Automatic Test Paper Generation Based on Ant Colony Algorithm

Dan Liu
School of Information Science and Technology, Shijiazhuang Tiedao University,
Shijiazhuang050043, China
E-mail: liudanld@126.com

Jianmin Wang
School of Information Science and Technology, Shijiazhuang Tiedao University,
Shijiazhuang050043, China
E-mail: wangjm@stdu.edu.cn

Lijuan Zheng
School of Information Science and Technology, Shijiazhuang Tiedao University,
Shijiazhuang050043, China
E-mail: zheng_lj@sina.com

***Abstract*—Traditional method of test paper generation is time-consuming, low efficient and low success. In order to improve the success rate of test paper, the paper puts forward an ant colony algorithm of automatic sets method. First, a mathematical model of constraint is built according to the requires of papers, and by using the ant colony algorithm, the optimal solution of grouping is obtained. Simulation results show that, compared with the traditional sets algorithm, the ant colony algorithm improves the efficiency of grouping, increase rate of grouping, and satisfies the current network on-line examination system of real-time higher requirements of grouping.**

***Index Terms*—test paper generation, ant colony algorithm, mathematical model**

I. INTRODUCTION

As a key link in the teaching process in the educational field, examination is not only an evaluation of students' learning ability but also a means to testing teachers' teaching effect. Under great influence of human factors in the content, marking criterion and scoring, traditional examinations show low reliability and validity. As our science and technology develops, computer has entered the educational field and become a powerful tool of teaching media and management. Examinations based on computers can greatly enhance the efficiency and make test paper generation, scoring and score statistics intelligent. Thus, the actual effect of teaching can be reflected authentically, objectively and comprehensively, which will in the end improve the teaching quality.

Automatic test paper generation on computers refers to the process in which questions are extracted from the examination base automatically and a test paper that can meet the requirements of question types, knowledge points, difficulty and score distribution is constituted. At present, the methods of automatic test paper generation on computers can be divided into three kinds: random variable algorithm, backtracking algorithm and artificial intelligence algorithm. Random variable algorithm extracts questions randomly and judge if they meet the constraint conditions of the test paper. Since it is simple to realize random variable algorithm, it has been extensively applied in practice. However, random variable algorithm often fails to meet the requirements of test papers and stands a great chance of generating invalid test papers. Therefore, the success rate of this method is pretty low. Furthermore, in the condition of vast question base, it takes a long time to search and reduces the efficiency of generating test papers. In consideration of this, some scholars improved the automatic test paper generation algorithm and put forward backtracking algorithm, which can effectively reduce the probability of invalid test papers, thus greatly improving the performance. Since this algorithm is also simple to realize, it is adopted in many situations. Although it avoids the blindness of random variable algorithm, it merely adjusts the process of generating test papers passively, and when the question base is vast, the times of backtracking will be largely increased, accompanied by a large internal storage and a long period of time wasted. Therefore, many scholars began to apply artificial intelligence technology to generate test papers automatically, trying to improve the performance of automatic test paper generation system and the quality of test papers, such as genetic algorithm and simulated annealing algorithm. Genetic algorithm can realize global optimization with a fast rate of convergence. However, its performance is mainly decided by genetic operators, which means different genetic algorithms need to be set up based on different requirements of test papers. Genetic algorithm is quite complicated, and automatic test paper generating

doi:10.4304/jsw.8.10.2600-2606

system based on this algorithm shows poor generality, expandability and flexibility. Therefore, designing reasonable algorithm to compose test papers has always been the difficult and hot issue in the study of automatic computer test paper generating system.

