

Project Manual Bachelor Year 2
Project 2.1
Play it!

2022-2023

Period 1.1, 1.2, and 1.3
Academic year 2022-2023
Data Science and Knowledge Engineering
Faculty of Science and Engineering
Maastricht University

Courses:

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

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 data structure and a smart evaluation function, which you can apply to your game. Additionally, your artificial player should be adaptive and present some form of learning. The three phases of this project can be described as follows:

1.1 Phase 1

The following lists provide the games from which you as a project group should choose one. If you have a favorite game and you want to implement this, you need to ask the examiners for permission. For each game, you should refer to the given link for the standard rules that you have to implement. Note that we have categorized them according to three (difficulty) criteria: 1) two-player or multi(more than two)-player; 2) perfect or imperfect information; 3) with or without chance.

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

- Tak ([https://en.wikipedia.org/wiki/Tak_\(game\)](https://en.wikipedia.org/wiki/Tak_(game)))
- Connections (https://boardgames.lovetoknow.com/Connections_Board_Game)
- Yinsh (<http://www.gipf.com/yinsh/rules/rules.html>)
- Breakthru (<https://boardgamegeek.com/boardgame/335/breakthru>)
- Cage (http://web.archive.org/web/20200701105455/http://www.marksteeregames.com/Cage_rules.html)
- TwixT (<https://en.wikipedia.org/wiki/TwixT>)

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

- Dice Chess (<https://brainking.com/en/GameRules?tp=95>) – chance
- Sneakthrough (<https://ludii.games/details.php?keyword=Sneakthrough>) – imperfect information
- 3-player Hex (<https://ludii.games/details.php?keyword=Three-Player%20Hex>) – multi-player
- 2-player Briscola (<https://www.pagat.com/acetan/briscola.html>) – imperfect information (you can extend it to multi-player)
- Imperial (<https://www.fgbradleys.com/rules/rules4/Imperial%20-%20Rules.pdf>) – multi-player
- Omega (https://nestorgames.com/rulebooks/OMEGA_EN.pdf) – multi-player

Finally, one “hard” games (with two difficulty criteria)

- Carcassonne (https://images.zmangames.com/filer_public/d5/20/d5208d61-8583-478b-a06d-b49fc9cd7aaa/zm7810_carcassonne_rules.pdf) – multi-player, chance

It is clear that the easier 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.

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.

Research different algorithms to design a computer player capable of playing against a human. Do a literature study of available AI techniques, decide which ones you will use for your game, and argue why they are suitable for it. You should look into different techniques from the fields of Machine Learning and Adversarial Search and choose at least one technique for each of the two fields. Please note that - unlike in previous projects where the examiners were satisfied with a vague idea of what you will do next - here, we want you to put enough effort into researching about the approaches already in phase 1.

In the presentation, you need to make sure that the examiners understand your approaches. UML diagrams may help you to make this clear. However, focus on the most interesting parts of your implementation rather than presenting a huge UML diagram nobody can read. Furthermore, walk the examiners through EVERY slide of your presentation (also through the future planning and the overview of who did what). Highlighting the most important parts will make everything readable.

Furthermore, by the end of this phase, you have to deliver a draft introduction for the final report. The draft introduction must include the definition of the problem statement and the research questions the group wants to address and the literature review. In the draft, it should also be clear which AI techniques you plan to implement for the computer player in the next phases. Please, check the report guidelines for more details on how to write the report. Peer feedback: by the end of phase 1, each group will have to anonymously evaluate the games implemented by two other groups and fill out a feedback form. Groups can consider this feedback to improve their programs in the subsequent phase. 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 (Backgammon is not allowed).

1.2 Phase 2

Implement at least one type of game-playing agent based on at least one of the AI techniques you examined and test it against a baseline agent. The simplest baseline agent you can use is one that chooses random actions for each game step, but you can also implement a more advanced baseline. Furthermore, perform a complexity analysis for your algorithm. Peer feedback: by the end of phase 2, each group will have to anonymously evaluate the games and computer players implemented by three other groups and fill out a feedback form. Groups can consider this feedback to improve their programs in the subsequent phase.

1.3 Phase 3

Improve the agent you developed in phase 2. If your agent was using only one among the Adversarial Search and Machine Learning techniques you selected in phase 1, make sure that in this phase you implement a hybrid agent that combines techniques from both areas. Finally, test and evaluate all the strategies you implemented.