Hindawi
Mathematical Problems in Engineering
Volume 2023, Article ID 2277185, 11 pages
https://doi.org/10.1155/2023/2277185

*Research Article*

# Modelling and Simulation of Intelligent English Paper Generating Based on SSA-GA

**Limin Han**,[1] **Hong Gao**,[2] **and Rongjie Zhai**[2]

[1]*School of Foreign Studies, Anhui Polytechnic University, Wuhu 241000, China*
[2]*School of Artificial Intelligence, Anhui Polytechnic University, Wuhu 241000, China*

Correspondence should be addressed to Limin Han; hanlimin1995@163.com

Received 18 September 2022; Revised 21 December 2022; Accepted 12 January 2023; Published 25 January 2023

Academic Editor: Laxminarayan Sahoo



To enhance the quality and efficiency of computer-enabled generation of papers for Test for English Majors Band 8 (TEM-8), a paper generation model supported by sparrow search algorithm-genetic algorithm was studied. First, a simplified test paper generation mathematical model was set up after analyzing and studying types and characteristics of TEM-8 tasks. In the model, quantity, type, difficulty, discrimination degree, scores, exposure, and answering time of test questions were taken into consideration. To enhance the optimizing effect of the genetic algorithm for searching test questions, the traditional genetic algorithm was improved by introducing the sparrow search algorithm into the model to achieve a better crossover rate, variance rate, optimization precision, and speed of the genetic algorithm. A new sparrow search-genetic algorithm (SSA-GA) was designed, and the optimizing effect of SSA-GA was verified to be ideal through optimizing six standard test functions. Then, SSA-GA was applied to conduct experimentation with test paper generation, and comparison with traditional genetic algorithms was also made. The values of best and average fitness of SSA-GA were better than those of the traditional genetic algorithm (GA) in the paper generation. Exposure rate and success rate in TEM-8 paper generation of SSA-GA were higher than those of traditional GA in TEM-8 paper generation. Results showed that the studied SSA-GA could implement test paper generation with higher speed and better quality.

## 1. Introduction

In the context of increasing social development and exchanges in economy, culture, and science and technology among all countries, English has become one of the most widely used communication languages [1, 2]. The educational circles of all countries have become aware of the important role English play in boosting their integration into the international community and promoting their all-round development and, consequently, are all intensifying their efforts in advancing English teaching [3–5]. In the course of English teaching, conducting tests of all types is inevitable. Hence, test paper generation has become a heavy task in the everyday work of English teachers. With the development of artificial intelligence (AI) technology, computer can simulate human thinking to replace mankind in fulfilling tasks that require intelligence. So far, many scholars have studied computer-aided test paper generation. For example, Wu explored a computer-based paper generation model in view of learning effect predictions to enhance the quality of computer test paper generation. In this prediction, the deep knowledge tracking method is used, and the test paper scores consider the weight of the technique, difficulty degree of the test papers, and distribution laws of test questions. Wu's model can ensure the distribution field of test questions and high validity of test paper generation effectively [6]. According to the fuzzy mathematical theory [7], test questions can be selected randomly in test data through quantitative processing of paper generation requirements. To solve the trouble in traditional test paper generation, such as the unclear objective functions and low precision of optimum solution, Yang et al. proposed the particle swarm optimization (PSO) algorithm and applied it to solve functions of test paper generation for multiple objectives and

has greatly enhanced the level of intelligent tests of the online testing system [8]. El-Rahman and Zolait designed a test bank and studied a shuttle algorithm, which ensured the online formation of test papers with varying difficulty levels and requirements and could save the time and cost of teaching greatly [9]. Duan et al. introduced an automatic method of test paper generation via the knowledge embedding method. First, the embedded knowledge and mechanism of test paper generation are studied and analyzed. Second, modelling is completed. The model can enhance the efficiency of test paper generation significantly [10]. To avoid the problem of low speed in traditional test paper generation, Zhang et al. offered a new test paper generation method, which uses random functions to generate multiple numbers and increase the random numbers. On this basis, random numbers are sequenced according to test requirements; the method is used to enable the simultaneous selection from the test bank and test paper generation, thereby enhancing the efficiency and reducing the duration of test paper generation [11]. Nguyen et al. provided a submodular meme approximation algorithm to address the disturbance of optimization by dual objectives during test paper generation. While strengthening the local search mechanism of the sub-module, this algorithm has upgraded the quality of test paper generation to a large degree [12]. Paul presented a genetic algorithm that can be used to quickly create a test paper generation template [13]. Lu et al. used Delphi technology together with the SQL server in completing the establishment of test bank and the automatic test paper generation system that can work stably [14]. Härtel et al. designed a language test paper generation system based on grammar and a goal-oriented framework and has a satisfactory effect [15]. Du et al. presented a method of searching knowledge points based on ontology and, on this basis, designed an automatic system of test paper generation [16]. Li et al. proposed an improved test paper generation algorithm of back testing, fit for test paper generation in simultaneous examination [17]. Ruisen et al. put forward a bottom-up method of creating a test bank of engineering graphics to set up a multipath method for question selection and test paper generation on this basis [18]. To address the low success rate of the computer-enabled intelligent test paper generation algorithm, many scholars advanced test paper generation models by means of the genetic algorithm [13, 19–21]. Practices prove that the coding method and iterative mode of the genetic algorithm are highly suitable for optimized solutions during test paper generation. In order to determine paper generation speed, the consuming time can be calculated on the basis of the complexity function [22]. The standard indexes can be set and the quality of the paper generation system can be evaluated through the fuzzy graph theory [23]. Therefore, many scholars have achieved many results in their studies [24–26].

Therefore, this paper takes the automatic test paper generation system for TEM-8 as the study object and applies the genetic algorithm to create an automatic test paper generation model for TEM-8. However, although the genetic algorithm suits the multigoal optimization of intelligent paper generation well, its parameters (e.g., crossover rate and variation probability) are selected via designer's experience, thereby resulting in different optimization effects of the genetic algorithm and unstable quality of test paper generation. Therefore, a wide space exists for studies and technical problems that are in need of solution. Theory of determining journey order based on graph's Wiener absolute index and inverse fuzzy mixed graphs, proposed by Poulik and Ganesh, is suitable for finding the parameters of GA [27, 28]. However, a lot of calculation is still needed. Optimizing algorithms such as particle swarm optimizing algorithm, the simulated annealing algorithm can be employed to find the suitable key parameters for GA. SSA, proposed recently with greater searching ability and less parameters, has been employed to optimize complicated problems with good effect [29]. In order to simplify the process of finding good parameters, this paper introduced the sparrow search algorithm into intelligent automatic test paper generation for optimization of GA key parameters to design a new sparrow-genetic algorithm for test paper generation.

