Imprimir

Sesión 4-Tema 3: Búsqueda en juegos y para problemas de satisfacción de restricciones

Tema 3: Búsqueda en juegos y para problemas de satisfacción de restricciones – Resumen

Afrontar juegos como árboles de búsqueda, en los que generamos subárboles hasta terminar la partida, cada estado de la partida es un nodo al que se le aplica una función de evaluación f(n). Objetivo del análisis del árbol, determinar valor raíz, MiniMax.

Tenemos 2 jugadores, el que máximiza la función f(n) y el que la minimiza, 2 oponentes. Cada nivel del árbol es un movimiento de cada jugador, una rama contempla una posible jugada. Cada nodo tiene un factor de ramificación, el nº de movimientos posibles por el jugador (nº de hijos).

La estrategia exhaustiva MiniMax evalúa los nodos hoja, recorre el camino del nodo hoja a la raíz para asignarle un valor, en cada decisión, si l átomo el jugador Min, le asocia el mínimo de los valores y viceversa.

Max, la computadora, calcula el minmax desde el estado actual, intenta hacer una previsión de qué puede elegir Min.

En juegos multijugador (más de 2 jugadores), cada nivel del árbol desde la raíz pertenece a un jugador, si hay 3, cada 3 niveles le toca de nuevo al primero. Sustituye valor del nodo por vector de valores. Según el jugador que toque en el nivel, escoge el que mejor le convenga, si no hay diferencias entre una decisión u otra, elige la que más le convenga a un aliado.

La poda alfa-beta trata de reducir el nº de nodos evaluados, alfa es la mejor opción hasta el momento para max (tiende a bajar), beta para min (tiende a subir).

Alfa <= minimax – V(n) <= beta.

Si alfa >= V(n) -> poda. Si beta <= V(n) -> poda.

Técnicas complementarias:

- Movimientos de libros: Imposible etiquetar todos los estados. Etiquetar aperturas y cierres (Estados concretos). V(n) = conocimiento + búsqueda.
- Espera del reposo: Para evitar picos en V(n), explora un nivel más después de ese pico.
- Técnica de bajada progresiva: Restricciones en el tiempo (antecesores).
- Poda heurística: Reordenamos los nodos, más prometedor a la izquierda del todo, buscamos más a su alrededor. Reducimos factor de ramificación cuanto más lejos del nodo prometedor.
- Continuación heurística: Hacemos una selección con los nodos más prometedores y los exploramos.

Los problemas con satisfacción de restricciones (CSP) están definidos por un conjunto de Variables definidos sobre Dominios (cada variable tiene su dominio) y un conjunto de restricciones que se pueden aplicar a cualquier variable.

Estas restricciones, normalmente se conocen cómo ecuaciones, o en este caso, un sistema de ecuaciones. Estos problemas se resuelven cuando encuentras una solución que asigne valor a las variables que satisfagan las restricciones.

Los **CSP binarios** son cuando una restricción sólo se puede aplicar a 2 variables, algunos ejemplos pueden ser los siguientes:

- Coloreado de mapas: dos mapas frontera no pueden tener el mismo color.
- Generación crucigramas: Construir un crucigrama legal donde la intersección de 2 slots esté una letra que cuadre para ambas palabras.
- N-reinas: Posicionar n reinas en un tablero de nxn de forma que no se puedan atacar.
- Criptoaritmética: Sustituir cada letra por un dígito distinto (distinta cifra, distinta letra) de manera que la suma sea correcta.
- Restricciones temporales: Encontrar asignación temporal consistente a un conjunto de sucesos que ocurren en el tiempo.

Métodos de resolución (encuentran solución o demuestran que no existe):

- Generación y test: prueba y error (explora el espacio entero)

- Backtacking: solución de forma gradual, instancia variables compatibles en el orden definido.
- Backjumping: Retroceso a la variable que está en conflicto con la actual, no a la instanciada anterior.
- Forward checking: Comprueba hacia el futuro desde la situación actual. Vk mira desde el dominio Dk+1 hasta Dn, elimina variables futuras que sean inconsistentes con valor actual. Si un dominio se queda sin variables, k-1 y repite con Vk. Cuando k = 1, sale sin solución. Al llegar k = n, sale con solución

look-ahead (forwarding)

Índice WIki individual