Programación de Restricciones

Carlos Linares López

Grupo de Planificación y Aprendizaje (PLG)
Departamento de Informática
Escuela Politécnica Superior
Universidad Carlos III de Madrid

21 de noviembre de 2013

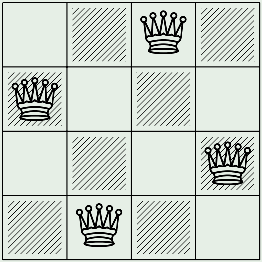
Definiciones

- Una red de restricciones $R = (X, D, C)$ consiste en un conjunto finito de valores $X = \{x_i\}_{i=1}^n$ definidas sobre dominios $D = \{D_i\}_{i=1}^n$, que contienen los posibles valores de cada variable $D_i = \{v_1^{(i}, v_2^{(i}, \ldots, v_k^{(i}\}$, y un conjunto de restricciones $C = \{C_i\}_{i=1}^t$
- Una restricción C_i es una relación R_i definida sobre un conjunto de variables S_i, $S_i \subseteq X$, que denota las asignaciones legales simultáneas

Ejemplo I

Problema de las N Reinas

Modelizar las restricciones del juego de las 4 Reinas, en el que deben disponerse 4 reinas sobre un tablero de ajedrez 4×4 de modo que ninguna está atacada por otra como se muestra, por ejemplo, a continuación:



Ejemplo II

- Si la restricción R_{ij} representa las posiciones legales que pueden ocupar las reinas en las columnas i y j entonces:

$$R_{12}=\{(1,3),(1,4),(2,4),(3,1),(4,1),(4,2)\}$$
$$R_{13}=\{(1,2),(1,4),(2,1),(2,3),(3,2),(3,4),(4,1),(4,3)\}$$

El resto de restricciones ($R_{14}, R_{23}, R_{24}, R_{34}$) se representan análogamente —y en algunos casos resultan en las mismas tuplas, por ejemplo $R_{23} = R_{12}$

Definiciones

- Las restricciones pueden describirse también de manera booleana o numérica.

Considera la siguiente representación alternativa del problema de las N reinas:

$$\begin{aligned} R_{ij} &= \{(v^{(i}, v^{(j}),\ \text{tal que } v^{(i} \in D_i, v^j \in D_j \\ &\quad v^{(i} \neq v^{(j}, |v^{(i} - v^{(j}| \neq |i-j|\} \end{aligned}$$

- En general, la forma de representar restricciones importa y puede haber formas sucintas de hacerlo.

Definiciones

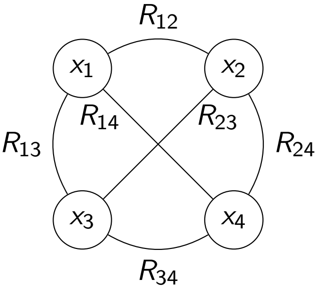
- Una *instanciación* de un subconjunto de variables es una asignación de valores de sus dominios a cada variable del subconjunto
- Una instanciación es *parcial* si afecta a un subconjunto S_i estricto de X

Definición 1

El objetivo de la satisfacción de restricciones consiste en encontrar instanciaciones globales consistentes

Grafo de restricciones

- Una red de restricciones $R = (X, D, C)$ puede representarse con un *grafo de restricciones* donde cada nodo representa una variable y los arcos conectan aquellos nodos cuyas variables están en la restricción



El grafo que se muestra a la izquierda representa en los vértices las variables x_i (que contienen la fila en la que se coloca la reina en la columna i-ésima) y en los arcos las restricciones que las vinculan.

Consistencia

- ¡No todas las instanciaciones parciales consistentes forman parte de la solución! pero las que no lo sean pueden eliminarse
- La segunda observación conduce a la arco consistencia
- La primera se tiene en cuenta en la camino consistencia

Arco consistencia

Definición 2 *Data una red de restricciones* $R = (X, D, C)$*, con* $R_{ij} \in C$*,* x_i *es arco consistente con* x_j *si y sólo si para cada valor* $a_i \in D_i$*, existe un valor* $a_j \in D_j$ *tal que* $(a_i, a_j) \in R_{ij}$

- Si un valor no participa en la solución de una subred de dos variables, entonces puede eliminarse

AC-REVISE (x_i, x_j)
FOR $a_i \in D_i$
IF $\nexists a_j$ tal que $(a_i, a_j) \in R_{ij}$
THEN $D_i = D_i - a_i$

Arco consistencia

- AC-REVISE comprueba la arco consistencia de dos variables
- Para forzarla en toda la red de restricciones R se usa AC-1:

AC-1 (R)
  REPEAT
    FOR (x_i, x_j) tal que $R_{ij} \neq \varnothing$
      AC-REVISE (x_i, x_j)
      AC-REVISE (x_j, x_i)
  UNTIL los dominios no cambian

Ejemplo I

2-coloring con tres vértices

Calcular la arco-consistencia del etiquetado con sólo dos colores (rojo y azul) de un grafo con tres vértices (X, Y y Z) que están conectados con el resto, de modo que dos vértices adyacentes no están etiquetados con el mismo color

