

Preguntas prueba 1 Inteligencia Artificial

Pregunta 1: El concepto de agente. Agentes Inteligentes vs Agentes Racionales. Arquitecturas de Agentes.

Un **agente** es cualquier cosa capaz de percibir su medioambiente con la ayuda de sensores y actuar en ese medio utilizando actuadores. El término percepción se utiliza en este contexto para indicar que el agente puede recibir entradas en cualquier instante. En general, un agente tomará una decisión en un momento dado dependiendo de la secuencia completa de percepciones hasta ese instante.

Un **agente racional** es aquel que hace lo correcto; en términos conceptuales, cada elemento de la tabla que define la función del agente se tendría que rellenar correctamente. En cada posible secuencia de percepciones, un agente racional deberá emprender aquella acción que supuestamente maximice su medida de rendimiento, basándose en las evidencias aportadas por la secuencia de percepciones y en el conocimiento que el agente mantiene almacenado.

Un agente racional es aquel que actúa con la intención de alcanzar el mejor resultado o, cuando hay incertidumbre, el mejor resultado esperado.

Un **agente inteligente**, es una entidad capaz de percibir su entorno, procesar tales percepciones y responder o actuar en su entorno de manera racional, es decir, de manera correcta y tendiendo a maximizar un resultado esperado, y un agente racional es un agente que tiene preferencias claras, en los modelos con incertidumbre su objetivo es maximizar los valores esperados, y siempre elige para llevar a cabo la acción con el resultado esperado óptimo por sí mismo, de entre todas las acciones posibles.

Existen diferentes **arquitecturas** de agentes:

- Sistemas de Producción:

1. Se selecciona la primera regla y se comprueba si se cumple su condición. En caso contrario, se continúa con la siguiente hasta que se encuentre una regla con condición con valor 1.
2. La acción de la primera regla encontrada cuya condición sea 1 es la que se ejecuta. Su acción puede ser:
 - La ejecución de una o varias acciones primitivas.
 - Una llamada a otro sistema de producción.
3. Acción por defecto: La última regla de producción del sistema debe ser del tipo $1 \rightarrow A$, para ejecutar una acción en caso de que ninguna de las reglas anteriores cumpla su condición de ejecución.

- Red Neuronal: red de unidades lógicas con umbral.

- Arquitectura de subsunción: consiste en agrupar módulos de comportamiento. Cada módulo de comportamiento tiene una acción asociada, recibe la percepción directamente y comprueba una condición. Si esta se cumple, el módulo devuelve la acción a realizar.

Un módulo se puede subsumir en otro. Si el módulo superior del esquema se cumple, se ejecuta este en lugar de los módulos inferiores.

Pregunta 2: Diferencias y similitudes entre Agentes Reactivos y Deliberativos.

Un **Agente Reactivo**, se basa principalmente en la regla de acción-reacción, percibe su entorno a través de sensores de los cuales procesa la información recibida y hace una representación interna de la misma. Escoge así, una acción posible considerando esa información y en respuesta manda señales de acción para los actuadores y la realizan. En este caso existe un gran papel por parte del diseñador para desarrollar a cada entrada sensorial una acción propia en cada momento, para anticiparse a todas las posibles situaciones existentes. Se diseñan completamente y por tanto es necesario anticipar todas las posibles reacciones para todas las situaciones.

Por su parte un **Agente Deliberativo** es un tipo de agente más avanzado, ya que es capaz de disponer de un modelo del mundo en el que se sitúa, además de un modelo de efectos de sus acciones sobre el mundo, y la posibilidad de razonar sobre esos modelos para decidir que hacer a la hora de conseguir un objetivo. Lo que caracteriza a los agentes deliberativos es la forma de conseguir sus objetivos, por medio de la llamada búsqueda en el espacio de estados, que supone un análisis de las distintas acciones para llegar de un estado inicial a uno final u objetivo. A la secuencia de acciones que lleva al agente desde un estado inicial a un estado destino se denomina plan, y a su búsqueda planificación.

Diferencias:

Agentes Reactivos	Agentes Deliberativos
Realizan pocos cálculos	Realizan muchos cálculos
Reacciona a eventos que ocurren en el ambiente sin un razonamiento complejo	Decisiones por medio de un razonamiento lógico, desarrolla planes
Todas las secuencias de acciones están almacenadas en memoria	Las secuencias de acciones son creadas al realizar la búsqueda.

