

Herramientas de gamificación para la enseñanza de técnicas de búsqueda heurística en entornos dinámicos

Jesús Giráldez-Cru, Pablo Mesejo, José Ángel Segura,
Juan Fernández-Olivares, Antonio González
Departamento de Ciencias de la Computación e Inteligencia Artificial
Universidad de Granada

{jgiralde, pmesejo, josesegmur, faro, a.gonzalez}@decsai.ugr.es

Resumen

La Búsqueda Heurística (BH) es uno de los campos clásicos y más estudiados de la Inteligencia Artificial, y una rama troncal de los estudios de Ingeniería Informática. BH consiste en encontrar la ruta óptima hacia un estado objetivo, dadas unas condiciones iniciales, y usando una función heurística que guía dicha búsqueda a través del espacio de posibles estados. Usualmente, este es un problema computacionalmente duro de resolver y, por tanto, de gran interés tanto para estudios académicos como industriales. En este trabajo, se describe una metodología para el aprendizaje de búsqueda heurística basada en gamificación. En concreto, esta estrategia usa el entorno de desarrollo de controladores de videojuegos GVG-AI, y ha sido aplicada en los estudios de Grado de Ingeniería Informática de la Universidad de Granada durante los cursos 2018-19, 2019-20 y 2020-21. Esta metodología, basada en el trabajo individual y el aprendizaje a través del juego, ha permitido adaptar la docencia de esta materia a la modalidad virtual sin apenas cambios en la organización docente. Además no se observan diferencias significativas en las calificaciones de ambas modalidades docentes, sugiriendo la robustez de la metodología propuesta en escenarios extraordinarios de docencia virtual.

Abstract

Heuristic Search (HS) is one of the classical and most studied fields in Artificial Intelligence, and a core subject of the degree in Computer Science. HS consists of finding the optimal path to a target state, given some initial conditions, and using a heuristic function that guides the search through the space of possible states. Usually, this is a computationally hard problem, and thus of great interest for both academical and industrial purposes. In this work, it is described a new methodology to learn heuristic search based on gamification. In particular, this strategy uses GVG-AI, a video games

controller development framework, and it has been applied in the degrees of Computer Science of the University of Granada (Spain) during the years 2018-19, 2019-20, and 2020-21. This methodology, based on the individual work and the learning through the game, has allowed to adapt the teaching of this subject to online courses with minimal changes in the teaching organization. Moreover, there is no significant differences in the grades of both teaching methods, suggesting the robustness of the proposed methodology in extraordinary scenarios of online teaching.

Palabras clave

Docencia online, gamificación, búsqueda heurística, inteligencia artificial, ingeniería informática.

1. Introducción

La Búsqueda Heurística (BH) es uno de los temas fundamentales que se abordan al realizar una introducción a la Inteligencia Artificial (IA) [18]. En concreto, el libro de referencia de Russell y Norvig [24] dedica uno de sus seis bloques (*Problem Solving*) principalmente a esta temática. A nivel de contenido curricular de los estudios de Ingeniería Informática, dentro del Computing Curricula 2013 [9], se encuentra el área de conocimiento de Sistemas Inteligentes, dentro del cual se identifican 4 unidades de conocimiento troncales: *IS1/Fundamental Issues*; *IS2/Basic Search Strategies*; *IS3/Basic Knowledge Representation and Reasoning*; y *IS4/Basic Machine Learning*. La segunda de ellas, relativa a estrategias de búsqueda, es la que presenta una mayor carga docente recomendada. Todo ello redundando en la relevancia de la disciplina dentro del amplio campo de la IA.

Las técnicas de búsqueda en IA se emplean para “buscar” soluciones a un problema en un espacio de estados. Dicho espacio generalmente se representa como un grafo dirigido, o árbol de búsqueda, en donde

cada nodo es una posible configuración del problema. Existen dos tipos de técnicas de búsqueda: técnicas no informadas y técnicas informadas. Las primeras no utilizan ningún conocimiento del problema a resolver, de ahí que se llamen técnicas de búsqueda ciega, exhaustiva o no informada. Por otro lado, las técnicas de búsqueda informada o heurística introducen cierto conocimiento genérico sobre el problema, guiando la búsqueda de forma “inteligente” hacia regiones más prometedoras. Se trata de introducir un conocimiento de alto nivel para resolver el problema en cuestión. Para ello, se emplean las llamadas heurísticas. Newell, Shaw y Simon en 1957 dieron la siguiente definición [20]: “Un proceso que puede resolver un problema dado, pero que no ofrece ninguna garantía de que lo hará, se llama una heurística para ese problema”. La forma más habitual de representar una heurística es definiendo una función que asocia a cada estado del espacio de estados un valor numérico que evalúa de algún modo lo prometedor que es ese estado para acceder a un estado objetivo. Resulta clave comprender que un método de búsqueda heurística no siempre garantiza encontrar la mejor solución o solución óptima, sino encontrar una solución buena o aceptable en unos márgenes de tiempo y memoria razonables.

