International Journal of Hybrid Information Technology
Vol.9, No.3 (2016), pp. 311-322
http://dx.doi.org/10.14257/ijhit.2016.9.3.29

Computer Game System Modeling and Searching Algorithm for Occupancy of Niujiu Card

Chuanxi Zhang, Dongmei Li* and Cong Dai

School of Information and technology in Beijing Forestry University, Beijing 100083, China
lidongmei@bjfu.edu.cn

Abstract

As a carrier of artificial intelligence research, computer game can establish a sound theoretical foundation for the research of non-zero-sum card games with imperfect information. In this paper, we first propose a game model based on the finite Moore automaton, which illustrates the implementation of the model for Niujiu card. At the same time, a novel search strategy combining IMP-minimax and Monte Carlo algorithm is presented in this model. Through the given algorithm, this paper accomplishes the simulation, including not only the process of minimal and optimal occupancy for the first player, but the available frequency of special card type. Experiments show that our model and algorithm are feasible and effective.

Keywords*: Finite automaton; Niujiu card; Computer game; IMP - minimax algorithm; Monte Carlo algorithm*

1. Introduction

Computer game is an important branch of artificial intelligence, which mainly targets at how to enable computer to simulate the thinking of human brains. However, either at home or abroad (in the domestic or overseas), the current study on computer game is mostly focusing on search algorithm and evaluation function [1-4].

Finite automaton is a dynamically and systematically mathematical model of discrete events. At present, computer game based on automaton has made a large number of achievements [5-7], which provides a good reference for modeling of automaton in computer game. Niujiu card is an ancient card game prevailing in northwestern China. Its logic of occupancy is complex and flexible. Niujiu card is appropriate for multi-player non-zero-sum games with imperfect information. By studying Niujiu card game model, a sound theoretical foundation could be established for its study, so that a good computer game theoretical model could be presented [8-10].

This paper first makes a modeling analysis aiming at three-player card game and provides formal definitions of card game model. Second, we build a theoretical game model based on finite automaton for occupancy of Niujiu Card. Now we know that search algorithm is based on IMP-minimax, and it can provide a timely linear strategy of the search tree for non-zero-sum games with imperfect information [11, 12]. According to this feature, an IMP-minimax algorithm search method is presented for special card types. And then, we propose a Monte Carlo estimation of distribution algorithms, which combines with IMP-minimax algorithm, estimating the probability of special card types. On this basis, this paper accomplishes the gaming process of minimal occupancy and optimal occupancy for the first player. Finally, we verify the feasibility and effectiveness of our model by experiments.

ISSN: 1738-9968 IJHIT

2. Card Game Model

2.1. Composition of Card Game Model

A card game model is defined as a 6-tuple {Ha, Hb, Pi, Qi, C, W}, the structure is shown in Figure1, where

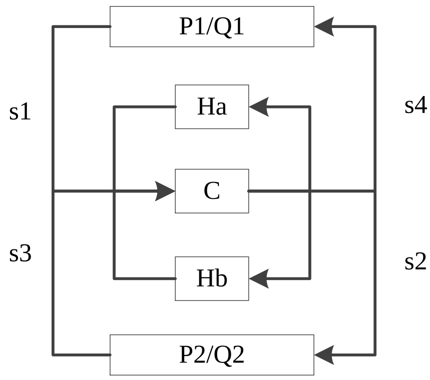


Figure 1. Composition of Card Game Model

(1)Ha represents the holding card type, which means the sum of the holding card types of all players.

(2) Hb represents the inholding card type, which means the sum of the un-holding card types of all players.

(3) Pi(i=1,2,3,⋯⋯) represents human players, where i represents the number of human players.

(4) Qi(i=1,2,3,⋯⋯) represents computer players, where i represents the number of computer players.

(5)C represents controller, which can judge the process of playing cards according to player's card type.

(6)W represents game interface.

