

Chapter 7

Evolving Bots' AI in Unreal™

Antonio M. Mora-García
University of Granada, Spain

Juan Julián Merelo-Guervós
University of Granada, Spain

ABSTRACT

A bot is an autonomous enemy which tries to beat the human player and/or some other bots in a game. This chapter describes the design, implementation and results of a system to evolve bots inside the PC game Unreal™. The default artificial intelligence (AI) of this bot has been improved using two different evolutionary methods: genetic algorithms (GAs) and genetic programming (GP). The first one has been applied for tuning the parameters of the hard-coded values inside the bot AI code. The second method has been used to change the default set of rules (or states) that defines its behaviour. Moreover, the first approach has been considered at two levels: individual and team, performing different studies at the latter level, looking for the best cooperation scheme. Both techniques yield very good results, evolving bots (and teams) which are capable of defeating the default ones. The best results are obtained for the GA approach, since it just performs a refinement considering the default behaviour rules, while the GP method has to redefine the whole set of rules, so it is harder to get good results. This chapter presents one possibility of AI programming: building a better model from a standard one.

INTRODUCTION

First Person Shooters (FPS) are action games where the player can only see the hands and the current weapon of his character, and has to fight against enemies by shooting at them. These games appeared at the end of the eighties in PCs

as one of the new pseudo-3D games, evolving concepts previously seen in others such as Maze Wars (1974). After the first, and famous, Wolfenstein™ and DOOM™ games, FPS games began to be played by millions of video gamers in individual player mode until the appearance of games which included multiplayer modes. Unreal™ [Unreal:Site, Unreal:Wikipedia], released for PCs by Epic Games in 1998, had great success

DOI: 10.4018/978-1-4666-1634-9.ch007

since it incorporated the best multiplayer mode to date. In that mode, up to eight players (on the same PC or connected through a network) fought among themselves, trying to defeat as many of the others players as possible, and getting the so-called *frag* for each defeat. The players move in a limited scenario or arena, where weapons and other useful items appear. The players can be human or automatic and autonomous ones, known as *bots*.

Each player has a life level which is decreased every time he receives a weapon impact; this decrement depends on the weapon power, the distance, and damaged area on the character. In addition, there are some items that can be used to increase this level or to protect the player. Also, many FPS games let the programmers modify part of their source code or engine, to build new maps, weapons or characters, and even change the enemies' artificial intelligence (AI) schemes, to get new autonomous bots.

In the latest FPS games, a change in the confrontation has been introduced: the *team battle*. There are many team modes, such as death match, conquer the hill, capture the flag or hunt and escape. The common aim in all of them is the cooperation of the individuals in each team to obtain a global gain.

That is, the main objective is to get good team behaviour, rather than a good individual conduct. But in principle, it is difficult to predict how an improvement in the individual AI of a bot can profit the whole team.

Following these ideas, in this chapter there have been implemented (and presented) two general approaches to evolve bots inside Unreal™. The first one was applied for tuning up a set of parameters, corresponding to some hard-coded values inside the bot AI code. The second method was implemented to change and improve the default set of rules (or states) that defines its behaviour.

Moreover, additional work has been performed on the implementation of *two approaches of team-based evolutionary methods*, devoted to optimizing the behaviour of the whole team, in

order to get the maximum number of frags against other teams.

In all cases we have implemented bots with a Genetic AI, or Genetic Bots (*G-Bots*). Evolutionary Algorithms, such as Genetic Algorithms [Goldberg, 1989] and Genetic Programming [Koza, 1992] have been used to improve the Unreal™ AI, given their well-known optimization capability. This way, each G-Bot improves its AI by playing a game and getting a better global behaviour in time, that is, defeating as much enemies as possible (getting frags) and being defeated as little as possible.

In the team-based case, every bot on the same team shares a common chromosome. As stated, the objective is to get bots whose behaviour would be good for the team profit.

The methodology described could be useful in AI designing tasks, starting from a common standard hard-coded or rule-based AI implementation, which would be significantly improved by means of evolutionary techniques.

UNREAL™ GAME FEATURES

As previously commented, Unreal™ is a very famous FPS for PCs published in 1998. It presented a very good single player mode, but the multiplayer possibilities gave it great success. Currently, there are many games which include multiplayer modes against humans or bots, but there are some features which made Unreal™ the best framework for developing our work:

- It includes bots with a high level AI, which was the best for a long time, since it introduced some novel techniques (such as pre-defined scripts, navigation points or states and events)
- It includes a proprietary engine programming language, called *UnrealScript* [UnrealScript], which combines C and Java syntax, with some useful features,

22 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:

www.igi-global.com/chapter/evolving-bots-unreal/66321

Related Content

Successful Game Development Partnerships between Academics and Physicians: Two Case Studies

Elena Bertozzi, Leonard R. Krilov and Dilys Walker (2013). *International Journal of Gaming and Computer-Mediated Simulations* (pp. 97-107).

www.irma-international.org/article/successful-game-development-partnerships-between-academics-and-physicians/93031

Sentiment-Enhanced Content-Based System for Online Recommendations and Rating Prediction

Akshi Kumar, Simran Seth, Shivam Gupta and Shubham (2020). *International Journal of Gaming and Computer-Mediated Simulations* (pp. 1-25).

www.irma-international.org/article/sentiment-enhanced-content-based-system-for-online-recommendations-and-rating-prediction/261255

New Wine in New Skins: Sketching the Future of Game Sound Design

Daniel Hug (2011). *Game Sound Technology and Player Interaction: Concepts and Developments* (pp. 384-415).

www.irma-international.org/chapter/new-wine-new-skins/46801

Rethinking E-Learning and Digital Natives

Il Tombul (2019). *Handbook of Research on Children's Consumption of Digital Media* (pp. 114-124).

www.irma-international.org/chapter/rethinking-e-learning-and-digital-natives/207863

Understanding Games Through Complexity Thinking Approach

Ghada Ahmed Deghedi (2018). *International Journal of Gaming and Computer-Mediated Simulations* (pp. 41-56).

www.irma-international.org/article/understanding-games-through-complexity-thinking-approach/214860