En general, los métodos heurísticos son preferibles a los métodos no informados en la solución de numerosos problemas, en los que una búsqueda exhaustiva necesitaría un tiempo demasiado grande. Es el caso de resolución de problemas complejos, como ganar a un campeón del mundo en los juegos del Ajedrez (Deep-Blue) [8] o Go (AlphaGo) [26], detección de contornos en imágenes ruidosas [16], testeo de sistemas electrónicos y electromecánicos [21], verificación de huellas dactilares [28], desplazamiento de robots móviles [14], y videojuegos [2], entre otros.

Por todo ello, la docencia y correcto aprendizaje de las técnicas de búsqueda heurística resultan de gran importancia en la formación de ingenieros en informática. No obstante, el aprendizaje, en ocasiones, se limita a entornos estáticos, en donde se lleva a cabo la mera búsqueda del camino más corto de un punto a otro. Esto limita severamente las posibilidades de la búsqueda heurística que, como hemos visto, puede ser empleada en distintos tipos de aplicaciones y problemas. En otros casos, la enseñanza se centra en cuestiones teóricas, más desacopladas de la práctica, lo que vuelve más árido todo el proceso y puede, con mayor facilidad, fomentar la desafección por la materia.

En este artículo, presentamos una propuesta docente sobre cómo abordar la enseñanza de técnicas de búsqueda heurística en entornos dinámicos mediante técnicas/herramientas de gamificación. Dichos entornos dinámicos requieren la integración “inteligente” de dos tipos de comportamientos:

- **Deliberativo:** representado por el algoritmo de búsqueda heurística, que permite planificar una ruta u otro tipo de actividades. En el caso de un videojuego, por ejemplo, calcular la ruta óptima para recoger algún objeto desde la posición actual del avatar.
- **Reactivo:** representado por la reacción ante cambios imprevistos en el entorno. En el caso de un videojuego, por ejemplo, la presencia de un enemigo.

En el contexto docente, la gamificación consiste en aplicar conceptos propios de videojuegos a contextos ajenos a éstos [27]. En nuestra propuesta para el aprendizaje de BH, las técnicas de gamificación se aplican a través de dos tipos de logros: niveles de dificultad y calificación. Esta propuesta se aplica en el contexto de un videojuego, cuya meta consiste en alcanzar una serie de objetivos en el menor tiempo posible. Para ello, el estudiante debe implementar un agente inteligente, que incorpore estrategias deliberativas y reactivas, y que juegue al videojuego de forma automática. Los distintos niveles del videojuego serán “desbloqueados”, a modo de logros, a medida que el estudiante avance en la implementación del agente inteligente. Para conseguirlo, estos niveles se organizan en progresiva dificultad de forma que para implementar el agente de cierto nivel, se debe partir de la implementación necesaria para superar el nivel anterior. Por otra parte, la calificación final depende (parcialmente) de una competición donde se evalúa el rendimiento de cada agente, es decir, el tiempo que cada agente requiere para resolver el nivel de mayor dificultad. Esto facilita una métrica objetiva con la que medir el rendimiento de cada agente. De esta manera, la gamificación también aparece estableciendo una competición donde el objetivo no es únicamente resolver el videojuego desbloqueando progresivamente sus niveles, sino resolverlo más rápido que el resto.

Este tipo de metodología ha facilitado que la adaptación de la docencia a la modalidad virtual en el curso 2019-20, debido a la pandemia de la COVID-19, haya sido directa, sin apenas ningún cambio en la organización docente de la asignatura. Se trata, por tanto, de una metodología flexible ante este tipo de contextos extraordinarios. Para analizarlo, se presenta un estudio estadístico entre las calificaciones del curso 2018-19 con docencia presencial y aquellas del curso 2019-20 con docencia virtual. El resultado de este análisis revela que no hay diferencias estadísticamente significativas entre las calificaciones de ambos cursos. Esto sugiere que la adaptación de la metodología propuesta es robusta a cambios en el tipo de docencia, debido al foco que pone en el trabajo autónomo e individual del estudiante.