There are multiple players in the game. However, game process can be regarded as a type of process between two players. Therefore, we present examples to illustrate our model by using two players interaction.

In the process of game, human player P1 or computer player plays a type of holding card type. Through step s1, holding card type is received by the controller C. If the next player is computer player Q2, controller C will judge the response of computer player according to inholding card type. And then controller C sends results through s2 to computer player Q2. If the next player is human player P2, controller C transfers results through s2 to human player P2.

According to the output results received through step s3, computer player Q2 or human player P2 sends information to the controller C. After C completes the same above-mentioned step, C sends information through s4 to computer player Q2 or human player P2.

2.2. Controller of Card Game Model

Controller C is defined as a 5-tuple {M, A, γ, Ri, T}.It is shown in Figure 2.

(1)Ri (i= 1, 2, 3) represents the game rules, different games have different rules.

(2) M represents finite automaton, which provides a procedural structure to screen current game rules Ri.

(3) γ represents evaluation functions, combining the searching algorithm, and can evaluate current situation in order to select optimal method.

(4) D represents algorithm base, which provides algorithms to compute finite automaton M and evaluation functions γ.

(5) T represents transverter, which transforms the information received by controller to graphical interfaces.

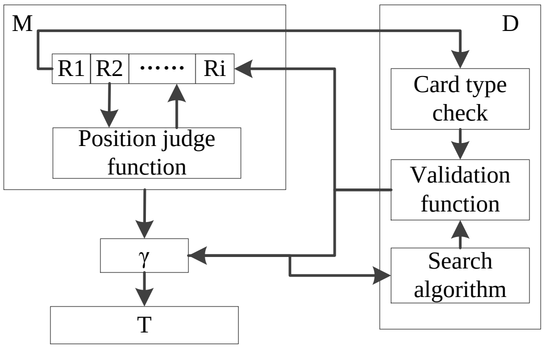


Figure 2. Moore Automaton-based Controller of Computer Game

Players sends holding card type to the finite automaton M. M invokes D type detection algorithm according to the game rules. If the verified results are correct by D, D will send verified results to M and invoke card position judgment function to record inholding card position. Meanwhile, card position judgment function sends results to M. After M verifies results, if results are correct, the system will invoke evaluation functions γ and γ invokes algorithm in the D for further processing. If validation functions verified results are correct, validation functions will send results to γ, and then γ sends final results to converter T, and T converts results to graphical and outputs to the game interface.

3. Realization of Finite Automaton Based Card Game Model for Niujiu Card

3.1. Formulas and Definition of Niujiu card

Niujiu card consists of one pack of poker excluding the two jokers and the card of No. 4. To introduce the terms of Niujiu cards, the following definitions are given.

Definition 1 Card mark x∏: In a pack of poker, card mark is used to distinguish whether the value of card is big or small. x is a notation on top left corner and lower right corner (x∈{1,2,3,5,6,7,8,9,10,J,Q,K}), and ∏ represents the meaning of card number mark.

Definition 2 Suit symbol: If a suit needs to be distinguished, the suit will add – or + to the front of card number expression. Suit symbols represent the number of a certain card suit, of either red or black. For example, the card number expression -n→1∏ represents the number of black cards marked with 1∏is n.

Definition 3 Card type φ: A set consists of designated same or different card number that has different number denoted as φ

3.2. Finite Automaton Based Recognition of Common Cards

3.2.1. Building of finite automaton M

According to the Niujiu card rules, every player plays 16 holding card type by certain rules. If the card type is the biggest, this card type is called available occupancy. Meanwhile, the final limit of occupancy cards is 1/3 of total cards, which is 16 cards. To maintain the rules that only two players have enough occupancy of cards at most, the situation that every player occupies 6 cards is accepted as a final state. In order to allow players to have more room to play further, we only consider available occupancy of six cards of first player. This situation is called six cards strategy. Computer automaton process is as follows.

(1)According to the number of occupancy, M confirms procession state of first player.

