A Game Informatical Analysis of Dark Chess by Game Refinement Theory

Shuo Xiong[1], Aoshuang Ye[2](✉), and Hiroyuki Iida[3]

[1] Huazhong University of Science and Technology, Wuhan, China
xiongshuo@hust.edu.cn
[2] Wuhan University, Wuhan, China
yasfrost@whu.edu.cn
[3] Japan Advanced Institute of Science and Technology, Nomi, Japan
iida@jaist.ac.jp

Abstract. This paper explores the evolutionary changes of Chinese chess variants such as Chinese dark chess. A computer program is created for each variant and self-play experiments are performed to collect many data such as the average number of possible moves and game length. These data are analyzed to examine the degree of game sophistication, while game refinement measure is employed for the assessment.

Keywords: Game refinement theory · Chinese dark chess · Flipping strategy

1 Introduction

Chinese chess and Chinese dark chess are all Chinese board games. These two games share similarities in aspects like pieces, rules, board size and etc. In this paper, we introduced game refinement theory as measure method to discuss the potential connections between two games. About the experiment, we focus on Chinese dark chess, which data of Chinese chess can be collected from the past research. In the Sect. 1, we introduce the Chinese dark chess and propose a variant version of it. In the Sect. 2, we introduce the game refinement theory. In the Sect. 3, we clarify the methodology about experiment which conducted on Chinese dark chess. In the discussion part, we compare the experiment data and find the possible potential links between two games [10].

1.1 Dark Chess

Chinese Dark chess has another name which was called "An Qi", "An" means dark and "Qi" means chess. In Chinese dark chess, players only use half part of the board, consists of 8×4 squares, totally 32 squares and for each player has 16 pieces and squares. The pieces as same as normal Chinese chess rule, which include one "Shuai" (means king, marked as K/k), two "Shi" (means

H. Santos et al. (Eds.): SmartCity 360 2019, LNICST 323, pp. 358–368, 2020.
https://doi.org/10.1007/978-3-030-51005-3_30

guard, marked as G/g), two "Xiang" (means Bishop, marked as B/b), two "Ju" (means rook, marked as R/r), two "Ma" (means knight, marked as N/n), two "Pao" (means cannon, marked as C/c) and five "Bing" (means pawn, marked as P/p) [1,2,6–8,10].

At the beginning of the game, all the pieces are randomly placed on the board with the chess icon facing down, so the type of the piece is unknown. When playing the game, the two players alternately move. There are two kinds of actions: first one is flip action, showing an unknown face down state; second one is moving action, moving a color displayed by yourself from the start point to the target point. For a cannon, it can skip another piece for long distance movement. An unknown fragment that is flipped is called a revealing fragment. From the above discussion, we know that the first player in Chinese Black must flip the pieces at the beginning.

With the exception of cannons, all types of debris can only move up and down in a 32 area to move or capture other fragments within one square. The cannon moves in the same way as other pieces, but when capturing pieces, like Chinese chess, they need to skip a piece to capture pieces of any distance in the same row or column [2]. The portion of the cannon that is skipped by the cannon is called a carriage. Chinese black chessboard is shown in Fig. 1.

As shown in Table 1. Each piece has a rank, A higher rank piece can capture the equal or lower rank pieces. However, there are some exceptions rules as follows:

1 Pawn is the weakest, however it can capture the strongest piece king
2 King is the most powerful but it could not capture pawn
3 Cannon cannot capture pieces directly, it has to jump over one piece

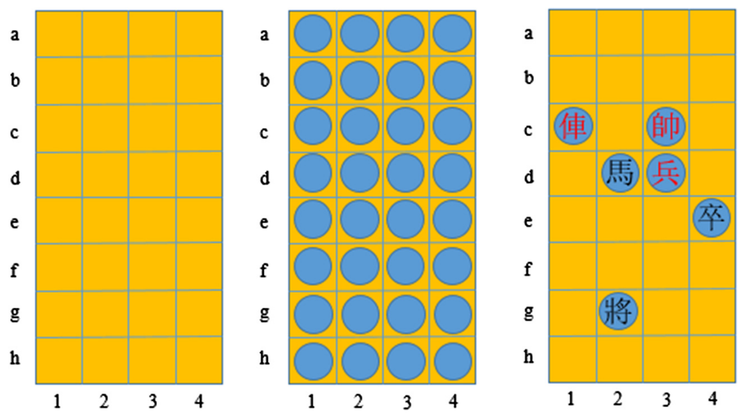

Fig. 1. Chinese dark chess board 4×8

The player wins when the opponent has no legal moves or all chips are captured. When neither player captures or reveals the piece within 40 steps, the game ends in draw. Repetition of positions also results in a draw. The state space complexity and the game tree complexity of Chinese Dark Chess were estimated to be 10^{37} and 10^{135}, respectively [2]. The game tree complexity is smaller than Chinese chess.

