

An Adaptive Cellular Genetic Algorithm Based on Selection Strategy for Test Sheet Generation

Ankun Huang¹, Dongmei Li^{1*}, Jiajia Hou² and Tao Bi¹

¹*School of Information Science and Technology, Beijing Forestry University, Beijing 100083, China*

²*School of Information, Renmin University of China, Beijing 100872, China*
lidongmei@bjfu.edu.cn

Abstract

Intelligent test sheet generation is a multi-objective constrained optimization problem. Genetic algorithm based on groups search strategy can provide a better solution for multi-objective optimization. Traditional genetic algorithm in test sheet generation process has many drawbacks, such as poor convergence, low fitness and high exposure times. To solve these problems, this paper proposes an adaptive cellular genetic algorithm based on selection strategy. Selection strategy can adaptively determine candidate test items set and the conceptual granularities according to the desired concept scope. Then, a new cellular population is formed by candidate test items. After evolution by the rule, genetic algorithms are executed. The experimental results show that the proposed algorithm gets rid of tests that do not meet the requirements which can reduce knowledge related errors, lower the exposure of tests, and increase the possibility of escape from local optima. In general, the algorithm proposed in this paper effectively improves the convergence speed as well as generates test papers more in line with people's demands.

Keywords: Cellular Automata, Genetic Algorithm, Intelligent Test Sheet Generation, Selection Strategy

1. Introduction

With the rapid developments of information and assessment technology, it is necessary to assess and diagnose learners' learning status by utilizing the computerized testing which can effectively analyze examinees' abilities and learning barriers. The test quality offered by a computerized testing system does not only depends on the quality of test items but also on a satisfied test sheet which meets the various assessment parameters requirements, such as exam time, degree of complexity, chapters, exposure times, test ID, scores and associated concepts [1]. Therefore, how to efficiently assist teachers in generating an appropriate test sheet which meets diverse assessment requirements has become an important research issue.

Test sheet generation is a multi-objective constrained optimization problem, which is a proven NP-hard problem [2]. Intelligent test sheet generation algorithm generally uses the following categories: maximum weight algorithm, analysis stochastic selection law, recollection trial method, and genetic algorithm [3]. Among the algorithms mentioned above, the genetic algorithm is the most commonly used algorithm in test sheet generation. The Genetic Algorithm (GA), a calculation model simulating natural selection and biological evolution process, has strong robustness and optimization function [4]. It has basic characteristic of achieving population diversity and global search by maintaining the potential solution between generations. The group search strategy of the genetic algorithm provides a better solution for multi-objective optimization. In recent years, GA for test sheet generation has made a good progress.

Hwang designed a mixed integer programming model and proposed approximate solutions by using the GA [5]. This algorithm can be efficient for large-scale tests. However, it cannot solve multi-objective constrained optimization problem efficiently and doesn't meet various assessment requirements. To solve this problem and improve genetic algorithm in test sheet generation, Huang presents an application of adaptive genetic algorithm [6], which effectively improved the speed and degree of convergence. However, there is high degree of error associated concept and unreasonable items distribution. For reasonable selection strategy and multi-objective constrained optimization problem, Lin proposed Adaptive Test Sheet Generation (ATSG) mechanism, consisting of a Candidate Item Selection Strategy (CISS) and an Aggregate Objective Function (AOF) [1]. CISS process can adaptively determine candidate test items set and the conceptual granularities according to the desired concept scope, and AOF applies GA algorithms to solve the mixed integer programming problem. However, it has slow convergence speed and optimization speed.

A novel cellular genetic algorithm (CGA) is innovatively brought forth for test sheet generating [7]. The novel approach can effectively improve the convergence rate, and further improve the convergence in intelligent test paper. After joining the associated concepts, it has unqualified items in the test sheet and high exposure times. A high-item exposure rate will decrease the qualities of test items and Item Bank Security [8].