As a new type of simulated and evolved heuristic algorithm, ant colony algorithm (ACA) was put forward by Marco Dorigo, an Italian scholar in accordance with the group behavior of ant foraging. In this algorithm, positive feedback parallel mechanism is adopted, which endows it with the ability of fast and intelligent search. Thanks to this, it has been effectively applied in UAV navigation, logistics addressing and color blending. Since automatic test paper generation and ant foraging share many similarities in optimal solution, ant colony algorithm offers a new channel to the solution of automatic test paper generation. Therefore, an automatic test paper method based on ant colony algorithm is put forward in this dissertation. It is suggested by case result that when comparing with other kinds of commonly used automatic algorithm to generate test papers, automatic test paper generating system based on ant colony algorithm improves the success rate of test paper generation, and accelerates its speed, which lead to higher quality of test papers being generated that can basically accommodate to the requirements of automatic test paper generation-high success rate and instantaneity.

Ⅱ. AUTOMATIC TEST PAPER GENERATION

Automatic test paper generation selects a certain number of questions from examination database which contains a large number of questions to generate a test paper that can satisfy the requirements of examination. Different questions can generate many test papers in different qualities. Automatic test paper generation is the core of automatic examination system, and the quality of questions is the key in improving examination quality, which relies on scientific test paper generation[4]. In other words, questions should be able to reflect the content of examination and the formation of questions should be so reasonable to reflect different examination targets differently. Therefore, the establishment of standardized examination database should firstly refer to the principles of test paper generation, the evaluation indexes of test paper and the relationship between each index to establish a test paper generating model.

A.. Basic Principles of Test Paper Composition

1) Test paper composition must comprehensively reflect and consider the depth and breadth of syllabus.

2) Test paper composition must be conducive to students' ability assessment, and be able to improve students' intellectual development.

3) Test paper composition must have hierarchy and steps based on the level of difficulty, which can not only expand the distance in scores without tricky or strange questions, but also reflect depth that can measure students' different knowledge level and intellectual differences.

4) Test paper composition must highlight the questions' inspiration and guiding role in students' learning methods.

5) The expression of test questions must be clear and error-free to improve the reliability of examinations.

B. Attribute Indexes of Test Questions

The questions involved in the test paper generation are stored in a database in the form of records. Common properties of questions include: scores of test paper, difficulty of test items, involved knowledge points of chapters, test patterns, estimation time and so on. Automatic test paper generation can be described as selection of a set of question combination from the database with certain question quantity, which meets the requirements of goals of the test paper generation[3]. Thus, the test paper generation is transformed into a multi-constraint objective problem, as follows: solve a test paper made by m questions and each question shall have n properties. This is equivalent to building a matrix of $m*n$ goal.

$$T = \begin{pmatrix} a_{11}a_{21} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1}a_{m2} & \cdots & a_{mn} \end{pmatrix} \quad (1)$$

In the matrix, each row represents a property of question. There are totally n properties; each line represents a question meets the nth attribute. There are totally m properties. In the target state of matrix, it is assumed that the first row represents the score index of each question, expressed by a_{i1} ; the second row represents the difficulty attribute of each question, expressed by a_{i2} ; the third row represents the time attribute of each question, expressed by a_{i3}; the fourth row represents the specification attribute of each question, expressed by a_{i4}; the fifth row represents the test pattern attribute of each question, expressed by a_{i5}; and so forth with other attributes (the number of properties is set by the users based on their requirements). Based on the above listed five attributes, target matrix should satisfy the following constraints:

1) Conditions of Total Score. a_{i1} indicates the score of ith question;

$$K = \sum_{i=1}^{m} a_{i1} \quad (2)$$

2)Total score share of K type question. a_{i5} indicates the test pattern attribute of ith question;

$$Q_t = \sum_{i=1}^{m} c_{ij} a_{i5} \quad (3)$$

$$c_{ij} = \begin{cases} 1, a_{i5} = k \\ 0, a_{i5} \neq k \end{cases}$$

3)The Total Time of Test Paper. a_{i3} indicates the times of each question;

$$T = \sum_{i=1}^{m} a_{i3} \quad (4)$$

4)Coefficient of difficulty. a_{i2} indicates the difficulty of ith question;

$$PS = \sum_{i=1}^{m} a_{i1} a_{i2} / K \quad (5)$$

5)Differentiation of Test Paper. a_{i4} indicates the differentiation of ith question;

$$DS = \sum_{i=1}^{m} a_{i1} a_{i4} / K \quad (6)$$

Property coefficients of a test paper are often mutually restrained. The computer automatic generation of test paper is not a simple random process but a complex issue of several goals. The issue is which algorithm its core adopts.

