9984, meaning that with the given algorithmic parameters, the linear correlations between $\overline{K}(\varepsilon)$ and N are not so remarkable as that for QPSO-Type 2, but the linearity for PSO is somewhat stronger than that for QPSO-Type 1. For further investigation, we visualize in Figs. 2–4 the results of each tested algorithm with other parameter settings. Fig. 2 shows that the value of $\overline{K}(\varepsilon)/N$ increases slowly as the dimension increases, implying that the time complexity may increase nonlinearly with the dimension. From Figs. 3 and 4, it can be seen that the time complexities of QPSO-Type 2 and PSO increase fairly linearly when dimension varies in the range of 2–20. However, both the QPSO-Type 1 and QPSO-Type 2 may have lower time complexity than PSO under the given parameter settings.

To explain the linear correlation we observed, let us consider an idealized random search algorithm with the constant probability $\rho(0 < \rho \leqslant 1)$ of improving objective function value. The algorithm is called Somewhat Adaptive Search (SAS), as in [88]. Denote the time complexity of PAS by $E[K_{PAS}(\varepsilon)]$, and by referring to [3], we can obtain that $E[K_{SAS}(\varepsilon)] = \frac{1}{\rho}E[K_{PAS}(\varepsilon)] = \ln\frac{\upsilon(S)}{\upsilon(V_\varepsilon)}$, where $\upsilon(\cdot)$ is the Lebesgue measure. For the testing problem, V_ε is an N-dimensional super-ball with radius $\sqrt{\varepsilon}$ and its volume is $\upsilon(V_\varepsilon) = \left[\pi^{N/2}/\Gamma\left(\frac{N}{2}+1\right)\right]\varepsilon^{\frac{N}{2}}$. Considering that $\ln\Gamma\left(\frac{N}{2}+1\right) = O(N)$, we obtain

$$E[K_{SAS}(\varepsilon)] = \frac{1}{\rho}\ln\frac{\delta^N}{\upsilon(V_\varepsilon)} = \frac{1}{\rho}\ln\left[\delta^N \cdot \Gamma\left(\frac{N}{2}+1\right)\Big/(\pi\varepsilon)^{\frac{N}{2}}\right] = \frac{N}{\rho}\ln\left[\frac{\delta}{\sqrt{\pi\varepsilon}}\right] + \ln\Gamma\left(\frac{N}{2}+1\right) = O(N), \quad (28)$$

where δ is the length of the search scope of each dimension. The above equation implies that the SAS has linear time complexity and the constant H is mainly determined by ρ if the values of δ and ε are given.

For the tested algorithm, the probability ρ generally varies with the number of function evaluations and dimension of the problem. Under certain parameter settings, the linear correlation between time complexity and dimension showed by QPSO-Type 2 or PSO indicates that ρ is relatively stable when the number of function evaluation is increasing and the dimension is varying in a certain interval. The value of ρ of QPSO-Type 1 seems to be less stable when the algorithm is running, leading the

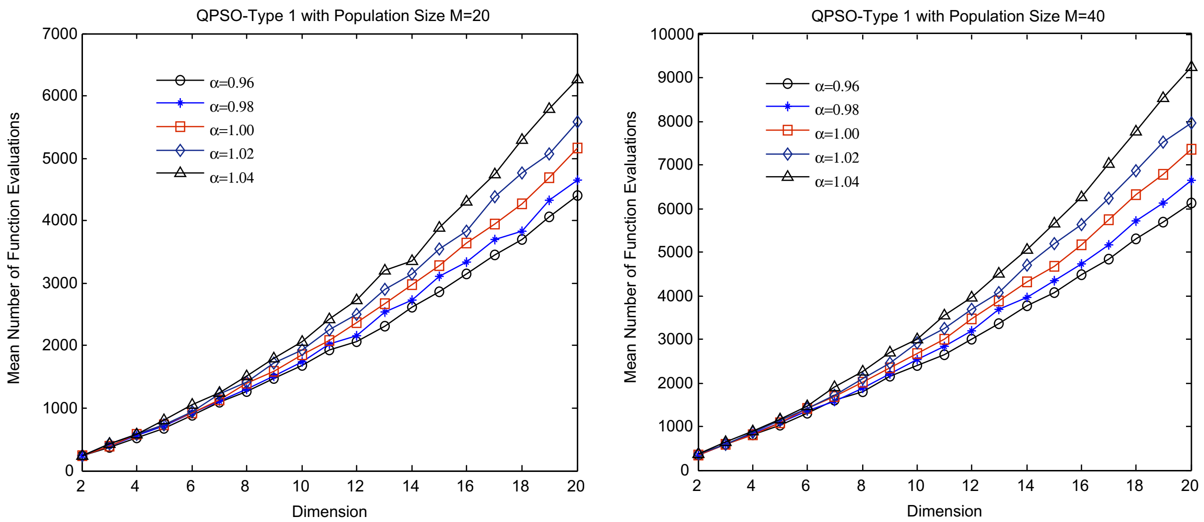

Fig. 2. Results of time complexity testing for QPSO-Type 1 with different values of α and population sizes.

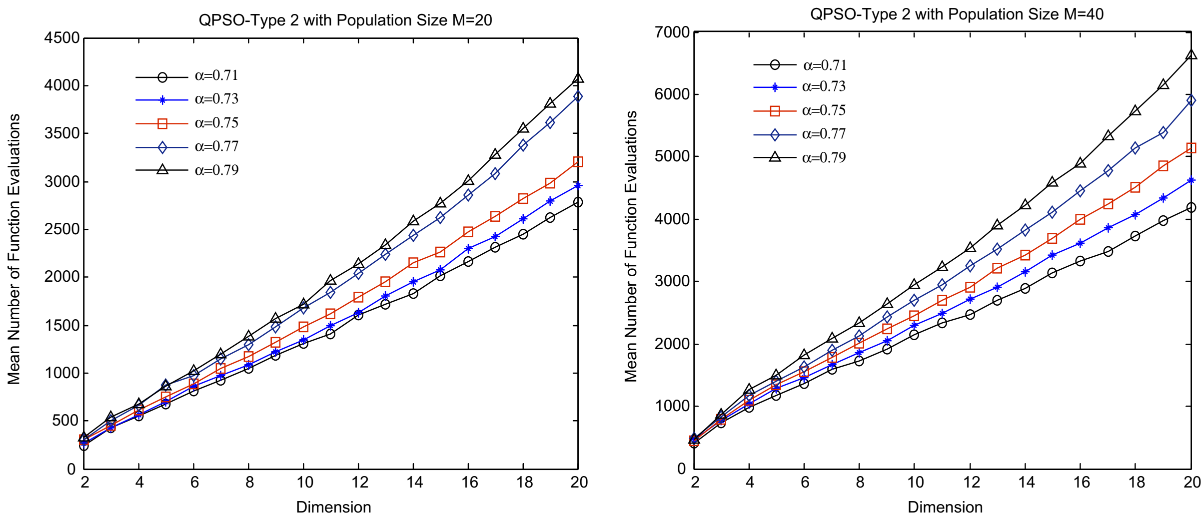

Fig. 3. Results of the time complexity testing for QPSO-Type 2 with different values of α and population sizes.

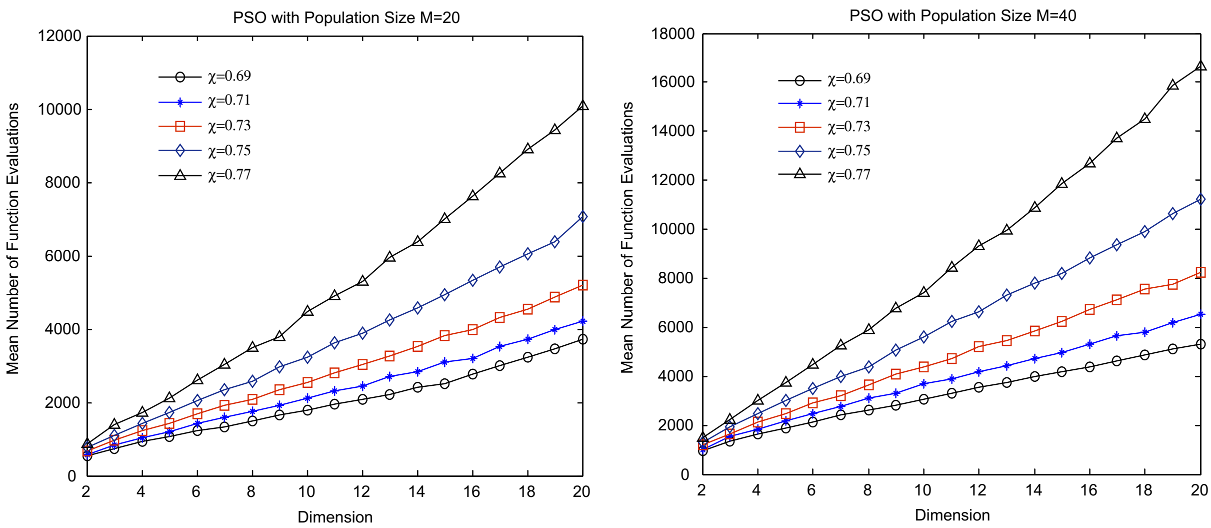

Fig. 4. Results of the time complexity testing for QPSO-Type 2 with different values of χ and population sizes.

time complexity to increase nonlinearly with the dimension. Nevertheless, since both types of QPSO may have larger ρ, the values of their H's are smaller than that of PSO.

4.3.2. Evaluation of the convergence rate

To test the convergence rate, we also had 50 trial runs for every instance with each run executed for a given maximum number of function evaluations, which were set as $m_{\max} = 200N$. For each algorithm, two groups of experiments were performed, one with population size $M = 20$, the other with $M = 40$. To compute the convergence rate, at the $(m-1)$th step of the function evaluation, we sampled the particle's position 30 times independently before the next update of the position. For each sample position, we calculated its objective function value denoted by f_m^k (where k is the number of sampled position), which in turn was compared with that of the global best position f_{m-1}. If $f_{m-1}^k < f_{m-1}$, we recorded f_m^k as it was; otherwise, we replaced it by f_{m-1}. After the 30th sampling, we computed $E[f_{m-1} - f_* | f_{m-1}] = \frac{1}{30}\sum_{k=1}^{30}(f_m^k - f_*)$ and thus could obtain the convergence rate c_m. It should be noticed that the sampling procedure did not affect the actual variables such as the particle's current position, its personal best position, the global best position, its velocity (for PSO algorithm) and so forth. After the sampling, the particle updated the position according to these variables.