A continuación, este artículo presenta una revisión

de algunos de los principales trabajos relacionados (sección 2) y el algoritmo más popular en el campo de la búsqueda heurística (A*, sección 3); el entorno en el que se desarrolla este proyecto docente (GVG-AI, en la sección 4); un ejemplo concreto de práctica que se puede materializar en el aula (sección 5), así como un análisis estadístico de las calificaciones en escenarios de docencia presencial y virtual (sección 6); y una serie de conclusiones al respecto (sección 7).

2. Trabajos relacionados

La ludificación, o gamificación, del aprendizaje es una aproximación educacional para motivar a los estudiantes a aprender mediante el uso de juegos y videojuegos en el entorno de aprendizaje [10]. Dicho proceso de gamificación ya ha sido propuesto y satisfactoriamente empleado en la docencia de la Ingeniería Informática con anterioridad [5]. En nuestra propuesta docente, se emplea un entorno de desarrollo de controladores para videojuegos que se presentará en la sección 5. Aparte de potenciar la diversión y motivación durante el aprendizaje de las técnicas de búsqueda heurística, nuestra propuesta permite introducir logros con los que desbloquear los distintos niveles del juego, así como una competición entre estudiantes que afecta (parcialmente) a su calificación final.

Podemos encontrar aproximaciones basadas en gamificación al aprendizaje de ciertas ramas de la IA, principalmente aprendizaje automático y/o centrado en niños y en la formación escolar más elemental [11, 12, 13, 25]. También existe alguna aproximación gamificadora a la enseñanza de técnicas de búsqueda heurística [6], pero se trata de una metodología docente basada en entornos inmersivos de realidad virtual, con todos los costes e implicaciones hardware y software que ello conlleva. Por otro lado, esta aproximación se centra en clarificar la explicación de conceptos teóricos, ignorando por tanto los aspectos prácticos de la búsqueda heurística en entornos dinámicos.

Las herramientas de gamificación también han sido usadas en otras áreas de conocimiento, como arquitectura de computadores [1], lenguas extranjeras [3], física [23], ingeniería mecánica [17], o arquitectura [19].

3. El algoritmo A*

El algoritmo A* [7] es un algoritmo de búsqueda de caminos en grafos para intentar calcular el camino óptimo entre dos nodos del mismo –el nodo inicial y el nodo objetivo–. Para ello, hace uso de una función heurística que guía el proceso de búsqueda. Nótese que este algoritmo puede aplicarse igualmente a cualquier proceso en el que se puedan representar los distintos

estados (nodos del grafo), las transiciones entre ellos (aristas del grafo) y el coste de dichas transiciones (peso de las aristas). Por ejemplo, un proceso industrial donde cada estado del proceso se representa en un nodo de un grafo, con un estado inicial (el actual) y un estado objetivo, y donde las posibles transiciones de un estado a otro se representan mediante las aristas de dicho grafo y el coste de cada transición se representa en el peso de cada arista.

Sea $G(V, w)$ un grafo dirigido y ponderado donde V representa el conjunto de nodos del grafo y $w : V^2 \rightarrow \mathbb{R}$ es la función de pesos entre cualquier par de nodos del grafo. Sean $x, y \in V$ respectivamente los nodos inicial y objetivo, se definen las funciones $f, g, h : V \rightarrow \mathbb{R}$, donde $g(n)$ representa el coste mínimo para alcanzar el nodo actual n desde el nodo inicial x , $h(n)$ representa la función heurística que estima el coste mínimo para alcanzar el nodo objetivo y desde el nodo actual n , y $f(n) = g(n) + h(n)$ es la suma de ambas.

Iterativamente, el algoritmo A* selecciona un nodo candidato n y lo expande. El conjunto de nodos candidatos está inicialmente formado únicamente por el nodo inicial x , y el proceso de expansión consiste en insertar en el conjunto de candidatos los vecinos del nodo expandido n , con sus correspondientes valores de $g(n)$ y $h(n)$. Este proceso se repite hasta que el nodo expandido es precisamente el nodo objetivo y . El criterio para seleccionar un nodo candidato es elegir aquel candidato con menor valor en la función $f(n)$. En caso de que un nodo sea alcanzado por caminos diferentes, se conservará el camino más corto, es decir, aquel camino con menor valor en $g(n)$.

