Imagen que contiene nombre de la empresa

Descripción generada automáticamente

UNIVERSIDAD POLITÉCNICA DE MADRID

ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA DE SISTEMAS INFORMÁTICOS

AGENTES INTELIGENTES EN UNITY

TRABAJO FIN DE GRADO

Grado en Ingeniería del Software

Curso académico 2020-2021

**Autor:** Rafael Sánchez Rodríguez

**Tutores:** Jesús Mayor Márquez y Alberto Díaz Álvarez

RESUMEN

Text

ABSTRACT

Text

ÍNDICE DE CONTENIDO

[INTRODUCCIÓN 1](#_Toc107835377)

[PLANTEAMIENTO DEL PROBLEMA 3](#_Toc107835378)

[OBJETIVOS 6](#_Toc107835379)

[ESTADO DE LA CUESTIÓN 7](#_Toc107835380)

[HISTORIA DE LA INTELIGENCIA ARTIFICIAL 11](#_Toc107835381)

[ALAN TURING 11](#_Toc107835382)

[PRIMEROS AÑOS Y PRIMER INVIERNO 12](#_Toc107835383)

[AÑOS 80 Y SEGUNDO INVIERNO 13](#_Toc107835384)

[HASTA LA ACTUALIDAD 13](#_Toc107835385)

[REDES NEURONALES ARTIFICIALES 15](#_Toc107835386)

[MACHINE LEARNING Y DEEP LEARNING 16](#_Toc107835387)

[AGENTES INTELIGENTES 17](#_Toc107835388)

[AGENTES BASADOS EN APRENDIZAJE 17](#_Toc107835389)

[ENTRENAMIENTO DE IA 19](#_Toc107835390)

[INTELIGENCIA ARTIFICIAL EN VIDEOJUEGOS 19](#_Toc107835391)

[DESARROLLO 22](#_Toc107835392)

[HERRAMIENTAS A UTILIZAR 22](#_Toc107835393)

[UNITY 22](#_Toc107835394)

[RESULTADOS 28](#_Toc107835395)

[CONCLUSIONES 29](#_Toc107835396)

[LINEAS FUTURAS 30](#_Toc107835397)

[BIBLIOGRAFIA 31](#_Toc107835398)

ÍNDICE DE GRÁFICAS Y FIGURAS

[Figura 1. Máquina pensando como una persona. 1](#_Toc107831537)

[Figura 2. Ejemplo Simple. 3](#_Toc107831538)

[Figura 3. Ejemplo Complejo. 3](#_Toc107831539)

[Figura 4. Tipos de Observaciones. 4](#_Toc107831540)

[Figura 5. Funcionamiento de un agente. 5](#_Toc107831541)

[Figura 6. Máquinas y resolución de problemas complejos. 7](#_Toc107831542)

[Figura 7.Tipos de Inteligencia Artificial. 8](#_Toc107831543)

[Figura 8. Evolución de la IA. 10](#_Toc107831544)

[Figura 9. Alan Turing. 11](#_Toc107831545)

[Figura 10. Inviernos de la IA, representación a color. 13](#_Toc107831546)

[Figura 11. Los Inviernos de la IA. 13](#_Toc107831547)

[Figura 12. Estructura de una Red Neuronal Artificial. 15](#_Toc107831548)

[Figura 13. Machine Learning vs Deep Learning. 16](#_Toc107831549)

[Figura 14.Funcionamiento básico de un Agente Inteligente 17](#_Toc107831550)

[Figura 15. Estructura de un Agente Inteligente basado en aprendizaje 18](#_Toc107831551)

[Figura 16. Agente inteligente finaliza satisfactoriamente su entrenamiento 19](#_Toc107831552)

[Figura 17. Trabajadores construyendo granjas en Civilization VI 20](#_Toc107831553)

[Figura 18. Partida de Starcraft 2 21](#_Toc107831554)

[Figura 19. Componentes GO: Encabezado 22](#_Toc107831555)

[Figura 20. Componentes GO: Transform 23](#_Toc107831556)

[Figura 21. Componentes GO: Mesh Filter 23](#_Toc107831557)

[Figura 22. Componentes GO: Mesh Renderer 23](#_Toc107831558)

[Figura 23. Componentes GO: Collider 24](#_Toc107831559)

[Figura 24. Componentes GO: Material 24](#_Toc107831560)

[Figura 25. Componentes GO: Rigidbody 25](#_Toc107831561)

[Figura 26. Componentes GO: Script 25](#_Toc107831562)

[Figura 27. Componentes GO: Cámara 26](#_Toc107831563)

[Figura 28. Componentes GO: Iluminación 26](#_Toc107831564)

ÍNDICE DE ABREVIATURAS

-IA: Inteligencia Artificial.

-ML: Machine Learning.

-DL: Deep Learning.

-TFG: Trabajo de Fin de Grado.

-IAG: Inteligencia Artificial General.

-GO: GameObject.

-UI: Interfaz de usuario.

# INTRODUCCIÓN

La Inteligencia Artificial (IA) es una de las herramientas más potentes de las que se dispone en la actualidad. Las máquinas son capaces de hacer tareas más rápida, precisa y eficazmente que un humano. Sin embargo, su principal problema es que no son capaces de adaptarse a situaciones imprevistas.

Tradicionalmente, las máquinas funcionan siguiendo unos comandos fijos dados por el operario/programador. Estas máquinas no pueden ser consideradas IA, ya que ningún tipo de inteligencia o toma de decisiones está presente, sólo la repetición mecánica de las instrucciones dadas. Un ejemplo serían los brazos robóticos en una fábrica de automóviles.

Por otro lado, la IA ofrece a la máquina unas directrices generales para conseguir un objetivo, y deja en manos de la IA decidir la mejor forma de conseguirlo. Esta IA podrá pensar de forma artificial, analizando su entorno y eligiendo la acción a tomar según estos parámetros. En otras palabras, es capaz de tomar decisiones. Hoy en día, en la mayoría de los casos la IA sigue estando bastante limitada en cuanto a sus opciones, y la toma de decisiones se reduce a reaccionar a un estímulo con una acción predeterminada. Un claro ejemplo sería la IA usada en videojuegos, la cual es capaz de responder a las acciones del jugador y el entorno, pero siempre de acuerdo con un plan preestablecido. En un juego de estrategia, una granja será construida si la comida escasea y en un juego de disparos los enemigos responderán a ser disparados buscando cobertura. Mientras que esto funciona bien a la hora de simular inteligencia, en algunos casos con más éxito que en otros, esta IA no “piensa” en el sentido tradicional de la palabra.

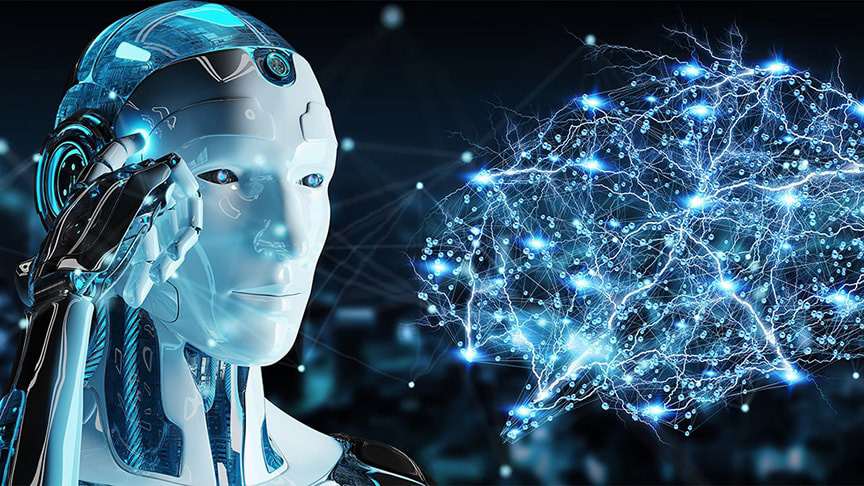


Figura . Máquina pensando como una persona.

Por desgracia, la tecnología no está todavía al nivel de crear IA capaz de pensar al mismo nivel que una persona, pero ya es capaz de conseguir cosas bastante impresionantes. Tecnologías como Machine Learning (ML) y posteriormente Deep Learning (DL), permiten que una IA pueda aprender por sí misma, basado en experiencia, tomando sus propias decisiones y muchas veces encontrando soluciones que un humano nunca habría imaginado.