For a certain m, the value of $\bar{c}_m$ was obtain by calculating the arithmetic mean of all the c_ms of 50 runs. Thus, $\bar{c}$ was obtained by using $\bar{c} = \left(\prod_{m=1}^{n_{\max}} \bar{c}_m\right)^{1/n_{\max}}$ and the expected convergence rate c was worked out by $c = (\bar{f}_{n_{\max}}/\bar{f}_0)^{1/n_{\max}}$. Besides, we also computed the correlation coefficient between c_m and f_m, and denoted it by $\theta(c_m f_m)$.

Tables 4–6 list the results generated by QPSO-Type 1 (α = 1.00), QPSO-Type 2 (α = 0.75) and PSO (χ = 0.729, $c_1 = c_2$ = 2.05). It can be seen from the three tables that for each algorithm on each problem, the value of $\bar{c}$ is larger than c and the correlation coefficient is negative. The fact indicates that the algorithms ran with sub-linear convergence. It can also be observed that the convergence rates of both types of QPSO (as shown in Tables 4 and 5) are smaller than that of PSO (as shown in Table 6) on each problem with the same population size except when N = 2. This implies that QPSO may converge faster. A closer look at the three tables reveals that $\theta(c_m, f_m)$ increases as the dimension increases. The reason may be that improvement of the function value was relatively harder when the dimension was high, making the convergence rate c_m so close to 1 that it changed little as f_m decreases. It should also be noted that for a given problem, when M = 40, the convergence rates of all the three algorithms are larger than those when M = 20. However, it cannot be concluded that a larger population size leads to a larger convergence rate, since the convergence rate also depends on other parameters. Smaller population size may result in a faster convergence on the problem with lower dimension, but the algorithm may encounter premature convergence when dimension is higher. On the other hand, although the algorithm with larger population size is not efficient on low-dimensional problems, it has less chance to result in premature convergence on high-dimensional problems. It can be inferred that if other parameters are given, when the dimension increases to a certain number, the convergence rate of the algorithm with smaller population size may exceed that with larger population size.

Table 4
Results for the convergence rate test of QPSO-Type 1.

N	n_{max}	M = 20					M = 40				
		$\bar{f}_0$	$\bar{f}_{n_{max}}$	$\bar{c}$	c	$\theta(c_m,f_m)$	$\bar{f}_0$	$\bar{f}_{n_{max}}$	$\bar{c}$	c	$\theta(c_m,f_m)$
2	400	6.3710	1.9311e−05	0.9772	0.9687	−0.1306	2.8309	4.5267e−05	0.9867	0.9728	−0.1513
3	600	19.1981	4.7244e−06	0.9810	0.9750	−0.1089	11.1715	2.8342e−04	0.9876	0.9825	−0.1341
4	800	32.2220	3.5506e−06	0.9842	0.9802	−0.0893	23.8846	2.5734e−04	0.9900	0.9858	−0.1333
5	1000	52.0046	6.6324e−06	0.9869	0.9843	−0.0734	44.3261	6.8896e−04	0.9912	0.9890	−0.1052
6	1200	74.4027	1.3265e−05	0.9887	0.9871	−0.0830	63.5427	6.9010e−04	0.9926	0.9905	−0.1120
7	1400	104.7848	1.3256e−05	0.9902	0.9887	−0.0870	82.9243	0.0012	0.9936	0.9921	−0.0999
8	1600	119.6822	1.7295e−05	0.9918	0.9902	−0.0745	103.6366	0.0027	0.9943	0.9934	−0.1009
9	1800	145.3857	7.2877e−05	0.9926	0.9920	−0.0710	127.3277	0.0032	0.9950	0.9941	−0.0924
10	2000	166.8939	6.9944e−05	0.9933	0.9927	−0.0581	139.2103	0.0046	0.9954	0.9949	−0.0848

Table 5
Results for the convergence rate test of QPSO-Type 2.

N	n_{max}	M = 20					M = 40				
		$\bar{f}_0$	$\bar{f}_{n_{max}}$	$\bar{c}$	c	$\theta(c_m,f_m)$	$\bar{f}_0$	$\bar{f}_{n_{max}}$	$\bar{c}$	c	$\theta(c_m,f_m)$
2	400	6.2432	2.2060e−05	0.9845	0.9691	−0.2165	3.5850	4.2072e−04	0.9883	0.9776	−0.2589
3	600	14.0202	1.2962e−05	0.9864	0.9771	−0.2134	12.6022	0.0012	0.9898	0.9847	−0.2304
4	800	33.3147	1.7636e−05	0.9867	0.9821	−0.1644	25.9027	0.0019	0.9920	0.9882	−0.2289
5	1000	50.7010	7.1771e−06	0.9881	0.9844	−0.1579	43.9053	0.0012	0.9927	0.9895	−0.1910
6	1200	68.6281	4.2896e−06	0.9889	0.9863	−0.1405	62.3872	0.0018	0.9933	0.9913	−0.1529
7	1400	100.3869	2.6875e−06	0.9899	0.9876	−0.1410	87.6376	0.0020	0.9939	0.9924	−0.1633
8	1600	132.1488	2.1607e−06	0.9910	0.9889	−0.1244	100.1323	0.0023	0.9945	0.9933	−0.1700
9	1800	144.2586	1.5950e−06	0.9915	0.9899	−0.1090	119.0672	0.0013	0.9951	0.9937	−0.1329
10	2000	172.0399	1.7459e−06	0.9921	0.9908	−0.0782	154.5341	0.0017	0.9951	0.9943	−0.1274

Table 6
Results for the convergence rate test of PSO.

N	n_{max}	M = 20					M = 40				
		$\bar{f}_0$	$\bar{f}_{n_{max}}$	$\bar{c}$	c	$\theta(c_m,f_m)$	$\bar{f}_0$	$\bar{f}_{n_{max}}$	$\bar{c}$	c	$\theta(c_m,f_m)$
2	400	6.2601	0.0050	0.9911	0.9823	−0.2151	2.9063	0.0252	0.9938	0.9882	−0.2362
3	600	18.2950	0.0061	0.9921	0.9867	−0.1664	10.1600	0.0636	0.9948	0.9916	−0.1726
4	800	36.3027	0.0053	0.9921	0.9890	−0.1333	22.8484	0.0830	0.9954	0.9930	−0.1478
5	1000	56.0108	0.0045	0.9932	0.9906	−0.1033	39.7976	0.1222	0.9954	0.9942	−0.1223
6	1200	75.1300	0.0042	0.9936	0.9919	−0.0812	60.8450	0.1102	0.9961	0.9948	−0.1088
7	1400	92.8699	0.0043	0.9940	0.9929	−0.0621	80.6101	0.1425	0.9964	0.9955	−0.0902
8	1600	125.3514	0.0036	0.9946	0.9935	−0.0731	98.5102	0.1624	0.9967	0.9960	−0.0608
9	1800	153.0013	0.0035	0.9949	0.9941	−0.0530	123.9556	0.1989	0.9969	0.9964	−0.0723
10	2000	170.3619	0.0034	0.9952	0.9946	−0.0446	135.5715	0.2071	0.9971	0.9968	−0.0636

5. Two improved QPSO algorithms

Although the QPSO method, particularly the QPSO-Type 2, has been showed to be efficient in solving continuous optimization problem, there is the possibility of improving the algorithm without increasing the complexity of its implementation. Here, we proposed two improved versions of the algorithm based on the QPSO-Type 2.

5.1. QPSO with random mean best position

In the first improved QPSO algorithm, the mean best position C in (15) is replaced by the *pbest* position of a randomly selected particle in the population at each iteration. For convenience, we denoted the randomly selected *pbest* position by C'_n. For each particle, the probability for its personal best position to be selected as C'_n is $1/M$. Consequently, the expected value of C'_n equals to C_n, that is,

$$E(C'_n) = \sum_{i=1}^{M} \frac{1}{M} P_{i,n} = C_n. \tag{29}$$

However, since the C'_n appears to be more changeful than C_n, the current position of each particle at each iteration shows to be more volatile than that of the particle in QPSO-Type 2, which diversifies the particle swarm and in turn enhances the global search ability of the algorithm. This improved algorithm is called QPSO with random mean best position (QPSO-RM).

5.2. QPSO with ranking operator

The second improvement involves a ranking operator proposed to enhance the global search ability of the QPSO algorithm. In the original QPSO and QPSO-RM, the neighborhood topology is the global best model so that each particle follows

Procedure of the QPSO-RO Algorithm:
Begin
Initialize the current position $X_{i,0}^j$ and the personal best position $P_{i,0}^j$ of each particle, evaluate their fitness values and find the global best position G_0; Set n=0.
While (termination condition = false)
Do
Set n=n+1;
Compute mean best position C_n;
Select a suitable value for α;
for (i=1 to M)
Update $P_{i,n}$ and rank the particles in ascending order according to the fitness values of their *pbest* positions;
Randomly select a particle q among the particles whose ranks are larger than $rank_i$ with the given selection probability in (31);
for j=1 to N
$\varphi_{i,n}^j = \text{rand1}(\cdot)$;
$p_{i,n}^j = \varphi_{i,n}^j P_{i,n}^j + (1-\varphi_{i,n}^j) P_{q,n}^j$;
$u_{i,n}^j = \text{rand2}(\cdot)$;
if (rand3(·) < 0.5)
$X_{i,n+1}^j = p_{i,n}^j + \alpha |X_{i,n}^j - p_{i,n}^j| \ln(1/u_{i,n+1}^j)$;
else
$X_{i,n+1}^j = p_{i,n}^j - \alpha |X_{i,n}^j - p_{i,n}^j| \ln(1/u_{i,n+1}^j)$;
end if
end for
Evaluate the fitness value of $X_{i,n+1}$, that is, the objective function value $f(X_{i,n+1})$;
Update $P_{i,n}$ and G_n
end for
end do
end

Fig. 5. The procedure of the QPSO-RO algorithm.

their own *pbest* position and the *gbest* position, leading the algorithm to fast convergence. However, the particle may be misguided by the *gbest*position if it is located at a local optimal point, particularly at the later stage of the search process. In such a case, the particles are all pulled toward the *gbest* position and have less opportunity to escape the local optimum even though some particles are located in promising regions where the better solutions or the global optimal solution can be found. As a result, the QPSO using this topology may be prone to encounter premature convergence.