Este algoritmo extiende al algoritmo de Dijkstra [4] mediante el uso de la función heurística $h(n)$ –el algoritmo de Dijkstra se puede ver como una versión simplificada del algoritmo A* donde únicamente se hace uso de la función $g(n)$ –. El objetivo de esta función es precisamente priorizar la búsqueda sobre aquellos nodos más prometedores en el camino óptimo, reduciendo así los tiempos de ejecución necesarios para obtenerlo.

En la Figura 1 se muestra un grafo de ejemplo y la función heurística $h(n)$ asociada a cada nodo, mientras que en el Cuadro 1 se describe el desarrollo del algoritmo A* en este grafo, donde el nodo inicial es A y el nodo objetivo es D . Inicialmente, el conjunto de candidatos está formado únicamente por el nodo inicial A , cuyo valor $f(A) = g(A) + h(A) = 0 + 10 = 10$. En la primera iteración se expande este nodo, insertando en el conjunto de candidatos a los vecinos del nodo expandido, es decir, los nodos B y C , con sus correspondientes valores de $f(B)$ y $f(C)$. En la siguiente iteración se expande el nodo del conjunto de candidatos con menor $f(n)$, es decir, C , y se generan sus sucesos-

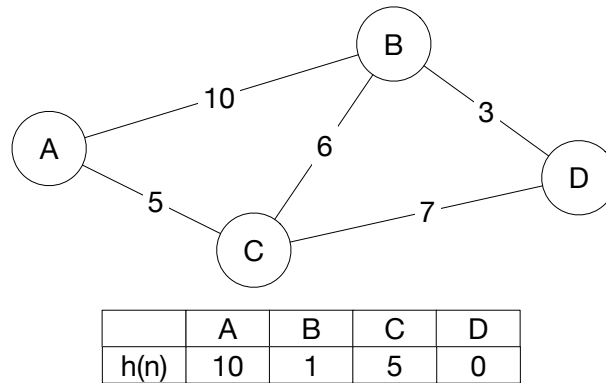


Figura 1: Grafo de ejemplo y su correspondiente función heurística $h(n)$ para alcanzar el nodo objetivo D.

Iteración	Nodo expandido n	Candidatos ($f(n) = g(n) + h(n)$)
0	-	A (10=0+10)
1	A	B (11=10+1), C (10=5+5)
2	C	B (11=10+1), B (12=11+1) , D (12=12+0)
3	B	D (12=12+0), C (21=16+5)
4	D	Fin —
Camino encontrado: A – C – D (coste=12)		

Cuadro 1: Desarrollo del algoritmo A* en el grafo de ejemplo de la Fig. 1, donde el nodo inicial es A y el nodo objetivo es D. Los valores de las funciones $f(n)$, $g(n)$ y $h(n)$ se indican entre paréntesis.

res. En este paso puede observarse cómo se descarta el camino $A - C - B$ puesto que es más largo que el camino ya existente $A - B$. Así continúan las iteraciones hasta que el nodo expandido es precisamente el nodo objetivo D. En este caso, el camino óptimo encontrado es $A - C - D$ con coste 12.

4. El entorno GVG-AI

GVG-AI (General Video Game - Artificial Intelligence) [22] es un entorno para el desarrollo de controladores inteligentes para jugar a videojuegos y generadores de contenido dinámico para los mismos. El objetivo de este entorno es, por una parte, ofrecer una plataforma versátil donde tanto los controladores inteligentes como los generadores de contenido sean capaces de aprender a jugar y generar contenido en cualquier videojuego (que puede ser inicialmente desconocido para el controlador/generador), y por otra, proveer una herramienta donde se pueda evaluar la capacidad de estos controladores en esta tarea. Este último objetivo ha dado lugar a la GVG-AI Competition, donde se evalúan anualmente las técnicas presentadas para este fin.

Con el fin de poder entrenar y/o testear los controladores, el entorno GVG-AI ofrece un conjunto de videojuegos de entrenamiento. Este conjunto se divide en varias categorías en función de la tipología del juego (2

jugadores, mapas, etc. . .). En la Figura 2 se muestra los videojuegos CamelRace y BoulderDash de la categoría “gridphysics”. Ésta es una categoría de videojuegos de un jugador (el jugador/controlador juega contra la máquina) donde el tiempo se discretiza en ticks¹, y el objetivo debe ser alcanzado en un número máximo de ticks y/o antes que el resto de rivales. En el videojuego CamelRace, el objetivo es llegar a la línea de meta, sorteando los posibles obstáculos del mapa, antes que el resto de contrincantes. En BoulderDash, el objetivo es recoger un número de gemas y dirigirse al portal de salida, realizando los túneles correspondientes para desplazarse por el mapa y evitando enemigos y obstáculos que pueden eliminar al avatar. El primer caso se trata de un juego completamente determinista, dado que no se ve afectado por eventos aleatorios; el segundo caso es semi-determinista dado que la componente estocástica del videojuego solo se produce si los enemigos se liberan.