A0: No valid cards available on play table.

A1: One first player valid card available on play table.

A2: Two first player valid cards available on play table.

A3: Three first player valid cards available on play table.

A4: Four first player valid cards available on play table.

A5: Five first player valid cards available on play table.

A6: Six first player valid cards available on play table.

(2)According to standard card type of occupancy, M confirms finite automaton alphabet. Notation and model of standard card type utilize what be set before. Playing card numbers are listed in brace regardless of sequence.

φ1a: Two-Niu card, formula is {-9∏, +9∏}.

φ1b: Three-Niu card, formula is {-9∏, +9∏, +9∏} or {-9∏,-9∏, +9∏}.

φ2a: Two-Xi card, formula is {-5∏, +5∏}.

φ2b: Three-Xi card, formula is {-5∏, +5∏, +5∏} or {-5∏,-5∏, +5∏}.

φ3: Liangzi card, formula is {x∏, x∏, x∏, x∏}.

φ4: Fish card: formula is {1∏, 2∏, 3∏}.

φ4abc:Yuliang card, formula is{1∏,1∏,1∏,1∏,2∏,3∏} or {1∏,2∏,2∏,2∏,2∏,3∏} or {1∏,2∏,3∏,3∏,3∏,3∏}.

φ5: Bai card, formula is {8∏, 10∏, K∏}.

φ5abc:Bailiang card, formula is {8∏,8∏,8∏,8∏,10∏,K∏} or {8∏,10∏,10∏,10∏ ,10∏,K∏} or {8∏,10∏,K∏,K∏,K∏,K∏}.

φ6: Xibaoliang card, formula is {+5∏,-5∏, x∏, x∏, x∏, x∏}.

φ7: Tian card, formula is {K∏}.

φ8: Hu card, formula is {Q∏}.

(3)According to the feature of finite automaton and alphabet, we design equivalent substitution below:

Let a=φ7. Let b=φ1a |φ2a. Let c=φ1b |φ2b |φ5. Let d=φ3. Let e=φ4abc |φ5abc |φ6

(4)Finite Automaton M, as is shown in figure 3.

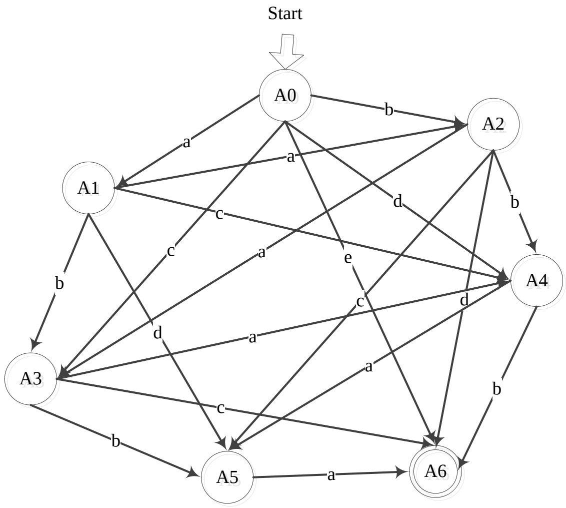


Figure 3. Finite Automation in Occupancy of Six Cards State Transformation

In finite automaton M, the process of state transformation function is as follows:

Step 1 According to the player's card type, initial state autocalls proper state transformation function, including three types named Bai card, Niu card and Xi card. And then automaton will execute state-jump according to different card type above.

Step 2 Every time automaton jumps to the next state, automaton will continue to execute state-jump until it arrives at terminal.

Step 3 If none of card types accords with the state transformation function, computer will output the hint that, “Cannot form standard card type”.

3.2.2. Algorithm design of available card type

Card type recognition algorithm of finite controller is used to judge whether available card type meets the conditions of transfer, so that it can control the skip of automaton.