In the proposed QPSO with ranking operator (QPSO-RO), each particle flies in the search space following its own *pbest* position and the *pbest*position of a randomly selected particle based on a ranking operator, whose fitness value is better than the considered particle. The selection procedure is as follows. Before the position update for particle *i* at every iteration, the *pbest* positions of all the particles are ranked in ascending order according to their fitness values, with the rank of the global best particle being *M* and that of the worst particle being 1. Given that the rank of the considered particle is $rank_i(1 \leqslant rank_i \leqslant M)$, each particle whose rank is larger than $rank_i$ will be considered as a candidate to be selected but the other particles will not be selected. In other words, the selection probability of particle $q(1 \leqslant q \leqslant M)$ is given by

$$PS_q = \begin{cases} \frac{2\times rank_q}{rank_i(rank_i-1)}, & rank_q > rank_i \\ 0, & rank_q \leqslant rank_i \end{cases}, \tag{30}$$

where $rank_i$ is the rank of particle *i*. From Eq. (30), it can be seen that the sum of the selection probabilities of all the candidates equals 1. For particle *i*, if the *pbest* position of particle *q* is selected, then the coordinates of the local attractor of particle *i* is determined by

$$p_{i,n}^j = \varphi_{i,n}^j P_{i,n}^j + \left(1 - \varphi_{i,n}^j\right) P_{q,n}^j, \quad \varphi_{i,n}^j \sim U(0,1), \tag{31}$$

where $P_{q,n}^j$ is the *j*th component of the *pbest* position of particle *q* with $rank_q > rank_i$.

Eq. (31) implies that particle *i* is attracted by both its own *pbest* position and the randomly selected $P_{q,n}$. Although the global best particle is selected with the highest probability, there is a better chance for the other particles to be selected as a part of the local attractor. This helps the particle swarm search other promising region and consequently enhance the global search ability of the algorithm. The procedure of the QPSO-RO algorithm is outlined in Fig. 5. Also note that rand$i(\cdot)$, $i = 1, 2, 3$, is used to denote random numbers generated uniformly and distributed on (0, 1).

6. Experiments on benchmark functions

Section 3 has theoretically proven that the QPSO algorithm is global convergent, which, however, is not sufficient to draw a conclusion that the QPSO is effective in real-world applications. Semi-theoretical evaluation of time-complexity and convergence rate of QPSO in Section 4 reveals that it has lower computational complexity and better convergence properties for the Sphere function, which is a unimodal function usually used to test the local search ability of an algorithm. Nevertheless, it is hard to generalize the same evaluation method to an arbitrary problem, particularly when the problem is multimodal, and it is inconclusive with respect to the overall performance of the algorithms using the Sphere function only. Hence, to evaluate the QPSO objectively, it would be better to compare it with other PSO variants using a large test set of optimization functions.

The goal of this section is thus to determine the overall performance of QPSO by using the first ten functions from the CEC2005 benchmark suite [77]. Furthermore, the proposed QPSO-RM and QPSO-RO were also experimented on these benchmark functions. A performance comparison was made among the QPSO algorithms (QPSO-Type 1 and QPSO-Type 2), QPSO-RM, QPSO-RO and other forms of PSO, including PSO with inertia weight (PSO-In) [68–70], PSO with constriction factor (PSO-Co) [10,11], the Standard PSO [4], Gaussian PSO [64], Gaussian Bare Bones PSO [35,36], Exponential PSO (PSO-E) [39], Lévy PSO [56], comprehensive learning PSO (CLPSO) [43], dynamic multiple swarm PSO (DMS-PSO) [42] and fully-informed particle swarm (FIPS) [46].

F_1 to F_5 of the CEC 2005 benchmark suite are unimodal, while functions F_6 to F_{10} are multi-modal. Each algorithm ran 100 times on each problem using 20 particles to search the global best fitness value. At each run, the particles in the algorithms started in new and randomly-generated positions, which are uniformly distributed within the search bounds. Every run of each algorithm lasted 3000 iterations and the best fitness value (objective function value) for each run was recorded.

For the QPSO-based algorithms, two methods of controlling α were used. One is the fixed-value method, in which the value of α was fixed at a constant during the search process. The other is time-varying method, in which the value of α decreased linearly in the course of running. For QPSO-Type 1 with the fixed-value method, α was fixed at 1.0, while for QPSO-Type 1 with the time-varying method, α decreased linearly form 1.0 to 0.9. For QPSO-Type 2 with the fixed-value method, the value of α was fixed at 0.75, while for QPSO-Type 2 with the time-varying method, α decreased linearly from 1.0 to 0.5 with regard to the iteration number. For QPSO-RM, α was set to be 0.54 when the fixed-value method was used, and decreased linearly from 0.6 to 0.5 when the time-varying method was used. For QPSO-RO, α was fixed at 0.68 and decreased linearly from 0.9 to 0.5 for the two parameter control methods, respectively. The parameter configurations for the QPSO-based algorithms were recommended according to our preliminary experiments or by the existing publications [80,81]. The other parameters of the remainder PSO variants were configured as recommended by the corresponding publications.

The mean best fitness values and standard deviations out of 100 runs of each algorithm on F_1 to F_5 are presented in Table 7 and those on F_6 to F_{10} in Table 8. To investigate if the differences in mean best fitness values between algorithms were significant, the mean values for each problem were analyzed using a multiple comparison procedure, an ANOVA (Analysis Of Variance), with 0.05 as the level of significance. The procedure employed in this work is called the "stepdown" procedure, which takes into account that all but one of the comparisons are less different than the range. When doing all pairwise comparisons, this approach is the best available if confidence intervals are not needed and the sample sizes are equal [17].

The algorithms were ranked to determine which algorithm could be reliably said to be the most effective for each problem. The algorithms that were not statistically different to each other were given the same rank; those that were not statistically different to more than one other groups of algorithms were ranked with the best-performing of these groups. For each algorithm, the resulting rank for each problem, the total rank and the average rank are shown in Table 9.

For the Shifted Sphere Function (F_1), QPSO-RM with either fixed or time-varying α generated better results than other methods. The results for the Shifted Schwefel's Problem 1.2 (F_2) show that, QPSO-RM with fixed α yielded the best results, but the performances of PSO-In and QPSO-Type 2 with linearly decreasing α were inferior to those of the other competitors. For Shifted Rotated High Conditioned Elliptic Function (F_3), when using fixed α, both the QPSO-RO and QPSO-RM outperformed the other methods in a statistical significance manner. QPSO-RO with fixed α also showed to be the winner among all the tested algorithms for the Shifted Schwefel's Problem 1.2 with Noise in Fitness (F_4). F_5 is the Schwefel's Problem 2.6 with Global Optimum on the Bounds, and for this benchmark, QPSO-RO with time-varying α yielded the best results. For benchmark F_6, the Shifted Rosenbrock Function, the QPSO-based algorithms except QPSO-Type 2 were superior to those of the other algorithms, among which there was no statistically significant difference except PSO-In and DMS-PSO. The results for the Shifted Rotated Griewank's Function without Bounds (F_7) suggest that QPSO-RM, either with fixed α or with time-varying α, was able to find the solution for the function with the best quality compared to the other methods. Benchmark F_8 is the Shifted Rotated Ackley's Function with Global Optimum on the Bounds. The QPSO-RM with time-varying α showed the best performance for this problem among the competitors. It can be seen that the performance differences between the QPSO-based algorithms and PSO-In are not statistically significant. The Shifted Rastrigin's Function (F_9) is a separable function, which the CLPSO algorithm was good at solving it. However, it can be observed that the QPSO-RO obtained

Table 7
Experimental results of mean best fitness values and standard deviations by algorithms and problems, F_1 to F_5 (best results in bold).

Algorithms	F_1	F_2	F_3	F_4	F_5
In-PSO (Std. Dev.)	3.8773e−013	785.0932	3.9733e+07	1.1249e+04	6.0547e+03
	(1.6083e−012)	(661.2154)	(4.6433e+07)	(5.4394e+03)	(2.0346e+03)
PSO-Co	1.5713e−026	0.1267	8.6472e+06	1.3219e+04	7.6892e+03
	(1.4427e−025)	(0.3796)	(9.1219e+06)	(6.0874e+03)	(2.3917e+03)
Standard PSO	8.2929e−026	78.2831	6.6185e+06	1.3312e+04	6.2884e+03
	(1.2289e−025)	(52.3272)	(3.0124e+06)	(4.1076e+03)	(1.4318e+03)
Gaussian PSO	7.3661e−026	0.0988	1.1669e+07	2.3982e+04	8.0279e+03
	(5.9181e−025)	(0.3362)	(2.5153e+07)	(1.2512e+04)	(2.3704e+03)
Gaussian Bare Bones PSO	1.7869e−025	16.8751	7.7940e+06	1.1405e+04	9.5814e+03
	(8.4585e−025)	(16.2021)	(4.3240e+06)	(6.7712e+03)	(3.0227e+03)
PSO-E	5.2531e−024	20.2750	6.2852e+06	8.2706e+03	7.2562e+03
	(2.2395e−023)	(15.2414)	(2.8036e+06)	(3.6254e+03)	(1.8666e+03)
Lévy PSO	1.1880e−024	36.9986	1.7366e+07	7.4842e+03	8.2543e+03
	(1.1455e−023)	(29.1360)	(1.9001e+07)	(6.6588e+03)	(2.2297e+03)
CLPSO	3.5515e−008	5.3394e+03	5.1434e+07	1.6069e+04	5.4958e+003
	(2.2423e−008)	(1.2207e+03)	(1.3489e+07)	(3.4776e+03)	(888.9618)
DMS-PSO	7.2525e−006	844.9978	1.2841e+07	2.7125e+003	2.9189e+003
	(2.2114e−005)	(350.2620)	(4.9745e+06)	(972.8958)	(811.5164)
FIPS	3.3157e−027	75.4903	1.0409e+07	1.0529e+04	4.3452e+003
	(2.5732e−028)	(76.1305)	(4.4786e+06)	(3.8510e+03)	(978.6149)
QPSO-Type 1 (α = 1.00)	3.5936e−028	40.2282	4.8847e+06	6.2397e+03	8.0749e+03
	(1.5180e−028)	(23.3222)	(2.1489e+06)	(2.4129e+03)	(1.7099e+03)
QPSO-Type 1 (α = 1.00 → 0.90)	5.0866e−029	4.5003	3.2820e+06	6.4303e+03	7.8471e+03
	(4.4076e−029)	(2.9147)	(1.9953e+06)	(2.9744e+03)	(1.7878e+03)
QPSO-Type 2 (α = 0.75)	1.9838e−027	0.1771	1.6559e+06	3.1321e+03	5.7853e+03
	(5.2716e−028)	(0.1137)	(7.1264e+05)	(2.0222e+03)	(1.2483e+03)
QPSO-Type 2 (α = 1.0 → 0.5)	1.2672e−027	120.6051	4.4257e+06	4.0049e+03	3.3684e+003
	(3.7147e−028)	(62.2340)	(2.3302e+06)	(2.7218e+03)	(975.6551)
QPSO-RM (α = 0.54)	**3.1554e−036**	**0.0715**	1.8544e+06	3.1443e+03	5.7144e+03
	(**2.3913e−036**)	(**0.0530**)	(6.4710e+05)	(3.8785e+03)	(1.4898e+003)
QPSO-RM (α = 0.6 → 0.5)	2.6728e−035	1.4099	2.1737e+06	2.1835e+003	4.3398e+03
	(6.5932e−035)	(7.8582)	(1.0089e+06)	(2.8487e+003)	(1.4313e+03)
QPSO-RO (α = 0.68)	1.5414e−027	0.1784	**1.6309e+006**	**1.9489e+003**	5.2202e+003
	(2.9964e−028)	(0.1217)	(**8.7302e+005**)	(**1.6002e+003**)	(1.2661e+003)
QPSO-RO (α = 0.9 → 0.5)	1.0747e−027	50.9939	4.7718e+006	2.1540e+003	**2.7469e+003**
	(2.3154e−028)	(48.6055)	(2.0760e+006)	(1.3635e+003)	(**723.4961**)