Table 1. Dark chess

Chinese Name	English Mark	Rank	Icon	Exception
Shuai/Jiang	King	6	帥 將	The highest rank but can't capture pawn
Shi	Guard	5	仕 士	
Xiang	Bishop	4	相 象	
Ju	Rook	4	俥 車	
Ma	Knight	2	傌 馬	
Bin/Zu	Pawn	1	兵 卒	The lowest rank but can capture King
Pao	Cannon	s	炮 砲	Capture all types of pieces by jumping over

1.2 Perfect Information Version of Dark Chess

In this part, we have imported a possible version of Chinese dark chess, which removed the flipping part from the game. Without flipping, the perfect information part of Chinese dark chess game can verify the game sophistication in another aspect. Without flipping, game strategy would be more concise. Chinese dark chess with perfect information shares every rule with the original Chinese dark chess. New version of Chinese dark chess is more like Chinese chess. In this paper, we propose this new game as intermediate version for discovering possible connection between Chinese chess-like games.

1.3 Related Works on Chinese Dark Chess

Chen et al. [2] used an alpha-beta algorithm with different display strategies in conjunction with the initial depth flip method to reduce the branching factor. They separate the opening game, the midfield game and the end game to apply different policies. Chen et al. [11] establish a database of endgames with reverse analysis. The created database is used for each first moving color, displaying up to 5 clips. They use 2 TB of memory to represent 1012 positions. The position status is stored as a win, lose or draw. Yan et al. [9] combined the chance node and Monte Carlo tree search (MCTS), then an uncertain Monte Carlo tree search model is proposed. They demonstrated a shorter simulation by adjusting three strategies called "Capture First", "Capture Stronger Piece First" and "Capture and Escape Stronger Piece First". As the decimation rate decreases, the winning rate increases and the simulation makes more sense for the MCTS.

2 Game Refinement Theory

The dynamics of decision options in the decision space has been investigated, which is a key factor in gauging game entertainment. Then, a measure of the refinement in games was proposed in 2003 year. The outcome of interesting games is always uncertain until the very end of the game. Thus, the variation in available options stays nearly constant throughout the game. In contrast to this, one player quickly dominates over the other in uninteresting games. Here options are likely to be diminishing quickly from the decision space. Therefore, the refined games are more likely to be seesaw games.

We review the early work of game refinement theory. The decision space is the minimal search space without forecasting. It provides the common measures for almost all board games. The dynamics of decision options in the decision space has been investigated and it is observed that this dynamics is a key factor for game entertainment. Thus a measure of the refinement in games was proposed.

A measure of game refinement (GR) theory is employed for assessing the degree of attractiveness of the game, which is derived from the game progress model [4]. The "game progress" is twofold. First is the game speed or scoring rate, while the other is the game information progress that emphasizes on the game outcome. Game information progress presents the degree of certainty of a game's result in time or steps. Having full information of the game progress, i.e. after its conclusion, the game progress $x(t)$ will be given as a linear function of time t with $0 \leq t \leq t_k$ and $0 \leq x(t) \leq x(t_k)$. It is assumed in the current model that the game information progress in any games is happening in our minds. We do not know yet about the physics in our minds, but it is likely that the acceleration of information progress is related to the force in mind. Hence, it is reasonably expected that the larger the value $\frac{x(t_k)}{(t_k)^2}$ is, the more the game becomes exciting due to the personal challenge faced by the players in achieving the game outcome. Thus, we apply its root square $\frac{\sqrt{x(t_k)}}{t_k}$, as a game refinement measure (denoted as GR). Generally, in board game we assume the Branching factor "B" as the $x(t_k)$ and the Depth of game "D" as the t_k. Then, Table 2 shows the measures of game refinement for three mind sports: chess, shogi and Go. It is conjectured that GR value of sophisticated mind sports varies between 0.07 and 0.08.

