

## Research Article

# A Hybrid Particle Swarm Optimizer for Curriculum Sequencing Problem

Xianjie Peng,<sup>1</sup> Xiaonan Sun,<sup>2</sup> and Zhen He<sup>3</sup> 

<sup>1</sup>Department of Human Resources, Tongji University, Shanghai, China

<sup>2</sup>Institute of Vocational Education, Tongji University, Shanghai, China

<sup>3</sup>Faculty of Education, Beijing Normal University, Beijing, China

Correspondence should be addressed to Zhen He; zhenhemail163@163.com

Received 20 March 2022; Accepted 11 May 2022; Published 6 June 2022

Academic Editor: Shi Cheng

Copyright © 2022 Xianjie Peng et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Curriculum sequencing problem is crucial to e-learning system, which is a NP-hard optimization problem and commonly solved by swarm intelligence. As a form of swarm intelligence, particle swarm optimization (PSO) is widely used in various kinds of optimization problems. However, PSO is found ineffective in complex optimization problems. The main reason is that PSO is ineffective in diversity preservation, leading to high risks to be trapped by the local optima. To solve this problem, a novel hybrid PSO algorithm is proposed in this study. First, a competitive-genetic crossover strategy is proposed for PSO to balance the convergence and diversity. Second, an adaptive polynomial mutation is introduced in PSO to further improve its diversity preservation ability. Furthermore, a curriculum scheduling model is proposed, where several constraints are taken into considerations to ensure the practicability of the curriculum sequencing. The numerical comparison experiments show that the proposed algorithm is effective in solving function optimization in comparison to several popular PSO variants; furthermore, for the optimization of the designed curriculum sequencing problem, the proposed algorithm shows significant advantages over the compared algorithms with respect to the degree of the satisfaction of the objectives, i.e., 20, 14, and 5 percentages higher, respectively.

## 1. Introduction

With the rapid developments of communication technologies, e-learning is growing as an important teaching method. However, the traditional e-learning systems commonly ignore reusability and are not adaptive [1]. Furthermore, the traditional e-learning systems are inefficient in flexibility and cannot provide either the adaptive learning or the fixed curriculum path for students [2, 3]. Due to its characteristic of NP hard, individualizing paths for each student are costly to providers, since teachers typically perform the sorting process manually. Therefore, it is of great value to automate the curriculum sequencing to reduce the cost of high-quality e-learning. To solve this issue, many kinds of models and optimization methods are carried out for curriculum sequencing problems [4–11], where particle

swarm optimization (PSO) [12] is widely adopted as the optimizer.

As a form of swarm intelligence, PSO builds on the population-based searching technique, where each particle (A feasible solution in the swarm) includes two attributes, i.e., the position and the velocity. The former represents the information of decision variables of the particles and the latter is the evolutionary direction of the particles. Based on equations (1) and (2), the swarm is able to keep updating and traverse the decision space to find better solutions:

$$v_i^d(t+1) = v_i^d(t) + r_1c_1(p_i^d(t) - gbest^d) + r_2c_2(p_i^d(t) - pbest_i^d), \quad (1)$$

$$p_i^d(t+1) = p_i^d(t) + v_i^d(t+1), \quad (2)$$

where  $v_i^d(t)$  is the  $d$ th dimension of the position of the  $i$ th particle in generation  $t$ ,  $gbest$  is the best particle of the current swarm and  $pbest_i$  is the best position searched by the  $i$ th particle so far,  $v_i$  is the velocity of the  $i$ th particle, and  $c_1$  and  $c_2$  are the coefficients, while  $r_1$  and  $r_2$  are two random number generated within  $(0, 1)$ . As one can see, PSO is of high simplicity.

However, PSO is found ineffective in complex problems, such as the multimodal and large-scale optimization problems [13, 14]. The main reason is that PSO lacks global searching ability and is prone to be trapped by the local optimum due to its poor ability in balancing the exploration and exploitation.

To this end, a huge amount of effort has been carried out, mainly being classified into three categories: the improvements on parameters [15–19], different kinds of topologies [20–22], and hybridizations with other kinds of techniques and algorithms [23, 24]. Nevertheless, the existing algorithms still cannot obtain the global optima when solving complex problems. The main reason is that the current PSO and its variants are still inefficient in diversity preservation. For instance, CSO tries to enhance its diversity preservation ability by guiding the updated particles with the mean position of the whole swarm. However, the mean position is shared by all the updated particles, resulting in risks in premature convergence.

To improve the diversity preservation ability of PSO, this study proposes a novel hybrid particle swarm optimization algorithm; the main contributions of this study are listed as follows:

- (1) A competitive-genetic crossover operation is proposed, which combines the competitive mechanism, the genetic crossover, and the elite strategy to improve the diversity of the exemplars and ensure a comprehensive exploitation ability of PSO
- (2) An adaptive polynomial mutation is introduced in PSO to further enhance its diversity preservation ability, leading to the proposed hybrid particle swarm optimization algorithm
- (3) A curriculum sequencing model is proposed and the proposed algorithm is applied into solving the curriculum sequencing problem to improve the efficiency of the e-learning systems

The rest of this study is organized as follows. Section 2 presents the literature review for the mainstream of the improvements on PSO. Section 3 provides a detailed description of the proposed algorithm. The experiments and the case study are conducted in Section 4, and we conclude this study in Section 5.

## 2. Related Work

As mentioned above, the current works for improving PSO can be mainly categorized into the following three classes.

*2.1. Control of the Parameters.* Different parameters in PSO are of different functions: the parameter in inertia

component mainly focuses on keeping the particles move according to the historical information for global optimization; the parameters in the social components are related to the convergence. Shi and Eberhart propose a linear control method to dynamically adjust the inertia weight [25]. The main goal of their work is to enhance the diversity preservation ability for PSO in the early optimization stage and emphasize the convergence ability for PSO in the later stage. Furthermore, they propose a fuzzy adjustment strategy for the inertia weight [26]. Ratnaweera et al. propose HPSO-TVAC to dynamically adjust the acceleration coefficients during the run [16]. Zhan et al. propose the adaptive particle swarm optimization algorithm, changing the acceleration coefficients according to a predefined evolutionary state estimation strategy [18]. Piotrowski et al. put forward a study to investigate the influences of the swarm size on the performance of PSO [27]. Tian and Shi propose a PSO variant, where a logistic map and a sigmoid-like inertia weight are utilized to initialize the swarm and balance the exploration and exploitation, respectively [28].