Ambos videojuegos son ejemplos adecuados para el aprendizaje y desarrollo de técnicas de búsqueda heurística, donde además se puede evaluar objetivamente el rendimiento de un controlador, dado que el comportamiento inteligente se dará en aquellos capaces de alcanzar el objetivo más rápido, es decir, en el menor número de ticks posible.

¹El tick, por tanto, es la unidad de tiempo empleada en nuestro entorno. Se considera que, para una misma tarea, es preferible conseguir el objetivo en un menor número de ticks.



Figura 2: Videojuegos CamelRace (arriba) y BoulderDash (abajo) del entorno GVG-AI.

5. Propuesta de metodología docente basada en gamificación

La metodología docente propuesta consiste en desarrollar un controlador, basado en A* (o en alguna de sus variantes), dentro del entorno GVG-AI, que guíe a un avatar a resolver un juego en distintos niveles. El juego escogido es el juego con índice 11 en los tipos de juego `singleplayer`² de la distribución de GVG-AI, denominado BoulderDash. El objetivo de la práctica es que los estudiantes se familiaricen con las técnicas de búsqueda heurística en entornos dinámicos, que incluyen una vertiente deliberativa y una reactiva. Para ello, se proponen 5 problemas/tareas de progresiva complejidad:

1. Comportamiento deliberativo simple: búsqueda del camino óptimo al portal (sin enemigos, pero con la posible presencia de obstáculos).
2. Comportamiento deliberativo compuesto: búsqueda de 10 gemas (en un mapa con un número superior o igual de gemas) y salida por el portal.
3. Comportamiento reactivo simple: mantenerse alejado de un enemigo durante un tiempo predeter-

minado (2000 ticks)³.

4. Comportamiento reactivo compuesto: mantenerse alejado de dos enemigos durante un tiempo predeterminado (2000 ticks).
5. Comportamiento reactivo-deliberativo: búsqueda de 10 gemas (en un mapa con un número superior o igual de gemas), evitando al enemigo presente en el mapa y, una vez se tengan todas, alcanzar el portal dentro de los límites de tiempo predeterminados (2000 ticks).

Nótese que estas tareas pueden entenderse como logros, puesto que se requiere superar un nivel antes de afrontar la implementación del siguiente. Por tanto, se entiende que un nivel está bloqueado hasta que no se ha superado el anterior.

Los mapas son proporcionados por los profesores de prácticas, aunque se recomienda que los estudiantes creen sus propios mapas y verifiquen el comportamiento de sus algoritmos en ellos. Por otro lado, resulta esencial conocer y comprender los detalles del juego:

- En cada tick, el controlador dispone de 50ms para decidir la acción a realizar. En caso contrario, el

²Se pueden encontrar en el fichero `examples/all_games_sp.csv`.

³Si el avatar está en contacto con un enemigo es eliminado y, por tanto, el videojuego no es superado.

controlador es eliminado y, por tanto, el estudiante suspende el ejercicio.

- En BoulderDash, un cambio de orientación implica un tick adicional al propio movimiento. Esto es importante, de cara a utilizar el menor número de ticks, e integrar este conocimiento en $g(n)$ de A^* .

Las características que presenta esta aproximación son las siguientes: permite plantear varios objetivos; priorizar objetivos; alcanzar un objetivo es un problema de búsqueda heurística en entornos dinámicos (pero teniendo que considerar aspectos como la presencia de enemigos y el límite de tiempo); integrar comportamientos reactivo y deliberativo; replanificar (dado que el plan elaborado puede fallar); y replantear objetivos.

En cuanto a la evaluación, el planteamiento del ejercicio permite asignar con mucha facilidad una calificación según el tiempo de ejecución. Debido a que el comportamiento del controlador es objetivamente medible usando como métrica el número de ticks empleados para resolver un nivel concreto, la resolución de la práctica se puede plantear mediante herramientas/técnicas de gamificación donde el objetivo es conseguir un controlador mejor que el resto (en cuanto al número de ticks que emplea). Para ello, nuestra propuesta contempla que parte de la calificación final del estudiante se obtenga mediante una competición donde se evalúan los controladores de todos los estudiantes. Por otro lado, desde el punto de vista de evaluar los distintos niveles que presentan un comportamiento estocástico (por ejemplo, en presencia de enemigos), resulta sencillo ejecutar varias veces cada nivel y evaluar automáticamente el número de veces que sobrevive el avatar. Todo esto representa una gran ventaja, dado que da seguridad al estudiante durante el proceso de evaluación (al eliminar la subjetividad y arbitrariedad) y facilita la corrección por parte del profesor.

