Solving Fanorona: Human vs. AI

Catia Teixeira, *Student, FEUP,* Rojan Aslani, *Student, FEUP*

***Abstract*—The abstract goes here.**

***Index Terms*—IEEE, IEEEtran, journal, LaTeX, paper, template.**

I. Introduction

FANORONA is a two-player strategic board game that originated in Madagascar. Its rules are quite similar to those of the game of checkers, but it has its own unique features.

A. Specification of the work

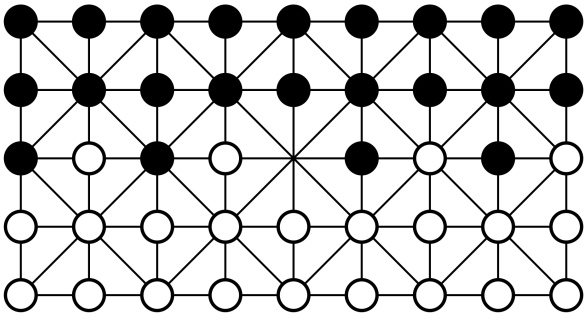
specification of the work performed (definition of the game or optimization problem to be solved)

Fanorona is played with black and white pieces placed on a rectangular board. The size of the board varies depending on the variant of the game being played. The three standard variants are Fanoron-Telo (3 by 3), Fanoron-Dimy (5 by 5), and Fanoron-Tsivy (5 by 9). An example of the Fanaron-Tsivy board is presented in Fig. 1 [1].

Players alternate turns, starting with White. Each player has 22 pieces of their color on the board. The pieces are placed where the gridlines cross, leaving the central point empty. A line represents the path along which a stone can move during the game. There are crossings that are strong and weak. In weak intersections, only horizontal moves are possible, while in a strong intersection, it is also possible to move a stone diagonally [1], [2].

There are two kinds of moves: non-capturing (*paika* move) and capturing. A *paika* move consists of moving one stone along a line to an adjacent intersection without removing any of the opponent's stones. Capturing moves, which imply the removal of one or more pieces of the opponent, are necessary for a win and have to be played in preference to *paika* moves. There are two ways to do a capturing move:

Fig. 1. 5 by 9 Fanoron-Tsivy board with the initial position of the stones [2].



1) **Approach:** moving the capturing stone to a point adjacent to an opponent's stone, which must be on the continuation of the capturing stone's movement line.
2) **Withdrawal:** the capturing stone moves from a point adjacent to the opponent's stone, away from the stone along the continuation of the line between them.

When an opponent stone is captured, all opponent pieces in line beyond that stone (as long as there is no interruption by an empty point or an own stone) are captured as well. Approach and withdrawal captures cannot be made at the same time – the player must choose one or the other.

The capturing piece is allowed to continue making successive captures, with the following restrictions:

- The piece is not allowed to arrive at the same position twice.
- It is not permitted to move twice consecutively in the same direction (first to make a withdrawal capture, and then to make an approach capture) as part of a capturing sequence.

The winner is chosen when they take all the opponent's pieces or leave the opponent in a position where they can't move. If neither player manages to win, then the game is a draw [1].

B. Complexity analysis

It is important to quantify the game's complexity as it allows us to know how difficult the a is to solve, and lets us choose the more adequate techniques.

C. Related work

related work with references to works found in a bibliographic search (articles, web pages and/or source code)

II. Methodology

SOFTWARE

A. Formulation of the problem

formulation of the problem as a search problem (state representation, initial state, objective test, operators (names, preconditions, effects and costs), heuristics/evaluation function) or optimization problem (solution representation, neighborhood/mutation and crossover functions, rigid constraints, evaluation functions),

B. Implementation details

implementation details (programming language, development environment, data structures, file structure, among others),

C. Approach

the approach (heuristics, evaluation functions, operators, ...)

D. Implemented algorithms

algorithms implemented (search algorithms, minimax, meta-heuristics)

UCB

MCTS

OUR DRAW

III. Results and Discussion

experimental results, using appropriate tables/plots and comparing the various methods, heuristics, algorithms and respective parameterizations for different scenarios/problems.

IV. Conclusion

The conclusion goes here.

For future works, it would be interesting to allow the players to make consecutive moves in one turn. This option was not implemented in the current algorithm. It would also be interesting to explore other draw situations, especially for the AI players.

Regarding the algorithm, in future works, it can definitely be optimized to decrease space complexity.

Another interesting topic to explore further would be successor generation ordering in the MINIMAX algorithms. This can speed-up the algorithm and reduce cases of draw, against other AI agents.

It would be important to future investigate the hyperparameter-tuning of the most relevant variables in UCB for MCTS. For MCTS the incorporation of draw condition in the rollout phase can possibly drastically decrease the computation time and improve its overall performance.

Appendix A
Proof of the First Zonklar Equation

Appendix one text goes here.

Appendix B

Appendix two text goes here.

Acknowledgment

The authors would like to thank...

References

[1] D. Walker, *A Book of Historic Board Games*. Lulu.com, 2014.

[2] M. Schadd, M. Winands, J. Uiterwijk, H. Herik, and M. BERGSMA, "Best play in fanorona leads to draw," *New Mathematics and Natural Computation*, vol. 04, pp. 369–387, 11 2008.

John Doe Biography text here.