

2022

29 / 09 / 2022

Resumen sesión teoría 29/09

SISTEMAS INTELIGENTES

ADRIAN UBEDA TOUATI 50771466R

Contenido

Ciclo de control básico	2
Hay 2 tipos de estrategias:	2
-Irrevocables.....	2
Ejemplo:.....	2
Desventajas:	2
-Tentativas:	2
Ejemplos:	2
Algoritmos A (aditivos)	3
Tener cuidado con encontrar una ley heurística:.....	3
Inconvenientes de mantener la admisibilidad	4
Relajación de la restricción de optimalidad	4

Ciclo de control básico

Dentro de una estrategia de control, se deben seguir 4 pasos:

- Exploración de la frontera
- Cálculo de reglas aplicadas
- Resolución de conflictos
- Aplicación de regla y memorización de estado

Es importante para la estrategia elegida, que tenga un avance metódico y que no tenga bucles

Hay 2 tipos de estrategias:

-Irrevocables: No permiten la vuelta atrás, por lo que permiten algoritmos voraces

Ejemplo: Descenso por gradiente:

- Conocimiento local, pero las equivocaciones alargan la búsqueda
- Importante especificar una evaluación que proponga un mínimo máximo en el estado final

Desventajas:

- Mestas
- Máximos locales
- Crestas

-Tentativas: Multi o mono camino, posibilidad de vuelta atrás

No informadas, son ciegas, no dependen de la naturaleza de la solución, pero rápidas en pequeños problemas

Ejemplos:

Búsqueda profunda: backtracking, el siguiente estado es el de mayor profundidad en el grafo

Búsqueda en anchura: Mayor prioridad a menor profundidad

Coste uniforme: Similar al de anchura de cada regla cuando el coste de sea unitario

Informadas: Informan de lo prometedor que es un nodo

Algoritmos A (aditivos)

$$f(n) = g(n) + h(n)$$

$g(n)$: Coste mínimo desde el estado inicial hasta en nodo n

$h(n)$: Coste del camino mínimo desde n hasta objetivo, se utilizan conocimientos heurísticos

Búsqueda heurística = dificultad, dominar, optimizar, admisibilidad

$$F(n) = C$$

C : Coste mínimo desde el estado inicial hasta en nodo solución

Función heurística admisible: garantiza un camino de coste mínimo hasta un objetivo
 $h(n) \leq h(n) \forall n$

Elegir bien $h(n)$ = menos búsquedas

Tener cuidado con encontrar una ley heurística:

- Puede quedar fuera de la solución optima
- Que se acercarse a la heurística optimo

Problemas de camino mínimo

Implementar algoritmo A^* y buscaremos el camino optimo a la distancia de Manhattan

Coste actual optimo:

$$g(x, y) = |x| + |y|$$

Heurística admisible:

$$h_1(x, y) = \text{sqrt}((m - x)^2 + (n - y)^2)$$

Heurística óptima:

$$h(x, y) = |m - x| + |n - y|$$

Inconvenientes de mantener la admisibilidad

Mucho tiempo

No es práctico para problemas grandes

La heurística encontrará antes la solución, pero mayor coste computacional

Relajación de la restricción de optimalidad

Se puede tener una solución con menos tiempo, y al mismo tiempo obtener solución óptima:

- Técnica de ajustes de peso: Usar una función $f()$ ponderada

- Técnica de la admisibilidad: Ganar tiempo, pero solución subóptima.

- Algoritmo de estimación de coste de búsqueda: Cambiamos ListaFocal a una sublista de ListaFrontera, que contiene solo nodos de menos al mejor valor de ListaFrontera gracias a un factor.

Y comparamos los algoritmos: La ponderación dinámica es mas sencilla, pero limitada a problemas cuya profundidad conocemos.