Based on ATSG and CGA, this paper proposes an Adaptive Cellular Genetic Algorithm based on Selection Strategy (ACGASS). This method does better than ATSG and CGA in aspects of global convergence and knowledge related errors.

2. Selection Strategy Determination of Concept Correlation Degree

In order to effectively understand students' problems in learning, it is generally required to make different ratio settings in concept granularity, discrimination and item difficulty of papers [9,10]. For example, when examining students' mastery of overall knowledge, concept granularity settings should be lower in order to include more knowledge points, and item difficulty settings should be lower with emphasis on the students' understanding of basic knowledge. In several researches, paper's discrimination should be a higher degree in order to involve more non-repetition of knowledge in the case of multiple test sheet generation. When examining students' mastery of certain sections and the degree of flexible use, concept granularity and item difficulty settings should be higher, while paper's discrimination should be lower. Therefore, how to adaptively determine the concept granularity of item bank before test sheet generation becomes the primary problem of intelligent test sheet generation.

Assume that a set of test items should be selected from item bank Q including n test items as test items of candidate item bank θ . Q is defined as:

$$Q = \{Q_1, Q_2, Q_3, Q_4 \dots Q_n\} \quad (1)$$

Each test item Q_i of Q is defined as:

$$i = (t_i, d_i, R_{ik}, e_i, s_i) \quad (2)$$

Where, t_i denotes the time of finishing the item; d_i denotes the difficulty degree of the item; R_{ik} denotes the correlation degree between the item and concept C_k ; e_i denotes the exposure times of the item; s_i denotes the scores of the item.

This paper builds a multi-branched tree H of the concept granularity, and every tree node consists of concepts. The higher layers a tree node is in, the smaller concept granularity is, which means this tree node covers greater range of knowledge. The lower layers a tree node is in, the larger concept granularity is, which means this tree node covers smaller range of knowledge and represents more specific concept. The leaf node

is the most specific concept. Every leaf node is associated with one concept test item Q at least, where correlation degree is denoted by R_{ik} in formula (2).

Assume that the concept granularity tree H contains m concepts C as the tree nodes, namely, $C = \{C_1, C_2, C_3, C_4, \dots, C_m\}$.

Φ is nonleaf descendent function, then

$$\Phi(C_i) \subset C \text{ iff } \exists j, (C_j \in \Phi(C_i)) \wedge \neg \exists k (C_k \in C_j) \quad (3)$$

Where $\Phi(C_i)$ is a set of descendent nodes of C_i . α is set as leaf nodes of tree H, and then $\alpha(C_i) \in C$ which represents leaf node concepts of C_i .

$$C_j \in \alpha(C_i) \text{ iff } (C_j \notin \Phi(C_i)) \wedge (C_j \in C_i) \quad (4)$$

Where C_j belongs to $\alpha(C_i)$ if and only if C_j is not a nonleaf descendent concept and a leaf concept of H.

R_{ij} represents the correlation degree between each test item Q_i of item bank Q in formula (1) and concepts in collection C_j as well as in formula (2).

Let C_0 denote the target item concept bank and C' denote the concept relevance bank of Q_i

$$C' \in C_j, C_j = \{C_1, C_2, \dots, C_n\} \quad 0 < R_{in} < 100 \quad (5)$$

First of all, the target item concept bank C_0 should be filtered according to formula (6) to form concept bank C.

$$C_i \in C_\alpha \text{ iff } (C_i \in \alpha(C_0)) \wedge (C_i \in C) \quad (6)$$

Then, items Q_i meeting the requirements should be selected out from C_α and added into candidate test item bank θ .

$$Q_i \in \theta \text{ iff } \exists j ((C_j \in C_\alpha) \wedge (R_{ij} > 0)) \wedge \neg \exists k (C_k \in C_\alpha \wedge (R_{ik} > 0)) \wedge (R_{ij} = \text{Max}(R_{ij})) \quad (7)$$

3. Cellular Automata and Mathematical Model of Test Sheet Generation Problem

3.1. Principle of Cellular Automata

Cellular automata, a kinetics model discrete both in time, state and space [11], can eventually produce complex behavior through some basic evolution rules. The most basic cellular automata consists of four parts: cellular, cellular space, neighbors and evolution rule, which can be expressed as

$$A = (L_r, S, N, f); \quad (8)$$

Descriptions of symbols in formula (8) are as follows:

A: Entire set of cellular.

