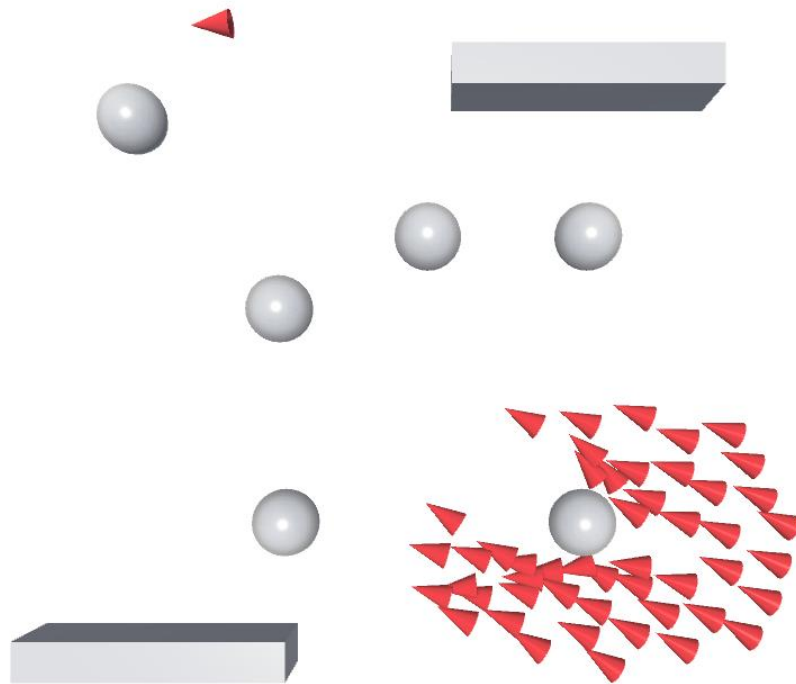


First Project of Artificial Intelligence for Games



In the first project of IAJ, the goal is to implement the BODIS algorithm (Craig Reynolds), where the behaviour of a flock of birds can be simulated by the combination of simple behaviours. The project is divided into tasks, which are grouped in levels. In order to facilitate the development of the projects, we recommend you to start working on a new level only after the previous one is finished.

The project must be submitted electronically in the form of a zip file until 17:00 of October 10th. Any project submitted after the deadline will receive 0.5 values of penalization for each hour of delay. We also recommend trying to finish the work corresponding to the two initial laboratory classes before starting to implement this project.

Level 1– Implementing flock basic movements

- Implement flock basic movements of separation, coesion and velocity matching as dynamic movement algorithms, according to the way they were taught in theoretical classes.

Level 2 – Combining basic movements

- Combine the separation, cohesion and velocity matching movements into a sigle movement using the blending movement algorithm.

- Create a scene in Unity, that automatically creates a variable set of boids¹ (1 to 50 or even more) that uses the created movement. The initial position and velocity for each boid should be determined randomly.
- Adjust the several movement parameters (neighbourhood radius, cone angle, separation force, weights for each movement type, etc), in order to obtain a realistic behaviour².

Nível 3 – Flee the Click

- Implement a behaviour that makes the flock flee from a click point whenever the user clicks with the left mouse button³.
- Integrate this behaviour with the flock movement behaviour.
- As soon as the flock (or part of it) reaches a predefined flee distance, this movement should stop affecting the flock behaviour, unless the user clicks another spot.

Nível 4 – Obstacle avoidance

- Add several obstacles to the scene (preferably of distinct types such spheres and walls).
- Implement and add an obstacle avoidance movement to the boid movement. Choose the best way to combine the several movements in order to make sure that no boid ever goes through an existing obstacle.

Nível 5 – Efficiency

- Use Unity's profiling mechanism to analyse the performance of your code.
- Taking into account the feedback from Unity's profiling, identify an aspect of the source code worth improving. Annex to the submitted code a picture with a print screen of the profiling analysis to justify your decision.
- Improve the efficiency of your code, by improving the aspect identified in the previous task. Run the profiling mechanism again and compare the new results with the previous results.

¹ The primitive `GameObject.Instantiate` can be used to create a clone of a particular `GameObject`.

²We recommend the use of visual debugging guides to better understand the individual contribution of each movement to the final result.

³ You will need to use the following Unity primitives: `Input.GetMouseButton`, `Input.mousePosition`, `camera.ScreenToWorldPoint`.