

Karl Isele  
CSE 5052 - David Carlyn  
March 28, 2022

### Royal Game of Ur - AI Project Proposal

I plan on programming a rational agent to play the Royal Game of Ur. The Game of Ur is a two-player racing strategy game that dates back to circa 2,500 B.C.E. [1]. Wikipedia describes a basic set of rules for the Game of Ur [2]. The performance measure is the agent's progress towards guiding all pieces off of the board before the opponent. The environment is the game board, Figure 1 below, and the game rules. The rules add random chance to the environment. Therefore, the Game of Ur is both adversarial and probabilistic. Dice determine how many spaces an agent can move a piece forward, but the agent can decide what piece to move forward. The agent can perceive the positions of pieces on the board and how many pieces each player still has in their inventory. A player wins when they have no pieces on the board or in their inventory.

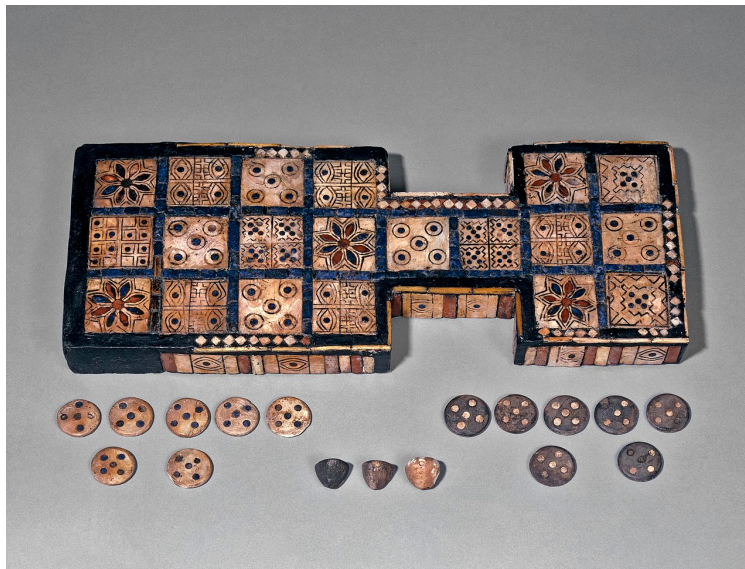


Figure 1: Royal Game of Ur board [3]

I plan on implementing an expectiminimax search algorithm. Each turn in the Game of Ur can involve up to 7 actions to choose from, resulting in  $7^n$  terminal nodes where  $n$  is the number of turns. There may be tens or even hundreds of turns in each approximately 30 minute Game of Ur, so computing the final utilities is unreasonable. Therefore, I will use an evaluation function for each state. An agent could evaluate actions according to maximizing immediate distance traveled, but an agent could also consider sabotaging the opponent and placing pieces in secure places, that is, the rosette in the combat zone.

## References

- [1] <https://www.youtube.com/watch?v=WZskjLq040I>
- [2] [https://en.wikipedia.org/wiki/Royal\\_Game\\_of\\_Ur#Basic\\_rules](https://en.wikipedia.org/wiki/Royal_Game_of_Ur#Basic_rules)
- [3] [https://www.britishmuseum.org/collection/object/W\\_1928-1009-378](https://www.britishmuseum.org/collection/object/W_1928-1009-378)

## Additional info

<https://www.tradgames.org.uk/games/Royal-Game-Ur.htm>