C. Target Function of Automatic Test Paper Generation

It is suggested by the mathematical model used in test paper generation based on computers that automatic test paper generation is a typical optimization issue with multiple objectives, and consequently, many objectives often need to be taken into consideration at the same time, and on the premise of restriction conditions, the smaller all target functions are, the better. However, certain conflicts exist among different objectives contained in automatic test paper generation. In other words, it is hardly possible for multiple objectives to achieve optimization in one solution. Therefore, under general conditions, in the optimization solution with multiple objectives, in accordance with certain utility functions, the most commonly used method is to generate multiple objectives to a single one, and solve with the optimization methods of single target functions[6-8]. This dissertation adopts target weighting method to model automatic test paper generation, whose basic thought is to endow a weight to each target, multiply the weight of all targets with their weight coefficient correspondingly, and add them up to generate a new target function(7).

$$\min \sum_{j=1}^{J} (d_{lj} + d_{uj}) \quad (7)$$

$$\begin{cases} \sum_{i=1}^{N} x_i = n \\ \sum_{i=1}^{N} a_{ji} x_i + d_{lj} - e_{lj} = L_j, \ j = 1,2,...,J \\ \sum_{i=1}^{N} a_{ji} x_i + d_{uj} - e_{uj} = U_j, \ j = 1,2,...,J \end{cases}$$

$x_i = \begin{cases} 1 \\ 0 \end{cases}, i = 1,2,...N$ when the ith question is selected, x_i is 1,otherwise is 0.

The specific target function of test paper generation finally achieved is shown as:

$$f(X) = w_1 \times d_1 + w_1 \times (d_{L1} + d_{U1} + d_{L2} + d_{U2}) + w_3 \times d_3 \quad (8)$$

Unlike traditional methods to generate test papers, which consume a relatively long time and generate test papers in low efficiency, and thus cannot satisfy the instantaneity requirement of automatic examination system, the strategy of automatic test paper generation targets at students' ability and selects questions from the examination database to generate a test paper. Ant colony algorithm is a kind of colony intelligence searching method, and is equipped with positive feedback paralleling mechanism, with strong searching capability, enabling it to be appropriate for the solution of automatic test paper generation, especially binary ant colony algorithm, which enables ant to only select between 0 and 1 in each moment due to its special random binary system chain structure. 0 and 1 actually correspond with "yes" and "no" of the question in examination database. Therefore, this dissertation adopts binary ant colony algorithm to conduct automatic test paper generation.

III. AUTOMATIC TEST PAPER GENERATION BASE ON ANT COLONY ALGORITHM

A . The Design of Automatic Test Paper Generation Based on Ant Colony Algorithm

Test Paper Coding Mode: Test paper coding is the first question automatic test paper generation based on ant colony algorithm needs to solve. Generally speaking, binary coding mode is the most commonly used data representation, and the easiest to simulate and realize two-valued logic in the Nature. Therefore, this dissertation adopts binary coding mode to code test papers. In practical process to generate test papers, the number of each type of questions is fixed, and as to the same question type, the answering time and score are also the same. As a result, the whole binary code can be divided into different functional blocks according to question type; each functional block corresponds with a kind of question type. Set the total length of binary code to $LC \times VN$. VN represents the number of question types; LC is the number of questions in corresponding type. Assuming there are $LC \times VN$ questions in the examination database, the coding mode is:

$$x_1 x_2 ... x_{LC\times VN}\ , x_i = \begin{cases} 1 \\ 0 \end{cases}. \quad (9)$$

Extra requirements are:

$$\sum_{i=1}^{k_1} b_i = m_1 \quad (10)$$

$$\sum_{i=k_1+1}^{k_l} b_i = m_l \quad (11)$$

In this formula, k_i represents the number of type i in the examination database, and m_i represents the number of questions in each question type required by type i test paper.