Table 8
Experimental results of mean best fitness values and standard deviations by algorithms and problems, F_6 to F_{10} (best results in bold).

Algorithms	F_6	F_7	F_8	F_9	F_{10}
In-PSO (Std. Dev.)	263.7252	0.9907	0.0414	39.5528	239.5814
	(437.4145)	(4.7802)	(0.2393)	(16.1654)	(72.2521)
Co-PSO	123.0243	0.0255	5.1120	96.7296	171.6488
	(266.2520)	(0.0327)	(4.5667)	(28.0712)	(58.5713)
Standard PSO	153.5178	0.0218	0.2744	79.1219	128.9865
	(246.1049)	(0.0165)	(0.6795)	(20.2619)	(32.3662)
Gaussian PSO	150.7872	0.0224	2.7722	103.6245	184.2657
	(303.3368)	(0.0178)	(1.4603)	(28.6113)	(57.3675)
Gaussian Bare Bones PSO	144.1377	0.0205	3.5460	80.9496	164.2914
	(165.2616)	(0.0208)	(6.1929)	(22.0621)	(72.8542)
PSO-E	189.8292	0.0493	3.5881	66.5112	163.7187
	(375.8636)	(0.0538)	(5.5286)	(20.9853)	(55.0921)
Lévy PSO	133.9526	0.0446	2.2168	74.0446	154.3838
	(293.8460)	(0.1182)	(1.3575)	(21.6913)	(76.3070)
CLPSO	117.3987	2.4151	1.1582e−04	**0.6990**	151.2854
	(54.8846)	(0.7533)	(6.7878e−05)	(**0.7983**)	(23.4628)
DMS-PSO	296.0911	0.3985	0.1213	39.9694	**112.8426**
	(347.1682)	(0.2502)	(0.3716)	(10.2384)	(**71.2957**)
FIPS	188.8304	0.0330	0.3843	64.6289	198.3699
	(294.0374)	(0.0464)	(0.5713)	(14.5907)	(21.7958)
QPSO-Type 1 (α = 1.00)	138.0746	0.0218	0.1217	56.4232	137.0334
	(209.1735)	(0.0204)	(0.4504)	(16.7090)	(38.5269)
QPSO-Type 1 (α = 1.10 → 0.90)	139.9815	0.0209	0.0916	54.4278	126.1298
	(206.8138)	(0.0203)	(0.3166)	(16.6044)	(44.9531)
QPSO-Type 2 (α = 0.75)	82.9908	0.0203	0.0683	39.0991	128.5351
	(119.836)	(0.0164)	(0.3080)	(12.4904)	(57.6255)
QPSO-Type 2 (α = 1.0 → 0.5)	88.0494	0.0208	2.0961e−014	29.9218	118.4549
	(159.7481)	(0.0130)	(1.9099e−014)	(10.5736)	(53.0216)
QPSO-RM (α = 0.54)	105.7474	0.0163	0.0762	42.4817	185.6351
	(155.4583)	(0.0134)	(0.3075)	(12.1384)	(46.6356)
QPSO-RM (α = 0.6 → 0.5)	89.6543	**0.0150**	**7.5318e−015**	43.8327	207.0548
	(151.6908)	(**0.0119**)	(**1.7046e−015**)	(17.881)	(14.4658)
QPSO-RO (α = 0.68)	63.9916	0.0219	0.0536	34.5288	159.9417
	(65.7906)	(0.0292)	(0.2653)	(15.0725)	(36.2107)
QPSO-RO (α = 0.9 → 0.5)	**61.0752**	0.0196	1.9611e−014	23.3014	143.4452
	(**72.2629**)	(0.0152)	(1.5498e−014)	(7.6051)	(43.9709)

Table 9
Ranking by algorithms and problems.

Algorithms	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	F_9	F_{10}	Total rank	Average rank
PSO-In	16	=16	17	=12	=9	=17	17	=1	=5	18	128	12.8
PSO-Co	=10	=1	=11	=15	=11	=7	=4	18	=17	=11	105	10.5
Standard PSO	=10	=13	=9	=15	=11	=7	=4	=10	=14	=1	94	9.4
Gaussian PSO	=10	=1	=11	18	=15	=7	=4	=14	=17	=11	108	10.8
Gaussian Bare Bones PSO	=13	=8	=11	=12	18	=7	=4	=14	=14	=11	112	11.2
PSO-E	=13	=8	=9	11	=11	=7	=13	=14	=12	=11	109	10.9
Lévy PSO	15	=10	16	=8	=15	=7	=13	=14	=14	=6	118	11.8
CLPSO	17	18	18	17	=6	=7	18	=10	1	=6	118	11.8
DMS-PSO	18	=16	=11	=4	=1	=17	16	=1	=5	=1	90	9.0
FIPS	9	=13	=11	=12	=4	=7	=13	13	=12	16	110	11.0
QPSO-Type 1 (α = 1.00)	4	=10	=6	=8	=15	=7	=4	=10	=10	=6	80	8.0
QPSO-Type 1 (α = 1.00 → 0.90)	3	7	5	=8	=11	=7	=4	=1	=10	=1	57	5.7
QPSO-Type 2 (α = 0.75)	8	=4	=1	=4	=9	=1	=4	=1	=5	=1	38	3.8
QPSO-Type 2 (α = 1.00 → 0.5)	6	15	=6	7	3	=1	=4	=1	=3	=1	47	4.7
QPSO-RM (α = 0.54)	1	=1	=1	=4	=6	=1	=1	=1	=5	17	38	3.8
QPSO-RM (α = 0.6 → 0.5)	2	=4	4	=1	=4	=1	=1	=1	=5	=11	34	3.4
QPSO-RO (α = 0.68)	7	=4	=1	=1	=6	=1	=4	=1	=3	=6	34	3.4
QPSO-RO (α = 0.9 → 0.5)	5	=10	=6	=1	=1	=1	=1	=1	2	=6	34	3.4

the better performance than the other QPSO-based algorithms. F_{10} is the Shifted Rotated Rastrigrin's Function, which appears to be a more difficult problem than F_9. For this benchmark, the QPSO-Type 2, QPSO-Type 1 with time-varying α and the standard PSO outperformed the other competitors in a statistically significant manner.

Table 9 shows that the QPSO-RO obtained a better overall performance than all the other tested algorithms, for the total and average ranks of QPSO-RO with both parameter control methods are smaller than those of the other algorithms.

Although the total and average ranks of QPSO-RO with fixed α and QPSO-RO with time-varying α are the same, the former obtained relatively more stable performance than the latter. It can be observed that QPSO-RO with fixed α yielded the best result for four of the tested benchmark problems in a statistically significant manner, with the worst rank being 6 for F_5 and F_{10}, and that QPSO-RO with time-varying αgenerated the best result for half of all the tested functions, and the worst rank is 10 for F_2. Compared with the QPSO algorithm, including QPSO-Type 1 and QPSO-Type 2, the QSPO-RO achieved a remarkable improvement of the overall algorithmic performance.

The second best-performing algorithm was the QPSO-RM algorithm, as indicated by the total and average ranks. Between the two parameter control methods, QPSO-RM with the time-varying α yielded a comparable overall performance with QPSO-RO. It can be seen that the performance of QPSO-RM with time-varying $\alpha$is more stable than that of QPSO-RM with fixed α, for the worst ranks of the two versions of QPSO-RM are 11 and 17 for F_{10}, respectively. It is obvious that QPSO-RM has slightly better overall performance than the two types of the original QPSO.

The two types of the original QPSO algorithm, as shown by the total ranks, achieved better overall performance than other PSO variants. Besides the evaluation of the convergence rate and time complexity of QPSO, these results further provided stronger evidence that the QPSO is a promising tool for optimization problems. Between QPSO-Type 2 and QPSO-Type 1, the former showed to have a better and more stable overall performance than the latter. Among the other PSO variants, the DMS-PSO and the standard PSO yielded better overall performance than the remainder competitors. It is evident from the ranking list that the standard PSO and PSO-Co were two great improvements over the PSO-In algorithm, which did not show comparable performance with the other competitors. The other four probabilistic algorithms did not work so effectively as the QPSO-based algorithm, DMS-PSO and the standard PSO. What should be noticed is that the CLPSO is very effective in solving separable functions such as F_9, but has slower convergence speed, as has been indicated in the related publication [43].

7. Conclusions

In this paper, we first investigated the convergence of the QPSO algorithm by establishing a Menger space for the algorithm, in which the algorithm is shown to be a contraction mapping, and its orbit is probabilistic bounded. Thus, we proved the fixed point theorem of the QPSO algorithm in the Menger space, showing that the algorithm converges to the global optimum in probability.