This paper is organized as follows. In the forth-coming section, models of intelligently generating TEM-8 paper are built on analyzing test organization and requirements. Subsequently, the sparrow search algorithm is employed to optimize the crossover and variation probabilities of the traditional genetic algorithm. On this basis, SSA-GA is developed and verified through six standard test functions. In the final section, experiments of intelligent test paper generation are carried out. Results show that SSA-GA can generate test papers for TEM-8 with higher quality and greater speed than traditional GA.

## 2. Analysis and Modeling of TEM-8 Intelligent Paper Generation

As an examination for undergraduates, TEM (including TEM-4 and TEM-8) is conducted to test how teaching syllabus for undergraduate English majors is implemented. TEM-8 tasks are as follows: listening comprehension, reading comprehension, language knowledge, translation, and writing. TEM-8 is an all-round examination to test the comprehensive linguistic proficiency of students majoring in English. Each year, a large number of students prepare for the examination by utilizing all types of professional English test banks. If a test bank can be utilized to quickly create the TEM-8 test papers that satisfy examination requirements, it will benefit a large number of students in their studies and examinations.

TEM-8 paper users can be divided into two groups: those preparing for an examination and those taking part in an examination. The former needs a personalized test paper that can provide more training aiming at their own weaknesses. At different stages, test papers with varying difficulty are needed. Meanwhile, the latter needs a standardized test paper, which should be similar to TEM-8 test papers in all aspects as much as possible so as to achieve a better simulation effect. To meet the needs of different users, a plan is adopted to make parallel search for multiattribute

parameters. The TEM-8 test papers are defined as a collection of multiple attributes, which includes quantity, type, difficulty, discrimination degree, scores, exposure, and answering time of test questions. The model of a paper can be created in light of the above attributes. First, the target matrix of one single-question model is shown in the following formula:

$$S_1 = [\, a_1 \;\; a_2 \;\; a_3 \;\; a_4 \;\; a_5 \;\; a_6 \;\; a_7 \,], \tag{1}$$

where $S_1$ stands for the model of one certain question, $a_1$ is the question type, $a_2$ is the question knowledge points, $a_3$ is the question score, $a_4$ is the question discrimination degree, $a_5$ is the question difficulty, $a_6$ is the expected question-answering time, and $a_7$ is the question exposure.

It is possible to create a question that meets the requirements by fully restraining the seven attributes. The target matrix of the single-question model is transformed into the matrix that includes the model $S$ of $n$ number of questions.

$$S = [\, S_1 \;\; S_2 \;\; S_3 \;\; \cdots \;\; S_n \,]^T. \tag{2}$$

Through the target matrix, the paper generation issue turns into the issue of solving multitarget restraints. In the matrix, each line means one question (there are $n$ number of questions) and includes the seven attributes of a question. Matrix $S$ is the target-state matrix of this paper generation method. Through the restraint and solution of the target matrix parameters, the online TEM-8 test paper generation is completed. The attributes of TEM-8 test paper are calculated as follows.

*2.1. Restraint of Question Types.* The attribute of question type plays a vital role in evaluating score distribution of paper in later period. It is necessary to conclude this attribute when test questions are stored in test bank. TEM-8 question types, such as listening comprehension, reading comprehension, language knowledge, translation, and writing, are marked with numbers 1–5, respectively. When a certain type of questions is chosen, its $a_1$ value is the number that corresponds to the question type.

The matrix of the attributes for question types in a TEM-8 test paper is as follows:

$$S\_\text{type} = [\, S_1 a_1 \;\; S_2 a_1 \;\; S_3 a_1 \;\; \cdots \;\; S_n a_1 \,], \tag{3}$$

where $S$_type is the matrix of question type and $S_i a_1$ is the question type of question no. $i$.

Through formula (4), the number of different elements in $S$_type is judged to obtain the quantity $S$_type_data in TEM-8 test papers.

$$S\_\text{type\_ data} = \text{numel}(\text{unique}(S\_\text{type})). \tag{4}$$

*2.2. Restraint of Knowledge Points in a Test Paper.* Knowledge points are very important and determined by test syllabuses. If knowledge points tested in a paper are not from the syllabuses, the generated paper is not a qualified one. The knowledge points of test questions differ from questions types because a one-to-one relationship exists between the question type and one question, whereas a multiple-to-multiple relationship exists between the knowledge points and the question itself. The same knowledge point may exist in multiple questions, and one question may include multiple knowledge points. The quantity of knowledge points and their appearance frequency decide the test paper difficulty and the examination requirements and effect.

The knowledge points for a single test question are shown in the following formula:

$$A_2 = [\, k_1 \;\; k_2 \;\; k_3 \;\; k_4 \;\; k_5 \;\; \cdots \;\; k_p \,], \tag{5}$$

where $A_2$ is the matrix of single questions and $K_i$ is no. $i$ knowledge point. Suppose TEM-8 includes $p$ knowledge points. A specific question is taken as an example. Value $k_i$, which corresponds to knowledge points included in the question, is set to 1, and value $i$, which corresponds to knowledge points not included, is set to 0.

$$A_2\text{\_point\_num} = \text{size}(A_2(A_2 \sim\; = 0), 2), \tag{6}$$

where $A_2$_point_num is the number of knowledge points included in a single question.

The knowledge point matrix S_point for TEM-8 test paper is shown in the following formula:

$$S\_\text{point} = [\, S_1 A_2 \;\; S_2 A_2 \;\; S_3 A_2 \;\; \cdots \;\; S_n A_2 \,], \tag{7}$$

where $S_i A_2$ is the knowledge point matrix of question no. $i$.

The total knowledge included in a TEM-8 test paper is expressed by the following formula:

$$S\_\text{point\_all} = S_1 A_2 + S_2 A_2 + S_2 A_2 + \cdots + S_n A_2, \tag{8}$$

where $S$_point_all is the matrix of the number of examinations for every knowledge point in a test paper.

$$S\_\text{point\_num} = \text{size}(S\_\text{point\_all}(S\_\text{point\_all} \sim\; = 0), 2), \tag{9}$$

where $S$_point_num is the sum of knowledge points included in a paper.

The attribute parameters of knowledge points can be obtained through calculating the number of knowledge points in a test and its total number in syllabuses.

$$S\_\text{point\_ data} = \frac{S\_\text{point\_num}}{p}, \tag{10}$$

where $S$_point_data is the attribute parameters of knowledge points in a paper.

*2.3. Restraint of the Total Score.* The total score is an important condition that restrains test paper generation. Under the conditions established for a test, it thereof should be an accurate value. The total score restraint is solved as follows:

$$S\_\text{sco\_ data} = \sum_{i=1}^{n} S_i a_4, \tag{11}$$

where $S$_sco_data is the total score of a paper and $S_i a_4$ is the score of question no. $i$. For example, in tests for TEM-8, $S$_sco_data is 100.

