Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2014, Article ID 907386, 19 pages
http://dx.doi.org/10.1155/2014/907386

Research Article

Fusion Global-Local-Topology Particle Swarm Optimization for Global Optimization Problems

Zahra Beheshti, Siti Mariyam Shamsuddin, and Sarina Sulaiman

UTM Big Data Centre, Universiti Teknologi Malaysia, Skudai, 81310 Johor, Malaysia

Correspondence should be addressed to Siti Mariyam Shamsuddin; mariyam@utm.my

Received 22 February 2014; Revised 18 May 2014; Accepted 24 May 2014; Published 25 June 2014

Academic Editor: Dan Simon

In recent years, particle swarm optimization (PSO) has been extensively applied in various optimization problems because of its structural and implementation simplicity. However, the PSO can sometimes find local optima or exhibit slow convergence speed when solving complex multimodal problems. To address these issues, an improved PSO scheme called fusion global-local-topology particle swarm optimization (FGLT-PSO) is proposed in this study. The algorithm employs both global and local topologies in PSO to jump out of the local optima. FGLT-PSO is evaluated using twenty (20) unimodal and multimodal nonlinear benchmark functions and its performance is compared with several well-known PSO algorithms. The experimental results showed that the proposed method improves the performance of PSO algorithm in terms of solution accuracy and convergence speed.

1. Introduction

PSO is a population-based metaheuristic algorithm introduced by Kennedy and Eberhart [1] in 1995. The algorithm imitates the social behavior of bird flocking or fish schooling to find the global best solution. Due to the simple concept, having a few parameters and being easy to implement, PSO has received much more attention to solve real-world optimization problems [2–6] in recent years. Nevertheless, PSO may easily get trapped in local optima when solving complex multimodal problems [7]. Hence, a number of variant PSO algorithms have been proposed in the literature to avoid the local optima and to find the best solution promptly.

The algorithm applies two different topologies to find a good solution: global and local topologies. In global topology, the position of each particle is affected by the best-fitness particles of the entire population in the search space while each particle is influenced by the best-fitness particles of its neighborhood in the local topology. Kennedy and Mendes proposed local (ring) topological structure PSO (LPSO) [8] and the Von Neumann topological structure PSO (VPSO) [9]. Mendes et al. [10] introduced the fully informed particle swarm (FIPS) algorithm and Ratnaweera et al. [11] suggested self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients (HPSO-TVAC). Other researchers presented the several variants of PSO algorithms such as dynamic multiswarm PSO (DMS-PSO) [12], comprehensive learning PSO (CLPSO) [13], median-oriented particle swarm optimization (MPSO) [14], centripetal accelerated particle swarm optimization (CAPSO) [15], quadratic interpolation PSO (QIPSO) [16], quantum-behaved particle swarm optimization (QPSO) [17], and adaptive particle swarm optimization (APSO) [18].

Although the aforementioned algorithms have obtained satisfactory results, there are still some disadvantages in their utilization. For example, LPSO presents a slow convergence rate in unimodal functions [14, 15] or CLPSO is not good for solving unimodal problems [13]. Moreover, some of the algorithms have a better performance than PSO but their structures are not as simple as PSO.

To overcome the disadvantages, this study introduces fusion global-local-topology particle swarm optimization (FGLT-PSO). The proposed algorithm performs a global search over the entire search space with a fast convergence speed using hybridizing two local and global topologies in PSO to jump out from local optima.

The remainder of this paper is organized as follows. In Section 2, a brief review of PSO is provided followed by some well-known PSO algorithms. The proposed algorithm is described in Section 3 in detail. In Section 4, FGLT-PSO is used to solve several benchmark functions and its performance is compared with the other PSO algorithms in the literature. Finally, conclusions and the future research directions are presented in Section 5.

2. Particle Swarm Optimization (PSO)

2.1. PSO Framework. The PSO algorithm is a population-based metaheuristic algorithm that applies two approaches of global exploration and local exploitation to find the optimum solution. The exploration is the ability of expanding search space, where the exploitation is the ability of finding the optima around a good solution. The algorithm is initialized by creating a swarm, that is, population of particles, with random positions. Every particle is shown as a vector $(\vec{X}_i, \vec{V}_i, \vec{P}_i)$ in a *d-dimensional* search space where $\vec{X}_i$ and $\vec{V}_i$ are the position and velocity, respectively, and $\vec{P}_i$ is the personal best position (P_{best}) found by the ith particle:

$$\begin{aligned} \vec{X}_i &= \left(x_i^1, x_i^2, \dots, x_i^d\right) \quad \text{for } i = 1, 2, \dots, N, \\ \vec{V}_i &= \left(v_i^1, v_i^2, \dots, v_i^d\right) \quad \text{for } i = 1, 2, \dots, N, \\ \vec{P}_i &= \left(p_i^1, p_i^2, \dots, p_i^d\right) \quad \text{for } i = 1, 2, \dots, N. \end{aligned} \tag{1}$$

In addition, the best position obtained by the entire population ($\vec{P}g$) is computed to update the particle velocity:

$$\vec{P}g = \left(p_g^1, p_g^2, \dots, p_g^d\right). \tag{2}$$

Based on $\vec{P}_i$ and $\vec{P}g$, the next velocity and position of the ith particle are computed using (3) and (4) as follows:

$$\begin{aligned} v_i^d(t+1) = {} & w \times v_i^d(t) + C_1 \times \text{rand}_1 \times \left(p_i^d(t) - x_i^d(t)\right) + C_2 \\ & \times \text{rand}_2 \times \left(p_g^d(t) - x_i^d(t)\right), \end{aligned} \tag{3}$$

$$x_i^d(t+1) = x_i^d(t) + v_i^d(t+1), \tag{4}$$

where $v_i^d(t+1)$ and $v_i^d(t)$ are the next and current velocity of the ith particle, respectively. w is inertia weight, C_1 and C_2 are acceleration coefficients, and rand_1 and rand_2 are random numbers in the interval $[0, 1]$. N is the number of particles; $x_i^d(t+1)$ and $x_i^d(t)$ are the next and current position of the ith particle.

Also, $|v_i^d(t+1)| < v_{\max}$ and $v_{\max}$ is set to a constant bounded based on the search space bound. A larger value of w encourages global exploration (searching new areas), while a smaller value provides a local exploitation.

In (3), the second and the third terms are called cognition and social term, respectively. The two models applied to choose $\vec{P}g$ are known as $g$best (for global topology) and $l$best (for local topology) models. In this paper, the $g$best model and $l$best model are called PSO and LPSO, respectively.

2.2. Improved PSO Algorithms. Since Kennedy and Eberhart introduced PSO algorithm, the algorithm and its improved schemes have been extensively applied in many problems [20–25]. Many researchers have proposed the variants of modified PSO through swarm topology [8, 9], parameter selection [19, 26], combining PSO with other evolutionary computation (EC) techniques [27, 28], integration of its self-adaptation [29], and so on.

LPSO [8] and VPSO [9] were proposed based on a local topology to avoid premature convergence rate in solving multimodal problems. FIPS algorithm [10] is another PSO algorithm which uses the information of the entire neighborhood to guide the particles for finding the best solution. Dynamic multiswarm PSO (DMS-PSO) [12] was suggested by Liang and Suganthan to dynamically enhance the topological structure. Ratnaweera et al. [11] proposed HPSO-TVAC algorithm based on linearly time-varying acceleration coefficients where a larger C_1 and a smaller C_2 are set at the beginning and gradually reversed throughout the search. Liang et al. [13] presented comprehensive learning particle swarm optimization (CLPSO) which focused on avoiding the local optima by encouraging each particle to learn its behavior from other particles on different dimensions.

In another research, a selection operator for PSO was first introduced by Angeline [30]. It is similar to what was used in a genetic algorithm (GA). Other researchers used a part of crossover [31] and mutation [29] operations from GA into PSO. Pant et al. proposed a quadratic crossover operator to PSO algorithm called quadratic interpolation PSO (QIPSO) [16]. An adaptive fuzzy particle swarm optimization (AFPSO) [19] was proposed to utilize fuzzy inferences for adjusting acceleration coefficients. Meanwhile, the quadratic crossover operator [16] was used in the proposed AFPSO algorithm (AFPSO-QI) [19] to have better performance in solving multimodal problems. Zhan et al. presented an adaptive particle swarm optimization (APSO) [18] using a real-time evolutionary state estimation procedure and an elitist learning strategy. A variant of PSO algorithm based on orthogonal learning strategy (OLPSO) [32] was introduced to guide particles for discovering useful information from their personal best positions and from their neighborhood's best position in order to fly in better directions. Gao et al. [33] used PSO with chaotic opposition-based population initialization and stochastic search technique to solve complex multimodal problems. The algorithm called CSPSO finds new solutions in the neighborhoods of the previous best positions in order to escape from local optima in multimodal functions. Beheshti et al. proposed median-oriented particle swarm optimization (MPSO) [14] and centripetal accelerated particle swarm optimization (CAPSO) [15] based on Newton's laws of motion to accelerate the learning and convergence of optimization problems.

3. FGLT-PSO: The Proposed Method

3.1. FGLT-PSO Algorithm. FGLT-PSO tends to overcome the disadvantages of PSO by avoiding local optima and accelerating convergence speed. According to [14, 15], PSO has shown a better performance than LPSO in unimodal problems

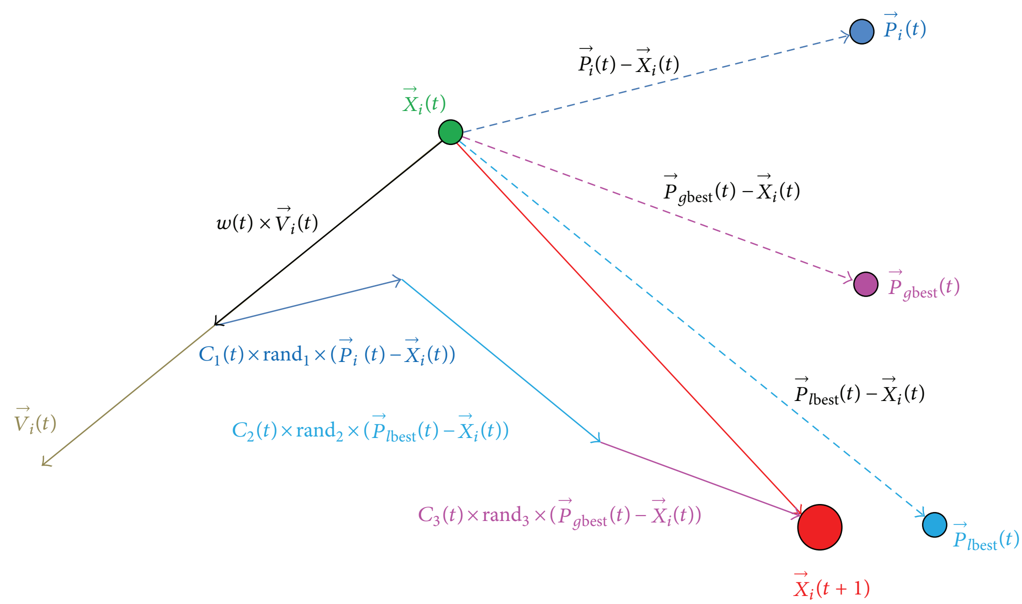

Figure 1: Graphical representation of the particle movement using FGLT-PSO algorithm.

and LPSO illustrates good results in multimodal problems. Hence, both local and global topologies are hybridized in FGLT-PSO to increase the convergence rate and to avoid trapping into local optima.