The Design of Fitness Function: As an index used to judge good or bad individuals in test paper group, fitness function value requires that characteristics of questions being solved should be satisfied. This dissertation adopts target function and fitness function on test paper generation, namely(12):

$$f(x) = \min \sum_{j=1}^{J} w_j \times (d_{lj} + d_{uj}) \quad (12)$$

Rules of Pheromone Update: In order to select questions in specified number from the examination database to form a test paper that can meet users' requirements and conditions of test paper generation, pheromone concentration of each question should be initialized and volatilization value should be calculated at first. Place ants to any question in the examination database, the ants will calculate its preference number in accordance with pheromone concentration and volatilization function during the process of moving, and then select the next direction based on the preference number. Store questions picked up by the ants to a section of memory space. After all ants finish selecting initial cycle solution, check the solution selected by ants to see if it can meet all restricting conditions of automatic test paper generation. If the solution fails to meet such conditions, modify it until it does. And then adopt area searching method to further improve the solution, so that it can be closer to the optimal solution. Find out the test paper with the best target function in this round, and call it the optimal iteration solution of this round. If the solution in this round is better than the globally optimal solution, then it can replace the latter and become the optimal solution. The last step is to update pheromone.

State Transition Probability: At the initial moment, pheromone concentration of each question is the same. In the moving process, ants select a question based on pheromone concentration and volatilization value. The probability question is chosen can be reflected in the intensity of pheromone. The higher pheromone of question gets, the more frequently such question will be selected by ants. In this way, ants will accumulate more pheromone concentration of this question. The volatilization of ith question is represented by η_i i, and the calculation formula is (13):

$$\eta_i = \left[\sum_{k=1}^{s} I_i(\theta_k) \right] / s \quad (13)$$

Then the transition probability of ants in the moving process is:

$$p_{ij}^k(t) = \begin{cases} \dfrac{\tau_{ij}^{\alpha}\eta_{ij}^{\beta}}{k\tau_{is}^{\alpha}\eta_{is}^{\beta}(t)}, s \in allowed, j \in allowed_k \\ 0, otherwise \end{cases} \quad (14)$$

Among which, η_{ij} represents partial volatilization value of the transition from question i to question j; $allowed_k$ represents available questions in the next step; α and β represent weight parameters respectively, which are used to coordinate pheromone and partial heuristic information of heuristic information factors on the path.

Record questions already selected by each ant with a taboo list, and continuously update in search process. $p_{ij}^k(t)$ represents ant $k's$ state transition probability from question i to question j at t moment.

Revise of Illegal Individuals: In the process of automatic test paper generation, the specified number of questions in each type is fixed. In ergodic process, ants will definitely form several illegal solutions, which should be modified to keep each individual in the group legal. The process occurs after ant colony finishes ergodic of all questions in the examination database and before path pheromone updates, aiming at modifying illegal individuals to make them reasonable. The specific steps are shown below:

1) Find out illegal individuals that cannot meet restricting conditions of test paper generation in ant colony.

2) Conduct multiple-refund operation on individuals.

3) Further this operation if there are still conflicts.

4) Conduct supplementary operation for any deficiency.

5) After refund and supplement is finished, the revise of illegal individuals is completed.

Termination Conditions of Algorithm: Under the circumstances when the number executed by ant algorithm has reached the maximum number of iteration, or the fitness value has reached the preset value, and better optimal solution fails to occur to replace the existed optimal solution even after many times of iteration, automatic test paper generation is completed.

