

GRETIVE: Un Bot Evolutivo para HearthStone basado en Perfiles

Alejandro Romero García, Antonio M. Mora García^[0000-0003-1603-9105]

Depto. Teoría de la Señal, Telemática y Comunicaciones
Universidad de Granada, España
algebro96@correo.ugr.es, amorag@ugr.es

Resumen. In recent years, the international AI competition at Hearthstone has exponentially increased its fame among the scientific community, which has actively participated with multiple players. One of the best, EVA, applied a greedy approach combined with an evolutionary algorithm (EA). However, almost all proposals (including EVA) were designed to work in a generalist way, that is, for any of the possible heroes of the game. This generalization has a great shortcoming, since the exclusive cards of each hero, as well as their possible different behavior profiles, are not exploited. This article follows a similar philosophy to EVA, also hybrid (Greedy + AE), but taking into account three archetypes or profiles very extended among the community of players: Aggro (offensive), Control (defensive) and Midrange (intermediate). So, in essence, three different behaviors have been optimized with the aim of creating a more specialized agent capable of using a different behavior engine depending on the hero he plays with. To demonstrate the value of this approach, several experiments have been carried out, comparing the evolved agents with EVA in many different games, using three different heroes. The results show an improvement on EVA for all three profile-based agents, and a great overall performance against other less competitive approaches.

Palabras clave: Digital Collectible Card Games, Juego Digital de Cartas Coleccionables, Hearthstone, Hearthstone AI Competition, SabberStone, Inteligencia Artificial, Algoritmos Evolutivos, Optimización basada en perfiles

1 Introducción

Hearthstone es un famoso y ampliamente extendido juego digital de cartas coleccionables (DCCG), que en 2018 llegó a 100 millones de jugadores enfrentándose en partidas *on-line*. Aún sigue estando muy activo, por lo que se ha convertido en un entorno muy interesante y desafiante para el diseño y desarrollo de agentes autónomos (*bots*) capaces de jugar de forma muy competitiva. Éstos han sido enviados como participantes de la competición internacional de IA en Heartstone [1], la cual ha traído

Este trabajo ha sido financiado en parte por los proyectos B-TIC-402-UGR18 (FEDER y Junta de Andalucía) y RTI2018-102002-A-I00 (Ministerio Español de Ciencia, Innovación y Universidades), así como los proyectos TIN2017-85727-C4-1-2-P (Ministerio Español de Economía y Competitividad) y TEC2015-68752 (también financiado con fondos FEDER)

de la mano un simulador, SabberStone [2], utilizado para desarrollar y probar los bots implementados.

Dichos agentes participantes aplicaron diversas técnicas, sin embargo, los enfoques más exitosos se centraron en dos técnicas: por una parte, un porcentaje significativo de competidores implementó alguna variante o mejora de algoritmo *Greedy*, como técnicas adaptadas de *Backtracking* o *Lookahead*. Por otro lado, los agentes que aplicaron técnicas basadas en *Monte Carlo Tree Search* (MCTS) [3] o variaciones de ella, también fueron muy numerosos y muchos de ellos terminaron en las primeras posiciones de la clasificación.

Mirando a los agentes participantes en la competición, la gran mayoría aplican técnicas *greedy* en el *track 'User-created deck'* (mazo definido por el jugador/agente antes de la partida), mientras que ambas técnicas se aplican en la misma medida en el *track 'Premade deck'* (mazos comunes para todos los jugadores/agentes).

Este artículo se centra en el diseño e implementación de un agente para el segundo de estos *tracks*, dado que la propia definición de mazos es un problema complejo por sí mismo [4]. Hemos elegido un enfoque *greedy* debido a su simplicidad y eficacia. Esta implementación se ha mejorado mediante un Algoritmo Evolutivo (AE) [5], que se ha aplicado para optimizar los parámetros (umbrales y pesos) considerados en las reglas de decisión que guían el comportamiento del bot. Este tipo de enfoque ya se usó en participantes anteriores, como EVA [6], que finalizó en segundo lugar en las competiciones de 2018 y 2019. Sin embargo, EVA tiene un comportamiento generalista, es decir, un conjunto común de reglas de decisión para cualquiera de los 9 héroes, los cuales, sin embargo, cuentan con diferentes conjuntos de cartas para usar en el juego, por lo que su efectividad se verá mermada.

De modo que el agente “GrEvolutivo” (*greedy* + evolutivo) propuesto aquí, ha sido diseñado para tener en cuenta diferentes comportamientos de acuerdo con tres perfiles principales (o ‘arquetipos’ en el lenguaje del juego):

- **Aggro:** un perfil claramente ofensivo en el que el objetivo principal es reducir la salud del héroe rival con cada carta que se juega. El objetivo es ganar la partida lo antes posible.
- **Control:** un perfil defensivo, en el que el jugador intenta evitar daños en los primeros turnos para llegar vivo a los últimos turnos y luego usar cartas de alta potencia para terminar el combate.
- **Midrange:** un perfil intermedio entre los dos anteriores. El objetivo es controlar desde el principio y ser agresivo una vez superados la mitad de los turnos del juego. Utiliza cartas con una buena relación entre ataque, defensa y coste.

Es importante tener en cuenta que estos arquetipos se refieren a los mazos (el conjunto de cartas) que un jugador puede componer de acuerdo con el héroe seleccionado. Por lo tanto, un mismo héroe podría jugar siguiendo uno de estos perfiles u otro. Si bien las cartas únicas asociadas con el héroe determinan principalmente uno de estos tres comportamientos.