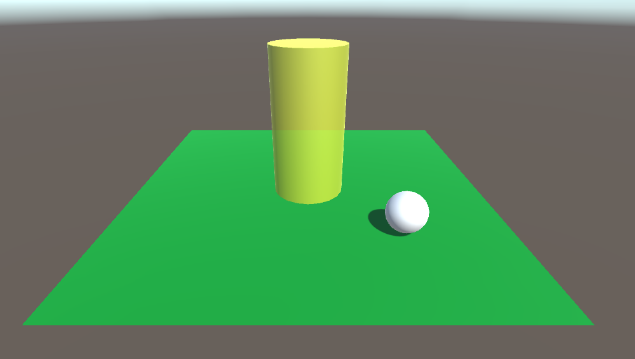
Este Trabajo de Fin de Grado (TFG) explorará la capacidad de un agente inteligente de aprender en un entorno virtual. Se espera observar tácticas de equipo al enfrentar dos grupos de agentes entre ellos y se variarán los parámetros del entrenamiento para ver cómo varía el resultado.

## PLANTEAMIENTO DEL PROBLEMA

Una IA moderna puede aprender basándose en la experiencia, pero esta experiencia debe venir de algún sitio. Por ello, estos agentes inteligentes son entrenados en entornos controlados, donde pueden explorar sin peligro de dañarse ellos, el material o a la gente cercana.

Por esta razón, los videojuegos son un excelente campo de entrenamiento para ellos, ya que proporcionan un entorno virtual libre de riesgo donde poder entrenar a los agentes.

Se plantean dos casos de ejemplo para esta sección, uno simple, donde una esfera en una superficie plana debe mantenerse dentro de una pequeña zona que varía de posición al cabo de un tiempo; y uno complejo, donde el agente se encarga de mantener la estabilidad de un avión a la hora de aterrizar.

Imagen de la pantalla de un video juego

Descripción generada automáticamente con confianza baja

Figura . Ejemplo Simple. Figura . Ejemplo Complejo.

En un videojuego se puede configurar de forma muy precisa en entorno en el que el agente entrenaría posteriormente. Desde un entorno simple con físicas básicas hasta entornos tremendamente complejos donde tener en cuenta muchas variables distintas, como el peso, el rozamiento, el viento, etc.

El agente deberá ser capaz de enfrentarse al entorno y cumplir su objetivo. Para ello, necesita recibir los datos necesarios. El agente no ve y tampoco sabe lo que tiene que hacer al principio, para entrenarlo necesita recibir observaciones de su entorno. Estas observaciones pueden venir de lanzar rayos que le lleven información sobre los objetos con los que colisionan, que sería lo más parecido a un sentido de la vista.

Otra forma es darle observaciones directamente. En un entorno sencillo esto puede simplemente ser la posición del objetivo, aunque el agente no sea originalmente capaz de identificarlo como tal todavía. En entornos más complejos podría necesitar valores como su peso, la velocidad y dirección del viento, etc.



Figura . Tipos de Observaciones.

Aparte de las observaciones, un agente debe de poder realizar acciones. Éstas vendrán definidas por el programador e incluyen todos los movimientos, cambios, o acciones distintas que el agente podrá realizar. En el caso simple, podría tener solo dos acciones: moverse adelante o atrás y moverse a la izquierda o a la derecha. En el caso complejo podrían ser acciones avanzadas que controlen los alerones del avión.

Una vez que el agente tiene acceso a todas las observaciones que necesita, así como definidas las acciones que puede realizar, el siguiente paso para entrenarlo es la entrega de recompensas y penalizaciones. Un agente no sabe nada al empezar el entrenamiento, no sabe lo que significan las observaciones que recibe ni qué es lo que tiene que hacer. El agente empezará a realizar las acciones que tiene disponibles de forma aleatoria. Es responsabilidad del programador premiarle o castigarle por realizar ciertas acciones. En el entorno simple, en el que el objetivo del agente es simplemente mantenerse dentro de una zona en una superficie, podría recibir pequeñas recompensas cada segundo que esté dentro y pequeñas penalizaciones por cada segundo que esté fuera. En el entorno más complejo, recibiría recompensas por mantener el avión estable y aterrizar sin incidentes y recibiría penalizadores por volcar el avión o estrellarse al aterrizar.

Una vez definidos todos los parámetros, el agente realizará acciones y observará cómo repercuten en sus observaciones y sus recompensas. Con dada nuevo paso irá aprendiendo más qué es beneficioso y qué es perjudicial, y buscará soluciones que le garanticen la mayor cantidad de recompensas con el menor número de penalizaciones.

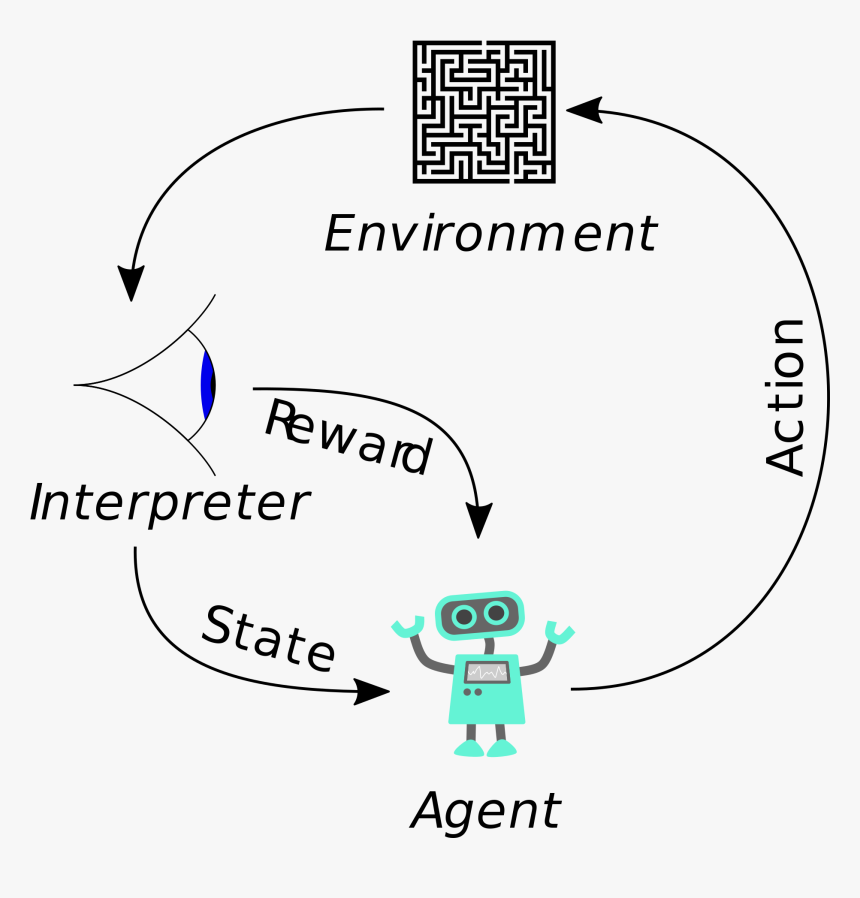


Figura . Funcionamiento de un agente.

El entorno propuesto para este TFG es el de un estadio espacial donde dos grupos de 3 naves cada uno compiten en un juego de capturar la bandera. El estadio incluirá obstáculos y las naves serán capaces de disparar. El objetivo es coger la bandera enemiga de su base y llevarla a la base aliada.

## OBJETIVOS

En orden de cumplir con los objetivos propuestos en el apartado anterior, se proponen objetivos en dos fases, la primera siendo la creación del entorno y el videojuego, y la segunda el entrenamiento de los agentes en sí mismos.

Los objetivos de la primera fase son:

* Diseñar el entorno de la base dónde se moverán las naves.
* Diseñar los obstáculos que se encontrarán en el entorno en forma de asteroides.
* Diseñar las naves, banderas y bases, de tal forma que haya una diferencia visible de color entre los dos equipos: el Equipo Rojo y el Equipo Azul.
* Crear las mecánicas pertinentes que permitan a las naves moverse cómo si estuvieran en gravedad 0, disparar proyectiles, reaparecer tras ser destruidas al pasar un corto tiempo y coger la bandera.