Por tanto, esta propuesta de metodología docente basada en la gamificación permite al estudiante desarrollar los distintos conceptos relativos a la búsqueda heurística en entornos dinámicos en distintos escenarios de progresiva dificultad. Este aprendizaje se desarrolla de manera independiente por parte de cada estudiante, donde el entorno del videojuego estimula su interés hacia conceptos de mayor complejidad gracias a la modularidad de la propuesta. Además, cabe resaltar de nuevo la facilidad con la que esta metodología se puede adaptar a escenarios de docencia virtual.

6. Análisis de calificaciones en docencia presencial y virtual

En esta sección analizamos la capacidad de adaptación, en cuanto a modelo pedagógico, de esta propuesta metodológica a la modalidad de docencia online. Pa-

ra ello, se analizan las calificaciones de los cursos académicos 2018-19 (con docencia presencial) y 2019-20 (con docencia virtual).⁴

	2018-19	2019-20
Media	8.526	7.271
Stdev.	1.033	2.784
Máximo	10.000	10.000
Perc. 75	9.000	9.625
Mediana	8.750	8.000
Perc. 25	8.250	5.800
Mínimo	5.000	0.000

Cuadro 2: Estadísticas de las calificaciones en los cursos 2018-19 y 2019-20. Los tamaños de las muestras son, respectivamente, de 58 y 56 calificaciones.

En la Figura 3 se representan los histogramas correspondientes a las calificaciones de ambos cursos académicos. Un resumen de algunas estadísticas puede encontrarse en el Cuadro 2. En un primer análisis numérico y visual, se puede apreciar ligeras diferencias en las calificaciones de ambos años.

Estas pequeñas diferencias pueden deberse a numerosos factores. La explicación más simple es el hecho de tener muestras de estudiantes diferentes, los cuales no tienen por qué tener el mismo rendimiento académico. A esta circunstancia debe sumarse los factores humanos relacionados con el inicio repentino de la pandemia y los cambios abruptos que ello supuso, tanto en la docencia como, en general, en la rutina diaria de todos nosotros.

No obstante, cabe preguntarse si verdaderamente existen diferencias estadísticamente significativas entre ambas muestras. Para verificarlo, realizaremos la prueba de la U de Mann-Whitney (también conocida como prueba de suma de rangos Wilcoxon) [15], una prueba estadística no paramétrica apropiada para analizar diferencias entre las dos muestras. En este caso, la hipótesis nula H_0 establece que las medianas de las dos muestras son iguales, mientras que la hipótesis alternativa H_1 establece que medianas difieren.

Esta prueba devuelve como resultado el estadístico $U = 3,616$, con $z = 1,5936$ como estadístico aproximado⁵ que establece el umbral de U para aceptar la hipótesis alternativa, y $p\text{-value} = 0,1110$, que indica la probabilidad de que sea posible el valor estadístico obtenido siendo la hipótesis nula cierta. Por tanto, se rechaza la hipótesis alternativa. Es decir, no existen diferencias estadísticamente significativas entre las calificaciones del curso 2018-19, con docencia presencial, y las del curso 2019-20, con docencia virtual. Este

⁴Todavía no existen datos disponibles sobre las calificaciones del curso 2020-21.

⁵Debido al tamaño de las muestras debe calcularse este estadístico aproximado.

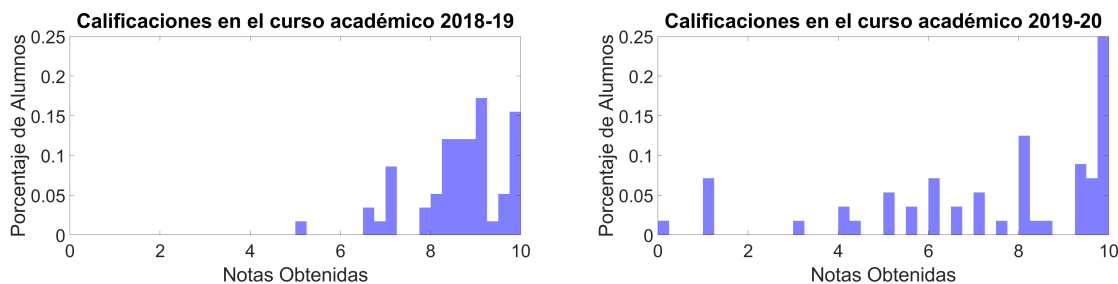


Figura 3: Histogramas de calificaciones en los cursos 2018-19 y 2019-20.

resultado sugiere la idoneidad de la aproximación empleada para tratar situaciones de docencia virtual como la actual. Posiblemente, esto es gracias al trabajo individual y autónomo del estudiante que las herramientas de gamificación facilitan.

