CS 5150 Project Proposal

Feiyi Ouyang

Yuheng Liu

**Summary Description**

The entertainment software industry grows very fast, even though it is relatively small compared to traditional entertainment like movies or TV, this industry has a promising future and many software development skills are required. Game development involves UI design, algorithm design and strategy design. The goal of this project is to build a game called “Missile!”, we will mainly focus on algorithm design aspect. Based on the definition of artificial intelligence, “Artificial intelligence is about making computers able to perform the thinking tasks that humans and animals are capable of.”, for this game’s implementation, we will apply general game AI algorithms. The movement of computer’s character will perform and make decisions as real human performs in the same scenario.

The purpose of this game is to design a player-computer interactive interface, the player will be identified as a plane, and the enemies will be identified as a missile.  The interface will be designed as an outer space while the player will control the plane to avoid missiles. The movement of missiles is controlled by AI algorithm that we will have implemented in this project. Once the missile hits the player, the game will end. The score of the player will be how many seconds that player hold without any missile hits the plane. Also, in order to improve the complexity, we will add a target for player to reach to increase the score. Once the player touched target, the target will appear again in a random position in the outer space.

**Environment**

For this project, we will use pygame to build the interface and interactive behavior between player and computer. This game will be a 2D game. As the features of pygame we discussed in the course, it will have the following features: “A basic game engine which can keep track of environment, changes in response in the environment and a game state machine”, we will include “many of the basic movement algorithms, jumping, folking, etc”. The definition of game environment is as following: “A dimension which collaborates game rules, objectives, subject, and theoretical aspects together as a whole to provide an interactive flow of activity.” According to this definition the rule for this game is to avoid missiles approaching the player. The object for payer is to remain this status as long as possible. The theoretical aspects for both player and computers will be discussed below as the strategic moves and key decision.

**Characters**

The character design of this game includes one player, multiple numbers of enemies and one target. The player will be shaped in a plane, the enemies and target will be shaped as dots for clearance. The plane is controller by player and the movement of enemies will be controlled by AI algorithms.

**Goals of each character**

The goal of player is to avoid collision of missiles, also identified as enemies. The player will control the plane by mouse and move it around. Also, we add a feature in this game by displaying a target that player can reach to increase the game score. The key to increase the game score is to reach the target while avoiding missiles. The calculation of the game score will be a weighted combination of how many seconds that player can hold without collision and how many targets that player can reach. The general goal of player is to gain as many as game scores as possible.

The goal of computer, also identified as enemies is to chase the player in the outer space interface and try to collide the player.

The goal of target is to be displayed among the whole game interface. Once it is touched by player, it will show up again in a positon with random coordinate x-value and y-value.

**Attack and defense behaviors of each character**

Attack behavior of player is try to reach the targets. Defense behavior would be to move around to avoid collisions of enemies.

For the behaviors of enemies, they will attack player by approaching, chasing, following and accelerating to the player, they would not have defense behavior because the player is the only one that try to survive.

Target’s behavior will be simple to design, it will wait to be touched by player and appear again in a random place. No attack or defense behavior are needed to be designed.

**Key algorithm**

Key algorithms that are applied should include random appearance of target using random number generator, the calculation of x-coordinate value and y-coordinate value, avoid duplications of positions. There should not be complex algorithm design and data structure during this part.

**Key AI algorithm**

There are numbers of game AI algorithm that will be applied to computer’s role since the player is controlled by human and the only function of target is to appear. Since the design part of this game should focus on movement of computer characters, some kinematic movement algorithm will be applied. We can combine multiple moving behavior to computer characters. We intend to implement the combination by applying following algorithms:

Since the number of computer characters will be increased as time pass to increase the difficulty, we can use different moving strategy on different objects. There will be further discussion on partition of number of computer characters that we will implement distinct moving algorithms on.

Seeking and chasing algorithm will be applied that request velocity on a long the line on direction of player and computer. Wandering algorithm and path following algorithm will also be implemented on some portion of computer characters to increase the complexity.

Conclusion as a table of key algorithms

|  |  |
| --- | --- |
| Seeking algorithm and Chasing algorithm | Main algorithm to obtain the status of computer characters by establishing goals as trying to catch the player’s character. |
| Wandering and Path Following | Increase complexity |

**Strategic moves and key decision**

Based on the goal of computer characters, the general strategic moves of computer characters are to predict the moving of player’s character. Strategies of design of movement is to increase some complexity. If there are many routines of different numbers of characters, it may confuse the player about which character is really going to hit the plane. The key decision is to adjust the portion of different algorithm on characters, based on time that player has been playing this game, the portion should be changed as time passes. For example, some of characters will try to chase and seek player, if there are some characters near the player that are just wandering or following the player, it may affect player’s judgments of moving the plane.

**Implementation approach**

Every implementation of objects and functions of algorithm will be done using pygame. There will be three classes for characters in the game: player class, computer class and target class. Player will control the plane by mouse. For the player class, we will implement the function of movement and initial the positions. The computer classes should contain the attributes of velocity, initial positions and acceleration. We will also implement the algorithm of chasing, seeking, wandering and path following. The function of increasing the the number of computer objects and deciding the portion of different algorithms on computer objects will also be implemented in main function. The target class will be designed that has attributes as initial position and the function to appear in a random place. We will use set in python data structure to avoid duplicated positions of each characters.