Los objetivos de la segunda fase son:

* Crear el agente que controlará las naves. Debe incluir las observaciones adecuadas, poder realizar todas las acciones esperadas y recibir recompensas y penalizaciones adecuadas.
* Crear los grupos de los agentes para que piensen en equipo.

Además de estos objetivos en cada fase, habría unos objetivos finales que son los que realmente se quieren observar:

* Comprobar que los agentes son capaces de cumplir el objetivo y llevar la bandera enemiga a su propia base.
* Comprobar distintas formas de recompensar al agente para ver cómo varía el entrenamiento y ver si alguna es más efectiva a la hora de entrenarlo.
* Variar la configuración interna del agente, para ver si esto acelera el proceso de entrenamiento o si reduce la efectivad del agente a la hora de lograr su objetivo.

# ESTADO DE LA CUESTIÓN

En este apartado se va a hablar de conceptos generales de la IA, su historia, dónde estamos en cuanto a su desarrollo, qué es una red neuronal artificial, qué es un agente inteligente, así como su aplicación en videojuegos.

Empezando con una definición de Inteligencia Artificial. La Inteligencia Artificial es el tipo de inteligencia expresada por las máquinas que les permite realizar tareas que tradicionalmente han sido relacionadas con capacidades cognitivas humanas como el aprendizaje o la resolución de problemas. (Russell & Norvig, 2009)

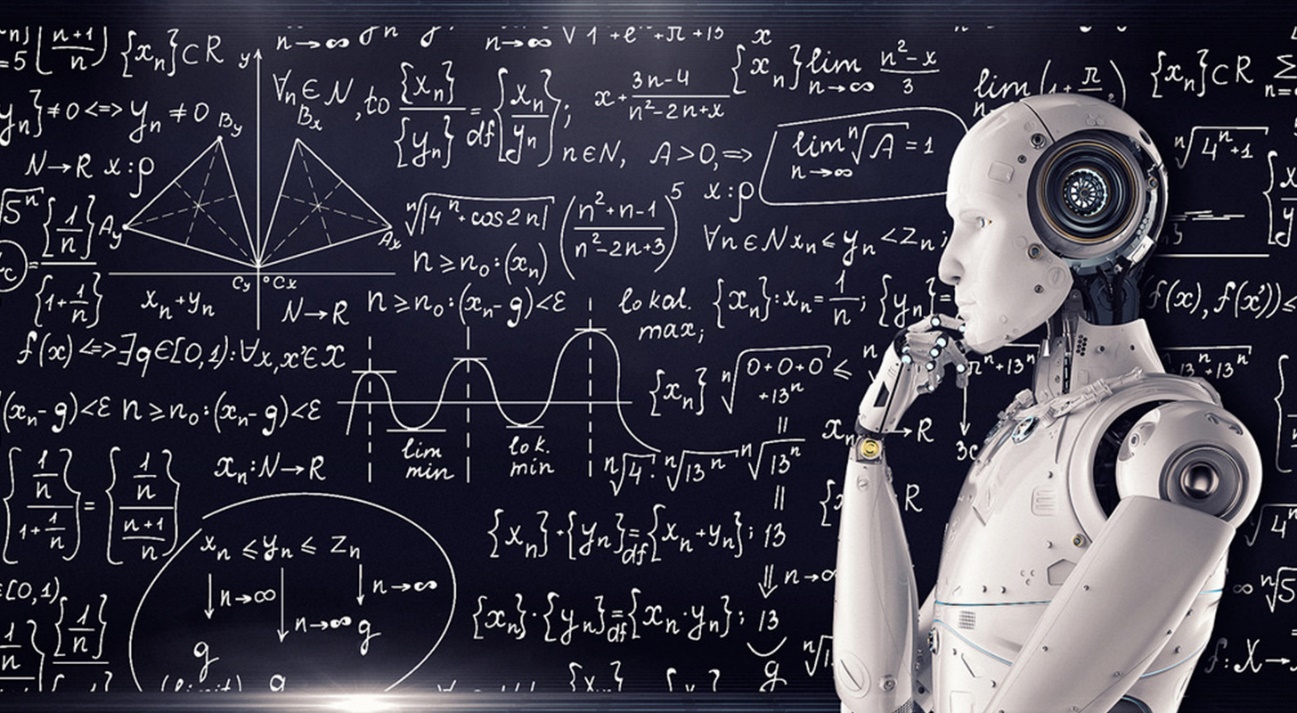


Figura . Máquinas y resolución de problemas complejos.

Según las máquinas se vuelven más capaces de realizar tareas complejas, tareas que se consideraba que requerían inteligencia son eliminadas de la definición de IA. Llamado el Efecto IA, se dice que “la IA es cualquier cosa que no se haya hecho todavía.” (Maloof, 2018)

Tareas que en la actualidad son consideradas de IA incluyen el entendimiento del lenguaje, la capacidad de competir en juegos de alto nivel de estrategia (ajedrez) y juegos de información imperfecta(póker), coches que se conducen solos, simulaciones militares, etc.

Los objetivos que se persiguen con la IA incluyen razonamiento, representación del conocimiento, planificación, aprendizaje, procesamiento del lenguaje natural, percepción, y la habilidad de mover y manipular objetos. (Russell & Norvig, 2009) Sin embargo, el mayor objetivo a largo plazo es la Inteligencia Artificial General (IAG), o Inteligencia Artificial Fuerte. La IAG sería la inteligencia mostrada por las máquinas capaz de pensar y razonar cómo un ser humano. (Kurzweil, 2005)

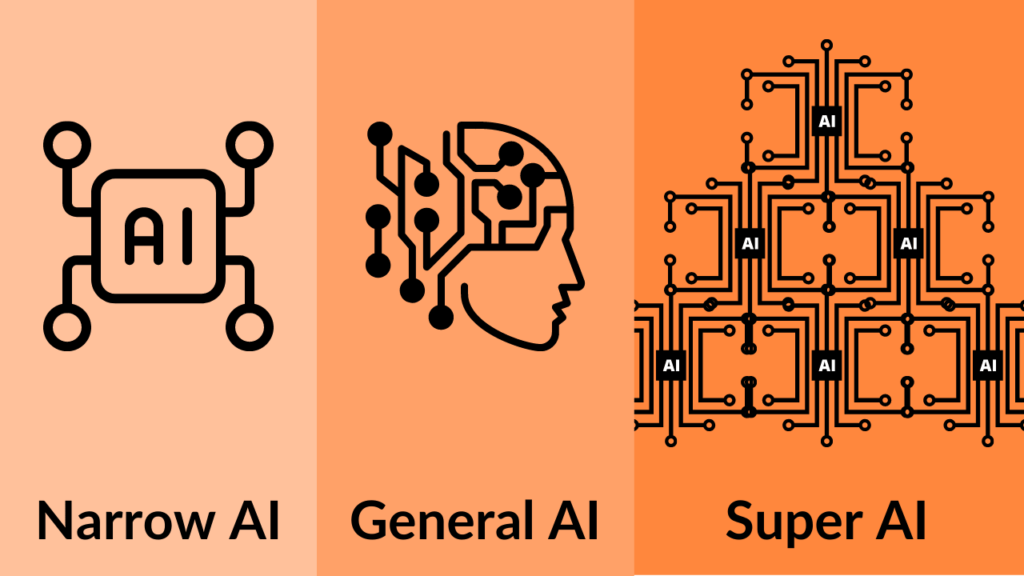


Figura .Tipos de Inteligencia Artificial.