Then, the effectiveness of the algorithm was evaluated by time complexity on the Sphere function. The linear correlation between complexity and the dimension of the problem was observed and analyzed. Besides, we also evaluated the performance of the QPSO algorithm with respect to the convergence rate, which was defined by the ratio of conditional expectation of the distance of objective function value to the global optimum at the next iteration and the distance at the current iteration. It was found that the algorithms ran in sub-linear convergence and the QPSO had smaller convergence rate than the PSO, which means the QPSO can converge faster with given parameters.

Two improvements of the QPSO, the QPSO-RO and QPSO-RM were proposed next. The QPSO algorithm, along with the two improved versions and other PSO variants were tested on a set of benchmark problems for an overall performance evaluation. The experimental results show that the QPSO is comparable with or even better than other forms of PSO in finding the optimal solutions of the tested benchmark functions, and also show that the two modified QPSO algorithms achieved remarkable improvement over the original QPSO algorithm.

Acknowledgements

This work is supported by the National Natural Science Foundation of China (Project Numbers: 61170119, 60973094), by the Natural Science Foundation of Jiangsu Province, China (Project Number: BK2010143), by the Fundamental Research Funds for the Central Universities (Project Numbers: JUSRP21012, JUSRP11023), and by the Foundation of Key Laboratory of Advanced Process Control for Light Industry (Jiangnan University), Ministry of Education, PR China.

Appendix A

Theorem 1. *Let a self-mapping T: $(E, F, \Delta) \to (E, F, \Delta)$ be the contraction mapping in Definition 7. If for every $x \in E$, $O_T(x; 0, \infty)$ is probabilistic bounded, then there exists a unique common fixed point x_* in E for T, and for every $x_0 \in E$, the iterative sequence $\{T^n x_0\}$ converges to x_* in $\mathcal{T}$.*

Proof. The proof of the theorem is achieved through the following two steps.

(1) First, we prove that for every $x_0 \in E$, the sequence $\{x_m\}_{m=0}^{\infty}$ is a $\mathcal{T}$-Cauchy Sequence in E, where

$$\{x_m\}_{m=0}^{\infty} = \{x_0, x_1 = T^{n(x_0)}x_0, \ldots, x_{m+1} = T^{n(x_m)}x_m, \ldots\}. \tag{A1}$$

Let $n_i = n(x_i)$, where $i = 0, 1, 2, \ldots$, and let m and i are two arbitrary positive integers. From (18) and (A1), we have

$$F_{x_m,x_{m+i}}(t) \geqslant F_{T^{n_{m-1}}x_{m-1},T^{n_{m+i-1}+\cdots+n_m+n_{m-1}}x_{m-1}}(t) \geqslant F_{x_{m-1},T^{n_{m+i-1}+\cdots+n_m}x_{m-1}}\left(\frac{t}{k}\right) \geqslant F_{x_{m-2},T^{n_{m+i-1}+\cdots+n_m}x_{m-2}}\left(\frac{t}{k^2}\right) \geqslant \cdots$$
$$\geqslant F_{x_0,T^{n_{m+i-1}+\cdots+n_m}x_0}\left(\frac{t}{k^m}\right) \geqslant \inf_{z\in\{T^s x_0\}_{s=0}^{\infty}} F_{x_0,z}\left(\frac{t}{k^m}\right) \geqslant \sup_{u<t/k^m}\inf_{z\in\{T^s x_0\}_{s=0}^{\infty}} F_{x_0,z}(u), \quad \forall t \geqslant 0. \tag{A2}$$

Considering that $\forall t \geqslant 0$, $\frac{t}{k^m}$ is strictly increasing with t and $\lim_{m\to\infty}\frac{t}{k^m}=\infty$, and that $O_T(x_0; 0, \infty)$ is probabilistic bounded, from (A2) we can obtain

$$\lim_{m\to\infty} F_{x_m,x_{m+i}}(t) \geqslant \lim_{m\to\infty}\sup_{u<t/k^m}\inf_{z\in\{T^s x_0\}_{s=0}^{\infty}} F_{x_0,z}(u) = \begin{cases}\sup\limits_{u>0}\inf\limits_{z\in\{T^s x_0\}_{s=0}^{\infty}} F_{x_0,z}(u), & \text{if} \quad t>0,\\ 0, & \text{if} \quad t=0,\end{cases} = \begin{cases}1, & \text{if} \quad t>0,\\ 0, & \text{if} \quad t=0,\end{cases} \tag{A3}$$

which means that the sequence $\{x_m\}_{m=0}^{\infty}$ is a $\mathcal{T}$-Cauchy Sequence in E. Thus (E,F,Δ) is $\mathcal{T}$-Complete and there exists a point x_* in E such that $x_n \xrightarrow{\mathcal{T}} x_*$.

(2) Now we prove that x_* is the fixed point of T^{n_*}, where $n_* = n(x_*)$. For any positive integer i and $\forall t \geqslant 0$, according to (18), the following inequality holds.

$$F_{x_i,T^{n_*}x_i}(t) \geqslant F_{x_{i-1},T^{n_*}x_{i-1}}\left(\frac{t}{k}\right) \geqslant \cdots \geqslant F_{x_0,T^{n_*}x_0}\left(\frac{t}{k^i}\right) \geqslant \sup_{u<t/k^i} F_{x_0,T^{n_*}x_0}(u). \tag{A4}$$

Taking the limit as $i \to \infty$ on both sides of the above inequality, we find that

$$\lim_{i\to\infty} F_{x_i,T^{n_*}x_i}(t) \geqslant \lim_{i\to\infty}\sup_{u<t/k^i} F_{x_0,T^{n_*}x_0}(u) \geqslant \begin{cases}\sup\limits_{u>0} F_{x_0,T^{n_*}x_0}(u), & \text{if} \quad t>0\\ 0 & \text{if} \quad t=0\end{cases} = \begin{cases}1 & \text{if} \quad t>0,\\ 0 & \text{if} \quad t=0.\end{cases} \tag{A5}$$

Since $x_n \xrightarrow{\mathcal{T}} x_*$, $\lim_{n\to\infty} F_{x_n,x_*}(t) = H(t)$. Thus we have

$$\lim_{i\to\infty} F_{x_i,T^{n_*}x_i}(t) \geqslant \lim_{i\to\infty} \Delta\left(F_{x_*,x_i}\left(\frac{t}{2}\right), F_{x_i,T^{n_*}x_i}\left(\frac{t}{2}\right)\right) = 1, \quad \forall t>0. \tag{A6}$$

From (18) and (A6), we obtain

$$F_{x_*,T^{n_*}x_*}(t) \geqslant \Delta\left(F_{x_*,T^{n_*}x_i}\left(\frac{t}{2}\right), F_{T^{n_*}x_*,T^{n_*}x_i}\left(\frac{t}{2}\right)\right) \geqslant \Delta\left(F_{x_*,T^{n_*}x_i}\left(\frac{t}{2}\right), F_{x_*,x_i}\left(\frac{t}{2k}\right)\right)$$
$$\geqslant \Delta\left(F_{x_*,T^{n_*}x_i}\left(\frac{t}{2}\right), F_{x_*,x_i}\left(\frac{t}{2}\right)\right) \to 1 (i\to\infty), \quad \forall t>0, \tag{A7}$$

namely, $F_{x_*,T^{n_*}x_*}(t) = 1, \ \forall t>0$. As a result, $x_* = T^{n_*}x_*$.

If there exists another point $y_* \in E$ such that $y_* = T^{n_*}x_*$, then

$$F_{x_*,y_*}(t) = F_{T^{n_*}x_*,T^{n_*}y_*}(t) \geqslant F_{x_*,y_*}\left(\frac{t}{2}\right), \quad \forall t \geqslant 0.$$

Iteratively, we have

$$F_{x_*,y_*}(t) \geqslant F_{x_*,y_*}\left(\frac{t}{k^n}\right), \quad \forall t \geqslant 0. \tag{A8}$$

Taking the limit as $n \to \infty$ on the right side of the above inequality and considering that $\lim_{n\to\infty}\frac{t}{k^n}=\infty$, we can obtain that $F_{x_*,y_*}(t)=1, \ \forall t \geqslant 0$, implying that $x_* = y_*$. As such, x_* is the unique fixed point of T^{n_*} in E. Since $Tx_* = TT^{n_*}x_* = T^{n_*}Tx_*$, Tx_* is also a fixed point of T^{n_*}. Thus $Tx_* = x_*$, which means that x_* is a fixed point of T. It is evident that x_* is the unique fixed point of T.

(3) Finally we have to prove that for every $x_0 \in E$, the iterative sequence $\{T^n x_0\}$ converges to x_* in $\mathcal{T}$. For every positive integer $n > n_*$ and $n = mn_* + s$ where $0 \leqslant s < n_*$, from (18), we have,

$$F_{x_*,T^n x_0}(t) = F_{T^{n_*}x_*,T^{mn_*+s}x_0}(t) \geqslant F_{x_*,T^{(m-1)n_*+s}x_0}\left(\frac{t}{k}\right) \geqslant \cdots \geqslant F_{x_*,T^s x_0}\left(\frac{t}{k^m}\right), \quad \forall t \geqslant 0. \tag{A9}$$

Let $m \to \infty$ on the rightmost side of (A9). Thus $n \to \infty$ on the leftmost side the inequality, and we obtain

$$\lim_{n\to\infty} F_{x_*,T^n x_0}(t) \geqslant \lim_{m\to\infty} F_{x_*,T^s x_0}\left(\frac{t}{k^m}\right) = 1, \quad \forall t \geqslant 0, \tag{A10}$$

which implies that $T^n x_0 \xrightarrow{\mathcal{T}} x_*$.

This completes the proof of the theorem. □

Theorem 2. *Consider the ordered pair (V, F), where F is a mapping of $V \times V$ into $\mathcal{D}$. For every $x, y \in V$, if the distribution function $F_{x,y}$ is defined by $F_{x,y}(t) = P\{|x - y| < t\}$, $\forall t \in R$, then (V, F) is a PM-space.*

Proof. To achieve the proof of the theorem, we only need to show that the mapping F satisfies conditions (PM-1) to (PM-4).

