

The Research and Solutions to the Multi-constrained Concurrent Optimization Problem in Automatic Test Paper Tasks

Xin Zhang, Mianrong Yang

Network Management Center Xinxiang University, Xinxiang, Henan Province, China

Computer and Information College Xinxiang University, Xinxiang, Henan Province, China

zx@xxu.edu.cn ,123ymr@163.com

Keywords: Test Group Papers, Genetic Algorithm, Multi-Constraint Solving.

Abstract. The problem of the multi- constrained concurrent optimization is a difficult artificial intelligence one that exists widely in the real life and confuses humans for years. Such as: the operating frequency of planes and trains, the course arrangement in school and the test paper grouping etc. This paper adopts the improved genetic algorithm, takes the test paper grouping problem for example, and puts forward a solution to the multi-constrained concurrent optimization problem. Finally, the paper aims to confirm the feasibility of this program through the experimental data.

The Introduction

With the advent of the internet age, the colleges and the departments of government are more and more in need of network test system in various types of Title Examination, Proficiency Test, Course Examination. Especially because of the rapid development of Distance Education in recent years, the network test system has becomes one of the hot spots in the networks researches and applications. The core content of the exam system is how to make the computer automatically extract the more satisfying test according to the requirement parameters set by the users, in order to combine better papers.

The background

Information Before the realization of test paper strategy, we have to understand some auxiliary knowledge, so that we can understand why such a test paper can be refined by our strategy.

First of all, we need to understand what kind of paper is a good paper. That is what constraints a good paper should satisfy. From the point of view of an overall paper, the test paper model is the distribution of some parameters. The process of the computer test paper is to select the more appropriate questions from the test database to meet the distribution of these parameters. Each distribution can be seen as a constraint needed to be satisfied, the multiple distributions can be seen as multiple constraints, so the process of test paper is to select a set of questions from the database, and allow the collection to satisfy the multiple constraints. Basically, the constraints of test paper contain the following aspects^[1]:

1) Constraints on knowledge point. It can be seen as constraints on range of chapters, including the percentage share of the chapters on the total papers.

2) Constraints on question type. It refers to the types of questions included in the paper, that is, to test by what type of test paper.

3) Constraints on question quantity. It means the quantity of questions included in the papers, when it is specific to each type of question, it refers to the quantity of each type. The constraint is positively correlated with the answer time constraints.

4) Constraints on answer time. It refers to the longest time to finish all the questions on the test paper.

5) Constraints on the difficulty. It refers to the overall difficulty degree of the paper.

6) Constraints on the discrimination. It refers to the ability of the papers to distinguish the

testees.

7) Constraints on the exposure. It refers to the times of questions have been tested, generally speaking, the better the lower of the exposure value. It is likely to assure the confidentiality and effectiveness of the test paper.

8) Constraints on the cognitive level. It is divided into six parts, such as: memorizing, understanding, application, analysis, synthesis and evaluation.

Genetic Algorithm and Test Paper Strategy

Genetic Algorithms is a kind of optimization method based on the natural selection and population genetics. It is a kind of calculation model simulating Darwin's genetic selection and the process of natural selection in biological evolution, which is first put forward by Professor John H. Holland, professor of the University of Michigan in the United States in 1975. It adopts the natural selection mechanism of "survival of the fittest", selects a more satisfying individual from a group, and adjusts the individuals continue to move toward the most optimal solution by the operation of selection, crossover and mutation etc.

The establishment of chromosomal papers. It is necessary to establish the appropriate chromosomal papers so as to apply Genetic Algorithms to the test paper. The establishment of the papers chromosome is to determine a way to turn the question record in the test database into genotype code according to the characteristics of test database^[2]. This paper adopts the modified binary code, assuming the number of some type of exam questions is L , it can be represented by a binary bit string which length is L , each of the value is 0 or 1, 0 means not to choose some question, 1 means to choose some question. So, a certain type of papers question corresponds to a 0, 1 constitutes of a binary string of L length, which is called sub- chromosome, by this way, the code of a sub- chromosome is formed. The test paper chromosome is made up of some sub-chromosome segments, each sub-chromosome represents a type of questions. For example, if the test paper needs three types of questions, they are multiple choice, blank-filling questions and programming questions, then the paper chromosome will be made up of three binary bit string sub-chromosomes.