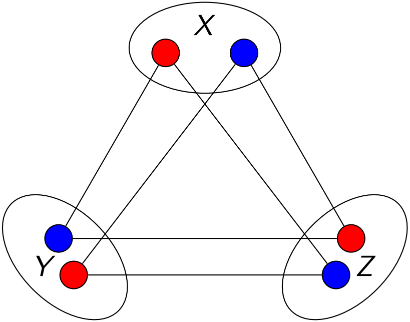
Ejemplo II

Las restricciones son:

$$R_{xy} = \{(r,a),(a,r)\}$$
$$R_{xz} = \{(r,a),(a,r)\}$$
$$R_{yz} = \{(r,a),(a,r)\}$$

donde r y a representan el color rojo y azul respectivamente

Ejemplo III



Como se puede ver en la figura, existe un coloreado para cada vértice que garantiza la arco consistencia. Sin embargo, no existe ninguna solución globalmente consistente

La arco consistencia sirve sólo para encontrar inconsistencias entre nodos adyacentes

Camino consistencia

Definición 3 *Una restricción R_{ij} es camino consistente en relación a la variable x_k si y sólo si para cada par $(a_i, a_j) \in R_{ij}$, existe un valor $a_k \in D_k$ tal que $(a_i, a_k) \in R_{ik}$ y $(a_k, a_j) \in R_{kj}$*

CC-REVISE (x_i, x_j, x_k)
  FOR $(a, b) \in R_{ij}$
    IF $\nexists c \in D_k$ tal que $(a, c) \in R_{ik} \wedge (b, c) \in R_{jk}$
      THEN $R_{ij} = R_{ij} - (a, b)$

Camino consistencia

- El algoritmo CC-REVISE fuerza la arco consistencia de dos variables respecto a una tercera
- Para forzar la arco consistencia en toda la red de restricciones R debe usarse CC-1:

CC-1 (R)
REPEAT
    FOR $k \longleftarrow 1, n$
        FOR $i, j \longleftarrow 1, n$
            CC-REVISE (i, j, k)
UNTIL ninguna restricción R_{ij} cambia

Ejemplo I

2-coloring con tres vértices

Aplicar la camino consistencia al problema de 2-coloring con tres vértices discutido en el ejemplo anterior
¿Existe una instanciación globalmente consistente, o es infactible?

Ejemplo II

Considérese por ejemplo la restricción $R_{xy} = \{(r, a), (a, r)\}$ con respecto a la variable x_z ($D_z = \{r, a\}$):

$R_{xy} = (r, a)$: $x_z = r$ es consistente con x_y pero no con la primera; análogamente $x_z = a$ es consistente con la primera pero no con la segunda. Se elimina (r, a) de R_{xy} puesto que no hay valores de x_z camino consistentes con élla.

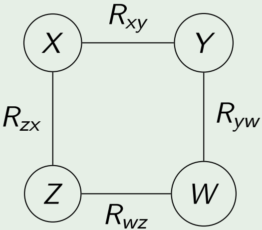
$R_{xy} = (a, r)$: $x_z = a$ es consistente con x_y pero no con la primera; análogamente $x_z = r$ es consistente con la primera pero no con la segunda. Como antes, se elimina (a, r) de R_{xy}

y como $R_{xy} = \emptyset$ no existen valores legales simultáneos para los nodos X e Y, detectando así la infactibilidad

Ejemplo III

2-coloring con cuatro vértices

Calcular la camino consistencia del etiquetado con sólo dos colores (rojo y azul) de un grafo con cuatro vértices (X, Y, Z y W) dispuesto como se muestra, de modo que dos vértices adyacentes no están etiquetados con el mismo color



Ejemplo IV

Las restricciones son:

$$\begin{aligned} R_{xy} &= \{(r, a), (a, r)\} \\ R_{yw} &= \{(r, a), (a, r)\} \\ R_{wz} &= \{(r, a), (a, r)\} \\ R_{zx} &= \{(r, a), (a, r)\} \end{aligned}$$

donde r y a representan el color rojo y azul respectivamente

Ejemplo V

Considerando la restricción $R_{xy} = (r, a)$ con respecto a:

- x_z : $x_z = a$ es consistente con x_x. Como R_{yz} no está definida, se verifica la camino consistencia de $R_{xy} = (r, a)$ respecto de x_z
- x_w : $x_w = r$ es consistente con x_y. Como R_{xw} no está definida, se verifica la camino consistencia de $R_{xy} = (r, a)$ respecto de x_w

Análogamente se demuestra que $R_{xy} = (a, r)$ es también camino consistente con respecto a x_z y x_w. Por lo tanto, R_{xy} queda intacta.

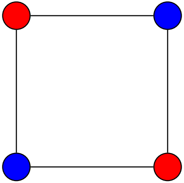
Ejemplo VI

Tomando el primer caso, se tiene entonces que $x_x = r, x_y = a, x_z = a$ es camino consistente. Examinando ahora $R_{yw} = (a, r)$ (puesto que $x_y = a$):

- x_z : $x_z = a$ es consistente con x_w. Como R_{yz} no está definida, se verifica la camino consistencia de $R_{yw} = (a, r)$ respecto de x_z
- x_x : $x_x = r$ es consistente con x_y. Como R_{xw} no está definida, se verifica la camino consistencia de $R_{yw} = (a, r)$ respecto de x_z

Ejemplo VII

Por lo tanto, como $R_{yw} = (a, r)$ es camino consistente con el resto de variables, la asignación $x_w = r$ está permitida que, junto con las asignaciones del paso anterior resultan en el siguiente coloreado:



Bibliografía

Dechter, Rina
Constraint Programming
Morgan Kaufmann, 2003