*2.4. Restraint of Discrimination.* The discrimination degree of test questions is used to evaluate students' mastery of relevant knowledge and content. It directly affects students' scores and reflects students' learning ability. It is an index that cannot be obtained directly but only through testing, statistics, and analysis. The discrimination degree of every question is obtained through the following formula with data from previous examinations:

$$S\_\text{dis}_i = \frac{\overline{D_h} - \overline{D_l}}{W}, \tag{12}$$

where $S$_dis$_i$ is discrimination degree of question no. $i$, $D_h$ is the average result of questions among examinees' high-score group (top 25%), $D_l$ is the average result of questions among examinees' low-score group (bottom 25%), and $W$ is the full score of questions.

The discrimination parameter $S$_dis_data of a test paper is expressed as follows:

$$S\_\text{dis}\_\text{data} = \frac{1}{n}\sum_{i=1}^{n} S\_\text{dis}_i. \tag{13}$$

For example, if $S$_dis_data $\geq 0.4$, the quality of a generated paper is very good; if $0.3 \leq S$_dis_data $< 0.4$, the quality is comparatively good; if $0.2 \leq S$_dis_data $<0.3$, the quality is comparatively bad; if $S$_dis_dat $< 0.2$, the quality is very bad.

*2.5. Restraint of Test Difficulty.* Test difficulty, which can greatly affect test takers' performance, is one of the core attributes in a test. The difficulty coefficient is an index used to test the difficulty degree of a test paper and is expressed by the following formula:

$$S_i a_5 = 1 - \frac{\overline{Y}}{W}, \tag{14}$$

where $S_i a_5$ is the difficulty of question no. $i$ and $\overline{Y}$ is the average score of all examinees.

For example, $S_i a_5$ is in the domain (0.0, 1.0). The greater this value is, the higher the difficulty is, which means few students can answer the question correctly.

The difficulty degree is solved through the following formula:

$$S\_\text{dif}\_\text{data} = \frac{1}{S\_\text{sco}\_\text{data}}\sum_{i=1}^{n} S_i a_5 \times S_i a_4, \tag{15}$$

where $S$_dif_data is the difficulty degree of a test paper.

*2.6. Expected Question-Answering Time.* The time on answering every question should be based on students' average answering time, and the total time spent on answering all questions should be less than the examination standard time.

$$S\_\text{tim}\_\text{data} = \sum_{i=1}^{n} S_i a_6, \tag{16}$$

where $S$_tim_data is the expected total question-answering time for a test paper, and its value should be less than or equal to the examination standard time. $S_i a_6$ is the expected question-answering time of question no. $i$. For example, in TEM-8, $S$_tim_data is 150 MIN.

*2.7. Exposure of Test Questions.* The paper exposure determines the confidentiality of one paper. The smaller the exposure is, the higher test quality is. The exposure of test questions is a parameter constantly updated with the increasing frequency of test paper generation. The initial questions exposure is set to $E$. In the subsequent paper generation, the exposure increases by 1 each time when it is selected, as is shown in the following formula:

$$S_i a_7 = \begin{cases} E + 1, & \text{selected}, \\ E, & \text{not selected}, \end{cases} \tag{17}$$

where $S_i a_7$ is the exposure of question no. $i$.

$$S\_\text{exp}\_\text{data} = \frac{1}{n}\sum_{i=1}^{n} S_i a_7, \tag{18}$$

where $S$_exp_data is the exposure of a test paper.

## 3. Design of Sparrow Search-Genetic Algorithm

The traditional genetic algorithm is widely applied in optimization solution. However, in the operation of population crossover and variation, determining the coefficients of crossover and variation through designers' experience will restrict the solution capability of the genetic algorithm. The solution capability of an algorithm can be enhanced to a certain extent by the probabilities of crossover and variation determined through people's experience in test paper generation. However, that will be a heavy workload for the people. The continuous changes in test questions will lead to increasing workload in setting the probabilities of crossover and variation. Therefore, the sparrow search algorithm is used in online optimization of the probabilities of crossover and variation in the genetic algorithm. Hence, a new sparrow search-genetic algorithm is designed and used in TEM-8 test paper generation.

A method is proposed herein to use the sparrow search algorithm to select the probabilities of crossover and variation of the genetic algorithm. When the sparrow search algorithm continuously compares and selects different probabilities of crossover and variation, the genetic algorithm solves the precision of the mean value of results repetitively to select the optimal probabilities of crossover and variation after 100 iterations of the sparrow search algorithm, which are applied to the genetic algorithm for paper generation.

The sparrow search algorithm is a newly-proposed (in 2020) swarm intelligence algorithm [30, 31], whose fundamental principle is to simulate the foraging and antipredator behavior of a sparrow population. The algorithm model

includes the detector model, participant model, and early warner model. With higher energy reserve and a high value of fitness, the detector mainly provides the sparrow population with foraging areas and orientation. The individuals selected as detectors are those with a fitness value among the top 20% of the entire population. A joiner follows a detector with the optimal value of fitness in looking for food to obtain his own energy reserve and increase its own value of fitness. The remaining 80% of the individuals in a population are joiners. Some joiners continuously observe the detectors to vie for a position that provides more food. An early warner will provide an alarm when he becomes aware of any danger and, in the meantime, will quickly transfer to a safer area for a better position. The sparrow population will walk randomly to approach other sparrows, which is an antipredator behavior. Meanwhile, when the value of alarm is larger than the safety value, the detector will push the joiners away from the dangerous area.

Given that a detector leads the flow of the entire sparrow population, he can search for food in any place. $R_2 \in [0,1]$ is a random number that stands for early warning value and ST $\in [0.5, 1]$ stands for safety value, which is 0.8 for this purpose.

$R_2 < \mathrm{ST}$ means that there is no predator nearby, and the detector will get into a broad search mode. His position is updated as follows:

$$X_{i,j}^{t+1} = X_{i,j}^{t} \cdot \exp\left(\frac{-i}{\alpha \cdot \mathrm{iter}_{\max}}\right), \tag{19}$$

where $X_{i,j}^{t}$ is the current position of the sparrow, $X_{i,j}^{t+1}$ is the updated position, $t$ is the current number of iterations, $\mathrm{iter}_{\max}$ is the maximum number of iterations, and $\alpha$ is a random number which is evenly distributed among (0, 1).

$R_2 \geq ST$ means that there is danger around and all sparrows have to fly to other safe areas. The specific position updating is presented as follows:

$$X_{i,j}^{t+1} = X_{i,j}^{t} + Q \cdot L, \tag{20}$$

where $Q$ is a random number that follows a normal distribution and $L$ is the matrix of $1 \times d$ with elements that are all 1 ($d$ is the dimension of variable), that is, every internal element is 1.