La investigación en IA está dividida en subcampos que, normalmente han tenido problemas comunicándose entre ellos. Estos subcampos pueden diferenciarse en consideraciones técnicas, como objetivos concretos, como la Robótica o el Machine Learning; el uso de herramientas concretas, como la Lógica o las Redes Neuronales Artificiales; o diferencias filosóficas o sociales. (McCorduck, 2004)

Para perseguir los objetivos de la IA, los principales enfoques que se han seguido son:

* Inteligencia Artificial Simbólica: a mediados de los años 50, gracias al acceso a ordenadores digitales, se empieza a investigar la posibilidad de que la inteligencia humana pueda reducirse a la manipulación de símbolos. Diversas Universidades de renombre hicieron sus investigaciones por separado, cada una a su manera. Carnegie Mellon University se centró en la simulación cognitiva, Standford se centró en la lógica y el MIT se centró en lo que llamaron “scruffy” o antilógica. Estos enfoques fueron llamados más tarde GOFAI, o “Good Old Fashioned AI” (Haugeland, 1985) Durante un tiempo estas técnicas consiguieron grandes avances que hicieron que muchos pensaran que este era el objetivo. En los años 70, las tres escuelas se juntaron en un modelo basado en el conocimiento, generando las primeras formas de software con IA exitoso. (McCorduck, 2004)
* Inteligencia Computacional: para los años 80, el progreso en inteligencia artificial simbólica se paró ya que no era posible imitar procesos cognitivos humanos como la percepción, la robótica, el aprendizaje o el reconocimiento de patrones. Muchos investigadores empezaron a buscar enfoques subsimbólicos. Estos enfoques eran capaces de enfrentarse al problema de la IA sin representaciones de conocimiento específicas. (Nilsson, 1998) La Inteligencia Artificial Integrada es un enfoque que rechaza los símbolos y se centra en los problemas que permiten a los robots moverse y sobrevivir, considerando que para conseguir un alto nivel de inteligencia ciertos aspectos del cuerpo como el movimiento o la vista son necesarios. (McCorduck, 2004) Otro enfoque que se desvía de los símbolos es la Inteligencia Artificial Computacional, que se centra en las conexiones generadas en redes neuronales artificiales. (Russell & Norvig, 2009)
* Métodos Estadísticos: en los años 90 los investigadores de Inteligencia Artificial empezaron a utilizar herramientas matemáticas mucho más sofisticadas como el modelo oculto de Markov, la teoría de la información y la teoría de la decisión. Esto permitió mucha más colaboración con campos como la matemática, haciendo que la investigación de IA se volviera mucho más científica. (Russell & Norvig, 2009) (McCorduck, 2004)

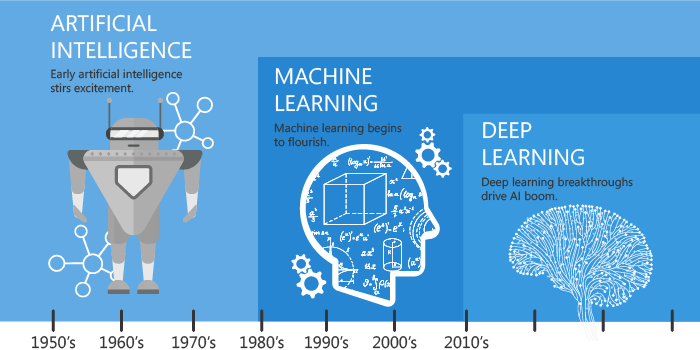


Figura . Evolución de la IA.

Las herramientas más usadas en la investigación de la Inteligencia Artificial son la optimización matemática, las redes neuronales artificiales y métodos basados en estadística, probabilidad y economía. Esto hace que el ámbito de la IA atraiga de muchos campos de conocimiento distintos como la informática, matemáticas, psicología, lingüística, filosofía, y muchos más.

## HISTORIA DE LA INTELIGENCIA ARTIFICIAL

Si bien el estudio de la IA es un suceso relativamente reciente, empezando su camino como disciplina científica a mediados de los años 50, el concepto de una inteligencia creada de forma artificial, viene desde la Antigüedad. En tiempos de los antiguos griegos ya existía este concepto en forma de criaturas mitológicas y fue estudiado por algunos filósofos. (Russell & Norvig, 2009).

Sin embargo, más allá de obras de ficción, la IA no fue sujeto de estudio hasta la llegada de la Teoría de la Computación de Turing, donde se sugiere que una máquina debería ser capaz de simular deducción matemática usando exclusivamente unos y ceros. Esto junto a varios avances concurrentes en neurobiología, teoría de la información y cibernética llevó a la creencia de poder crear un cerebro electrónico. El primer trabajo reconocido de IA es el diseño de unas neuronas artificiales en 1943. (Russell & Norvig, 2009)

### ALAN TURING

No se puede hablar de historia de la Inteligencia Artificial sin mencionar al gran Alan Turing, padre de la computación y la inteligencia artificial y el señor gracias al cual los informáticos tenemos trabajo.



Figura . Alan Turing.

Nacido en Londres en junio de 1912, es más famoso por su trabajo en ciencia de teoría computacional, mas ese es sólo una de sus grandes aportaciones. En 1936, crea la máquina de Turing, una máquina basada en un modelo matemático abstracto que manipula símbolos en una cinta de acuerdo con una serie de reglas. Esta máquina de Turing es capaz de hacer cualquier algoritmo que un ordenador puede realizar. Un sistema que sea capaz de hacer todos estos algoritmos se dice que es “Turing Complete” (Hodges, 1983)

Turing fue un miembro crucial de la unidad británica de criptoanálisis durante la segunda guerra mundial, a la que proporcionó técnicas de decodificación usando máquinas electromecánicas que ayudaban a contrarrestar los efectos de la máquina enigma. Gracias a él, se consiguieron grandes victorias en el frente atlántico. Es posible que nuestra historia fuera radicalmente distinta sin sus aportaciones. (Copeland, 2012)

Tras la guerra, trabajó en el Laboratorio Nacional de Física y en la Universidad de Manchester, trabajos en los que diseña varios de los primeros computadores electrónicos programables digitales. También durante este periodo, en 1950, formula la famosa Prueba de Turing, que permite evaluar la inteligencia de una máquina según si sus respuestas son distinguibles de las de un ser humano. (Leavitt, 2007)

En 1952 fue procesado por homosexualidad, eligiendo castración química en vez de ir a la cárcel. Dos años más tarde se suicida con cianuro, a escasos días de su 42 cumpleaños. No fue hasta 2009 que el gobierno británico pidió perdón por el trato que recibió, seguido poco más tarde, en 2013 por un perdón de la reina de Inglaterra y una ley en 2017 que perdonaba a todos que fueran juzgados igual, conocida como la “Ley de Alan Turing”. (BBC, 2016)

### PRIMEROS AÑOS Y PRIMER INVIERNO

Pese a ello, no sería hasta 1956 que el término “Inteligencia Artificial” sería adjudicado a una nueva rama de estudio de la ciencia, separándolo de la cibernética. (McCorduck, 2004) Se hicieron avances asombrosos en los primeros años, como jugar a las damas o hablar inglés, o cualquier cosa que hiciera a una máquina parecer inteligente. Al principio el mayor inversor era el Departamento de Defensa de los Estados Unidos, pero había varios laboratorios dedicados a la inteligencia artificial por todo el mundo. (McCorduck, 2004)

El optimismo inicial era tal que varios académicos preveían que sería posible crear inteligencias artificiales generales en cuestión de una generación. (Minsky, 1967)

Sin embargo, como suele pasar con las predicciones, especialmente las meteorológicas, las cosas no ocurrieron como se esperaba. Esas primeras estimaciones fallaron al medir la dificultad de varias tareas que quedaban por realizar, el progreso se ralentizó y para 1974 Estados Unidos paró sus inversiones. Los años siguientes serían considerados el Primer Invierno de la IA, una etapa de financiación y avance escasos. (Russell & Norvig, 2009)



Figura . Inviernos de la IA, representación a color.

### AÑOS 80 Y SEGUNDO INVIERNO

A principios de los años 80, el éxito comercial de los sistemas expertos revivió el interés en la IA y una nueva ola de inversiones fue dedicada a su investigación. Los sistemas expertos son un tipo de programa que simula el conocimiento y capacidad analítica de expertos humanos. (Nilsson, 1998)

Esta nueva ola de inversiones y optimismo no duraría mucho, con la IA volviendo a ser olvidada en 1987, iniciando el Segundo Invierno de la IA, este durando mucho más tiempo que el anterior. (McCorduck, 2004)

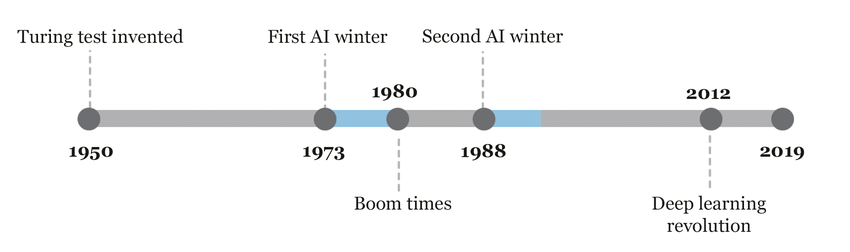


Figura . Los Inviernos de la IA.

