BhTSL, Behavior Trees Specification and Processing

Miguel Oliveira
Centro ALGORITMI, DI, Universidade do Minho, Portugal

Pedro Mimoso Silva
Centro ALGORITMI, DI, Universidade do Minho, Portugal

Pedro Moura
Centro ALGORITMI, DI, Universidade do Minho, Portugal

José João Almeida
Centro ALGORITMI, DI, Universidade do Minho, Portugal

Pedro Rangel Henriques
Centro ALGORITMI, DI, Universidade do Minho, Portugal

── Abstract ──────────────────────────────────

In the context of game development, there is always the need for describing behaviors for various entities, whether NPCs or even the world itself. That need requires a formalism to describe properly such behaviors.

As gaming industry has been growing, many approaches were proposed. First, finite state machines were used and evolved to hierarchical state machines. As this wasn't enough, a more powerful concept appeared. Instead of using states for describing behaviors, people started to use tasks. This concept was incorporated in behavior trees.

This paper focuses in the specification and processing of these behavior trees. A DSL designed for that purpose will be introduced. It will also be discussed a generator that produces LaTeX diagrams to document the trees, and a Python module to implement the behavior described. Aditionally, a simulator will be presented. These achievements will be illustrated using a concrete game as a case study.

1 Introduction

At some point in the video-game history, NPCs were introduced. With them came the need to describe behaviors. And with this behaviors came the need of the existence of a formalism so that they can be properly specified.

As time passed by, various approaches were proposed and used, like finite and hierarchical state machines. These are state-based behaviors, that is, the behaviors are described through states. Altough this is a clear and simplistic way to represent and visualize small behaviors, it becomes unsustainable when dealing with bigger and more complex behaviors. Some time later, a new and more powerful concept was introduced: using tasks instead of states to describe behaviors. This concept is incorporated in what we call behavior trees.

Behavior trees were first used in the videogame industry in the development of the game *Halo 2*, released in 2004. The idea is that people create a complex behavior by only programming actions (or tasks) and then design a tree structure whose leaf nodes are actions and the inner nodes determine the NPC's decision making. Not only these provide an easy

and intuitive way of visualizing and designing behaviors, they also provide a good way to work with scalability through modularity, solving the biggest issue from state-based design. Since then, multiple gaming companies adopted this concept and, in recent years, behavior trees are also being used in areas like Artificial Inteligence and Robotics.

This paper focuses in the specification and processing of these behavior trees. For this purpose, it will be introduced an DSL, and a compiler that can generate a LATEX diagram to visualize and a Python module to implement the described behavior.

These achievements will be illustrated using a concrete game as a case study.

- Motivation - Design goals, ir direto ao assunto

2   State of the Art

- conceitos básicos

3   Architecture and Specification

- Gramática - Especificação

4   Tools

- processador

5   Example

6   Conclusion