Project 2-1

Play it!

An investigation into strategy games

Academic Year 2020-2021

Courses

Databases (KEN2110)
Linear Programming (KEN2520)
Probability and Statistics (KEN2130)
Machine Learning (KEN2240)
Graph Theory (KEN2220)
Reasoning Techniques (KEN2230)

Play it!

PROJECT 2-1

The central topic of this project consists of strategy games. The aim is to develop algorithms, which are able to play the game as good as possible.

1. Project description

The central topic of this project consists of strategy games. You are required to implement a human-human player version and a human-computer player version using object-oriented programming. For the artificial player you will need to take into account that you have to implement a game tree data structure and a smart evaluation function, which you can apply to your game tree. Additionally you can use any heuristics you might find applicable for the game at hand. The three phases of this project can be described as follows:

Block 1:

The following lists provide the games from which you as a project group should choose one. Note that we have categorized them according to three (difficulty) criteria: 1) two-player or multi(more than 2)-player; 2) perfect or imperfect information; 3) with or without chance.

The "easiest" games are perfect-information, two-player games without chance:

- Abalone
- Clobber
- Cram
- Dots-and-Boxes
- Dvonn
- Gipf

The more difficult games (with one difficulty criterion) are:

- Gin-Rummy (imperfect information)
- Backgammon (chance)
- Chinese Checkers (multi-player)
- Azul, (multi-player)
- Terra Mystica, (multi-player)
- Power Grid, (multi-player)
- Blokus, (multi-player)
- La Strada (multi-player)

Finally two "hard" games (with two or even three difficulty criteria)

- Ticket To Ride (multi-player, with imperfect information, but no chance)
- Risk (multi-player, with imperfect information, and chance)

It is clear that the more easy games are easier to implement, but of course also less challenging. It is strongly recommended that you clearly discuss this within your group and estimate your skills when choosing one of the games.

For the chosen game, make an analysis and model of the data structures and algorithms needed (possibly in a UML-like style) to implement your game. Implement – in Java – the game chosen such that two or more human players can play it "on the computer" and make sure you have a good visualization of the game.

Start to research different algorithms to design a computer player capable of playing against a human. Argue why, from a mathematical viewpoint, your evaluation function is good. To obtain a good evaluation function it is useful to do a literature study of available A.I. techniques. Decide which ones you will use for your game.

Hint: Have a look into different techniques from the field of Machine Learning.

It is furthermore possible to choose a game that is not on the provided list. In that case, you have to ask the examiners for permission.

Block 2:

Define and implement your game tree data structure and evaluation function. Implement and compare different strategies of play based on the A.I. techniques you examined. Furthermore, perform a complexity analysis for your algorithm.

Block 3:

Continue implementing strategies (and comparing them).

Project examiners: Jos Uiterwijk, Matus Mihalak