**JUEGO TRES EN RAYA UTILIZANDO ALGORITMO MINMAX**

**Camila Cabrera**

**Mateo Encalada**

**Ariel Raura**

**Joe Cordero**

**Universidad de las Américas**

**ISWZ3408-2700- Matemáticas discretas**

**María Cristina Jiménez Hernández**

**28 de diciembre de 2022**

**Objetivo**

Indagar sobre el alcance del algoritmo minmax y su aplicabilidad en el juego tres en raya

**Marco teórico**

La ganancia o pérdida, dentro de situaciones en las que tenemos que tomar decisiones, ha sido un tema de continua investigación, es por ello por lo que, para comprender la mejor manera de lograr una ganancia en situaciones donde dos personas toman decisiones, se ha implementado el algoritmo Minimax, el cual, para entenderlo, necesitamos definir unos conceptos clave en el tema de estudio, como son algoritmos, combinatoria, árbol de búsqueda binaria y algoritmo Minimax.

* **Algoritmo**

Oxford Learner’s Dictionaries propone la siguiente definición de algoritmo: Un algoritmo es un conjunto ordenado de operaciones sistemáticas que permite hacer un cálculo y hallar la solución de un tipo de problemas.

* **Combinatoria**

Así también Oxford Learner’s Dictionaries nos presenta a la combinatoria como: Parte de la matemática que estudia la formación de subconjuntos o agrupaciones de elementos partiendo de un conjunto dado, teniendo en cuenta la ordenación y el número de esos elementos.

* **Árbol de búsqueda**

Un árbol de búsqueda binaria es un árbol binario en el que las claves están dispuestas de tal manera que, en cada nodo, las claves del subárbol izquierdo son menores que la clave del nodo y las claves del subárbol derecho son mayores que las del nodo y que, por definición, no se permiten duplicados en el árbol. (Malerba and Murthy, 1998)

* **Algoritmo Minimax**

Finalmente, estos tres conceptos nos permitirán comprender mejor lo que es el algoritmo Minimax, pues este algoritmo se compone de una optimización combinatoria dentro de los árboles de búsqueda. (Borovska and Lazarova, 2007)

Según Iguider, Bousselam, Elissati, Chami, and En-Nouaary (2019) se nos presenta la siguiente definición del algoritmo Minimax:

El algoritmo Minimax se aplica en juegos de dos jugadores (jugadores Min y Max). Cada jugador conoce los posibles movimientos del oponente. El objetivo de Minimax es encontrar el mejor movimiento para jugar para el jugador Max, desde un estado de juego determinado. El principio es construir un árbol de búsqueda, que represente todas las posibles evoluciones del juego desde el estado actual. Una función de evaluación toma un estado en el juego y otorga una puntuación numérica asociada a cada jugador. Los jugadores Max y Min utilizan la misma función de evaluación para sus puntuaciones. Para una profundidad determinada en el árbol de búsqueda, los nodos inferiores del árbol (estados del juego) se evalúan mediante la función de evaluación. Luego, los nodos superiores se rellenan hasta la parte superior del árbol. Los nodos para el jugador MAX reciben el valor máximo de sus hijos. Los nodos del reproductor MIN reciben el valor mínimo de sus hijos. De hecho, el jugador Max intenta maximizar su puntuación y el jugador Min también intentará maximizar su puntuación (minimizar la puntuación de Max).

**Organización del trabajo**

|  |  |
| --- | --- |
| **Integrante** | **Responsabilidad** |
| Camila Cabrera | 1. tabla de estudio combinatorio del juego (Árbol de posibilidades). 2. Marco teórico, 3. Conclusiones |
| Joe Cordero | 1. tabla de tiempos que se demoran los peores casos y los mejores casos 2. Marco teórico 3. Presentación del problema |
| Ariel Raura | 1. instrucciones para ejecutar el código en Windows y en Linux 2. Marco teórico, 3. Presentación del problema |
| Mateo Encalada | 1. Descripción del juego “Tres en Raya”,Explicación de las principales rutinas. 2. Presentación del problema 3. Conclusiones |