Taking Bai card as an example, to mark played card in a pair of card, controller defines an array whose size is 16 and the value of every position is 0. If the position of this array is occupied, controller will change value of corresponding position to 1. In the process of judging Bai card, whether 8∏, 10∏, K∏ are coexisting is depending on a number parameter λ that adds 1 when it gets one card type mentioned before. Controller saves results to output the array when the value of λ is 3. Moreover, it skips to the next state according to the occupied number. If the value does not equal 3, controller will save card type K∏ in output array. Meanwhile, according to occupied number, it skips to the next state shown in figure 4.

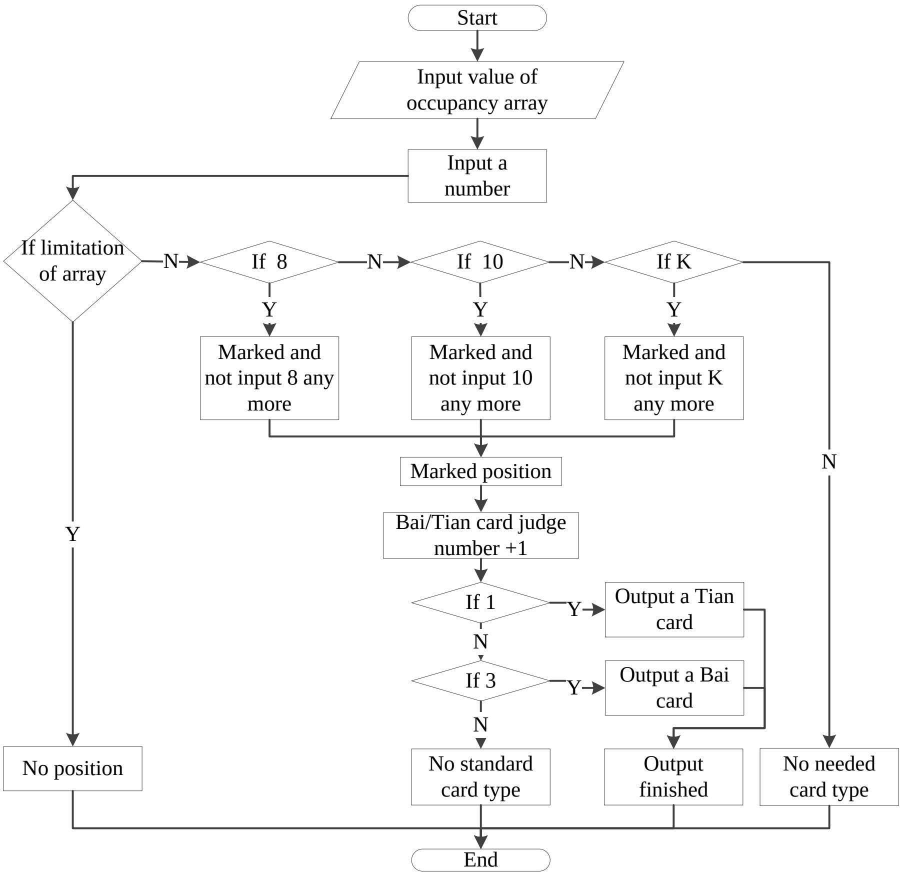


Figure 4. Recognition Algorithm of Bai Card

Niu card and Xi card remainder algorithm utilizes the process of number remainder to stimulate different situation. Taking Niu card as an example, a parameter μ is defined to receive value. When μ meets $+9\prod$, the value equals +7. When μ meets $-9\prod$, the value equals 9, which results in the following situations below:

$\varphi 1=\{9\prod | n\rightarrow 9\prod \wedge n=1\}$

$\varphi 1\text{-}1=\{9\prod | n\rightarrow 9\prod \wedge n=-1\}:\mu\%10=9$

$\varphi 1\text{-}2=\{9\prod | n\rightarrow 9\prod \wedge n=-2\}:\mu\%10=8$

$\varphi 1\text{-}3=\{9\prod | n\rightarrow 9\prod \wedge n=+1\}:\mu\%10=7$

$\varphi 1\text{-}4=\{9\prod | n\rightarrow 9\prod \wedge n=+2\}:\mu\%10=4$

$\varphi 1a=\{9\prod |(-n\rightarrow 9\prod \wedge n=1)\wedge(+m\rightarrow 9\prod \wedge m=1)\}:\mu\%10=6$

