**3.2 The design and implementation of desktop version user interface (1500)**

**3.2.1 Frame Structure**

The overview of the developed code hierarchy is as follows. Firstly, there is a window interface including a menu, a chessboard, a record board and a function button. Secondly, the chessboard controls pieces and records their locations. Thirdly, the record board not only notes each movement down but also shows tips and search results. Lastly, the engine is packed as a thread running before each step is done. The UI design is not only considered about the feasibility and practicality, but also appearance according to other excellent designs.

**3.2.2 User Interface**

Swing packages in Java constitute the whole GUI for desktop part. Desktop version GUI mainly has four parts, which are menu, chessboard, button and record board as shown in Figure 3.2.2.

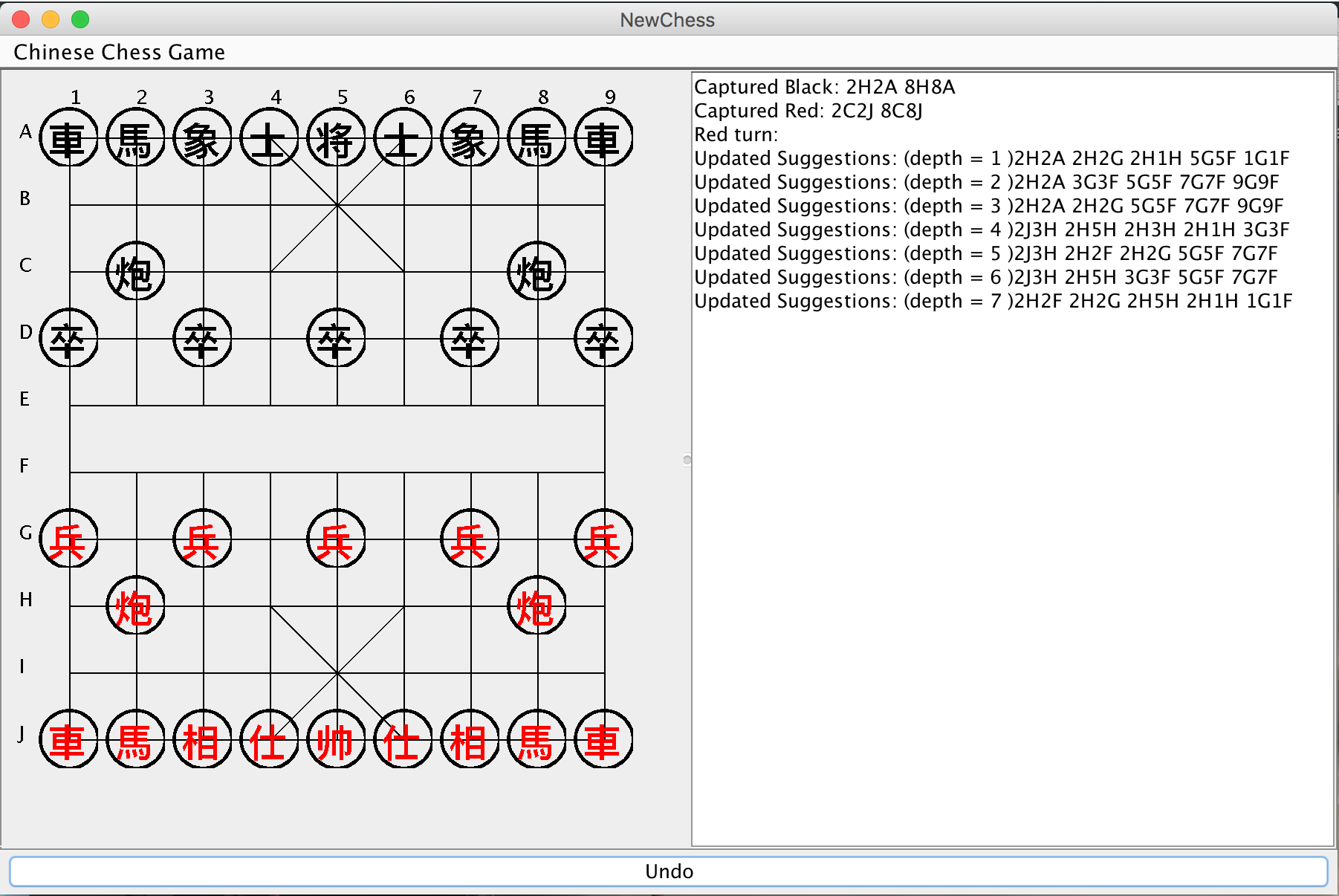


Figure 3.2.2

In the upper left corner, there is a menu bar, by which users can exit or restart game in the menu.

The left half of the interface shows the graphical chessboard. Most chessboards use numbers or alphabets (Bodlaender, H. and Duniho, F., 2014). Some prefers lowercase letters while others prefer uppercase letter. It uses ‘A’ to ‘J’ and ‘1’ to ‘9’ to give a certain and clear point in the board when the historical points are listed. The game of Advanced Xiangqi has two sides, with one being black and the other being red. As the rule shows, red goes first and user could start first or later. If the user moves piece invalid or chooses wrong side, there is an alert window, which gives a warning on it. When we click one piece, the color will change to blue to tell users that they have selected a piece .

Right half of the interface is the record board, containing these pieces that would be captured, next movement tips and this turn color.

At the bottom of the interface, an “Undo” button can delete the last movement and move the last moved piece back.

**3.2.3 Code Development**

### 3.2.3.1Chessboard

This Advanced Xiangqi game is like other board games, including graphical interface, a board and an engine. Mentioned the three chessboards used in this game, there are three aspects.

As for the graphical chessboard, simple lines are drawn in the graphical interface. Most chess games even adopt 3D images so as to beautify the User Interface. Compared with this method, lines are needed more calculation and testing. The more significant thing is that Swing can also create the board, which was learned in last semester. The second board uses a normal 2D-list. Each coordinate has its “Point”, which might have piece on it.