L_r : Cellular space. The subscript r represents the spatial dimension, and r can be any positive integer.

S: The finite set of cellular states. Each cellular only has one state in every moment. S can have the value $\{1, 0\}$, where 1 represents a "live" state of the cellular and 0 represents a "dead" state of the cellular.

N: A set of all cellular within an area.

f: Evolution rule, namely a state transition function, denoted by:

$$f: s_i^{t+1} = f(s_i^t, s_n^t). \quad (9)$$

Evolution rule is the core of the cellular automaton. Via evolution rule, cellular can form different states after several generations of evolution. For two-dimensional cellular, after several generations of evolution, its initial state in the evolution rule tends to

become four types: smooth, cycle, chaos and complex. The most commonly used evolution rule is the typical rule "life game" developed by the famous mathematician Conway. In the rule of "life game", cellular's state can only be "1" or "0", namely "live" or "death". However, the approach of this rule is relatively simple, so this paper adopts an improved evolution rule proposed in [7].

3.2. Mathematical Model of Test Sheet Generation Problem

A test sheet consists of m test items, and each test item has n attributes. Each test item can be represented by an n -dimensional vector l , as shown below:

$$l = (a_1, a_2, a_3 \dots a_n) \quad (10)$$

This paper considers seven attributes: question number, difficulty, chapter, time, exposure, scores and concept correlation degree. a_1 represents question number of test item. a_2 represents difficulty of test item. a_3 represents chapter of test item. a_4 represents time of test item. a_5 represents exposure of test item. a_6 represents scores of test item. a_7 represents concept correlation degree of test item.

The structure of a test sheet can be described with an $m \times n$ matrix, as shown in formula (11).

$$M = \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} & \dots & a_{1n} \\ a_{21} & a_{22} & a_{23} & a_{24} & \dots & a_{2n} \\ a_{31} & a_{32} & a_{33} & a_{34} & \dots & a_{3n} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{m1} & a_{m2} & a_{m3} & a_{m4} & \dots & a_{mn} \end{pmatrix} \quad (11)$$

where, the first column of the matrix generally indicates the number of questions, and $a_{ij}(i=1,2,3 \dots, m; j=1,2 \dots, n)$ represents j -th attribute of test item in i -th row.

4. Author Name(S) And Affiliation(S)

Author names and affiliations are to be centered beneath the title and printed in Times New Roman 12-point, non-boldface type. Multiple authors may be shown in a two or three-column format, with their affiliations below their respective names. Affiliations are centered below each author name, italicized, not bold. Include e-mail addresses if possible. Follow the author information by two blank lines before main text.

4. Adaptive Cellular Genetic Algorithm Based on Selection Strategy

4.1 Determination of Cellular Space and Coding Scheme

Cellular space uses two-dimensional space, that is the spatial dimension $r = 2$. Each cellular are distributed in the two-dimensional space, adopting 8-neighbor model. For the boundary cellular, according to the method proposed in [11,12], up and down, left and right connection way is used to make the entire cellular space form a topological torus structure, which is good for improving the diversity of the population. Evolution rule in [7] is adopted, as shown in formula (12) and (13).

$$0 \neq 2, 3 \quad \left\{ \begin{array}{l} 1 \quad s=2, 3 \\ \text{If } s^t=1, \text{ then } s^{t+1}= \end{array} \right. \quad (12)$$

$$1 \quad s=3 \quad \left\{ \begin{array}{l} \text{If } s^t=0, \text{ then } s^{t+1}= \\ 0 \quad s \neq 3 \end{array} \right. \quad (13)$$

where s^t represents the states of cellular at time t . s^{t+1} represents the states of cellular at time $t+1$. s represents the survival number of cellular neighbors at time t .