*2.2. Modifications on the Topologies.* The main idea of the methods in this category is designing new information exchanging topologies to improve the diversity preservation ability to enhance the search ability of PSO. Mendes et al. investigate a fully informed topology, where particles learn from their neighbors instead of the global best particle in the swarm [20]. Liang et al. propose comprehensive learning strategy which allows each particle's historical personal best position to be the exemplar for others [21]. Chen et al. propose ALCPSO, which dynamically replaces the global best particle with another individual according to several predefined rules to guide the updated particles [29]. Cheng et al. propose FBE, where the whole swarm is divided into two subswarms, and a competitive mechanism is designed to select exemplars for particles [30]. The weak particles in the competition learn from the best particle in the internal subswarm and randomly selected particles from the external subswarm, while a mutation operation is executed on the strong particles. Furthermore, Cheng and Jin propose the competitive swarm optimizer (CSO), which only updates a half of the particles and adopts the strong particles in the competition to guide the weak particles [22]. Inspired by the social learning actions, Cheng and Jin propose SLPSO, which allows particles to learn from all the particles that are better than themselves [31]. Yang et al. propose DLLSO [32], which divides particles into different levels based on their fitness value. Afterwards, each particle chooses two different exemplars from superior levels. Zhang et al. propose a modified PSO by introducing a dynamic neighborhood-based learning strategy to enhance the diversity preservation ability for PSO [33]. Zeng et al. propose DNSPSO, where a novel velocity updating mechanism is designed to adjust the personal best position and the global best position based on a designed distance-based dynamic neighborhood to enhance the information sharing among the particles [34].

**2.3. Hybridization with Other Techniques.** Hybridizing PSO with other techniques is able to make use of the advantages of both PSO and other techniques. Van et al. propose CCPSO-SK and CCPSO-HK by integrating the co-operative co-evolutionary framework and PSO to solve the large-scale optimization [35, 36]. Li and Yao put forward CCPSO2, where the Gaussian and Cauchy mutation are proposed to update individuals to balance the diversity and convergence [37]. Qin et al. propose a PSO variant by dividing the whole swarm into learned and learning subswarms at each generation; afterwards, the learning subswarm will learn from the learned subswarm with respect to a random probability [38]. Li et al. propose a hybrid algorithm by incorporating the update mechanism of PSO into biogeographic optimization algorithm to improve the exploration ability of the algorithm [12]. The genetic learning particle swarm optimization put forward by Gong et al. adopts crossover and mutation operators to enhance the exploration ability of PSO [39]. Similarly, Chen et al. design two kinds of crossover operations to breed promising exemplars [23]. Chen et al. propose HPSOSSM, which uses a logistic map sequence to enhance the diversity and adopts a spiral-shaped mechanism to improve the convergence speed [40].

### 3. Proposed Algorithm

As discussed above, the main issue in PSO is that PSO is ineffective in diversity preservation. Targeting against this issue, this study puts forward two improvements for PSO: a competitive-genetic crossover operator and an adaptive mutation operator. The details are presented as follows.

**3.1. Competitive-Genetic Crossover Operator.** Competitive mechanism-based learning strategy proposed in [22] has been demonstrated to be effective in both convergence and diversity preservation. The reason is that it not only can be used to select exemplars for particles for convergence but also is beneficial to the diversity preservation by only updating half of the particles. On the contrary, genetic algorithm (GA) [41] potentially has better convergence compared with PSO, since the promising information in the population of GA can be directly copied to other solutions and kept to the offsprings, while PSO updates its particles according to stochastic strategies. Motivated by this, a competitive-genetic crossover operator (CGCO) is proposed to conduct convergence and improve the diversity preservation ability for PSO.

First, the competitive mechanism is conducted to determine the strong particles' set (the winners in the competition) and the weak particles' set (the losers in the competition) at each generation, which can be shown in Figure 1. To be specific, (i) the whole swarm is randomly divided into  $N/2$  subswarms, i.e., each subswarm includes two particles, (ii) the particles in each subswarm conduct the competition according to their fitness to determine the strong particle and the weak particle, i.e., the winner and loser in the competition, respectively.

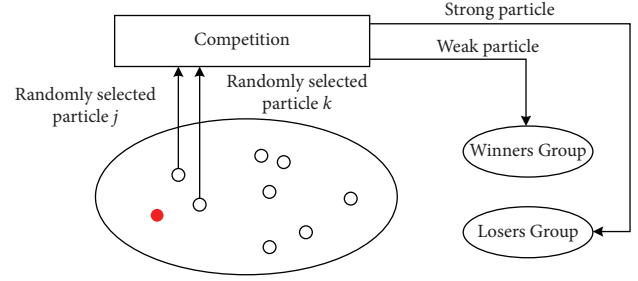


FIGURE 1: The competition mechanism.

Second, a multipoint crossover is conducted on the paired strong particles and the corresponding weak particles. Different from the mechanism in GA, only the strong particles' information will be copied into the weak particles in the proposed crossover operator. On the contrary, pbest of the particles are further adopted in the crossover operation. The crossover operation for a particle  $p_{l,i}$  can be formulated as

$$\begin{aligned} p_{l,i}[: ] &= \text{Concatenat}(p_{l,i}[0: \text{pos}_1], p_{w,i}[\text{pos}_1: \text{pos}_2], p_{l,i}[\text{pos}_2: ]), \\ p_{l,i}[: ] &= \text{Concatenat}(p_{l,i}[0: \text{pos}_3], \text{pbest}_i[\text{pos}_3: \text{pos}_4], p_{l,i}[\text{pos}_4: ]), \end{aligned} \quad (3)$$

where  $p_{l,i}[: ]$  is the position of the  $i$ th particle in the losers group,  $p_{w,i}$  is the corresponding winner and pbest is the historical personal best position of  $p_{l,i}$ , and  $\text{pos}_1$ ,  $\text{pos}_2$ ,  $\text{pos}_3$ , and  $\text{pos}_4$  are four randomly generated indexes for crossover.

Finally, an elite strategy is adopted: an offspring is kept to the next generation if it has improvements in comparison to its parent (the corresponding losers) or the offspring will be kept if a generated random number is less than 0.5.

Together, the pseudocode of CGCO is shown in Algorithm 1.

**3.2. Adaptive Mutation Operator.** Although CGCO is beneficial to diversity by only updating half of the particles, it only exchanges existing information between the particles. Therefore, there still exist high risks that the swarm will be trapped by the local optima. To solve this problem, an adaptive mutation operator (AMO) is proposed.

First, each particle is initialized with a counter in the swarm initialization stage.

Second, in the optimization stage, the mutation rate for the  $i$ th particle is computed according to

$$Mu_r(i) = Mu_{\max} \frac{\text{counter}(i)}{g_n}, \quad (4)$$

where  $\text{counter}(i)$  records how many times that the  $i$ th particle successfully has been updated, while  $g_n$  is the cost number of the generation and  $Mu_{\max}$  is the predefined maximum of the mutation rate.