### HASTA LA ACTUALIDAD

Poco a poco la IA fue recuperando su reputación y para finales de los años 90 y principios del siglo XXI, la IA se usaba para buscar soluciones específicas a problemas específicos, como la logística, el *data mining* y diagnósticos médicos. (Russell & Norvig, 2009)

Este enfoque tan centrado permitió a los investigadores producir resultados verificables, explotar más métodos matemáticos y colaborar con más campos de estudio como la estadística, la economía y las matemáticas. (Russell & Norvig, 2009)

Los avances a la hora de producir ordenadores más rápidos, las mejoras de los algoritmos y el acceso a grandes cantidades de datos permitieron grandes avances en Machine Learning. Esto llevó al nacimiento del Deep Learning en 2012, que ha dominado el mercado desde entonces. (McKinsey & Company, 2018)

## REDES NEURONALES ARTIFICIALES

Las Redes Neuronales Artificiales son un sistema computacional creado para simular una red neuronal biológica cómo la de los cerebros animales. Se basa en las conexiones entre nodos, llamados neuronas artificiales.

Estas conexiones pueden mandar mensajes de unas neuronas a otras, igual que en un cerebro biológico. Tras recibir una señal, una neurona analizará el mensaje y mandará sus propios mensajes.

En el caso de las neuronas artificiales, el mensaje que transmite la señal es un número real, y su procesamiento se basa en pasarlo por una función matemática, cuyo resultado será el que dicte que mensaje y a quién mandará la neurona. Las neuronas tienen un peso relativo que indica lo importantes que son en la red neuronal.

Al igual que los ogros o las cebollas, una red neuronal artificial tiene capas, en las que ordena a sus neuronas. Normalmente cada capa está especializada según el tipo de transformación que realizan las neuronas a sus entradas. Las señales viajan desde la primera capa, la capa de *inputs*, hasta la última capa, la capa de *outputs*, tras atravesar una o varias capas intermedias.



Figura . Estructura de una Red Neuronal Artificial.

## MACHINE LEARNING Y DEEP LEARNING

En la década de los 80, apareció una rama de la AI, el machine learning, que utiliza algoritmos matemáticos que permiten a las máquinas aprender. El machine learning es una forma analítica de resolver problemas mediante la identificación, la clasificación o la predicción. Los algoritmos aprenden de los datos introducidos y luego utilizan este conocimiento para sacar conclusiones de nuevos datos.

Ya en el siglo XXI, en el año 2012, apareció una nueva rama del machine learning, el deep learning. El deep learning es muy similar al machine learning, pero usa algoritmos distintos. Mientras que el machine learning trabaja con algoritmos de regresión o con árboles de decisión, el deep learning usa redes neuronales artificiales.

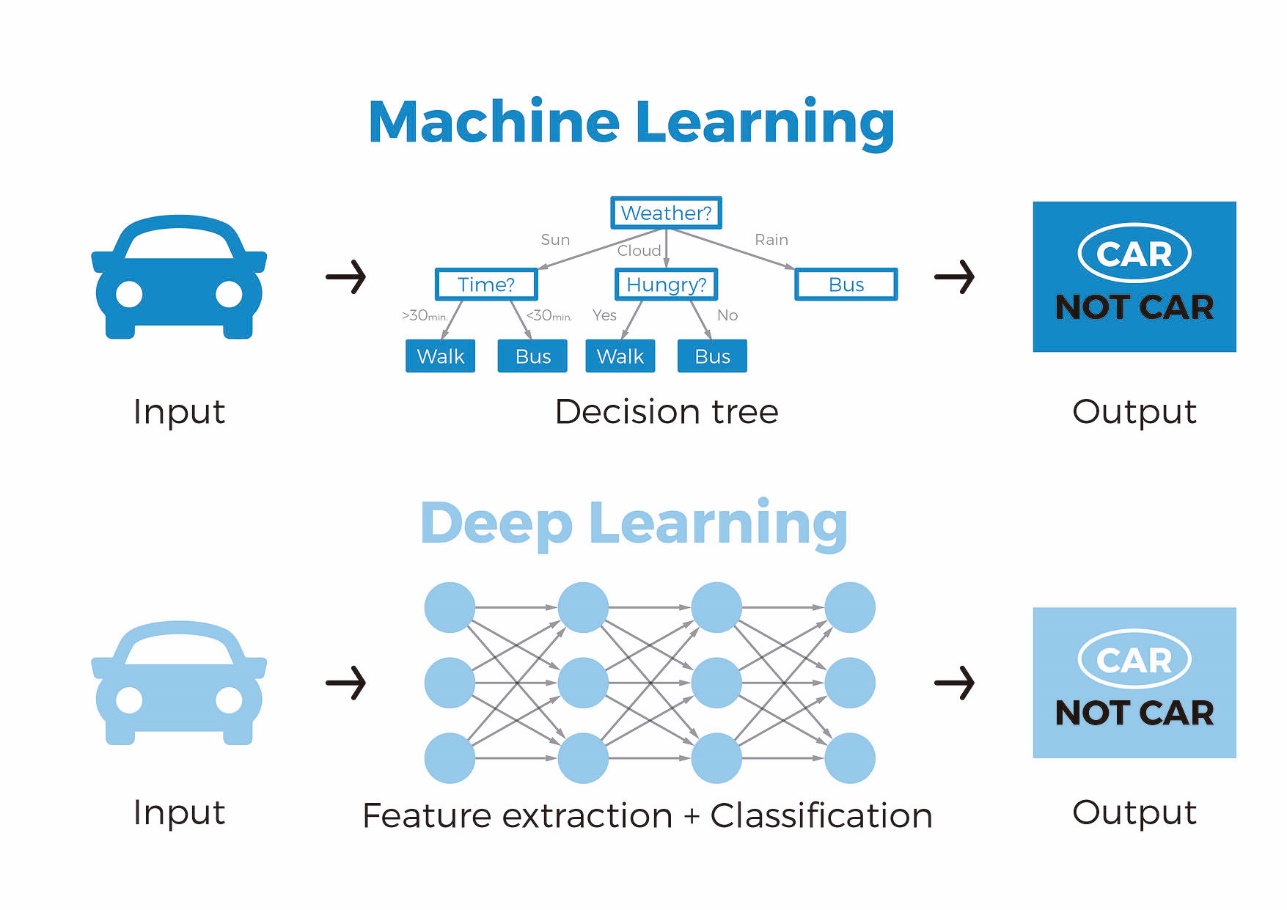


Figura 13. Machine Learning vs Deep Learning.

## AGENTES INTELIGENTES

Un agente inteligente es una entidad autónoma que actúa dirigiendo sus actividades hacia completar un objetivo en un entorno concreto haciendo observaciones mediante unos sensores. También pueden aprender o usar conocimiento para conseguir sus objetivos. (Russell & Norvig, 2009)

Los agentes perciben el entorno, lo analizan y realizan una acción de acuerdo con ciertos parámetros. Según el resultado de realizar esta acción el agente es premiado o castigado.

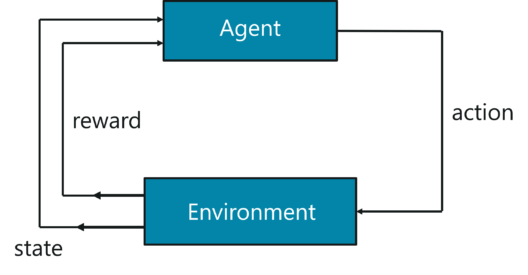


Figura .Funcionamiento básico de un Agente Inteligente

Según cómo perciben, analizan, la forma de los parámetros de decisión y de cómo se den las recompensas, existen varios tipos de agentes, desde aquellos agentes simples basados en acciones reflejo, los agentes basados en modelos, los basados en objetivos, los basados en utilidad y los basados en aprendizaje. (Russell & Norvig, 2009)

### AGENTES BASADOS EN APRENDIZAJE

Los agentes basados en aprendizaje tienen la ventaja de que operan en entornos completamente desconocidos y obtienen soluciones a sus problemas que sobrepasan lo que un conocimiento inicial le podría dar.

