MGAIA Assignment 2 General Video Game AI

Mike Preuss, Matthias Müller-Brockhausen, Ioannis Chios

March 2021

1 Introduction

The goal of this assignment is to experiment with the General Video Game AI Framework. This framework is used to create submissions for the GVG-AI Competition[1] which explores the problem of creating controllers for general video game playing. For this assignment you will have to work in groups of 2 students.

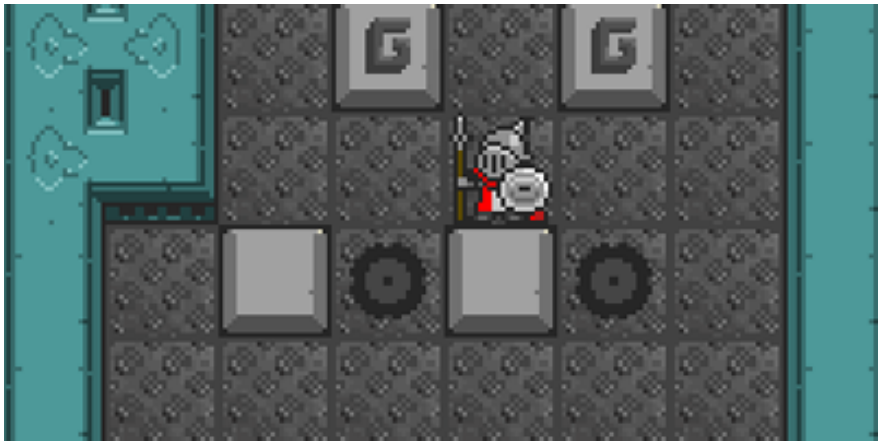

Figure 1: Sokoban, a 2-player cooperative game that is part of the GVG-AI competition

General video game playing is difficult because the agent has to be able to make decisions in real time, without knowing the rules of the game beforehand. The issue is complicated even further by introducing more players into the game, resulting in most of the games being non-deterministic, since the other player's or non-player character's actions are happening at the same time, and they have to be anticipated. All these elements present a significant challenge which makes developing effective general game playing agents a very interesting topic of research.

We want you to write agents for the single player planning track. To do this you need a Java Compiler on your machine (or use a DSLab machine which has one preinstalled) and clone the GVG-AI framework code[2]. In there, your entrypoint can be found under (gym_gvgai/envs/gvgai/src/tracks/singlePlayer/Test.java). Said test file contains a lot of comments that should help you get started, and there is also a whole book you can refer to in case you are stuck[3] If your Java knowledge is a little bit rusty this shell script will compile and run said file. Save it in the root folder of the cloned repository and run it.

[1] http://www.gvgai.net/
[2] https://github.com/rubenrtorrado/GVGAI_GYM
[3] https://gaigresearch.github.io/gvgaibook/

Listing 1: Shell Script to run code from the single player track

```
# Build the client
src_folder='./gym_gvgai/envs/gvgai/src'
build_folder='builtclient'
rm -rf ${build_folder}
mkdir -p ${build_folder}
find ${src_folder} -name "*.java" > sources.txt
javac -d ${build_folder} @sources.txt
# run built client
java -classpath ${build_folder} tracks.singlePlayer.Test
```

You will also have to adjust line 29 to point to a valid csv file. You are free to build your own game csv or use the one we provide here (to use it also save it as games.csv in the root folder and adjust line 29 to point to "games.csv"). Furthermore to prevent a blank screen and your agent from being super slow you have to replace the file in "gym_gvgai/envs/gvgai/src/core/competition/CompetitionParameters.java" with the contents found in the following link[4].

Listing 2: Example CSV-File to load different games your agent will have to play

```
0,gym_gvgai/envs/gvgai/examples/gridphysics/butterflies.txt
1,gym_gvgai/envs/gvgai/examples/gridphysics/chopper.txt
2,gym_gvgai/envs/gvgai/examples/gridphysics/deflection.txt
3,gym_gvgai/envs/gvgai/examples/gridphysics/frogs.txt
4,gym_gvgai/envs/gvgai/examples/gridphysics/intersection.txt
5,gym_gvgai/envs/gvgai/examples/gridphysics/jaws.txt
6,gym_gvgai/envs/gvgai/examples/gridphysics/chase.txt
7,gym_gvgai/envs/gvgai/examples/gridphysics/boulderdash.txt
8,gym_gvgai/envs/gvgai/examples/gridphysics/crossfire.txt
9,gym_gvgai/envs/gvgai/examples/gridphysics/bait.txt
```

Your main task will be to implement a general game playing agent. You are free to extend any of the sample agents found in line 23 to 26 (MCTS, RS, RHEA, OLET). Your agent should be able to play any game from the GVG-AI competition. To test and evaluate your implementation you can use the train set above or build your own. You can find a full list of single player games here[5].

2 Submission

Make sure to nicely document everything that you do and why you do it. Your final submission is a ZIP file containing the following:

- Java Source code for your agent that should run without further setup if we adjust line 66 to point to your agent class under "tracks.singlePlayer.advanced.student.Agent".
- A self-contained pdf report. This report contains an explanation of the extensions you applied to the sample agent, how they help in improving your agents performance and overall conclusions.

If you have any questions about this assignment, please visit our lab sessions on Thursdays where we can help you out. In case you cannot make it, you can post your questions in the Microsoft Teams Channel or on the Brightspace discussion forums, where other students can also read and reply to your questions.

The final deadline for this assignment is the 01.04.2021. We also expect you to hand in a preliminary version of your agent (without report) by 22.03.2021! Failure to do so will result in point subtraction! The performance of the agent is not important, but expect that we will run some agents in a competitive mode

[4] https://raw.githubusercontent.com/GAIGResearch/GVGAI/master/src/core/competition/CompetitionParameters.java

[5] https://raw.githubusercontent.com/GAIGResearch/GVGAI/master/examples/all_games_sp.csv

in the labs. Submission is only important to us in order to encourage you to start working on it early, as the main aim of this is not to re-implement something or get something to work but to test some creative changes, from parameter settings mixing different approaches to introducing some new ideas, based on the rich GVGAI interface. If you start only a few days before the final deadline you will definitely not have enough time to perform some reasonable experiments, which will result in a sub-optimal grade.