### 3.2.3.2Chess Piece & movement

Similar to the chessboard style, a circle and a Chinese word in it make each piece through Swing method.

In that case, clicking event responses when the mouse clicks in the rectangular area that as the circlecenter as the same center and uses the diameter of the side length. In contract to image clicking, “Rectangle” makes code more complicated. Each movement would be recorded in the “ArrayList”, after being tested in rule system. Likewise, engine would start the thread after completion of one movement. That is because of “Undo” function and graphical changes.

### 3.2.3.3Record board Content

As long as the game starts, record board would start to show the searching results of engine and the pieces that might be captured in next turn. In terms of engine thread, record board would add one suggestion on after which engine has searched one depth deeper. The depth would plus one when half-turn is finished. This kind of record board is not equipped in almost online Advanced Xiangqi games. It is not easy to meet the requirements of a beautiful interface and the required functions. In order to list more suggestions and cues, player can wait longer. After the testing, it usually takes 10 seconds to do Depth-First-Search and each depth equals to 5s. About 1-2 minutes, depth would increase to 6-7. With the dramatically growth of search size, suggestion would appear much slower.

### 3.2.3.4Undo Button

“Undo” button is the only one applied in the interface, which could be used operating miss that player takes movements back. In the traditional Xiangqi game, there is no chance to take pieces back, even operating mistakes. This function is particularly designed for friendly match. It is necessary to point out the differences between the rules for real game and the desktop version.

This “undo” function deletes the last movement recorded and restore to the original state. Cue words would appear in the record board so that you can avoid to repeat this movement. It is no direct correlations with engine. But it is still need to call undo function in engine. Once calling the engine, the thread of engine has to be stopped and then restarted.

**3.2.4 Modifications**

With in-depth understanding of Advanced Xiangqi and the functions that users need, the scientific methods and requirements also change or become obsolete. This paragraph mainly introduces the original design and the final output, including modifications and their reasons.