A joiner will follow a detector in search of food and may vie with the detector to increase his own food. $i > n/2$ means that no. $i$ joiner with a poor fitness may starve to death. At this time, the joiner needs to look for food in other places. The formula of position updating is presented as follows:

Given $i > n/2$,

$$X_{i,j}^{t+1} = Q \cdot \exp\left(\frac{X_{\mathrm{worst}}^{t} - X_{i,j}^{t}}{a \cdot \mathrm{iter}_{\max}}\right), \tag{21}$$

where $X_{\mathrm{worst}}^{t}$ is the worst position in a sparrow population at the $t$ time iteration.

Given $i \leq n/2$, no. $i$ joiner randomly finds a position for foraging near the optimal position of the detector.

$$X_{i,j}^{t+1} = X_{p\mathrm{best}}^{t+1} + \left|X_{i,j}^{t} - X_{p\mathrm{best}}^{t+1}\right| \cdot A^{+} \cdot L, \tag{22}$$

where $X_{p\mathrm{best}}^{t+1}$ is the optimal position of the detector at $t+1$ time iteration. $A$ is a multidimensional matrix with internal elements that are 1 or $-1$, and $A^{+} = A^{T}\,(AA^{T})^{-1}$.

$f_i$ is the fitness value of current individual $i$, and $f_g$ and $f_w$ are the best and worst fitness values of the current sparrow population, respectively. $f_i > f_g$ means that a sparrow is on the edge of its population. For the sparrows to be aware of the danger, their position updating formula is expressed as follows:

$$X_{i,j}^{t+1} = X_{g\mathrm{best}}^{t+1} + \beta \cdot \left|X_{i,j}^{t} - X_{g\mathrm{best}}^{t}\right|, \tag{23}$$

where $X_{gbest}$ is the present global optimal position in a population and $\beta$ is the step length parameter that follows a normal distribution with 0 mean value and 1 variance.

$f_i = f_g$ means the sparrow is in the population center and has become aware of the danger. The sparrow draws close to other individuals to protect itself from the attack of predators. Its position updating formula is expressed as follows:

$$X_{i,j}^{t+1} = X_{i,j}^{t} + k \cdot \left|\frac{X_{i,j}^{t} - X_{\mathrm{worst}}^{t}}{(f_i - f_w) + \varepsilon}\right|, \tag{24}$$

where $\varepsilon$ is a very small constant used to avoid the denominator of 0. $k \in [-1, 1]$ is used to control the sparrow's movement direction. Figure 1 displays the flowchart of the sparrow search-genetic algorithm.

In order to verify solution capability of the studied sparrow search algorithm-genetic algorithm, we solved different test functions for an analytical comparison with the general genetic algorithm. Table 1 lists the functions and parameters used in the test.

$F1$–$F3$ are unimodal functions. Only one global optimum in the section, which can reduce the impact upon algorithm solution caused by local optimal solution, is considered. Therefore, these functions can evaluate and compare the solution capability of the algorithm before and after optimization very well. $F4$–$F6$ are multimodal functions. Different from unimodal functions, multimodal functions include multiple locally optimal solutions, and their number will show an exponential growth with increasing problems (number of design variables). Falling into local optimal solution, the problem is most likely to appear in optimized algorithms. Therefore, the solution of multimodal functions is leveraged to observe the algorithm ability to jump out of local optimum. Through solution of different types of functions, stability of the algorithm solution can be verified.

Figure 2 shows the solution and iteration processes of optimizing test functions by using the sparrow-genetic algorithm involved in our study. In the process, normal GA is the solution and iteration curve of the general genetic algorithm, with 0.2 being selected as the value for its crossover and variation probabilities. SSA-GA corresponds to the improved genetic algorithm that uses the sparrow search algorithm to select crossover and variation probabilities. This time, the sparrow search algorithm selected a crossover probability of 0.26958 and a variation probability of 0.43144.