7. Conclusiones

En este trabajo hemos presentado una aproximación basada en gamificación para la enseñanza de técnicas de búsqueda heurística en entornos dinámicos, en los que se requiere la integración de comportamientos reactivos y deliberativos. Para ello, se hace uso de un entorno de desarrollo de controladores de videojuegos, que permite una gran flexibilidad a la hora de emplear distintos juegos, plantear distintos objetivos, e incrementar o decrementar la dificultad. Nuestra propuesta es modular, en donde se parte de un comportamiento deliberativo simple, se avanza hacia un comportamiento deliberativo compuesto, pasando por un reactivo simple y un reactivo compuesto, y se finaliza con la integración de todos estos aspectos en un ejercicio reactivo-deliberativo. El modo en que se ha definido el juego facilita enormemente la evaluación de los ejercicios, aportando un framework objetivable en el que los alumnos perciben como justa su propia evaluación y simplifica el trabajo evaluador del docente. Esta metodología docente permite una fácil adaptación a la modalidad de docencia virtual, gracias a su diseño modular y progresivo basado en el trabajo individual e independiente de cada estudiante. El análisis estadístico de las calificaciones en modalidades presencial y virtual demuestra que no hay diferencias estadísticamente significativas entre ambas, sugiriendo la robustez de la propuesta en el caso de docencia virtual cuando no sea posible continuar con la modalidad presencial.

Como trabajo futuro, quedaría realizar la validación experimental de las ventajas de esta aproximación docente con respecto a otras, dado que consideramos que nuestra aproximación facilita el aprendizaje y mejora las calificaciones obtenidas por los estudiantes. Adicionalmente, sería interesante evaluar el uso de distin-

tos elementos de gamificación, como el impacto del tipo de videojuego en los logros conseguidos o la influencia del tipo de jugador en los resultados.

Referencias

- [1] Francisco J. Andújar, Arturo Gonzalez-Escribano, Javier Bastida Ibañez, y Yuri Torres de La Sierra. Aplicación de gamificación competitiva y colaborativa en asignaturas básicas de arquitectura de computadoras. En *XXVI Jornadas sobre la Enseñanza Universitaria de la Informática (JENUI 2020)*, 2020.
- [2] Vadim Bulitko, Yngvi Björnsson, Nathan Sturtevant, y Ramon Lawrence. Real-time heuristic search for pathfinding in video games. En *Artificial Intelligence for Computer Games*, pp. 1–30, 2011.
- [3] Alejandro Calderón Sánchez y Mercedes Ruiz. Aprendizaje integrado de contenidos y lenguas extranjeras mediante juegos serios para la formación bilingüe en ingeniería del software. En *XXVI Jornadas sobre la Enseñanza Universitaria de la Informática (JENUI 2020)*, 2020.
- [4] Edsger W. Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1:269–271, 1959.
- [5] Carina S. González González y Alberto Mora Carreño. Técnicas de gamificación aplicadas en la docencia de ingeniería informática. *ReVisión*, 8, 2015.
- [6] Foteini Grivokostopoulou, Isidoros Perikos, y Ioannis Hatzilygeroudis. An innovative educational environment based on virtual reality and gamification for learning search algorithms. En *2016 IEEE Eighth International Conference on Technology for Education (T4E)*, pp. 110–115, 2016.
- [7] Peter E. Hart, Nils J. Nilsson, y Bertram Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE Trans. Syst. Sci. Cybern.*, 4(2):100–107, 1968.