Under normal circumstances, encoding method of genetic algorithm adopts binary encoding or decimal encoding. However, in the case of abounding sample groups, binary encoding will cause the length of chromosome extremely long after encoding, which will lead to a long period of time waste whether in coding or decoding and a massive space cost. Therefore, there is no doubt that decimal encoding is the best choice.

4.2. Parameter Settings

① Determine sample size. In order to ensure the diversity of population, the sample size determined in this paper is 625.

② Determine selection rate. Selection is to select individuals from each generation that satisfies the required fitness value into the next generation for breeding. This paper adopts probabilistic methods proportional to fitness value in [12] to select, so individuals with high fitness can be easily selected to breed.

③ Determine the crossover and mutation rates. This paper uses the results calculated by Lu [13], that is, crossover rate is 0.6, and mutation rate is 0.1.

4.3. Determination of the Fitness Function

Test sheet generation sets seven attributes for each question, namely question number, difficulty, chapter, time, exposure, scores and concept correlation degree, expressed as follows by vector g :

$$g = (a_1, a_2, a_3, \dots, a_7) \quad (14)$$

Suppose there are n questions of a test sheet, which determines a $n*7$ matrix q , as shown in formula (15).

$$q = \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} & \dots & a_{17} \\ a_{21} & a_{22} & a_{23} & a_{24} & \dots & a_{27} \\ a_{31} & a_{32} & a_{33} & a_{34} & \dots & a_{37} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{n1} & a_{n2} & a_{n3} & a_{n4} & \dots & a_{n7} \end{pmatrix} \quad (15)$$

Where c_i 、 x_i 、 e_i ($i=1,2,\dots,m$; m represents the number of chapters) respectively represent fraction of each chapter required by users, fraction of each chapter in actual generated test sheet, and errors of each chapter users allow.

Fitness function F is calculated specifically as follows:

First of all, function f_1 is determined:

$$f_1 = \sum_{i=1}^n a_{i2} a_{i6} \quad (16)$$

Function f_2 is determined:

$$f_2 = \sum_{i=1}^n a_{i4} a_{i6} \quad (17)$$

Function f_3 is determined:

$$f_3 = \sum_{i=1}^n a_{i5} a_{i6} \quad (18)$$

Function f_4 is determined:

$$f_4 = \sum_{i=1}^n a_{i7} a_{i6} \quad (19)$$

Function d_1 is determined:

$$d_1 = f_1 / 100 - dif \quad (20)$$

Where dif is the difficulty value users require, ranging in (0,100).

Function d_2 is determined:

$$d_2 = f_2 / 100 - time \quad (21)$$

Where $time$ is the time value users require, ranging in (0,100).

Function d_3 is determined:

$$d_3 = f_3 / 100 - p \quad (22)$$

Where p represents exposure degree users require, ranging in (0,100).

Function d_4 is determined:

$$d_4 = f_4 / 100 - concept \quad (23)$$

Where $concept$ represents concept correlation degree users require, ranging in (0,100).

Function d_5 is determined:

$$d_5 = \sum_{i=1}^m h_i \quad (24)$$

Function h_i in formula (24) is determined:

$$h_i = \text{Max}(|c_i - x_i| - e_i, 0) \quad (25)$$

The objective function H is determined:

$$H = (d_1 + d_2 + d_3 + d_4 + d_5) / 5 \quad (26)$$

It can be seen from the objective function H in formula (26) that the value of objective function H ranges in (0,100). The smaller its value is, more consistent the generated test sheet is with users' requirements. In general, the fitness of individuals should be the larger, so the target is transformed as follows:

$$H = 100 - H \quad (27)$$

4.4. Steps of Adaptive Cellular Genetic Test Sheet Generation Algorithm Based on Selection Strategy

Step 1. Based on adaptive selection strategy given in section 2, test items in line with the requirements are selected from item bank Q and added to candidate item bank θ .