Table 2. Measures of game refinement for various types of games

Game	$x(t_k)$	t_k	R
Chess	35	80	0.074
Chinese chess	38	95	0.065
Shogi	80	115	0.078
Go	250	208	0.076

3 Methodology

3.1 Design of Dark Chess Engine

In past tournaments of Chinese dark chess, most of programs used the expectiminimax tree [11]. Normal minimax tree can not simulate flipping part because it is stochastic. The expectiminimax tree combines flipping node with movement nodes. Hence, we get the method which can search both parts in same game tree. We also implement alpha-beta pruning which efficiently reduce the simulating time.

In expectiminimax tree search, a heuristic value will be given to flipping nodes. This value combines probability and weight values of pieces. In this dark chess engine, we only search one level flipping nodes. Because the weakness of the expectminimax tree search is that taking too much time on searching flipping [11]. Bacause heuristic function can not differs flipping orders which has similar location, Therefore, for flipping part, one-level searching only makes decision whether flip or not. The action is still decided by flipping function.

To improve the strength of AI, a strong flipping strategy is necessary in Chinese dark chess. Firstly, we designed a relatively weaker version of flipping strategy, which purely depends on simple board evaluation. Then, another version is designed for final version AI. In our system, flipping strategy follows one principle that AI always choose the safest way. Under this principle, there are two phases of flipping are given. In the phase 1, AI will flip the piece which can not be captured in next move to ensure absolute safety. In the phase 2, AI will evaluate all positions and give them a value. This value follows the following formula.

$$Risk = C_1 \sum_{i=1}^{n} P_i + C_2 \ (1 - \sum_{i=1}^{n} P_i) \tag{1}$$

In this formula, C_1 and C_2 means fixed constant. In our experiment, C_1 is 1 and C_2 is 1. P_i means the probability that covered piece could capture the target piece. A function is designed to calculate the probabilities.

Recent studies have shown that MCTS is an efficient algorithm for games of no chance [11]. And MCTS can be implemented to any game without specific knowledge. Therefore, we implemented MisirlouV3 which tied for second place in preliminary and the final fourth in the UEC-GAT tournament. Normally, MCTS highly depends on the number of simulation to improve the quality of searching. Hence, we set a thresh-hold value of winning ratio in this AI. In this paper, any move being chosen will be checked by function. If this choice is disqualified, the program will activate the sub-AI automatically. The sub-AI designed based on simple principle, which AI will find the most valuable target and move closer to it.

3.2 Resign System

Different with Chinese chess, Chinese dark chess has its own chain of capturing. On specific circumstances, Chinese dark chess can be ended before all pieces being eliminated. For example, when one player has no method to capture opponent's top rank piece, game ends. Therefore, the time after the game outcome has been determined is considered as meaningless time. Hence, we can design a system for judging these situations. We conduct 1000 self-play experiments for same level AIs to collect data. By using this data, a model based on decision tree has been established. According to this classification model, we can design a reasonable judge system for Chinese dark chess experiment to get real game length. The accuracy for this model is 80.3%.

Table 3. Statistics of 1000 self-play experiments

Type	0	1	2	3	4	5
King (Win)	170	305				
King (Lose)	423	102				
Pawn (Win)	19	100	124	118	83	31
Pawn (Lose)	72	174	114	99	52	14
Cannon (Win)	204	175	96			
Cannon (Lose)	362	139	24			

According to Table 3 and its decision tree model, first two rules for resign system is set. Other rules are set from 3–6:

1 If your number of king is 1, your number of cannon can not be 0.
2 If your number of king is 0, your number of pawn should be at least 2.
3 If your number of king is 0, your top rank should be at least same level as opponent's top rank.
4 If your number of cannon is 0, number of your top rank piece is 1, which same with your opponent. This game ends with draw. (If top rank is king, number of pawn should be 0 too)
5 If your number of king and cannon are 0, opponent's number of top rank piece is larger.
6 If your number of king is 1, your number of remaining piece is 1. Opponent's number of remaining pieces is larger than 1 and it includes king or pawn.