(1) For every $x, y \in V$, since $|x - y| \geqslant 0$, $F_{x,y}(0) = P\{|x - y| < 0\} = 0$, implying that F satisfies condition (PM-1).
(2) For every $x, y \in V$, if $F_{x,y}(t) = H(t)$ for every $t > 0$, we have $F_{x,y}(t) = P\{|x - y| < t\} = 1$, which implies that for every positive integer m, $P\{|x-y| < \frac{1}{m}\} = 1$. Therefore, $P\{\bigcap_{m=1}^{\infty}(|x-y| < \frac{1}{m})\} = P\{|x-y| = 0\} = 1$, namely, $x = y$. Contrarily, if $x = y$, namely $|x - y| = 0$, then for every $t > 0$ $F_{x,y}(t) = P\{|x - y| < t\} = 1$; or for every $t < 0$, $F_{x,y}(t) = P\{|x - y < t\} = 0$. Thus $F_{x,y}(t) = H(t)$, that is, F satisfies (PM-2).
(3) By the definition of F, we find it evident that F satisfies (PM-3).
(4) For ever $x,y,z \in V$, if $F_{x,y}(t_1) = 1$ and $F_{y,z}(t_2) = 1$, then $F_{x,y}(t_1) = P\{|x - y| < t_1\} = 1$ and $F_{y,z}(t_2) = P\{|y - z| < t_2\} = 1$. Since $|x - z| \leqslant |x - y| + |y - z|$, so

$$F_{x,z}(t_1+t_2) = P\{|x-z| \leqslant t_1+t_2\} \geqslant P\{|x-y|+|y-z| < t_1+t_2\} \geqslant P\{(|x-y| < t_1) \cap (|y-z| < t_2)\} = P\{|x-y| < t_1\} + P\{|y-z| < t_2\} - P\{(|x-y| < t_1) \cup (|y-z| < t_2)\} = 2 - P\{(|x-y| < t_1) \cup (|y-z| < t_2)\} \geqslant 1.$$

Accordingly, $F_{x,z}(t_1 + t_2) = P\{|x - z| < t_1 + t_2\} = 1$, which implies that F satisfies (PM-4).
This completes the proof of the theorem. □

Theorem 3. *Triplet (V, F, Δ) is a Menger space, where $\Delta = \Delta_1$.*

Proof. To achieve the proof, we only need to prove that (V, F, Δ) satisfies Menger's triangle inequality (PM-4)′. For every $x, y, z \in V$ and every $t_1 \geqslant 0$, $t_2 \geqslant 0$, since

$$|x-z| \leqslant |x-y| + |y-z|,$$

we have

$$\{|x-z| < t_1+t_2\} \supset \{|x-y|+|y-z| < t_1+t_2\} \supset \{|x-y| < t_1\} \cap \{|y-z| < t_2\}. \tag{A11}$$

Hence

$$\begin{aligned} F_{x,z}(t_1+t_2) &= P\{|x-z| < t_1+t_2\} \geqslant P\{|x-y|+|y-z| < t_1+t_2\} \geqslant P\{(|x-y| < t_1) \cap (|y-z| < t_2)\} = P\{|x-y| \\ &< t_1\} + P\{y-z| < t_2\} - P\{(|x-y| < t_1) \cup (|y-z| < t_2)\} \geqslant P\{|x-y| < t_1\} + P\{y-z| < t_2\} - 1 \\ &= F_{x,y}(t_1) + F_{y,z}(t_2) - 1 = \max\{F_{x,y}(t_1) + F_{y,z}(t_2) - 1, 0\} = \Delta_1(F_{x,y}(t_1), F_{y,z}(t_2)) \end{aligned}$$

implying that (V, F, Δ), where $\Delta = \Delta_1$, satisfies Menger's triangle inequality (PM-4)′. Therefore, (V, F, Δ) is a Menger space.

This completes the proof of the theorem. □

Theorem 4. *The mapping T is a contraction mapping of the Menger space (V, F, Δ).*

Proof. If $t = 0$, it is evident that T satisfies the contractive condition in Definition 7. In the rest part of the proof, we assume that $t > 0$. Given $f' \in V$, $\forall f'' \in V$ and $\forall t > 0$, we suppose that

$$F_{f'f''}(t) = P\{|f' - f''| < t\} = 1 - \delta, \tag{A12}$$

where $0 < \delta < 1$. Let $V(t) = \{f: f \in V, f - f_* < t\}$, where f_* is the global minimum of $f(X)$. $V(t)$ is measurable and its Lebesgue measure $v[V(t)] > 0$. Letting $S(t) = \{X: f(X) \in V(t)\}$, we have that $v[S(t)] > 0$, due to the almost everywhere continuity of $f(X)$.

For the start point f_0, let $a_0(t) = P\{f_0 \in V(t)\} = g_0(t)$. At the precedent iterations, for every random variable $f_n = T^n f_0$, according to the update equation of QPSO, we can let

$$g_n(t) = 1 - \prod_{i=1}^{M}\left[1 - \int_{S(t)} \theta_{X_{i,n}}(G_{n-1}, P_{i,n-1}, x)\,dx\right],$$

where

$$\theta_{X_{i,n}}(G_{n-1}, P_{i,n}, x) = \prod_{j=1}^{N}\left|\int_{P_{i,n}^j}^{G_n^j} \frac{1}{L_{i,n}^j}\exp\left(-2|x-p|/L_{i,n}^j\right)dp\right|.$$

Thus we have

$$a_n(t) = P\{f_n \in V(t), f_k \notin V(t), k = 1,2,\ldots,n-1\} = g_n(t)\prod_{i=0}^{n-1}[1-g_{n-1}(t)]. \tag{A13}$$

Accordingly,

$$F_{f_n f_*}(t) = P\{|f_n - f_*| < t\} = P\{f_n \in V(t)\} = \sum_{i=0}^{n} a_n(t) = 1 - \prod_{i=1}^{n}[1-g_i(t)]. \tag{A14}$$

Since for every $0 \leqslant n < \infty$, we have $\left|X_{i,n}^j\right| < \infty$, $\left|C - X_{i,n}^j\right| < \infty$ or $\left|p - X_{i,n}^j\right| < \infty$. According to (14) or (15), we have $0 < L_{i,j,n} < \infty$, which implies that $\theta_{i,n}(\hat{y}_n, y_{i,n}, x_{i,n})$ is Lebesgue integrable and $0 < g_i(t) < 1$. We thus immediately have that $\sup_{n>0} F_{f_n f_*}(t) = 1$, according to (A14). Hence, for the given δ and f', there exists a positive integer $n_1(f')$ such that whenever $n \geqslant n_1(f')$,

$$F_{T^n f', f_*}(t) = P\{|T^n f' - f_*| < t\} = P\{T^n f' \in V(t)\} > 1 - \frac{\delta}{2},$$

and there also exists a positive integer $n_2(f')$ such that whenever $n \geqslant n_2(f')$,

$$F_{T^n f'', f_*}(t) = P\{|T^n f'' - f_*| < t\} = P\{T^n f'' \in V(t)\} > 1 - \frac{\delta}{2}.$$

Let $n(f') \geqslant \max\{n_1(f'), n_2(f')\}$. Thus both of the following two inequalities are satisfied

$$F_{T^{n(f')} f', f_*}(t) = P\{|T^{n(f')} f' - f_*| < t\} = P\{T^{n(f')} f' \in V(t)\} = P\{T^{n(f')} f' - f_* < t\} > 1 - \frac{\delta}{2},$$
$$F_{T^{n(f')} f'', f_*}(t) = P\{|T^{n(f')} f'' - f_*| < t\} = P\{T^{n(f')} f'' \in V(t)\} = P\{T^{n(f')} f'' - f_* < t\} > 1 - \frac{\delta}{2}.$$

Since the diameter of $V(t)$ is t, that is, $\sup_{x,y \in V(t)} |x - y| = t$. If $T^{n(f')} f', T^{n(f')} f'' \in V(t)$, $|T^{n(f')} f' - T^{n(f')} f''| < t$. As a result, it is satisfied that

$$\{|T^{n(f')} f' - T^{n(f')} f''| < t\} \supset \{(T^{n(f')} f' \in V(t)) \cap T^{n(f')} f'' \in V(t)\} = \{(T^{n(f')} f' - f_* < t) \cap (T^{n(f')} f'' - f_* < t)\}.$$

Thus we have that

$$\begin{aligned} F_{T^{n(f')} f'', T^{n(f')} f''}(t) &= P\{|T^{n(f')} f' - T^{n(f')} f''| < t\} \geqslant P\{(T^{n(f')} f' - f_* < t) \cap (T^{n(f')} f'' - f_* < t)\} = P\{T^{n(f')} f' - f_* < t\} + P\{T^{n(f')} f'' - f_* \\ &< t\} - P\{(T^{n(f')} f' - f_* < t) \cup (T^{n(f')} f'' - f_* < t)\} > 1 - \frac{\delta}{2} + 1 - \frac{\delta}{2} - P\{(T^{n(f')} f' - f_* < t) \cup (T^{n(f')} f'' - f_* < t)\} \\ &> 2 - \delta - 1 = 1 - \delta, \end{aligned}$$

and accordingly

$$F_{T^{n(f')} f', T^{n(f')} f''}(t) > F_{f', f''}(t). \tag{A15}$$

Since F is monotonically increasing with t, there must exist $k \in (0,1)$ such that

$$F_{T^{n(f_i)} f', T^{n(f_i)} f''}(t) \geqslant F_{f', f''}\left(\frac{t}{k}\right). \tag{A16}$$

It implies that T satisfies the contractive condition in Definition 7.

This completes the proof of the theorem. □

Theorem 5. *f_* is the unique fixed point in V such that for every $f_0 \in V$, the iterative sequence $\{T^n f_0\}$ converges to f_*.*

Proof. For Menger space (V, F, Δ), where $\Delta = \Delta_1$, T is a contraction mapping as shown by Theorem 4. Given $f_0 \in V$, we have $f_n = T^n f_0 \in [f_*, f_0]$ for every $n \geqslant 1$. This implies that $O_T(f_0; 0, \infty) = \{f_n = T^n f_0\}_{n=0}^{\infty} \subset [f_*, f_0]$. Thus for every $t > f_0 - f_*$, $\inf_{f', f'' \in O_T(f_0; 0, \infty)} F_{f', f''}(t) = 1$. Accordingly, we have

$$\sup_{t>0} \inf_{f', f'' \in O_T(f_0; 0, \infty)} F_{f', f''}(t) = 1, \tag{A17}$$

which means that the orbit generated by T at f_0 is probabilistic bounded. By Theorem 1, there exists a unique common fixed point in E for T. Since $f_* = Tf_*$, f_* is the fixed point. Consequently, for every $f_0 \in V$, the iterative sequence $\{T^n f_0\}$ converges to f_* in $\mathcal{T}$.

