Agent for DigDug Game

Inteligencia Artificial 2023/2024

José Mendes – 107188
Filipe Obrist - 107471

universidade de aveiro

Introduction

This report details the workings of our developed agent for the Dig Dug game. To accomplish this, we created the “student.py” file based on the “client.py” file as requested, eliminating *pygame* dependencies as they were deemed unnecessary.

Game Objectives and Classes

The primary goal of the game is to accumulate points by eliminating as many enemies as possible in each level while evading their attacks to avoid being killed. To realize this objective, we devised four distinct Python files, each serving a specific purpose:

- **tree_search.py**: Based on the tree search used in the practical classes;
- **didug.py**: Search domain from tree search;
- **agent.py**: Logic to solve the game (attack and avoid attacks);
- **student.py**: Based on “client.py” file, calls the method *update_state* from “agent.py”, returning DigDug's next move.

Algorithm

The algorithm implemented in the "agent.py" file can be divided in three distinct segments:

- **No Enemies in DigDug's Tunnel**: When no enemies are present in DigDug's tunnel, the algorithm seeks out a new tunnel for DigDug to enter, while ensuring the avoidance of other tunnels (preventing enemy escape) along with rocks. This process uses a *tree_search* methodology, which will be elaborated upon later.
- **Enemies Present in DigDug's Tunnel**: In case enemies are detected within DigDug's tunnel, the agent's behavior involves targeting the closest enemy and taking action, including shooting if the enemy is within range, or going after him, aligning with the enemies column or row.
- **Handling Close Proximity to Enemies**: To prevent DigDug from being eliminated by an enemy, specific precautions are taken. When in close proximity to a Pooka, DigDug avoids moving in the direction the Pooka is headed to prevent interception. Similarly, when facing a Fygar, DigDug refrains from moving toward its intended path. Also, if the Fygar is walking horizontally, DigDug maintains a distance of at least three squares in front to evade its fire.

Whenever the *update_state* function is invoked from “student.py”, the algorithm evaluates DigDug's current tunnel status and the presence of enemies, along with enemies inside their own tunnels, prioritizing the the ones inside the current tunnel. This evaluation guides DigDug's decision-making process as described in the previously segments, ensuring strategic actions are taken accordingly to maximize DigDug's survival and score in the game.

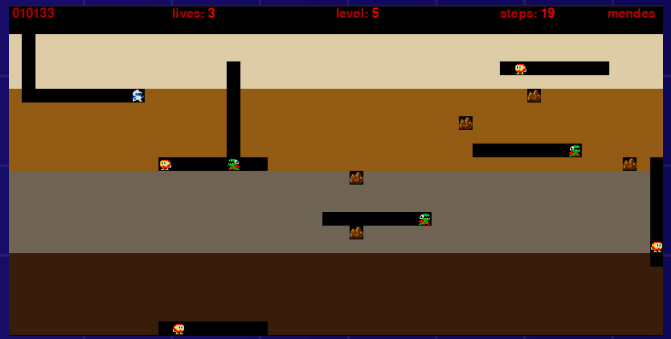
Tree Search

In order to facilitate DigDug's movement into unexplored tunnels without going through other tunnels and getting stuck in rocks, a simple tree search was used.

This process utilizes a tree search technique similar to what was taught in practical classes, integrated into the “digdug.py” file as a new Search Domain. It operates by analyzing the provided map, marking specific areas as off-limits where DigDug cannot go, such as squares around tunnels, rocks, and spaces directly under rocks. This helps prevent additional enemies from entering DigDug's tunnel unnecessarily and avoids potential entrapment by rocks. The objective will be the closest tunnel entry.

The heuristic applied in this search method calculates the sum of differences between DigDug's current position and the targeted goal position, both vertically and horizontally.

This tree search mechanism is being used with a greedy strategy in the function search inside the agent.py file. To put into perspective a function was developed to display how the map is being "viewed" by our Agent ("E" is the possible entries, "O" is the off-limits" and "R" is the rocks):



Changes since the first delivery

Between deliveries, a few ideas in the agent's movement logic were changed:

- Running away direction is now randomized between possible choices to prevent looping.
- The way DigDug goes after enemies has been changed. Previously, DigDug attempted to constantly position itself behind an enemy, which caused complications as enemy intelligence increased (level 10). To simplify things, now he just goes to the nearest space in the same column or row as the enemy.
- Finally, dodging enemies was also altered. DigDug now dodges Pooka's movements and Fygar's fire with greater eficiency.

Benchmark

Highest scores:

- 1st Delivery: 77748
- Final Delivery: 73291

Average number of steps in level 10 with 3 attempts:

- 1st Delivery: 2432
- Final Delivery: 1897