Congklak, a traditional game solution approach with breadth first search

Robbi Rahim1,*, Nuning Kurniasih2, Abdurrozzaq Hasibuan3, Liesna Andriany4, Asep Najmurokhman5, S Supriyanto6, W Wardayani7, Rahmat Hidayat7, D S W Lubis7, Darmawan Napitupulu8, Anggara Trisna Nugraha9, Henry Kristian Siburian10, and Dahlan Abdullah11

1Universiti Malaysia Perlis, School of Computer and Communication Engineering, Perlis, Malaysia
2Universitas Padjadjaran, Faculty of Communication Science, Library and Information Science Program, Bandung, Indonesia
3Universitas Islam Sumatera Utara, Department of Industry Engineering, Medan, Indonesia
4Universitas Islam Sumatera Utara, Department of Language and Literature, Medan, Indonesia
5Universitas Jenderal Achmad Yani, Department of Electrical Engineering, Cimahi, Indonesia
6Politeknik LP3I Medan, Department of Business Administration, Medan, Indonesia
7Sekolah Tinggi Ilmu Manajemen Sukma, Department of Management, Medan, Indonesia
8Indonesian Institute of Sciences, Research Center for Quality System and Testing Technology, Jakarta, Indonesia
9Universitas Muhammadiyah Sidoarjo, Department of Electrical Engineering, Sidoarjo, Indonesia
10Sekolah Tinggi Manajemen Informatika dan Komputer, Department of Informatics, Medan, Indonesia
11Universitas Malikussaleh, Department of Informatics, Lhokseumawe, Indonesia

Abstract. Congklak is a favorite populist game played at least 2 (two) players or in this article were human user and computer (AI). An essential point of playing congklak is to collect as many congklak seeds as possible to win the game, to win form from congklak it need a technique and it’s different for each players. Breadth First Search algorithm is a search algorithm which process to visit each node and the neighbors node to generate optimal graph that gives the best solution to computer (AI) and in this case were to complete the congklak game. Breadth First Search algorithm could be uses an alternative solution for optimal solution to win congklak with the help of mathematical computation.

1 Introduction

Congklak is a traditional game and it’s known in Indonesia, congklak has a different name in every region but the way it is played the same [1,2]. Congklak played at least by 2 (two) players, when playing congklak there is no optimal movement if played by the user (human) because the user will choose a lot of seeds and it’s different when applied specific algorithm to computer (AI) it will choose the fastest way to finish the game.

Many AI methods or artificial intelligence can be applied to solve the problem for choosing the best congklak seeds for game finish quickly and best movement, one of which is by using the tracking tree that is found in search algorithms such as Boyer-Moore [3], Knuth-Morris-Pratt [4], Hashing Search [5,6], Depth First Search (DFS) [7], Raita [8] and Breadth First Search (BFS) [9]. The search is performed by determining the initial state and Goal State, after determining initial state and goal state then the artificial intelligence that applied in the algorithm allows to find the optimal solution from the game.

The BFS algorithm [10,11] describes the problem-solving process which must take every best decision at each step by visiting each node of the tree that is likely to be the optimal solution for completion of congklak games.

2 Methodology

The searching procedure with BFS algorithm is a searching perform by visiting each node systematically at each level until the goal state is found [9,10,12,13]. Alternatively, in other words, a search is done by visiting nodes per level until a goal state is detected. If there is a solution, BFS search ensures the discovery of the solution with the shortest path. The process of BFS algorithm on the congklak game can see in the following example.

A. Initial State I:
   a) Granary-1: 7
   b) Granary-2: 7
   c) Granary-3: 7

B. Initial State II:
   a) Granary-1: 5, 2 congklak seeds already exist in granary 2 and granary 3
   b) Granary-2: 8
   c) Granary-3: 8

C. Initial State III:
   a) Granary-1: 5, 2 congklak seeds already exist in granaries 2 and granary 3

* Corresponding author: usurobbi85@zoho.com

© The Authors, published by EDP Sciences. This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (http://creativecommons.org/licenses/by/4.0/).
b) Granary-2: 7. 1 congklak seeds already in the granary 3

c) Granary-3: 9

D. Initial State IV:

a) Granary-1: 6, 2 congklak seeds are already in granaries 2, and the granaries 3 and 1 of the granary 3 are in this section

b) Granary-2: 8, 1 seed granary 3 is in this section

c) Granary-3: 7

Game congklak will be an order X x Y so that users can determine their own desired order. In the order X x Y, 2 boxes will be used by the starting point and point of destination, the rest to produce a path to look the path that will ultimately determine the shortest path to the destination point.

It has been explained in the analysis of the problem that the order can be adjusted to the range X x Y, one example of calculation in this congklak game will be explained by a minimum order of 3x3 because the calculation on any order will be the same [9,13,14]. In the determination of the weight of each node, it will give the value under the closest distance to the destination, for example, the node is farthest away from the destination then given a small weight while the closest node with the goal given higher weight. Example calculation as in figure 1 which given the smallest weight value is 1 and the weight is plus 1.

![Fig. 1. Map space conditions to be calculated with BFS.](image)

The above process is a step that must be done to complete the game congklak, so the process will do until all the congklak seed into the big barn, figure 2 is the visualization of congklak with the accomplishment of BFS algorithm that is created, this game was created using Visual C# and this game implemented multimedia [15,16] system to make game visualization more exciting and easy to use.

![Fig. 2. Initial State Congklak.](image)

After the initial state of the congklak where n = 7 and p = 7, the following values are taken from any value position to be transferred to the congklak granary, the value in the small granary is taken randomly, figure 3 until 8 are the few of result of congklak seeds movement.