This completes the proof of the theorem. □

Theorem 6. *The sequence of function values $\{f_n, n \geqslant 0\}$ generated by QPSO converges to f_* in probability.*

Proof. Since $\{f_n, n \geqslant 0\}$ converges to f_* in $\mathcal{T}$, by Definition 5 and the definition given in Theorem 4, for every $\varepsilon > 0$, $\lambda > 0$, there exists $K = K(\varepsilon, \lambda)$ such that whenever $n \geqslant K$,

$$F_{f_n f_*}(\varepsilon) = P\{|f_n - f_*| < \varepsilon\} = P\{f_n \in V_\varepsilon\} > 1 - \lambda. \tag{A18}$$

Due to the arbitrariness of λ, (A18) implies that $f_n \xrightarrow{P} f_*$.

This completes the proof of the theorem. □

Theorem 7. *Let $\bar{c} = \left(\prod_{i=1}^{n} \bar{c}_i\right)^{1/n}$ where $\bar{c}_i = E(c_i)$. If $\{c_n, n > 0\}$ and $\{f_n - f_*, n > 0\}$ are two negatively correlated (or positively correlated or uncorrelated) sequences of random variables, then $c < \bar{c}$ (or $c > \bar{c}$ or $c = \bar{c}$).*

Proof. According to the properties of conditional expectations, we have

$$E[(f_n - f_*)] = E[E[(f_n - f_*)|f_{n-1}]], \tag{A19}$$

for all $n > 0$. If $\{c_n, n > 0\}$ and $\{f_n - f_*, n > 0\}$ are negatively correlated, it follows that

$$Cov(c_n, f_n - f_*) = E[c_n(f_n - f_*)] - E(c_n)E(f_n - f_*) < 0,$$

namely

$$E[c_n(f_n - f_*)] < E(c_n)E(f_n - f_*), \tag{A20}$$

for all $n > 0$. By (A19) and (A20), we therefore have

$$\begin{aligned} \varepsilon_n &= E(f_n - f_*) = E\{E[(f_n - f_*)|f_{n-1}]\} = E[c_n(f_{n-1} - f_*)] < E(c_n)E(f_{n-1} - f_*) = \bar{c}_n E(f_{n-1} - f_*) = \bar{c}_n E\{E[(f_{n-1} - f_*)|f_{n-2}]\} \\ &= \bar{c}_n E[c_{n-1}(f_{n-2} - f_*)] < \bar{c}_n \bar{c}_{n-1} E(f_{n-2} - f_*) = \cdots < \bar{c}_n \bar{c}_{n-1} \bar{c}_{n-2} \ldots \bar{c}_1 E(f_0 - f_*) = \left(\prod_{i=1}^{n} \bar{c}_i\right) E(f_0 - f_*) = \bar{c}^n \varepsilon_0, \end{aligned} \tag{A21}$$

implying that $c = (\varepsilon_n/\varepsilon_0)^{1/n} < \bar{c}$.

If $\{c_n, n > 0\}$ and $\{f_n - f_*, n > 0\}$ are positively correlated,

$$Cov(c_n, f_n - f_*) = E[c_n(f_n - f_*)] - E(c_n)E(f_n - f_*) > 0.$$

Similarly, we have $c > \bar{c}$.

If $\{c_n, n > 0\}$ and $\{f_n - f_*, n > 0\}$ are uncorrelated,

$$Cov(c_n, f_n - f_*) = E[c_n(f_n - f_*)] - E(c_n)E(f_n - f_*) = 0.$$

Replacing each sign of inequality in (A21) by an equal sign, we find that $c = \bar{c}$.

This completes the proof of the theorem. □

References

[1] P.J. Angeline, Evolutionary optimization versus particle swarm optimization: philosophy and performance differences. Evolutionary programming VII, Lecture Notes in Computer Science 1447 (1998) 601–610.

[2] P.J. Angeline, Using selection to improve particle swarm optimization, in: Proceedings of the 1998 IEEE International Conference on Evolutionary Computation, ICEC 1998, Anchorage, AK, USA, 1998, pp. 84–89.

[3] W.P. Baritompa, B.-P. Zhang, R.H. Mlandineo, G.R. Wood, Z.B. Zambinsky, Towards pure adaptive search, Journal of Global Optimization 7 (1995) 93–110.

[4] D. Bratton, J. Kennedy, Defining a standard for particle swarm optimization, in: Proceedings of the 2007 IEEE Swarm Intelligence Symposium, 2007, pp. 120–127.

[5] F. van den Bergh, An Analysis of Particle Swarm Optimizers, Ph.D. dissertation, Univ. Pretoria, Pretoria, South Africa, 2002.

[6] F. van den Bergh, A.P. Engelbrecht, A cooperative approach to particle swarm optimization, IEEE Transactions on Evolutionary Computation 8 (2004) 225–239.

[7] H.-G. Beyer, Toward a theory of evolution strategies: on the benefits of sex-the $(\mu/\mu,\lambda)$ theory, Evolutionary Computation 3 (1995) 81–111.

[8] S.-S. Chang, On some fixed point theorems in probabilistic metric space and its applications, Probability Theory and Related Fields 63 (1983) 463–474.

[9] S.-S. Chang, Y.-J. Cho, S.-M. Kang, Nonlinear Operator Theory in Probabilistic Metric Spaces, Nova Science Publishers, New York, 2001.

[10] M. Clerc, The swarm and the queen: towards a deterministic and adaptive particle swarm optimization, in: Proceedings of the 1999 Congress on Evolutionary Computation, CEC 99, Washington, DC, USA, 1999, pp. 1951–1957.

[11] M. Clerc, J. Kennedy, The particle swarm-explosion, stability and convergence in a multidimensional complex space, IEEE Transactions on Evolutionary Computation 6 (2002) 58–73.

[12] M. Clerc, Particle Swarm Optimization, ISTE Publishing Company, London, UK, 2006.

[13] L.S. Coelho, A novel Gaussian quantum-behaved particle swarm optimizer applied to electromagnetic design, IET Science, Measurement & Technology 1 (2007) 290–294.

[14] L.S. Coelho, P. Alotto, Global optimization of electromagnetic devices using an exponential quantum-behaved particle swarm optimizer, IEEE Transactions on Magenetics 44 (2008) 1074–1077.

[15] L.S. Coelho, A quantum particle swarm optimizer with chaotic mutation operator, Chaos, Solitons & Fractals 37 (2008) 1409–1418.

[16] G. Constantin, I. Istratescu, Elements of Probabilistic Analysis with Applications, Springer, 1989.

