QWIXX: GAME THEORY OPTIMIZATION

Introduction

Qwixx is a single or multiplayer dice game in which the objective of each player is to maximize the amount of points scored and/or to score more points than one's opponents. Players cross out numbers in four different colored rows from left to right based on rolls from six different dice. The game ends when either two rows become locked (at least five spaces are crossed off including the furthest right space) or a player commits four penalties (the player is unable to make a move on four separate turns). Since the game involves a set number of states and decisions at each turn, an optimal strategy exists to guide players to make decisions that will maximize their score and increase their chances of winning. The goal of this project is to identify this optimal strategy through both simulation and mathematical based solutions, and to produce an adjustable framework for answering similar problems.

Previous Work

Up to this point, little formal work has been done on Qwixx. Reddit user Bmhowe34 has applied Markov decision processes and dynamic programming to maximize the expected score (115.48 points) during a single-player game. One setback to this analysis is that it is incredibly computationally taxing, taking several hours to compute solutions. Using absorbing Markov chains, they also computed a "race to lock" strategy which uses the expected number of rolls left to lock a row in order to end the game as quickly as possible. Although this strategy had a significantly lower expected score (74.98 points) than the optimal strategy in single-player games, when competing in two-player games, the "race to lock" strategy won more than two-thirds of the time, indicating that maximizing one's score does not necessarily result in wins.

Methodology

Two complementary approaches will be taken in order to find an optimal strategy. (1) The first is simulation based, outputting scores and results from game simulations based on several different coded strategies. This allows us to visualize long run trends through distributions and identify which decisions should be prioritized to produce higher scores and more wins. In multiplayer games, these simulations provide us with an easy way to see how different strategies compete against one another. (2) The second is mathematical based, using Markov decision processes and dynamic programming to calculate the move with the highest probability for success based on the current state of the game. This is the same approach taken by Bmhowe34 to find an optimal single-player strategy.

References

https://drive.google.com/file/d/0B0E4VFlFjnCuME9sZGhrbGRIWXc/view?resourcekey=0-xBGSNx3qX1pF1tjD7yZJWA