Step 2. Initialization. The initial population A is randomly generated. Total amount of sample in population A is T , and each sample is encoded in decimal.

Step 3. Select appropriate data structure to simulate 8-neighbor model. All the individuals of cellular set A are randomly arranged in the topological torus structure.

Step 4. Calculate each cellular's fitness value in cellular space using fitness function H . The initial state of cellular, "1" or "0" is determined and added into evolution rule shown in formulas (12) and (13) to evolve.

Step 5. Calculate fitness value of contemporary cellular, select samples meeting the requirements, and add them to set S .

Step 6. Individuals selected from set S in order do every two cross to get new individuals, and then calculate its fitness value using the fitness function H. If the value is larger than the corresponding female parent's fitness value, the female parent's genetic information will be replaced by the new individual's genetic information, otherwise not replaced.

Step 7. Sample variability. After variability, calculate the new individual's fitness value, using fitness function F, and then the new one is added to cellular space to replace the original one.

Step 8. Determine whether the maximum evolution algebra is achieved. If it is achieved, output the individual which has the maximum cellular fitness value in current cellular space, otherwise, go to step 4.

5. Second and Following Pages Experiment and Analysis

5.1. Experimental Methods

Six kinds of experimental methods have been conducted in this paper in order to evaluate the effectiveness of the proposed algorithm. The former three kinds of methods evaluate ATSG, CGA and ACGASS algorithms without the limitation of concept correlation degree, randomly selecting all the test items from the original item bank Q, which is called no selection strategy method. While the later three kinds of methods evaluate ATSG, CGA and ACGASS algorithms with the limitation of concept correlation degree, which is called the method based on selection strategy.

During the test sheet generation, six kinds of experimental methods use the same test sheet generation parameters. Because of the reason that it is difficult to estimate the final range of the fitness value of test papers, this paper uses a fixed genetic algebra. After several generations of genetic, the maximum fitness value of the sample population is chosen as the judgment standard to compare the fitness value of different methods. In order to get more accurate results, the operation is repeated 50 times for each set of test sheet generation parameters. Finally, the results includes average difficulty of test papers, average concept correlation degree, average exposure, average fitness and average time of test sheet generation of the six kinds of experimental methods respectively.

Five kinds of gradient containing different amounts of test items, namely 500,1000,1500,2000 and 2500, is applied to item bank, totally involving the knowledge of six chapters. Sample size is 625, and the maximum evolution algebra is 1000. For each sample paper, its full marks are 100, having a total of 10 questions, including four types of questions, which are two multiple choice, two fill in the blank, two short answer, four algorithm design question, 10 points for each question. Attributes of each question are generated by random function.

Test sheet generation parameters are set as follows: the difficulty requirement is 52; examination time is 120 minutes; scores of each chapter are 10,10,20,10,30,20; chapter error is limited to 10; concept correlation degree is 50 and exposure's upper limit is 5.

5.2. Analysis of Experimental Results

For six kinds of experimental methods, this paper respectively conducts 50 times optimization operations, and gets the results of corresponding methods, including difficulty, concept correlation, exposure, fitness and time of test sheet generation.

In order to compare and analyze the number of test items in test papers that don't meet the concept requirements, six kinds of experimental methods are divided into two categories, selection strategy and non-selection strategy. As a result, we get the average number of test items that don't meet the concept requirements under various test scales, which is shown in Figure 1.