$\varphi 1b\text{-}1=\{9\prod | n\rightarrow 9\prod \wedge n=2+(-1)\}:\mu\%10=3$

$\varphi 1b\text{-}2=\{9\prod | n\rightarrow 9\prod \wedge n=(-2)+1\}:\mu\%10=5$

$2\varphi 1a:\mu\%10=2$

We can easily find that no same value appears in all possible sets which make it easier to judge Niu card or Xi card. This function can identify all required card types.

3.3. IMP-minimax and Monte Carlo based algorithm recognition of special card type

In the process of occupancy, except the situation of available card type, some special occupancy is also the available card type.

There exists much hidden information in the dealing process, so that this computer game is a game with imperfect information. Therefore, this playing process can be applied to describe special card type by IMP-minimax algorithm. However, special card types are complex and restricted due to miscellaneous conditions, so that they are usually difficult to produce. Therefore, we need to introduce a Monte Carlo algorithm to estimate the probability of special card types in order to make them quickly and accurately established.

IMP - minimax algorithm applies a timely linear search strategy for games with imperfect information. Under this algorithm this strategy is the most optimal. Specially, at first player stage and having some natural properties, we propose a search algorithm of special card type based on identification IMP – minimax. If the number of a card type held by the first player just equals the idea number of search tree, this card type is called the perfect recall.

In Niujiu card, if a player holds a certain number of 9∏, 8∏, 10∏, K∏, Fish card and Hu card will be available occupancy in some situation. Based on this situation, Let 3ɸ8=1, 2ɸ8=2, ɸ8=3, ɸ4=4 as the number of all parameter of leaf nodes valued 9∏, 8∏, 10∏, K∏. Values of right nodes are set as above four. Two search algorithms of max set in extreme cases are given as follows:

(1)Perfect limitation of max set shown in Figure 5.

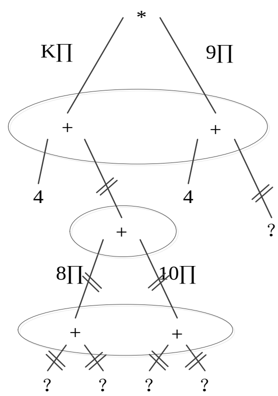


Figure 5. Search Tree of Perfect Limitation of Max Set

In Figure 5, all values of left nodes in search tree are the numbers of card type. Let the value of card type 9∏ and K∏ to be the number 4. According to IMP-alpha-beta pruning algorithm, Formula 1 shown following is worked out:

(1)

Let the expectation in search tree to be V(X). All nodes are denotated by ω. $P\pi(\omega)$ represents π set of X. In$max\{extend(Y)\}$, if Y is a child node and X is a π set, formula $\sum P\pi(\omega)H(\omega)$ substitutes $max\{extend(Y)\}$. $P\pi(\omega)$ represents the value of all nodes of π set. $H(\omega)$ represents the value of certain number leaf nodes.

As is shown in figure 5, all Pπ(ω) values of leaf nodes of left sub trees are (4+4)/2=4. According to the limitation that the same card type is 4 at most, when the value of K∏ is 4, the left sub trees of 8∏ and 10∏ can be cut. Putting the value of four card types set before to the leaf nodes of right sub trees, all value of right sub trees are 4 in an extreme case. All Pπ(ω) value of leaf nodes of right sub trees are (4+4)/2≤4 in this case. Therefore, all right sub trees are cut. Meanwhile, V(X) equals 2.

As is shown in the deduction above, when a player holds 4φ1 and 4φ7 card types, 3φ8, 2φ8, φ8 and φ4 are special types of available card type.