With  $Mu_r$ , each particle conducts the polynomial mutation to improve the diversity of the swarm. One can find that the better a particle, the smaller the corresponding mutation rate. This is reasonable since the information of the

```

Input: swarm  $P(t)$ , fitness value vector fitness, and swarm size  $N$ 
Output: The offsprings  $P_{\text{off}}$ 
(1)  $P_{\text{off}} \leftarrow \emptyset$ 
(2)  $P_{\text{winners}}$  and  $P_{\text{losers}} \leftarrow$  Conduct the competition mechanism
(3) for  $i = 1$  to  $N/2$  do
(4)  $p_{l,i}, p_{w,i} \leftarrow$  Select the  $i$ th particle and the corresponding winner in  $P_{\text{losers}}$  and  $P_{\text{winners}}$ 
(5)  $p_1 \leftarrow$  Conduct the crossover operation between  $p_{l,i}$  and  $p_{w,i}$ 
(6) if  $p_{\text{off}}$  is better than  $p_{l,i}$  then
(7)    $P_{\text{off}} \leftarrow P_{\text{off}} \cup p_{\text{off}}$ 
(8) else
(9)   if rand  $\leq 0.5$  then
(10)     $P_{\text{off}} \leftarrow P_{\text{off}} \cup p_{l,i}$ 
(11)   else
(12)     $P_{\text{off}} \leftarrow P_{\text{off}} \cup p_{\text{off}}$ 
(13)   end
(14) end
(15) end

```

ALGORITHM 1: Pseudocode of CGCO.

promising particles should be kept with a large likelihood for convergence.

3.3. *HPSO-GA*. To sum up, a competitive-genetic crossover operator and an adaptive mutation strategy are proposed to conduct convergence and diversity preservation for PSO. By integrating these two strategies together, a novel hybrid PSO is proposed, which is refer to HPSO-GA in the following for simplicity. The procedure of HPSO-GA can be found in Figure 2.

## 4. Experiments and Results

In this section, numerical comparisons are first conducted to test the performance of the proposed HPSO-GA; second, a curriculum sequencing model is put forward. Afterwards, HPSO-GA and other three algorithms are applied into solving the proposed curriculum sequencing problem to test the reliability of the proposed algorithm.

### 4.1. Numerical Comparisons

4.1.1. *Compared Algorithms*. To test the numerical optimization performance of HPSO-GA, five popular PSO variants are selected in the comparisons, including ALCPSO [29], LIPS [20], CSO [22], HPSO-TVAC [16], and DLLSO [32]; the benchmarks posted by CEC 2013 is adopted to test the algorithms.

4.1.2. *Experimental Settings*. For a fair comparison, all the compared algorithms adopt the suggested parameter settings in the corresponding references; the swarm size for the compared algorithms and the dimensionality of the benchmarks are set to 100; the maximum number of the fitness evaluations is adopted as the termination criterion, which is set to  $5E + 05$ . The parameter settings for HPSO-

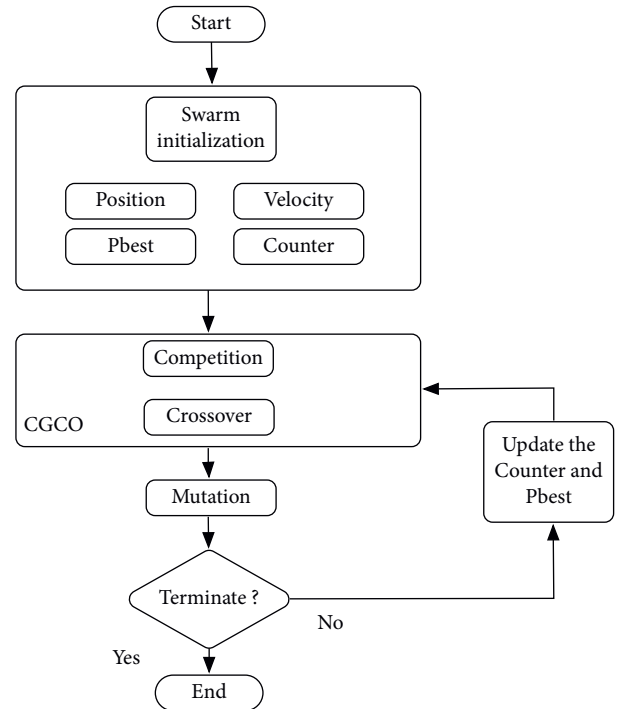


FIGURE 2: The flowchart of the proposed algorithm.

GA are set as follows: the crossover rate is set to 1 and the maximum mutation rate is set to 0.05; each algorithm is run 31 times on each benchmark. The Wilcoxon rank sum test is adopted for the statistical analysis between the peer algorithms and HPSO-GA, where the significance level is set to 0.05.

4.1.3. *Results*. Table 1 shows the results of the comparison, where the mean of the corresponding results are recorded. The symbols “+,” “-,” and “=” in the bottom of the table

TABLE 1: The comparison on CEC 2013 benchmarks ( $D = 100$ ).

