

## Búsqueda en un espacio de estados

1. **Supongamos que, para resolver un problema de espacio de estados, dos personas deciden representar de la misma manera los estados y los operadores, y definen de la misma manera el estado inicial, el final y la descripción de las operaciones. Sin embargo, aplicando el procedimiento de búsqueda en profundidad obtienen soluciones distintas ¿Por qué puede ocurrir esto?**

Pues puede deberse a que la base de datos de estados del agente no sean iguales y que priorice una acción antes que otra o que esa acción se resuelva de manera distinta.

2. **-¿En qué afecta al modelo general de resolución mediante búsqueda del agente deliberativo el no disponer de información completa sobre el estado del mundo o al no tener certeza acerca de los efectos de las acciones sobre el mundo?**

Que el agente no podría crear una planificación de sus movimientos, y por tanto no podría realizar acciones que resolviesen el problema.

3. **-Describe en cinco líneas las diferentes estrategias de control para un sistema de búsqueda.**

Puede ser irrevocable, de tipo backtracking o retroactiva y de búsqueda de grafos, las diferencias entre las tres radica en que las irrevocables no guardan información de las acciones a largo plazo, las retroactivas solo guardan el camino que llevamos andado y las de búsqueda que se realizan a través de un árbol de acciones que son nuestra información de uso.

4. **Describir brevemente el parecido y diferencias entre la búsqueda en anchura y el descenso iterativo.**

El recorrido iterativo siempre ejecuta el mismo llevando los puntos desde los más a la izquierda hasta los puntos más a la derecha, mientras que la búsqueda por anchura va buscando desde los puntos a través de los nodos que están a su mismo nivel.

5. **Enumerar las condiciones que se requieren para que un problema se pueda descomponer. Enumerar las condiciones que se requieren para un problema se pueda resolver mediante descomposición. ¿Son las mismas?**

6. **¿Qué es la búsqueda no informada y en qué se diferencia de la búsqueda informada (o heurística)?**

La diferencia es que en la búsqueda informada no tenemos ninguna base especializada que sepa cómo resolver el problema, por tanto debemos usar una búsqueda exhaustiva mientras que en la búsqueda informada usamos bases de conocimiento de un especialista que nos puede llevar a búsquedas más óptimas.

- 7. Describir en menos de cinco líneas el algoritmo de escalada de reinicio aleatorio, y detallar las condiciones para que sea aplicable a un problema.**

Es un algoritmo iterativo que comienza con una solución arbitraria a un problema, luego intenta encontrar una mejor solución incrementando un único elemento de la solución.

- 8. ¿Qué característica esencial aporta el algoritmo de enfriamiento simulado frente al resto de métodos de escalada? ¿Qué le aporta? ¿Cuál es la principal dificultad de aplicar dicho algoritmo en un caso concreto?**

Evita los óptimos locales, porque el algoritmo también visita resultados peores al resultado de solución, la dificultad del mismo reside en encontrar los parámetros probabilísticos que le dan todo el potencial al programa como encontrar la temperatura inicial.

- 9. ¿Cómo combinación de qué dos estrategias de búsqueda puede verse el algoritmo A\*?**

Como la mezcla de costo uniforme y de profundidad.

- 10. El algoritmo A\* utiliza una lista de ABIERTOS y una lista de CERRADOS. Describe el propósito de cada una de esas listas.**

En abiertos se guardan los nodos que faltan por visitar y en cerrados se guardan los nodos ya visitados que no son el mejor padre.

- 11. La búsqueda A\* utiliza una heurística combinada para seleccionar el mejor camino a seguir a través del espacio de estados hacia el objetivo. Define las dos funciones utilizadas ( $h(n)$  y  $g(n)$ ).**

- 12. El algoritmo A\* no termina mientras no se seleccione un nodo objetivo para su expansión. Sin embargo, podríamos encontrar un camino al objetivo mucho antes de que se seleccione un nodo objetivo para su expansión. ¿Por qué no se termina en el momento en que se encuentre un nodo objetivo? Ilustra la respuesta con un ejemplo.**

Porque puede ser que el nodo objetivo al que lleguemos no sea el nodo óptimo.

- 13. ¿Cuál es el objetivo de una función heurística aplicada a la búsqueda en el espacio de estados?**

Encontrar la mejor solución dentro del espacio de estados.

- 14. ¿Cuál es la definición de heurística admisible?**

Pues se clasificaría que dentro de la búsqueda de la mejor opción si los datos son trabajables o admisibles, este algoritmo o heurística nos permitiera encontrar la mejor opción.

- 15. ¿Qué condiciones garantizan que el algoritmo A\* obtenga la solución óptima?**

Si el árbol es finito y la función  $h(n)$  es admisible.

- 16. ¿Es la búsqueda primero en anchura un caso especial de la búsqueda de coste uniforme? Razona la respuesta.**

Se podría ver así, en el caso de que el árbol tuviese los nodos colocados en orden y siempre los

nodos hijos fueran mayores al padre y a los hermanos del mismo, también tendrían que estar ordenados entre sí en anchura.