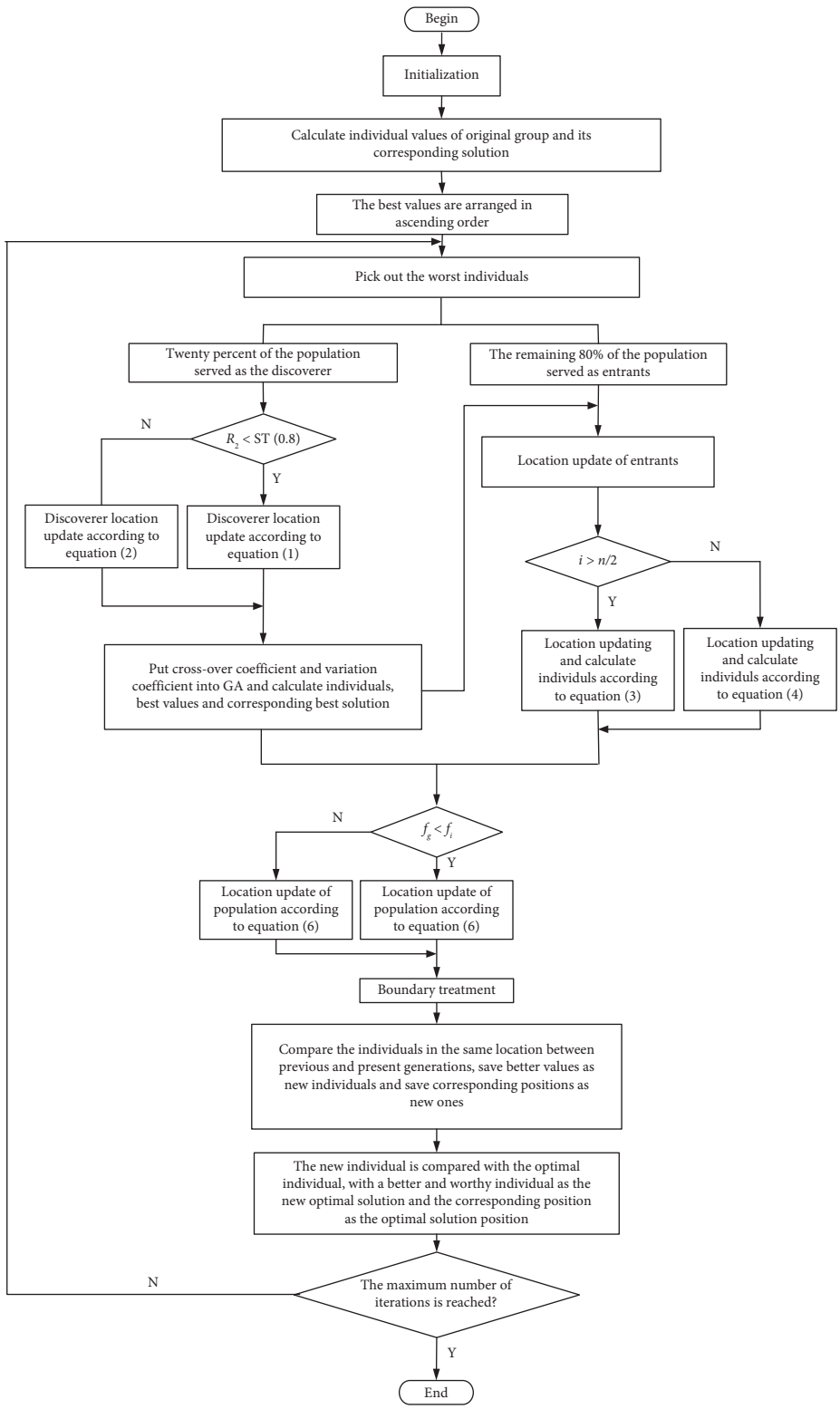


Figure 1: Flowchart of SSA-GA.

To better display the difference of the two algorithms in the convergence process, Figures 2(a)–2(e) use logarithmic coordinate axis as longitudinal coordinates. Given that the optimal value of the functions in Figure 2(f) is a negative value, a linear coordinate axis is used as longitudinal coordinates.

As it is shown in Figures 2(a)–2(f), the sparrow search-genetic algorithm displays a higher convergence speed and

Table 1: Test functions.

| Function | $V$_no | Range | $f_{\min}$ |
|---|---|---|---|
| $F_1(x) = \sum_{i=1}^{n} x_i^2$ | 30 | [−100, 100] | 0 |
| $F_2(x) = \sum_{i=1}^{n} \lvert x_i \rvert + \prod_{i=1}^{n} \lvert x_i \rvert$ | 30 | [−10, 10] | 0 |
| $F_3(x) = \sum_{i=1}^{n} (\sum_{j-1}^{n} x_j)^2$ | 30 | [−100, 100] | 0 |
| $F_4(x) = 1/4000 \sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n} \cos(x_i/\sqrt{i}) + 1$ | 30 | [−600, 600] | 0 |
| $F_5(x) = \sum_{i=1}^{11} [a_i - x_1(b_i^2 + b_i x_2)/b_i^2 + b_i x_3 + x_4]^2$ | 4 | [−5, 5] | 0.00030 |
| $F_6(x) = 4x_1^2 - 2.1x_1^4 + 1/3x_1^6 + x_1x_2 - 4x_2^2 + 4x_2^4$ | 2 | [−5, 5] | −1.0316 |

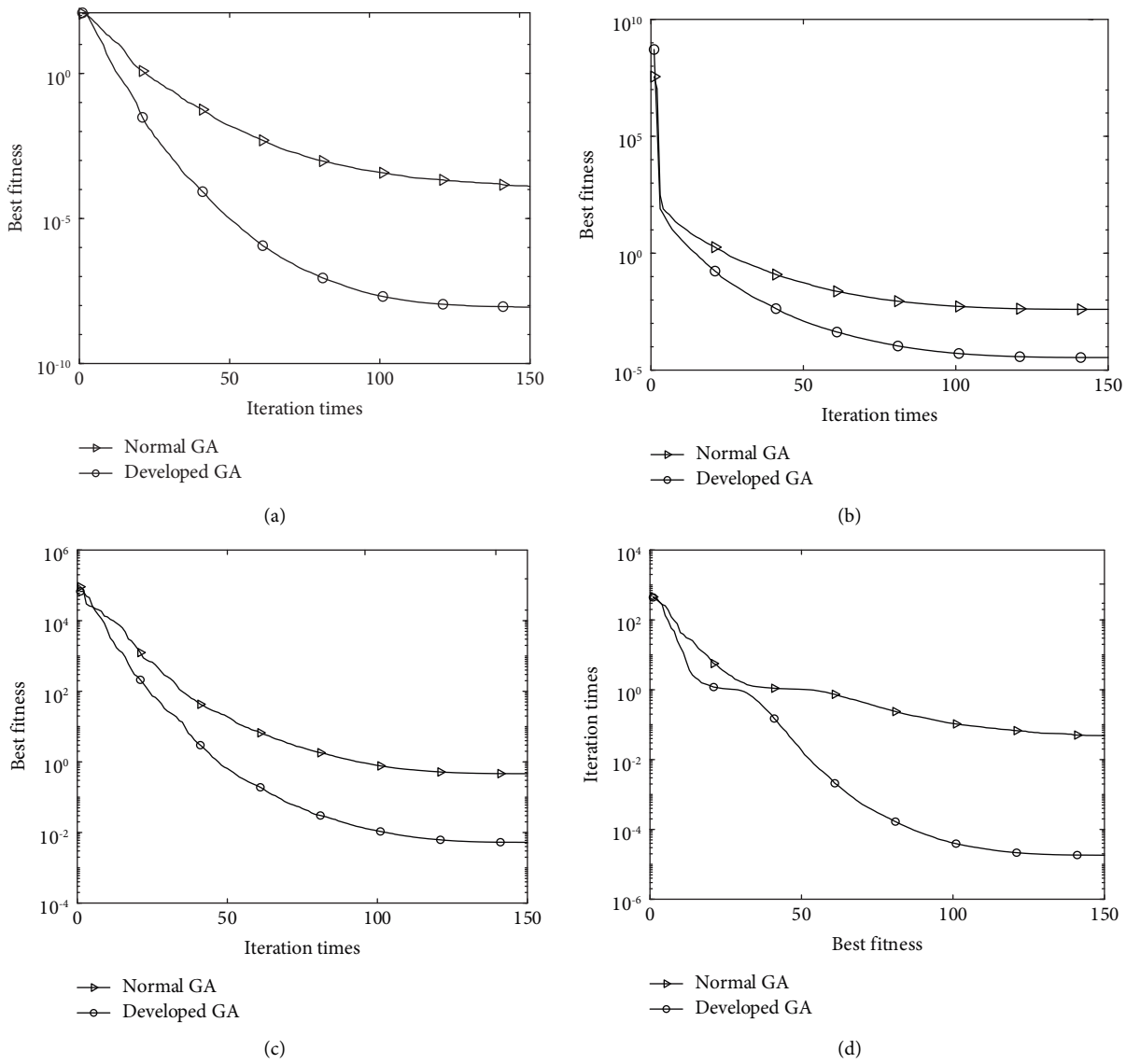


Figure 2: Continued.