In FGLT-PSO algorithm, each particle uses the best position found by its neighbors ($\vec{P}_{lbest}$) to update the particles' velocities:

$$\vec{P}_{lbest} = \left(p^1_{lbest}, p^2_{lbest}, \ldots, p^d_{lbest}\right), \tag{5}$$

$$\begin{aligned} v^d_i(t+1) = {} & w(t) \times v^d_i(t) + C_1(t) \times \text{rand}_1 \\ & \times \left(p^d_i(t) - x^d_i(t)\right) + C_2(t) \times \text{rand}_2 \\ & \times \left(p^d_{lbest}(t) - x^d_i(t)\right). \end{aligned} \tag{6}$$

The next position of each particle is computed based on the current position, $x^d_i(t)$, the next velocity, $v^d_i(t+1)$, and the best position found by the swarm, $\vec{P}_{gbest}$, as follows:

$$\begin{aligned} x^d_i(t+1) = {} & x^d_i(t) + v^d_i(t+1) + C_3(t) \times \text{rand}_3 \\ & \times \left(p^d_{gbest}(t) - x^d_i(t)\right), \end{aligned} \tag{7}$$

$$\vec{P}_{gbest} = \left(p^1_{gbest}, p^2_{gbest}, \ldots, p^d_{gbest}\right). \tag{8}$$

In (6), w is computed as

$$w(t) = w_{\max} - \frac{t \times (w_{\max} - w_{\min})}{T}. \tag{9}$$

Also, $C_1(t)$, $C_2(t)$, and $C_3(t)$ are acceleration coefficients and modified according to (10):

$$C_j(t) = C_{j\min} + \frac{t \times \left(C_{j\max} - C_{j\min}\right)}{T}, \quad j = 1, 2, 3, \tag{10}$$

where t and T are the current iteration and the number of maximum iterations, respectively.

The second term in (6) is called the cognition term, and the third terms in (6) and (7) are named the social terms. In (7), $|x^d_i(t+1)| < x_{\max}$ and $x_{\max}$ is set to a constant based on the search space bound.

3.2. Analysis of FGLT-PSO. A metaheuristic algorithm explores new spaces to avoid trapping in a local optimum in the initial steps. Due to the poor exploration in the standard PSO (PSO), it can sometimes find local optima in multimodal problems. Sometimes, if a particle falls into a local optimum, it will not be able to get out of it. That is, if $\vec{P}_{gbest}$ obtained through the population lies in a local optimum while the current position and the personal best position of particle i are in the same local optimum, the second and third terms of (3) tend to zero and w decreases linearly to near zero. Consequently, the next velocity of particle i tends to zero, and its next position in (4) does not change; thus, the particle remains in the local optimum. Hence, the main aim in FGLT-PSO is to overcome the poor exploration and to increase the convergence rate by combining the local and global searches as shown in Figure 1. The particles move in the search space based on the best solutions found by their neighbors ($\vec{P}_{lbest}$) and the swarm ($\vec{P}_{gbest}$). At the beginning, the particles search new spaces. By lapse of iterations, the exploration should fade out and the exploitation should fade in. It means the particles accelerate to the good solution and make search around it to find the best solution.

4. Experimental Results

In this section, the FGLT-PSO algorithm is compared with some well-known PSO algorithms. The algorithms are tested using various unimodal and multimodal functions in different dimensions. Several benchmark functions [34, 35] are selected to evaluate the performance of proposed method.

Table 1: Dimensions, ranges, and global optimum values of test functions used in the experiments.

Test function	Dimension (n)	$[\text{Range}]^n$	X_{opt}	F_{opt}
$F_1(x)$	10/30/50	$[-100, 100]^n$	0	0
$F_2(x)$	10/30/50	$[-10, 10]^n$	0	0
$F_3(x)$	10/30/50	$[-100, 100]^n$	0	0
$F_4(x)$	10/30/50	$[-1.28, 1.28]^n$	0	0
$F_5(x)$	10/30/50	$[-10, 10]^n$	1	0
$F_6(x)$	10/30/50	$[-500, 500]^n$	$[420.96]^n$	$-418.9829 \times n$
$F_7(x)$	10/30/50	$[-32, 32]^n$	0	0
$F_8(x)$	10/30/50	$[-50, 50]^n$	0	0
$F_9(x)$	10/30/50	$[-5.12, 5.12]^n$	0	0
$F_{10}(x)$	10/30/50	$[-10, 10]^n$	0	0
$F_{11}(x)$	10/30/50	$[-5, 5]^n$	0	0
$F_{12}(x)$	10/30/50	$[-5.12, 5.12]^n$	0	0
$F_{13}(x)$	10/30/50	$[-0.5, 0.5]^n$	0	0
$F_{14}(x)$	10/30/50	$[-100, 100]^n$	0	0
$F_{15}(x)$	10/30/50	$[-100, 100]^n$	0	-450
$F_{16}(x)$	10/30/50	$[-600, 600]^n$	0	-180
$F_{17}(x)$	10/30/50	$[-100, 100]^n$	1	390
$F_{18}(x)$	10/30/50	$[-5.2, 5.2]^n$	0	-330
$F_{19}(x)$	10/30/50	$[-32, 32]^n$	0	-140
$F_{20}(x)$	10/30/50	$[-5.2, 5.2]^n$	0	-330

4.1. Benchmark Functions. Twenty (20) minimization functions are applied in the experimental study including unimodal, multimodal, rotated, shifted, and shifted-rotated functions as detailed in Table 1. In the table, *Range* and n are the feasible bound and the dimension of each function, respectively. F_{opt} is the optimum value of function. Among the benchmarks, functions (1)–(5) are unimodal functions and functions (6)–(9) are in the class of multimodal functions. Functions (10)–(14) are rotated and functions (15)–(18) are shifted unimodal and multimodal functions. Two functions (19) and (20) are shifted-rotated multimodal functions.

In unimodal functions, the convergence rate of search algorithm is more interesting than the final results because other methods have been designed to optimize these kinds of functions. In multimodal functions, finding an optimal (or a good near-global optimal) solution is important. These functions are more difficult to optimize because the number of local optima exponentially increases as the dimension increases. Therefore, the search algorithms should not become trapped in a local optimum and should be able to obtain good solutions.

The rotation of function increases the function complexity. It does not affect the shape of function. The variable $\vec{Y}$ is computed using an orthogonal matrix M [36] and applied to obtain the fitness value of rotated function as follows:

$$\vec{Y} = M \times \vec{X}. \tag{11}$$

In shifted functions, the global optimum $\vec{X}^* = (x_1^*, x_2^*, \ldots, x_n^*)$ is shifted to the new position $\vec{O} = (o_1, o_2, \ldots, o_n)$. All the test functions are shown as follows.

(1) Sphere Model (unimodal function). Consider

$$F_1(x) = \sum_{i=1}^{n} x_i^2. \tag{12}$$

(2) Shifted's Schwefel's Problem (unimodal function). Consider

$$F_2(x) = \sum_{i=1}^{n} |x_i| + \prod_{i=1}^{n} |x_i|. \tag{13}$$

(3) Schwefel's Problem 1.2 (unimodal function). Consider

$$F_3(x) = \sum_{i=1}^{n} \left(\sum_{j=1}^{i} x_j \right)^2. \tag{14}$$

(4) Quartic Function, That Is, Noise (unimodal function). Consider

$$F_4(x) = \sum_{i=1}^{n} i x_i^4 + \text{random}[0, 1). \tag{15}$$

(5) Rosenbrock's Function (multimodal function). Consider

$$F_5(x) = \sum_{i=1}^{n-1} \left[100 \left(x_i^2 - x_{i+1} \right)^2 + \left(x_i - 1 \right)^2 \right]. \tag{16}$$

F_5 is unimodal in a 2-dimension or 3-dimension search space but can be treated as a multimodal function in high-dimensional cases.

(6) Generalized Schwefel's Problem 2.26 (multimodal function). Consider

$$F_6(x) = \sum_{i=1}^{n} -x_i \sin\left(\sqrt{|x_i|} \right). \tag{17}$$

(7) Ackley's Function (multimodal function). Consider

$$F_7(x) = -20 \exp\left(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^{n} x_i^2} \right) - \exp\left(\frac{1}{n} \sum_{i=1}^{n} \cos 2\pi x_i \right) + 20 + e. \tag{18}$$

(8) Generalized Penalized Function (multimodal function). Consider

$$F_8(x) = \frac{\pi}{n} \left\{ 10 \sin(\pi y_1) + \sum_{i=1}^{n-1} (y_i - 1)^2 \times \left[1 + 10 \sin^2(\pi y_{i+1}) \right] + (y_n - 1)^2 \right\} + \sum_{i=1}^{n} u(x_i, 10, 100, 4)$$

Table 2: Minimization results for the unimodal and multimodal functions (maximum iteration = 5000 and $n = 10$).

	Function	FGLT-PSO	PSO	LPSO	QIPSO
F_1	Avg. best solution	**0.000e + 000**	1.405e − 135	1.452e − 063	1.526e − 138
	SD	**0.000e + 000**	6.789e − 135	5.681e − 063	5.723e − 138
	Median best solution	0.000e + 000	1.243e − 139	7.381e − 065	3.961e − 142
	Avg. iteration for finding the best solution	2646	5000	5000	5000
F_2	Avg. best solution	**2.741e − 273**	2.992e − 077	4.130e − 038	6.663e − 079
	SD	**0.000e + 000**	1.490e − 076	7.564e − 038	1.234e − 078
	Median best solution	6.578e − 277	3.849e − 079	9.945e − 039	1.019e − 079
	Avg. iteration for finding the best solution	4952	5000	4999	5000
F_3	Avg. best solution	**1.637e − 116**	3.215e − 044	5.110e − 014	3.919e − 044
	SD	**8.969e − 116**	1.565e − 043	1.126e − 013	2.117e − 043
	Median best solution	2.369e − 127	1.857e − 048	5.320e − 015	3.570e − 048
	Avg. iteration for finding the best solution	4630	4999	4999	4999
F_4	Avg. best solution	**3.209e − 004**	4.738e − 004	1.269e − 003	6.514e − 004
	SD	3.006e − 004	2.057e − 004	4.861e − 004	3.241e − 004
	Median best solution	2.114e − 004	4.595e − 004	1.197e − 003	5.797e − 004
	Avg. iteration for finding the best solution	3715	4404	4414	4459
F_5	Avg. best solution	**2.973e − 001**	1.820e + 000	1.661e + 000	2.040e + 000
	SD	**1.015e + 000**	1.336e + 000	1.281e + 000	1.622e + 000
	Median best solution	4.614e − 004	2.014e + 000	1.727e + 000	1.862e + 000
	Avg. iteration for finding the best solution	4537	4811	4988	4837
F_6	Avg. best solution	**−3.910e + 003**	−3.452e + 003	−3.892e + 003	−3.424e + 003
	SD	**1.184e + 002**	2.307e + 002	1.843e + 002	2.775e + 002
	Median best solution	−3.953e + 003	−3.475e + 003	−3.834e + 003	−3.475e + 003
	Avg. iteration for finding the best solution	2974	2726	3918	2635
F_7	Avg. best solution	**4.441e − 015**	4.322e − 015	**4.441e − 015**	**4.441e − 015**
	SD	**0.000e + 000**	6.486e − 016	**0.000e + 000**	**0.000e + 000**
	Median best solution	4.441e − 015	4.441e − 015	4.441e − 015	4.441e − 015
	Avg. iteration for finding the best solution	376	3147	3772	3109
F_8	Avg. best solution	**4.712e − 032**	**4.712e − 032**	**4.712e − 032**	**4.712e − 032**
	SD	**1.670e − 047**	**1.670e − 047**	**1.670e − 047**	**1.670e − 047**
	Median best solution	4.712e − 032	4.712e − 032	4.712e − 032	4.712e − 032
	Avg. iteration for finding the best solution	420	3166	3895	2687
F_9	Avg. best solution	9.495e − 002	9.667e − 001	1.574e + 000	**0.000e + 000**
	SD	2.720e − 001	4.560e + 000	1.065e + 000	**0.000e + 000**
	Median best solution	9.236e − 008	0.000e + 000	2.000e + 000	0.000e + 000
	Avg. iteration for finding the best solution	4856	3357	4366	3401
	Avg. rank	**1.1**	2.9	3.4	2.6
	Final rank	**1**	3	4	2
	Algorithms	**FGLT-PSO**	PSO	LPSO	QIPSO

$$y_i = 1 + \frac{x_i + 1}{4},$$

$$u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m & x_i > a \\ 0 & -a \le x_i \le a \\ k(-x_i - a)^m & x_i < -a. \end{cases} \tag{19}$$

(9) Noncontinuous Rastrigin's Function (multimodal function). Consider

$$F_9(x) = \sum_{i=1}^{n} \left[y_i^2 - 10\cos(2\pi y_i) + 10 \right],$$
$$\text{where} \quad y_i = \begin{cases} x_i & |x_i| \le 0.5 \\ \dfrac{\text{round}(2x_i)}{2} & |x_i| \ge 0.5. \end{cases} \tag{20}$$

Table 3: Minimization results for the rotated and shifted unimodal and multimodal functions (maximum iteration = 5000 and $n = 10$).