Functions	Property	ALCPSO	CSO	HPSO-TVAC	LIPS	DLLSO	HPSO-GA
$F_1$	Mean	7.47E-12	2.28E-13	2.72E-08	8.71E-01	6.37E-13	2.20E-13
	p-value	<b>5.25E-10</b> <sup>+</sup>	<b>5.00E-03</b> <sup>+</sup>	<b>5.35E-10</b> <sup>+</sup>	<b>5.35E-10</b> <sup>+</sup>	<b>1.38E-08</b> <sup>+</sup>	-
$F_2$	Mean	1.32E+08	3.17E+06	2.65E+07	4.09E+08	1.76E+07	3.05E+06
	p-value	<b>1.42E-09</b> <sup>+</sup>	5.61E-01 <sup>=</sup>	<b>1.42E-09</b> <sup>+</sup>	<b>1.42E-09</b> <sup>+</sup>	<b>2.56E-09</b> <sup>+</sup>	-
$F_3$	Mean	4.64E+10	1.33E+08	2.16E+10	3.24E+13	9.15E+09	3.27E+08
	p-value	<b>1.41E-09</b> <sup>+</sup>	6.75E-06-	<b>1.41E-09</b> <sup>+</sup>	<b>1.41E-09</b> <sup>+</sup>	<b>1.42E-09</b> <sup>+</sup>	-
$F_4$	Mean	5.20E+04	8.76E+04	6.66E+04	2.52E+05	7.35E+04	2.38E+04
	p-value	<b>2.02E-09</b> <sup>+</sup>	<b>1.41E-09</b> <sup>+</sup>	<b>1.41E-09</b> <sup>+</sup>	<b>1.41E-09</b> <sup>+</sup>	<b>1.60E-09</b> <sup>+</sup>	-
$F_5$	Mean	3.02E-12	1.14E-13	2.77E-04	1.45E+03	1.03E-05	3.22E-13
	p-value	<b>7.18E-10</b> <sup>+</sup>	4.16E-11-	<b>7.40E-10</b> <sup>+</sup>	<b>7.40E-10</b> <sup>+</sup>	<b>7.25E-10</b> <sup>+</sup>	-
$F_6$	Mean	6.83E+02	2.00E+02	2.40E+02	2.95E+03	2.73E+02	1.97E+02
	p-value	<b>1.42E-09</b> <sup>+</sup>	<b>1.41E-09</b> <sup>+</sup>	<b>1.42E-09</b> <sup>+</sup>	<b>1.42E-09</b> <sup>+</sup>	<b>1.42E-09</b> <sup>+</sup>	-
$F_7$	Mean	1.80E+02	1.10E+01	5.17E+04	2.66E+03	6.90E+01	2.00E+00
	p-value	<b>1.42E-09</b> <sup>+</sup>	<b>1.41E-09</b> <sup>+</sup>	<b>1.42E-09</b> <sup>+</sup>	<b>1.42E-09</b> <sup>+</sup>	<b>1.60E-09</b> <sup>+</sup>	-
$F_8$	Mean	2.20E+01	2.20E+01	2.20E+01	2.20E+01	2.10E+01	2.05E+01
	p-value	8.77E-01 <sup>+</sup>	<b>5.01E-05</b> <sup>+</sup>	<b>1.30E-02</b> <sup>+</sup>	<b>1.17E-02</b> <sup>+</sup>	5.35E-01 <sup>=</sup>	-
$F_9$	Mean	1.26E+02	4.20E+01	1.37E+02	1.18E+02	5.90E+01	2.56E+01
	p-value	<b>1.42E-09</b> <sup>+</sup>	<b>1.60E-03</b> <sup>+</sup>	<b>1.42E-09</b> <sup>+</sup>	<b>1.42E-09</b> <sup>+</sup>	<b>2.21E-07</b> <sup>+</sup>	-
$F_{10}$	Mean	1.20E+01	1.64E-01	2.60E+01	1.37E+03	2.50E+01	1.21E-01
	p-value	<b>1.42E-09</b> <sup>+</sup>	4.15E-01 <sup>=</sup>	<b>1.42E-09</b> <sup>+</sup>	<b>1.42E-09</b> <sup>+</sup>	<b>1.42E-09</b> <sup>+</sup>	-
$F_{11}$	Mean	2.57E+02	5.20E+01	3.22E+02	7.45E+02	1.18E+02	4.40E+01
	p-value	<b>1.41E-09</b> <sup>+</sup>	2.80E-06 <sup>+</sup>	<b>1.41E-09</b> <sup>+</sup>	<b>1.41E-09</b> <sup>+</sup>	<b>8.28E-09</b> <sup>+</sup>	-
$F_{12}$	Mean	6.70E+02	8.04E+02	1.72E+03	7.99E+02	1.40E+02	7.99E+02
	p-value	3.31E-04-	<b>9.90E-01</b> <sup>=</sup>	<b>1.41E-09</b> <sup>+</sup>	7.415E-01 <sup>=</sup>	1.41E-09-	-
$F_{13}$	Mean	1.05E+03	8.01E+02	2.11E+03	1.30E+03	3.75E+02	7.95E+02
	p-value	<b>7.38E-09</b> <sup>+</sup>	<b>3.00E-03</b> <sup>+</sup>	<b>1.42E-09</b> <sup>+</sup>	<b>1.42E-09</b> <sup>+</sup>	1.42E-09-	-
$F_{14}$	Mean	6.58E+03	1.45E+03	3.77E+03	1.11E+04	5.19E+03	2.33E+03
	p-value	<b>1.42E-09</b> <sup>+</sup>	4.67E-06-	<b>1.31E-07</b> <sup>+</sup>	<b>1.42E-09</b> <sup>+</sup>	<b>2.03E-09</b> <sup>+</sup>	-
$F_{15}$	Mean	2.39E+04	2.92E+04	2.11E+04	1.41E+04	9.30E+03	2.11E+04
	p-value	1.89E-02 <sup>+</sup>	<b>4.78E-02</b> <sup>+</sup>	1.60E-09 <sup>-</sup>	1.42E-09 <sup>-</sup>	1.41E-09 <sup>-</sup>	-
$F_{16}$	Mean	3.00E+00	5.00E+00	3.00E+00	1.00E+00	2.00E+00	2.00E+00
	p-value	5.50E-03 <sup>+</sup>	<b>1.41E-09</b> <sup>+</sup>	4.54E-07 <sup>-</sup>	1.04E-08 <sup>-</sup>	3.07E-04 <sup>-</sup>	-
$F_{17}$	Mean	5.50E+02	7.80E+02	8.40E+02	1.39E+03	1.98E+02	4.94E+02
	p-value	<b>2.80E-03</b> <sup>+</sup>	<b>1.41E-09</b> <sup>+</sup>	<b>1.80E-09</b> <sup>+</sup>	<b>1.42E-09</b> <sup>+</sup>	1.42E-09 <sup>-</sup>	-
$F_{18}$	Mean	1.20E+03	9.10E+02	2.53E+03	1.51E+03	3.09E+02	8.98E+02
	p-value	<b>1.99E-07</b> <sup>+</sup>	<b>2.50E-03</b> <sup>+</sup>	<b>1.42E-09</b> <sup>+</sup>	<b>1.42E-09</b> <sup>+</sup>	1.60E-09 <sup>-</sup>	-
$F_{19}$	Mean	4.90E+01	1.10E+01	8.20E+01	5.70E+02	1.80E+01	1.00E+01
	p-value	<b>1.41E-09</b> <sup>+</sup>	5.35E-01 <sup>=</sup>	<b>1.41E-09</b> <sup>+</sup>	<b>1.41E-09</b> <sup>+</sup>	<b>5.21E-09</b> <sup>+</sup>	-
$F_{20}$	Mean	5.00E+01	4.99E+01	4.99E+01	4.99E+01	5.00E+01	4.75E+01
	p-value	<b>1.04E-02</b> <sup>+</sup>	<b>4.28E-02</b> <sup>+</sup>	1.18E-01 <sup>=</sup>	3.97E-01 <sup>=</sup>	<b>1.04E-02</b> <sup>+</sup>	-
$F_{21}$	Mean	3.30E+02	3.70E+02	4.20E+02	4.40E+02	4.10E+02	3.90E+02
	p-value	5.86E-01 <sup>=</sup>	3.17E-01 <sup>=</sup>	<b>5.17E-08</b> <sup>+</sup>	<b>4.39E-09</b> <sup>+</sup>	<b>5.12E-07</b> <sup>+</sup>	-
$F_{22}$	Mean	6.97E+03	1.40E+03	5.05E+03	1.56E+04	5.48E+03	2.18E+03
	p-value	<b>1.42E-09</b> <sup>+</sup>	8.86E-06 <sup>-</sup>	<b>1.42E-09</b> <sup>+</sup>	<b>1.42E-09</b> <sup>+</sup>	<b>1.42E-09</b> <sup>+</sup>	-
$F_{23}$	Mean	2.38E+04	2.88E+04	2.57E+04	2.01E+04	1.10E+04	2.20E+04
	p-value	7.86E-01 <sup>=</sup>	1.04E-01 <sup>=</sup>	6.55E-01 <sup>=</sup>	2.98E-02 <sup>-</sup>	2.98E-02 <sup>-</sup>	-
$F_{24}$	Mean	5.50E+02	3.00E+02	6.10E+02	5.80E+02	3.70E+02	2.05E+02
	p-value	<b>1.42E-09</b> <sup>+</sup>	<b>2.57E-09</b> <sup>+</sup>	<b>1.42E-09</b> <sup>+</sup>	<b>1.42E-09</b> <sup>+</sup>	<b>1.80E-09</b> <sup>+</sup>	-
$F_{25}$	Mean	6.30E+02	4.10E+02	6.00E+02	7.10E+02	4.60E+02	3.43E+02
	p-value	<b>1.42E-09</b> <sup>+</sup>	<b>2.87E-08</b> <sup>+</sup>	<b>1.42E-09</b> <sup>+</sup>	<b>1.42E-09</b> <sup>+</sup>	<b>1.42E-09</b> <sup>+</sup>	-
$F_{26}$	Mean	6.20E+02	3.90E+02	6.80E+02	5.40E+02	4.40E+02	2.10E+02
	p-value	<b>1.42E-09</b> <sup>+</sup>	<b>1.60E-09</b> <sup>+</sup>	<b>1.42E-09</b> <sup>+</sup>	<b>5.91E-05</b> <sup>+</sup>	<b>1.42E-09</b> <sup>+</sup>	-
$F_{27}$	Mean	3.56E+03	1.33E+03	4.45E+03	3.75E+03	1.99E+03	7.80E+02
	p-value	<b>1.42E-09</b> <sup>+</sup>	<b>6.80E-07</b> <sup>+</sup>	<b>1.42E-09</b> <sup>+</sup>	<b>1.42E-09</b> <sup>+</sup>	<b>2.87E-08</b> <sup>+</sup>	-
$F_{28}$	Mean	4.72E+03	2.89E+03	1.76E+04	8.90E+03	3.38E+03	1.54E+03
	p-value	<b>1.99E-07</b> <sup>+</sup>	<b>8.87E-05</b> <sup>+</sup>	<b>1.41E-09</b> <sup>+</sup>	<b>1.41E-09</b> <sup>+</sup>	<b>4.67E-06</b> <sup>+</sup>	-
	w/l/t	25/1/2	18/4/6	24/2/2	23/3/2	20/7/1	-