De este modo, el enfoque propuesto aplicará un AE para evolucionar (optimizar) los pesos de las reglas de decisión consideradas por el algoritmo *greedy*, componiendo un conjunto diferente de valores por perfil, ya que el objetivo es

conseguir comportamientos especializados adaptados para gestionar mazos específicos, dependiendo del héroe que controle el bot.

El agente creado ha sido probado en varios experimentos realizados siguiendo las reglas de la competición, dentro del mismo simulador utilizado en ese campeonato, llamado SabberStone [2]. Éste se ha comparado con EVA, como un algoritmo de referencia que obtuvo muy buenos resultados en las últimas ediciones de la competición.

2 Hearthstone, Competición de AI, Algoritmos Evolutivos

Hearthstone: heroes of Warcraft es un juego digital de cartas coleccionables multijugador (DCCG), desarrollado por la compañía Blizzard Entertainment. Es un juego de cartas por turnos que enfrenta a dos oponentes, que pueden elegir un héroe entre 9 clases diferentes y definir mazos específicos de 30 cartas antes de pelear (algunas de esas cartas son comunes a todos los héroes y otras son exclusivas de uno de ellos). Las cartas salen del mazo en el juego de forma aleatoria, y los jugadores usan sus limitados cristales de maná (que se incrementan en cada turno) para lanzar hechizos, equipar armas o invocar esbirros para atacar a su rival, con el objetivo de reducir la salud del oponente a cero para derrotarlo.

La Figura 1 muestra una captura de una partida, donde podemos ver a los héroes luchando, sus esbirros correspondientes, las cartas restantes y una carta que se está jugando. Debe tenerse en cuenta que cada carta tiene dos valores asociados: ataque en color rojo y defensa en color amarillo, y también cada uno de ellos tiene un costo asociado en número de cristales de maná (número en azul) cuando está en la mano. Esto significa que un héroe debe poseer al menos este número de cristales para jugar esa carta.



Fig. 1. Imagen de una partida de Hearthstone.. El héroe del jugador está en la parte inferior (cartas visibles) y el enemigo en la parte superior (cartas ocultas). Los esbirros en la parte superior del tablero pertenecen al enemigo y los de la parte inferior a nuestro jugador. Los puntos de maná se representan como cristales azules en la parte inferior.

El juego es un excelente banco de pruebas para la implementación de agentes autónomos. Por esta razón, hace dos años se lanzó la competición de IA para Hearthstone [1]. Se trata de un campeonato internacional que trata de encontrar la mejor implementación de IA. Hay dos tracks diferentes: "Mazos predefinidos" (se deben usar mazos específicos de *Aggro*, *Control* y *Midrange*) y "Mazos definidos por el usuario" (el agente puede definir su propio conjunto de cartas para vencer a los oponentes). Nos centraremos aquí en el primer *track*, que se basa en un torneo todos contra todos que enfrenta a los agentes uno por uno en 100 partidas diferentes utilizando un conjunto de nueve mazos distintos. Seis de estos mazos ya son conocidos, al ser mazos bastante famosos basados en arquetipos.

La competición se ejecuta utilizando un simulador de Hearthstone llamado SabberStone [2], que resulta el más preciso sobre el juego, ya que implementa los efectos de las cartas descritos en el Libro de reglas avanzado de Hearthstone [7].

Los Algoritmos Evolutivos (AEs) son métodos inspirados en la evolución natural de las especies [1], de modo que optimizan poblaciones de individuos que modelan (codifican) soluciones potenciales para un problema como un conjunto de valores (genes). Un AE normalmente sigue un proceso de selección de los mejores individuos, recombinación de su material genético (sus valores genéticos) y la mutación de algunos de ellos, con el objetivo de crear una descendencia que, combinada con los padres anteriores, definirá una nueva población de mejores individuos en promedio. Este proceso se repite varias generaciones (iteraciones) hasta que se cumple un criterio de parada. Para realizar la selección y el reemplazo, es obligatorio definir una función de evaluación o fitness que debe asignar un valor a cada individuo representando su calidad como solución al problema.

Los AEs se han aplicado ampliamente para resolver una gran cantidad de problemas de optimización, incluida la mejora de motores de comportamiento (Inteligencia Artificial), como también propone el presente trabajo.

3 Estado del arte

Los juegos de cartas coleccionables digitales, y en concreto Hearthstone se han convertido en uno de los principales ámbitos de investigación en los últimos años, debido a su éxito, junto con el surgimiento de simuladores y 'frameworks' abiertos, como Metastone [8] o SabberStone [2], que han simplificado la implementación y prueba de agentes autónomos para jugar al juego, o la creación y validación automática de mazos.

Esta última línea ha sido estudiada dentro de Hearthstone en trabajos como [4], en el que se definieron diferentes mazos mediante un AE, tratando de encontrar la mejor combinación entre 700 cartas para diferentes héroes. Bhatt et al. en [9] realizaron un estudio similar sobre la creación de mazos mediante Estrategias Evolutivas, pero se centraron en el análisis del juego desde el punto de vista del equilibrio del jugador, comprobando si es posible construir mazos imbatibles para el resto de jugadores.