B. The implementation of Ant Colony Algorithm

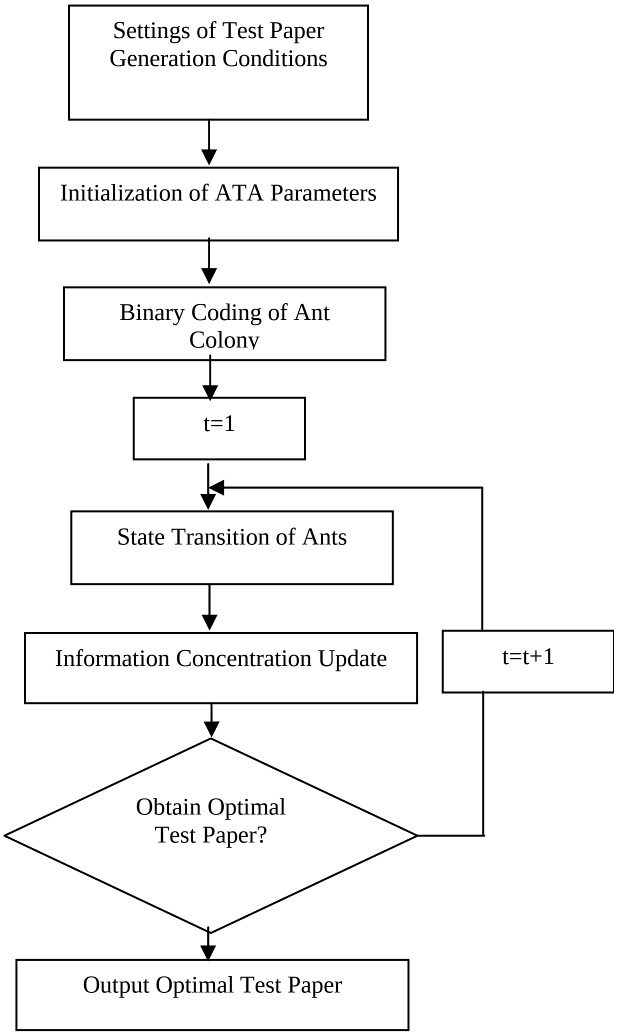

Figure 1. The implementation of Ant Colony Algorithm

1) Set requirements in test paper generation.

2) Initialization of ant colony and path. All initial pheromones on the path of test questions are 0; initial time t=0.

3) Select a question number randomly from the examination database and put into corresponding taboo list, and set the index number of ants' taboo list to 1.

4) Add 1 to cycle index.

5) In accordance with state transition probability formula, individual ants select question numbers that should be selected in the next step.

6) Transit ants to new questions after selection, insert corresponding question numbers to the taboo list, and modify index number in the taboo list.

7) Contrast questions selected by each ant to requirements input by users, if such requirements are satisfied, a test paper is achieved.

8) If the test paper is better than overall optimal test paper, the former can replace the latter.

9) Partial and overall pheromone update.

10) End condition judgment. If end condition is satisfied, input the optimal test paper, otherwise, skip to Step 4) and continue with test paper generation. The detailed procedure of test paper generation is shown in Figure 1.

Ⅳ. EXPERIMENTAL RESULT AND ANALYSISCOMPARISON OF TEST PAPER EFFICIENCY

A . Requirement of Test Papergeneration and Parameter Settings

In order to test the performance of automatic test paper composition stated in this dissertation, the experiment was made under the conditions-CPU3GHz, 2G internal memory, Windows XP operating system, and Visual C++ programming language. There are 1000 test papers in the question bank, covering ten chapters in total. See the structure of test papers and the distribution of scores in each chapter in Table 1 and Table 2.

TABLE I.

SCORE DISTRIBUTION OF EACH CHAPTER

Chapter	Fraction	Chapter	Fraction
1	20	6	5
2	10	7	5
3	10	8	10
4	20	9	5
5	10	10	5

TABLE II.

