Design and Application of Tibetan Long Chess Using Monte Carlo Algorithm and Artificial Intelligence

Junjie Zhou*

School of Information Science and Technology, Bohai University, Jinzhou, Liaoning, 121007

*Corresponding author: junjiezhou2021@bhu.edu.cn

Abstract. In order to study the algorithm of Tibetan Jiuqi, a new kind of chess in Chinese computer game, there are too many possibilities to be considered in solving the problems of falling pieces, and the influence of falling pieces on Xingzi and Tiezi. On the basis of forming different chess patterns as the main source of raising pieces, this paper puts forward a design scheme which takes the Monte Carlo algorithm randomly as the core and the final evaluation algorithm as the secondary one, and takes a variety of possibilities in Tibetan chess as a breakthrough. Get different benefits from different chess types to cooperate with the final evaluation algorithm to get the assignment of each replicated chessboard. It is proved by the program computer game that the program is complete and the three stages of Tibetan chess can be completed successfully. The experiment shows that the Monte Carlo algorithm is very suitable for the AI development of Tibetan chess.

1. Introduction

Since the invention of chess, it has become more and more popular, and this game method has been successfully combined with computers. In the computer field, the first artificial intelligence game robot-"AlphaGo" has been successfully produced by using advanced algorithms. Computer games can not only make the algorithm more powerful, but also spread chess culture. The chess game introduced in this paper is Tibetan Jiuqi [1] [2], the meaning of Tibetan Jiuqi: "Jiuqi" in Jiuqi is Tibetan transliteration, which means jigsaw puzzle. Jiuqi is played in the form of layout, medium raisin and last Feizi. Jiuqi has a long history and has strong interest, competition and vitality. Jiuqi is the heritage of Tibetan long history and heavy cultural precipitation. In 2006, Jiuqi was listed as the first batch of intangible cultural heritage list in Qinghai Province, including 11 Road, 14 Road and so on. At present, the artificial intelligence of chess is developing rapidly, but due to the lack of research literature on Jiuqi, there is no research on Tibetan chess and other famous chess. In this paper, the AI of Tibetan chess is studied by taking the Monte Carlo algorithm as the core and the global final evaluation algorithm as the second model.
2. Introduction of algorithms in Chess

2.1. Monte Carlo algorithm
Monte Carlo algorithm [3], also known as statistical simulation algorithm, is a kind of very important numerical calculation method under the guidance of probability and statistical theory. The main purpose is to establish a probability model, and then use this algorithm to simulate the process over and over again to produce a random variable of a certain probability distribution. Then the statistical method can be used to estimate the digital characteristics of the model, and the estimated value can be used to analyze the practical problems and obtain the optimal solution with the maximum probability, which is suitable for the problems with the theory of probability and statistics as the core.

However, it is very suitable for chess, which is a probability problem of playing chess, which coincides with the model of Monte Carlo algorithm. In the realization of Tibetan chess for a long time, blank positions will appear in four stages, and each blank position will lead to different directions of the chessboard. It is very suitable to establish a probability model using Monte Carlo algorithm.

2.2. Final evaluation algorithm
The final evaluation algorithm is essentially a self-defined function, which assists the Monte Carlo algorithm to calculate the final assignment in each probability, which directly affects the final choice of the optimal solution. This algorithm is an indispensable part of the model.

In chess, it is suitable for Tibetan chess, Gobang, which has a variety of chess layouts, because the final evaluation algorithm can assign values to different chess layouts, resulting in a variety of different assignment results, representing the value of each move, so as to better choose the highest probability optimal solution.

2.3. Final evaluation algorithm
Alpha-Beta pruning algorithm is a core search algorithm that reduces the number of nodes evaluated by minimax algorithm in the search tree. It is suitable for human-computer game confrontation. According to the optimal solution of the previous step, it continues to further evaluate the situation of the next round and get the optimal solution of the second round. In chess artificial intelligence, Alpha-Beta pruning algorithm is also the most frequently used recursive algorithm, Alpha is the best value, and Beta is the worst value.

If the result of a play is less than or equal to Alpha, it is a very bad play; if the result of a play is greater than or equal to Beta, the whole node will be invalidated; if the result of a play is greater than Alpha but less than Beta, the move can be considered by the chess player. But there will be exceptions, when the second selection and deduction must be carried out on the first layer of the game tree in this algorithm.

2.4. B* algorithm
The B* (Branch Star) algorithm (A* algorithm optimization) mainly studies two problems on the minimax search tree: one is how to reduce the complexity of the combinatorial search and delete the useless branches in advance; the other is how to reasonably determine the depth limit of the search and predict the game tree of the algorithm, so as to make the whole process run more stably and accelerate the research progress.