(2) Imperfect limitation of max set shown in Figure 6.

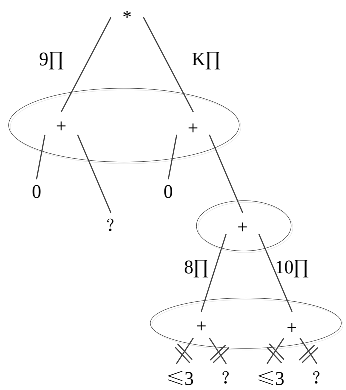


Figure 6. Search Tree of Imperfect Limitation of Max Set

According to the deduction above, right sub trees can be cut as long as the value of Pπ(ω) of right sub trees' leaf nodes is less than or equal to the value of Pπ(ω) of left sub trees' leaf nodes. Therefore, in an extreme case, when "?" in figure 8 is equal to 0, with Pπ(ω)=(0+0)/2=0 being met, V(X) is equal to 0.

With the deduction above, we can also find that, when a player holds 0ф1 and 0ф 7 and the number of 8∏ and 10∏ less than or equal to 3, 3ф8,2ф8,ф8 and ф4 cannot be special available card type as occupancy.

These two are extreme situations. In the process of playing card, with a player holding different cards, V(X) ranges from 0 to 2.

Above process has realized the IMP - minimax algorithm-based recognition of special card types. However, the presence of the special card types are not regular, and we here introduce Monte Carlo algorithm to estimate the frequency of special card type.

Monte Carlo algorithm is a calculation method based on random numbers [13]. According to the search tree of IMP - minimax algorithm, it is not hard to find the probability of perfect recall in card number 9∏, 8∏, 10∏, K∏ is much greater than ф4a and Q∏. In order to insure the accuracy of result, we choose one of the most extreme probability as the simulation times. That is to say, consecutive 4 times at the very beginning can be called perfect recall. The probability is given as follows:

So the Monte Carlo algorithm select the needle using 200000 times as a sample. The flow chart of the algorithm is shown in figure 7.

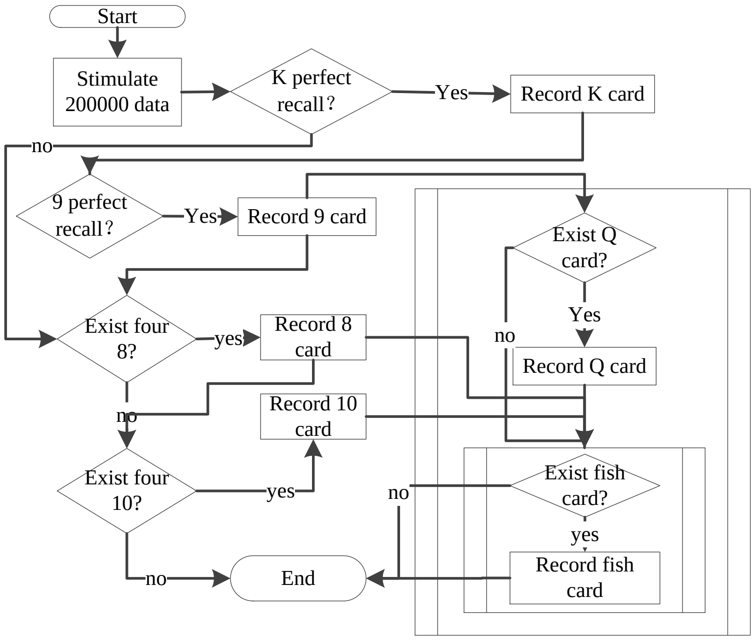


Figure 7. Flow Chart of Monte Carlo Check

According to figure 7, Step of examination of special cards type are given as follows:

Step 1 This chart first judges whether K ∏ exists perfect recall. If K ∏ exists perfect recall, the chart will choose to judge card ∏ type 9, or it will chooses to judge card type 8 ∏.

