The Design and Realization of Connect-six Computer Game Software

Xinyang Wang [1, a], Xiali Li [1, a], Rui Wu[1, a], Hailong Xie [1, a]

[1] Department of Information Engineering, Minzu University of China, Beijing, 100081, China

[a]xiaer_li@163.com

Keywords: Computer Games; Connect-six game; Search Algorithm; Window-based Alpha-Beta Pruning Search Algorithm; Evaluation Function

Abstract. This article design and implement the new connect-six computer game software. We use C# language to develop a man-machine program with the optimized search engine, move generator and situation evaluation function. We use window-based Alpha-Beta pruning search algorithm and evaluation function based on present move.The tests show that the software we developed is effective and with high level.

Introduction

Computer game is one of important research fields in artificial intelligence and connect-six has attracted more and more the interests of researchers as a new emerging chess game. However, it's easy to learn but hard to master. One person set two moves each time (except the first move), which offers the game intensive attack and variety. With flexible chess road and no complex rule like gobang, connect-six win high popularity among the folk [1].

Currently, famous researches on connect-six are the connect-six computer game program named NCTU6 (National Chiao Tung University 6), which was designed and developed by the research group led by Professor Yicheng Wu. This program adopts α-β search tree, depth 3, and combines threat-space search based on double threats[2]. Threats refers to: supposing part A need to move T's pieces to prevent part B from K's pieces, then we call part A has T threats; if a threat comes out after one side moves a piece, then we call this piece as single-threat move[3]; if two threats come out, then we call the certain piece as double-threats move; if triple-threat move comes out, then the side wins. For connect-six game, to win means to block all the threats from one's rival, and at the same time make at least three threats. An efficient strategy for winning the game is to combine real threats with highly possible potential threats.

Besides NCTU6 other existent programs for connect-six computer game are EVG by former NTU professor ShunQin Xu team and X6 by Siyuan Liu and Shijing Yan. Both of them adopt the strategy discussed in reference[4].

Structure of Game Program

The program designed for computer connect-six game mainly contains four parts as follows: board generating, move generating, algorithm searching and value accessing. To choose and design a most suitable searching engine for the program, we need to test, analyze and adjust existent searching algorithms, as well as optimizing the engine. In testing and debugging the program, we optimize it by ways like manual setting, man-machine game etc. We also debug multiple parameter and weighs, and the ways of assessing values.

2.1 board representation

In order to quickly access the position of each piece, this project will adopt 19*19 two dimentional matrix to represent the chess board. According to symmetry, we code pieces of both sides to make operation like value-assessment convenient. As for the programming language, we choose C sharp language, which combines VB's easy visual operation and high efficiency of C++. They are helpful for designing the program.

2.2 move generating

The operation of program requires frequent and complex judgment as well as searching, and this process is the bottleneck in software performance. The move generation module mainly solves the problem of determining an optimum moving method in scores even hundreds of ways. This project will use board scanning, template matching and preset table method to generate function, and finally get an optimum move method.

2.3 evaluate function

It's necessary to do the value accessing of present situation, because it influences the developing tendency of game condition. We use evaluate function to judge the present situation and mark it. If this assessment is not accurate or even erroneous, then even a deeper searching will do no good to help win the game. Thus, this project will study and improve the traditional evaluation on pieces value, pieces position and pieces flexibility to reach a more precise value assessment and boost the program's performance.

2.4 searching engine

In order to find a situation with highest evaluation and get the best moving method, we should find out an efficient and rapid searching algorithm. This is the core of game, also a key factor for success. Traditional depth-first algorithm can only get a situation in a superficial scale with low pieces value. This project uses Alpha-Beta pruning search to access match pieces, and then decide which piece to move.

Window-Based Alpha-Beta Pruning Search Algorithm

All the computer game of chess involves searching algorithm. It's used to find the optimum path in game tree according to present situation and the set searching depth and width. Game tree is a list of all the possible moves and situations for computers and users. Both sides alternatively and rationally expand the tree, and each node of the tree represents a certain situation.

Root node represents the present situation which needs to be calculated, while middle node represents a certain situation in the process of game; and leaf node is the bottom of the tree, representing the deductive situations. The largest distance between leaf node and root node is the searching depth. The whole game tree represents a searching tree including all the possible game, which take present situation as a starting point.

The searching engine module in connect-six computer game system needs to do following tasks: make sure that the searching algorithm can be used; add heuristic information about searching to the tree; try to narrow the scale of game tree in order to avoid redundant calculation. Other computer games also need to take this problem into consideration.