Una profundización del esquema genérico de la Figura 14, sería este:

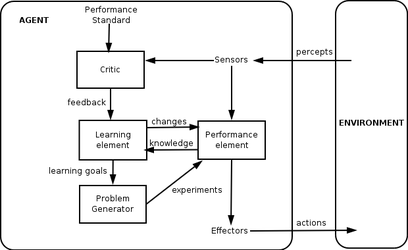


Figura . Estructura de un Agente Inteligente basado en aprendizaje

Consta de diversas partes:

* El “learning element” o elemento de aprendizaje se encarga de realizar mejoras en la toma de decisiones.
* El “performance element” o actuador es responsable de elegir las acciones externas a realizar.
* El “critic” o crítico determina lo bien o mal que el agente está cumpliendo su objetivo.
* El “problem generator” o generador de problemas sugiere acciones que podrían llevar a experiencias nuevas de las que aprender más.

El actuador sería el agente en sí, que recibe observaciones y toma decisiones de acuerdo con unos preceptos. El elemento de aprendizaje, el crítico y el generador de problemas son los añadidos que le permiten aprender. Esto genera un agente que se modifica a sí mismo, mejorando sus capacidades y aumentando las posibilidades de éxito en sus objetivos.

## ENTRENAMIENTO DE IA

Existen varias formas de entrenar una inteligencia artificial, dependiendo del problema, los recursos disponibles y los objetivos. Las principales son el aprendizaje supervisado, el aprendizaje no supervisado y el aprendizaje por refuerzo. (Russell & Norvig, 2009)

El aprendizaje supervisado usa un conjunto de datos de muestra, previamente etiquetados y clasificados, para generar un modelo predictivo basado en la combinación de entradas y salidas de la muestra. El agente aprende de ese conjunto de entradas y salidas generando una función inferida que puede aplicar en casos nuevos.

El aprendizaje no supervisado funciona de forma similar al supervisado, pero el conjunto de prueba no está clasificado ni etiquetado, es decir, en ved de contener entradas y salidas, sólo contiene entradas. Esto permite al agente generar sus propias categorías para clasificar los datos.

El aprendizaje por refuerzo es el que permite a un agente inteligente tomar acciones en un entorno y ser recompensado o penalizado por ellas. El objetivo de un agente inteligente entrenando por refuerzo es maximizar las recompensas acumuladas. Es el estilo de aprendizaje más parecido a la forma humana de aprender. El refuerzo son las recompensas y penalizaciones, que permiten al agente saber cuando está haciendo algo bien y cuando debe explorar otras posibilidades.

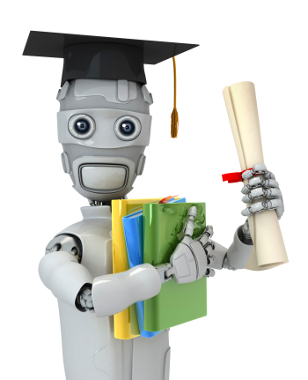


Figura . Agente inteligente finaliza satisfactoriamente su entrenamiento

## INTELIGENCIA ARTIFICIAL EN VIDEOJUEGOS

En los videojuegos la inteligencia artificial se utiliza principalmente para generar personajes no jugadores (PNJs), o NPCs de sus siglas en inglés, capaces de generar respuestas adaptativas o inteligentes similares a una inteligencia humana.

Otra forma importante que toma la inteligencia artificial en los videojuegos es como controlador. Por ejemplo, la IA que controla a un oponente en los juegos Age of Empires o Civilization.

Si bien la inteligencia artificial ha sido fundamental en los videojuegos desde sus orígenes, esta IA es diferente de la académica. Su objetivo es mejorar la experiencia de juego del jugador, no para realizar machine learning o tomar decisiones.

La principal forma que tienen los videojuegos de representar una inteligencia artificial es mediante árboles de decisión, cuya complejidad variará según lo complejo que sea el juego en sí.

Otra forma bastante común de IA en juegos de estrategia es el método 4X, que divide la carga de trabajo de una IA central con toda la información en 4 sub-IA con acceso a limitado a información, cada una especializada en una tarea. Por ejemplo, la IA central puede notar una falta de alimentos, y manda a la IA de crecimiento que construya una granja. A partir de ese momento la IA general deja de ser relevante, y la IA de crecimiento se encarga del resto, véase seleccionar un lugar adecuado, activar un trabajador cercano y ponerle a construir la granja.



Figura . Trabajadores construyendo granjas en Civilization VI

Si bien la IA es importante en la industria del videojuego, IA de aprendizaje automático es raramente el caso utilizado. Uno de los mayores ejemplos donde sí se usa aprendizaje es en AlphaStar.

AlphaStar es una inteligencia artificial desarrollada por DeepMind, la sección de deep learning de Google. Usando entrenamiento por refuerzo, AlphaStar se convirtió en el mejor “jugador” de Starcraft 2 del mundo, superando a los mejores jugadores a nivel mundial.

Starcraft 2 es un juego de estrategia en tiempo real famoso por su complejidad, debido a los grandes niveles de precisión y rapidez necesarios para controlar todo lo que ocurre en el mapa al mismo tiempo.



Figura . Partida de Starcraft 2

A base de entrenarse contra sí misma en lo que habrían sido 100 años para una persona, AlphaStar fue capaz de derrotar a los mejores jugadores del mundo, y todo ello estando limitada igual que lo estaría una persona.

# DESARROLLO

En esta sección se va a hablar del proceso que se ha usado para desarrollar este TFG. Empezando por una descripción de las herramientas a utilizar para mejor comprensión de las secciones posteriores cuando se haga referencia a componentes de Unity, por ejemplo.

Las Herramientas a utilizar van a ser:

* Unity versión 2020.3.2f1: Unity es el motor de desarrollo de videojuegos que se va a usar para crear el entorno en el que los agentes van a entrenar.
* Visual Studio Comunity 2019 versión 16.9.3: Visual Studio es un entorno de desarrollo integrado (IDE) que se usará para desarrollar el código necesario para el correcto funcionamiento tanto del entorno en el que se moverán los agentes cómo la conexiones entre este entorno y el agente, así como la configuración de los agentes.
* MLAgents versión 2.0.0-exp.1: MLAgents es un plugin de Unity que permite la creación de los agentes y proporciona las herramientas necesarias para hacerlos funcionar.

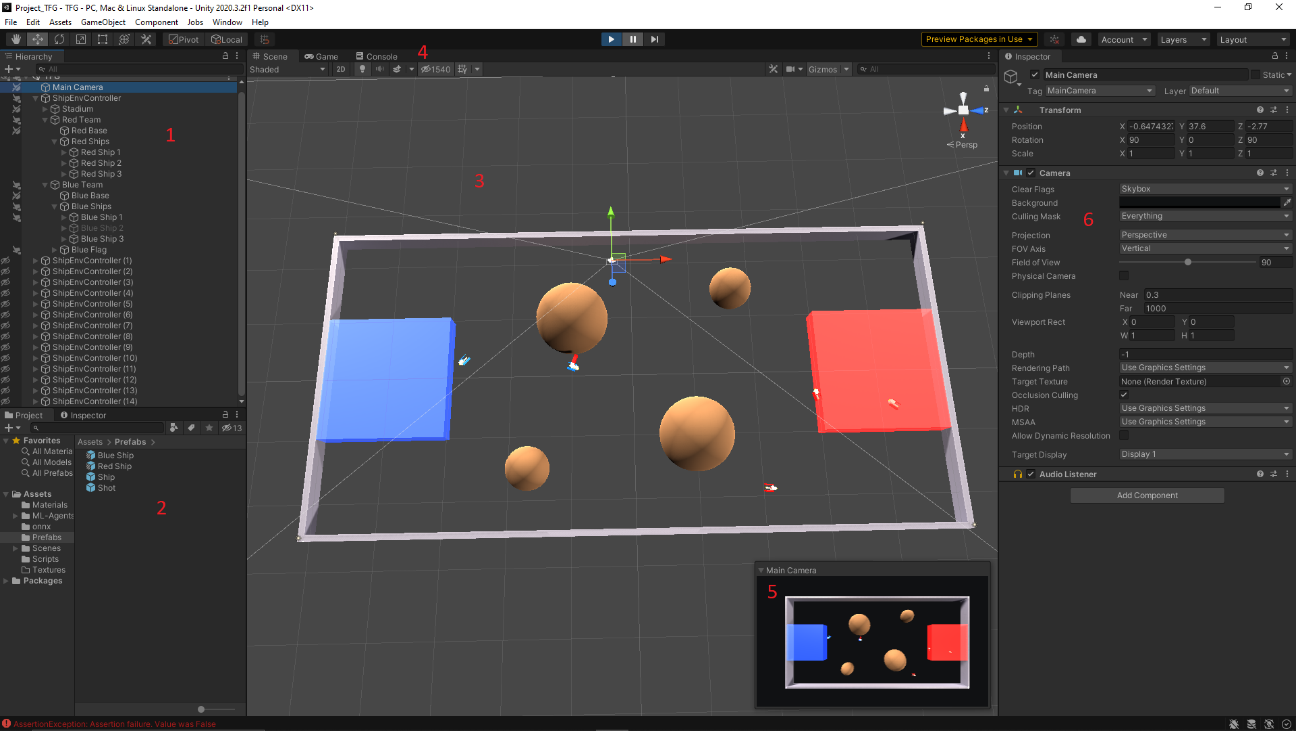
A continuación se describirán los componentes y conceptos necesarios de cada una de las herramientas.

