
Chapter 5 - Learning in GVGAI

Exercises

These exercises are also available at this book’s website: <https://gaigresearch.github.io/gvgaibook/>.

1 Download and Installation

All the code of GVGAI learning environment is available in a GitHub repository¹. Use the checkpoint² in order to run the same version presented in this chapter. All the code of GVGAI Gym is available in a GitHub repository³, as well as the instructions. Use the checkpoint⁴ in order to run the same version presented in this chapter. We recommend to use the GVGAI Gym implementation. A step-by-step tutorial video about how to download, install and run the code using Jupyter notebook can be found on YouTube⁵.

2 Play a game randomly

The exercise is to randomly play a level of a game, for example *Aliens*. We recommend doing this using Jupyter notebook. This enables the code to be easily modified and re-run. The sample code is given in Figure ?? . You can ran the sample code directly. The line below indicates the environment to use.

```
env = gym.make('gvgai-aliens-lvl0-v0')
```

If you would like to play some other games, please replace “aliens” with another one (i.e. “escape” or “bait”). A full list of available games can be found in the folder `gym_gvgai/envs/games`. For each of the games, there are normally five levels. You can indicate the level that you want to play by replacing “lvl0”. For instance, you can play the second level of game *Zelda* using `env = gym.make('gvgai-zelda-lvl1-v0')`.

¹ <https://github.com/GAIGResearch/GVGAI>

² <https://github.com/GAIGResearch/GVGAI/commit/a0e267416f27b5a0ac5b30644ed0354a6e4b4705>

³ https://github.com/rubenrtorradogvgai_gym

⁴ https://github.com/rubenrtorradogvgai_gym/commit/20cac6e9e783e3ef381eb87446a21f6bb21557e3

⁵ <https://www.youtube.com/watch?v=084KgRt6AJI&feature=youtu.be>

3 Train a more sophisticated agent

The exercise is to train a more sophisticated agent, e.g., A2C. Implementations of some classic reinforcement learning algorithms are available in the GitHub repository, *OpenAI Baselines*⁶. Otherwise, you can try *Stable Baselines*⁷, the improved implementations based on *OpenAI Baselines*.

4 Create your own agents

A recent survey [1] showcases a myriad of deep learning methods for games that can be used in GVGAI. The generality and flexibility of this benchmark allows for experimental setup that can go beyond the current state of the art. Examples are as follows.

- Train your method in multiple levels at the same time, to make sure the algorithm learns concepts beyond particular levels. Test them in other levels of the game.
- Modify the VGDL description of a given game to create slight variations of the same game. Could your algorithm learn faster (or better) by training in a set of games that are closer in the game space?
- Train your algorithm in different games and test them in some others. Maybe you can group them by genre first and study the learning capabilities of these methods within the same type of game.

References

1. N. Justesen, P. Bontrager, J. Togelius, and S. Risi, “Deep Learning for Video Game Playing,” *arXiv preprint arXiv:1708.07902*, 2017.

⁶ <https://github.com/openai/baselines/>

⁷ <https://github.com/hill-a/stable-baselines>