	Function	FGLT-PSO	PSO	LPSO	QIPSO
F_{10}	Avg. best solution	**7.056e − 208**	4.585e − 059	2.358e − 021	1.127e − 058
	SD	**0.000e + 000**	8.736e − 059	7.425e − 021	5.346e − 058
	Median best solution	1.930e − 209	7.785e − 060	3.417e − 022	1.023e − 060
	Avg. iteration for finding the best solution	4741	4999	4997	4999
F_{11}	Avg. best solution	−6.602e + 001	**−6.645e + 001**	−6.616e + 001	−6.613e + 001
	SD	1.736e + 000	**1.175e + 000**	9.989e − 001	1.720e + 000
	Median best solution	−6.606e + 001	−6.614e + 001	−6.607e + 001	−6.588e + 001
	Avg. iteration for finding the best solution	3408	2930	3099	3185
F_{12}	Avg. best solution	**1.604e + 001**	2.352e + 001	2.319e + 001	2.278e + 001
	SD	**2.974e + 000**	3.762e + 000	4.012e + 000	4.387e + 000
	Median best solution	1.589e + 001	2.358e + 001	2.351e + 001	2.240e + 001
	Avg. iteration for finding the best solution	2394	3656	3786	3660
F_{13}	Avg. best solution	**0.000e + 000**	1.351e + 000	7.438e − 008	7.957e − 001
	SD	**0.000e + 000**	3.079e + 000	4.072e − 007	2.428e + 000
	Median best solution	0.000e + 000	0.000e + 000	0.000e + 000	0.000e + 000
	Avg. iteration for finding the best solution	448	3192	4877	3446
F_{14}	Avg. best solution	1.165e − 001	1.032e − 001	**9.987e − 002**	1.132e − 001
	SD	3.790e − 002	1.826e − 002	**0.000e + 000**	3.457e − 002
	Median best solution	9.987e − 002	9.987e − 002	9.987e − 002	9.987e − 002
	Avg. iteration for finding the best solution	1998	2916	3268	2795
F_{15}	Avg. best solution	**−4.500e + 002**	−4.396e + 002	−4.480e + 002	−4.406e + 002
	SD	**8.039e − 014**	7.334e + 000	2.470e + 000	7.991e + 000
	Median best solution	−4.500e + 002	−4.450e + 002	−4.500e + 002	−4.450e + 002
	Avg. iteration for finding the best solution	1324	757	3561	1213
F_{16}	Avg. best solution	**−1.798e + 002**	−1.745e + 002	−1.793e + 002	−1.766e + 002
	SD	**3.789e − 001**	8.186e + 000	5.222e − 001	3.309e + 000
	Median best solution	−1.800e + 002	−1.780e + 002	−1.789e + 002	−1.780e + 002
	Avg. iteration for finding the best solution	2922	2592	3398	2711
F_{17}	Avg. best solution	**3.915e + 002**	9.048e + 006	6.752e + 004	6.596e + 006
	SD	**6.315e + 000**	1.678e + 007	3.676e + 005	1.638e + 007
	Median best solution	3.900e + 002	2.014e + 006	3.933e + 002	4.867e + 002
	Avg. iteration for finding the best solution	4831	3983	4884	4090
F_{18}	Avg. best solution	**−3.287e + 002**	−3.215e + 002	−3.247e + 002	−3.233e + 002
	SD	**1.113e + 000**	5.979e + 000	3.144e + 000	4.822e + 000
	Median best solution	−3.290e + 002	−3.225e + 002	−3.250e + 002	−3.225e + 002
	Avg. iteration for finding the best solution	4809	3193	4509	3331
F_{19}	Avg. best solution	−119.744	−119.745	**−119.804**	−119.746
	SD	5.750e − 002	7.142e − 002	**5.558e − 002**	6.911e − 002
	Median best solution	−119.731	−119.748	−119.804	−119.73
	Avg. iteration for finding the best solution	1849	2240	2072	2414
F_{20}	Avg. best solution	**−3.196e + 002**	−3.091e + 002	−3.137e + 002	−3.078e + 002
	SD	**4.236e + 000**	7.174e + 000	5.338e + 000	7.995e + 000
	Median best solution	−3.196e + 002	−3.102e + 002	−3.130e + 002	−3.079e + 002
	Avg. iteration for finding the best solution	3252	3315	4721	3479
	Avg. rank	**1.8**	3.2	2.1	2.9
	Final rank	**1**	4	2	3
	Algorithms	**FGLT-PSO**	PSO	LPSO	QIPSO

TABLE 4: Comparison of FGLT-PSO with PSO, LPSO, and QIPSO for the unimodal and multimodal functions using Wilcoxon's rank sum test ($n = 10$).

Function	Wilcoxon's rank sum test	PSO	LPSO	QIPSO
F_1	P value	1.2118e − 012	1.2118e − 012	1.2118e − 012
	h-value	**1**	**1**	**1**
	z-value	−7.10402	−7.10402	−7.10402
F_2	P value	3.0199e − 011	3.0199e − 011	3.0199e − 011
	h-value	**1**	**1**	**1**
	z-value	−6.6456	−6.6456	−6.6456
F_3	P value	3.0199e − 011	3.0199e − 011	3.0199e − 011
	h-value	**1**	**1**	**1**
	z-value	−6.6456	−6.6456	−6.6456
F_4	P value	1.0576e − 003	6.1210e − 010	3.5923e − 005
	h-value	**1**	**1**	**1**
	z-value	−3.27475	−6.18728	−4.13225
F_5	P value	1.5581e − 008	1.8731e − 007	1.0095e − 008
	h-value	**1**	**1**	**1**
	z-value	−5.65504	−5.21151	−5.72912
F_6	P value	6.3474e − 011	1.2724e − 001	9.3829e − 010
	h-value	**1**	0	**1**
	z-value	−6.53532	−1.52509	−6.11957
F_7	P value	3.3371e − 001	—	—
	h-value	0	0	0
	z-value	0.966667	—	—
F_8	P value	—	—	—
	h-value	0	0	0
	z-value	—	—	—
F_9	P value	1.9097e − 005	1.6703e − 006	6.2470e − 010
	h-value	−1	**1**	−1
	z-value	4.27519	−4.7897	6.18407
	1 (better)	**6**	**6**	**6**
	0 (same)	2	3	2
	−1 (worse)	1	0	1

(10) Rotated Schwefel's Problem 2.22 (unimodal function). Consider

$$F_{10}(x) = \sum_{i=1}^{n} |y_i| + \prod_{i=1}^{n} |y_i|, \quad y = M \times x. \tag{21}$$

(11) Rotated 2^n Minima Function (multimodal function). Consider

$$F_{11}(x) = \frac{1}{n} \sum_{i=1}^{n} \left(y_i^4 - 16 y_i^2 + 5 y_i \right), \quad y = M \times x. \tag{22}$$

(12) Rotated Rastrigin's Function (multimodal function). Consider

$$F_{12}(x) = \sum_{i=1}^{n} \left[y_i^2 - 10 \cos(2\pi y_i) + 10 \right], \quad y = M \times x. \tag{23}$$

(13) Rotated Weierstrass' Function (multimodal function). Consider

$$F_{13}(x) = \sum_{i=1}^{n} \left(\sum_{k=0}^{k\max} \left[a^k \cos\left(2\pi b^k (y_i + 0.5)\right) \right] \right) - n \sum_{k=0}^{k\max} \left[a^k \cos\left(2\pi b^k \times 0.5\right) \right], \quad y = M \times x, \quad a = 0.5, \quad b = 3, \quad k\max = 20. \tag{24}$$

(14) Rotated Salomon's Function (multimodal function). Consider

$$F_{14}(x) = 1 - \cos\left(2\pi \sqrt{\sum_{i=1}^{n} y_i^2} \right) + 0.1 \sqrt{\sum_{i=1}^{n} y_i^2}, \quad y = M \times x. \tag{25}$$

Table 5: Comparison of FGLT-PSO with PSO, LPSO, and QIPSO for the rotated and shifted unimodal and multimodal functions using Wilcoxon's rank sum test ($n = 10$).

Function	Wilcoxon's rank sum test	PSO	LPSO	QIPSO
F_{10}	P value	$3.0199e-011$	$3.0199e-011$	$3.0199e-011$
	h-value	**1**	**1**	**1**
	z-value	−6.6456	−6.6456	−6.6456
F_{11}	P value	$4.1191e-001$	$7.3940e-001$	$8.4180e-001$
	h-value	0	0	0
	z-value	0.820536	0.33265	0.19959
F_{12}	P value	$1.8567e-009$	$1.8500e-008$	$5.5329e-008$
	h-value	**1**	**1**	**1**
	z-value	−6.00987	−5.62547	−5.43328
F_{13}	P value	$2.1577e-002$	$2.1577e-002$	$8.1523e-002$
	h-value	**1**	**1**	0
	z-value	−2.29773	−2.29773	−1.74192
F_{14}	P value	$1.1439e-001$	$1.4425e-004$	$7.7181e-001$
	h-value	0	−1	0
	z-value	1.57878	3.80076	0.290007
F_{15}	P value	$1.5553e-011$	$1.2883e-011$	$1.7966e-008$
	h-value	**1**	**1**	**1**
	z-value	−6.74264	−6.76995	−5.63053
F_{16}	P value	$1.2755e-010$	$1.9667e-003$	$5.5338e-010$
	h-value	**1**	**1**	**1**
	z-value	−6.43006	−3.09521	−6.20317
F_{17}	P value	$7.3270e-011$	$3.6443e-008$	$5.0030e-010$
	h-value	**1**	**1**	**1**
	z-value	−6.51381	−5.50727	−6.21901
F_{18}	P value	$5.7512e-006$	$3.5043e-007$	$4.0269e-005$
	h-value	**1**	**1**	**1**
	z-value	−4.53533	−5.09408	−4.10593
F_{19}	P value	$9.2344e-001$	$2.3885e-004$	$8.6499e-001$
	h-value	0	−1	0
	z-value	0.0960988	3.67393	0.170021
F_{20}	P value	$5.2991e-008$	$9.2051e-005$	$4.6756e-008$
	h-value	**1**	**1**	**1**
	z-value	−5.44097	−3.91064	−5.46322
1 (better)		**8**	**8**	**7**
0 (same)		3	1	4
−1 (worse)		0	2	0

(15) Shifted Schwefel's Problem 2.21 (unimodal function). Consider

$$F_{15}(x) = \max_i\{|z_i|, 1 \le i \le n\} + f\text{bias}_{15},$$
$$f\text{bias}_{15} = -450, \quad z = x - o. \tag{26}$$

(16) Shifted Generalized Griewank's Function (multimodal function). Consider

$$F_{16}(x) = \frac{1}{4000}\sum_{i=1}^{n} z_i^2 - \prod_{i=1}^{n}\cos\left(\frac{z_i}{\sqrt{i}}\right) + 1 + f\text{bias}_{16},$$
$$f\text{bias}_{16} = -180, \quad z = x - o. \tag{27}$$

(17) Shifted Rosenbrock's Function (multimodal function). Consider

$$F_{17}(x) = \sum_{i=1}^{n-1}\left[100\left(z_i^2 - z_{i+1}\right)^2 + \left(z_i - 1\right)^2\right] + f\text{bias}_{17},$$
$$f\text{bias}_{17} = 390, \quad z = x - o + 1. \tag{28}$$

(18) Shifted Rastrigin's Function (multimodal function). Consider

$$F_{18}(x) = \sum_{i=1}^{n}\left[z_i^2 - 10\cos\left(2\pi z_i\right) + 10\right] + f\text{bias}_{18},$$
$$f\text{bias}_{18} = -330, \quad z = x - o. \tag{29}$$

Table 6: Minimization results for the unimodal and multimodal functions (maximum iteration = 10000 and $n = 30$).