PARAMETER SETTINGS

Questions	Number of Question	Fraction
Choice	30	2
Blank filling	20	1
True or false	20	1

B. Comparison of the Efficiency of Test Paper Generation

In order to examine the efficiency of test paper generating algorithm put forward in this dissertation, random drawing and genetic algorithm should be adopted to conduct ten groups of experiments on automatic test paper generation under the same conditions. The time spent on each group of successful experiments is shown in Figure 2. It can be seen from the result of Table 1 that the speed of random drawing is relatively slow, test paper generation costs a long time and the efficiency of such generation is extremely low. The time genetic algorithm costs is also far longer than ant colony algorithm does, which is mainly because in hereditary variation process, it is easy for genetic algorithm to generate illegal individuals, which take a long time to revise. Ant colony algorithm in this dissertation adopts positive feedback paralleling mechanism, which is equipped with the ability of fast and intelligent searching, which enables it to find out the optimal test paper generating plan in the shortest time, thus effectively improving the efficiency of test paper generation.

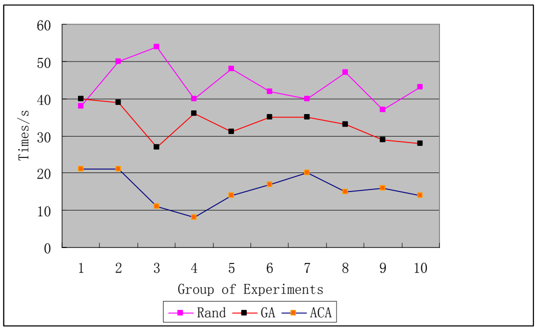

Figure 2. Comparison of test paper efficiency

C. Comparison of the Success Rate of Test Paper Generation

The comparison of the success rate of several test paper generating algorithms can be seen in Figure 3 and Figure 4. It is suggested by the result that random drawing has the lowest success rate in test paper generation, and the most times of failures, followed by genetic algorithm. The test paper generation based on ant colony algorithm put forward in this dissertation reaches 100% success rate when the number required is not big. While when the number required is big, it can keep above 99% success rate. However, random drawing and genetic algorithm suffer from a substantial increase in the times of failures. It is suggested by the result of contrast that test paper generating method based on ant colony improves the success rate and efficiency of test paper generation. Therefore, test paper generating method based on ant colony algorithm is quite suitable for the current requirements of online examinations.

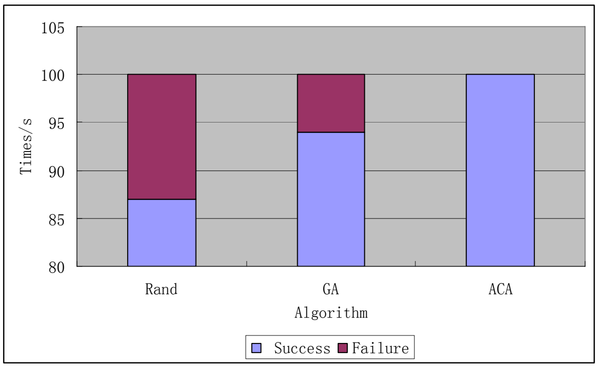

Figure 3. Comparison of the success rate of test paper generation(100 times)

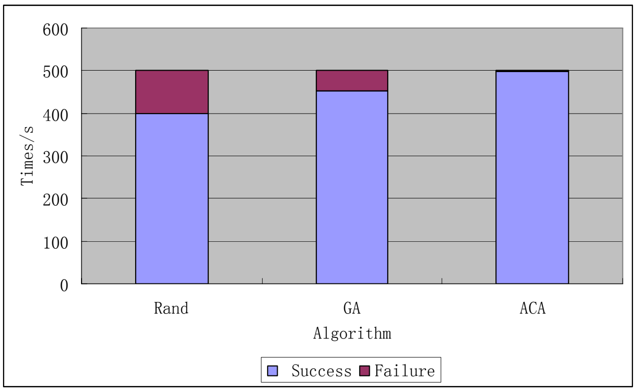

Figure 4. Comparison of the success rate of test paper generation(500 times)