**17. ¿Son la búsqueda primero en anchura, búsqueda primero en profundidad, y la búsqueda de coste uniforme casos especiales de la búsqueda primero el mejor? Razona la respuesta.**

No, la búsqueda primero el mejor es de tipo heurística y necesitamos información especial para poderla llevarla a cabo.

**18. ¿Es la búsqueda de coste uniforme un caso especial de la búsqueda A\*? Razona la respuesta.**

No, porque A\* abre los nuevos hijos y trabaja sobre ellos hasta llegar a una solución y luego trabaja sobre las demás según una solución mínima, mientras que la de coste mínimo abre todo el árbol de una manera menos eficiente, solo podríamos decir que estos algoritmos se parecen en casos muy particulares.

**Búsqueda con adversario y juegos**

**19. Explica qué se interpreta por juegos de Información Perfecta. Da un par de ejemplos de juegos con información perfecta, y otro de un juego que no lo sea.**

Un juego de información perfecta es aquel en los jugadores tienen a su disposición toda la información de la situación del juego. El ajedrez sería un juego de este tipo. Mientras que un juego de información no completa aparece un tipo de jugador que no es igual al otro por tanto podría trabajar o no de una manera distinta a nuestro propio jugador.

**20. ¿Qué se entiende por Búsqueda con Adversario, y en qué difiere de la búsqueda tradicional?**

La diferencia de la búsqueda con adversario y la búsqueda tradicional se tiene en que cada nivel del árbol cambia el jugador por tanto se realizaría un movimiento nuestro y acto seguido se realizaría un movimiento de nuestro rival, de esta forma se realizarían movimientos hasta que uno de los dos jugadores gane la partida. La diferencia entre ambos reside en la subdivisión de niveles, en la tradicional todos los niveles son de un jugador y en la de adversario hay un nivel de uno y el siguiente es de su rival.

**21. La mayor parte de los programas para juegos no almacenan los resultados de las búsquedas de un movimiento para el siguiente movimiento, ya que normalmente empiezan de cero cada vez que es el turno del ordenador. ¿Por qué?**

Porque nuestro rival también ha realizado un movimiento tras el nuestro y esto produce dos cambios en el tablero, por tanto tendríamos que guardar más datos sobre estos cambios y tampoco son de gran utilidad si hemos utilizado para la implementación de nuestra inteligencia una poda de alfa-beta que se introduce en los movimientos de los jugadores de muchas posibilidades.

**22. ¿Por qué los procedimientos de búsqueda en juegos parten de la posición inicial en vez de empezar por la posición objetivo y realizar la búsqueda hacia atrás?**

Porque en medio de nuestros movimientos se realizan movimientos del rival y por tanto pueden desvarar los movimientos que nosotros tomemos llevándonos a una derrota sencilla.

**23. El procedimiento minimax ¿es un procedimiento primero en profundidad o en anchura?**

Primero es en profundidad ya que busca el primer nodo para poder iniciar la comparación por anchura.

**24. ¿Por qué podemos decir que la poda alfa-beta es un algoritmo mejor que el procedimiento minimax para la resolución de juegos?**

Es mejor porque cuando encontramos ramas del árbol que nos llevan a victoria se produce un gran número de podas en el árbol siendo un algoritmo que en buenas condiciones será mucho más rápido.

**Búsqueda en un espacio de estados**

**25. ¿Requiere percibir un agente deliberativo mientras construye el plan? ¿y cuando lo ejecuta?**

-

**26. ¿Qué diferencia hay entre una arquitectura de control retroactiva y una búsqueda en profundidad sobre grafos?**

Que en la búsqueda retroactiva mantenemos una lista de todos nuestros movimientos y vamos comprobando todos los estados hasta llegar al nodo que nos da el resultado que deseamos.

**27. ¿Qué condiciones debe verificar un problema para que se pueda descomponer? ¿y para que sea resuelto de forma descomponible?**

-Un problema es descomponible si el problema se puede descomponer en problemas que se resuelvan de la misma forma y sean más pequeños que el problema inicial.

**28. ¿Qué es un grafo Y/O y para que se utiliza sobre problemas descomponibles?**

-

**29. ¿Utilizarías un método de escalada para problemas que requieran obtener una secuencia de acciones? Razona la respuesta.**