Sin embargo, la principal línea de investigación en este ámbito es claramente el diseño y la aplicación de agentes autónomos para jugar al juego, impulsado en gran

medida por la aparición de la mencionada Competición de IA de Hearthstone. En torno a 30 agentes se han presentado en cada edición y algunos de ellos han derivado en publicaciones. El presente trabajo también se centra en la generación de un agente para dicha competición.

Hay varios artículos diferentes relacionados con agentes de Hearthstone. Por ejemplo, Bursztein en [10] describió un enfoque muy competitivo basado en análisis estadísticos o Grad [11], que presentó un agente utilizando Redes Neuronales. Sin embargo, la mayoría de las propuestas son variaciones de Monte Carlo Tree Search (MCTS). Santos et al. [12] propusieron un enfoque estándar y Swiechowski et al. [13] mejoraron el MCTS para manejar información imperfecta en este juego, por citar dos.

Siguiendo una técnica más simple, García-Sánchez et al. propusieron en [6] un agente ‘greedy’, bautizado como EVA. Éste considera 21 atributos o factores diferentes extraídos del juego, que se basan en las diferencias entre los estados durante una partida. Se definió una función de evaluación que calcula las diferencias entre el estado actual del juego, y cada posible estado resultante de la elección de cada posible acción a la vez. Para ello, utilizan los 21 valores. El algoritmo es simple, pero la función de evaluación fue definida por jugadores expertos y combina muchos factores. Además, hay un conjunto de pesos asociados a estos factores que fueron optimizados por medio de un AE. Este agente terminó en segunda posición en las competiciones de 2018 y 2019.

En este estudio, hemos tomado EVA como referencia, porque se trata esencialmente de un enfoque ‘greedy’ como el bot propuesto aquí, basado también en una función de evaluación que considera muchos factores del estado del juego, y que alcanzó muy buena posición en las competiciones pasadas. Sin embargo, hemos definido un nuevo agente ‘greedy’ que considera un conjunto de variables diferente para el proceso de toma de decisiones. Así pues, nuestro objetivo en este trabajo es partir de esta "simple" propuesta y analizar la influencia de la incorporación de arquetipos en el proceso de optimización, definiendo mecanismos específicos basados en perfiles. Sostenemos que este diseño llevará a crear agentes más competitivos que los diseñados para lograr un comportamiento general (como EVA), es decir, sin tener en cuenta la clase de héroe con el que se juega, ni el arquetipo de los mazos. El agente propuesto se describe en la siguiente sección.

4 **GRETIVE: Nuestro agente ‘grevolutivo’**

4.1 **Descripción general del agente.**

El bot de Hearthstone propuesto, llamado GRETIVE, está compuesto por un conjunto de reglas de decisión, siguiendo una estrategia ‘greedy’ con el fin de decidir la mejor jugada en cada turno. Esta decisión se basa en simular que se juega cada carta disponible (con suficiente maná) en la mano de nuestro héroe o realizar alguna acción en el tablero (como un poder de héroe, por ejemplo). Así, la simulación, ejecutada en SabberStone, ofrece como salida el estado del tablero una vez que la carta fuese utilizada, es decir, los esbirros finales del héroe y del enemigo, así como su estado (tanto para los héroes como para dichos esbirros). A partir de esta información, se extraerán del tablero diversas variables relativas a la salud restante, los posibles

puntos de ataque, los hechizos, la provocación, etc. Estos factores se multiplican por un conjunto de pesos, que asignan una importancia relativa a cada uno. Después todos se suman para obtener un "valor de preferencia" para esa decisión.

Una vez evaluadas todas las jugadas posibles, GRETIVE ejecutará la que tenga el mejor valor de preferencia. La fórmula utilizada para calcular ese valor se muestra en la Ecuación 1, que combina factores como el poder de ataque de las fuerzas del héroe y las enemigas, la provocación de sus esbirros (muy importante en este juego para atraer un ataque y proteger al héroe), así como su salud total. Esta función se ha definido basándose en el conocimiento de varios jugadores expertos en el juego.

$$\begin{aligned}
 \text{Valor de preferencia} = & w1 \cdot \text{Ataque total de los propios esbirros} + \\
 & w2 \cdot \text{Ataque total de los esbirros enemigos} + \\
 & w3 \cdot \text{Salud total de los propios esbirros} + \\
 & w4 \cdot \text{Salud total de los esbirros enemigos} + \\
 & w5 \cdot \text{Salud total de los propios esbirros con Provocar} + \\
 & w6 \cdot \text{Salud total de los esbirros enemigos con Provocar} + \\
 & w7 \cdot \text{Armadura de Héroe propio} + \\
 & w8 \cdot \text{Armadura de Héroe enemigo} + \\
 & w9 \cdot \text{Salud del héroe propio} + \\
 & w10 \cdot \text{Salud del héroe enemigo}
 \end{aligned} \tag{1}$$

Es importante señalar que los pesos pueden ser positivos, negativos o incluso cero (eliminando así el factor correspondiente).

Como puede deducirse, las decisiones dependen en gran medida del conjunto de pesos, cuyo valor debe fijarse cuidadosamente. La mayoría de los participantes en la competición se basaron en un conjunto fijo de pesos, buscando un buen comportamiento general independiente del héroe y del mazo en sí. Sin embargo, como ya se ha dicho, GRETIVE optimizará estos pesos mediante un Algoritmo Evolutivo basado en perfiles. Así, el enfoque propuesto consiste en optimizar el conjunto de pesos en busca de una mejora en la estrategia de juego del agente, asumiendo un estilo de juego específico, basado en cualquiera de los arquetipos de HS mencionados anteriormente. Mediante la función de fitness se busca no sólo promover la victoria, sino también que el agente persiga una técnica específica para lograr un resultado de juego determinado.

