## SISTEMAS DE PRODUCCIÓN

Los sistemas de producción están compuestos por tres partes

* Definición de la BDG (Base de datos global)
* Conjunto de reglas de producción
* Estrategia de control

### BDG (Base de datos global)

Es una forma de representar el estado. La BDG está formada por un conjunto de variables que podrán tomar cada uno de los valores que permite el problema a representar.

Una vez definida la BDG, se define la BDGi (inicial) y BDGf. (final). Son dos estados posibles de la BDG.

BDGi= Estado inicial del problema

BDGf = Estado al que se quiere llegar.

En el ejemplo del rompecabezas que consiste en ordenar las fichas sin levantarlas del tablero, la BDG será la matriz de enteros.

Se define la BDG como una matriz de 3\*3, donde cada valor de entero deberá estar entre 0 y 8 (El cero indica posición vacía).

Estado inicial del problema

|  |  |  |
| --- | --- | --- |
| 7 | 8 | 1 |
| 2 | 5 | 3 |
| 4 | 6 | 0 |

Estado final del problema

|  |  |  |
| --- | --- | --- |
| 1 | 2 | 3 |
| 4 | 5 | 6 |
| 7 | 8 | 0 |

### Conjunto de Reglas de producción

Las reglas son acciones (un movimiento que se hace que tiene como consecuencia un cambio de estado). Por ejemplo, en el caso del juego de ordenar las fichas sin levantarlas las acciones (reglas) serían cuatro: 🡨, 🡪, 🡩, 🡫 pero no se puede mover 🡪 si se está en la última columna.

Se parte de un estado inicial y se va pasando por otros estados hasta llegar al estado final (BDG final). Entonces se debe poder pasar de un estado a otro estado. Las reglas de este conjunto son las que permiten realizar este cambio de estado.

¿Cómo se define una regla?

Una regla de producción es un par acción- Condición, que tiene la forma general: if CONDICION then ACCION.

Entonces se debe poder definir el conjunto de acciones que se puede realizar si una condición es TRUE (cada acción definida tiene ciertas condiciones que cumplir).

### Estrategia de control

Va a indicar que reglas se van a elegir y que orden se van a aplicar. Sería un algoritmo que va a armar el grafo. Las estrategias son independientes del problema.

#### Tipos de Grafos

##### Grafo Implícito

Al aplicar todas las reglas posibles a todos los nodos, se formó un enorme grafo, y a este se lo llama grafo implícito.

##### Grafo Explicito

Es aquel grafo que realmente se arma al aplicar la estrategia de control al grafo implícito. Es un subconjunto del grafo Implícito.

Condiciones para que sea una estrategia de control sea válida:

1. No puede tener reglas al azar, se tiene que fijar en un criterio.
2. No puede aplicar siempre la misma regla, no sirve como sistema de producción.

Estrategias Estándar o generales: Si bien se pueden generar las estrategias que se quiera, ya existen algunas predefinidas.

La BDG y las reglas son dependientes del problema, sin embargo, la estrategia no lo es.

#### Caracterización de las Estrategias

##### Admisibles

Cuando encuentra primero la mejor solución (la de menor costo). Encuentra la mejor solución, pero el tiempo de búsqueda puede que no sea el mejor, el tiempo dependerá de donde se encuentre ubicada la solución.

##### No admisible

La solución que se encuentre no necesariamente va a ser la mejor.

Toda solución tiene un costo, entonces si se supone que el costo es constante por cada cambio de estado, entonces va a ser la que esté más cerca de la raíz (la más corta).

En este ejemplo la solución óptima es la que recorre el camino de color negro.

##### Informados

Dependen de que se tenga información acerca del problema.

##### No informados

No tienen información del problema.

##### Tentativos

Guardan información del camino hecho para buscar la solución.

##### Irrevocables

No guardan información del camino hecho para buscar la solución.

#### Estrategias estándar

Estas ya están definidas.

Las dos primeras son los clásicos recorridos de grafos, el resto es solo para sistemas de producción,

##### PA - Primero a lo ancho (viene de la teoría de grafos)

Dado un estado se aplican todas las reglas posibles y se generan todos los posibles estados, si llega al final termina, sino se aplica el mismo criterio a cada nodo. O sea, se recorre el grafo por niveles.

Es Eficaz. Llega a la solución y es la mejor.

No es Eficiente. Emplea muchos recursos computacionales.

La solución que encuentra es la más corta (la mejor).

Admisible: SI

Dado un estado se aplican todas las reglas aplicables (las que la condición da verdadero), se aplican todas en el orden en que están definidas. Por ejemplo, si se tienen 3 reglas aplicables, se van a tener 3 posibles estados resultantes.

Si alguno de estos nodos es el estado final llegué a la solución, sino sigo aplicando todas las reglas aplicables.