**30. ¿Cómo implementarías una estrategia retroactiva para problemas en los que dispones de una función heurística?**

**31. ¿Qué es una función heurística? ¿Para qué se usa?**

**32. ¿Qué métodos heurísticos tienen garantía de conseguir la solución óptima? ¿Bajo que condiciones?**

**33. ¿Por qué los métodos de escalada utilizan criterios aleatorios en algunas decisiones?**

**34. ¿Qué interpretación tiene la función  $f$  en el algoritmo  $A^*$ ?**

**35. ¿Hay siempre garantía de que el algoritmo  $A^*$  encuentra solución si ésta existe?**

**36. ¿Encuentra  $A^*$  siempre la solución óptima? Razona la respuesta**

**37. ¿En qué consiste la búsqueda dirigida?**

**38. ¿Qué interés tiene la versión paramétrica del algoritmo  $A^*$  con función  $f(n)=g(n)+wh(n)$ ?**

**39. ¿En qué contexto utilizarías la arquitectura de percepción/planificación/actuación?**

**40. Describe la búsqueda jerárquica y da un ejemplo de su uso.**

41. Describe la búsqueda orientada por subobjetivos y da un ejemplo de su uso.
42. ¿En qué modelo real se basa el algoritmo de enfriamiento simulado? ¿Qué propiedades de ese modelo son útiles para los métodos de escalada?
43. ¿Cómo se mide la adecuación con el entorno en un algoritmo genético?
44. ¿Qué caracteriza a los algoritmos genéticos como métodos de escalada?
45. Describe en detalle bajo qué condiciones el algoritmo de enfriamiento simulado decide cambiar a un estado peor.
46. Describe como se realiza una mutación en un cromosoma en un algoritmo genético.
47. Describe como se realiza un cruce entre dos nodos en un algoritmo genético.
48. Describe cómo se realiza la selección en un algoritmo genético.
49. En un algoritmo de búsqueda "primero el mejor" ¿que nodos se incluirán en la lista CERRADOS?

#### **Búsqueda con adversario y juegos**

50. ¿Por qué la estructura básica para representar un juego es un árbol y no un grafo?

Porque en un árbol las soluciones son las hojas y solo hay un camino para llegar a una hoja (desde la raíz), mientras que en los grafos para llegar a un nodo podríamos ir desde varios caminos.

51. ¿Qué relación hay entre el árbol del juego y un árbol Y/O?

-

52. ¿Por qué en juegos se llama a la función heurística función de evaluación estática?

Porque la función heurística se encarga de valorar un tablero o un nodo de un árbol por tanto no se evalúan los movimientos que nos han llevado hasta él, sino la posición o valoración de como se encuentra la partida ahora mismo.

53. ¿Cuáles son los cuatro elementos que definen un algoritmo sobre juegos para obtener la mejor jugada inmediata?

54. ¿Por qué en el algoritmo Minimax es conveniente utilizar estrategias de búsqueda retroactivas para la búsqueda parcial?

55. ¿Por qué si aplicamos directamente la función de evaluación estática a los sucesores de un nodo sobre los que tenemos que decidir cuál es la mejor jugada el modelo no funcionará bien?

56. ¿Cuáles son las dos heurísticas en la que se basa la regla minimax?

57. ¿Qué es el factor de ramificación y cómo afecta a la complejidad de un juego?

58. ¿Cómo se mide la complejidad de un algoritmo que obtiene la mejor jugada para un juego?

59. ¿Por qué el algoritmo Minimax no es el algoritmo estándar para resolver juegos?

60. A grandes rasgos, ¿cuál es el análisis de complejidad de la poda alfa beta?

61. ¿Sería muy complejo adaptar el modelo de resolución de juegos al caso de juegos que no sean bipersonales?

62. ¿Cómo se puede resolver un juego en el que hay incertidumbre?

63. ¿Qué es el efecto horizonte? ¿Tiene solución?
64. ¿Podría utilizarse el algoritmo STATUS para resolver el juego del ajedrez? ¿Por qué?
65. ¿Qué tipo de juegos podría resolver el algoritmo STATUS?
66. Describa la diferencia entre resolver un juego y encontrar la mejor jugada inmediata
67. Describe las ideas básicas de la heurística propuesta en el juego de las damas de Samuel
68. Describe las ideas básicas de la heurística propuesta por Turing para el juego del ajedrez

#### **Búsqueda en un espacio de estados**

69. Relacione la idea de trabajar con soluciones parciales frente a soluciones completas con el uso de métodos de escalada.
70. ¿Puede utilizarse una estrategia irrevocable para problemas que impliquen encontrar un camino? Razonar la respuesta.
71. ¿Qué aporta el uso de aleatoriedad en los métodos de escalada?
72. ¿En qué se diferencia el método de escalada estocástico del método de escalada de primera opción? ¿Qué relación tienen estos dos métodos con los otros métodos de escalada no estocásticos?
73. ¿Qué es un programa de enfriamiento? ¿En qué algoritmos se usa? ¿Qué influencia tiene en la solución del problema?
74. Los algoritmos de búsqueda por el mejor nodo se pueden aplicar sobre representaciones de un problema en forma de grafo o de árbol. Analice las ventajas e inconvenientes de ambas sobre el algoritmos A\*.
75. Si ejecutando el algoritmo A\* paramos devolviendo una solución en el momento en el que un nodo objetivo entra en ABIERTOS, ¿qué puede ocurrir? propón un ejemplo que lo ilustre.