indicate that the results of HPSO-GA are significantly better, significantly worse, or statistically similar to the results obtained by the corresponding peer algorithms, respectively. The best results of the average performance are highlighted by gray.

As shown by the results, HPSO-GA wins 16 times out of the 28 benchmarks with respect to the mean results. With a deeper insight, for the five unimodal functions  $F_1$  to  $F_5$ , HPSO-GA wins for 3 times; for the basic multimodal functions  $F_6$  to  $F_{20}$ , HPSO-GA wins for 8 times; for the composition functions  $F_{21}$  to  $F_{28}$ , HPSO-GA wins for 5 times. The statistic analysis results also indicate the competitive performance of HPSO-GA: it significantly outperforms the peer algorithms for 25, 18, 23, 24, and 20 times, respectively.

Therefore, one can see that the proposed algorithm is competitive in large-scale optimization (functions with dimensionality of 100). This can be explained by (i) the proposed CGCO is more suitable for convergence than PSO's position update strategy, since the information of the promising solutions can be directly fed into the updated particles; (ii) CGCO is able to enhance the diversity preservation ability for the proposed algorithm by diverse the exemplars; (iii) the proposed CGCO only updates half of the particles at each generation, which also results in a good diversity preservation ability; (iv) the proposed adaptive mutation operator can effectively further improve the diversity preservation ability for the proposed algorithm.

**4.2. Curriculum Sequencing Optimization.** In this section, a curriculum sequencing model is proposed. Afterwards, the proposed HPSO-GA and three algorithms are tested on the proposed model.

**4.2.1. Problem Description.** Traditional e-learning systems are neither reusable nor adaptive [3]. Furthermore, current e-learning systems commonly fix the learning paths for each student and are ineffective in providing adaptive learning schemes. A main reason is that the curriculum sequencing problem is a NP-hard optimization problem, resulting in difficulties to e-learning provider for properly scheduling the courses for students. Targeting against this issue, various kinds of curriculum sequencing models and optimization algorithms have been proposed [42, 43].

The curriculum sequencing problem can be commonly formulated by  $(L, DV, CT)$ , where  $L = l_1, l_2, \dots, l_n$  represents the students with three properties, for instance, including the student's available time  $l_i(t)$ , student's ability  $l_i(a)$ , and student's objectives  $l_i(O_k)$ ;  $DV$  is a mapping function which maps a student  $l_i$  to a finite ordered set of learning objectives (LOs) that can be assigned to  $l_i$ .  $CT$  is the constraints. The goal of the curriculum sequencing is that scheduling courses for students and satisfying all the constraints as much as possible.

In this study, we build the curriculum sequencing model by taking the following constraints into considerations.

- (1) Various learning objectives for students
- (2) International cooperation teaching
- (3) Two-side satisfaction, e.g., both the requirements of teachers and students should be satisfied
- (4) Suitable strength, e.g., students should be assigned with courses that not exceed their ability level

Following the above concerns, the proposed model is constructed as equations (5)–(9):

$$C_{\text{time},1} = \frac{\sum_i^{LN} \text{sign}(A_{\text{Time}} | \sum l_{i,\text{Time}})}{LN}, \quad (5)$$

$$C_{\text{time},2} = \frac{\sum_i^{LN} \prod_k^{CN} \text{sign}(A_{\text{Time},k} | \sum l_{i,\text{Time}})}{LN}, \quad (6)$$

$$C_{\text{stun}} = \frac{\sum_i^{CN} \text{sign}(A_{\text{course},i} | \sum \text{stu}_{\text{num}})}{CN}, \quad (7)$$

$$C_{\text{diff}} = \frac{\sum_i^{LN} \prod_k^{CN} \text{sign}(A_{\text{course},k,\text{diff}} | l_{i,\text{diff}})}{LN}, \quad (8)$$

$$C_O = \frac{\sum_i^{LN} \text{sign}(l_{i,o} | \sum_k^{CN} A_{\text{course},k,o})}{LN}, \quad (9)$$

$$CS(\%) = C_{\text{time},1} + C_{\text{time},2} + C_{\text{stun}} + C_{\text{diff}} + C_O, \quad (10)$$

where  $C_{\text{time},1}$  and  $C_{\text{time},2}$  are the time constraints for students, which are used to examine whether the assigned total courses' time and each course fits the students' available time,  $C_{\text{stun}}$  is the maximum number of students that a course can fit,  $C_{\text{diff}}$  means whether the courses assigned to a student can fit the difficulty level that the student is in,  $C_O$  means whether the courses assigned to a student can meet his/her learning goals,  $CN$  and  $LN$  are the course number assigned to the corresponding student and the number of all the students, respectively,  $A_{\text{Time}}$  and  $A_{\text{time},k}$  are the total time of the courses and the time of the  $k$ th course,  $A_{\text{course},i}$  is the maximum number of the students that the  $i$ th course can fit,  $\text{stu}_{\text{num}}$  is the corresponding number of students assigned to the  $i$ th course,  $l_{i,o}$  is the learning objectives of the  $i$ th student while  $A_{\text{course},k,o}$  is the objective involved in the  $k$ th course,  $\text{sign}(\text{condition}_1 | \text{condition}_2)$  means whether the two conditions are fit to each other, if it is, it returns 1 or it returns 0; finally, the goal of the curriculum sequencing problem is to maximize the sum of the satisfaction degree (%) of the five constraints.