- [8] Feng-Hsiung Hsu. Ibm's deep blue chess grandmaster chips. *IEEE Micro*, 19(2):70–81, 1999.
- [9] Joint Task Force on Computing Curricula, Association for Computing Machinery (ACM), and IEEE Computer Society. Computer science curricula 2013: Curriculum guidelines for undergraduate degree programs in computer science. *Association for Computing Machinery*, 2013.
- [10] Karl M. Kapp. *The gamification of learning and instruction: game-based methods and strategies for training and education*. Joh Wiley and Sons, 2012.
- [11] Kapsu Kim. An artificial intelligence education program development and application for elementary teachers. *Journal of The Korean Association of Information Education*, 23:629–637, 2019.
- [12] Kapsu Kim y Youngki Park. A development and application of the teaching and learning model of artificial intelligence education for elementary students. *Journal of The Korean Association of Information Education*, 21:139–149, 2017.
- [13] Kapsu Kim y Youngki Park. Development of a board game-based gamification learning model for training on the principles of artificial intelligence learning in elementary courses. *Journal of The Korean Association of Information Education*, 23:229–235, 2019.
- [14] Sven Koenig y Maxim Likhachev. Improved fast replanning for robot navigation in unknown terrain. En *2002 IEEE International Conference on Robotics and Automation, ICRA 2002.*, pp. 968–975. IEEE, 2002.
- [15] Henry B. Mann y Donald R. Whitney. On a test of whether one of two random variables is stochastically larger than the other. *Annals of Mathematical Statistics*, 18:50–60, 1947.
- [16] Alberto Martelli. An application of heuristic search methods to edge and contour detection. *Communications of the ACM*, 19(2):73–83, 1976.
- [17] Carmen Mata, Elena M. Beamud, y David Calvo-Parra. Utilización de técnicas de gamificación para potenciar el aprendizaje en una asignatura optativa. En *XXIII Congreso Universitario de Innovación Educativa en las Enseñanzas Técnicas, CUIEET 2015*, pp. 822–830, 2015.
- [18] John McCarthy. What is artificial intelligence? <http://jmc.stanford.edu/articles/whatisai/whatisai.pdf>, 2007.
- [19] Miguel Muñoz-Calvente, Pelayo Fernández, Mario López, y Javier Gracia. Gamificación en la impartición de cálculo de estructuras. En *XXVI Congreso Universitario de Innovación Educativa en las Enseñanzas Técnicas, CUIEET 2018*, pp. 612–620, 2018.
- [20] Allen Newell, John C. Shaw, y Herbert A. Simon. Empirical explorations of the logic theory machine: a case study in heuristic. En *Western Joint Computer Conference: Techniques for reliability*, pp. 218–230, 1957.
- [21] Krishna R. Pattipati y Mark G. Alexandridis. Application of heuristic search and information theory to sequential fault diagnosis. *IEEE Transactions on Systems, Man, and Cybernetics*, 20(4):872–887, 1990.
- [22] Diego Pérez-Liébana, Jialin Liu, Ahmed Khalifa, Raluca D. Gaina, Julian Togelius, y Simon M. Lucas. General video game AI: A multitrack framework for evaluating agents, games, and content generation algorithms. *IEEE Transactions on Games*, 11(3):195–214, 2019.
- [23] Montserrat Rivas Ardisana, Pablo Álvarez Alonso, y Pedro Gorría Korres. La ludificación como herramienta de motivación en la asignatura bilingüe waves and electromagnetism. En *XXVI Congreso Universitario de Innovación Educativa en las Enseñanzas Técnicas, CUIEET 2018*, pp. 600–611, 2018.
- [24] Stuart J. Russell y Peter Norvig. *Artificial Intelligence - A Modern Approach, Third International Edition*. Pearson Education, 2010.
- [25] Bawornsak Sakulkueakulsuk, Siyada Witoon, Potiwat Ngarmkajornwivat, Pornpen Pataranutaporn, Werak Surareungchai, Pat Pataranutaporn, y Pakpoom Subsoontorn. Kids making AI: Integrating machine learning, gamification, and social context in STEM education. En *2018 IEEE International Conference on Teaching, Assessment, and Learning for Engineering (TALE)*, pp. 1005–1010, 2018.
- [26] David Silver, Aja Huang, et. al. Mastering the game of go with deep neural networks and tree search. *Nature*, 529(7587):484–489, 2016.
- [27] Miguel De Simón-Martín, Ana M. Diez-Suárez, Jorge Blanes-Peiró, David Borge-Diez, y A. González-Martínez. Aplicación de técnicas de gamificación para la consolidación de conocimientos en asignaturas del Área de ingeniería eléctrica y energética. En *IX Congreso Nacional de Ingeniería Termodinámica*, 2015.
- [28] David M. Tumey, Tianning Xu, y Craig Arndt. U.S. patent no. 6,963,659. *U.S. Patent and Trademark Office*, 2005.