In the B* algorithm, the deleted data cannot be recovered, and the problem of over-fitting is easy to occur, while the Monte Carlo algorithm can list most of the possibilities and use the final evaluation algorithm to calculate the assignment and comparison, which is more accurate.
3. Introduction of algorithms in Chess

3.1. Falling rule
The grid in the center of the chessboard uses the points connected diagonally as the starting point for both sides of the game. Begin to play the sub-layout in turn, and the chess starts after the pieces are all over the chessboard.

![Tibetan chess board.](image)

3.2. Line subrule
When playing chess, first remove the pieces at both ends of the diagonal of the central square of the board and start playing chess. The black side takes the lead in the move, (because the white side took the lead in the previous layout).

3.3. The rules of Tiezi and Feizi
The adjacent point of the vertical (horizontal) line of the pawn is the pawn of the other party, and when the third point is the blank point, the pawn can jump and eat the pawn of the other side. Dancing and eating each other's chess pieces can be eaten alone or even.

(1) Chess Gate: there are a total of 196 squares on the chessboard. When playing chess, the four vertices on each grid are square pieces, forming a smallest square and four vertices are called a chess gate. Chess is that each time a chess door is formed, any piece of the opposing team can be eaten. The chess door is divided into single chess door and double chess door. The double chess door is formed by the connection of two single chess doors. When playing chess, a double door is formed every time. Can eat any two pieces of each other.

(2) Single door: if there is a space between two doors and there can be a piece that can close both sides of the door, this form is called a game. (2) Single door: if there is a space between two doors and a piece can close both sides of the door back and forth.

It is the basic condition to guarantee the chess action needed to win chess.

(3) Double chess: if there is a space between two chess doors, and you can move one piece back and forth, you can close two chess doors and eat any two pieces of each other.

This is also the basic formation needed to win chess.
4. Grab: if you walk a chess piece and a straight line is full of each other's chess pieces, you can eat any two pieces of each other's chess pieces. This chess shape is called "Wu" (meaning "grab") in Tibetan.

5. Evil spirit: if you walk a chess piece, all the pieces on this straight line are all on one side and form a single or double door, you can eat three or four pieces of each other.

6. Lan: if the pieces on two straight lines are all on one side, and when forming a double door, the pieces on both sides can play chess at the same time and can eat each other's eight pieces at random. This chess shape is called "Lan" in Tibetan (meaning wearing shoes).

7. Exactly: when walking a chess piece, the pieces on the vertical and horizontal lines of two different directions are all on one side (cross, horizontal to side, vertical to the bottom) and can eat any four or eight pieces of each other when forming a single or double gate, this chess shape is called "cha" (meaning cross).

8. Songran: if the pieces on the three parallel lines are all on one side, and double doors are formed on both sides, when the four-stage doors are formed in the middle, the pieces on both sides and the pieces in the middle can play chess at the same time, and any 12 pieces of the other party can be eaten. This chess shape is called "Songran" (meaning three rows).

9. How: this is the last move, and if there are only 14 pieces left, the Tibetan side of the chess form is called "how" and the player can move at will and is not restricted by the pace of the game. (9) How: this is the last move. If there are only 14 pieces left, the one who is called "how" in Tibetan can play at will and is not restricted by the pace.

10. Zinger: (meaning to squeeze or squeeze out) in chess, when one side's piece is blocked by the horizontal and vertical walking of the other side, that side loses. (10) Zinger: (meaning to squeeze or squeeze out) in chess, when one side's piece is blocked by the horizontal and vertical walking of the other side, that side loses.

4. The Application of Monte Carlo algorithm in the four stages of Tibetan Chess

4.1. Falling seed
Luozi is the first stage of Tibetan chess for a long time, and the first thing for Tibetan chess is to fill all the blank positions on the chessboard before the next stage can be carried out. In the falling stage, the most important algorithm is the Monte Carlo algorithm, the model of the Monte Carlo algorithm has been introduced in the above module, and the layout of Tibetan chess is very consistent with this model.

Its main core idea is to find the space position on the chessboard, record its coordinates, try to drop pieces in each space, such as playing chess in the (APower1) space, and then make a copy of the chess. Make the next prediction on the copied board, and generate a certain prediction assignment, and this assignment will be called to the global final evaluation function mentioned in this article.

Using this function, you can assign the corresponding value to the situation in the copied chessboard, and it is convenient to finally choose the falling son method with the highest probability. There are six methods to assign the settings in the global final evaluation function: the basic assignment of each grid is 3, the assignment of each chess gate is 10, the big chess gate is 10, and the half chess door is 5.