4.2 AE basado en perfiles.

Hemos implementado un Algoritmo Genético (AG) con un enfoque generacional [14]. Cada **individuo** se representa como un vector de diez valores reales, que representan los diez pesos $[w1, \dots, w10]$ utilizados en la Ecuación 1. Cada uno de estos pesos, corresponde a una característica cuantificable del juego. Permiten valorar y comparar diferentes estados de una partida, y se utilizan a la hora de determinar la siguiente acción a realizar por el agente, en base al resultado obtenido de la valoración de los estados derivados de cada acción candidata.

Se ha aplicado una **Estrategia de Selección elitista**, en la que se eliminan los individuos que tienen el peor fitness, para dar cabida a nuevos individuos como

resultado del cruce de los que tienen mejores valores en la evaluación. El esquema del proceso de selección y reemplazo se muestra en la Figura 2.

Por cada cinco individuos al azar, dos son eliminados de la población (los que tienen menor puntuación), dos son seleccionados para la reproducción (los mejores), y sus dos descendientes reemplazarán a los dos eliminados. El individuo restante sobrevivirá y pasará a la siguiente generación.

ELIMINADO	ELIMINADO	INDIFERENTE	REPRODUCIDO	REPRODUCIDO
< - peor valoración fitness		< -	- >	mejor valoración fitness - >

Fig. 2. La estrategia de selección del AE implementado. % de la población son eliminados y posteriormente reemplazados por los nuevos individuos generados por el cruce de las mejores partes de %, que son elegidos como padres.

Se ha aplicado un **operador estándar de cruce de un solo punto**, por lo que se elegirá una posición intermedia en los cromosomas, y la primera parte de un individuo (vector) se unirá con la segunda parte del otro, y luego se unirán las dos partes restantes. Se ha implementado una **mutación no uniforme** para la variación de los individuos. Así, se elige un pequeño porcentaje de la población y se modifica significativamente al menos un gen (peso) de esos individuos.

Con el objetivo de optimizar el perfil, se han definido **tres funciones de fitness** diferentes, que se describen en la siguiente sección.

4.3 Funciones de evaluación basadas en perfiles.

La **evaluación** de un individuo se realiza a través de una sucesión de partidas simuladas en las que el individuo aplica sus pesos actuales a las diferentes selecciones requeridas para las acciones en el juego. Para ello se aplica una **función de evaluación basada en perfiles**, es decir, el valor de fitness/aptitud se calculará teniendo en cuenta diferentes factores, dependiendo del comportamiento del objetivo o la estrategia que queremos optimizar para el agente. Así, medirá de forma cuantitativa lo bien que ha jugado el individuo y cuánto se ha adaptado al comportamiento esperado. Los factores considerados han sido elegidos por un jugador experto y los pesos utilizados se han obtenido tras un exhaustivo procedimiento de experimentación.

Es importante señalar que para hacer frente a la incertidumbre o al ruido presente en la optimización de los juegos no deterministas [15], como es el caso de Hearthstone, cada individuo será reevaluado en la siguiente generación. Además, la evaluación consistirá en la realización de 40 partidas diferentes, con el objetivo de valorar con la mayor precisión posible la calidad de cada individuo. El rival en todas esas partidas será un agente “Aggro Greedy” definido por uno de los autores para una clase Guerrero, y el individuo utilizará la clase de héroe que se está optimizando, es decir, la correspondiente al arquetipo.

Antes de presentar las funciones de fitness, la Tabla 1 muestra los términos/variables utilizados en todas ellas, para mayor claridad. Se refieren al estado del juego/tablero al final de cada partida.

La **función de fitness del perfil Aggro** es la siguiente:

Si el individuo vence:

$$F_{AG} = 180 - 7 \cdot GT + PH + PA + 2 \cdot SPMA + 3 \cdot (PB - EB) \quad (2)$$

Si el individuo pierde:

$$F_{AG} = -130 + 5 \cdot GT - 10 \cdot (EH + EA) - 3 \cdot SEMA + 5 \cdot SPMA - EB \quad (3)$$

Estas ecuaciones tienen como objetivo promover un "comportamiento agresivo" para el agente, es decir, tratar de hacer daño al oponente rápidamente (en pocos turnos), por lo que GT tiene gran relevancia, con un gran número de esbirros (PB), o simplemente centrándose en el daño al oponente, teniendo en cuenta el poder de ataque de los esbirros del jugador (SPMA), y los del enemigo cuando se pierde.

Tabla 1. Valores considerados para evaluar al agente (individuo), 'P' se refiere al Jugador y 'E' al enemigo. Los perfiles son 'A' para Aggro, 'C' para Control, y 'M' para Midrange.