V. Conclusion

With the development of internet and computer technology, the degree of educational informationization has been rising, and with striking changes in the mode of examination, intelligent test paper generating algorithm is becoming increasingly important. Aiming at the solution procedure of such a typical constraint satisfaction issue of test paper generation, this dissertation suggests adopting group intelligent searching method-ant colony algorithm in automatic test paper generation. It is suggested by the result of such simulation that when comparing with traditional and automatic methods to generate test papers, ant colony algorithm can find out the optimal plan in test paper generation in the shortest time, especially when the number required is big, it can still maintain a high success rate, thus better realizing the strategy of test paper generation, and laying a firm foundation for constructing a perfect automatic examination system based on computers.

Acknowledgement

The authors are extremely grateful to the anonymous reviewers for their constructive and valuable comments, which have contributed a lot to the improved presentation of this paper. The work was partially supported by scientific research project of Hebei education department(no.Z2012151).

REFERENCES

[1] Roex, A., and Degryse, J. A., “A Computerized Adaptive Knowledge Test as an Assessment Tool in General Practice: A Pilot Study”, *Medical Teacher*, vol.26, no.2, pp.178-183, 2004

[2] Chen, C. M., Lee, H. M., and Chen, Y. H, “Personalized E-learning System Using Item Response Theory”, *Computers & Education*, vol. 44 , pp.237-255, 2005.

[3] Guzman, E., and Conejo, R., “Self-Assessment in a Feasible, Adaptive Web-based Testing System”, *IEEE Transactions on Education*, vol.48, no.4, pp.688-695, 2005.

[4] Wang, P., Dai, H., and Ding, S., “Computerized Adaptive Testing in Chinese Mainland: A Review”, *JDCTA: International Journal of Digital Content Technology and its Applications*, vol. 4, no. 6, pp. 94-99, 2010.

[5] Wang, P., and Gao, F.,"Full-Information Item Bifactor Analysis of the Job Burnout Scale for Chinese College Teachers", *JCIT: Journal of Convergence Information Technology*, vol. 5, no. 7, pp. 155- 160, 2010.

[6] Mason, B. J., Patry, M., and Bernstein, D.J., “An Examination of the Equivalence Between Non- Adaptive Computer-Based Test and Traditional Testing”, *Journal of Educational Computing Research*, vol.24, no.1, pp.29-39, 2001..

[7] Ying, M.H., Wu, L.J., and Huang, H. H., “An Adaptive Testing System Based on Personality and Positive Psychology”, *14th Conference on Information Management & Practice*, Taiwan, pp.1-17, 2008.

[8] Anderson, L. W., Krathwohl, D. R., Airasian, P. W., Cruikshank, K. A., Mayer, R. E., Pintrich, P. R., Raths, J., and Wittrock, M. C. *A Taxonomy for Learning, Teaching, and Assessing: A Revision of Bloom’s Taxonomy of Educational Objectives*, Longman, USA, 2001.

[9] Barnes, J. A., “Class and Committees in a Norwegian Island Parish”, *Human Relations*, vol.7, no.1, pp.39-58, 1954.

[10] Opsahl, T., Agneessens, F., and Skvoretz, J., “Node centrality in weighted networks: Generalizing degree and shortest paths”, *Social Networks* , vol.32, no.3, pp.245-251, 2010.

