Matthew Dunn & Viictoria Kim
Dominating Strategies Code Description
June 28, 2019

The goal of our work was to reduce the computation time of the Colonel Blotto Game Matlab simulation. Our approach to the problem was to reduce the number of strategies that each agent will use during the simulation. This was done in an effort to speed up the code and allow for simulations to be quicker and simulations with more complicated scenarios to run in a reasonable amount of time.

To reduce the computation time of the Matlab code, the dominating strategies code reduces the number of strategies each agent will utilize before computation begins. This is accomplished by analyzing the strategies of each player. If one strategy will offer a player a greater or equal payoff in all cases than another, than that strategy is considered to be dominant and we can remove the non-dominant strategy from our simulation.

For a game with two attackers, there are four possible game results. There is the possibility that both attackers win, one or the other wins, or none win. Each player has a different preference for these four possible outcomes. These player preferences are quantified and stored in a matrix where the greatest value is associated with that player's most favorable strategy. This is what allows us to determine which strategies are dominant by choosing strategies with the highest preference value. For each player, adjacent strategies are compared. The strategy which results in the highest preference value is kept while the other strategy is discarded.

The code we have written decreases the size of the matrix input to the code, and therefore the time it takes for the code to run. Future work includes making this code more general, so different numbers of agents may be considered. Also it is possible that theory may support reducing strategies in some other way, so this may also be considered.