Vuelvo a aplicar todas las reglas aplicables.

Se va armando el árbol completo (Implícito), el proceso termina cuando encuentre la solución, pero puede pasar que no la encuentre es por eso que se debe fijar un corte que podría ser por ejemplo por profundidad de niveles.

Esta estrategia recibe el nombre de primero a lo ancho porque barre a lo ancho por niveles.

##### PP (Primero en Profundidad)

Dado un estado inicial, se toma la primera regla y se aplica. Siempre se aplica la primera y así hasta que no se pueda seguir, en este caso se retrocede al anterior y se toma la siguiente regla.

En este método tiene una condición, se debe fijar un nivel máximo de profundidad, porque puede ocurrir que se descienda mucho y no se pueda terminar nunca. Se puede caer en aplicar siempre la primera regla y nunca terminar.

Desventajas: No siempre la primera solución que encuentre es la mejor.

Eficiencia: Es mejor que el anterior porque no armas todo el grafo.

Prolog hace PP.

Se parte del primer estado y se aplica la primera regla, si es la condición de fin termina, sino se sigue.

Si es la condición de fin paro, sino sigo.

Se Vuelve a aplicar la primera regla y se genera otro nodo más, se aplica la primera regla hasta que no tener más reglas aplicables, porque se llegó a un nodo hoja. Entonces se sube un nivel y se vuelve a aplicar la primera.

En este método se fija un nivel máximo de profundidad porque si no sigue bajando y aplicando la primera regla, porque puede que no encuentre nunca la solución como es el caso de:

Solución

Sigue bajando

Entonces sigue bajando. En esta estrategia no puedo asegurar de que sea la mejor solución.

Solución 112

Nivel de Profundidad

Solución

Esta estrategia será más eficiente porque arma una parte del árbol, porque la solución seguramente (porque depende de donde se encuentre localizada la solución en el árbol) la va a encontrar rápido.

Si la solución se encuentra aquí, tardaría aproximadamente lo mismo que PA.

##### HC (Subiendo montañas)

A diferencia de los anteriores acá se elige la mejor regla, la que permita acercarse más a la solución que se está buscando. Para eso se define una función que indica que tanto acerca a la solución cada opción (regla que genera un siguiente estado).

Nuevamente aplico las reglas aplicables que cumplen con la condición.

Entonces se aplica esa función a cada estado.

Elijo la regla que más acerque a la solución. ¿Cómo se sabe cuánto acerca una regla a la solución? Se necesita una función de evaluación, entonces se usan heurísticas, que son funciones basadas en la experiencia. Se Toman las n posibles reglas se les aplica la función de evaluación y se elige la que me deje un valor máximo en la función de evaluación. Si la función de evaluación es mala puede ocurrir que nunca se llegue a la solución.

El método va a ser tan bueno como buena sea la función de evaluación.

La eficiencia es muy buena, se va por un camino directo

No es admisible, puedo no llegar nunca a la solución.

La función depende del problema.

Este es informado e irrevocable (me juego por lo que dice la función y no puedo volver para atrás si me equivoque).

Se llama así porque voy subiendo, buscando el sendero más cerca del destino al que quiere llegar (como lo haría un escalador).

##### A\* (A estrella)

Es una mejora de PP, acá se elige la regla que determina la función, a diferencia del anterior si falla en un camino se intenta con otro.

Es tentativo, e informado. Es informado porque usa una función.

La función f está formada por dos funciones (g y h: dan los costos).

f = g + h

f: función de evaluación.

g: parte determinística. Costo desde el nodo inicial hasta el nodo actual. (Lo ya recorrido)

h: Costo del nodo actual hasta la solución o estado final. (Es lo que falta por recorrer)

Al principio toma importancia la función heurística, cuando se está llegando a la solución la parte determinística toma más importancia, si falla sube y sigue por otra alternativa.

Se puede demostrar que es admisible, bajo determinadas condiciones. Es el más difícil de implementar.

##### DI (Descensos Iterativos)

PP

PP

PP

PP

Dn

Es una combinación de PA y PP. Aplicando sucesivamente PP con distintos niveles. La idea es aplicar PP y sino encuentra la solución se vuelve a aplicar PP, y así sucesivamente. Se va descendiendo e incrementando el nivel de profundidad.

En el caso de PP, no sabía si encontraba la mejor solución, y era fácil de implementar.

No es admisible, pero está más cerca que PP de la mejor.

Dn: 1 🡪 Degenera en PA. Es admisible.

Dn: ∞ 🡪Si el delta es muy grande se va a estar muy lejos de ser la mejor solución.

1< Dn <∞ Si se elige bien el delta se tiene una alta probabilidad de encontrar la mejor solución.

Es fácil de implementar.