[17] R.W. Day, G.P. Quinn, Comparisons of treatments after an analysis of variance in ecology, Ecological Monographs 59 (1989) 433–463.
[18] H.M. Emara, H.A.A. Fattah, Continuous swarm optimization technique with stability analysis, in: Proceedings of the 2004 American Control Conference, Boston, MA, 2004, pp. 2811–2817.
[19] F. Gao, Parameters estimation on-line for Lorenz system by a novel quantum-behaved particle swarm optimization, Chinese Physics B 17 (2008) 1196–1201.
[20] V. Gavi, K.M. Passino, Stability analysis of social foraging swarms, IEEE Transactions on Systems, Man and Cybernetics 34 (2003) 539–557.
[21] R. Hassan, B. Cohanim, O. De Weck, G. Venter, A comparison of particle swarm optimization and the genetic algorithm, in: Proceedings of the 46th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Austin, TX, USA, 2005, pp. 1–13.
[22] J. He, X. Yao, Towards an analytic framework for analysing the computation time of evolutionary algorithms, Artificial Intelligence 145 (2003) 59–97.
[23] J. He, X. Yao, Drift analysis and average time complexity of evolutionary algorithms, Artificial Intelligence 127 (2001) 57–85.
[24] J. He, X. Yao, From an individual to a population: an analysis of the first hitting time of population-based evolutionary algorithms, IEEE Transactions on Evolutionary Computation 6 (2002) 495–551.
[25] I. Istratescu, A fixed point theorem for mappings with a probabilistic contractive iterate, Romanian Journal of pure and applied mathematics 26 (1981) 431–435.
[26] V. Istratescu, I. Sacuiu, Fixed point theorems for contraction mappings on probabilistic metric spaces, Romanian Journal of pure and applied mathematics 18 (1973) 1375–1381.
[27] S. Janson, M. Middendorf, A hierarchical particle swarm optimizer and its adaptive variant, IEEE Transactions on Systems, Man and Cybernetics, Part B: Cybernetics 6 (2005) 1272–1282.
[28] M. Jiang, Y.P. Luo, S.Y. Yang, Stochastic convergence analysis and parameter selection of the standard particle swarm optimization algorithm, Information Processing Letters 102 (2007) 8–16.
[29] K.O. Jones, Comparison of genetic algorithm and particle swarm optimization, in: Proceedings of the 2005 International Conference on Computer System and Technologies, Varna, Bulgaria, 2005, pp. IIIA1–6.
[30] V. Kadirkamanathan, K. Selvarajah, P.J. Fleming, Stability analysis of the particle dynamics in particle swarm optimizer, IEEE Transactions on Evolutionary Computation 10 (2006) 245–255.
[31] J. Kennedy, R.C. Eberhart, Particle swarm optimization, in: Proceedings of the 1995 IEEE International Conference on Neural Networks, Perth, WA, Australia, 1995, pp. 1942–1948.
[32] J. Kennedy, The behavior of particles, in: Proceedings of the 7th Annual Conference on Evolutionary Programming, San Diego, CA, 1998, pp. 581–589.
[33] J. Kennedy, Small worlds and mega-minds: effects of neighborhood topology on particle swarm performance, in: Proceedings of the 1999 Congress on Evolutionary Computation, CEC99, 1999, pp. 1931–1938.
[34] J. Kennedy, Stereotyping: improving particle swarm performance with cluster analysis, in: Proceedings of the 2002 Congress on Computational Intelligence, Honolulu, HI, USA, 2002, pp. 1671–1676.
[35] J. Kennedy, Bare bones particle swarms, in: Proceedings of the 2003 IEEE Swarm Intelligence Symposium, Indianapolis, IN, USA, 2003, pp. 80–87.
[36] J. Kennedy, Probability and dynamics in the particle swarm, in: Proceedings of the 2004 Congress on Evolutionary Computation, June 2004, pp. 340–347.
[37] J. Kennedy, In search of the essential particle swarm, in: Proceedings of the IEEE World Congress on Computational Intelligence, 2006, pp. 1694–1701.
[38] R.A. Krohling, Gaussian swarm: a novel particle swarm optimization algorithm, in: Proceedings of the IEEE Conference on Cybernetics and Intelligent Systems, Singapore, December 2004, pp. 372–376.
[39] R.A. Krohling, L.S.Coelho, PSO-E: particle swarm with exponential distribution, in: Proceedings of the IEEE Congress on Evolutionary Computation, 2006, pp. 1428–1433.
[40] S. Kumar, B.D. Pant, Some common fixed point theorems for mappings satisfying a new contraction condition in Menger spaces, Journal of Mathematical Sciences 5 (2005) 227–234.
[41] K. Leung, Q. Duan, Z. Xu, C.K. Wong, A new model of simulated evolutionary computation-convergence analysis and specification, IEEE Transactions on Evolutionary Computation 5 (2001) 3–16.
[42] J.J. Liang, P.N. Suganthan, Dynamic multiswarm particle swarm optimizer (DMS-PSO), in: Proceedings of the 2005 IEEE Swarm Intelligence Symposium, Pasadena, CA, USA, 2005, pp. 124–129.
[43] J.J. Liang, A.K. Qin, P.N. Suganthan, S. Baskar, Comprehensive learning particle swarm optimizer for global optimization of multimodal functions, IEEE Transactions on Evolutionary Computation 10 (2006) 281–295.
[44] Z.J. Li, X.D. Liu, X.D. Duan, Comparative research on particle swarm optimization and genetic algorithm, Computer and Information Science 3 (2010) 120–127.
[45] S.-Y. Li, R.-G. Wang, W.-W. Hu, J.-Q. Sun, A new QPSO based BP neural network for face detection, in: B.-Y. Cao (Ed.), Fuzzy Information and Engineering, Advances in Soft Computing, 40, Springer, 2007.
[46] R. Mendes, J. Kennedy, J. Neves, The fully informed particle swarm: simpler, maybe better, IEEE Transactions on Evolutionary Computation 8 (2004) 204–210.
[47] K. Menger, Statistical metrics, Proceedings of the National Academy of Science USA 28 (1942) 535–537.
[48] S.M. Mikki, A.A. Kishk, Quantum particle swarm optimization for electromagnetics, IEEE Transactions on Antennas and Propagation 54 (2006) 2764–2775.
[49] A. Mohais, R. Mendes, C. Ward, C. Postoff, Neighborhood re-structuring in particle swarm optimization, in: Proceedings of the 18th Australian Joint Conference on Artificial Intelligence, Sydney, 2005, pp. 776–785.
[50] E.V. Onet, Particle swarm optimization and genetic algorithms, Journal of Computer Science and Control Systems 2 (2009) 43–46.
[51] E. Ozcan, C.K. Mohan, Particle swami optimization: surfing the waves, in: Proceedings of the 1999 IEEE Congress on Evolutionary Computation, CEC'99, Washington, DC, 1999, pp. 1939–1944.
[52] S.C. Pandey, P. Tripathi, Global output convergence of RNN in Menger probabilistic metric space, in: Proceedings of the 2008 International MultiConference of Engineers and Computer Scientists, 2008, pp. 101–104.
[53] B.D. Pant, S. Kumar, Some common fixed point theorems for commuting mappings in Menger spaces, Journal of Natural & Physical Sciences 19 (2005) 29–37.
[54] D. Parrott, X. Li, Locating and tracking multiple dynamic optima by a particle swarm model using speciation, IEEE Transactions on Evolutionary Computation 10 (2006) 440–458.
[55] X. Qi, F. Palmeiri, Theoretical analysis of evolutionary algorithms with infinite population size in continuous space. Part I: Basic properties, IEEE Transactions on Neural Networks 5 (1994) 102–119.
[56] T.J. Richer, T.M. Blackwell, The Levy particle swarm, in: Proceedings of the 2006 Congress on Evolutionary Computation, Vancouver, Canada, 2006, pp. 808–815.
[57] G. Rudolph, Local convergence rates of simple evolutionary algorithms with Cauchy mutations, IEEE Transactions on Evolutionary Computation 1 (1997) 249–258.
[58] R. Saadati, M. Amini M, D-boundedness and D-compactness in finite dimensional probabilistic normed spaces, Proceedings of the Indian Academy of Science (Math. Sci.) 115 (2005) 483–492.
[59] B. Schweizer, A. Sklar, Statistical metric spaces, Pacific Journal of Mathematics 10 (1960) 313–334.
[60] B. Schweizer, A. Sklar, E. Thorp, The metrization of statistical metric spaces, Pacific Journal of Mathematics 10 (1960) 673–675.
[61] B. Schweizer, A. Sklar, Statistical metric spaces arising from sets of random variables in n-Euclidean spaces, Theory Probability and its Applications 7 (1962) 447–456.

[62] B. Schweizer, A. Sklar, Triangle inequalities in a class of statistical metric spaces, Journal of the London Mathematical Society 38 (1963) 401–406.
[63] B. Schweizer, A. Sklar, Probabilistic Metric Spaces, Dover Publications, 1983.
[64] B. Secrest, G. Lamont, Visualizing particle swarm optimization-Gaussian particle swarm optimization, in: Proceedings of the 2003IEEE Swarm Intelligence Symposium, SIS'03, 2003, pp. 198–204.
[65] V.M. Sehgal, Some Fixed Point Theorems in Functional Analysis and Probability, Ph.D. dissertation, Wayne State University, 1966.
[66] C. Sempi, Probabilistic metric spaces, in: K P Hart et al. (Eds.), Encyclopaedia of General Topology, Kluwer, 2003, pp. 288–292.
[67] A.N. Šerstnev, On a probabilistic generalization of metric spaces, Kazanskogo Gosudarstvennogo Universiteta Uchenye Zapiski 124 (1964) 3–11.
[68] Y. Shi, R.C. Eberhart, A modified particle swarm optimizer, in: Proceedings of the 1998 IEEE International Conference on Evolutionary Computation, ICEC'98, Anchorage, AK, USA, 1998, pp. 69–73.
[69] Y. Shi, R.C. Eberhart, Parameter selection in particle swarm optimization, in: Proceedings of the 7th Conference on Evolutionary Programming, EP'98, 1998, pp. 591–600.
[70] Y. Shi, R.C. Eberhart, Empirical study of particle swarm optimization, in: Proceedings of the 1999 Congress on Evolutionary Computation, CEC '99, 1999, pp. 1945–1950.
[71] S.L. Singh, B.D. Pant, Common fixed point theorems for commuting mappings in probabilistic metric spaces, Honam Mathematical Journal 5 (1983) 139–150.
[72] S.L. Singh, B.D. Pant, Common fixed point theorems in probabilistic metric spaces and extension to uniform spaces, Honam Mathematical Journal 6 (1984) 1–12.
[73] S.L. Singh, S.N. Mishra, B.D. Pant, General fixed point theorems in probabilistic metric and uniform space, Indian Journal of Mathematics 29 (1987) 9–21.
[74] F.J. Solis, R. J-B. Wets, Minimization by random search techniques, Mathematics of Operations Research 6 (1981) 19–30.
[75] D.R. Stark, J.C. Spall, Rate of convergence in evolutionary computation, in: Proceedings of the 2003 American Control Conferences, 2003, pp. 1932–1937.
[76] P.N. Suganthan, Particle swarm optimizer with neighborhood operator, in: Proceedings of the 1999 Congress on Evolutionary Computation, CEC'99, Wasington, DC, USA, pp. 1958–1961.
[77] P.N. Suganthan, N. Hansen, J.J. Liang, K. Deb, Y.-P. Chen, A. Auger, S. Tiwari, Problem Definitions and Evaluation Criteria for the CEC 2005 Special Session on Real-parameter Optimization, Technical Report, Nanyang Technological University, Singapore, May 2005. KanGAL Report #2005005, IIT Kanpur, India.
[78] J. Sun, B. Feng, W.B. Xu, Particle swarm optimization with particles having quantum behavior, in: Proceedings of the 2004 Congress on Evolutionary Computation, CEC'04, Portland, USA, 2004, pp.326–331.
[79] J. Sun, W.B. Xu, B. Feng, A global search strategy of quantum-behaved particle swarm optimization, in: Proceedings of the 2004 IEEE conference on Cybernetics and Intelligent Systems, Singapore, 2004, pp.111–116.
[80] J. Sun, W.B. Xu, B. Feng, AddTaptive parameter control for quantum-behaved particle swarm optimization on individual level, in: Proceedings of the 2005 IEEE International Conference on Systems, Man and Cybernetics, Hawaii, USA, 2005, pp. 3049–3054.T.
[81] J. Sun, W. Fang, X. Wu, V. Palade, W. Xu, Quantum-behaved particle swarm optimization: analysis of the individual particle's behavior and parameter selection, Evolutionary Computation (2011), doi:10.1162/EVCO_a_00049.
[82] R.M. Tardiff, Topologies for probabilistic metric spaces, Pacific Journal of Mathematics 65 (1976) 233–251.
[83] E. Thorp, Generalized topologies for statistical metric spaces, Fundamenta Mathematicae 51 (1962) 9–21.
[84] I.C. Trelea, The particle swarm optimization algorithm: convergence analysis and parameter selection, Information Processing Letters 85 (2003) 317–325.
[85] M. Vose, Logarithmic convergence of random heuristic search, Evolutionary Computation 4 (1997) 395–404.
[86] J. Wang, Y. Zhou, Quantum-behaved particle swarm optimization with generalized local search operator for global optimization, in: Proceedings of the 2007 International Conference on Intelligent Computing, 2007, pp. 344–352.
[87] Y. Yu, Z.-H. Zhou, A new approach to estimating the expected first hitting time of evolutionary algorithms, Artificial Intelligence 172 (2008) 1809–1832.
[88] Z.B. Zabinsky, R.L. Smith, Pure adaptive search in global optimization, Mathematical Programming 53 (1992) 323–338.
[89] Y. Zhao, Z. Fang, K. Wang, H. Pang, Multilevel minimum cross entropy threshold selection based on quantum particle swarm optimization, in: Proceedings of the Eighth ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing, 2007, pp. 65–69.