

Inteligencia Artificial I

Ejercitación: CSP

1. Formulación CSP para el Sudoku

Variables:

X_{ij} donde $0 \leq i < 9$ y $0 \leq j < 9$ (Matriz de 9x9)

Dominios:

$D_{ij} = \{1,2,3,4,5,6,7,8,9\}$

Restricciones:

- No puede haber números repetidos en la misma fila

$X_{i0} \neq X_{i1} \neq X_{i2} \neq X_{i3} \neq X_{i4} \neq X_{i5} \neq X_{i6} \neq X_{i7} \neq X_{i8}$ para $0 \leq i < 9$

- No puede haber números repetidos en la misma columna

$X_{0j} \neq X_{1j} \neq X_{2j} \neq X_{3j} \neq X_{4j} \neq X_{5j} \neq X_{6j} \neq X_{7j} \neq X_{8j}$ para $0 \leq j < 9$

- No puede haber números repetidos en cada una de las 9 sub-matrices de 3x3

2. Demostración de inconsistencia para la asignación parcial {WA=red, V=blue}

Estado inicial:

WA	NT	Q	NSW	V	SA	T
R G B	R G B	R G B	R G B	R G B	R G B	R G B

Se asigna el color rojo a WA:

WA	NT	Q	NSW	V	SA	T
R	R G B	R G B	R G B	R G B	R G B	R G B

Se ejecuta el algoritmo AC-3 para verificar la consistencia de arco

- Chequeo consistencia de NT a W -> elimino R de NT
- Chequeo consistencia de SA a W -> elimino R de SA

No hay más inconsistencias, se devuelven los dominios actualizados:

WA	NT	Q	NSW	V	SA	T
R	G B	R G B	R G B	R G B	G B	R G B

Se asigna el color azul a V:

WA	NT	Q	NSW	V	SA	T
R	G B	R G B	R G B	B	G B	R G B

Se ejecuta el algoritmo AC-3 para verificar la consistencia de arco

- Chequeo consistencia de SA a V -> elimino B de SA
- Chequeo consistencia de NSW a V -> elimino B de NSW
- Chequeo consistencia NT-SA -> elimino G de NT
- Chequeo consistencia Q-SA -> elimino G de Q
- Chequeo consistencia NSW-SA -> elimino G de NSW
- Chequeo consistencia Q-NSW -> elimino R de Q
- Chequeo consistencia Q-NT -> elimino B de Q

No hay más inconsistencias, se devuelven los dominios actualizados:

WA	NT	Q	NSW	V	SA	T
R	B		R	B	G	R G B

El dominio de Q quedó vacío, por lo tanto, el CSP es inconsistente.

**En el algoritmo AC-3 se chequean las consistencias de todos los arcos, incluí solo aquellos en los que se hacen eliminaciones.

3. Complejidad AC-3 para un árbol estructurado

La complejidad en el peor caso cuando se ejecuta AC-3 en un árbol estructurado CSP es $O(nd^2)$. Dado que es un árbol, para n nodos (variables) se tienen $n-1$ arcos, cada arco (X_k, X_i) puede insertarse en la cola sólo d veces, porque X_i tiene como máximo d valores para suprimir; la comprobación de la consistencia de un arco puede hacerse en $O(d)$ veces; entonces el tiempo total, en el peor caso, es $O(nd^2)$.

4. Correctitud del algoritmo CSP para árboles estructurados

Si ordenamos las variables desde la raíz a las hojas de tal modo que el padre de cada nodo en el árbol precede a su hijo y etiquetamos a las variables desde X_1 a X_n , aplicando la comprobación de consistencia de arco para j desde n hasta 2 , al arco (X_i, X_j) , donde X_i es el padre de X_j , quitando los valores del DOMINIO[X_i] que sea necesario, el algoritmo asegura que cualquier valor suprimido no puede poner en peligro la consistencia de arcos que ya han sido tratados. Después de este paso el CSP es directamente arco-consistente, entonces la asignación de valores en él no requiere ninguna vuelta atrás. Esto nos asegura que también sea n -consistente.

En un árbol la 2-consistencia o consistencia de arco implica n -consistencia ya que cualquiera 2 variables están relacionadas por a lo sumo 1 camino y cada variable perteneciente a ese camino tiene un solo padre, por lo que es suficiente que cada variable y su padre cumplan con la consistencia de arco para que todo el camino y por lo tanto todo el árbol sea consistente.