## HERRAMIENTAS A UTILIZAR

### UNITY

Para este proyecto se va a usar la versión 2020.3.2f1. Para empezar, una introducción a la interfaz (UI) y mencionar los componentes principales que se van a usar en Unity y sus tipos.

#### Interfaz de Usuario



#### GameObject

El componente más importante en Unity es el GameObject. Todo el entorno creado en Unity está formado por GameObjects con distintas funcionalidades, desde elementos de escenografía hasta cámaras y luces, incluyendo a las formas virtuales de los agentes. Un GameObject puede tener asociados una gran diversidad de componentes que serán los que le den las propiedades particulares. Los componentes más relevantes de cara a este proyecto son:

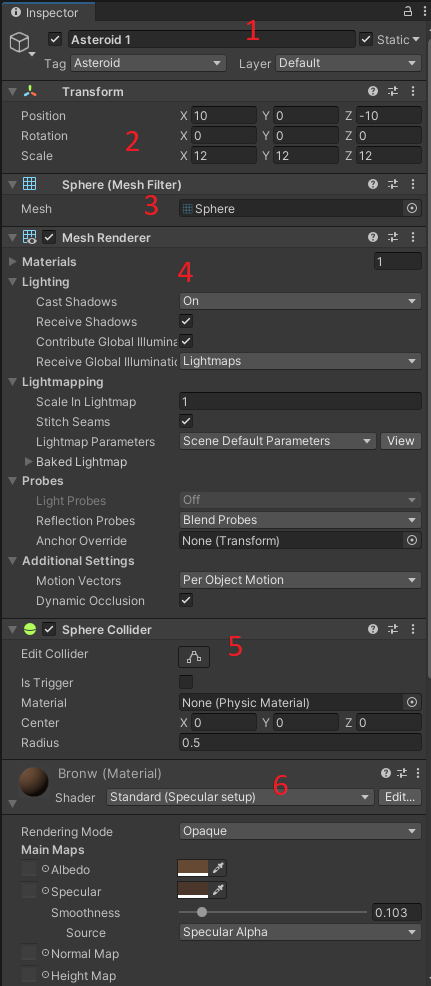


Figura . Componentes GO: Encabezado

1. Encabezado: incluye el nombre del GameObject; el tag, que se usa para relacionar GameObjects y poder identificar elementos similares; la capa, que se usa para poder activar y desactivar grupos de GameObjects para facilitar el desarrollo del entorno, en este caso siendo un entorno relativamente sencillo, sin niveles, y que precisaba de la visualización constante de todos los componentes, se ha dejado todo en una única capa default.

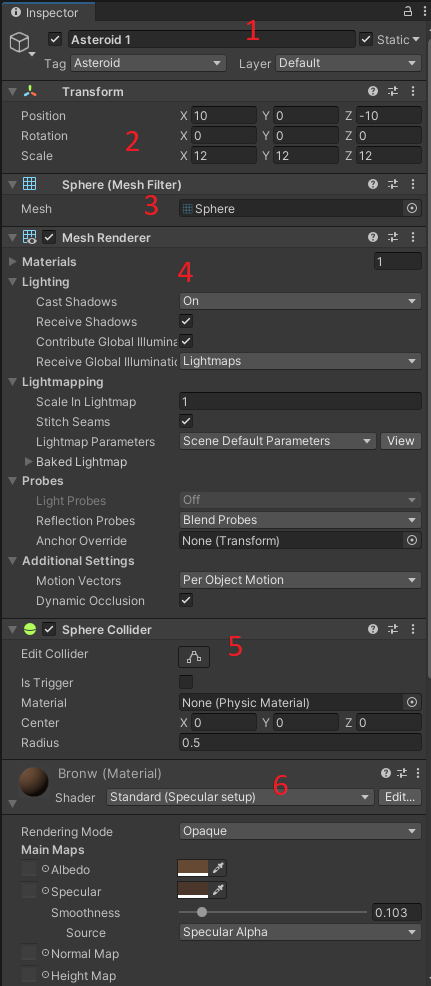


Figura . Componentes GO: Transform

1. Transform: el transform es el único componente obligatorio que tienen todos los GameObjects. Representa su posición, escala y rotación en el entorno virtual, y esta puede ser referida como la transform global o local. La transform global tiene el 0 donde el entorno tiene el 0, el default. La transform local por lo general es la misma que la global, hasta que el GameObject pasa a ser hijo de otro GameObject, en cuyo caso el 0 pasará a ser la posición del padre en el entorno. Se detallará más sobre padres e hijos en la sección de Jerarquía.

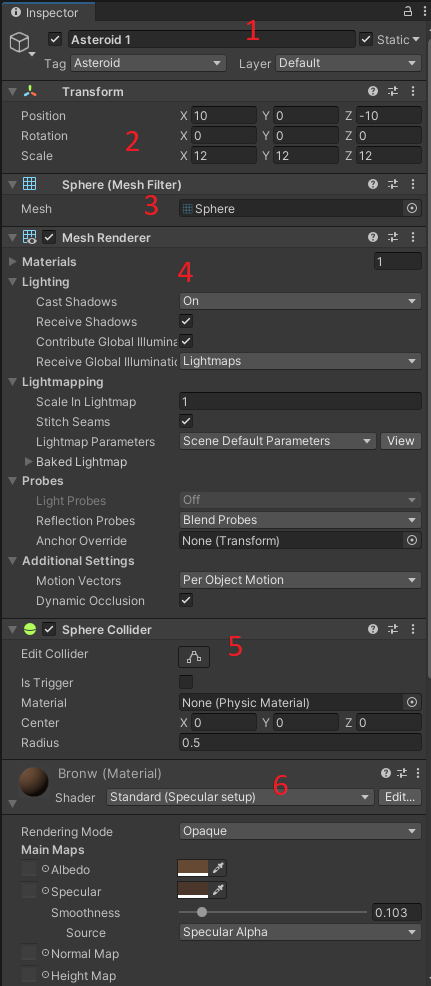


Figura . Componentes GO: Mesh Filter

1. Mesh Filter: el mesh filter es el componente que determina la forma que toma el GameObject. Hay algunos prefabricados como la esfera, el cubo, el cilindro… pero también se puede usar uno genérico que genera una forma según un archivo que tú le proporciones.

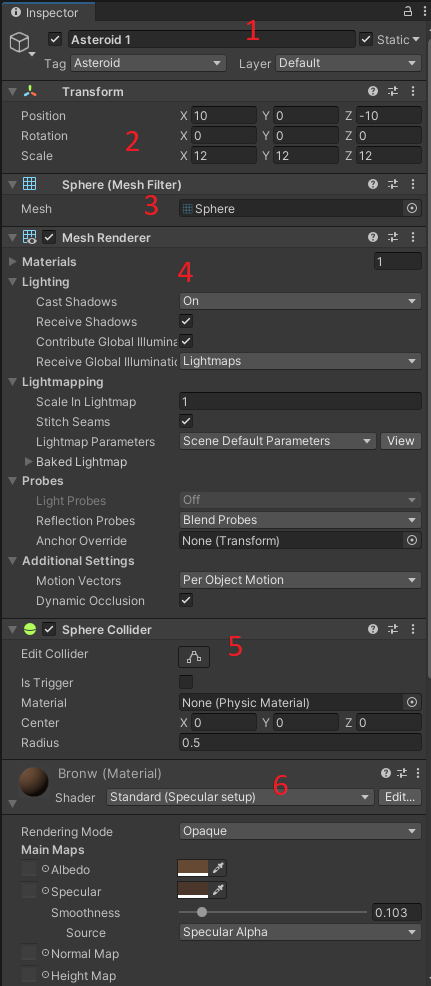


Figura . Componentes GO: Mesh Renderer

1. Mesh Renderer: el mesh renderer es el componente que genera la forma virtual del mesh filter en el entorno según las especificaciones que se determinan en los detalles.