![Fig. 3. Congklak Seeds first movement.](image)

![Fig. 4. Congklak Seed Movement.](image)

![Fig. 5. Congklak Seed 4 Movement.](image)

![Fig. 6. Congklak Seed 5 Movement.](image)

![Fig. 7. Congklak Seed 12 Movement.](image)

![Fig. 8. Goal State.](image)

Figure 3 until figure 8 is an example of movement congklak. 17 movements are required to complete the congklak game by applying the AI BFS algorithm to the computer. Flowchart application of BFS algorithm on congklak seed movement can be seen in Figure 9 below.
Figure 9 is a flowchart process of congklak seed movement until all congklak seeds run out and the player won, in this case, is a computer that implements the concept of artificial intelligence with breadth-first search algorithm.

3 Results and Discussion

Implementation of BFS algorithm on congklak games designed using Visual C# programming language can be seen in some of the following test results:

Figure 11 shown that the computer turn will pick up the best seeds congklak to move from each granary to other granary and take many seeds to win the game.

Figure 12 shows the number of scores on the computer is 49 and the computer controls the dominant game, Figure 13 shows an active yellow block indicating the game on the user side (human), after logically selecting a movement that allows winning the game results obtained as shown 14.
Score user (human) has reached 25 and computer 50 with playing conditions on the computer side, and this game will continue until each one of players empties the seeds in his side.

4 Conclusion

Implementation of BFS algorithm as an algorithm for artificial intelligence in congklak can help to finishing the game more faster using computer AI, but in experiment not all game won by computer AI, this is because human intelligence are taking various scenario better than computer. The processing of intelligence artificial requires high computation and need addition of other algorithms such as heuristics and machine learning algorithms for better implementation.

References

1. G. Hermawan, “Implementasi Algoritma Greedy Best First Search pada Aplikasi Permainan Congklak untuk Optimasi Pemilihan Lubang dengan Pola Berfikir Dinamis,” in Seminar Nasional Teknologi Informasi dan Multimedia (SNASTIA), pp. 1–6. (2012)
2. M. F. Kasim, “Playing the game of Congklak with reinforcement learning,” in Proceedings of 2016 8th International Conference on Information Technology and Electrical Engineering: Empowering Technology for Better Future, ICITEE 2016, (2017)
3. R. Rahim, A. S. Ahmar, A. P. Ardyanti, and D. Nofriansyah, “Visual Approach of Searching Process using Boyer-Moore Algorithm,” J. Phys. Conf. Ser., vol. 930, no. 1, p. 012001, (Dec. 2017)
4. R. Rahim, I. Zulkarnain, and H. Jaya, “A review: search visualization with Knuth Morris Pratt algorithm,” in IOP Conference Series: Materials Science and Engineering, vol. 237, no. 1, p. 012026. (2017)
5. R. Rahim, I. Zulkarnain, and H. Jaya, “Double hashing technique in closed hashing search process,” IOP Conf. Ser. Mater. Sci. Eng., vol. 237, no. 1, p. 012027, Sep. (2017)
6. R. Rahim, Nurjamiyah, and A. R. Dewi, “Data Collision Prevention with Overflow Hashing Technique in Closed Hash Searching Process,” J. Phys. Conf. Ser., vol. 930, no. 1, p. 012012, (Dec. 2017)
7. R. Rahim et al., “Block Architecture Problem with Depth First Search Solution and Its Application,” J. Phys. Conf. Ser., vol. 954, no. 1, p. 012006, (2018)
8. R. Rahim et al., “Searching Process with Raita Algorithm and its Application,” J. Phys. Conf. Ser., vol. 1007, no. 1, p. 012004, Apr. (2018)
9. R. Ratnadewi, E. M. Sartika, R. Rahim, B. Anwar, M. Syahril, and H. Winata, “Crossing Rivers Problem Solution with Breadth-First Search Approach,” in IOP Conference Series: Materials Science and Engineering, vol. 288, no. 1. (2018)
10. Y. Zhou, W. Wang, D. He, and Z. Wang, “A fewest-turn-and-shortest path algorithm based on breadth-first search,” Geo-Spatial Inf. Sci., vol. 17, no. 4, pp. 201–207, (2014)
11. S. Beamer, K. Asanovic, and D. Patterson, “Searching for a Parent Instead of Fighting Over Children: A Fast Breadth-First Search Implementation for Graph500,” Tech. Rep. UCB/EECS-2011-117, EECS Dep. Univ. California, Berkeley, pp. 1–9, (2011)
12. R. Zhou and E. A. Hansen, “Breadth-first heuristic search,” Artif. Intell., vol. 170, no. 4–5, pp. 385–408, (2006)
13. E. Zunic, A. Djedovic, and B. Zunic, “Software solution for optimal planning of sales persons work based on Depth-First Search and Breadth-First Search algorithms,” in 2016 39th International Convention on Information and Communication Technology, Electronics and Microelectronics, MIPRO 2016 - Proceedings, pp. 1248–1253, (2016)
14. U. A. Acar, A. Charguéraud, and M. Rainey, “A Work-efficient Algorithm for Parallel Unordered Depth-first Search,” Proc. Int. Conf. High Perform. Comput. Networking, Storage Anal., p. 67:1–67:12, (2015)
15. Sriadhi, “Model of the Material Inventory Management Using Multimedia based Information System,” in IOP Conference Series: Materials Science and Engineering, vol. 180, no. 1. (2017)
16. R. K. Roslan and A. Ahmad, “3D Spatial Visualisation Skills Training Application for School Students Using Hologram Pyramid,” JOIV Int. J. Informatics Vis., vol. 1, no. 4, p. 170, (Nov. 2017)