CPSC 490 Project Proposal: Computational Intelligence for

*Settlers of Catan*

Christopher Kim and Omar Ashraf
Advisor: James Glenn

**Introduction and Game Mechanics**

This work seeks to address two main goals: 1) develop a mobile application for the board game, *Settlers of Catan* (henceforth referenced to as *Catan*), that allows human players to play against other human players and artificial intelligence agents, and 2) apply computational intelligence concepts to design and implement an artificial intelligence agent to competitively play *Catan*.

*Catan* is a stochastic board game played between 2-4 players on a board of hexagonal tiles. Each tile is assigned a particular resource (brick, wood, sheep, wheat, or ore) as well as a numeric value between 2 and 12 (inclusive). The configuration of these tiles is randomly generated and subject to several constraints to ensure fairness. The goal of the game is to accumulate 10 “victory points” which can be earned through the following ways:

1. **Settlements** - Settlements are placed on the corners of hexagonal tiles and is the primary means of acquiring resources. On each player’s turn, two dice are rolled and if a player has a settlement adjacent to a tile that contains the number that was rolled, he or she receives a resource card that matches the resource of the tile. Players place two settlements in the initial stage of the game and can build up to 5 settlements by trading in specific resource cards. Each settlement counts as one victory point.

2. **Cities** - Cities are upgraded settlements and can be built by trading in specific resource cards. If a player has a city adjacent to a tile that contains the number that was rolled, he or she receives two resource cards of the corresponding resource. Each city counts as two victory points.

3. **Longest Road** - Settlements and cities are connected to one another by roads. If a player builds 5 connecting roads, he or she attains *Longest Road* which corresponds to two victory points. Unlike settlements and cities, the owner of *Longest Road* can change throughout the game as players build additional roads.

4. **Largest Army** - Resource cards can also be used to buy development cards. There are 3 main types of development cards: ability cards, which allow players to use one of three abilities; knight cards; and victory point cards. Knight cards allow players to steal resources from other players. When a player activates 3 knight cards, he or she attains *Largest Army* which
corresponds to two victory points. Similar to *Longest Road*, the owner of *Largest Army* can change throughout the game as players activate additional knight cards.

5. **Victory Point Card** - The final category of development cards is the victory point card. When a player draws a victory point card, he or she is awarded a single victory point.

**Project Details**

This project consists of two main parts. Part I will involve the development of an Android-based mobile application that will allow human players to play against one another and against AI agents. This will be divided into two main components: the front-end Graphical User Interface and the game framework (decoupled from the application that will allow us to both run automated simulations to test AI performance and manually play the game). Part II will involve the design and implementation of an AI agent to competitively play *Catan*. Our current plan is to utilize a hierarchical deep reinforcement learning approach that combines deep Q learning with *a priori* heuristic-based knowledge to develop an agent that can rival a human’s performance. By hierarchical, we distinguish between lower-level strategies (such as build-a-settlement, build-a-city, build-a-road, or buy-a-development-card) and higher-level strategies (such as roads-and-settlements, cities-and-development, or port-opening). We will use heuristics to map from lower-level strategies to higher-level strategies and employ deep reinforcement learning to learn lower-level strategies. We build on Pfeiffer’s work, *Reinforcement Learning of Strategies for Settlers of Catan* (Pfeiffer, 2003), which uses a similar hierarchical approach but uses model trees instead of Q learning for learning state-action values.

**Division of Work and Deliverables**

Initially, I will be working with my partner Omar Ashraf on the implementation of the game framework, keeping in mind object-oriented design principles that will allow us to easily switch between human players and AI agents. This will allow us to use our framework for both the gameplay within the application as well as the training and testing of the AI agent. Once the game framework has been completed, I will create an intuitive Graphical User Interface that users can seamlessly interact with while Omar conducts a literature review of similar games and computational intelligence techniques that can be extended to our agent.

By the time the front-end component has been completed, we anticipate that a rough model for the AI agent based on previous work for *Catan* and similar games will have been completed. I will focus on generating heuristics to bridge the gap between the higher-level strategies and the lower-level strategies. This will involve running multiple simulations that will investigate various factors that might improve upon the AI agent’s performance. These factors could include initial settlement placement, initial
placement order, and optimal resource card spending. The higher-level strategies that we will be testing include:

1. **Roads and Settlements** - The focus of this game strategy is rapid building and expanding as quickly as possible. Thus, it is important to secure healthy amounts of every resource with less focus on ore and build roads to hinder other players’ development. Attaining *Longest Road* is crucial to this strategy.

2. **Cities and Development** - The focus of this game strategy is upgrading settlements to cities as quickly as possible and acquiring development cards. Therefore, the most important resources to focus on is ore and wheat. Attaining *Largest Army* is crucial to this strategy.

3. **Port Opening** - The focus of this game strategy is to start with a port that will allow you to monopolize a single resource. When a settlement is built adjacent to a port, the owning player can trade with the bank at a two-to-one ratio.

Once the architecture of the AI agent has been decided, I will work simultaneously with Omar in the training and testing phase. This will be one of the most significant aspects of the project as we fine-tune the parameters of the architecture to yield the best performance. This phase might also require us to revisit the architecture of the agent and revise the architecture as necessary.

**Stretch Goals**

If the deliverables defined above have been completed before the deadline, we want to address two major stretch goals. The first is to utilize some other computational intelligence technique to compare and/or improve upon the performance of the AI agent. An example of such a technique is Monte-Carlo Tree Search which has been applied to similar stochastic games. The second stretch goal is to build a trading component into the gameplay of the mobile application. We have purposely omitted trading for the sake of simplification, as the inclusion of trading would significantly increase the size of the action space, making it difficult to train our model.

**Timeline**

- **09/20/2018** - Submission of Project Proposal
- **09/21/2018 - 10/13/2018 (~3 wks)** - Implementation of Game Framework
- **10/14/2018 - 10/27/2018 (~2 wks)** - Implementation of Graphical User Interface
- **11/11/2018 - 12/04/2018 (~3.5 wks)** - Training, Testing, and Refinement of Model