

Tipo de artículo: Artículo original

Módulo de Inteligencia artificial de NPC para videojuegos de género shooter

Artificial intelligence module of NPC for videogames of the shooter genre

Marcos Henrique Pelegrino Infante¹ , <https://orcid.org/0000-0001-8122-5751>

Enelis Blanca Cuba Rondón² , <https://orcid.org/0000-0001-5997-7906>

¹ Centro Vertex de Tecnologías Interactivas, Facultad 4, Universidad de las Ciencias Informáticas. mhpilgrim97@gmail.com

² Centro Vertex de Tecnologías Interactivas, Facultad 4, Universidad de las Ciencias Informáticas. ebcuba@uci.cu

* Autor para correspondencia: mhpilgrim97@gmail.com

Resumen

La inteligencia artificial ha formado parte de la evolución de la informática desde su primer reconocimiento como término y campo en los años 50. En su desarrollo se fue acoplado a distintos campos y medios, incluidos los videojuegos, de los cuales, hasta el más sencillo, puede poseer personajes no jugables con características inteligentes. En este trabajo se desarrolló un módulo de comportamientos inteligentes para personajes no jugables para videojuegos del género *shooter*, utilizando una de las técnicas de inteligencia artificial más utilizadas en la creación de videojuegos. El mismo fue realizado para usar con el motor gráfico *Unity* y guiado por la metodología de desarrollo de software XP, con el objetivo de brindar a los desarrolladores una base técnica reutilizable, flexible y extensible para así facilitar el proceso en futuros proyectos. Se realizó un demo con el módulo para probar su reusabilidad y flexibilidad.

Palabras clave: inteligencia artificial, comportamientos inteligentes, módulo, Unity, videojuego.

Abstract

Artificial intelligence has been part of the evolution of informatics since its first acknowledgement as a term and a field in the 50's. During its development it started to take part of different fields and media, including videogames, of which, even the simplest, can possess non-player characters with smart characteristics. In this work it was developed a module of smart behaviors for non-player characters of videogames of the shooter genre, using one of the most used techniques in the videogame development. The same was made to be used with the graphic engine Unity and guide by the XP development methodology, with the goal of giving the developers a reusable technical base, flexible and extensible in order to ease the process in future projects. A demo of the module was made in order to test its reusability and flexibility.

Keywords: artificial intelligence, module, smart behaviors, Unity, videogame.

Recibido: 20/12/2021

Aceptado: 18/03/2022

En línea: 04/06/2022



Esta obra está bajo una licencia *Creative Commons* de tipo **Atribución 4.0 Internacional**
(CC BY 4.0)

Introducción

En la Universidad de las Ciencias Informáticas (UCI) se encuentra el Centro Vertex de Tecnologías Interactivas, donde existe una línea para el desarrollo de videojuegos. Actualmente, se trabaja en la creación de varios videojuegos de diferentes tipos. Entre los videojuegos en desarrollo se encuentra un juego de género *shooter*, un tipo de juego que consiste principalmente en disparar a enemigos y evadir sus proyectiles. En la actualidad, el proyecto cuenta con enemigos inertes, mostrándose como objetos inmóviles, haciéndolos predecibles y fáciles de vencer. Esto puede causar que la jugabilidad del videojuego pierda atractivo para el usuario y no sea comercial (Parkinson & Martínez, 2021), (Rozo et al., 2018). A partir de la situación antes mencionada, se identifica el siguiente problema de investigación: *¿cómo dotar de comportamientos inteligentes a los personajes no jugables del prototipo de género shooter del Centro Vertex de Tecnologías Interactivas?*

Como objeto de estudio se define: los comportamientos inteligentes de los personajes no jugables en los videojuegos de género *shooter*. Para dar solución al problema de investigación planteado, se define como objetivo general: desarrollar un módulo de comportamientos inteligentes aplicable a personajes no jugables para videojuegos de género *shooter*. Dentro del objeto de estudio de la investigación se precisa como campo de acción: comportamientos inteligentes en NPC de videojuegos de género *shooter* del Centro Vertex de Tecnologías Interactivas.

Materiales y métodos

Métodos teóricos:

Histórico y Lógico: se empleó para la fundamentación y sistematización de los aspectos teóricos contemplados en el desarrollo de la investigación acerca de la evolución y las tendencias actuales del desarrollo de mecánicas de videojuegos y demás elementos relacionados con el contenido del trabajo.

