|  |  |
| --- | --- |
|  | **Wydział Finansów i Zarządzania** **Kierunek: Informatyka** |

Dawid Miśkiewicz

(numer albumu: 81794)

Temat pracy

The Comparison of Evolutionary Algorithms, Genetic Algorithms, and AI in Flappy Bird Development

Opiekun merytoryczny:

Mgr inż., Sebastian Bień

Wrocław 2024

Table of content

Introduction

1. Flappy Bird Game  
1.1 Overview of Flappy Bird Game Development

1.1.1 Tools and technologies used for development (pygame)

1.2 Designing the Game Architecture

1.2.1 Structuring the game logic (game loop, event handling, physics).

1.2.2 Creating sprites and animations (player, obstacles, background).

1.3 Implementing Game Physics

1.3.1 Simulating gravity and player movement.

1.3.2 Collision detection between the bird and pipes.

2. Evolutionary Algorithms

2.1 What it is

2.1.1 Definition and inspiration from natural selection.

2.1.2 Key components: population, fitness, selection, crossover, mutation.

2.2. How does it work

2.2.1 Initialization of the population.

2.2.2 Evaluation of individuals using a fitness function.

2.2.3 Selection of parents for reproduction.

2.2.4 Generating offspring through crossover and mutation.

2.3 Example of use

2.3.1 Solving combinatorial problems (e.g., traveling salesman problem).

2.3.2 Evolving game-playing agents.

2.4 Advantages and Limitations

3. Genetic Algorithms

3.1 What it is

3.1.1 A type of evolutionary algorithm based on genetics and natural selection.

3.1.2 Representation of solutions as chromosomes (bit strings, real numbers, etc.).

3.2. How does it work

3.2.1 Encoding solutions into a genetic format.

3.2.2 Fitness evaluation and ranking of individuals.

3.2.3 Genetic operators: selection (e.g., roulette wheel, tournament), crossover, mutation.

3.3 Example of use

3.3.1 Feature selection in machine learning.

3.3.2 Evolving strategies for board games.

3.3.3 Parameter optimization in engineering problems.

4. Comparison of Algorithms  
4.1 Evolutionary Algorithms in Flappy Bird  
4.2 Genetic Algorithms in Flappy Bird  
4.3 AI Approaches in Flappy Bird  
4.4 Strengths and Weaknesses of Each Approach

5. Results and Analysis  
5.1 Performance Comparisons  
5.2 Optimization Efficiency  
5.3 Game Performance Evaluation

CONCLUSIONS

BIBLIOGRAPHY

TABLE OF FIGURES

SUMMARY

Introduction

1. Flappy Bird Game

This chapter provides an overview of the development of the Flappy Bird game, highlighting the tools and technologies implemented, the design and implementation of the game architecture, and the core mechanics that govern its functionality.

1.1 Flappy Bird Game – basic information

Flappy Bird is a game developed in 2013 by a Vietnamese programmer Dong Nguyen under his company DOTGEARS COMPANY LIMITED. The game was released in May 2013 on the IOS platform and in January 2014 on the Android platform. At the start of 2014, the game gained an unexpected spike in popularity, leading to mixed reviews caused of its difficulty. The objective of the game is to pass through pipes with a **sprite** using only one button to jump. The player’s score is determined by the amount of pipes passed.

1.2 Overview of Flappy Bird Game Development

1.2.1 Tools and technologies used for development

The app was developed in the programming language Python using the game engine Pygame in an integrated development environment Visual Studio Code. The following technologies were used to achieve the goals set in the project:

* Python 3.11.9
* Pygame 2.4.0
* neat-python 0.92

1.3 Designing the Game Architecture

1.3.1 Structuring the game logic

The main game loop is managed by main() and manual\_play() functions. Inside the loop occur two steps. Event Handling is where the loop processes events such as window events for exiting the game and key presses for jumping the bird.

Obraz zawierający tekst, zrzut ekranu, Czcionka, numer

Opis wygenerowany automatycznie

The next step is physics that are handled by the Bird class in the move() method, where gravity and jump mechanics are applied to the bird. The bird’s position is updated for each frame based on velocity that is adjusted with gravity. Pipes move with constant speed and their height is random, as defined in the Pipe class.

Obraz zawierający tekst, Czcionka, zrzut ekranu, linia

Opis wygenerowany automatycznie

1.3.2 Creating sprites and animations

Player (Bird):

The bird is represented using three images (in flight) that have been loaded using pygame.image.load() and scaled to fit inside the window. Animation is controlled in the draw() method inside the Bird class, which shows different bird images according to a counter.

Obraz zawierający tekst, zrzut ekranu, Czcionka

Opis wygenerowany automatycznie

The bird is then drawn with a rotation to simulate tilting during movement. The rotation is handled by pygame.transform.rotate().

Obstacles (Pipes):

Pipes are represented by an image (PIPE\_IMG) and have a top and bottom section. The Pipe class creates, moves, and draws these pipes. Pipes are portrayed through the draw() method of the Pipe class

Obraz zawierający tekst, Czcionka, zrzut ekranu, numer

Opis wygenerowany automatycznie

Pipes move across the screen in the move() method by decreasing their x position, and new pipes are periodically created.

2. Evolutionary Algorithm

The part concerned with operational principles and design principles of the application developed for this particular project will be presented in this chapter. It aims to go into the methodology and implementation issues, focusing on the key components with interrelations and, further still, discussing practical examples of use cases.

2.1 What is it?

An evolutionary algorithm is a computer-based approach to problem-solving modeled on natural evolution through the application of computer science and artificial intelligence. It simulates certain biological mechanisms, namely reproduction, mutation, and recombination, in order to solve these problems. Such algorithms are based on Darwin's conception of natural selection whereby weaker solutions are sorted from stronger ones and the more fit alternatives are preserved and bred, generation after generation. The ultimate goal is the development of solutions that are optimal enough to fulfill the desired purpose.

2.1.1 Definition and inspiration from natural selection