3.3 Choice of Search Algorithm

For constructing strong AI, we use expectminimax algorithm with alpha-beta pruning. Normal min-max tree takes too much time to simulate one turn and can not search the flipping part. Expectiminimax with alpha-beta pruning not only efficiently reduce the simulating time but also combine the strategic part and flipping part. After implementation, simulation time for one turn is approximately 6 s [3].

Alpha-Beta pruning is one the most effective pruning method for min-max tree. Alpha-Beta pruning set two bounds for min-max tree,which are alpha bound and beta bound. Alpha bound and beta bound are both stored in every node. Beta bound will be set as $+\infty$ and alpha bound is $-\infty$. Different with normal min-max tree, alpha-beta pruning always check its bounds before it obtain the value of node. As pseudo code 1 shows, for all max nodes, their alpha bounds will be updated after selection. On the contrary, for those min nodes, their beta bounds will be updated. If beta is smaller than alpha or equal to alpha, rest of branched will be excluded from the tree. Because it meaningless to search them already [5].

Algorithm 1. Alpha-beta method implemented in Chinese Dark Chess AI

```
function ALPHA-BETA(node)
  if (node is a terminal node)
    return the heuristic value from evaluation
  if (node is a max node)
    temp := −∞
    for each child in node
      temp := max(Alpha-beta(child))
      α := max(α, temp)
      if β ≤ α
      break
    return temp
else
    temp := +∞
    for each child in node
      temp := min(Alpha-beta(child))
      β := min(β, temp)
      if β ≤ α
      break
    return temp
end function
```

4 Discussion

In order to promote data accuracy, we design three AIs who has tree different search depth to simulate the game process. First AI is the weakest one, which has 2 levels of search depth. Relatively, second 2 AI has 4 levels of search depth and third AI has 6 levels of search depth. Then, These three AIs fight against each other with 9 types of tournaments. Battles between two levels has been conducted for 50 times. After all the experiments, we know the average branching factor and game depth as Table 4 and 5 shows.

Table 4. Average branching factor of Chinese dark chess

B	1st		
2nd	Level 1	Level 2	Level 3
Level 1	20.2446	19.956	20.0259
Level 2	21.2534	20.4253	19.8365
Level 3	19.4695	20.0463	19.3593

Table 5. Average game length of Chinese dark chess

D	1st		
2nd	Level 1	Level 2	Level 3
Level 1	63.3	69.96	73.08
Level 2	57.4	63.56	73.2
Level 3	73.56	74.64	78.84

Table 6. The winning ratio of each level

W	1st		
2nd	Level 1	Level 2	Level 3
Level 1	48%/50%/2%	36%/60%/4%	44%/56%/0%
Level 2	64%/24%/12%	44%/56%/0%	48%/52%/0%
Level 3	60%/40%/0%	52%/48%/0%	52%/44%/4%

According to data given above, we notice the game refinement value is in zone value. This result suits for Chinese dark chess's success in the market. Comparing the data of perfect information dark chess with imperfect information dark chess, the game length get much longer after implementation of flipping. With implementation of flipping, Chinese dark chess becomes more competitive than perfect information Chinese dark chess. The game refinement value of perfect information Chinese dark chess is much higher than most of popular games, which means players can enjoy its impact. But without competition, fun of perfect information Chinese dark chess fades fast. Sophisticated players get longer game depth in experiments. Because sophisticated player could cause seesaw effect in the end of game. According to Table 6, Higher rank AI obviously get advantage in tournaments. Compare with the perfect information AI, the strong AI's advantage aren't obvious. The reason could be high quality flipping policy make up the strength rift between two version. After experiments of Chinese dark chess, we could see another result which comes from perfect information version of Chinese dark chess.

From Table 6 and 9, we notice the weaker player also can defeat stronger player with considerable possibility. On the other hand, Chinese Dark Chess is a typical imperfect information game. We conducted another set of 100 self-play experiments for Chinese dark chess to calculate more accurate game refinement value as Table 10 shows. Data of Chinese chess is also given in Table 2.