Inductivo y Deductivo: mediante el uso de este método, se realizó un estudio enfocado en los videojuegos del género *shooter* que incorporan una IA avanzada, para posteriormente, analizar a detalle sus características. Haciendo lo antes mencionado se crearon las bases para el desarrollo del módulo gracias a una mayor comprensión sobre el objeto de investigación.

Análisis y Síntesis: este método se utilizó en el análisis de conceptos, definiciones y la bibliografía existente sobre la IA y su uso en videojuegos del género *shooter*.

Métodos empíricos:

Consulta de información: se utilizó para la búsqueda de información sobre la IA aplicada a videojuegos del género



shooter en diversas fuentes de información como libros, revistas científicas, tesis de doctorado, trabajos científicos y sitios webs especializados.

Metodología de desarrollo: Entre las posibles metodologías se encuentran la metodología XP (*Extreme Programming*, por sus siglas en inglés), Scrum y el Marco de Trabajo Ingenieril para el desarrollo de videojuegos, esta última siendo descartada por no tratarse del desarrollo de un videojuego. Esto deja como posibilidades a XP y Scrum, que presentan semejanzas como ser metodologías ágiles, ambas hacen uso de las Historias de Usuario y realizan continuamente entregas al cliente en cortos períodos de tiempo. A pesar de sus parecidos, la más acorde a la problemática es la metodología XP, porque sus tiempos de entrega son en menores lapsos de tiempo, los cuales están comprendidos en un máximo de dos semanas por iteración, mientras que Scrum llega a las tres semanas y más. XP se basa en retroalimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. XP se define como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico.

Herramientas y tecnologías: Para la selección de las herramientas y tecnologías a utilizar en la propuesta solución se tuvo en cuenta el entorno de trabajo utilizado en el centro Vertex, específicamente en la línea de desarrollo de videojuegos, pues usando estos se ha demostrado que se obtienen aplicaciones de mayor calidad en poco tiempo. También la solución estará diseñada para ser aplicada en el desarrollo de videojuegos, por lo que, para lograr integraciones futuras con mayor calidad, es recomendable usar las mismas tecnologías. A continuación, serán descritas las herramientas y tecnologías con sus características:

Unity 2020.3.12f1: se utilizará para unir y organizar los elementos que conforman el módulo haciendo uso de las facilidades y librerías que brinda el software. Es un motor gráfico ampliamente usado a nivel global tanto por personas como un pasatiempo, como por empresas profesionales, ya sea para el desarrollo de videojuegos como de animación 2D y 3D (*Unity*, n.d.) Esta herramienta permite crear contenido para el entretenimiento multiplataforma (es capaz de exportar tanto para dispositivos Android como iOS, Windows, PlayStation, Xbox y sistemas de realidad virtual como Oculus), ya que cuenta con una extensa y bien detallada documentación (*Unity*, n.d.). Se usará en este caso en su versión *LTS* (del inglés *Long Term Service*) o de **soporte a largo plazo**, la que suele ser la versión más estable creada para largos períodos de desarrollo sin la necesidad de tener que actualizar el motor gráfico constantemente.



Visual Studio Code 1.58: se utilizará como editor de código para crear y modificar los *scripts* del lenguaje de programación C# utilizados en la creación del proyecto. Disponible para múltiples plataformas como Windows, MacOS y Linux. Viene con soporte nativo para JavaScript, TypeScript y Node.js y cuenta con un amplio ecosistema de extensiones para otros lenguajes como Java, C++, Python y C#.

Lenguaje C#: se utilizará como lenguaje de programación para los *scripts* del módulo. Es un lenguaje de multipropósito con gran flexibilidad y múltiples disciplinas de la programación como imperativa, declarativa, funcional, genérica, orientada a objetos y orientada a componente creado y diseñado por Anders Hejlsberg, quien en sus inicios le otorgó el nombre de *Cool (C-like Object Oriented Language)* pero por razones de registro se cambió a C# ([C#] *Un Poquito de Historia de C#*, n.d.). Desde la salida de C# 2.0 en 2005 ha crecido ampliamente como lenguaje a la par de otros como Java, formando su propia identidad, ya que hace uso de la cosificación para proveer objetos genéricos que pueden ser usados como otras clases, con generación de código al nivel del tiempo de carga de las clases.