[11] Lo, L., and Lin, S. W., “The Effect of Price Presentation, Sales Restrictions, and Social Networks on Consumer EWOM Intention “, *AISS : Advances in Information Sciences and Service Sciences*, vol. 3, no. 1, pp. 1- 9, 2011

[12] Burshtein David, Goldenberg Idan, “Improve Linear Programming Decoding of LDPC Codes and Bounds on the Minimum and Fractional Distance”, *IEEE Trans. Inform*. Theory, vol. 57, no. 11, pp. 7386-7402, 2011.

[13] Kudekar Shrinivas, Johnson Jason K, Cherkov M, “Improve Linear Programming Decoding Using Frustrated Cycles”, available at http://arxiv.org/abs/1105.4665.

[14] Zhang Xiaojie; Siegel Paul H, “Adaptive Cut Generation Algorithm for Improved Linear Programming Decoding of Binary Linear Codes”, *IEEE Trans. Inform. Theory*, vol. 58, no. 10, pp. 6581-6594, 2012.

[15] Fan Xiaona, Yan Qinglun, Gong Dianxuan, “New Algorithm for Convex Nonlinear Programming Problems”, *Journal of AISS*, Vol. 4, No. 17, pp. 442 ~ 450, 2012

[16] Li Yiming, Li Yi, Zhou Mingtian, “A Mixed Algorithm to Minimize the Wire Length in Postfloorplanning”, *Journal of IJACT*, Vol. 3, No. 6, pp. 320 ~ 328, 2011

[17] Lunglmayr M, Berkmann J, Huemer M, “Combined linear programming / belief propagation decoder” , *IET Electron. Lett*., vol.44, no.12, pp. 751-752, 2008.

[18] Wang Yige, Yedidia Jonathan S, Draper Stark C, “Multi-stage Decoding of LDPC Codes” , *In Proceeding of the IEEE Int. Symp. on Inf. Theory (ISIT)*, pp. 2151-2155, 2009.

[19] Zhang Xian, Ma Piming, “Modified multistage linear programming decoding of LDPC codes” , *In Proceeding of the IEEE 13th Int. Conf. on Comm. Tech. (ICCT)*, pp. 104-108, 2011.

[20] Planjery Shiva Kumar, Vasic Bane, Declercq David, “Decimation-Enhanced Finite Alphabet Iterative Decoders for LDPC Codes on the BSC” , *In Proceeding of IEEE Int. Symp. on Inf. Theory (ISIT)*, pp. 2383-2385, 2011.

[21] Planjery Shiva Kumar, Vasic Bane, Declercq David, “Enhancing the error correction of finite alphabet iterative decoders via adaptive decimation” , *In Proceeding of the IEEE Int. Symp. On Inf. Theory (ISIT)*, pp. 2876-2880, 2012.

[22] Khajehnejad Amin, Dimakis Alexandros G, Hassibi Babak, et al., “Reweighted LP Decoding for LDPC Codes” , *IEEE Tans. on Inf. Theory*, vol. 58, no. 9, pp. 5972-5984, 2012.

[23] Tanner R M, Sridhara D, Fuja T, “A class of group-structured LDPC codes” , *In Proceedings of the 6th Int. Symp. Commun. Theory and Applications (ISCTA)*, pp. 365–370, 2001.

Dan Liu was born in Shijiazhuang, Hebei Province. She received her M.Sc. degree from School of Information Science and Technology Shijiazhuang Tiedao University, China, in 2011. She is currently a lecture at School of Information Science and Technology Shijiazhuang Tiedao University, China. Her research interests include soft engineering

Jianmin Wang received his M.Sc. degree from Yanshan University, China, in 2008. He is currently a associate professor at School of Information Science and Technology Shijiazhuang Tiedao University, China. Her research interests include soft engineering

Lijuan Zheng received the B.A’s and M. A’s degrees in computer application from North China Electronic Power University, Baoding, China, in 2000 and 2003 respectively. Since 2007, she has been a Ph. D. degree candidate in information security from Beijing Jiao Tong University, Beijing, China. Her research interests include security protocol analysis and design, trusted computing.