With the above designs, the proposed model can ensure that (i) the requirements of the students can be maximized and (ii) the time availability of the different students (e.g., the students live in different countries, resulting in different time availability) can be taken into considerations. Note that all the above constraints are built on 10 days period; such design is flexible to the student; they can dynamically change their learning goals and course difficulty level according to their improvements.

4.2.2. *Simulation and Results.* To test the proposed algorithm and model, comparison experiments are conducted with the following settings.

(1) *Model Settings.* First, to build the dataset, a real e-learning dataset from an information technology diploma program is considered, where 1000 students are randomly selected. The dataset is obtained from the corresponding student affairs' system. Each student is represented by an ID, a set of learning objectives, and an ability indicator and total/sectional available time as follows:  $S = \{\text{ID: integer, objective: integer vector, ability: integer ranging between 1 and 5, total/sectional available time: integer, integer vector}\}$ . Here, we randomly select 30% students as the foreign students with a time difference of 12 hours, while the maximum number of students that a course can fit is set to 30. Each course is represented by its ID, difficulty level, required time, and covered learning objectives as follows:  $\text{Course} = \{\text{ID: integer, difficulty level: integer ranging between 1 and 5, time required: integer, objective: integer vector}\}$ . Each solution of the optimization algorithms is coded by integer strategy and has a fixed dimensionality (the number of students multiplied by the maximum number of the courses that a student can take); the maximum number of the courses that a student can take during 10 days is set to 80. The time period adopted to conduct the experiments is 60 days. Furthermore, we only randomly select three learning objectives for each student in the comparison and test the scalability of the algorithms with respect to different number of the learning objectives.

(2) *Compared Algorithms.* Three algorithms are selected in the comparison: genetic algorithm, SwarmRW, and SwarmRW-rnd [3]. For the genetic algorithm, the crossover rate is set to 1 and the mutation rate is set to 0.001; for the two swarm-based algorithms, the suggested parameter settings in the corresponding references are adopted; for the proposed HPSO-GA, the crossover rate is set to 1 and the maximum mutation rate is set to 0.05. The swarm size for each algorithm is set to 200 due to the large number of decision variables; the maximum FEs is set to  $3E + 06$ . The Wilcoxon rank sum test is adopted to conduct the statistical test, where the significance level is set to 0.05.

(3) *Results.* Figure 3 shows the comparisons between HPSO-GA and the peer algorithms with respect to CS (%). As one can see, HPSO-GA outperforms GA, SwarmRW, and SwarmRW-rnd by more than 26 %, 11 %, and 7 %, respectively. The corresponding Wilcoxon rank sum test results show that the HPSO-GA significantly outperforms the compared algorithms:  $p\text{value} = 4.21E - 10$  between HPSO-GA and GA,  $p\text{value} = 3.55E - 04$  between HPSO-GA and SwarmRW, and  $p\text{value} = 1.26E - 08$  between HPSO-GA and SwarmRW-rnd.

Furthermore, the runtime of the algorithms is shown in Figure 4. One can see that SwarmRW is of the most computational simplicity, followed by the GA. HPSO-GA and SwarmRW-inc are comparable with respect to the runtime. However, in general, the runtime of HPSO-GA is acceptable

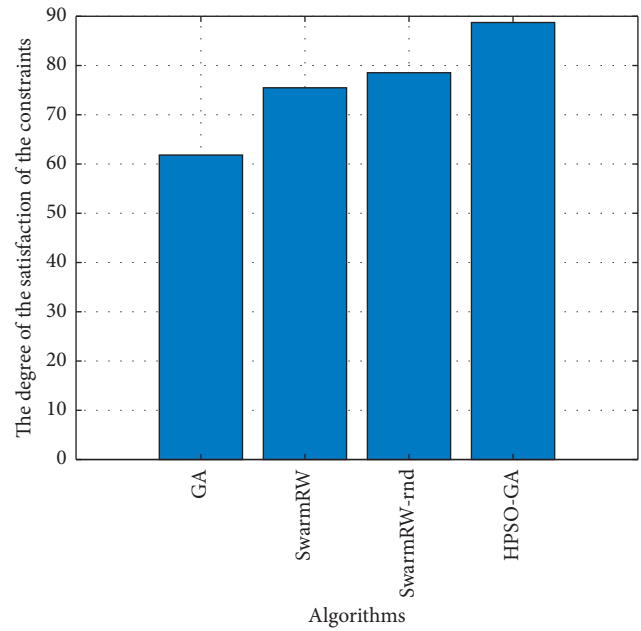


FIGURE 3: Comparison between HPSO-GA and the peer algorithms with respect to the CS (%).

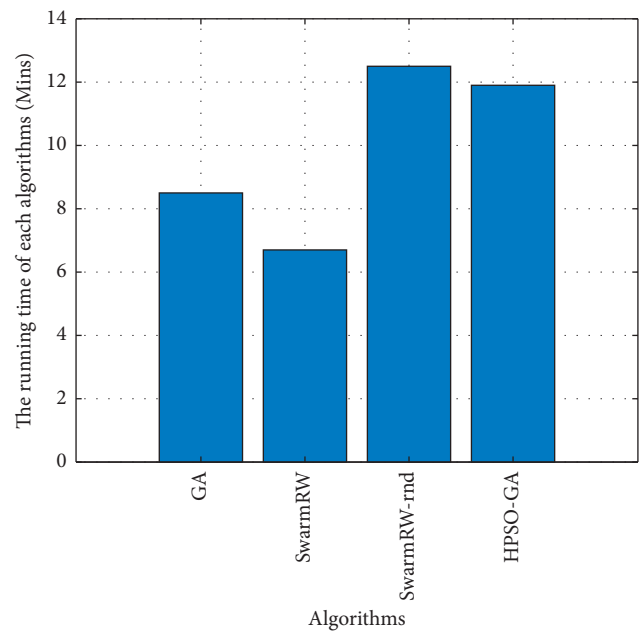


FIGURE 4: Comparison between HPSO-GA and the peer algorithms with respect to the runtime.

in comparison to the GA and SwarmRW, and it brings improvements in the optimization results.

In addition, the satisfaction of individual constraint obtained by different algorithms is shown in Table 2. As shown by the results, HPSO-GA obtains the best results for 4 times out of the 5 constraints, followed by SwarmRW-rnd wins on one constraint. The results demonstrate that the proposed HPSO-GA is not only promising in the overall goal but also competitive in individual constraints.