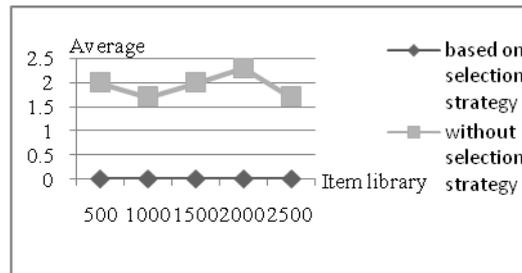


Figure 1. Average Number of Test Items that don't Meet the Requirement

As can be seen from Figure 1, in the non-selection strategy algorithms, there are test items in test papers that don't meet preset concept requirements, while in the algorithms based on selection strategy, all the test items are in line with preset concept requirements. The reason to explain it is that selection strategy can adaptively set requirements based on concept to select the corresponding candidate item bank.

In order to analyze the errors of concept correlation, this paper respectively takes the same selection strategy and non-selection strategy algorithm for ASTG, CGA and ACGASS to do summation operation of concept correlation under the same test scale, and then calculate its average value, which is compared with preset concept correlation to get the concept correlation errors. Comparison results of the errors are shown in Figure 2.

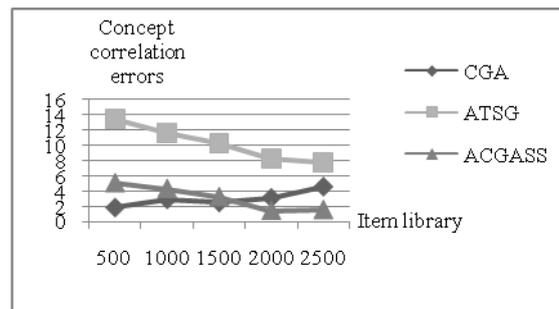


Figure 2. Concept Correlation Errors Under the Same Strategy

As can be seen from Figure 2, with the increasing of test scale, concept correlation error of algorithm AGASS and ATSG gradually decreases, which is good to generate test sheet more precisely satisfying the concept correlation requirements of users. Meanwhile, the error of ACGASS is obviously less than that of ATSG algorithm. CGA algorithm demonstrates the superiority in small-scale item bank. But with the increasing of test scale, disadvantages of CGA algorithm are increasingly apparent. After analysis, adding cellular automata to traditional genetic algorithm for test sheet generation can further meet concept correlation requirements of users.

In order to analyze the fitness degree, six kinds of experimental methods are divided into two categories: selection strategy and non-selection strategy. We obtain the average fitness value of two strategies under different test scales. The results are shown in Figure 3.

As can be seen from Figure 3, the fitness value in both cases shows an upward trend, but three algorithms based on selection strategy have a higher average value of fitness. The reason for it is that before test sheet generation, adaptive selection strategy filters test items in item bank to rebuild the candidate item bank, which will help to improve the convergence and reduce unnecessary genetic manipulation when generating test sheet.

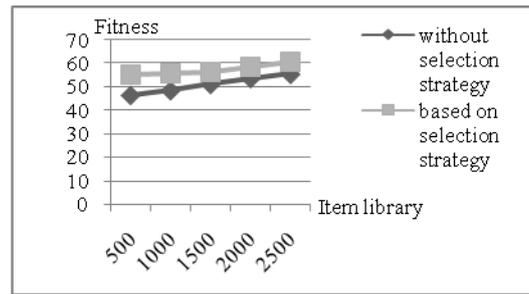


Figure 3. Fitness of the Same Test Sheet Generation Algorithm

6. Conclusions

It is universally acknowledged that traditional genetic algorithm in test sheet generation process has many drawbacks, such as poor convergence, low fitness and high knowledge related errors. To solve these problems, this paper proposes an adaptive cellular genetic algorithm based on selection strategy with a combination of ATGS and CGA. This method effectively reduces solving space of test sheet generation, improves fitness, and lowers knowledge related errors which makes it more effective to complete assessment of people being tested. In future studies, we can do in-depth research in aspects of selection strategy and evolution rules of cellular automata to make error accuracy and convergence speed of test sheet generation further improved.

