# Deckbuilding AI Using Monte Carlo Tree Search

### Overview

The artifact will be a competitive AI for the deckbuilding game of Dominion. The AI algorithm will use Monte Carlo Tree Search (MCTS) as the base with chance nodes to simulate the hidden information such as what cards you draw and what cards your opponent has drawn. To prove the strength of the AI, there will be a human vs AI mode as well as a AI vs AI mode. The AI vs AI mode will have options for the MCTS AI and simple state machine Ais that have been used by other theses.

## Features

|  |  |  |
| --- | --- | --- |
| Area | Feature | Description |
| Game Mechanics | Dominion Base Game | 2 Player Base Dominion game is built for easy control for both a human and an AI player. It is built very efficiently to allow for massive amounts of games on multiple threads |
| Game Mechanics | Dominion Base Rules | Dominion has base rules for any game excluding the cards themselves. This includes Drawing, choosing the 10 cards to add, and buy, action, and discard phases |
| Game Mechanics | Dominion Cards | Only cards in both first and second edition are implemented. This removes many of the complicated cards and still leaves 18 possible cards out of 10 that will be in a single game |
| UI/UX | Card Dragging | Player can drag cards from hand to play |
| UI/UX | Hand | Player's current hand is shown at the bottom of the screen |
| UI/UX | Basic Game Information | During the player's turn the player's current gold amount, amount of buys available, and amount of actions available. |
| UI/UX | Grey out invalid cards | Cards that can't be played are greyed out or shown as invalid plays |
| UI/UX | Go to Next Phase/End Turn | There is a button to move to the next phase and end turn |
| UI/UX | Opponents turns move conveyance | Opponents plays are highlighted to show what they did |
| AI | Big Money State Machine | A state machine using the Big money algorithm where only the highest money value is bought until the average money density reaches enough to buy a province every turn |
| AI | Single Witch State Machine | A state machine using the Single Witch algorithm where Big Money is used except for getting a single witch as soon as possible |
| AI | MCTS | MCTS is implemented with exposed variables for tweaking the algorithm like selection method and Exploration parameter |
| AI | MCTS Move Score | There is a toggleable UI element to show the Ais current move score for each buy |
| Analysis | Game Simulation | Dominion games can go into simulation mode where the games go as fast as possible and aren't renderered on screen. |
| Main Menu | Buttons | Game has buttons for Play game, Quit game, and setup simulation |
| Analysis | Simulation controls | Simulation controls for numer of simulations, which AIs play against each other, variations for MCTS, how long to let MCTS think, etc. |
| Analysis | Results Output Log | The results of the simulations are outputted both on screen and out to a text file |
| Play Game Menu | Start Game | There is a start game button on the play game menu |
| Play Game Menu | Random cards or chosen cards | You can choose whether the card piles are random or specified from a given list |
| Results | MCTS vs Single Witch and Big Money analysis | The results between MCTS and Single Witch/Big Money are written up to show how well (or poorly) MCTS did with the different variations |

## Milestones

### Milestone 1: 9/09/2020

* Build Towers of Hanoi
* Build MCTS AI for Towers of Hanoi
* Build Tic Tac Toe
* Build MCTS AI for Tic Tac Toe

### Milestone 2: 9/23/2020

* Prototype Dominion with limited controls and rules (use number keys to choose cards and enter to confirm choice)
* Add in a limited set of cards (only money, victory point, and only simple action cards)
* Build Monte Carlo (no tree) AI for Dominion to play a single action from a known game state

### Milestone 3: 10/07/2020

* Implement first iteration of chance nodes in Monte Carlo Tree Search
* Test different variations of handling hidden information in MCTS

### Milestone 4: 10/21/2020

* Continue iteration on hidden information
* Build simple state machines for Big Money and Single Witch to compare against
* Begin adding in UI/UX features like click and dragging cards to play and clickable buttons

### Milestone 5: 11/04/2020

* Continue iteration on MCTS
* Implement UI/UX features like greying out invalid cards and opponent action conveyance
* Create simple simulation mode to pit 2 Ais against each other and spit out results

### Milestone 6: 11/18/2020

* Continue iteration on MCTS
* Polish Simulation controls page
* Polish Menu screens to have features likes choosing which cards to add to game and/or random

### Milestone 7: 12/01/2020 – Final Deliverable

* Finish up iterating on MCTS
* Polish final elements like showing AI score for a move