TABLE 2: The comparisons on individual constraint satisfaction (%) of different algorithms.

Functions	Property	GA	SwarmRW	SwarmRW-rnd	HPSO-GA
$C_{time,1}$	Mean	58.81	78.92	79.87	90.14
	Std	3.87	4.15	3.66	3.98
$C_{time,2}$	Mean	63.76	74.78	78.09	87.88
	Std	3.12	2.67	3.22	3.03
$C_{stun}$	Mean	66.33	80.65	90.12	89.88
	Std	1.67	2.21	1.89	2.56
$C_{diff}$	Mean	58.65	70.45	71.34	86.86
	Std	2.76	2.34	3.11	3.78
$C_O$	Mean	61.59	72.62	73.28	88.91
	Std	1.11	2.84	3.96	3.48

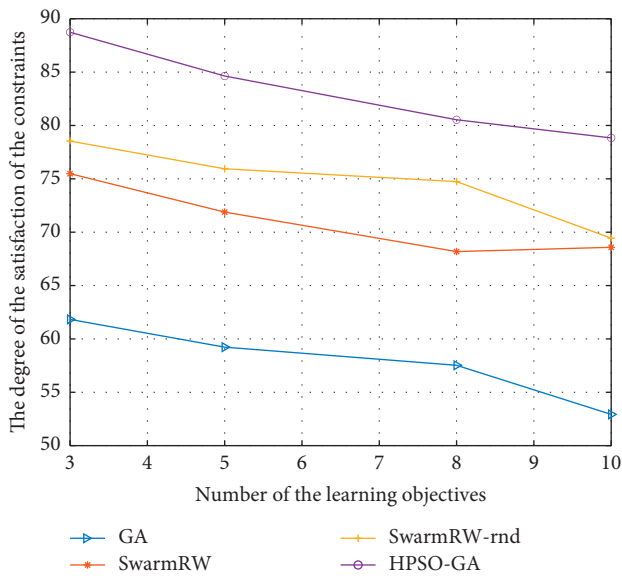


FIGURE 5: The scalability comparison between HPSO-GA and the peer algorithms with respect to the learning goals.

Finally, the scalability of different algorithms with respect to the variation of students' learning objective is tested; the results are shown in Figure 5. It can be observed that all the algorithms perform worse with the increasing of the learning objectives. This can be explained by that the difficulty of covering all the learning objectives of students becomes harder with the increasing of the number of the learning objectives. Therefore, it poses higher challenges to the searching ability of the algorithms. Nevertheless, the proposed HPSO-GA always outperforms other compared algorithms with respect to different number of learning objectives, which demonstrates the satisfied scalability of the proposed algorithm.

In summary, one can find that (i) the proposed curriculum sequencing model can be combined with real-world dataset and (ii) the proposed HPSO-GA is effective in solving the proposed model and provide proper curriculum paths for students.

## 5. Conclusions

Targeting against the modeling and optimization of the curriculum sequencing, this study first proposes a novel

hybrid PSO algorithm by designing new particle update strategies, where a competitive-genetic crossover operator and an adaptive mutation operator are proposed for both convergence and diversity preservation. In the experiments, the numerical comparisons show that the proposed algorithm is competitive in comparison to several peer algorithms, indicating the effectiveness of the proposed strategies. On the contrary, several constraints are taken into considerations to model the curriculum sequencing problem, where the sectional available time expends the reliability of the proposed model for international e-learning systems.

Furthermore, the period-based model strategy is potentially beneficial to students to dynamically change their learning strategies, which is a future study point of this study.

## Data Availability

The data are not available due to the nature of this research as participants of this study did not agree for their data to be shared publicly.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

This study was supported by the Study on the Mechanism of Industrial-Education Co-Cultivation for Interdisciplinary Technical and Skilled Personnel in Chinese Intelligent Manufacturing Industry (Planning project for the 14th Five-Year Plan of National Education Sciences (BJA210093)).

## References

- [1] A. Y. Alqahtani and A. A. Rajkhan, "E-learning critical success factors during the covid-19 pandemic: a comprehensive analysis of e-learning managerial perspectives," *Education Sciences*, vol. 10, no. 9, p. 216, 2020.
- [2] M. Ebner, S. Schön, C. Braun et al., "COVID-19 epidemic as E-learning boost? Chronological development and effects at an Austrian university against the background of the concept of "E-Learning readiness"," *Future Internet*, vol. 12, no. 6, p. 94, 2020.