**Presentación del trabajo**

El árbol de decisión es un modelo predictivo que divide el espacio de los predictores agrupando observaciones con valores similares para la variable respuesta o dependiente. En concreto la creación de un árbol de decisión de un problema de clasificación se lleva a cabo aplicando el algoritmo de Hunt que se basa en la división en sub-conjuntos que buscan una separación óptima. Dado un conjunto de registros de entrenamiento de un nodo, si pertenecen a la misma clase se considera un nodo terminal, pero si pertenecen a varias clases, se dividen los datos en sub-conjuntos más pequeños en función de una variable y se repite el proceso. Si bien es cierto q el aprendizaje del árbol de decisiones emplea una estrategia de divide y vencerás mediante la realización de una búsqueda codiciosa para identificar los puntos de división óptimos dentro de un árbol. Este proceso de división se repite de forma recursiva de arriba hacia abajo hasta que todos o la mayoría de los registros se hayan clasificado bajo etiquetas de clase específicas. Que todos los puntos de datos se clasifiquen o no como conjuntos homogéneos depende en gran medida de la complejidad del árbol de decisión. Esto resulta en que independientemente de la importancia de la elección, un árbol de decisiones es una herramienta sencilla para evaluar tus opciones y encontrar la solución ideal, de ahí que, la presente investigación descriptiva tiene como objetivo identificar ¿Cuáles son las necesidades de construir un árbol de decisiones en el desarrollo actualmente?

El alcance de la investigación es descriptivo-práctico al pretender describir y determinar el uso de arboles binarios o toma de decisiones en las posibilidades del juego tres en raya, mediante el uso del algoritmo minimax, podemos construir un arbol de busqueda binario que permite a la computadora encontrar el mejor movimiento en el juego.

La investigación a la vez, alcanzara un nivel de profundidad en cuanto al tema de árboles binarios y de toma de decisiones, para su correcto uso en el estudio de posibilidades.

**Conclusiones**

* Algoritmo de decisión para minimizar la pérdida máxima aplicada en juegos de adversarios**,** Minimax lo que hace informalmente es considerar todos los casos que se pueden dar desde donde estás y en base a esto, elegir el movimiento que te lleve a un resultado mejor asumiendo que el contrario hará el movimiento que más te perjudique.
* La búsqueda con adversarios se usa normalmente en juegos en los que intervienen más de un jugador, aunque también la podemos encontrar en contextos en los que se han de tomar decisiones (acciones futuras) en entornos con cierta incertidumbre.
* El problema de minimax es que el árbol de estados que crea es muy grande para juegos más complejos, pero con el 3 en raya hay pocos y lo puede calcular entero, de manera que el ordenador juega de forma perfecta.
* En conclusión gracias al algoritmo minmax nos permite calcular las posibles jugadas de los jugares (2 jugadores como mucho), con este algoritmo se pudimos estudiar las combinaciones posibles del juego tres en raya.

**Bibliografía**

* Oxford Learner’s Dictionaries | Find definitions, translations, and grammar explanations at Oxford Learner’s Dictionaries. (s. f.). <https://www.oxfordlearnersdictionaries.com/>
* Joseph F. Malerba and Narayan Murthy. 1998. An optimal search tree: modified complete binary search tree. In Proceedings of the 36th annual Southeast regional conference (ACM-SE 36). Association for Computing Machinery, New York, NY, USA, 173–177. <https://doi-org.bibliotecavirtual.udla.edu.ec/10.1145/275295.275354>
* Plamenka Borovska and Milena Lazarova. 2007. Efficiency of parallel minimax algorithm for game tree search. In Proceedings of the 2007 international conference on Computer systems and technologies (CompSysTech '07). Association for Computing Machinery, New York, NY, USA, Article 14, 1–6. <https://doi-org.bibliotecavirtual.udla.edu.ec/10.1145/1330598.1330615>
* Adil Iguider, Kaouthar Bousselam, Oussama Elissati, Mouhcine Chami, and Abdeslam En-Nouaary. 2019. Embedded systems hardware software partitioning using minimax algorithm. In Proceedings of the 4th International Conference on Smart City Applications (SCA '19). Association for Computing Machinery, New York, NY, USA, Article 31, 1–6. https://doi-org.bibliotecavirtual.udla.edu.ec/10.1145/3368756.3369009