	Function	FGLT-PSO	PSO	LPSO	QIPSO
F_1	Avg. best solution	**0.000e + 000**	5.397e − 069	2.747e − 028	3.502e − 069
	SD	**0.000e + 000**	1.161e − 068	4.078e − 028	1.664e − 068
	Median best solution	0.000e + 000	2.321e − 070	1.047e − 028	2.727e − 072
F_2	Avg. best solution	**7.598e − 110**	7.000e + 000	3.849e − 028	2.333e + 000
	SD	**4.085e − 109**	7.944e + 000	9.015e − 028	4.302e + 000
	Median best solution	1.684e − 136	1.000e + 001	1.331e − 031	1.188e − 071
F_3	Avg. best solution	**5.959e − 020**	1.189e + 004	1.399e + 002	5.733e + 003
	SD	**1.762e − 019**	6.605e + 003	9.380e + 001	5.484e + 003
	Median best solution	5.483e − 021	1.083e + 004	1.254e + 002	5.000e + 003
F_4	Avg. best solution	6.415e − 003	**1.320e − 003**	9.664e − 003	1.635e − 003
	SD	2.858e − 003	**8.008e − 004**	3.621e − 003	9.357e − 004
	Median best solution	5.941e − 003	1.088e − 003	9.106e − 003	1.333e − 003
F_5	Avg. best solution	**5.718e − 001**	1.129e + 003	1.751e + 001	3.858e + 002
	SD	**1.225e + 000**	3.023e + 003	2.041e + 001	1.832e + 003
	Median best solution	1.707e − 001	7.428e + 001	9.494e + 000	2.139e + 001
F_6	Avg. best solution	**−1.048e + 004**	−9.095e + 003	−9.858e + 003	−9.227e + 003
	SD	5.996e + 002	7.444e + 002	4.872e + 002	7.524e + 002
	Median best solution	−1.058e + 004	−9.051e + 003	−9.828e + 003	−9.232e + 003
F_7	Avg. best solution	7.771e − 002	1.013e − 014	2.931e − 014	**8.941e − 015**
	SD	2.015e − 001	3.312e − 015	1.433e − 014	**2.457e − 015**
	Median best solution	7.994e − 015	7.994e − 015	2.576e − 014	7.994e − 015
F_8	Avg. best solution	5.164e − 003	1.037e − 002	1.160e − 025	**1.571e − 032**
	SD	2.274e − 002	3.163e − 002	3.552e − 025	**5.567e − 048**
	Median best solution	1.571e − 032	1.591e − 032	3.737e − 027	1.571e − 032
F_9	Avg. best solution	**2.136e + 001**	5.827e + 001	4.281e + 001	4.310e + 001
	SD	**8.612e + 000**	3.080e + 001	2.064e + 001	2.901e + 001
	Median best solution	2.212e + 001	5.450e + 001	3.753e + 001	3.500e + 001
	Avg. rank	**1.8**	3.3	2.6	2.3
	Final rank	**1**	4	3	2
	Algorithms	**FGLT-PSO**	PSO	LPSO	QIPSO

(19) Shifted Rotated Ackley's Function (multimodal function). Consider

$$F_{19}(x) = -20\exp\left(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} z_i^2}\right) - \exp\left(\frac{1}{n}\sum_{i=1}^{n}\cos 2\pi z_i\right) + 20 + e + f\text{bias}_{19}, \quad (30)$$

where $f\text{bias}_{19} = -140$; $z = (x - o) \times M'$ is a linear transformation matrix with condition number = 100.

(20) Shifted Rotated Rastrigin's Function (multimodal function). Consider

$$F_{20}(x) = \sum_{i=1}^{n}\left[z_i^2 - 10\cos(2\pi z_i) + 10\right] + f\text{bias}_{20}, \quad (31)$$

where $f\text{bias}_{20} = -330$; $z = (x - o) \times M'$ is a linear transformation matrix with condition number = 2.

4.2. Results of FGLT-PSO. The results of FGLT-PSO are provided in three sections. In Section 4.2.1, the acceleration coefficients C_1, C_2, and C_3 in the proposed method are changed according to (10) and in Section 4.2.2, these factors are constant. In these sections, FGLT-PSO is evaluated using the benchmark functions with dimensions 10, 30, and 50. The number of maximum iterations is set at 5000 for $n = 10$, at 10000 for $n = 30$, and at 15000 for $n = 50$. The population size is set to 50 ($N = 50$). Also, w decreases linearly from 0.9 to 0.4.

In Section 4.2.3, the results of FGLT-PSO are compared with those of several well-known PSO algorithms from [19] on the common functions. In this section, the population size is set to 30 ($N = 30$), n is 30, and the number of maximum iterations is set at 10000.

The ring topology is used as the neighborhood structure in the $l$best model for the FGLT-PSO and LPSO algorithms and the number of neighbours for each particle is three. The algorithms are run independently 30 times for the benchmark functions and the results are averaged.

Table 7: Minimization results for the rotated and shifted unimodal and multimodal functions (maximum iteration = 10000 and $n = 30$).

Function		FGLT-PSO	PSO	LPSO	QIPSO
F_{10}	Avg. best solution	**8.707e − 020**	3.422e + 002	7.818e + 000	5.318e + 001
	SD	**4.547e − 019**	1.556e + 003	2.173e + 001	1.986e + 002
	Median best solution	4.028e − 043	1.332e − 006	2.267e − 003	8.579e − 020
F_{11}	Avg. best solution	−4.963e + 001	−4.981e + 001	−4.933e + 001	**−4.993e + 001**
	SD	1.098e + 000	1.324e + 000	1.417e + 000	1.489e + 000
	Median best solution	−4.930e + 001	−4.985e + 001	−4.927e + 001	−4.968e + 001
F_{12}	Avg. best solution	**1.514e + 002**	1.935e + 002	1.902e + 002	1.886e + 002
	SD	**8.330e + 000**	1.509e + 001	1.246e + 001	1.233e + 001
	Median best solution	1.511e + 002	1.978e + 002	1.899e + 002	1.850e + 002
F_{13}	Avg. best solution	**7.160e − 001**	3.101e + 001	1.185e + 001	2.046e + 001
	SD	**9.412e − 001**	1.248e + 001	1.136e + 001	1.804e + 001
	Median best solution	2.067e − 001	3.613e + 001	5.997e + 000	3.319e + 001
F_{14}	Avg. best solution	4.065e − 001	**3.465e − 001**	3.632e − 001	3.599e − 001
	SD	1.230e − 001	**5.074e − 002**	5.561e − 002	7.240e − 002
	Median best solution	3.999e − 001	2.999e − 001	3.999e − 001	3.499e − 001
F_{15}	Avg. best solution	−4.280e + 002	−4.044e + 002	−4.175e + 002	**−4.500e + 002**
	SD	7.441e + 000	7.592e + 000	8.860e + 000	2.151e − 006
	Median best solution	−4.266e + 002	−4.056e + 002	−4.155e + 002	−4.500e + 002
F_{16}	Avg. best solution	**−1.794e + 002**	−8.340e + 001	−1.705e + 002	−9.806e + 001
	SD	**1.112e + 000**	4.560e + 001	4.344e + 000	3.839e + 001
	Median best solution	−1.798e + 002	−9.269e + 001	−1.704e + 002	−1.053e + 002
F_{17}	Avg. best solution	**4.225e + 002**	1.658e + 009	8.168e + 007	1.549e + 009
	SD	**5.410e + 001**	1.409e + 009	5.644e + 007	1.599e + 009
	Median best solution	3.940e + 002	1.136e + 009	6.932e + 007	1.205e + 009
F_{18}	Avg. best solution	**−2.874e + 002**	−2.174e + 002	−2.455e + 002	−2.219e + 002
	SD	**1.680e + 001**	2.897e + 001	1.196e + 001	2.614e + 001
	Median best solution	−2.902e + 002	−2.199e + 002	−2.458e + 002	−2.227e + 002
F_{19}	Avg. best solution	−1.1912e + 002	−1.1912e + 002	**−1.1915e + 002**	−1.1913e + 002
	SD	5.597e − 002	5.015e − 002	5.377e − 002	6.957e − 002
	Median best solution	−1.1911e + 002	−1.1911e + 002	−1.1915e + 002	−1.1913e + 002
F_{20}	Avg. best solution	**−2.367e + 002**	−1.237e + 002	−1.499e + 002	−1.375e + 002
	SD	**2.561e + 001**	3.612e + 001	2.719e + 001	4.160e + 001
	Median best solution	−2.427e + 002	−1.256e + 002	−1.481e + 002	−1.374e + 002
Avg. rank		**1.8**	3.5	2.4	2.4
Final rank		**1**	3	2	2
Algorithms		**FGLT-PSO**	PSO	LPSO	QIPSO

4.2.1. The Results of Proposed Method with Variable Acceleration Coefficients. Four algorithms of FGLT-PSO, PSO, LPSO, and QIPSO are randomly initialized and run on benchmark functions. The average best solution, the standard deviation (SD), and the median of the best solution in the last iteration are reported in Tables 2, 3, 6, 7, 10, and 11. The best results from among the algorithms are shown in bold numbers. In the tables, the algorithms are ranked based on the average best results.

Moreover, Wilcoxon's rank sum test [37] is conducted in order to determine whether the results obtained by the FGLT-PSO are different from those generated by other algorithms with a statistical significance. The tests are shown in Tables 4, 5, 8, 9, 12, and 13, where h-value = 1 indicates the case in which proposed algorithm significantly outperformed the compared algorithm with 95% certainty, h-value = −1 represents that the compared algorithm is significantly better than the proposed algorithm, and h-value = 0 denotes that the results of the two considered algorithms are not significantly different. In these tables, rows 1 (better), 0 (same), and −1 (worse) give the number of functions that the FGLT-PSO performs significantly better than, almost the same as, and significantly worse than the compared algorithm, respectively.

The acceleration coefficients C_1, C_2, and C_3 are updated based on (10). Their minimum and maximum values are as follows: $C_{1\,\min} = 0.5$, $C_{1\,\max} = 2$, $C_{2\,\min} = 1$, $C_{2\,\max} = 2$, $C_{3\,\min} = 0.5$, and $C_{3\,\max} = 1.5$.

Table 8: Comparison of FGLT-PSO with PSO, LPSO, and QIPSO for the unimodal and multimodal functions using Wilcoxon's rank sum test ($n = 30$).