Step 2 If the chart choose to judge card type 8 ∏, and then 8 ∏ does not exist perfect recall, the chart will choose to judge card type 10 ∏.

Step 3 In the whole process, if the chart has any kind of perfect recall, it will jumps to judge special card type Q ∏or fish card. If the chart exists perfect recall of special card types, it will records the card number.

To sum up, the algorithm sums up the 200000 times of perfect recall in card types 9 ∏, 8 ∏, 10 ∏, K ∏ respectively. At the end of the statistics, the algorithm judges above statistical results of fish card and tiger card, and then computes the probability of fish card and tiger card in the case of the V (X) = 2.

Considering a variety of special card types in a simulated condition at the same time, special card type in the following two situations requires special dealing.

(1) Fish card in V (X) = 2 and 0 < V (X) < 2 both cases are ideal existence. In this case, when summing up the special fish card types, computer needs to remove duplicate fish card number.

(2) Fish card and tiger card are existing at the same time under the condition of V (X) = 2.In this case, when summing up all card types, computer needs to remove duplicate fish card and tiger number.

As is shown in figure 8, computer computes the number and probability of fish special card and tiger special card in detail. Needling under 200000 cases, the frequency of 9 ∏, 8 ∏, 10 ∏, K ∏ are roughly same, about 240000 times. Times of perfect recall of 9∏ is about 2400. Times of perfect recall of K∏ is about 800 times. Times of perfect recall of 9∏ is three times than K∏. Times of perfect recall of 8∏ and 10∏ are about 1200. Times of perfect recall of 9∏ is double than 8∏ and 10∏. Times of special type card type 9∏ is about 1200, and probability is about 0.61%. Probability of tiger card type is very low, almost to zero. In order to gain more accurate needle results, table 1 shows the result of random five times for needle.

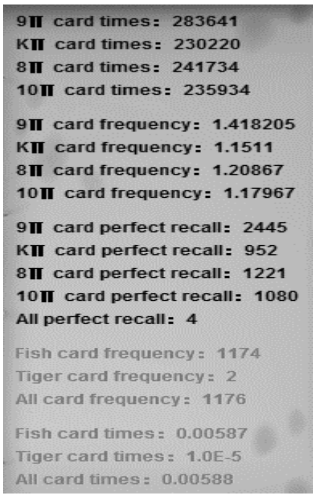


Figure 8. Statistic of Results of Monte Carlo Experiments

Table 1. 5 Times Statistic of Results of Monte Carlo Experiments

Types/Number	283289	283407	283905	283762	282778
9∏ card times	229800	230054	230125	229932	229966
K∏ card times	241898	242240	241261	241625	241365
8∏ card times	236196	235952	235951	236252	235618
10∏ card times	1.412	1.411	1.420	1.419	1.419
9∏ card frequency	1.149	1.150	1.151	1.150	1.150
K∏ card frequency	1.209	1.211	1.206	1.208	1.207
8∏ card frequency	1.181	1.180	1.180	1.181	1.178
10∏ card frequency	2441	2439	2404	2376	2429
9∏ card perfect recall	962	923	984	999	997
K∏ card perfect recall	1298	1172	1198	1202	1232
8∏ card perfect recall	1079	1138	1071	1123	1018
10∏ card perfect recall	4	6	4	7	4
All perfect recall	1233	1138	1211	1267	1206

Fish card times	4	4	4	5	1
Tiger card times	1236	1141	1213	1271	1207
All card times	0.006	0.006	0.006	0.006	0.006
Fish card frequency	0.000	0.000	0.000	0.000	0.000
Tiger card frequency	0.006	0.006	0.006	0.006	0.006
All card frequency	283289	283407	283905	283762	282778

In table 1, needle under 1000000 cases, the frequency of fish and tiger special card types are stable. The frequency of fish special card type is stable in 0.006. The frequency of fish special card type is almost zero. The above experiment efficiently computes the special card type probability within the controlled error range.