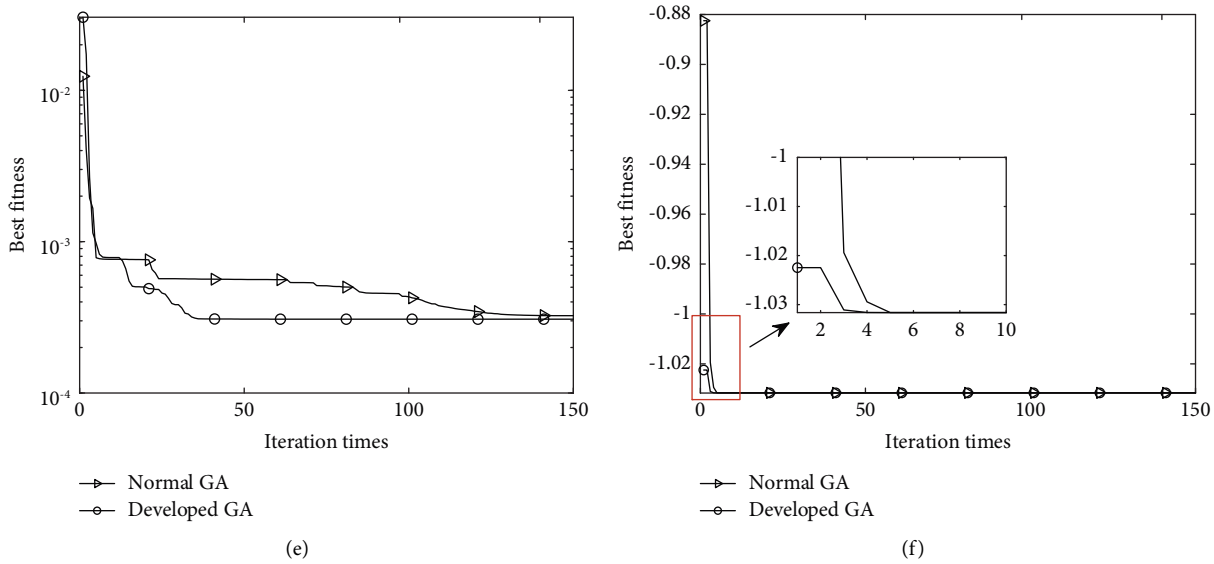


Figure 2: Comparison of the process of solving test functions $F1$–$F6$ by using the two algorithms. (a) Solution process of test function $F1$. (b) Solution process of test function $F2$. (c) Solution process of test function $F3$. (d) Solution process of test function $F4$. (e) Solution process of test function $F5$. (f) Solution process of test function $F6$.

solution precision in the solution process of different test functions. This finding proves that the sparrow search-genetic algorithm has a better solution capability and a higher solution precision. Particularly, from the solution process of multimodal functions $F4$–$F6$, the sparrow search-genetic algorithm has markedly enhanced its ability to jump out of local optimum and achieved a higher convergence speed, thereby achieving a significant optimization of the genetic algorithm.

To verify the high solution precision and better universality and stability of the sparrow search-genetic algorithm, the mean values and standard deviation of 20 solution results are calculated. The results are presented in Table 2.

The comparison of the mean values and standard deviation of 20 solution results shows that the sparrow search-genetic algorithm has more accurate mean values of the solution results than a normal GA, indicating that the solution precision of SSA-GA is generally higher. For SSA-GA, the standard deviation of the results after 20 solutions is smaller, indicating that SSA-GA is more stable in terms of solution capability. Thus, the SSA-GA algorithm can be proven to have a higher convergence speed, and the solutions are more accurate and stable.

Table 2: Mean values and standard deviation of multiple solution results of different functions.

| $F$ | Normal GA | | Developed GA | |
|---|---|---|---|---|
| | Ave | Std | Ave | Std |
| $F1$ | $1.0611E-04$ | $5.3042E-05$ | $2.5142E-08$ | $1.8045E-08$ |
| $F2$ | $3.9673E-03$ | $9.6285E-04$ | $4.4253E-05$ | $1.2281E-05$ |
| $F3$ | $1.0801E-02$ | $8.906E-03$ | $2.5217E-03$ | $2.2736E-03$ |
| $F4$ | $6.6021E-02$ | $2.8997E-02$ | $1.1326E-02$ | $1.5153E-02$ |
| $F5$ | $1.5536E-03$ | $4.4329E-03$ | $3.0104E-04$ | $2.8123E-04$ |
| $F6$ | $-1.0316$ | $4.4541E-13$ | $-1.0316$ | $1.8506E-13$ |

## 4. Experiment of Intelligent Test Paper Generation Based on SSA-GA Algorithm

A standard test bank is used to conduct simulation experiment of a separate TEM-8 paper generation through the traditional genetic algorithm and SSA-GA algorithm. The simulation experiment uses the Windows 7 operating system, a CPU of Intel Core i5, and an internal memory of 8 GB. The test bank includes banks for listening comprehension, reading comprehension, language knowledge, translation, and writing. Every test bank includes 5000 questions, which form a test paper composed of 25 questions of listening comprehension, 22 questions of reading comprehension, 10 questions of language knowledge, 1 translation task, and 1 writing task. Table 3 shows the question types, question number, and score share of a TEM-8 paper formed by the genetic algorithm.

The crossover and variation probabilities of the test paper generation via the SSA-GA algorithm are obtained through the use of sparrow search algorithm optimization. The values of the crossover and variation probabilities are 0.26274 and 0.48163. The number of individuals in a population and the evolutionary algebra is 50 and 250. The test paper generation experiments by the two algorithms were run for 100 times to form 100 sets of TEM-8 papers. Figure 3 shows the convergence curve of the best fitness of the two algorithms. Through analyzing convergence curve of the best fitness, the normal GA algorithm is proved to have certain advantages in the early period. However, after 60 iterations, the value of the fitness function changes slowly. The studied SSA-GA algorithm can better jump out of local optimum for an improved evolution. The fitness of the SSA-GA algorithm obtained after the quick excess of the normal GA algorithm can reach the optimal value sooner.

Table 3: Types, quantity, and scores of TEM-8 paper.

| Question type | Question number | Score share in paper |
|---|---|---|
| Listening comprehension | 25 | 25 |
| Reading comprehension | 22 | 30 |
| Language knowledge | 10 | 10 |
| Translation | 1 | 15 |
| Writing | 1 | 20 |

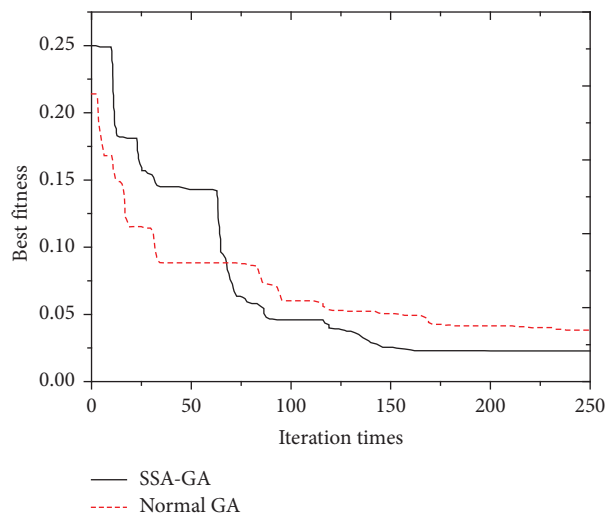


Figure 3: Best convergence curves of the two algorithms in the experiment.