4.2. Line seed
Xingzi is part of the second stage of Tibetan chess. After the chessboard is covered with pieces and the two pieces of the central diagonal are removed, this stage begins. The main idea is that at this stage, it is still used to find the blank position first, to copy the chessboard according to the coordinates of the blank position, to make a prediction model on the copied chessboard, and to calculate the value of each kind of chessboard. To determine the final optimal solution, if after the line, if there is the emergence of the chess door, then carry on the lifting stage, and re-assign the overall situation after raising the piece. Due to different computer configurations, the number of random simulations can be customized. It is recommended that the number of random simulations be 20000 to 50000. Too many times will cause the computer to stutter and cause the program to crash.
4.3. Lift seed
The pick-up is after the end of the row-piece stage in 4.2, if a chess door is formed, it enters the pick-up stage, and the pick-up stage begins to distinguish between black and white from (AReagle 1). If it is a chess piece of different colors, it begins to copy the chessboard and predicts the level of the assignment after the chess piece is removed.

For example, in figure 2, after the end of the line, you can first mention the albino of (FForce 3), because as long as you get rid of it (FForce 3), you will create a double-edged chessboard, which will make the final assignment of the global final function to this duplicate chessboard much higher than that of other replicated chessboards, and mention the third son of the other party, reducing the other party's assignment and indirectly increasing your own side's assignment.

![Figure 2. Black cube and prediction of the next step.](image)

4.4. Flying son
When there are 14 pieces left on one side, when entering the flying stage, we need to find and record all the blank positions. In the application of the local Monte Carlo algorithm, each copy chessboard uses the global final function to calculate the final assignment.

After flying, you need to judge whether it is directly into the chess door and enter the pick-up stage of the code:

```c
i=x2;

j=y2;

if(i-1>=0&&core[i-1][j]==var) {
```
if (j - 1 >= 0 && core[i][j - 1] == var) {
    get_score = get_score + get_three;
}

if (core[i - 1][j - 1] == var) {
    eat_num++;
    get_score = get_score + get_four;
    if (get_lian == 1) canWalkChess.setGet_lian(1);
}

if (j + 1 <= 13 && core[i][j + 1] == var) {
    get_score = get_score + get_three;
}

if (core[i - 1][j + 1] == var) {
    eat_num++;
    get_score = get_score + get_four;
    if (get_lian == 1) canWalkChess.setGet_lian(1);
}

if (i + 1 <= 13 && core[i + 1][j] == var) {
    if (j - 1 >= 0 && core[i][j - 1] == var) {
        get_score = get_score + get_three;
    }
}
eat_num++;
get_score = get_score + get_four;

if(get_lian==1)canWalkChess.setGet_lian(1);
}
}

if(j+1<=13&&core[i][j+1]==var){
get_score = get_score + get_three;
if(core[i+1][j+1]==var)
eat_num++;
get_score = get_score + get_four;
if(get_lian==1)canWalkChess.setGet_lian(1);
}
}

5. Conclusion
Tibetan chess for a long time as one of the better preserved and inherited Tibetan chess, in the computer
game, compared with go, chess and other chess is far away, Tibetan chess still needs to be studied and
innovated.
This paper mainly compiles the program according to the chess type of Tibetan chess and the basic
code of Gobang, and applies the Monte Carlo algorithm and the final evaluation algorithm to the
computer game of Tibetan chess, and describes in detail how to apply the algorithm to the four stages
of Tibetan chess, although this program cannot fight with the master thoroughly, but the program is
stable and the content is complete.

Reference
[1] Ma Shaolong, Shang Tao. The inheritance and protection of Tibetan chess culture. College of
physical Education, Northern University for nationalities. Tibet Autonomous region Tibetan
Chess Association. 2017 (04).
[2] Liu Qiang. Tibetan Chess-- A textual Research on the Origin of Jiuqi. School of physical
Education and Health, Aba normal University. 2017 (06).
[3] Zhu Lulu, Li Bo. Monte Carlo method and its application. Hubei: central China normal University,
2014. Master's thesis.
[4] Onion_ywc original. Alpha-beta pruning algorithm and practice. Original reference: Zhu Fuxi.
Basic course of artificial Intelligence (second Edition). CSDN.March 24, 2018, 20:44.
[5] Li Xiali, Wu Licheng, Li Yongji. Research on Tibetan "Jiu" chess computer game based on chess
type. Journal of Intelligent Systems, 2018, Volume 13, No. 4. Beijing: central University for nationalities, August 2018.