ACRÓNIMO	DESCRIPCIÓN	PERFIL	ACRÓNIMO	DESCRIPCIÓN	PERFIL
GT	Cantidad de turnos de juego.	A-C-M	[P/E]B	Cantidad de esbirros que el jugador posee.	A
[P/E]H	Cantidad de salud que ese jugador tiene.	A-C-M	S[P/E]MH	Cantidad de puntos de salud de todos los esbirros que el jugador posee.	C
[P/E]A	Cantidad de armadura que ese jugador tiene.	A-C-M	S[P/E]MTH	Cantidad de puntos de salud de todos los esbirros con la habilidad de Provocar que el jugador posee.	C
S[P/E]MA	Cantidad de puntos de ataque de todos los esbirros que el jugador posee	A-C-M	S[P/E]MTC	Cantidad de maná del coste de todos los esbirros con la habilidad de Provocar que el jugador posee.	C

En el caso del **perfil de Control**, la función de fitness se define como:

Si el individuo gana:

$$F_{CT} = 150 + GT + 2 \cdot (PH + PA) + 2 \cdot SPMA + 2 \cdot (SPMH + SPMTH) - 4 \cdot SEMA \quad (4)$$

Si el individuo pierde:

$$F_{CT} = -200 + 10 \cdot GT - 4 \cdot (EH + EA) + 5 \cdot SPMA + 2 \cdot (PB + SPMTC + SPMTH) - 6 \cdot SEMA \quad (5)$$

Esta función busca una victoria tardía, se centra en la eliminación de los esbirros del enemigo evitando el daño (PH), el control del tablero y jugar cartas poderosas en rondas posteriores del juego. Por lo tanto, la habilidad de *Provocar* en los esbirros es muy relevante (SPMTH, SPMTC), porque obliga al enemigo a atacarlos primero, además, la salud restante de los esbirros (SPMH) también es importante para defender mejor a nuestro héroe. Si el agente pierde, se consideran los valores del enemigo.

Por último, la función de fitness para el **perfil de Midrange** se define como:

Si el individuo gana:

$$F_{MR} = 180 - 3 \cdot GT + 3 \cdot (PH + PA) + 2 \cdot SPMA \quad (6)$$

Si el individuo pierde:

$$F_{MR} = -120 + 5 \cdot GT - 5 \cdot (EH + EA) - 5 \cdot SEMA + 3 \cdot SPMA \quad (7)$$

Esta función de evaluación está entre los comportamientos de los perfiles anteriores. Busca la victoria en una etapa intermedia del juego, controlando el tablero completamente al principio y pasando de unas cartas muy eficientes (en términos de coste/poder) a una actitud totalmente agresiva. La salud y la armadura del héroe (PH, PA) es más importante aquí, así como el ataque de los esbirros (SPMA). Como antes, cuando el agente pierde, las unidades y el estatus del enemigo se vuelven muy relevantes.

5 Experimentos y resultados

5.1 Configuración experimental

La primera parte de los experimentos se centra en el estudio del proceso de optimización (es decir, el rendimiento de AE). Para ello se han considerado 30 individuos, 30 generaciones, 40 partidas en las evaluaciones, probabilidad de cruce de 0,4 y probabilidad de mutación de 0,1. En cada evaluación se consideran los mazos

predefinidos para el héroe, ya que la clase de héroe conducirá a un perfil o estrategia predominante utilizando sus cartas exclusivas (por clase).

5.2 Optimización basada en perfiles

En primer lugar, llevamos a cabo varias ejecuciones de optimización, considerando los arquetipos más relacionados con la clase del héroe (según el Hearthstone Advanced Rulebook [7]). Los perfiles optimizados han sido: Cazador (Aggro), Druida (Midrange) y Chamán (Control). Analizaremos, por tanto, el rendimiento del AE en diez ejecuciones para cada uno de los tres pares héroe-perfil.

Las Figuras 3, 4 y 5 muestran los resultados de la optimización de AE en promedio de las diez ejecuciones. La Figura 3 muestra el rendimiento del AE al optimizar los pesos de agentes de perfil Aggro.

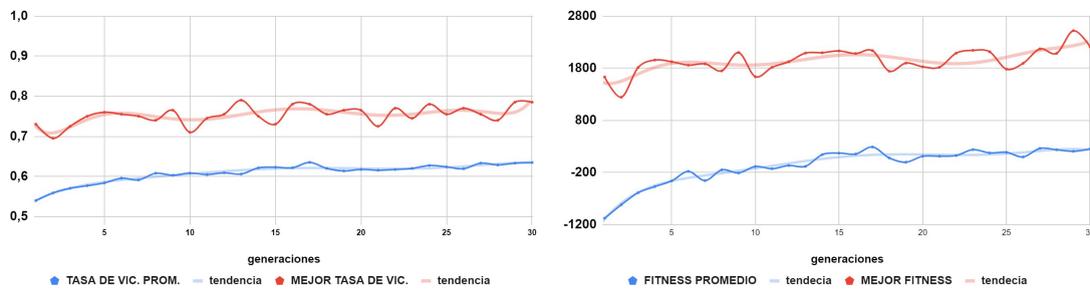


Fig. 3. Resultados del perfil de 'Aggro'. Evolución de la tasa de victorias (izquierda) y del fitness (derecha) en un promedio de diez ejecuciones diferentes. Los mejores valores o máximo y los valores medios están representados en los gráficos.