Similitudes:

Agentes Reactivos y Agentes Deliberativos
Ambos pueden tener en cuenta sus acciones anteriores a la hora de decidir la siguiente acción (memoria)
A la hora de decidir una acción, se tendrá en cuenta el estado actual del agente (sensores y actuadores)

Pregunta 3: El concepto de Heurística. Importancia de la Heurística en I.A.

Las Heurísticas son criterios, métodos, o principios para decidir cual, entre una serie de cauces alternativos de acción, promete ser más efectivo a la hora de lograr alguna meta. Esto representa un compromiso entre dos exigencias: criterios simples y, al mismo tiempo, ser capaces de discriminar entre buenas y malas opciones.

La construcción de funciones heurísticas puede considerarse un proceso de descubrimiento. Es muy difícil articular el mecanismo por el cual se llega a estas funciones. En general: las heurísticas se descubren a partir de modelos simplificados del dominio del problema.

Los valores de la función heurística son usados para determinar que operación ejecutar a continuación.

Un inconveniente de los métodos heurísticos es que en ocasiones no es posible conocer la calidad de la solución, cuán cerca está el óptimo(x^* de la solución heurística encontrada (x_{heu}). (Si el problema es de maximización lo único que sabemos es que $(x_{heu}) \leq x^*$

IMPORTANCIA

En IA basada muy fuertemente en búsqueda, los métodos heurísticos han sido empleados desde el principio. Una heurística encapsula el conocimiento específico que se tiene sobre un problema, y sirve de guía para que un algoritmo de búsqueda pueda encontrar una solución. En general no garantiza la solución óptima, pero que en media produce resultados satisfactorios en la resolución de un problema. Eventualmente, puede devolver soluciones óptimas bajo ciertas condiciones (requiere demostración).

Se usa en problemas de recorridos (mapas), en juegos, como el ajedrez o puzzles.

Pregunta 4: Características esenciales de los Métodos de Escalada.

Los Métodos de escalada buscan en un entorno local del nodo en curso. (Modificación de la búsqueda en profundidad). Una búsqueda local consiste en seleccionar la solución mejor en el vecindario de una solución inicial, e ir viajando por las soluciones del espacio hasta encontrar un óptimo (local o global).

Los distintos métodos de búsqueda local se diferencian en como determinar la “vecindad” de cada nodo en estudio.

Algoritmo de escalada simple

- Considera que la vecindad de un nodo es un conjunto de hijos obtenidos secuencialmente hasta que aparece uno mejor que el padre (según la función de evaluación).
- Se parte de un nodo inicial.
- Para cada nodo en curso se obtiene la vecindad según el criterio anterior.
- Si no se obtiene un hijo mejor que el padre stop.
- Cuando se obtiene un hijo mejor que el padre el hijo pasa a ser el nodo en curso y el proceso continua.

Algoritmo de escalada por la máxima pendiente (hill climbing)

- Considera que la vecindad de un nodo es el conjunto de todos sus hijos obtenidos secuencialmente.
- Se parte de un nodo inicial.
- Para cada nodo en curso se obtiene la vecindad según el criterio anterior.
- Se selecciona el hijo mejor que pasa a ser el nodo en curso y el proceso continua.
- Si no se obtiene un hijo mejor que el padre stop.
- Completitud: no tiene porque encontrar la solución
- Admisibilidad: no siendo completo, aun menos será admisible
- Eficiencia: rápido y útil si la función es monótona (de)creciente

Algunas variaciones estocásticas

- Escalada estocástica: Escoge aleatoriamente entre los sucesores con mejor valoración que el estado actual.
- Escalada de primera opción: Se generan aleatoriamente sucesores, escogiendo el primero con mejor valoración que el estado actual
- Escalada con reinicio aleatorio: Se repite varias veces la búsqueda, partiendo cada vez de un estado inicial distinto, generado aleatoriamente.
 - “si no te sale a la primera, inténtalo otra vez”
 - Si la probabilidad de éxito de una búsqueda individual es p , entonces el número esperado de reinicios es $1/p$

Algoritmo de Enfriamiento Simulado

El problema de los métodos de escalada simple y simple por máxima pendiente es que su solución podría no ser un máximo global.