### 3.2.4.1Interface graphics

In the original design, the first modification is that there are only a few cue words for players. Considering the output of suggestions, we have to build a wider board to contain these tips.

Secondly, buttons are designed to foresee the next three steps, check possibility of the selected piece and give feedback, give the next step advice from the temporary chessboard situation and undo last movement respectively, which are “Show”, “Check”, “Search” and “Undo”. From their usage, it is easy to figure out that “Show” is similar to “Search”. Consequently, two buttons are combined and only remain “Search” which is based on the result of “Show”. Because it is difficult to measure the feedback in the “Check” button that how to evaluate and what criteria to define.

### 3.2.4.2Thread running

Due to the unpracticed running multi-thread in the Java files, single thread was adopted as other work we did. In that case, only clicking buttons can make engine run. Eventually, multi-thread between engine and UI surface is used, after being suggested from supervisor.

### 3.2.4.3Details added

The easiest-to-use and most intuitive feedback is the graphics in the interface. Obviously the color of circles could be changed easily; perhaps because of this reason, highlighted clicked piece is added in the final version. Similarly, pop-up boxes with information could also give direct response to players.

In the record board, movement contains specific coordinate, piece name and colour while suggestion updated only has coordinates. Because the numerous pieces mentioned, it is not appropriate in the same way. After repeated comparison tests, coordinates are remained and other unnecessary information is deleted.

**3.2.5 Problems**

Even though we have made a great effort on this programme, there are still some problems on desktop version. It would be discussed in two different situations: one is bug remaining and the other is not found.

Desktop version goes through the longest time and indeed has enough time to debug all problems that have been found. Some outcomes failed to live up to the expectation. The first that has to be pointed out here is that the UI design is not beautiful only using Swing, not to mention the unsuitable colour. The second is the output in the record board and this version is the most concise and clear style. Then, with regard to codes, some useless ones should be cut and the variables used should be narrowed down even further. In that case, the structure of the whole programme is not reasonable. For example, it needs ArrayList to record moved pieces and their locations for the “undo” function. Because of the limited variables used in ArrayList, we have to create two ArrayLists to record the moved pieces and captured the information concerning their names, colors and coordinates.

There are still some suitable or user-friendly places that need to be improved. Here are some examples below.

1. Chess piece looks ugly but no solution has been found.
2. The English explanation seems good but not concise enough.
3. When player clicks “Undo”, the thread with engine could not continue the former search but restart which would cost more time.

|  |  |  |
| --- | --- | --- |
| Time | Task | Status |
| 2015.9.18 | Timeline first made | Finished |
| 2016.9.25 | After first meeting with supervisor, timeline was changed | Finished |
| 2015.9.28 | Check the timeline | Finished |
| 2015.10.5 | Check the needed equipments | Finished |
| 2015.10.9 | Start work as planning -- research | Finished |
| 2015.10.30 | Start Desktop version | Finished |
| 2015.11.30 | Finish Desktop edition UI and engine | Finished |
| 2015.12.1 | Start Interim report | Finished |
| 2015.12.7 | Finish Interim report draft | Finished |
| 2015.12.10 | Meet with supervisor about the interim report | Finished |
| 2015.12.13 | Improve interim report | Finished |
| 2015.12.14 | Finalization | Finished |
| 2015.12.18 | Submit interim report | Finished |
| 2016.1.20 | Start android edition (learn android language by ourselves) | Finished |
| 2016.2.27 | Finish android edition UI part | Delay（Finished on March 30） |
| 2016.3.3 | Finish android edition movement | Delay（Finished on March 30） |
| 2016.3.10 | Finish android edition engine | Finished |
| 2016.3.17 | Finish android edition part | Delay（Finished on April 15） |
| 2016.3.28 | Start testing | Delay |
| 2016.4.20 | Work for presentation and improve GRP final report | Finished |
| 2016.4.29 | Final GRP work and report deadline |  |
| 2016.5.1 | Final presentation practice |  |
| 2016.5.4 | Open day and presentation |  |