Como se puede ver, tanto la tasa de fitness como la de victorias siguen una distribución creciente, que se puede ver más claramente al observar las líneas de tendencia. Esta es la tendencia deseable para un algoritmo de optimización, es decir, la población va mejorando en cada generación. Además, incluso si el valor de referencia para la optimización es el fitness, los gráficos de la tasa de ganancia muestran un comportamiento bastante similar, por lo que las funciones de evaluación parecen estar correctamente definidas, ya que los valores altos de fitness corresponden a tasas de victorias más altas, que es, a su vez, el principal objetivo de cualquier agente (es decir, "ganar tanto como sea posible").

Las Figuras 4 y 5 muestran resultados similares a los anteriores, con una clara tendencia creciente y gráficos congruentes de fitness y tasa de victorias. Sin embargo, es notable la gran similitud de los gráficos de la Figura 5, tanto el mejor fitness como la mejor tasa de ganancia, así como el fitness medio y la tasa media de ganancia, siguen distribuciones casi idénticas, lo que puede entenderse como una definición muy correcta de la función de evaluación para el perfil Midrange con respecto a la correspondiente tasa de victorias de los agentes generados.

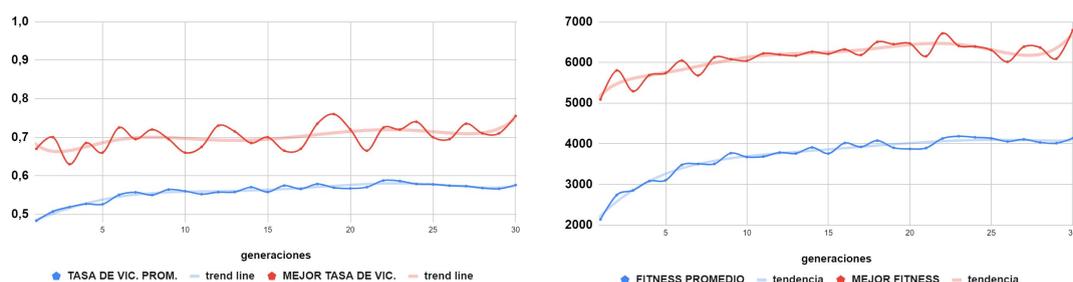


Fig. 4. Resultados del perfil de 'Control'. Evolución de la tasa de victorias (izquierda) y del fitness (derecha) en un promedio de diez ejecuciones diferentes. Los mejores valores o máximos y los valores medios están representados en los gráficos.

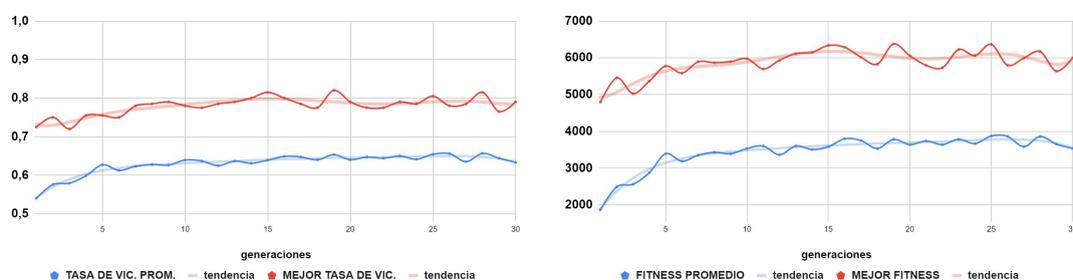


Fig. 5. Resultados del perfil de 'Midrange'. Evolución de la tasa de victorias (izquierda) y del fitness (derecha) en un promedio de diez ejecuciones diferentes. Los mejores valores o máximos y los valores medios están representados en los gráficos.

Tabla 3. Estadísticas de la media de diez ejecuciones para cada perfil optimizado (fit \leftrightarrow fitness, t.v. \leftrightarrow tasa de victorias).

Perfil	RESULTADOS						
	param. estadístico.	Max fit.	Min fit.	Media fit.	Max. t.v.	Min. t.v.	Media t.v.
AGGRO	Media	1943,2	-2361,1	-59,3	0,75	0,45	0,61
	Desviación	234,1	581,3	334,8	0,02	0,05	0,02
CONTROL	Media	6139,9	1126,5	3712,6	0,70	0,40	0,55
	Desviación	370,8	699,7	488,4	0,03	0,03	0,02
MIDRANGE	Media	5871,7	802,9	3448,2	0,78	0,46	0,63
	Desviación	365,5	741,2	451,3	0,02	0,04	0,02

En cuanto al rendimiento real de los agentes durante la evolución (en términos de tasas de victorias) siempre se sitúa en torno al 60%, porque el rival considerado en las

evaluaciones es bastante duro y agresivo. Además, es muy relevante el orden de los turnos de juego, como se mostrará en el próximo experimento.

Los resultados numéricos se presentan en la Tabla 3. Como se puede observar en la tabla, la estrategia Aggro es la más arriesgada, ya que muestra la mayor variación entre los individuos buenos (alto valor de fitness) y los malos, teniendo una media de fitness negativo, aunque las tasas de victoria alcanzadas son muy buenas. El MidRange es el perfil que alcanza las mejores tasas de victorias. Además, debemos recordar que los valores de fitness no son comparables entre los perfiles, ya que se obtienen mediante funciones diferentes.

La Tabla 4 muestra los pesos obtenidos tras la evolución para los tres perfiles optimizados. Como se puede ver las diferencias entre ellos son bastante acusadas en la mayoría de los casos, lo que indica que el comportamiento que éstos determinan será muy distinto, como así se demuestra durante las partidas y se ve en los resultados.

