

# **Project Proposal**

Strategically Exploring the Possibilities for  
Humanly Competitive and Intelligent Agent  
Players for Settlers of Catan

Johannes Algelind (algelind@student.ch...)

Sven Andersson (andsve@student.ch...)

Sicheng Chen (sicheng@student.ch...)

Andreas Rönqvist (ronnqvist@student.ch...)

April 13, 2010

# Contents

## Summary

We want to create an agent, that can play the board game Settlers of Catan, focusing on making it specifically good at the starting/setup stage and its usage of settlements and roads. It should be able to find good spots to build at that gives it the most important resources. We aren't going to focus on including all possible features of the board game, but focus on only the core features in our agent (i.e. Development Cards and trading will not be utilized to keep the simplicity), but if we have enough time more advanced features will be implemented.

## Key ideas

We are planning on dividing the game play in two parts. The first part is the actual game setup, where the players need to place their initial settlements and roads. The second part is the actual game playing, where the dice is rolled and settlements are bought and cards are dealt.

As there are so few actions the player can do each turn, the main idea is to have a long going plan. The AI's task will be to insert appropriate actions in to the planning queue. An appropriate plan might be to understand that it needs to get more ore and then develop a plan to get more ore. This could be solved by building a settlement at an ore site, if found possible.

Actions that we are planning to support are mainly to build roads, settlements and cities in a strategically good way. The AI should scan its close-by area, say all spots maximum 3 roads from any of its settlements or roads and rank each spot based on how good it would be to build a settlement there. Resources that the AI is missing gets ranked high, harbors are considered and this is weighted versus upgrading an existing settlement to a city.

Actions that might be inside the scope of the project are those of trading and buying and using development cards. These actions are not necessary to win the game, but provides an advantage if used. If we cannot find time to implement these actions, we should estimate how much better the AI would be if it could use them.

Outside the scope of the project, but still interesting, is that of playing aggressively towards the opponents. It's possible to block the opponents or build at a spot just to prevent them from building there. This kind of strategic play is hard even for human players and thus not in the scope of this project.

By using the existing server of JSettlers we already have the basic game logic and message passing. Our client receive all information about the game and we can keep track of the board layout in our local datastructure. To make an action, the client only has to send simple messages to the server. We believe our approach of dividing the game play in two parts is a good solution as the initiation does not return later and doesn't have to be considered when playing in the second part. Also, by sticking to the basic foundations of the game, the AI can quite easily value its options and aim for its longterm goal by ranking the steps that work towards it higher than other steps.

If we succeed, we will have created an AI that will be competitive to the robots that comes with JSettlers. If we manage to implement trading and developement cards the AI might even be competitive to humans.

## Survey

aoe

## Evaluation

aoe

## First results with small programs

By using the JSettlers server, we have a working game logic. To use this, we have developed a client in Python that communicates with the server. The client connects to the server, creates a new game and requests the server to

start it. The server adds robots to the empty spots and starts the game. The client then receives the board layout and all other messages that the server passes. The client stores the game elements in its own local datastructure that it will use to decide upon its next move.

Making this program has allowed us to confirm that our approach will work and that a Python client is compatible with the Java server.

## Project management

aoe

## Website

Our website is updated with news and relevant information for our project, which also has some introduction information about the project itself.

- <http://sweetfish.github.com/TIN171/>

We also have a central repository where we store our source code for the agent and toy applications.

- <http://github.com/sweetfish/TIN171>