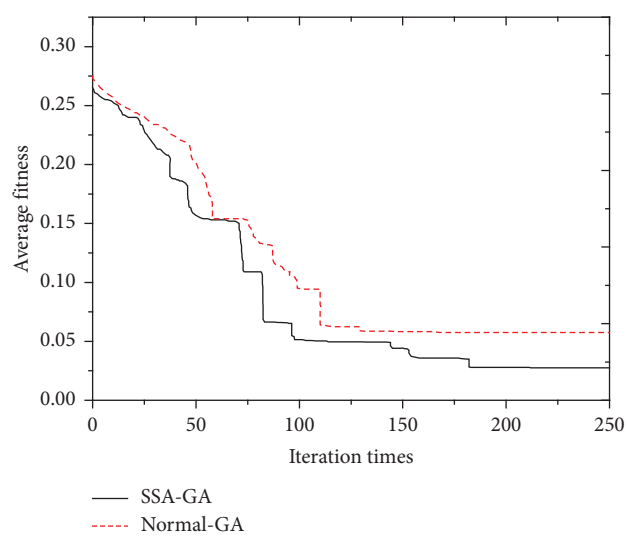


Figure 4: Average convergence curves of the two algorithms in the experiment.

Meanwhile, we determined the relationship curve that reflects the iteration-based changes of average fitness of the two algorithms, as shown in Figure 4. In Figure 4, the SSA-GA algorithm manifests marked advantages with the continuation of the iteration process. Meanwhile, we calculated the average fitness of 50, 100, 150, 200, and 250 iterations, as shown in Table 4. In Table 4, compared with that of the traditional genetic algorithm, the average fitness of the SSA-GA algorithm after 50, 100, 150, 200, and 250 iterations is improved by 22.3%, 27.3%, 24.1%, 35.1%, and 38.5%, respectively.

To further examine the performance of the studied SSA-GA algorithm and normal GA algorithm, time consumed by test paper generation is calculated, as shown in Figure 5. In Figure 5, given that the size of English test database and iterations are the same, the studied SSA-GA algorithm consumes less time than normal GA and will show even more obvious advantages with increasing iterations.

Table 4: Average fitness comparison of different iterations.

| Iteration | Normal GA | SSA-GA |
|---|---|---|
| 50 | 0.201 | 0.156 |
| 100 | 0.084 | 0.061 |
| 150 | 0.058 | 0.044 |
| 200 | 0.057 | 0.037 |
| 250 | 0.057 | 0.035 |

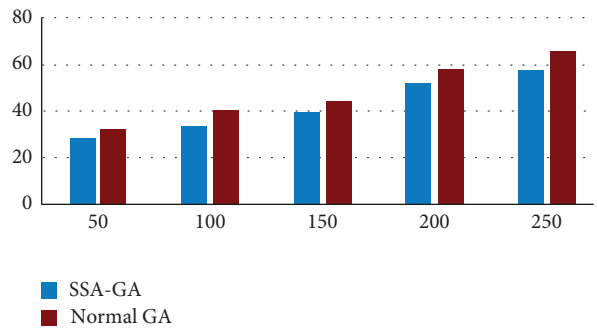


Figure 5: Average time consumption of TEM-8 test paper generation.

Table 5: Comparison of test paper generation results.

| Algorithm | Exposure rate (%) | Success rate (%) |
|---|---|---|
| Normal GA | 4.51 | 93 |
| SSA-GA | 2.74 | 99 |

The exposure of test questions is an important factor for evaluating test paper quality. The higher the exposure is, the lower quality a test has, and the more similar the papers are. Therefore, the exposure and success rate of the two algorithms were calculated, as shown in Table 5. For the TEM-8 papers created by the studied SSA-GA algorithm, the exposure rate is smaller than that of normal GA algorithms and success rate of SSA-GA is higher than that of normal GA. This comparison shows that the studied SSA-GA algorithm exceeds the normal GA algorithm in improving test quality and is more suitable for TEM-8 paper generation.

## 5. Conclusions

Taking TEM-8 intelligent test paper generation as the core, this study proposed a strategy for intelligent test paper generation based on the sparrow-genetic algorithm. Tasks that have been fulfilled are summarized as follows:

(1) Firstly, types, scores, and focuses of test papers for TEM-8 examinees were analyzed. On this basis, test questions were divided into 7 restraints, such as types, knowledge points, scores, discrimination, difficulty and exposure of test questions, and expected question-answering time, to build a mathematical model.

(2) To upgrade the precision and speed of the genetic algorithm optimization, a new sparrow search algorithm was introduced to optimize the crossover and variation probabilities of the traditional genetic algorithm for the design of the sparrow-genetic algorithm. Six standard test functions were also used to test the sparrow-genetic algorithm. The results proved that the studied sparrow search-genetic algorithm has a stronger optimization capability.

(3) A new sparrow search-genetic algorithm was used to test TEM-8 intelligent paper generation system, and the results were compared with the test paper generation with the normal GA algorithm. The finding showed the new algorithm was better than the traditional one in terms of best fitness, average fitness, time of paper generation, and questions exposure. Hence, it can be proved that the designed sparrow search-genetic algorithm had a better effect on test paper generation.

In conclusion, experiments have proved that the SSA-GA algorithm proposed in this paper can generate test papers for TEM-8 with higher quality and greater speed. Therefore, the SSA-GA algorithm can help English teachers generate TEM-8 papers intelligently. However, with Internet being widely used in every aspect of human lives, many students prefer to study English online. In the future, we will focus our study on combining the SSA-GA algorithm with Internet and building online test paper generation system via the SSA-GA algorithm.

## Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

This research was supported by Key Research Project of Humanities and Social Science of Universities in Anhui Province (no. SK2020A0226).

## References

[1] P. K. Liu and X. Han, "Comparative analysis on similarities and differences of hydrogen energy development in the World's top 4 largest economies: a novel framework," *International Journal of Hydrogen Energy*, vol. 47, no. 16, pp. 9485–9503, 2022.

