Convexity-based Collision Detection For A 2D Game Engine

COMP110 - Computer Architecture Essay

TE182503

December 1, 2015

Please include an abstract of at most 100 words (these do not count towards your word count).

1 Introduction

Collision detection is how a game engine detects how objects in world interact. Specifically when and how objects collide. Collision detection is vital within the players experience. It encompasses all aspects of the world and how they interact. Poor collision detection can break players emersion and make the game hard to play. Good collision detection is vital within games. Good collision detection can be seen as collision detection which is accurate and responsive, having an algorithm which is less time consuming is vital for having the collisions calculated so the collisions are not delayed. From the papers I looked at I found an improved algorithm to be the best using that criteria.

2 Your section title here

An improved algorithm for collision detection in 2D grapple games, kiaqiang guo is one paper I looked at. The algorithm has the objects in world bound by rectangles and circles which are aligned with the in worlds axis [1]. The algorithm is more accurate using circles as well as rectangles for fitting tightly to the object. The algorithm uses a global search to identify potential collisions and then uses a local resolution which runs calculations to see if the objects have overlapped [1]. In this algorithm the formula works out the distance between centre coordinates and the vertexes and uses that distance and point to line to assess if the vertexes were inside another object [1].

The construction of a predictive collision 2D game engine, Gaius Mulley talks about the algorithm which predicts and stores the information for collisions which could occur. The PGE can be seen as an event simulator which predicts and simulates collisions. The structure used for the predictive game engine has multiple modules [2]. The predictive game engine works by having polyhedra broken into line segments and calculates the smallest time for the next two objects which will collide [2]. The calculation is carried out to see when the collision will occur and stores this information. The objects which are set for collision are made out of circles or line segments so collisions between circle circle, line - circle and line - line [2].

The design of collision detection algorithm in 2D grapple games, Yunlan tan, Changxin Liu, is an algorithm for collision detection in 2D grapple games. The algorithm uses coordinate values of the centre a rectangle and vertexes which create a collision detection area and when the vertexes are enclosed it detects a collision [3]. Objects within the world are bounded by axis-aligned the rectangular boxes or circles which make up the collision zones. The paper presents three algorithms for different situations in game

[3]. The first is where roles get translation movement, the second is where roles make contact with walls in game and the third is for rotational movements for roles[3].

The Improved algorithm for collision detection in 2D grapple games uses the global search to improve the run time of the algorithm. This reduces run time because it finds pairs of objects which are potentially collision pairs [1]. This is a more general search eliminating non-potential pairs. Then the local resolution runs collision calculations on those pairs. In experiments presented in the paper the run times are as follows; circle - circle took 6.2ms, circle rectangle took a minimum of 4.02ms and maximum of 14.13ms and rectangle rectangle took a minimum of 10ms and maximum of 160.1ms [1]. Compared to the run time of collision detection algorithm in 2D grapple games papers results which are; for algorithm 1, 1.2ms up to 2.4ms, for algorithm 2, 15.7ms to 251.29ms, and for algorithm 3, 10ms up to 160ms [2]. The run times in this paper is comparing specific aspects of the game; each algorithm is suitable for different moments while the improved algorithm for collision detection in 2D games test is focused on time of collision between geometric objects [1]. The results show the improved algorithm takes minimal time and is more versatile as whereas the algorithms in the collision detection in 2d grapple games are specific to areas of use. The improved algorithm looks at the time between shapes so they can be used throughout the game. The construction of a predictive collision 2d game engine. Is far more complex and does scale well but with current technology the engine is not as efficient [3]. This is because the engine is working out the potential collisions and having to store the information of the collision. The approach from the improved algorithm paper is simpler. It works out the collision area using axis aligned objects and when there is overlap returns that there is a collision. This makes this algorithm using technology available now less time consuming.

3 Recommendation

I recommend An Improved algorithm if collision detection in2D grapple games because it has minimal running time so the collision will be calculated quickly with current technology available. It has tight fit axis-aligned objects which means it is the same level of accuracy as the Collision detection in 2D grapple games but is more time efficient. This collision detection method is also with current technology less time consuming then the method which predicts all of the collision and has to store that data.

4 Conclusion

The essay looked at 3 papers to assess which one I would recommended based on the quality criteria of how time consuming it. The work contributes to the world of collision detection because they introduce new methods which improve on previous methods and provide a platform for further improvements. Looking at 3 papers which present different methods of detecting collisions within a world. Two of the papers had clear results while the predictive game engine did not have clear timed results, but it is clear the algorithm scales better so it would be good to have the algorithms ran on the same system for more clearer results.

References

[1] K. Guo and J. Xia, "An improved algorithm of collision detection in 2d grapple games," in *Intelligent Information Technology and*

Security Informatics (IITSI), 2010 Third International Symposium on, pp. 328–331, April 2010.

[2] G. Mulley, “The construction of a predictive collision 2d game engine,” in *Modelling and Simulation (EUROSIM), 2013 8th EUROSIM Congress on*, pp. 68–72, Sept 2013.

[3] Y. Tan, C. Liu, and Y. Yu, “The design of collision detection algorithm in 2d grapple games,” in *Information Engineering and Computer Science, 2009. ICIECS 2009. International Conference on*, pp. 1–4, Dec 2009.