Técnica de inteligencia artificial:

Finite State Machine (FSM): Máquina de Estado Finito (en español), es un algoritmo relativamente simple de IA, donde se crean una lista de posibles eventos que el NPC puede experimentar. Posteriormente se asignan respuestas específicas que el NPC tendría en cada caso (Ituarte & Floris, 2021), (Ituarte, 2021). Un ejemplo del uso de esta técnica es en el juego Wolfenstein 3D de 1992, en el cual, los desarrolladores tuvieron en cuenta todas las posibilidades que podría experimentar un soldado enemigo en una situación específica como:

1. El usuario abre una puerta y entra en el rango de visión del enemigo.
2. El enemigo lo nota y procede a atacar.
3. Si el jugador desaparece y los puntos de salud de la IA son bajos, procede a curarse.
4. El jugador no ha vuelto a aparecer, el enemigo pasa a patrullar.

Esta lista es compilada, y en cada situación le dirá al NPC cómo actuar. En la imagen 1 se puede apreciar un ejemplo de los distintos estados entre los que alterna una IA enemiga usando esta técnica.



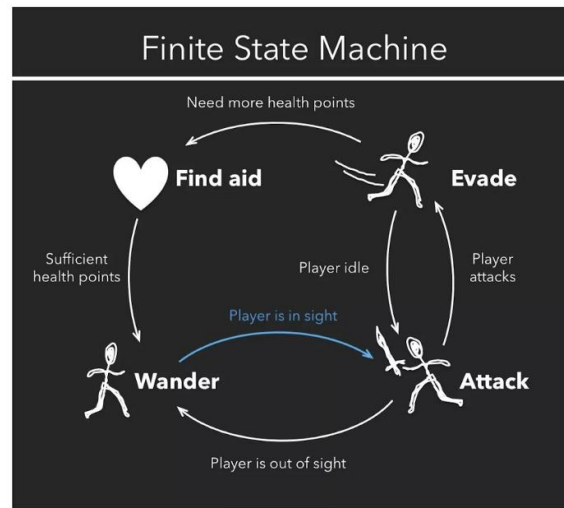


Figura 1. Secuencia de estados de la IA de un enemigo.

Aunque el ejemplo es bastante simple, se puede lograr un ciclo mucho más complejo y con mucho más detalle en dependencia de lo que se quiera lograr (Torregrosa et al., 2018), (Carmena Plaza, 2018). *FSM*, aunque eficiente, no es factible para juegos con árboles de acciones más complejos como los juegos del género de estrategia, donde esta clase de acciones puede volverse predecible por ser tan repetitiva, dando lugar a que el usuario solo tenga que aprender el patrón de la IA para superar a la misma.

Resultados y discusión

Se creó un paquete de Unity con las herramientas básicas para dotar de comportamientos inteligentes a los personajes no jugables de videojuegos de género shooter. El producto final contiene un demo para poner en práctica las funcionalidades del módulo. Destaca principalmente por su libre acceso y código abierto, permitiendo modificaciones por parte de los desarrolladores de forma completamente gratuita. Las relaciones de clases del proyecto se ven reflejadas en la figura 2, mientras en la figura 3 se puede apreciar el uso de la técnica de máquina de estado finito en la solución:



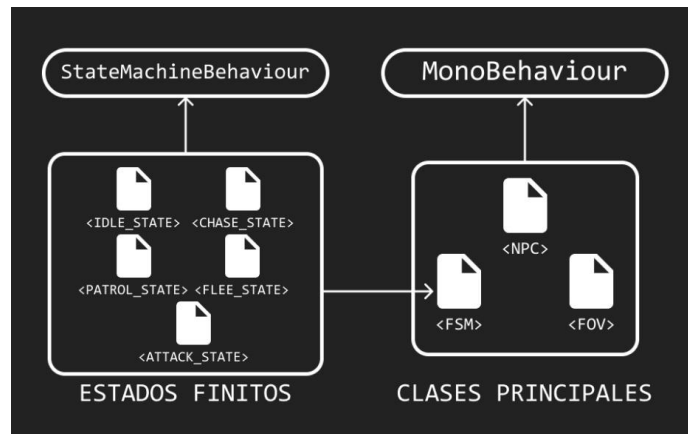


Figura 2. Relación de clases del módulo.