Function	Wilcoxon's rank sum test	PSO	LPSO	QIPSO
	P value	$1.2118e-012$	$1.2118e-012$	$1.2118e-012$
F_1	h-value	**1**	**1**	**1**
	z-value	−7.10402	−7.10402	−7.10402
	P value	$2.8991e-011$	$3.0199e-011$	$2.9822e-011$
F_2	h-value	**1**	**1**	**1**
	z-value	−6.65161	−6.6456	−6.64745
	P value	$3.0199e-011$	$3.0199e-011$	$3.0199e-011$
F_3	h-value	**1**	**1**	**1**
	z-value	−6.6456	−6.6456	−6.6456
	P value	$1.2057e-010$	$2.2539e-004$	$1.9568e-010$
F_4	h-value	−1	**1**	−1
	z-value	6.43862	−3.68871	6.3647
	P value	$1.2018e-008$	$1.2472e-004$	$5.0922e-008$
F_5	h-value	**1**	**1**	**1**
	z-value	−5.69948	−3.83667	−5.44806
	P value	$1.2018e-008$	$1.2472e-004$	$5.0922e-008$
F_6	h-value	**1**	**1**	**1**
	z-value	−5.69948	−3.83667	−5.44806
	P value	$4.2942e-002$	$3.7277e-001$	$2.8314e-003$
F_7	h-value	−1	0	−1
	z-value	2.02428	−0.891289	2.98547
	P value	$1.7374e-001$	$2.4305e-002$	$6.6167e-004$
F_8	h-value	0	**1**	−1
	z-value	−1.36028	−2.25228	3.40499
	P value	$1.9990e-006$	$2.7726e-005$	$7.2926e-004$
F_9	h-value	**1**	**1**	**1**
	z-value	−4.75352	−4.19138	−3.37834
	1 (better)	**6**	**8**	**6**
	0 (same)	1	1	0
	−1 (worse)	2	0	3

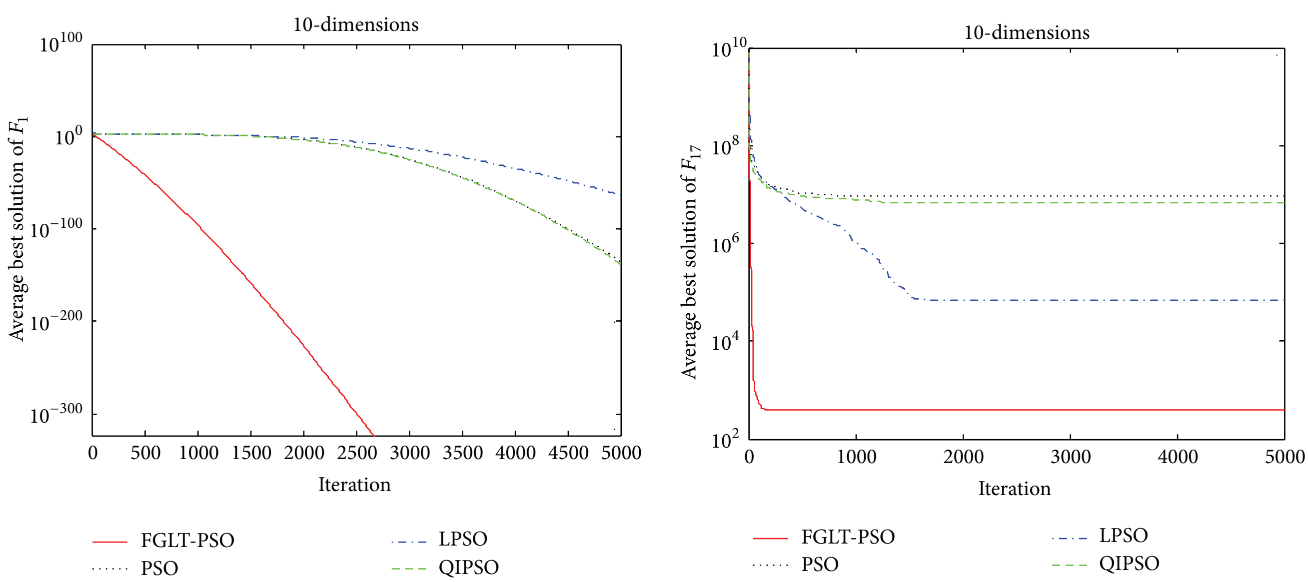

Figure 2: Convergence performance of FGLT-PSO, PSO, LPSO, and QIPSO for the test functions F_1 and F_{17} ($n = 10$).

Table 9: Comparison of FGLT-PSO with PSO, LPSO, and QIPSO for the rotated and shifted unimodal and multimodal functions using Wilcoxon's rank sum test ($n = 30$).

Function	Wilcoxon's rank sum test	PSO	LPSO	QIPSO
F_{10}	P value	$3.2922e-010$	$1.4041e-010$	$3.1451e-009$
	h-value	**1**	**1**	**1**
	z-value	−6.28435	−6.41545	−5.92384
F_{11}	P value	$1.0233e-001$	$3.0418e-001$	$4.4642e-001$
	h-value	0	0	0
	z-value	1.63368	−1.02752	0.761398
F_{12}	P value	$1.0937e-010$	$1.4643e-010$	$3.0199e-011$
	h-value	**1**	**1**	**1**
	z-value	−6.4534	−6.40905	−6.6456
F_{13}	P value	$1.4288e-008$	$1.3281e-010$	$1.1736e-003$
	h-value	**1**	**1**	**1**
	z-value	−5.66991	−6.42392	−3.24523
F_{14}	P value	$1.3470e-003$	$6.7707e-003$	$3.5527e-003$
	h-value	−1	−1	−1
	z-value	3.20578	2.70792	2.91537
F_{15}	P value	$2.8840e-010$	$4.7657e-005$	$2.8252e-011$
	h-value	**1**	**1**	−1
	z-value	−6.30489	−4.06683	6.65541
F_{16}	P value	$2.9543e-011$	$5.3712e-011$	$2.9543e-011$
	h-value	**1**	**1**	**1**
	z-value	−6.64883	−6.56027	−6.64883
F_{17}	P value	$3.0199e-11$	$3.0199e-11$	$3.0199e-11$
	h-value	**1**	**1**	**1**
	z-value	−6.6456	−6.6456	−6.6456
F_{18}	P value	$1.0937e-010$	$1.2870e-009$	$1.4643e-010$
	h-value	**1**	**1**	**1**
	z-value	−6.4534	−6.06901	−6.40905
F_{19}	P value	$8.7663e-001$	$4.0595e-002$	$7.7312e-001$
	h-value	0	**1**	0
	z-value	0.155236	2.04764	0.288296
F_{20}	P value	$3.0199e-011$	$7.3891e-011$	$4.9752e-011$
	h-value	**1**	**1**	**1**
	z-value	−6.6456	−6.51254	−6.57168
1 (better)		**8**	**9**	**7**
0 (same)		2	1	2
−1 (worse)		1	1	2

In these tables, the benchmark functions are divided to two categories: (1) unimodal and multimodal functions and (2) rotated, shifted, and shifted-rotated unimodal and multimodal functions. The experimental results demonstrate that FGLT-PSO performs superior results for most of the functions in all tested dimensions.

Tables 2 and 3 show the experimental results for all benchmark functions with dimension $n = 10$. As illustrated, the FGLT-PSO algorithm surpasses the PSO, LPSO, and QIPSO algorithms in minimizing functions (1)–(6), (10), (12), (13), (15)–(18), and (20). Moreover, the proposed method provides significant improvements in functions (1), (2), (3), (10), (13), (17), and (18). In these functions, the convergent results attain the optimal (or good near optimal) solutions. In Tables 2 and 3, the average iteration for finding the best solution is computed. The average iteration is the required iterations to find the best solution by each algorithm. As shown, FGLT-PSO finds the best solutions faster than the other algorithms in the majority of functions. Also, it is noticeable that the FGLT-PSO algorithm achieves the best solution in a considerably lower iteration in functions (7) and (8). In these functions, the algorithms show identical results.

According to Wilcoxon's rank sum test in Tables 4 and 5 for $n = 10$, the results of the FGLT-PSO are statistically significantly different from the three compared algorithms. The superior convergence rate of FGLT-PSO is shown in

Table 10: Minimization results for the unimodal and multimodal functions (maximum iteration = 15000 and n = 50).

Function		FGLT-PSO	PSO	LPSO	QIPSO
F_1	Avg. best solution	**5.239e − 232**	7.5857e − 049	7.297e − 020	2.446e − 049
	SD	**0.000e + 000**	2.1243e − 048	7.695e − 020	8.478e − 049
	Median best solution	2.541e − 251	4.5415e − 050	4.843e − 020	1.395e − 050
F_2	Avg. best solution	**9.246e − 075**	3.400e + 001	2.194e − 015	1.367e + 001
	SD	**3.646e − 074**	1.714e + 001	1.439e − 015	1.159e + 001
	Median best solution	2.684e − 080	3.000e + 001	1.881e − 015	1.000e + 001
F_3	Avg. best solution	**1.098e − 008**	4.167e + 004	2.775e + 004	3.585e + 004
	SD	**2.405e − 008**	1.404e + 004	8.532e + 003	1.967e + 004
	Median best solution	5.928e − 009	4.002e + 004	2.859e + 004	3.434e + 004
F_4	Avg. best solution	5.070e − 002	6.015e + 000	6.452e − 002	**2.055e − 002**
	SD	2.802e − 002	5.352e + 000	1.488e − 002	**4.363e − 003**
	Median best solution	4.205e − 002	5.385e + 000	6.409e − 002	1.971e − 002
F_5	Avg. best solution	**6.128e + 000**	1.568e + 003	6.024e + 001	4.700e + 002
	SD	**5.716e + 000**	3.409e + 003	3.032e + 001	1.817e + 003
	Median best solution	5.080e + 000	8.179e + 001	7.397e + 001	7.942e + 001
F_6	Avg. best solution	**−1.505e + 004**	−1.364e + 004	−1.470e + 004	−1.339e + 004
	SD	9.481e + 002	9.581e + 002	8.491e + 002	1.025e + 003
	Median best solution	−1.515e + 004	−1.349e + 004	−1.445e + 004	−1.343e + 004
F_7	Avg. best solution	1.299e + 000	2.049e + 000	1.145e − 005	**1.782e − 014**
	SD	5.560e − 001	4.661e + 000	6.223e − 005	**3.695e − 015**
	Median best solution	1.282e + 000	2.220e − 014	1.782e − 009	1.510e − 014
F_8	Avg. best solution	1.061e − 001	1.451e − 002	**3.831e − 012**	2.074e − 003
	SD	1.935e − 001	3.535e − 002	**1.984e − 011**	1.136e − 002
	Median best solution	2.081e − 005	1.308e − 032	2.881e − 015	9.423e − 033
F_9	Avg. best solution	**4.759e + 001**	1.980e + 002	1.872e + 002	1.546e + 002
	SD	**2.294e + 001**	4.888e + 001	3.958e + 001	5.192e + 001
	Median best solution	4.636e + 001	1.890e + 002	1.850e + 002	1.475e + 002
Avg. rank		**1.7**	3.7	2.3	2.3
Final rank		**1**	3	2	2
Algorithms		**FGLT-PSO**	PSO	LPSO	QIPSO