Chromosome evaluation function. Chromosome evaluation function can also be referred to as the fitness function, which is a measure of the quality of solutions. It aims to detect whether the test papers represented by a chromosome can better meet a variety of constrains, it can also be understood that whether the chromosomes are more suitable to the judgment index of the natural environment^[3].

The problem of a test paper is a solution to the multiple constraints, these constraints can be transformed into a paper target, through the way of the separate values of each parameter in the model test paper, so, and the value of the parameter limit can be qualified as the multiple targets to the realization of test paper. For example, if the score value ($v_D = i$) of the required paper difficulty level is represented by a_i ($i = 1, 2, \dots, n$), and n stands for the total number of the paper difficulty level.

Therefore, the total score of the paper is MF , it is obviously that,
$$\sum_{i=1}^n a_i = MF .$$

After being processed, the constraints were transformed into part of the objective function. Assuming da_i is the deviation (absolute difference) between the actual total score and the required total score, then $\sum_{i=1}^n da_i / n$ is the average deviation of between the actual score and predetermined score in each difficulty level of the currently selected items. The average deviation reflects the difference degree between the ratios of each selected items and the predetermined value. The smaller the average is, the more satisfying the test paper is, also the smaller the error is. On the country, it means that the selected questions are far from being satisfied with the requirements and the errors are bigger.

Similarly, there are such constrains on type of questions, chapter, time and score. It can be understood in this way, if b_i represents the value of a constraint needed to be satisfied, then db_i

represents the error between the corresponding actual value and required value under the constraints, $\sum_{i=1}^n db_i / n$ represents the average deviation. This constraint can also be transformed into part of the targeted function by this way. The average value reflects the difference degree between the selected questions' actual value and predetermined value. The smaller the average is, the more satisfying the test paper is, also the smaller the error is. On the contrary, it means that the selected questions are far from being satisfied with the requirements and the errors are bigger.

Through the above approaches, the average deviations can be treated as the separate targeted components, the targeted function is thus formed by combining them together. Because the test paper requires multiple constraints, and the constraints usually restrict each other, so it is hard to satisfy them all at the same time, meanwhile, when a test paper is composed, there are different requirements for the constraints, that is, some constraints require priority to be satisfied, others may be appropriate to relax the restrictions. Therefore, when these target components are combined, each should be allocated a certain weight, indicating their importance. Assuming that the weights of each component are: p_1, p_2, \dots, p_n ($p_1 + p_2 + \dots + p_n = 1, 0 \leq p_i \leq 1$), then, if the components are put together, the problem of the evaluation of the papers turns out to be the minimizing of the targeted fitness function as shown in the following (1):

$$\text{fitness} = p_1 \left[\left(\sum_{i=1}^l |da_i| \right) / l \right] + p_2 \left[\left(\sum_{i=1}^m |db_i| \right) / m \right] + \dots + p_n \left[\left(\sum_{i=1}^p |dg_i| \right) / p \right] \quad (1)$$

Determination of the initial population. The initial population that is made up of N individuals is generated randomly. The individual generates a new generation of groups by the role of genetic operators. N is an important parameter in genetic algorithm, which stands for population size and has a major impact on efficiency of computational algorithm. N needs to obtain an appropriate value, on the one hand, if it is too small, the search is not efficient by dealing with a small number of chromosomes at one time, also it is likely to fall into the local optimal solution, on the other hand, if it is too large, that means the fitness value of each generation is too big, the efficiency of the computational algorithm is low^[4].

This article takes the requirements for the most basic kinds of questions as the initial constraint, resulting in a series of individuals to meet these requirements. As for the algorithm, if it can not meet this requirement, then it will be re-randomized within a limited selection until to find the individual which can meet the requirements for the species number.

