John Rowley

**S**trategy **G**ame **P**laying **I**ntelligent **A**gent (SGPIA)

**Introduction and Project Success**

SGPIA is an intelligent agent that plays, learns, and devises strategies for strategy games. Over time SGPIA will develop new strategies and adjust old strategies. SGPIA will refine its strategy over many iterations of playing. SGPIA will come with 2 custom built strategy board games for it to play on. SGPIA will be developed to be compatible with strategic board games like Risk, Monopoly, and Settlers of Catan.

**Research**

All AIs are some kind of intelligent agent, and there are many different models that the intelligent agent can use. Most of the AIs developed to play strategy games are simple reflex agents, they act only on the current state. Deep Blue was a simple reflex agent, it would find the value of its current state and make a move that improved its value and decreased the opponents. Problem with these AIs is that they ignore previous history, cannot change values to be more efficient, cannot see patterns, and they cannot adjust their strategy. Another problem with simple reflex agents is that as the game gets more complex, the amount of time needed to make a move goes up exponentially.

To create a better AI a different model is needed. A learning agent is the answer, it can review previous history, can modify values and how it works, adapt in real time, discover patterns, and can make predictions. The learning agent can do these things because of its structure, it first critiques its move, which is sent as feedback to the learning element, which changes the performance element. After the performance element is changed, it will perform the action on the environment.

The learning element needs a type of machine learning. The learning agent could use a decision tree, but with the complexity of strategy board games, the tree would grow too large. SGPIA needs to learn and evolve over time, the best type of machine learning algorithm for this task is a genetic algorithm.

**Benefits**

The benefits of using a learning agent over a simple reflex agent is: It can learn from past events, modify its performance element, adapt in real time, discover patterns, and make predications. With all these benefits SGPIA will out-perform a simple reflex agent after some training. Even if the simple reflex agent’s algorithm is changed to improve it, SGPIA will adapt and defeat the simple reflex agent.

**Goals**

1. Have SGPIA win against a human player
2. Have SGPIA adjust its strategy to deal with unknown strategies
3. Have SGPIA’s improved strategies win faster than the older strategy against the same player and strategy

**Target Users**

This project is designed to learn to play the game and play against opponents. SGPIA will include 2 board games to play on.

There are multiple types of people who will be using SGPIA. One type is a person who wants to play against a very hard AI. Since a learning agent improves over time and out-performs a reflex agent, the player will have a much harder time defeating it. SGPIA is also more difficult than a reflex agent is because it will adapt its strategies. Reflex agents follow a linear path, and once a player has found a way to defeat it, all of the challenge has been lost. Since SGPIA adapts its strategies, a player can’t defeat it with a single strategy.

Another type of person who would use SGPIA is a person who wants to play against an AI who is new to the game like they are and will improve at about the same rate they do. SGPIA starts out with no knowledge of the game and must learn like the player. Each game they play together improves the skill of SGPIA, meaning that the skill of the AI and player should be relatively close each game.

Creators of games can also use this AI for their game. They can train SGPIA to a certain level and cap its learning, so the AI stays at a certain difficulty level. By capping the level the creator can easily get the desired difficulty level, compared to coding the AIs difficult level which often is either too easy or too hard.

**Project Backlog**

Gamer stories

1. I want to play against an AI that develops its skills along with me, so that the AI’s skill level stays near mine. Like a person who is new to the game.
2. I want to play against an AI that will adapt to what I do instead of following a rigid strategies. I want the AI to play a game that flows instead of following a straight rigid path.
3. I developed a new strategy and I want to test it out, but I don’t want to try it against human players. I want to face an AI that slowly adapts and allows me to refine the strategy.
4. I want to play against a really difficult AI, the AIs I play against are hard at first but once I find out their strategy, I can easily beat the AI and it’s not challenging anymore.

Developer Stories

1. I tried to create different difficulty AIs, but they are either too easy or too hard.
2. The AIs I have been creating is not difficult enough, players can easily defeat it.
3. I want an AI that can adapt to the players actions, so it doesn’t feel like a computer player, but have it feel like a human player.

**2-Week Plan**

The beginning two weeks will be focus on creating and integrating the AI’s framework and the games. The AI’s framework will be designed so that it can easily connect to the games and accept moves generated from the AI. It will also be designed that its output should be compatible with most strategy board games. After the framework has been set up, the games will be created so that the AI can play them. Both the framework and games will be coded in C# with Visual studio 2012.

**Week 1**

For the first week I will create the framework that will hold the AI and allow it to interact with the game. After the framework is completed I will create a dummy game that has some functionality to test how the framework can interact with it.

The test to ensure that the framework is created correctly is to perform functions from both Risk and Monopoly (such as moving and drawing a card) from the framework. If the actions are able to effect the game and works correctly, the framework was built correctly.

**Week 2**

Week two will be the creation one of the strategy board games. The game I will be creating is Monopoly or Monopoly like game, it will first be created with simple functionality and rules, after testing the functionality, the functionality will be increased and tested with the framework until the game has full functionality and is compatible with the framework.

I will demonstrate the functionality of the game by playing a single round, and I will also make some actions from the AI framework that will change the state of the game correctly.