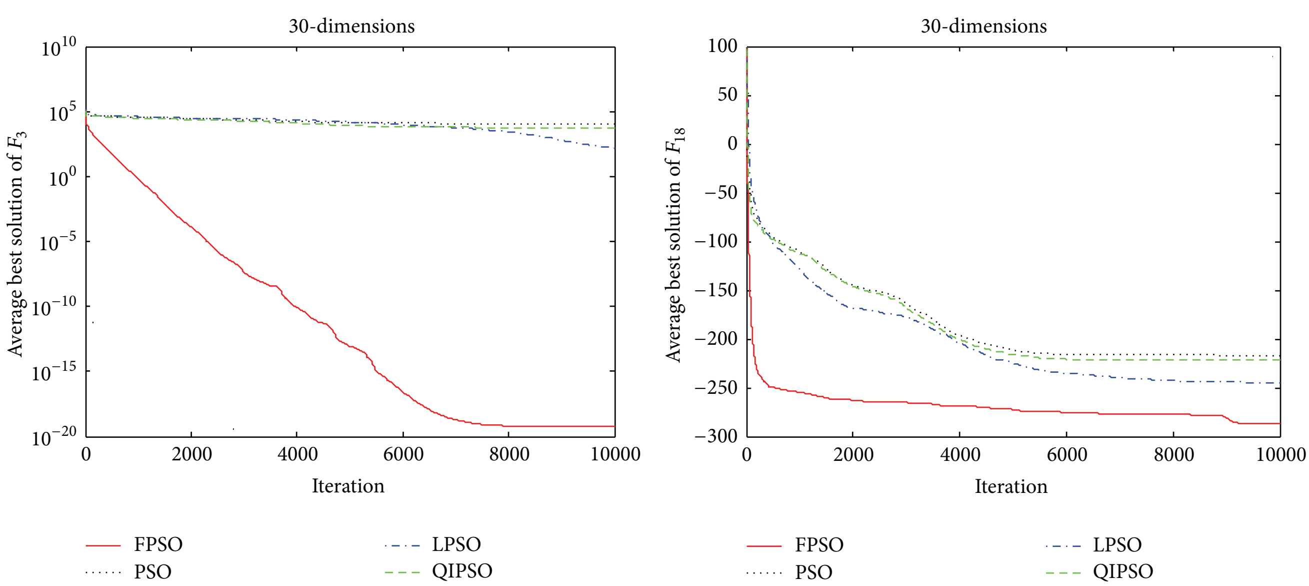

Figure 3: Convergence performance of FGLT-PSO, PSO, LPSO, and QIPSO for the test functions F_3 and F_{18} (n = 30).

Table 11: Minimization results for the rotated and shifted unimodal and multimodal functions (maximum iteration = 15000 and $n = 50$).

	Function	FGLT-PSO	PSO	LPSO	QIPSO
F_{10}	Avg. best solution	**9.936e − 031**	6.202e + 009	5.150e + 003	3.989e + 006
	SD	**5.442e − 030**	2.576e + 010	1.153e + 004	1.825e + 007
	Median best solution	1.121e − 048	1.763e + 005	1.184e + 002	5.637e − 007
F_{11}	Avg. best solution	**−4.401e + 001**	−4.328e + 001	−4.357e + 001	−4.385e + 001
	SD	**1.043e + 000**	1.331e + 000	9.431e − 001	1.329e + 000
	Median best solution	−4.391e + 001	−4.344e + 001	−4.339e + 001	−4.367e + 001
F_{12}	Avg. best solution	**3.155e + 002**	4.234e + 002	3.995e + 002	4.045e + 002
	SD	**1.166e + 001**	3.689e + 001	1.813e + 001	2.468e + 001
	Median best solution	3.171e + 002	4.236e + 002	3.995e + 002	4.062e + 002
F_{13}	Avg. best solution	**1.660e + 001**	6.640e + 001	5.210e + 001	4.946e + 001
	SD	**6.875e + 000**	2.371e + 000	1.517e + 001	2.599e + 001
	Median best solution	1.759e + 001	6.697e + 001	6.163e + 001	6.361e + 001
F_{14}	Avg. best solution	7.599e − 001	**5.899e − 001**	8.067e − 001	6.099e − 001
	SD	3.103e − 001	**8.030e − 002**	9.046e − 002	8.847e − 002
	Median best solution	6.999e − 001	5.999e − 001	7.999e − 001	5.999e − 001
F_{15}	Avg. best solution	−3.976e + 002	−3.795e + 002	−4.019e + 002	**−4.499e + 002**
	SD	5.964e + 000	1.974e + 001	6.096e + 000	**1.849e − 003**
	Median best solution	−3.974e + 002	−3.815e + 002	−4.021e + 002	−4.50e + 002
F_{16}	Avg. best solution	**−1.791e + 002**	8.313e + 001	−1.551e + 002	6.894e + 001
	SD	**1.212e + 000**	9.121e + 001	1.216e + 001	7.896e + 001
	Median best solution	−1.797e + 002	7.911e + 001	−1.565e + 002	7.481e + 001
F_{17}	Avg. best solution	**4.260e + 002**	8.648e + 009	3.341e + 008	8.196e + 009
	SD	**7.188e + 001**	4.442e + 009	2.597e + 008	5.588e + 009
	Median best solution	3.977e + 002	8.332e + 009	2.599e + 008	6.614e + 009
F_{18}	Avg. best solution	**−1.920e + 002**	−4.497e + 001	−9.832e + 001	−3.612e + 001
	SD	**2.634e + 001**	4.206e + 001	1.963e + 001	4.934e + 001
	Median best solution	−1.944e + 002	−4.804e + 001	−9.679e + 001	−2.665e + 001
F_{19}	Avg. best solution	−1.189e + 002	−1.189e + 002	**−1.190e + 002**	−1.189e + 002
	SD	4.303e − 002	4.364e − 002	5.121e − 002	5.576e − 002
	Median best solution	−1.189e + 002	−1.189e + 002	−1.190e + 002	−1.189e + 002
F_{20}	Avg. best solution	**−1.016e + 002**	1.537e + 002	5.974e + 001	1.082e + 002
	SD	**4.898e + 001**	8.906e + 001	5.117e + 001	9.020e + 001
	Median best solution	−1.004e + 002	1.539e + 002	6.594e + 001	1.035e + 002
	Avg. rank	**1.5**	3.6	2.3	2.7
	Final rank	**1**	4	2	3
	Algorithms	**FGLT-PSO**	PSO	LPSO	QIPSO

Figure 2. The results in this figure illustrate that FGLT-PSO tends to find the global optimum in F_1 and F_{17} faster than PSO, LPSO, and QIPSO and obtains the highest accuracy for these functions from among all the algorithms.

The minimization results of the benchmark functions with dimension $n = 30$ are presented in Tables 6 and 7. As seen in these tables and Tables 8 and 9, the FGLT-PSO outperforms the PSO, LPSO, and QIPSO algorithms in functions (1), (2), (3), (5), (6), (9), (10), (12), (13), (16), (17), (18), and (20). The largest difference in performance between the proposed algorithm with PSO, LPSO, and QIPSO occurs for the functions (1), (2), (3), (5), (16), (17), (18), and (20). Figure 3 illustrates the progress of the average best solution over 30 runs for F_3 and F_{18}. As demonstrated, the FGLT-PSO shows a higher convergence rate than the other algorithms.

Tables 10 and 11 present the results of algorithms for the test functions with dimension $n = 50$. As illustrated in these tables and regarding the results of Wilcoxon's rank sum in Tables 12 and 13, the performance of proposed method is the best in most of the functions especially for functions (1), (2), (3), (5), (6), (9), (10), (11), (12), (13), (16), (17), (18), and (20). The superior convergence rate of FGLT-PSO is shown in Figure 4. The results in this figure show that the FGLT-PSO performs the best for the test functions F_{16} and F_{20} with $n = 50$.

In addition, it is considerable that the PSO, LPSO, and QIPSO algorithms return the results far from the global optima as the dimension increases. This problem is clear in the functions (3), (5), (9), (10), (16), (17), (18), and (20) in Tables 10 and 11 with $n = 50$. These all indicate that

Table 12: Comparison of FGLT-PSO with PSO, LPSO, and QIPSO for the unimodal and multimodal functions using Wilcoxon's rank sum test ($n = 50$).

Function	Wilcoxon's rank sum test	PSO	LPSO	QIPSO
F_1	P value	$3.0199e-011$	$3.0199e-011$	$3.0199e-011$
	h-value	**1**	**1**	**1**
	z-value	−6.6456	−6.6456	−6.6456
F_2	P value	$2.9673e-011$	$3.0199e-011$	$2.9229e-011$
	h-value	**1**	**1**	**1**
	z-value	−6.64819	−6.6456	−6.65041
F_3	P value	$3.0199e-011$	$3.0199e-011$	$3.0199e-011$
	h-value	**1**	**1**	**1**
	z-value	−6.6456	−6.6456	−6.6456
F_4	P value	$2.5306e-004$	$3.9881e-004$	$3.1589e-010$
	h-value	**1**	**1**	−1
	z-value	−3.65915	−3.54087	6.29077
F_5	P value	$6.1210e-010$	$9.7555e-010$	$1.2870e-009$
	h-value	**1**	**1**	**1**
	z-value	−6.18728	−6.11336	−6.06901
F_6	P value	$2.8790e-006$	$9.3341e-002$	$2.5711e-007$
	h-value	**1**	0	**1**
	z-value	−4.67927	−1.67803	−5.15244
F_7	P value	$7.7050e-006$	$3.0199e-011$	$1.4811e-011$
	h-value	−1	−1	−1
	z-value	4.47322	6.6456	6.74974
F_8	P value	$7.7087e-002$	$8.2796e-003$	$1.4193e-007$
	h-value	0	−1	−1
	z-value	1.76784	2.64045	5.26273
F_9	P value	$3.3384e-011$	$3.3384e-011$	$4.6159e-010$
	h-value	**1**	**1**	**1**
	z-value	−6.63081	−6.63081	−6.23164
	1 (better)	**7**	**6**	**6**
	0 (same)	1	1	0
	−1 (worse)	1	2	3

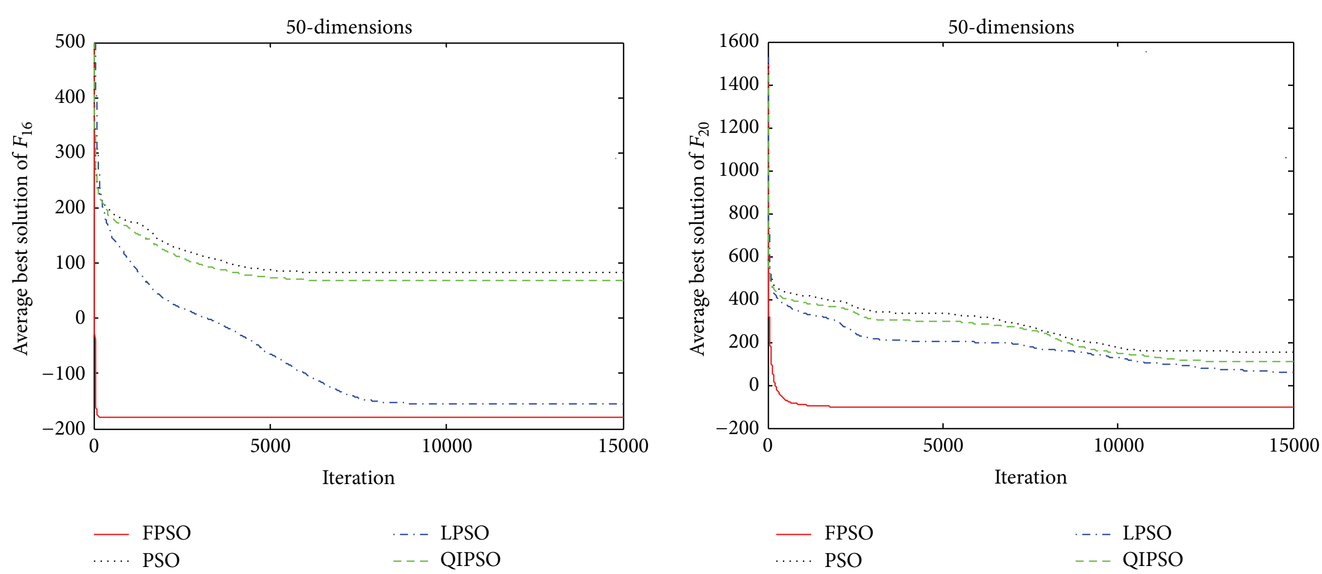

Figure 4: Convergence performance of FGLT-PSO, PSO, LPSO, and QIPSO for the test functions F_{16} and F_{20} ($n = 50$).