Selection Operator. Operator Selection aims to select the chromosomes from the group in accordance with certain rules and probability. The choice of chromosome purpose is to find the adaptability relatively strong groups of individuals allowed to participate in genetic, so their offspring can have strong adaptability. The specific method is to make use of the fitness of the fitness function to the fitness of the chromosome calculation, and then it will selectively replicate the better from the initial populations according to the rule of "survival of the fittest", the replication is the basic operators of genetic algorithm, it allows the best individual to breed in the new group of the next generation, whether an individual is replicated based on its fitness, if the fitness is better, it will be replicated, otherwise it will be eliminated. The result of cross copy and selection is to keep the total number of individuals in the new group equals to that in the original group. This paper adopts the roulette wheel selection method as described in the following^[5]:

To calculate the fitness of each individual within the group and accumulate their countdown's value, and then to obtain the fitness of the corresponding accumulation of the chromosome, the last accumulation is the cfitness (last size).

1) In the interval of $[0, \text{cfitness}(\text{last size})]$, the random number R which is uniformly distributed is generated.

2) To compare cfitness in turn with R , the first occurring cfitness that is greater than or equal to R is chosen as the copy object.

3) Repeat 2), 3) until the needed number of individuals is obtained.

Among them, the cumulative fitness is (2):

$$cfitness (Gen_i) = 1 / fitness (Gen_i) + cfitness (Gen_{i-1}) \quad (2)$$

On the surface, replication of individual choice is random. But the Choice is based on the total value difference of two adjacent cumulative fitness ΔS_i , $\Delta S_i = S_i - S_{i-1} = f_i$. In the formula, f_i is the reciprocal of the fitness of the individual number i . Because the more outstanding of an individual gene should be the smaller value of fitness, the greater of its f_i , the greater the distance ΔS_i , the greater the chance for the random number to fall in the intervals, the more likely for the individual numbers i to be chosen. Roulette wheel selection method not only embodies the "survival of the fittest" principle, and also maintains a variety of individual states.

Crossover Operator. The main features of the genetic algorithm are the use of the crossover operator. On the one hand, it maintains the characters of the excellent individuals in the original group to some extent; on the other hand, it allows the genetic algorithm to explore the new space, so that the individuals in the new group gain the diversity^[6]. This paper adopts the method of crossover operation through the single-point cross-operator. Assuming that the crossover probability is PC , a random individual number between $[0,1]$ is generated according to i , if the individual i is less than the corresponding random number PC , then it is chosen as one of the cross parent. After the cross, it may produce infeasible solutions, in order to ensure that the new generations can still satisfy the constraints, this paper has improved the single break point crossover operator, that is, and the cross takes place in the same kinds of question section. an still satisfy the original constrains.

Mutation Operator. The basic content of mutation operator is to adjust some gene values on the position of some individual strings of the group. The use of mutation operator in genetic algorithm qualifies GM the better ability of global random search, and maintains the diversity of the group^[7]. Mutation is another kind of method in GM to produce the new individuals. The selection of the mutation individual and the determination of position are generated by random. Assuming that the mutation probability is P_m , for each random number between $[0,1]$ generated according to the number i , if the number is less than P_m , then it takes mutation operation here.

The mutation operator adopted in this paper has been improved based on this theory, that is, it enforces the double-mutation within the same kinds of question paragraph. The advantage is that protect the already relatively well-chromosome structure from being destroyed. Variation after, to ensure that the chromosome the gene is still better meeting the current constraints, the theory is shown in the following Fig. 1:

The judgment of the stop conditions. If there is an optimal individual in the group, or the pre-set stop conditions have been satisfied, it shuts down after the optimal solution output, otherwise it continues to the next step, and starts a new round of genetic role. Since the genetic algorithm does not make use of the gradient and other information of the objective function, so in the process of evolution, it can not determine the location of the individual in the solution space. Therefore, it is unable to determine the convergence of the algorithm through the traditional methods and terminate the algorithm. The commonly-used approach is to pre-set a maximum evolution algebra or algorithm, and terminates when there is no significant improvement in the fitness value after many continuous solution generations. In this paper, the stop condition is that when the fitness value of the solution has no significant improvement or has achieved the expected value of the fitness parameters.