Acknowledgements

This work is supported by Beijing Higher Education Reform(No. 2013-ms047), Beijing Forestry University Resources Sharing Course(Data Structure), the National College Students' Training Programs of Innovation and Entrepreneurship(No. G201410022036) and Beijing Forestry University Special Research of Campus Informatization (Construction of the Teaching Resources Sharing Platform for Program Design and Algorithms Courses).

References

- [1] H. Y. Lin, J. M. Su and S. S. Tseng, "An adaptive test sheet generation mechanism using genetic algorithm", *Mathematical Problems in Engineering*, vol. 2012, (2012), pp. 1-18.
- [2] T. H. Cormen, C. E. Leiserson and R. L. Rivest, "Introduction to algorithms", Cambridge: MIT Press, (2009).
- [3] G. X. Xiao and W. C. Zhao, "Iterant problems replacement method based on genetic algorithm with test paper intelligent", *Generation Computer Engineering*, vol. 38, no. 11, (2012), pp. 150-152.
- [4] J. H. Holland, "Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence", U Michigan Press, (1975).
- [5] G. J. Hwang, B. M. T. Lin. and H. H. Tseng, "On the development of a computer-assisted testing system with genetic test sheet-generating approach", *IEEE Transactions on Systems Man, and Cybernetics, Part C: Applications and Reviews*, vol. 35, no. 4, (2005), pp. 590-594.
- [6] B. L. Huang, "Application of adaptive genetic algorithm in intelligent test paper composition", *Computer Engineering*, vol. 37, no. 14, (2011), pp. 150-152.
- [7] K. Lu, D. M. Li and J. Yu, "Intelligent test paper construction method based on cellular genetic algorithm", *Computer Engineering and Applications*, vol. 49, no. 16, (2013), pp. 57-60.
- [8] S. W. Chang and T. N. Ansley, "A comparative study of item exposure control methods in computerized adaptive testing", *Journal of Educational Measurement*, vol. 40, no. 1, (2003), pp. 71-103.
- [9] J. P. Leighton and M. J. Gierl, "Cognitive diagnostic assessment for education: Theory and applications", Cambridge University Press, (2007).
- [10] C. Y. Su and T. I. Wang, "Construction and analysis of educational assessments using knowledge maps with weight appraisal of concepts", *Computers & Education*, vol. 55, no. 3, (2010), pp. 1300-1311.
- [11] J. Kari, "Theory of cellular automata: a survey theoretical", *Computer Science*, vol. 334, no. 1, (2005), pp. 3-33.
- [12] Y. F. Liu and S. Y. Liu, "Algorithm based on genetic algorithm for sudoku puzzles", *Computer Science*, vol. 37, no. 3, (2010), pp. 225-226,233.

- [13] Lu Y. H. and Liu H., "Auto-generating examination papers based on integer coding and adaptive genetic algorithm", *Computer Engineering*, vol. 31, no. 23, (2006), pp. 232-236.

Authors



Ankun Huang, he is an undergraduate student in School of Information and Technology, Beijing Forestry University. He joined Institute of Artificial Intelligence in Beijing Forestry University in 2012 as a research assistant. His main research interests include genetic algorithm and intelligent information processing.



Dongmei Li, she received her master degree in Institute of Software, Chinese Academy of Sciences and her Ph.D. degree from Beijing Jiaotong University. Currently she is a lecturer in School of Information and Technology, Beijing Forestry University. Her main research interests include artificial intelligent, knowledge engineering and semantic Web.



Jiajia Hou, she received her bachelor degree from Beijing forestry University. Currently she is a mater student in School of Information, Renmin University of China. Her main research interests include genetic algorithm and intelligent information processing and semantic Web.



Tao Bi, he is an undergraduate student in School of Information and Technology, Beijing Forestry University. He joined Institute of Artificial Intelligence in Beijing Forestry University in 2012 as a research assistant. His main research interests include genetic algorithm and intelligent information processing.