Table 13: Comparison of FGLT-PSO with PSO, LPSO, and QIPSO for the rotated and shifted unimodal and multimodal functions using Wilcoxon's rank sum test ($n = 50$).

Function	Wilcoxon's rank sum test	PSO	LPSO	QIPSO
F_{10}	P value	$3.0199e-011$	$3.0199e-011$	$3.0199e-011$
	h-value	**1**	**1**	**1**
	z-value	−6.6456	−6.6456	−6.6456
F_{11}	P value	$3.0317e-002$	$6.5671e-002$	$6.7350e-001$
	h-value	**1**	0	0
	z-value	−2.16592	−1.84066	−0.421356
F_{12}	P value	$3.0199e-011$	$3.0199e-011$	$3.0199e-011$
	h-value	**1**	**1**	**1**
	z-value	−6.6456	−6.6456	−6.6456
F_{13}	P value	$3.0199e-011$	$1.2057e-010$	$1.8916e-004$
	h-value	**1**	**1**	**1**
	z-value	−6.6456	−6.43862	−3.73307
F_{14}	P value	$7.3131e-003$	$3.0749e-002$	$3.0494e-002$
	h-value	−1	**1**	−1
	z-value	2.68224	−2.16031	2.16361
F_{15}	P value	$2.2256e-004$	$5.4038e-004$	$2.7547e-011$
	h-value	**1**	−1	−1
	z-value	−3.69193	3.4599	6.65912
F_{16}	P value	$3.0142e-011$	$3.0123e-011$	$3.0142e-011$
	h-value	**1**	**1**	**1**
	z-value	−6.64588	−6.64597	−6.64588
F_{17}	P value	$3.0199e-011$	$3.0199e-011$	$3.0199e-011$
	h-value	**1**	**1**	**1**
	z-value	−6.6456	−6.6456	−6.6456
F_{18}	P value	$3.0199e-011$	$4.0772e-011$	$3.0199e-011$
	h-value	**1**	**1**	**1**
	z-value	−6.6456	−6.60125	−6.6456
F_{19}	P value	$2.3399e-001$	$5.8737e-004$	$9.2344e-001$
	h-value	0	−1	0
	z-value	1.19015	3.43738	0.0960988
F_{20}	P value	$3.3384e-011$	$1.3289e-010$	$8.9934e-011$
	h-value	**1**	**1**	**1**
	z-value	−6.63081	−6.42383	−6.48297
	1 (better)	**9**	**8**	**7**
	0 (same)	1	1	2
	−1 (worse)	1	2	2

the proposed algorithm, FGLT-PSO, is more powerful and robust than the others for solving unimodal and multimodal functions.

4.2.2. The Results of Proposed Method with Constant Acceleration Coefficients. In this section, the C_1, C_2, and C_3 as acceleration coefficients are set at constant values to compare with the presented results in the Section 4.2.1. The coefficient of cognition term (C_1) and social terms (C_2 and C_3) are considered as $C_1 = 2$ and $C_2 = C_3 = 1$. Table 14 shows the results of proposed method for the benchmark functions with dimensions 10, 30, and 50. As seen, the FGLT-PSO with the constant acceleration coefficients performs well in most of the functions. As the dimension increases, the FGLT-PSO with the variable acceleration coefficients (Section 4.2.1) shows the better performance than the constant one for functions (1), (2), (3), (5), (17), and (20). Also, the FGLT-PSO with the constant acceleration coefficients presents a better performance for functions (7), (13), and (18).

4.2.3. Comparison with the Other PSO Algorithms. In this section, several well-known PSO algorithms are selected to assess the performance of proposed algorithm for the benchmarks. The PSO, QIPSO, FIPS, DMS-PSO, CLPSO, AFPSO, and AFPSO-QI algorithms are considered for the comparison. The details of these algorithms are listed in Table 15. The FGLT-PSO is run 30 times and the average best

Table 14: The results of FGLT-PSO with constant acceleration coefficients and different dimensions ($N = 50$).

Functions	Iteration = 5000, $n = 10$ Avg. best solution ± SD	Iteration = 10000, $n = 30$ Avg. best solution ± SD	Iteration = 15000, $n = 50$ Avg. best solution ± SD
F_1	$0.000e + 000 \pm 0.000e + 000$	$1.936e - 067 \pm 1.060e - 066$	$1.687e - 038 \pm 5.602e - 038$
F_2	$6.678e - 160 \pm 3.658e - 159$	$4.922e - 049 \pm 2.305e - 048$	$7.064e - 032 \pm 3.167e - 031$
F_3	$7.520e - 035 \pm 3.579e - 034$	$2.026e - 003 \pm 9.321e - 003$	$2.062e + 000 \pm 2.616e + 000$
F_4	$1.621e - 004 \pm 8.583e - 005$	$9.078e - 004 \pm 3.306e - 004$	$1.930e - 003 \pm 6.461e - 004$
F_5	$3.086e + 000 \pm 1.772e + 000$	$3.542e + 001 \pm 2.864e + 001$	$7.854e + 001 \pm 4.435e + 001$
F_6	$-3.957e + 003 \pm 1.375e + 002$	$-1.052e + 004 \pm 4.069e + 002$	$-1.643e + 004 \pm 7.233e + 002$
F_7	$4.086e - 015 \pm 1.084e - 015$	$5.507e - 015 \pm 1.656e - 015$	$9.484e - 002 \pm 3.612e - 001$
F_8	$4.712e - 032 \pm 1.670e - 047$	$4.492e - 002 \pm 1.379e - 001$	$7.063e - 002 \pm 1.508e - 001$
F_9	$2.567e + 000 \pm 9.714e - 001$	$2.040e + 001 \pm 6.100e + 000$	$5.183e + 001 \pm 1.117e + 001$
F_{10}	$6.599e - 249 \pm 0.000e + 000$	$3.539e - 066 \pm 1.939e - 065$	$4.923e - 019 \pm 2.696e - 018$
F_{11}	$-6.620e + 001 \pm 1.111e + 000$	$-5.036e + 001 \pm 9.041e - 001$	$-4.482e + 001 \pm 9.061e - 001$
F_{12}	$1.659e + 001 \pm 4.479e + 000$	$1.602e + 002 \pm 1.096e + 001$	$3.369e + 002 \pm 1.585e + 001$
F_{13}	$0.000e + 000 \pm 0.000e + 000$	$0.000e + 000 \pm 0.000e + 000$	$1.5294e - 007 \pm 1.1896e - 007$
F_{14}	$9.987e - 002 \pm 2.247e - 017$	$3.932e - 001 \pm 1.081e - 001$	$3.065e - 001 \pm 2.537e - 002$
F_{15}	$-4.500e + 002 \pm 1.493e - 014$	$-4.268e + 002 \pm 1.016e + 001$	$-4.420e + 002 \pm 3.137e + 000$
F_{16}	$-1.799e + 002 \pm 2.005e - 001$	$-1.787e + 002 \pm 3.851e + 000$	$-1.775e + 002 \pm 3.055e + 000$
F_{17}	$4.001e + 002 \pm 2.060e + 001$	$4.661e + 002 \pm 1.083e + 002$	$5.748e + 005 \pm 1.022e + 006$
F_{18}	$-3.283e + 002 \pm 8.796e - 001$	$-2.956e + 002 \pm 8.540e + 000$	$-2.396e + 002 \pm 1.763e + 001$
F_{19}	$-1.1974e + 002 \pm 5.990e - 002$	$-1.191e + 002 \pm 6.798e - 002$	$-1.189e + 002 \pm 3.710e - 002$
F_{20}	$-3.196e + 002 \pm 4.136e + 000$	$-1.994e + 002 \pm 2.829e + 001$	$-2.684e + 001 \pm 4.506e + 001$

Table 15: Some well-known PSO algorithms in the literature.

Algorithm	Topology	Parameter settings
PSO	Global star	ω: 0.9–0.4, $C_1 = C_2 = 2.0$
QIPSO	Global star	ω: 0.9–0.4, $C_1 = C_2 = 2.0$
FIPS	Local U-ring	$\chi = 0.729, \sum ci = 4.1$
DMS-PSO	Dynamic multiswarm	ω: 0.9–0.2, $C_1 = C_2 = 2.0, m = 3, R = 5$
CLPSO	Comprehensive learning	ω: 0.9–0.4, $C = 1.49445, m = 7$
AFPSO	Global star	ω: 0.9–0.4, C_1, C_2 are based on fuzzy rule [19]
AFPSO-QI	Global star	ω: 0.9–0.4, C_1, C_2 are based on fuzzy rule [19]
FGLT-PSO	Global star and local ring	ω: 0.9–0.4, C_1: 0.5–2, C_2: 1-2, C_3: 0.5–1.5

solutions and the SD of results for eight common multimodal benchmark functions are compared with the reported results by [19] as illustrated in Table 16. The maximum iteration is 10000, $n = 30$, and $N = 30$. As seen, the FGLT-PSO provides better results than the other algorithms for the majority of functions (functions (11), (12), (13), (14), (17), and (20)) and has the first rank.

5. Conclusions

In this study, a fusion global-local-topology PSO algorithm (FGLT-PSO) has been presented to extend the search capability and to improve convergent efficiency by combining local and global topologies. The algorithm is a global search algorithm with several advantages. The benefits of algorithm can be summarized as the following: FGLT-PSO has a simple concept and structure; it is easy to implement and is not sensitive to increase of the dimension.

A set of standard benchmarks, including unimodal, multimodal, rotated, shifted, and shifted-rotated unimodal and multimodal functions, have been used to evaluate the proposed algorithm. The average best results obtained by the FGLT-PSO have been compared with PSO, LPSO, QIPSO, FIPS, DMS-PSO, CLPSO, AFPSO, and AFPSO-QI. The experimental results show that the proposed FGLT-PSO algorithm enhances the accuracy of results compared with the other algorithms.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Table 16: Comparison results of eight PSO algorithms [19] with FGLT-PSO for eight benchmark functions (maximum iteration = 10000, $n = 30$, and $N = 30$).