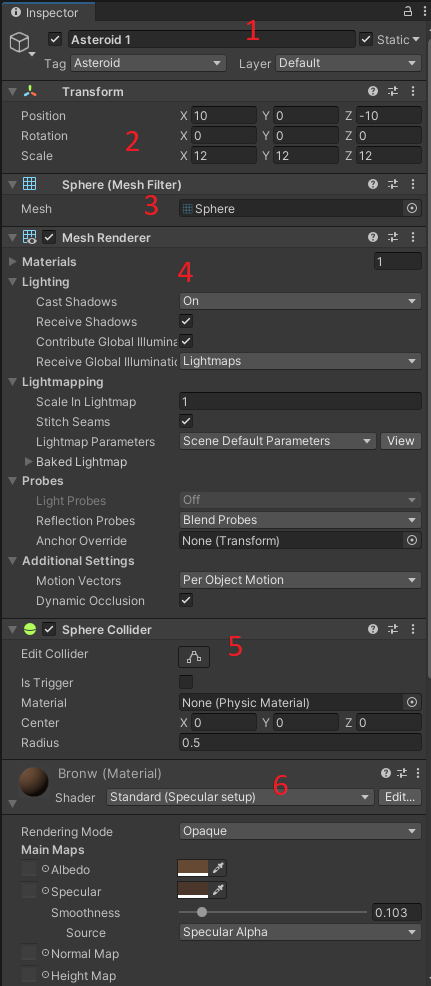


Figura . Componentes GO: Collider

1. Collider: el collider es el componente que permite al GameObject interactuar con otros GameObjects del entorno, permitiendo lanzar triggers al colisionar con otros objetos y usar parámetros como el nombre, el tag o una variable de un script para causar una reacción. Al igual que el mesh filter, hay algunos por defecto con formas específicas, pero se puede añadir una forma concreta o usar una versión automática que sigue la forma del mesh filter. Sin embargo, fuera de las formas básicas, las otras opciones requieren de muchos recursos para funcionar, así pues, se suelen usar las formas básicas.



Figura . Componentes GO: Material

1. Material: el material determina el color y la superficie del GameObject según determinan los detalles y el componente material. Los materiales son otro tipo de elemento de Unity diferente del GameObject y determinan la textura, colores y patrones de las superficies, se hablará más en detalle de ellos más adelante.

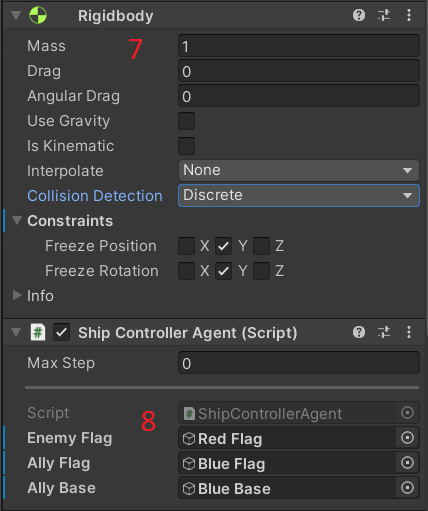


Figura . Componentes GO: Rigidbody

1. Rigidbody: el rigidbody es el elemento que se encarga de las físicas. Contiene la masa, parámetros de gravedad, rozamiento… puede limitar los ejes en los que un elemento puede moverse y girar y tiene una versión especial de collider incluida en el.

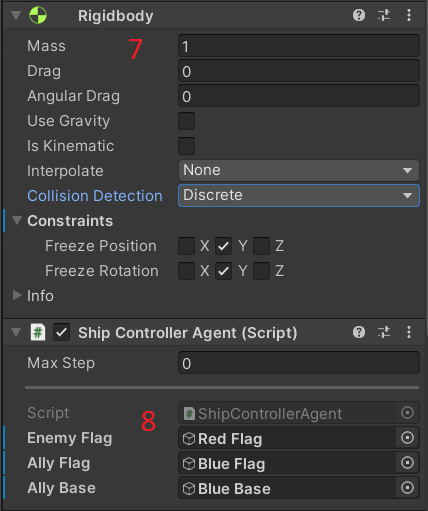


Figura . Componentes GO: Script

1. Scripts: los scripts son el código que hay por debajo de los componentes de Unity y determina cómo se relacionan entre ellos. Están escritos en C# y pueden tener variables públicas que luego aparecen en Unity, pudiendo darles valores en tiempo real en la propia interfaz. También se pueden arrastrar componentes, de este GameObject o de otros directamente desde la interfaz, evitando complejas búsquedas de componentes u objetos mediante código. Más detalles de las características particulares del código para Unity serán descritas en el apartado de Visual Studio.

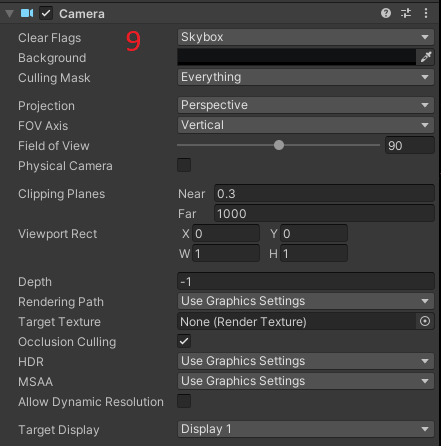


Figura . Componentes GO: Cámara

1. Cámara: La cámara es el componente que representa el punto de vista desde el que el jugador puede ver. Como desarrollador se puede usar para ver cómo se ve el juego “desde dentro”. Puede configurarse con una gran diversidad de parámetros para conseguir diferentes efectos.

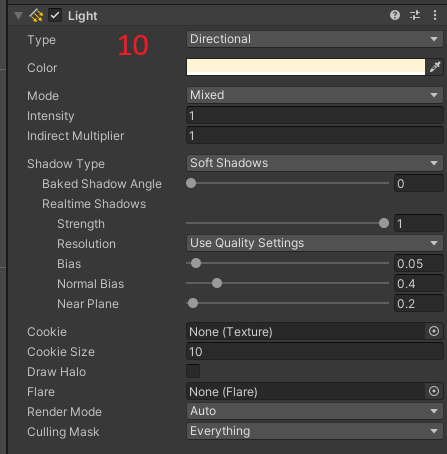


Figura . Componentes GO: Iluminación

1. Iluminación: el componente de iluminación genera un foco de luz que ilumina la escena. Esta fuente de luz puede estar configurada de distintas maneras para conseguir distintos efectos.

La mayoría de los parámetros de estos componentes se dejarán en sus valores estándar dado el escaso conocimiento sobre que hacen la mayoría de ellos y solo hacer que todo se vea peor tras intentar editarlos. No influyen a la hora de crear el entorno para los agentes y por tanto tampoco son relevantes al resultado del experimento.

#### Materiales

Los materiales son los componentes que contienen las cualidades de las superficies. Pueden ser añadidos a GameObjects para darles color, rugosidad y otros efectos.

# RESULTADOS

TEXT.

# CONCLUSIONES

TEXT.

# LINEAS FUTURAS

TEXT.

# BIBLIOGRAFIA

Haugeland, J. (1985). *Artificial Intelligence: The Very Idea.*

Kurzweil, R. (2005). *The Singularity is Near.*

Maloof, M. (2018). *Artificial Intelligence: An Introduction.*

McCorduck, P. (2004). *Machines Who Think.*

McKinsey & Company. (13 de Abril de 2018). *McKinsey Analitics.* Obtenido de https://www.mckinsey.com/business-functions/mckinsey-analytics/our-insights/ask-the-ai-experts-whats-driving-todays-progress-in-ai

Minsky, M. (1967). *Computation: Finite and Infinite Machines.*

Nilsson, N. (1998). *Artificial Intelligence: A New Synthesis.*

Russell, S. J., & Norvig, P. (2009). *Artificial Intelligence: A Modern Approach.*