- [3] M. E. B. Menai, H. Alhunitah, and H. A. Salman, "Swarm intelligence to solve the curriculum sequencing problem," *Computer Applications in Engineering Education*, vol. 26, no. 5, pp. 1393–1404, 2018.
- [4] L. D. Marcos, J. J. Martínez, and J. A. Gutiérrez, "Swarm intelligence in e-learning: a learning object sequencing agent based on competencies," in *Proceedings of the 10th Annual Conference on Genetic and Evolutionary Computation*, pp. 17–24, Atlanta, Georgia, July 2008.
- [5] L. H. Wong and C. K. Looi, "A survey of optimized learning pathway planning and assessment paper generation with swarm intelligence," in *Intelligent Tutoring Systems in E-Learning Environments: Design, Implementation and Evaluation* IGI Global, Hershey, PA, USA, 2011.
- [6] L. H. Wong and C. K. Looi, "Swarm intelligence: new techniques for adaptive systems to provide learning support," *Interactive Learning Environments*, vol. 20, no. 1, pp. 19–40, 2012.
- [7] E. Kurilovas, I. Zilinskiene, and V. Dagiene, "Recommending suitable learning scenarios according to learners' preferences: an improved swarm based approach," *Computers in Human Behavior*, vol. 30, pp. 550–557, 2014.
- [8] S. Al-Muhaideb and M. E. B. Menai, "Evolutionary computation approaches to the curriculum sequencing problem," *Natural Computing*, vol. 10, no. 2, pp. 891–920, 2011.
- [9] L. de Marcos, R. Barchino, J. Martínez, and J. Gutiérrez, "A new method for domain independent curriculum sequencing: a case study in a web engineering master program," *International Journal of Engineering Education*, vol. 25, no. 4, p. 632, 2009.
- [10] L. de Marcos, R. Barchino, J. J. Martínez, J. A. Gutiérrez, and J. R. Hilera, "Competency-based intelligent curriculum sequencing: comparing two evolutionary approaches," vol. 3, pp. 339–342, in *Proceedings of the 2008 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, vol. 3, IEEE, Sydney, Australia, December 2008.
- [11] A. F. Martins, M. Machado, H. S. Bernardino, and J. F. de Souza, "A comparative analysis of metaheuristics applied to adaptive curriculum sequencing," *Soft Computing*, vol. 25, no. 16, pp. 11019–11034, 2021.
- [12] D. Li, W. Guo, L. Wang, and M. Chen, "Particle swarm optimization-based solution updating strategy for biogeography-based optimization," in *Proceedings of the 2016 IEEE congress on Evolutionary Computation (CEC)*, pp. 455–459, IEEE, Vancouver, Canada, July 2016.
- [13] D. Li, W. Guo, A. Lerch, Y. Li, L. Wang, and Q. Wu, "An adaptive particle swarm optimizer with decoupled exploration and exploitation for large scale optimization," *Swarm and Evolutionary Computation*, vol. 60, Article ID 100789, 2021.
- [14] D. Li, W. Guo, L. Wang, and Q. Wu, "A modified apsodee for large scale optimization," in *Proceedings of the 2021 IEEE Congress on Evolutionary Computation (CEC)*, p. 1976, June 2021.
- [15] Y. l. Zheng, L. h. Ma, L. y. Zhang, and J. x. Qian, "Empirical study of particle swarm optimizer with an increasing inertia weight," vol. 1, pp. 221–226, in *Proceedings of the 2003 Congress on Evolutionary Computation, 2003. CEC'03*, vol. 1, pp. 221–226, IEEE, Canberra, Australia, December 2003.
- [16] 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.
- [17] A. Chatterjee and P. Siarry, "Nonlinear inertia weight variation for dynamic adaptation in particle swarm optimization," *Computers & Operations Research*, vol. 33, no. 3, pp. 859–871, 2006.
- [18] Z. H. Zhan, J. Jun Zhang, Y. Yun Li, and H. S. H. Chung, "Adaptive particle swarm optimization," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 39, no. 6, pp. 1362–1381, 2009.
- [19] M. Taherkhani and R. Safabakhsh, "A novel stability-based adaptive inertia weight for particle swarm optimization," *Applied Soft Computing*, vol. 38, pp. 281–295, 2016.
- [20] 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.
- [21] 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.
- [22] R. Cheng and Y. Jin, "A competitive swarm optimizer for large scale optimization," *IEEE Transactions on Cybernetics*, vol. 45, no. 2, pp. 191–204, 2014.
- [23] Y. Chen, L. Li, J. Xiao, Y. Yang, J. Liang, and T. Li, "Particle swarm optimizer with crossover operation," *Engineering Applications of Artificial Intelligence*, vol. 70, pp. 159–169, 2018.
- [24] R. L. Tang, Z. Wu, and Y. J. Fang, "Adaptive multi-context cooperatively coevolving particle swarm optimization for large-scale problems," *Soft Computing*, vol. 21, no. 16, pp. 4735–4754, 2017.
- [25] Y. Shi and R. C. Eberhart, "Empirical study of particle swarm optimization," vol. 3, pp. 1945–1950, in *Proceedings of the 1999 congress on evolutionary computation-CEC99 (Cat. No. 99TH8406)*, vol. 3, pp. 1945–1950, IEEE, Washington, DC, USA, July 1999.
- [26] Y. Shi and R. C. Eberhart, "Fuzzy adaptive particle swarm optimization," vol. 1, pp. 101–106, in *Proceedings of the 2001 congress on evolutionary computation (IEEE Cat. No. 01TH8546)*, vol. 1, pp. 101–106, IEEE, Seoul, South Korea, May 2001.
- [27] A. P. Piotrowski, J. J. Napiorkowski, and A. E. Piotrowska, "Population size in particle swarm optimization," *Swarm and Evolutionary Computation*, vol. 58, Article ID 100718, 2020.
- [28] D. Tian and Z. Shi, "Mpsos: modified particle swarm optimization and its applications," *Swarm and Evolutionary Computation*, vol. 41, pp. 49–68, 2018.
- [29] W.-N. Chen, J. Zhang, Y. Lin et al., "Particle swarm optimization with an aging leader and challengers," *IEEE Transactions on Evolutionary Computation*, vol. 17, no. 2, pp. 241–258, 2013.
- [30] R. Cheng, C. Sun, and Y. Jin, "A Multi-Swarm Evolutionary Framework Based on a Feedback Mechanism," in *Proceedings of the 2013 IEEE Congress on Evolutionary Computation*, pp. 718–724, IEEE, Cancun, Mexico, June 2013.
- [31] R. Cheng and Y. Jin, "A social learning particle swarm optimization algorithm for scalable optimization," *Information Sciences*, vol. 291, pp. 43–60, 2015.
- [32] Q. Yang, W. N. Chen, J. D. Deng, Y. Li, T. Gu, and J. Zhang, "A level-based learning swarm optimizer for large-scale optimization," *IEEE Transactions on Evolutionary Computation*, vol. 22, no. 4, pp. 578–594, 2018.
- [33] X. Zhang, H. Liu, and L. Tu, "A modified particle swarm optimization for multimodal multi-objective optimization," *Engineering Applications of Artificial Intelligence*, vol. 95, Article ID 103905, 2020.

- [34] N. Zeng, Z. Wang, W. Liu, H. Zhang, K. Hone, and X. Liu, "A dynamic neighborhood-based switching particle swarm optimization algorithm," *IEEE Transactions on Cybernetics*, pp. 1–12, 2021.
- [35] F. vandenBergh and A. P. Engelbrecht, "A cooperative approach to particle swarm optimization," *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 3, pp. 225–239, 2004.
- [36] F. V. D. Bergh, *An Analysis of Particle Swarm Optimizers*, Ph.D. thesis, University of Pretoria, Pretoria, South Africa, 2007.
- [37] X. Li and X. Yao, "Cooperatively coevolving particle swarms for large scale optimization," *IEEE Transactions on Evolutionary Computation*, vol. 16, no. 2, pp. 210–224, 2011.
- [38] Q. Qin, S. Cheng, Q. Zhang, L. Li, and Y. Shi, "Particle swarm optimization with interswarm interactive learning strategy," *IEEE Transactions on Cybernetics*, vol. 46, no. 10, pp. 2238–2251, 2016.
- [39] Y.-J. Gong, J.-J. Li, Y. Zhou et al., "Genetic learning particle swarm optimization," *IEEE Transactions on Cybernetics*, vol. 46, no. 10, pp. 2277–2290, 2016.
- [40] K. Chen, F. Y. Zhou, and X. F. Yuan, "Hybrid particle swarm optimization with spiral-shaped mechanism for feature selection," *Expert Systems with Applications*, vol. 128, pp. 140–156, 2019.
- [41] S. Mirjalili, "Genetic algorithm," *Studies in Computational Intelligence, in: Evolutionary Algorithms and Neural Networks*, Springer, New York, NY, USA, pp. 43–55, 2019.
- [42] C. Chen, C. Liu, and M. Chang, "Personalized curriculum sequencing utilizing modified item response theory for web-based instruction," *Expert Systems with Applications*, vol. 30, no. 2, pp. 378–396, 2006.
- [43] M. D. O. C. Machado, N. F. S. Bravo, A. F. Martins, H. S. Bernardino, E. Barrere, and J. F. d. Souza, "Meta-heuristic-based adaptive curriculum sequencing approaches: a systematic review and mapping of the literature," *Artificial Intelligence Review*, vol. 54, no. 1, pp. 711–754, 2021.