- Un modo de evitar que la búsqueda local finalice en óptimos locales, hecho que suele ocurrir con los algoritmos tradicionales de búsqueda local, es permitir que algunos movimientos sean hacia soluciones peores
- Pero si la búsqueda está avanzando realmente hacia una buena solución, estos movimientos “de escape de óptimos locales” deben realizarse de un modo controlado
- Se basa en principios de Termodinámica. (Analogía entre el proceso de enfriamiento y el algoritmo de enfriamiento simulado)

Ventajas: Al ser un método probabilístico, tiene capacidad para salir de óptimos locales. Es eficiente. Es fácil de implementar.

Inconvenientes: Encontrar la temperatura inicial T_i , el método de actualización de temperatura α , el número de vecinos a generar en cada estado y el número de iteraciones óptimo es una tarea que requiere de muchas pruebas de ensayo y error hasta que ajustamos los parámetros óptimos.

Pregunta 5: Características esenciales del Algoritmo A*.

El algoritmo A* se basa en la búsqueda de primero el mejor, donde su función de evaluación $f(n)$ es la suma de dos componentes:

- $g(n)$: la distancia actual desde el nodo origen hasta el nodo a etiquetar.
- $h(n)$, la distancia estimada desde este nodo a etiquetar hasta el nodo destino.

En cada caso se selecciona el nodo más prometedor que se haya generado hasta ese momento (función heurística). A continuación se expande el nodo elegido generando todos sus sucesores y si alguno de ellos es meta el proceso acaba. Si no continúa con el algoritmo.

Realmente, el algoritmo de A* es una búsqueda en anchura. Si en una rama por la que estoy explorando no aparece la solución, la rama puede parecer menos prometedora que otras por encima de ella y que se habían ignorado. Podemos pues abandonar la rama anterior y explorar la nueva. Sin embargo la vieja rama no se olvida. Su último nodo se almacena en el conjunto de nodos generados pero aún sin expandir.

- Es un algoritmo completo, en caso de existir una solución, siempre dará con ella.
- Si para todo nodo n del grafo se cumple $g(n)=0$, nos encontramos ante una búsqueda voraz. Si para todo nodo n del grafo se cumple $h(n)=0$, A* pasa a ser una búsqueda de coste uniforme no informada.
- Para garantizar la optimalidad del algoritmo, la función $h(n)$ debe ser admisible. De no cumplirse esta condición, el algoritmo pasa a denominarse simplemente A, y a pesar de que sigue siendo completo, no se asegura que el resultado obtenido sea el camino de coste mínimo.
- La complejidad computacional está relacionada con la calidad de la heurística.
- Se necesita mucho espacio para ejecutar éste algoritmo.
- El rendimiento de los algoritmos de búsqueda heurística, depende de la calidad de la función heurística.

Pregunta 6: Describe los elementos característicos de un Algoritmo Genético.

¿Que problemas pueden resolverse mediante un Algoritmo Genético?

Un algoritmo genético es un algoritmo bioinspirado, cuyos elementos característicos son los siguientes:

Inicialización: Se genera aleatoriamente la población inicial, que está constituida por un conjunto de cromosomas los cuales representan las posibles soluciones del problema.

Evaluación: A cada uno de los cromosomas de esta población se le aplicará la función de aptitud para saber qué tan “buena” es la solución que se está codificando. A partir de los mejores se irá generando nueva población.

Condición de termino: El AG se deberá detener cuando alcance la solución óptima, pero ésta generalmente se desconoce, por lo que se deben utilizar otros criterios de parada. Normalmente se usan 2 criterios:

- Correr el algoritmo un número máximo de iteraciones (generaciones).
- Detenerlo cuando no haya cambios en la población.

Mientras no se cumpla la condición de termino, se hace lo siguiente.

Selección: Después de saber la aptitud de cada cromosoma, se procede a elegir los cromosomas que serán cruzados en la siguiente generación. Los cromosomas con mejor aptitud tienen mayor posibilidad de ser seleccionados.

Recombinación: La recombinación es el principal operador genético, representa la reproducción sexual, opera sobre 2 cromosomas a la vez para generar 2 descendientes donde se combinan las características de ambos cromosomas padres.

Mutación: Modifica al azar parte del cromosoma de los individuos, y permite alcanzar zonas del espacio de búsqueda que no estaban cubiertas por los individuos de la población actual.

Reemplazo: Una vez aplicados los operadores genéticos, se seleccionan los mejores individuos para conformar la población de la generación siguiente.

Principalmente, estos algoritmos se usan para resolver problemas de optimización, programación automática, economía, y sistemas sociales (colonias de insectos, sistemas multiagente, etc)