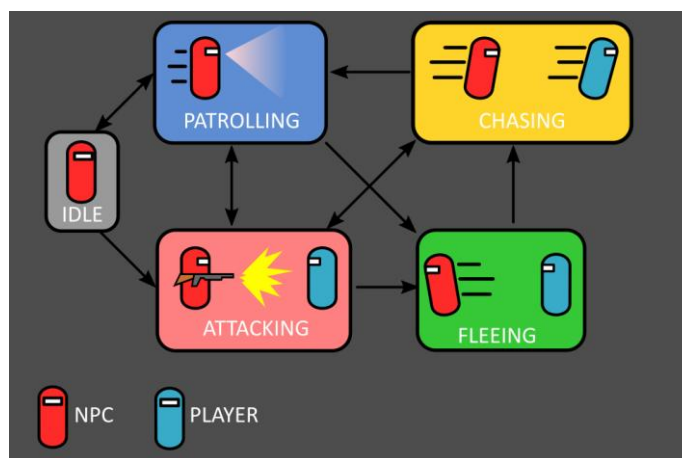


Figura 3. Diagrama de la técnica FSM aplicada.

Conclusiones

Se seleccionó la de *FSM* por sus características que la hacen acorde a la situación problemática. Se llegó a la conclusión de que la metodología más adecuada es la XP, esto por las condiciones del equipo para la realización del trabajo. Con la culminación de la investigación se lograron los siguientes resultados:

- Se creó un módulo para videojuegos *shooter* que contribuye a mejorar la jugabilidad de los mismos.



- Se implementó una técnica básica de inteligencia artificial que cumple con los requisitos funcionales trazados que mejora la experiencia de usuario de los videojuegos de estilo *shooter* que carecen de estas funcionalidades.

Conflictos de intereses

Los autores no poseen conflictos de intereses.

Contribución de los autores

1. Conceptualización: Enelis Blanca Cuba Rondón.
2. Curación de datos: Enelis Blanca Cuba Rondón.
3. Análisis formal: Enelis Blanca Cuba Rondón.
4. Investigación: Marcos Henrique Pelegrino Infante.
5. Metodología: Marcos Henrique Pelegrino Infante.
6. Administración del proyecto: Enelis Blanca Cuba Rondón.
7. Recursos: Marcos Henrique Pelegrino Infante.
8. Software: Marcos Henrique Pelegrino Infante.
9. Supervisión: Enelis Blanca Cuba Rondón.
10. Validación: Enelis Blanca Cuba Rondón.
11. Visualización: Marcos Henrique Pelegrino Infante.
12. Redacción – borrador original: Enelis Blanca Cuba Rondón.
13. Redacción – revisión y edición: Enelis Blanca Cuba Rondón.

Referencias

- Carmena Plaza, J. (2018). *Desarrollo de Técnicas de Inteligencia Artificial Aplicadas a los Juegos de Atari* https://e-archivo.uc3m.es/bitstream/handle/10016/29516/TFG_Jorge_Carmena_Plaza.pdf?sequence=1
- Ituarte, I. A. (2021). Comentario del libro Coicaud, Silvia (2020). Potencialidades didácticas de la inteligencia artificial. *Boletín SIED*(3), 100-106. <https://revista.sied.mdp.edu.ar/index.php/boletin/article/download/50/77>



- Ituarte, I. A., & Floris, C. (2021). “La Inteligencia artificial es como un gran caleidoscopio, que cada vez adquiere mayor desarrollo” Entrevista a Silvia Coicaud. *Boletín SIED*(3), 90-99.
<https://revista.sied.mdp.edu.ar/index.php/boletin/article/download/59/76>
- Parkinson, C., & Martínez, R. (2021). Técnicas de percepción para el uso de Inteligencia Artificial en el desarrollo de los videojuegos: Caso de Estudio Proyecto 1810. XXVII Congreso Argentino de Ciencias de la Computación (CACIC)(Modalidad virtual, 4 al 8 de octubre de 2021),
- Rozo, E. J. B., Montoya, R. C., & Páez, J. (2018). Videojuegos: Avances tecnológicos en aplicación de física e inteligencia artificial. *Letras ConCiencia Tecnológica*, 61-78.
<https://revistas.itc.edu.co/index.php/letras/article/download/114/110>
- Torregrosa, J., Lara-Cabrera, R., Bello-Orgaz, G., & Shih, P.-C. (2018). Inducción de emociones a través del diseño de videojuegos. XVIII Conferencia de la Asociación Española para la Inteligencia Artificial,