Due to the specialty in connect-six that two sides move two pieces each time, so they need to think about the comprehensive effect of the two moves. Common searching algorithm must take two moves as a single move, so how to change "two" into "one" is a problem about comprehensive search [5]. In order to narrow the tree's scale and avoid all the redundant calculation to increase the searching efficiency, we must filter nodes in the game tree to exclude unnecessary search. So the program designed in this article adopts α-β pruning searching algorithm.

Alpha-Beta pruning chooses the optimum path in all the feasible paths[6]. In man-machine game, choosing the position of pieces is of the biggest significance. For different results, we accordingly make judgments to different levels. In searching the game tree, size of some nodes have no influence on their value. Before searching the value of a node, its effective range has already set, and we call it a window. We use Alpha to represent the lower bound, and Beta the upper bound. In the process of searching, the child nodes with smaller value than Alpha will be cut out, and the child nodes with larger value than Beta will put the search on end. It's called α-β pruning[7].

Evaluation Function Based on Present Move

Evaluation function[8] generally consists of five elements, including value of fixed pieces, piece position, piece flexibility, threat and protection, and dynamical adjustment. Each value is also composed of values from several aspects. Put all effective combinations of linear values together, we will get a final assessing value.

The evaluation function of connect-six is just a subset of five elements. It’s only related to the position of present piece and surrounding pieces, so it can be simplified. Firstly we start from the present move and present situation of the board, and take the present move as the center, scanning those points with step length 13 in fixed length in four directions(horizontally, vertically, left oblique, right oblique)respectively. Then we will get state arrays of the four directions and each has a length of 13.In those arrays, each position has four values:-1,0,1,2,which respectively represents outside board, black piece, white piece and no piece. The evaluation function of present move is:

$$value = value + aStepStatus[i] + 1$$
$$i = 0, 1, \cdots, 12$$

In the function, value represents the assessing value of a certain move, whose initial value is set by man. AStepStatus represents the state arrays in four directions, and each value maps a step Value. Step Value is got comprehensively from four directions, horizontally, vertically, left oblique, and right oblique.

Through the program designed above, we get a stable assessing system for connect-six.the following Tab1 . and Tab2. are the testing result of assessing system with NCTU6 by professor Yicheng Wu together with his team; with EVG by former NTU professor Shunqin Xu and his team; with X6 by Siyuan Liu and Shijing Yan. In these tables, T is the time parameter of each piece move.

Tab1. Test results (T=10s)

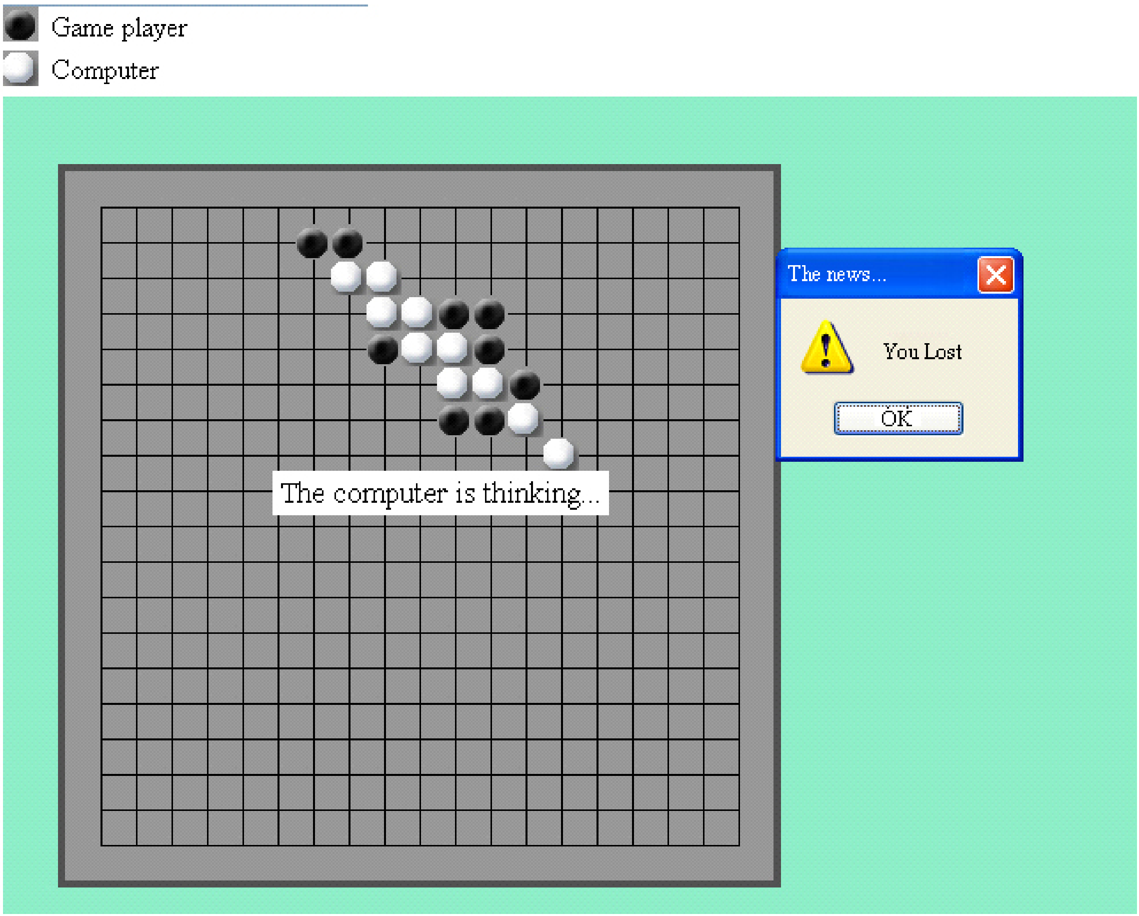
On the offensive	Defensive position	Results
Evaluation Function	NCTU6	5:0
NCTU6	Evaluation Function	0:5
Evaluation Function	EVG	8:0
EVG	Evaluation Function	0:10
Evaluation Function	X6	7:0
X6	Evaluation Function	0:7

