02285 AI and MAS

Mandatory Assignment 3

Due: Monday, 2nd of June at 13.00

Mikkel Birkegaard Andersen

1 Introduction

This document describes the mandatory programming project of 02285. The project is designed to be very flexible, allowing many different group sizes and levels of ambition. Successful implementations of the project range all the way from basic solutions using standard techniques to highly research relevant multi-agent systems. Due to the flexibility of the project, group sizes can range from 3 to 6 students. The expectations and assessment of your project will obviously depend on the group size.

2 Goal of the project

The goal of the project is to implement an AI client that can compete in the Multi-Agent Programming Contest, 2013 edition. You will compete in the full game, except that there will be no achievements, meaning that you will not have to deal with upgrades. The simulation will otherwise be configured as the standard one given in `sim1.xml` of the default download.[1] You can use the clients provided in the default download and the released agents of last year's competition to test your client.

You are free to implement your AI client in whatever programming language you prefer, using any techniques that you like. There are no set requirements for your solution.

3 Schedule

There will be 4 hour project sessions every Tuesday from 13 to 17 for the rest of the semester (except on the seminar day). You can ask for help with both the project and the report.

Tuesday, 8th of April at 13.00. Deadline for the group registration. There will be a group assignment on CampusNet for this purpose. You should come up with a great team name, as this is how you will be identified at the competition. Just upload an empty file named after your team and make sure that all group members join the hand-in.

Tuesday, 29th of April, 13.00 - 17.00. Seminar day. Each group will prepare a 15-minute presentation + slides about their work so far, to be given to your fellow students and a teacher. You might have to be selective about what you present, while remembering to

[1] The one available at https://multiagentcontest.org/getting-started

keep the same level of abstraction as for the two mini-reports. After the presentation there will be roughly 15 minutes of discussion, critique and feedback.

Your performance at the seminar will be evaluated pass/fail. If you pass, it will not effect your final grade.

Tuesday, 27th of May at 13.00. Game day. You will be pitting your program against the other groups in a battle for points and glory! (But not grades – see below). We will find a number of suitable rooms, where we can watch the carnage unfold on a big screen. You will connect your client to the simulation server in the same way as for the "real" competition, so no need to worry about getting it to run on an unfamiliar machine.

The competition will be a knock out tournament. With 16 groups, this gives a Round of 16, Quarter-finals, Semi-finals and a final, with pairings for all stages (except for the final, of course) determined by random draw.

The goal of the competition is to motivate you to create a strong program, and to allow comparison of the strengths and weaknesses of the various approaches. Your result in the competition will not affect your grade. Also, a group of 6 people obviously has a lot more resources for making a good solution than a group of 3 does.

4 Report

In addition to implementing your AI client, you should write a short report prepared in the style of a conference article. The report should be at most 15 pages long. It should focus on your problem analysis, your solution, and an analysis of your findings and results. The reader of the report is expected to be familiar with the project description given in this document. The reader is also expected to be familiar with the entire curriculum of the course, so there is no need to repeat *any* of that in the report. If you expect the reader to be familiar with additional work in the area, please make the necessary references. Also make sure to reference any specific work that you might have used in completing the project (books, articles, implementations, online resources, etc.).

Your report should describe your ideas, solutions, and original contributions. You should describe your implemented system in terms of its overall algorithmic functionality, not in terms of implementation details. Your report should make clear what methods within AI and MAS you have been implementing, using the correct computer science nomenclature. Finally, make sure to reflect on strengths and weaknesses of your solution.

It is very important that your report is clear about who has done what in the project, and who has written which parts of the report. From http://shb.dtu.dk/default.aspx?documentid=2811&Language=en-GB&lg=&version=2013%2f2014:

> In order to grade a group project on its own the individual contributions of the students have to be marked in the report. *Students stating that they have contributed equally to the report without marking which sections they individually are responsible for does not fulfill the requirement of individualization.* The students individual sections have to be clearly marked in the common report.

Again: "Students stating that they have contributed equally to the report without marking which sections they individually are responsible for does not fulfill the requirement of individu-

alization". This means that *if you do not individualize your report, you will automatically fail!!!*

5 Handing In

The completed project should be submitted electronically via CampusNet using the assignment "Mandatory Assignment 3 Final Handin". The deadline for submission is:

Monday, 2nd of June at 13.00.

Each group makes a single submission consisting of the following:

a) A **pdf file** containing the report. Please put the following information on the front page of your report:

- Course number (02285), course name (AI and MAS), and date.
- Name of the group.
- Study numbers and full names of all group members.

b) Source code and executable code of your implemented software.

All these files should be archived in a single zip-file named `project02285.zip` to be uploaded to CampusNet (no `.rar`, `.tar`, `.gz` or other formats, please!).

6 Expected workload

As previously announced, the programming project constitutes approximately 3.75 of the 7.5 ECTS in the course. This means that the expected workload of the programming project is approximately 100 hours per student.

Good luck with your project. Be creative! Have fun!