[2] S. Georgiopoulos, A. R. Ahmadi, S. Do, M. Mokkas, and P. Partner, "UK Power Networks develop the world's first market-based approach to delivering efficient network constraint management," *CIRED - Open Access Proceedings Journal*, vol. 2020, no. 1, pp. 773–775, 2020.

[3] Y. Shi and H. Shi, "Practical teaching of business English majors based on intelligent machine teaching," *Computational Intelligence and Neuroscience*, vol. 2022, Article ID 3155292, 12 pages, 2022.

[4] Y. Y. Zhao, "Research on college English teaching evaluation based on neural network," *Mathematical Problems in Engineering*, vol. 2022, Article ID 7653986, 10 pages, 2022.

[5] H. Wu and Y. X. Yuan, "Interactive English online teaching system based on B/S model," *Advances in Multimedia*, vol. 2022, Article ID 8700304, 11 pages, 2022.

[6] Z. Y. Wu, T. He, C. J. Mao, and C. Huang, "Exam paper generation based on performance prediction of student group," *Information Sciences*, vol. 532, pp. 72–90, 2020.

[7] S. Poulik and G. Ghorai, "Applications of graph's complete degree with bipolar fuzzy information," *Complex and Intelligent Systems and Intelligent Systems*, vol. 8, no. 2, pp. 1115–1127, 2022.

[8] B. Yang, H. Xie, K. Ye, H. Qin, Z. Rong, and A. Liu, "Analysis of intelligent test paper generation method for online examination based on UML and particle swarm optimisation," *International Journal of Information and Communication Technology*, vol. 18, no. 3, pp. 317–333, 2021.

[9] S. El-Rahman and A. H. Zolait, "Automated test paper generation using utility based agent and shuffling algorithm," *International Journal of Web-Based Learning and Teaching Technologies*, vol. 14, no. 1, pp. 69–83, 2019.

[10] Z. H. Duan, H. L. Hu, G. S. Hu, F. Hao, X. D. Luo, and Y. Y. He, "Auto-generated test paper based on knowledge embedding," *International Journal of Information and Education Technology*, vol. 9, no. 9, pp. 671–677, 2019.

[11] H. Y. Zhang, X. Zhang, and W. Q. Zhang, "Method of rapidly generating test papers and quickly reducing papers based on random code and random numbers," *Engineering Intelligent Systems*, vol. 26, no. 2-3, pp. 65–73, 2018.

[12] M. L. Nguyen, S. C. Hui, and A. C. M. Fong, "Submodular memetic approximation for multiobjective parallel test paper generation," *IEEE Transactions on Cybernetics*, vol. 47, no. 6, pp. 1562–1575, 2017.

[13] D. V. Paul, "Automatic question paper pattern generation using GA approach," *INFOCOMP Journal of Computer Science*, vol. 19, no. 1, pp. 1–10, 2020.

[14] X. B. Lu, L. Z. Hao, and L. F. Yan, "Test paper materials auto-generation system," *Applied Mechanics and Materials*, vol. 484-485, pp. 808–811, 2014.

[15] J. Härtel, L. Härtel, and R. Lämmel, "Test-data generation for xtext: tool paper," *Lecture Notes in Computer Science*, vol. 171, pp. 342–351, 2014.

[16] J. F. Du, C. Liu, D. Cheng, and J. Cheng, "An ontology-based system for generating mathematical test papers," *Communications in Computer and Information Science*, vol. 480, pp. 237–244, 2014.

[17] C. Li, J. Q. Yany, Y. H. Wang, and S. R. Zhang, "An improved backtracking search paper - generating algorithm," *Fire Control and Command Control*, vol. 44, no. 9, pp. 144–148, 2019.

[18] L. I. Ruisen, S. Zhang, G. Yi, L. Xue, and J. Tan, "Research on the multiple associated engineering graphics test database with multi attribute and the multi-mode intelligent test paper generating system," *JOURNAL OF GRAPHICS*, vol. 39, no. 2, pp. 373–380, 2018.

[19] J. D. Watson and F. H. C. Crick, "Optimizing intelligent-generating test paper model by improved genetic algorithm," *Journal of Huazhong University of Science and Technology*, vol. 41, no. 5, pp. 82–85, 2013.

[20] Y. H. Lu and H. Liu, "Auto-generating examination papers based on integer coding and adaptive genetic algorithm," *Jisuanji Gongcheng/Computer Engineering*, vol. 31, no. 23, pp. 232-233, 2005.

[21] S. Y. Guan, D. Y. Liu, D. Jin, X. H. Wang, and K. Su, "Genetic algorithm with local search for automatic test paper generation," *Journal of Jilin University (Science Edition)*, vol. 47, no. 5, pp. 961–968, 2009.

[22] S. Poulik and G. Ghorai, "Estimation of most effected cycles and busiest network route based on complexity function of graph in fuzzy environment," *Artificial Intelligence Review*, vol. 55, no. 6, pp. 4557–4574, 2022.

[23] S. Poulik and G. Ghorai, "Empirical results on operations of bipolar fuzzy graphs with their degree," *Missouri Journal of Mathematical Sciences*, vol. 32, no. 2, pp. 211–226, 2020.

[24] K. H. Peng, Y. F. Huang, and J. M. Yao, "Comparison of automatic test paper generation for database technology courses of various artificial intelligence algorithms," *Computer Systems and Applications*, vol. 27, no. 3, pp. 210–216, 2018.

[25] M. Z. Hu, *Research and Design of the Platform for Information Security Competition Based on Auto-Generation Test Paper Algorithm*, Beijing University of Posts and Telecommunications, Beijing, China, 2018.

[26] J. Meng and Y. H. Wang, "The design of intelligent test paper generation for English test based on genetic algorithm," *Theory and Practice of Education*, vol. 41, no. 33, pp. 61–64, 2021.

[27] S. Poulik and G. Ganesh, "Determination of journeys order based on graph's Wiener absolute index with bipolar fuzzy information," *Information Sciences*, vol. 545, pp. 608–619, 2021.

[28] S. Poulik and G. Ghorai, "New concepts of inverse fuzzy mixed graphs and its application," *Granular Computing*, vol. 7, no. 3, pp. 549–559, 2022.

[29] J. Xue and Bo Shen, "A novel swarm intelligence optimization approach: sparrow search algorithm," *Systems Science and Control Engineering*, vol. 8, no. 1, pp. 22–34, 2020.

[30] R. Thenmozhi, A. W. Nasir, V. K. Sonthi et al., "An improved sparrow search algorithm for node localization in WSN," *Computers, Materials and Continua*, vol. 71, no. 1, pp. 2037–2051, 2022.

[31] A. Fathy, T. M. Alanazi, H. Rezk, and Y. Yousri, "Optimal energy management of micro-grid using sparrow search algorithm," *Energy Reports*, vol. 8, pp. 758–773, 2022.