Tabla 4. Pesos resultantes aplicados a los diferentes perfiles obtenidos. Se señala con escala de color la magnitud de los pesos.

Perfil	Ataque Esbirros propios	Ataque Esbirros enemigos	Salud Esbirros Propios	Salud Esbirros enemigos	Salud Esbirros propios Provocar	Salud Esbirros enemigos Provocar	Armadura héroe propio	Armadura héroe enemigo	Salud héroe propio	Salud héroe enemigo
AGGRO	34,09	-48,19	26,89	-33,90	11,48	-44,58	30,49	-46,80	19,93	-42,86
CONTROL	38,80	-92,03	23,85	-22,77	13,82	-52,33	43,58	-45,76	14,22	-27,63
MIDRANGE	32,65	-47,93	35,85	-48,95	21,18	-29,31	52,53	-18,80	27,24	-39,47

5.3 Rendimiento del agente

En este último experimento, los agentes obtenidos (uno por perfil) serán probados contra un rival duro: EVA, para demostrar el valor de la propuesta.

En primer lugar, para cada perfil, se ha seleccionado el mejor individuo por ejecución como el de mayor valor de fitness y tasa de victorias en toda la ejecución y después se ha elegido un campeón del perfil entre esos 10 mejores enfrentando a cada uno contra el Guerrero ‘Greedy Aggro’, utilizado durante la evolución, en 200 combates, siendo cada uno de ellos jugador 1 y jugador 2 en 100 combates. El que haya obtenido mayor índice de victorias será el campeón. Por lo tanto, después de este proceso, hay un Campeón de ‘Aggro’, un Campeón de ‘Control’ y un Campeón de ‘Midrange’, llamados GRETIVE-AG, GRETIVE-CT y GRETIVE-MR.

Cada campeón se ha enfrentado al agente EVA usando la misma clase de héroe, es decir, Cazador para las partidas GRETIVE-AG vs EVA, Chamán para GRETIVE-CT vs EVA, y Druida para GRETIVE-MR vs EVA; y los mazos generados automáticamente para ambos agentes. Se han realizado 2000 enfrentamientos en cada

caso, jugando 1000 como el jugador 1 nuestro agente y 1000 como el jugador 1 EVA. Los resultados obtenidos se presentan en la Tabla 5.

En vista de los resultados, podemos notar el alto impacto que tiene el orden de turno (quién juega primero) en los enfrentamientos, siendo mucho más relevante cuando la partida se enfrenta a rivales duros, como EVA. Ya que ser el jugador 1 significa jugar durante el primer turno, lo que es bastante relevante para el desarrollo del enfrentamiento.

Tabla 5. Porcentajes de victoria obtenidos por cada agente basado en perfil enfrentado con el agente EVA, siendo jugador 1 y jugador 2, y la tasa de victoria total por cada agente.

Resultados de los enfrentamientos con EVA			
<i>Agente</i>	<i>Victorias como jugador 1</i>	<i>Victorias como jugador 2</i>	<i>Tasa de victoria total</i>
GRETIVE-AG	64,9%	36,8%	50,85%
GRETIVE-MR	65,3%	39,8%	52,55%
GRETIVE-CT	65,8%	41,9%	53,85%

Se puede ver en la columna de tasa de ganancia que nuestros agentes son capaces de ganar una mayor cantidad de partidas que el agente EVA. Además de que GRETIVE-CT es el que tiene el mejor rendimiento. Es cierto que las diferencias con EVA no son altas, pero debemos tener en cuenta que dicho agente acabó en segunda posición en las dos últimas Competiciones de IA de Hearthstone, entre 30 participantes cada vez. Lo que nos lleva a pensar que nuestros agentes también podrían alcanzar una buena posición en el ranking. Además, hemos encontrado un mazo predefinido (de la competición) con un perfil adecuado, el mazo Midrange de Druida, es decir, un mazo para un héroe ideal para ese perfil. Se han realizado 2.000 partidas enfrentando a nuestro agente contra EVA, utilizando tanto este héroe como el mazo, y siendo el jugador 1 y el jugador 2 1.000 veces cada uno. Los resultados para este Druida GRETIVE-MR han sido: 70,6% de victorias como jugador 1, 55,9% de victorias como jugador 2, y una tasa de victoria total del 63,25%. Mirando los resultados se puede observar una clara mejora con respecto a los resultados presentados en la Tabla 5. Esto demuestra el valor del enfoque de optimización propuesto, que aumentará el rendimiento del agente cuando se utilice el mazo arquetipo apropiado.

6 Conclusiones y trabajo futuro

En este trabajo se ha presentado una propuesta para la creación de agentes autónomos que sean jugadores competitivos en el juego Hearthstone. Para ello, se ha partido de un motor de comportamiento 'greedy' capaz de decidir la mejor jugada en cada momento, y se ha aplicado un Algoritmo Evolutivo (AE) para optimizar los pesos de los factores de juego considerados para tomar esta decisión (como la salud de nuestro propio héroe, o el ataque de nuestros esbirros en el tablero, por ejemplo).