Table 7. Average branching factor of perfect information dark chess

B	1st		
2nd	Level 1	Level 2	Level 3
Level 1	16.7774	17.3665	17.377
Level 2	16.8505	17.5443	17.106
Level 3	17.123	17.0273	17.2313

Table 8. Average game length of perfect information dark chess

D	1st		
2nd	Level 1	Level 2	Level 3
Level 1	24.98	17.52	23.32
Level 2	20.92	20.52	22.32
Level 3	29.76	23.76	22.94

Table 9. The winning ratio of each level of perfect information dark chess

W	1st		
2nd	Level 1	Level 2	Level 3
Level 1	54%/46%/0%	40%/56%/4%	56%/44%/0%
Level 2	52%/48%/0%	50%/48%/2%	32%/64%/4%
Level 3	64%/36%/0%	84%/16%/0%	47%/51%/2%

Table 10. Self-play experiments

	B	D	WR	R
Chinese dark chess	20.6675	61.32	46%/49%/5%	0.074

From the table above, we notice that game refinement value of Chinese dark chess is a little lower than the zone value. This means Chinese dark chess is more competitive game than we supposed.

In order to get more sophisticated game. Almost all sophisticated games' refinement value gather in zone 0.07 ˜ 0.08. Chinese chess, and Chinese dark chess are all popular Chinese board games, which have game refinement zone value. In consideration of perfect information Chinese dark chess is a variant version we proposed. Huge difference with zone is reasonable. On one hand, flipping part not only improve the fairness in the game but also extend the game length. On the other hand, random initial place mechanism makes game length even shorter. Therefore we can see two different possible modification from Chinese chess. Chance based mechanism takes charge of the game, which makes their game refinement value higher in perfect information Chinese dark chess. From the self-play experiments, we can see that Chinese dark chess is a fair game, which has almost fifty-fifty percentage of winning.

5 Conclusion

In this paper, we designed the experiment platform for Chinese dark chess. According to the data we collected, a new possible evolution process for three Chinese board games has been proposed. In the AI tournament, we found the Chinese dark chess is a quite fair game no matter the sequence of players. Some effective change which applied in Chinese dark chess could be taken into consideration. Currently, because Chinese dark chess's chance based mechanism, professional tournament isn't suitable.

References

1. Abraham, R., Marsden, J.E., Marsden, J.E.: Foundations of Mechanics. Benjamin/Cummings Publishing Company Reading, Massachusetts (1978)
2. Chen, B.-N., Shen, B.-J., Hsu, T.: Chinese dark chess. ICGA J. **33**(2), 93–106 (2010)
3. Ding, C.H.Q., He, X., Zha, H., Gu, M., Simon, H.D.: A min-max cut algorithm for graph partitioning and data clustering. In: Proceedings 2001 IEEE International Conference on Data Mining, pp. 107–114. IEEE (2001)
4. Iida, H., Takahara, K., Nagashima, J., Kajihara, Y., Hashimoto, T.: An application of game-refinement theory to Mah Jong. In: Rauterberg, M. (ed.) ICEC 2004. LNCS, vol. 3166, pp. 333–338. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-28643-1_41
5. Knuth, D.E., Moore, R.W.: An analysis of alpha-beta pruning. Artif. Intell. **6**(4), 293–326 (1975)
6. Kreps, D.M.: Game Theory and Economic Modelling. Oxford University Press, Oxford (1990)
7. Rivest, R.L.: Game tree searching by min/max approximation. Artif. Intell. **34**(1), 77–96 (1987)
8. Roth, A.E.: The early history of experimental economics. J. Hist. Econ. Thought **15**(02), 184–209 (1993)

9. Yen, S.J., Chou, C.W., Chen, J.-C., Wu, I.-C., Kao, K.Y.: The art of the Chinese dark chess program DIABLE. In: Chang, R.S., Jain, L., Peng, S.L. (eds.) Advances in Intelligent Systems and Applications - Volume 1. SIST, vol. 20, pp. 231–242. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-35452-6_25
10. Yen, S.-J., Chou, C.-W., Chen, J.-C., Wu, I.-C., Kao, K.-Y.: Design and implementation of Chinese dark chess programs. IEEE Trans. Comput. Intell. AI Games **7**(1), 66–74 (2014)
11. Yen, S.-J., Chou, C.-W., Chen, J.-C., Wu, I.-C., Kao, K.-Y.: Design and implementation of Chinese dark chess programs. IEEE Trans. Comput. Intell. AI Games **7**(1), 66–74 (2015)