Tab2. Test results (T=20s)

On the offensive	Defensive position	Results
Evaluation Function	NCTU6	11:0
NCTU6	Evaluation Function	0:7
Evaluation Function	EVG	1:0
EVG	Evaluation Function	0:3
Evaluation Function	X6	6:0
X6	Evaluation Function	0:10

Test and Analysis on Pieces Value of Game Program

The connect-six PC game program above can be running on computer, with humanized operation function, friendly contact surface, and higher pieces value. It won the 3rd position in 2014 ‘Cheng Li Cup’ National computer Contest for College Students. The program can play chess with high-level connect-six lovers, with high searching efficiency and fast operation speed. The website of online connect-six game is: http://10.119.64.33:8008/Chess.aspx?u=b .The following figure 1 is the online demo.



● Game player

○ Computer

The computer is thinking...

The news...

You Lost

Fig1. Online demo

This program still needs improving its pieces value, and the followings are several measures:

(1)Raise live four weights and live three weights to overcome its low menacing in the beginning of the game.

(2)Calculate two pieces comprehensively of each time, if it comes out a double live four, then give it a weight only smaller than five combs. If it cannot, then give a live three or live to, in order to create a double live four.

(3)Raise the weight of live four and live three, enabling the program to make live four when possible and disperse the rival's pieces. If there is no condition for live four, then it will create more opportunities for double live four later. The program can disperse the rival's pieces, wakening its threat, while it keeps finding out a way to win.

Summary

This article designed and implemented a simple connect-six game program, which won the 3rd place in National Computer Game for Colleges and achieved the goal of online game. However, it needs to work much to improve the performance of the game.

Acknowledgement

This article is supported by the practical innovating project "design and realization of connect-six man-machine game program based on Android and website" in Minzu University of China. This work is also supported in part by 2013 Beijing university youth talent plan No. YETP1294.

References

[1] Wenjie Min. Research on Key Tech of Connect-six Computer Game [D] Chongqing: Chongqing Jiao tong University 2010

[2] Li Guo. Research and Realization of Connect-six Computer Game and its System [D] Chong Qing: Chong Qing University, 2007

[3] Jiping Huang, Zhang Dong, Miao Hua. Research and Realization on Connect-six Intelligent Game system [J]: Computer Knowledge and Technology (Academic Exchange), 2009, (9):7198-7200

[4] Qiying Xiao, Zhengzhi Wang. Game Tree Searching and static evaluation function [J] computer application research, 1997, (4).

[5] Zhangtang Li, Huamin Chen. Game tree Parallel Searching Algorithm [J] parallel search, 1998, 19(10); 53-56

[6] Xing’er Zhang, Zhengyu Pan. Target-Oriented Optimum α-β Searching Strategy and its application in game problem [J] software journal, 1993, (04).

[7] Yang Tao, Zhijun He, Ruizhao Yu. Game Tree Function and its optimal calculation [J]. Computer Journal, 1988, (02).

[8] Xiaochuan Zhang, Guangnian Chen, Shiqiang Zhang, Kejun Sun, Zushu Li. Estimating Function of Connect-six Game [J], Journal of Chongqing University of Technology (nature and science), 2010, (02).