The analysis of test paper parameters and test results

We have already understood the operation strategies by using the genetic algorithms to solve the multi-constrained test paper questions, then what have mentioned in the strategies such as the parameters of population size N , crossover probability P_c and mutation probability P_m , what number should these 3 related parameters be set to, it is illustrated through the following tests:

1) Let the crossover probability $P_c = 0.5$, mutation probability $P_m = 0.1$, to maintain the same values of P_c and P_m , and take different values for parameters of population size N for 20 test paper tests respectively, the results are as shown in the following table1.

2) Let the population size parameter $N = 150$, mutation probability $P_m = 0.1$, to maintain the

value of N and P_m , and take different values for crossover probability P_c for 20 test paper test respectively, the results are as shown in the following table2.

3) Let the population size parameter $N = 150$, mutation probability $P_c = 0.5$, to maintain the value of N and P_m , and take different values for mutation probability P_m for 20 test paper test respectively, the results are as shown in the following table3.

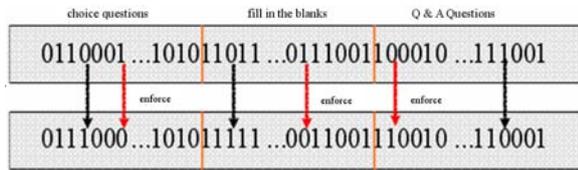


Fig. 1 Schematic diagram of the process of forced mutation

Table 1 Population size parameters of the impact of N

Parameters of population size N	The average evolution algebra	The number of successful test paper	average fitness value
20	26.1	11	0.32
40	15.5	17	0.19
80	12.2	19	0.11
100	10.3	20	0.09
150	8.9	20	0.06
200	7.9	20	0.06

Table 2 Crossover probability of P_c can

crossover probability P_c	The average evolution algebra	The number of successful test paper	average fitness value
0.1	8.1	17	0.29
0.2	8.5	20	0.21
0.3	8.9	20	0.11
0.5	9.3	20	0.06
0.7	9.9	20	0.15

Table 3 Mutation probability of P_m on the test paper

mutation probability P_m	The average evolution algebra	The number of successful test paper	average fitness value
0.001	8.2	20	0.09
0.01	8.5	20	0.09
0.05	8.3	20	0.08
0.1	8	20	0.06
0.15	9.2	20	0.10

Conclusion

Through the test results it can be seen that if the value of population size parameter N is too small, the program will stop at the immature stage, otherwise it will result in the lower efficiency of the implementation process by making the search space too large. Whether the crossover probability P_c and mutation probability P_m are too large or too small, it will cause the average fitness to head for the negative direction. According to the test we can see that when $N = 150$, $P_c = 0.5$, $P_m = 0.1$, it is likely to get the better fitness papers.

References

- [1] Zhouqian. Educational Evaluation and Statistics[M], Beijing Science Press, 1998: pp.224-252. (in Chinese)
- [2] Lixiaoyong, Wangying. Q & Automation Management System Algorithmic[J], Northwest Normal University(Natural Science), 2002, vol.38, no.4, pp.41-43. (in Chinese)
- [3] Zhangxing, Liaoqimei. Method of constructing the fitness function in automatically question algorithm group[J], Computer Engineering and Science, 2007, vol.29, no.1, pp.89-90.(in Chinese)
- [4] Hedakuo, Wangfuli, Jiamingxing. The uniform design of Genetic algorithm initial population of and operating parameters[J], Northeastern University(Natural Science), 2005, vol.26, no.9, pp.830-831. (in Chinese)
- [5] Helio, J. C. Barbosa and Afonso, C. C. Lemonge. "A new adaptive penalty scheme for genetic algorithms" [J], 2003, 156: pp.215-251 Information Sciences.
- [6] Sunchunyan, The study of Intelligent Grouping System based on technology of Integer coded genetic algorithm[J], Huaihua University, 2009, vol.28, no.11, pp.39-40. (in Chinese)
- [7] Zhang chen, Zhan zhihui, Comparison of genetic algorithm selection strategy [J], Computer Engineering and Design. 2009, vol.30, no.23, pp.5471-5472. (in Chinese)