Building a usable interface and a computer opponent for a new game

Samer Sabri
Computer Science Senior Project (CPSC 490)
Advisor: Professor James Aspnes

February 7, 2013
1 Motivation

I have acquired a solid background in programming through “data structures and programming techniques” and “systems programming and computer organization” as well as a general understanding game theory through “game theory and political science.” In addition, “design and analysis of algorithms” introduced me to the theoretical apparatus that would allow me to put these different interests in practice.

As a passionate chess player, I started looking into computer chess, but a discussion with Professor Eisenstat and a quick glance at the computer chess literature convinced me that there was very little that could be done in one semester that would be an innovative contribution.

After consulting with Professor Aspnes, I decided to find a game that would be much less studied than chess and explore different possible algorithmic techniques (including machine learning and randomized algorithms) that would allow me to eventually come up with a respectable computer player for the game. I also decided to make the interface as user-friendly as possible and maybe enable users to play online against each other. This will allow me to achieve several goals:

- Learn to choose the most efficient representation for moves and positions
- Deepen my knowledge of algorithms by exploring algorithms related to game and applying them to a particular problem
- Continue to improve my programming abilities
- Learn about design and UI principles and understand how to integrate different interfaces and different languages in a coherent end product (e.g., web page coded in Javascript running a computer player written in C)

I created a simple tile-based game called “Angry Artists” (see description in section 2). The game appears to be novel, which makes looking into algorithmic techniques to build a new player much more interesting. However, more time needs to be spent on choosing a good representation for the game and building a platform for the game to be played on.
2 Describing “Angry Artists”

Angry artists is a two-player perfect information tile-based game. One possible story that could motivate the game is that each player is an artist collaborating with the other on a contemporary work of art. Each turn, one of the artists places a polygonal tile on a polygonal canvas with the constraint that the color of any side of the tile being placed has to agree with the color of the side of the tile adjacent to it. Because they are angry, the artists are trying to place as many of their tiles on the canvas and they win this passive-aggressive game if they are able to place more tiles on the canvas than their partner/opponent.

2.1 A simple version to start

I will start with a simple version of the game where the canvas is a triangle composed of 9 smaller triangles. The pieces are tiles of the size of the small triangles, and the pieces’ sides are labeled with a + or a -. Pieces are randomly assigned such that player 1 receives 5 pieces and player 2 receives 4 pieces. See figure 1 for a representation of the situation before any moves are played.

Player one starts by placing a piece, then the players alternate. A piece can only be placed in an unoccupied square, meaning that no piece has been placed on it yet (I mark occupied squares with a star in figures 2 and 3).

If and only if a player cannot place a piece, she is allowed to skip her turn. The game ends when both players are unable to place any of their pieces. The winner is the player who has placed the most pieces. The only exception to that rule is that player 2 wins if the board is "filled" and the players have no pieces left.

See figures 1, 2, and 3 for the first two moves of a game of Angry Artists.
2.2 Variations and possibilities

There are several ways to expand on the simple game shown and vary its complexity. I present the main variations here:

- The triangle figure can be generalized to an n-sided polygon
- Pieces can be composed of \( k < n \) sides of the polygon
- The rule about not being able to place a piece in an occupied square can be dropped
- Instead of allocating victory based on the largest number of pieces placed, the rule can be that the first player to be unable to place a piece loses

2.3 Issues to be resolved

In the process of creating an interface to play the game and a computer player, several questions have to be resolved. I present some of these questions as well as some preliminary answers.

- Internal representation: I need to find the easiest and most efficient way to represent the board and the pieces. The solution for now is a representation of the game as an edge-labeling problem in a graph. Each player has a set of adjacent labels (each piece represents at most three possibilities in the triangle game for example) that she can choose from at each move

- Determining the labeling on pieces: It is important for the game to be fair and interesting. For that to be the case, the initial labels on the pieces need to be arranged in a way that would allow for at least most pieces to be placed in case one wants to “fill the board.” The difficulty is in finding a good principle for that assignment to be done algorithmically

- Searching the tree: After a certain level of complexity, “solving” the game become impractical. Different methods need to be thought of to search the move tree; these methods are dependent on different properties of this tree I will need to discover through study as well as experimentation
User Interface: From a user’s perspective, the way the pieces and the board are represented graphically matters in how well a player is able to play the game and how much she enjoys it. That will need to be determined by iteration and through collecting feedback from helpful volunteers.

Figure 1: Board and piece allocation before any moves are played.
3 Plan and deliverables

3.1 Plan

The first phase of the project will be dedicated to learning Javascript and writing a webpage that will allow two human players to play against each other. This will push me to work on internal representation, piece allocation, as well as scaling to different variants.

After that phase, I will explore different possible algorithms for a computer player, implement some of them, and play these algorithms against humans and against each other to decide which algorithm works best for
Finally, if there is time left in the semester, I will work on a web interface that will allow players to play each other or a computer online.

### 3.2 Deliverables

In addition to the abstract and reports expected from the Computer Science Department, the three deliverables that are to be expected are:

- A webpage for human players to play against each other
- Two or more computer algorithms for Angry Artists
• An analysis of the strengths and weaknesses of different algorithms

If time allows it, a fourth deliverable can be expected, namely a web interface for players to play each other or a computer online.