PSOs	Functions Avg. best solution ± SD	Avg. best solution ± SD	Avg. best solution ± SD	
	F_{11}	F_{12}	F_{13}	
PSO	$-4.529e+001 \pm 1.911e+000$	$3.202e+002 \pm 1.470e+001$	$3.835e+001 \pm 1.482e+000$	
QIPSO	$-3.332e+001 \pm 1.781e+000$	$3.175e+002 \pm 2.324e+001$	$4.077e+001 \pm 2.015e+000$	
FIPS	$-1.955e+001 \pm 8.477e+000$	$4.341e+002 \pm 3.499e+001$	$4.155e+001 \pm 1.363e+000$	
DMS-PSO	$-4.572e+001 \pm 1.703e+000$	$2.837e+002 \pm 1.606e+001$	$3.632e+001 \pm 1.225e+000$	
CLPSO	$-4.529e+001 \pm 1.269e+000$	$2.633e+002 \pm 1.196e+001$	$3.496e+001 \pm 1.768e+000$	
AFPSO	$-4.547e+001 \pm 1.608e+000$	$2.663e+002 \pm 1.200e+001$	$3.609e+001 \pm 2.540e+000$	
AFPSO-QI	$-4.678e+001 \pm 1.212e+000$	$2.533e+002 \pm 1.263e+001$	$3.135e+001 \pm 3.301e+000$	
FGLT-PSO	$\mathbf{-4.920e+001 \pm 1.091e+000}$	$\mathbf{1.580e+002 \pm 1.065e+001}$	$\mathbf{9.933e+000} \pm 3.737e+000$	
	F_{14}	F_{17}	F_{18}	
PSO	$1.703e+001 \pm 2.554e+000$	$3.217e+009 \pm 3.880e+009$	$-1.953e+002 \pm 3.282e+001$	
QIPSO	$1.520e+001 \pm 1.319e+000$	$2.347e+009 \pm 1.872e+009$	$-1.963e+002 \pm 2.964e+001$	
FIPS	$2.660e+001 \pm 1.417e+000$	$1.340e+003 \pm 2.044e+003$	$-2.173e+002 \pm 3.076e+001$	
DMS-PSO	$1.292e+001 \pm 1.328e+000$	$3.362e+008 \pm 3.089e+008$	$-2.456e+002 \pm 1.293e+001$	
CLPSO	$1.194e+001 \pm 1.365e+000$	$5.943e+002 \pm 5.069e+001$	$-2.605e+002 \pm 7.359e+000$	
AFPSO	$1.038e+001 \pm 1.379e+000$	$9.700e+007 \pm 1.197e+008$	$-2.718e+002 \pm 1.072e+001$	
AFPSO-QI	$8.462e+000 \pm 9.477e-001$	$8.832e+007 \pm 9.793e+007$	$\mathbf{-2.736e+002} \pm 9.667e+000$	
FGLT-PSO	$\mathbf{5.899e-001 \pm 2.820e-001}$	$\mathbf{5.204e+002} \pm 1.328e+002$	$-2.658e+002 \pm 2.078e+001$	
	F_{19}	F_{20}	Avg. rank	Final rank
PSO	$-1.191e+002 \pm 7.093e-002$	$-1.118e+002 \pm 4.289e+001$	8	8
QIPSO	$-1.191e+002 \pm 5.677e-001$	$-1.152e+002 \pm 3.390e+001$	6.6	7
FIPS	$\mathbf{-1.199e+002 \pm 3.239e-002}$	$-1.437e+002 \pm 5.164e+001$	5.6	6
DMS-PSO	$-1.192e+002 \pm 6.117e-002$	$-1.910e+002 \pm 1.994e+001$	4.4	5
CLPSO	$-1.190e+002 \pm 3.781e-002$	$-1.312e+002 \pm 2.437e+001$	4.3	4
AFPSO	$-1.197e+002 \pm 4.280e-002$	$-1.267e+002 \pm 2.727e+001$	3.9	3
AFPSO-QI	$-1.198e+002 \pm 3.854e-001$	$-1.339e+002 \pm 2.208e+001$	2.4	2
FGLT-PSO	$-1.191e+002 \pm 4.805e-002$	$\mathbf{-2.166e+002 \pm 3.117e+001}$	**1.8**	**1**

Acknowledgments

The authors thank the Research Management Centre (RMC), Universiti Teknologi Malaysia (UTM), for supporting in R & D and *Soft Computing Research Group (SCRG)*, Universiti Teknologi Malaysia (UTM), Johor Bahru, Malaysia, for the inspiration and moral support in conducting this research. Also, they hereby would like to appreciate the postdoctoral program, Universiti Teknologi Malaysia (UTM), for the financial support and research activities. This work is supported by the Ministry of Higher Education (MOHE) under Fundamental Research Grant Scheme FRGS-4F347.

References

[1] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Conference on Neural Networks*, pp. 1942–1948, December 1995.

[2] K. Y. Chan, T. S. Dillon, and C. K. Kwong, "Polynomial modeling for time-varying systems based on a particle swarm optimization algorithm," *Information Sciences*, vol. 181, no. 9, pp. 1623–1640, 2011.

[3] V. Fathi and G. A. Montazer, "An improvement in RBF learning algorithm based on PSO for real time applications," *Neurocomputing*, vol. 111, pp. 169–176, 2013.

[4] H. Huang, H. Qin, Z. Hao, and A. Lim, "Example-based learning particle swarm optimization for continuous optimization," *Information Sciences*, vol. 182, no. 1, pp. 125–138, 2012.

[5] B. Y. Qu, J. J. Liang, and P. N. Suganthan, "Niching particle swarm optimization with local search for multi-modal optimization," *Information Sciences*, vol. 197, pp. 131–143, 2012.

[6] H. Wang, I. Moon, S. Yang, and D. Wang, "A memetic particle swarm optimization algorithm for multimodal optimization problems," *Information Sciences*, vol. 197, pp. 38–52, 2012.

[7] Z. Beheshti and S. M. Shamsuddin, *International Journal of Advances in Soft Computing & Its Applications*, vol. 5, no. 1, pp. 1–35, 2013.

[8] J. Kennedy and R. Mendes, "Population structure and particle swarm performance," in *Proceedings of IEEE international conference on Evolutionary Computation*, vol. 2, pp. 1671–1676, 2002.

[9] J. Kennedy and R. Mendes, "Neighborhood topologies in fully informed and best-of-neighborhood particle swarms," *IEEE Transactions on Systems, Man, and Cybernetics Part C*, vol. 36, pp. 515–519, 2006.

[10] R. Mendes, J. Kennedy, and J. Neves, "The fully informed particle swarm: simpler, maybe better," *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 3, pp. 204–210, 2004.

[11] A. Ratnaweera, S. K. Halgamuge, and H. C. Watson, "Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients," *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 3, pp. 240–255, 2004.

[12] J. J. Liang and P. N. Suganthan, "Dynamic multi-swarm particle swarm optimizer," in *Proceedings of the IEEE Swarm Intelligence Symposium (SIS '05)*, pp. 127–132, June 2005.

[13] J. J. Liang, A. K. Qin, P. N. Suganthan, and S. Baskar, "Comprehensive learning particle swarm optimizer for global optimization of multimodal functions," *IEEE Transactions on Evolutionary Computation*, vol. 10, no. 3, pp. 281–295, 2006.

[14] Z. Beheshti, S. M. H. Shamsuddin, and S. Hasan, "MPSO: median-oriented particle swarm optimization," *Applied Mathematics and Computation*, vol. 219, no. 11, pp. 5817–5836, 2013.

[15] Z. Beheshti and S. M. Hj. Shamsuddin, "CAPSO: centripetal accelerated particle swarm optimization," *Information Sciences*, vol. 258, pp. 54–79, 2014.

[16] M. Pant, T. Radha, and V. P. Singh, "A new particle swarm optimization with quadratic interpolation," in *Proceedings of the International Conference on Computational Intelligence and Multimedia Applications (ICCIMA '07)*, pp. 55–60, December 2007.

[17] P. Liu, W. Leng, and W. Fang, "Training ANFIS model with an improved quantum-behaved particle swarm optimization algorithm," *Mathematical Problems in Engineering*, vol. 2013, Article ID 595639, 10 pages, 2013.

[18] Z.-H. Zhan, J. Zhang, Y. Li, and H. S.-H. Chung, "Adaptive particle swarm optimization," *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, vol. 39, no. 6, pp. 1362–1381, 2009.

[19] Y.-T. Juang, S.-L. Tung, and H.-C. Chiu, "Adaptive fuzzy particle swarm optimization for global optimization of multimodal functions," *Information Sciences*, vol. 181, no. 20, pp. 4539–4549, 2011.

[20] Z. Beheshti, S. M. Shamsuddin, E. Beheshti, and S. S. Yuhaniz, "Enhancement of artificial neural network learning using centripetal accelerated particle swarm optimization for medical diseases diagnosis," *Soft Computing*, 2013.

[21] Z. Beheshti, S. M. Shamsuddin, and S. S. Yuhaniz, "Binary accelerated particle swarm algorithm (BAPSA) for discrete optimization problems," *Journal of Global Optimization*, vol. 57, no. 2, pp. 549–573, 2013.

[22] M. A. Cavuslu, C. Karakuzu, and F. Karakaya, "Neural identification of dynamic systems on FPGA with improved PSO learning," *Applied Soft Computing Journal*, vol. 12, no. 9, pp. 2707–2718, 2012.

[23] L. Liu, S. Yang, and D. Wang, "Force-imitated particle swarm optimization using the near-neighbor effect for locating multiple optima," *Information Sciences*, vol. 182, no. 1, pp. 139–155, 2012.

[24] Z. Beheshti and S. M. Shamsuddin, *Centripetal accelerated particle swarm optimization and its applications in machine learning [Ph.D. thesis]*, Universiti Teknologi Malaysia, 2013.

[25] Y. Wang, J. Zhou, C. Zhou, Y. Wang, H. Qin, and Y. Lu, "An improved self-adaptive PSO technique for short-term hydrothermal scheduling," *Expert Systems with Applications*, vol. 39, no. 3, pp. 2288–2295, 2012.

[26] Q. Luo and D. Yi, "A co-evolving framework for robust particle swarm optimization," *Applied Mathematics and Computation*, vol. 199, no. 2, pp. 611–622, 2008.

[27] P. S. Andrews, "An investigation into mutation operators for particle swarm optimization," in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '06)*, pp. 1044–1051, July 2006.

[28] M. Lovbjerg, T. K. Rasmussen, and T. Krink, "Hybrid particle swarm optimizer with breeding and subpopulations," in *Proceedings of the 3rd Genetic and Evolutionary Computation Conference*, pp. 469–476, 2001.

[29] S. Tsafarakis, C. Saridakis, G. Baltas, and N. Matsatsinis, "Hybrid particle swarm optimization with mutation for optimizing industrial product lines: an application to a mixed solution space considering both discrete and continuous design variables," *Industrial Marketing Management*, vol. 42, no. 4, pp. 496–506, 2013.

[30] P. J. Angeline, "Using selection to improve particle swarm optimization," in *Proceedings of the IEEE International Conference on Evolutionary Computation (ICEC '98)*, pp. 84–89, May 1998.

[31] Y.-P. Chen, W.-C. Peng, and M.-C. Jian, "Particle swarm optimization with recombination and dynamic linkage discovery," *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, vol. 37, no. 6, pp. 1460–1470, 2007.

[32] Z.-H. Zhan, J. Zhang, Y. Li, and Y.-H. Shi, "Orthogonal learning particle swarm optimization," *IEEE Transactions on Evolutionary Computation*, vol. 15, no. 6, pp. 832–847, 2011.

[33] W.-F. Gao, S.-Y. Liu, and L.-L. Huang, "Particle swarm optimization with chaotic opposition-based population initialization and stochastic search technique," *Communications in Nonlinear Science and Numerical Simulation*, vol. 17, no. 11, pp. 4316–4327, 2012.

[34] K. Tang, X. Yao, P. N. Suganthan et al., "Benchmark functions for the CEC'2008 special session and competition on large scale global optimization," Tech. Rep., Nature Inspired Computation and Applications Laboratory, USTC, Anhui, China, http://nical.ustc.edu.cn/cec08ss.php.

[35] X. Yao, Y. Liu, and G. Lin, "Evolutionary programming made faster," *IEEE Transactions on Evolutionary Computation*, vol. 3, no. 2, pp. 82–102, 1999.

[36] R. Salomon, "Re-evaluating genetic algorithm performance under coordinate rotation of benchmark functions. A survey of some theoretical and practical aspects of genetic algorithms," *BioSystems*, vol. 39, no. 3, pp. 263–278, 1996.

[37] F. Wilcoxon, "Individual comparisons by ranking methods," *Biometrics Bulletin*, vol. 1, no. 6, pp. 80–83, 1945.