Este AE se basa en los llamados perfiles, que son arquetipos o estrategias frecuentemente seguidos por muchos jugadores en el juego. Cada uno de estos perfiles implica normalmente no sólo una forma de juego, sino también el uso de tipos específicos de cartas, que también dependen del héroe que el jugador está controlando. Así pues, hay tres funciones de evaluación diferentes en el AE, relacionadas con los perfiles Aggro (agresivo), Control (conservador) y Midrange (medio). Los tres agentes optimizados, uno por perfil, han sido probados en 2000 enfrentamientos contra un agente del estado del arte llamado EVA. El cual, es también un algoritmo Greedy diseñado por jugadores expertos que fue optimizado por medio de un AE, pero apuntando a un comportamiento general (independiente del héroe). EVA terminó en segunda posición en la Competición Internacional de Hearthstone [1] en 2018 y 2019. Los resultados de estas partidas muestran una mejora con respecto al agente EVA, obteniendo una tasa de victorias mayor que este agente en los tres casos (nuestros tres agentes, uno por perfil). La diferencia no es muy grande para los partidas que utilizan mazos por defecto, pero cuando se ha considerado un héroe y una baraja apropiada, la diferencia en las tasas de victorias crece mucho.

En un futuro próximo los probaremos definiendo mazos específicos para cada uno de los arquetipos, con el objetivo de obtener un mayor rendimiento. También pretendemos crear un agente para la competición, capaz de utilizar cualquiera de los tres perfiles en los diferentes enfrentamientos, basado en el héroe con el que jugar o basado en el enemigo. El agente podría incluso cambiar la estrategia durante el juego según el mazo o incluso las cartas en mano. Además, las funciones de evaluación utilizadas en el AE pueden mejorarse, teniendo en cuenta la información adicional de los enfrentamientos, como el cementerio (es decir, las cartas utilizadas durante la partida). En la misma línea, el estudio también podría hacerse considerando diferentes tipos de oponentes/perfiles en la etapa de evaluación del AE, ya que el uso de sólo un perfil de 'Guerrero Aggro' en todos los casos podría afectar el proceso de optimización. Por último, el algoritmo Greedy utilizado como motor de comportamiento de los agentes también podría ser mejorado, incluyendo una búsqueda/decisión en varios niveles (analizando más de una posible acción a la vez), buscando combinaciones de cartas, ya que los 'combos' son muy poderosos y efectivos en este juego.

Referencias

1. Hearthstone AI Competition, <http://www.is.ovgu.de/Research/HearthstoneAI.html>.
2. HearthSim, SabberStone - Hearthstone simulator, <https://hearthsim.info/sabberstone/>.
3. G.M.J.B. Chaslot, M.H.M. Winands, J.W.H.M. Uiterwijk, H.J. van den Herik, B. Bouzy, "Progressive Strategies for Monte-Carlo Tree Search". *New Mathem. and Natural Comput.* 4 (3): 343-359, 2008.
4. P. García-Sánchez, A.P. Tonda, G. Squillero, A.M. Mora, J.J. Merelo Guervós, "Evolutionary deckbuilding in Hearthstone", in: *IEEE Conference on Computational Intelligence and Games, CIG 2016, Greece, September 20-23, IEEE*, pp. 1-8, 2016.
5. T. Bäck, *Evolutionary algorithms in theory and practice*. Oxford University Press, 1996.
6. P. García-Sánchez, A.P. Tonda, A.J. Fernández-Leiva, C. Cotta, "Optimizing Hearthstone agents using an evolutionary algorithm". *Knowl. Based Syst.* vol. 188, 2020.

7. Hearthstone, Gamepedia - Advanced Rulebook, https://hearthstone.gamepedia.com/Advanced_rulebook.
8. Demilich1, MetaStone - A Hearthstone simulator, 2015, <https://github.com/demilich1/metastone>.
9. A. Bhatt, S. Lee, F. de Mesentier Silva, C.W. Watson, J. Togelius, A.K. Hoover, "Exploring the hearthstone deck space", in: S. Dahlskog et al. (Eds.), Proceedings of the 13th International Conference on the Foundations of Digital Games, FDG 2018, Malmö, Sweden, August 07-10, ACM, pp. 18:1–18:10, 2018.
10. E. Bursztein, "I am a legend: Hacking hearthstone using statistical learning methods", in: IEEE Conference on Computational Intelligence and Games, CIG 2016, Santorini, Greece, September 20-23, 2016, IEEE, pp. 1–8, 2016.
11. L. Grad, "Helping AI to play Hearthstone using neural networks", 2017 Federated Conference on Computer Science and Information.
12. A. Santos, P.A. Santos, F.S. Melo, "Monte Carlo Tree Search experiments in Hearthstone", in: IEEE Conference on Computational Intelligence and Games, CIG 2017, New York, NY, USA, August 22-25, 2017, IEEE, pp. 272–279, 2017.
13. M. Swiechowski, T. Tajmajer, A. Janusz, "Improving Hearthstone AI by combining MCTS and supervised learning algorithms", in: 2018 IEEE Conference on Computational Intelligence and Games, CIG 2018, Maastricht, the Netherlands, IEEE, pp. 1–8, 2018.
14. D. E. Goldberg, Genetic Algorithms in search, optimization and machine learning. Addison Wesley, 1989.
15. A.M. Mora, A. Fernández-Ares, J.J. Merelo, P. García-Sánchez, C.M. Fernandes, "Effect of noisy fitness in real-time strategy games player behaviour optimisation using evolutionary algorithms". *J. Comput. Sci. Technol.* 27(5), 1007–1023, 2012.