6. Conclusions

Game model and algorithm research of multi-player non-zero-sum games are always difficult issues in computer game. Taking the ancient card game named Niujiu card as an example, this paper summarizes the components of computer game model. Second, an search algorithm based on IMP-minimax is proposed respectively, combining with Monte Carlo algorithm to estimate the frequency of card type. This game model lays a solid theoretical basis for research and algorithm application of computer game. The next step will research strategy deeply and the model of second and third players in Niujiu card.

Acknowledgement

This work is financially supported by the Fundamental Research Funds for the Central Universities (No.TD2014-02).

References

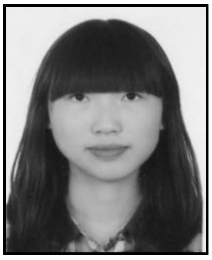
[1] J. Mutch and D. G. Lowe, "Object class recognition and localization using sparse features with limited respective fields", International Journal of Computer Vision, vol.80, no. 1, **(2008)**, pp. 45-57.
[2] X. Du, M. Zahang and X. Wang, "Self-Optimizing Evaluation Function for Chinese-Chess", International Journal of Hybrid Information Technology, vol.7, no. 4, **(2014)**, pp. 163-172.
[3] J. H Miller, "The coevolution of automata in the repeated prisoners' dilemma", Journal of Economic Behavior Organization , vol.29, no. 4 , **(1996)**, pp. 87 -112
[4] G. Qinqin, L. Shuqin and B. Hua, "Research on evaluation function computer game of Amazon", Computer Engineering and Applications, vol.48, no. 34, **(2012)**, pp. 50-54.
[5] A. Neyman and D. Okada, "Two-person repeated games with finite automata", International Journal of Game Theory, vol.29, no. 3, **(2000)**, pp. 309 -325.
[6] M. Piccione and A. Rubinstein, "Finite automata play a repeated extensive Game", Journal of Economic Theory, vol.61, no. 1, **(1993)**, pp. 160-168.
[7] G. Itzhak, "The complexity of computing best-response automata in repeated Games", Journal of Economic Theory, vol.45, no. 2, **(1988)**, pp. 342 -352.
[8] K. Zhang, Z.-W. Mo, "Minimization of lattice-valued moore type of finite automaton", Advances in Intelligent and Soft Computing, vol.147, no.4, **(2012)**, pp. 247-256.
[9] H. Masateru, Noguchi, Shoichi, "On the equilibrium points of homogeneous automata games", Electronics and Communications in Japan, vol.59, no.6, **(1976)**, pp. 37-46.
[10] V. Schantz, Anton. "Cellular automaton evacuation model coupled with a spatial game", Lecture Notes in Computer Science, vol.86, no.10, **(2014)**, pp. 371-382.
[12] W. Bruce, Ballard, "The*-minimax search procedure for trees containing chance nodes", Artificial Intelligence , vol.21, no. 1,2 , **(1983)**, pp. 327-350.

[13] E. Donald, Knuthand Ronald W and Moore, "An analysis of alpha-beta pruning", Artificial Intelligence, vol.6, no. 4, **(1975)**, pp. 293-326.
[14] B. Bouzy, "Associating Domain-dependent Knowledge and Monte Carlo Approaches within a Go Program", Information Sciences, Heuristic Search and Computer Game Playing IV, vol.40, no.4, **(2005)**, pp. 247–257.

Authors

Chuanxi Zhang, 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 major research interests include computer game 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 an associate professor in School of Information and Technology, Beijing Forestry University. Her main research interests include artificial intelligent, knowledge engineering and semantic Web.



Cong Dai, She is an undergraduate student in School of Information and Technology at Beijing Forestry University. She joined Institute of Artificial Intelligence in Beijing Forestry University in 2013 as a research assistant. Her main research interests include artificial intelligent and semantic web.