

ARTIFICIAL INTELLIGENCE IN VIDEOGAMES

PRACTICE PROJECT

Realizado por:
JACINTO ARIAS MARTÍNEZ
ADRIÁN SÁNCHEZ LÓPEZ

April, 2012

Abstract

Contents

1	Project definition and design principles	3
1.1	Game definition	3
1.2	Motivation	3
1.3	Platform	3
2	Artificial intelligence principles I: Computational Theory	4
2.1	Game elements	4
2.2	Knowledge extraction	4
2.3	Artificial intelligence applications	4
2.3.1	Movement problems	4
2.3.2	Path finding and collision avoidance	4
2.3.3	Units tracing	4
3	Artificial intelligence principles II: Representation and algorithms	5
3.1	Application of search techniques	5
3.2	Application of rule based systems	5
3.3	Application of CBR	5
4	Artificial intelligence principles III: Implementation	6
4.1	Software engineering and game engine implementation issues	6
4.2	AI techniques integration	6
4.2.1	Movement models	6
4.2.2	Behaving and action models	6
4.2.3	Strategy and meta-game models	6

1 Project definition and design principles

1.1 Game definition

The game we are proposing is an RTS game in which you will be managing a large horde of units in order to break down a enemy fortress.

The main feature of this game is that you won't be able to directly control your units, in case of that, you will give them orders and the units should behave according to them but with a high free will grade.

The horde will be separated into groups of units that you will be able to select. Using the mouse and the graphic interface provided, you will give these "intentional" orders to your troops, an example can be "approach to that position" or "became aggressive"

1.2 Motivation

We have chosen this kind of game because we think that the AI integration possibilities are huge. AI principles can be integrated in many of the game components, for example, in the game engine, AI techniques can be used in order to define and implement the units free will, also it can be combined with an agent-based approach. AI could also be applied to make opponents for the game, we should define their strategies, decision, etc...

1.3 Platform

The game will be developed in the 3D engine Ogre, and it should be multi-platform (both windows and linux). The models and artwork will be designed in blender and gimp. So this game is fully free software made.

2 Artificial intelligence principles I: Computational Theory

2.1 Game elements

In this game your goal is to conquer the centre of the map.

There are three main elements on the game:

- **Zombies:** This is the interactive element of the game. You will control groups of approximately ten zombies that will start spread over the map. You can order the zombies to move to a point, but if they are hungry they won't obey you and will wander randomly. If they can reach the centre of the map they should attack the enemy, but beware because they can shoot and kill the zombies.
- **Food:** Spread along the map you can also find food, food is necessary for the zombies to get strong and obey your orders. If a group of zombies find food they will eat it.
- **Enemies:** In the centre of the map you will find a bunch of enemies. They will be patrolling all the time and trying to kill your zombies if they come close enough. Your goal is to kill every enemy on the map.

You can also find obstacles and other passive elements along the map.

2.2 Knowledge extraction

2.3 Artificial intelligence applications

In this section we will identify some of the problems that can be solved using artificial intelligence solutions.

The most immediate problem to solve is the behaviour of the units. With most of the techniques we should solve problems like units movement, path finding, collision avoidance or unit tracking.

Other aspect of the game that can be involved in AI developing can be the opponent strategy. We can implement such methods in order to give the opponent more intelligence making the game most interesting. For example we can apply these techniques for choosing the army or placing in on the map.

2.3.1 Movement problems

When speaking about movement we have to difference between zombies and enemies, because their movement models are not the same.

The zombies model is based in two parameters, first of all we have the destination point which is the zombies goal to reach, on the other hand we have the hunger of the zombies group. If the zombies are hungry they would tend to wander around while reaching the point, if the hunger level gets low the zombies will appear to come closer to the point obeying the player's order directly.

When programming the zombies movement we should take those parameters in consideration and apply a straight movement and a movement pattern for randomness when wandering. These movement pattern should be applied by using AI techniques, with different techniques we could achieve richer patterns going from a random pattern to behaviours like birds flocks, particle swarm or something different.

A rules based system, a neural network or CBR can be implemented to achieve this.

2.3.2 Path finding and collision avoidance

2.3.3 Units tracing

3 Artificial intelligence principles II: Representation and algorithms

3.1 Application of search techniques

3.2 Application of rule based systems

3.3 Application of CBR

4 Artificial intelligence principles III: Implementation

4.1 Software engineering and game engine implementation issues

As this subject is not focused on videogames programming, this section will cover the basis of the implementation in order to identify and follow the different elements of the game engine. We are going to provide the few diagrams and models needed to understand the structure and the function of this application.

4.2 AI techniques integration

In this section we will talk about the structure of the game engine and how we should integrate the different techniques in order to provide not only a videogame but also a modular platform in which we can test the different approaches seen in class and study its behaviours.

On the one hand and as the game engine, we are going to introduce the different elements that we should integrate to provide a videogame interface and its elements. On the other hand as a platform for testing AI techniques we are going to let an empty place in which we can plug in the AI techniques as different modules implementing an specific model; for example, we are going to implement the hordes units movement in the game engine letting a place to plug in a model, which can be a random model or a richer approach like a rule based system or a neural network.

4.2.1 Movement models

4.2.2 Behaving and action models

4.2.3 Strategy and meta-game models