Universidad del Valle
Escuela de Ingeniería de Sistemas y Computación
Programación por Restricciones
Desarrollo: Taller Modelamiento e Implementación CSPs

Juan Marcos Caicedo Mejía (1730504-3743)

2020

PARTE 1 (CSPs)

1. Sudoku

- Variables:
  - `subcuadricula`

    Representa la dimensión de cada subcuadricula. En un Sudoku común y corriente esto tiene un valor de 3.
  - `cuadricula`

    Representa la dimensión de toda la cuadricula grande (la que contiene las subcuadriculas). Al igual que la variable anterior, en un Sudoku común esta variable tiene el valor de 9.
  - `entrada`

    Representa el tablero de Sudoku inicial que posee algunos números preestablecidos. En esta matriz, el 0 representa una celda vacía.
  - `salida`

    Representa el tablero variable sobre el cual las restricciones tomarán acción y construirán la solución del Sudoku.
  - `subcuadricula_uno`

    Es un array que contiene los 9 números correspondientes a la primera subcuadricula
  - `subcuadricula_dos`

    Es un array que contiene los 9 números correspondientes a la segunda subcuadricula
  - `subcuadricula_tres`

    Es un array que contiene los 9 números correspondientes a la tercera subcuadricula
  - `subcuadricula_cuatro`

    Es un array que contiene los 9 números correspondientes a la cuarta subcuadricula
  - `subcuadricula_cinco`

    Es un array que contiene los 9 números correspondientes a la quinta subcuadricula

- `subcuadricula_seis`

  Es un array que contiene los 9 números correspondientes a la sexta subcuadricula

- `subcuadricula_siete`

  Es un array que contiene los 9 números correspondientes a la séptima subcuadricula

- `subcuadricula_ocho`

  Es un array que contiene los 9 números correspondientes a la octava subcuadricula

- `subcuadricula_nueve`

  Es un array que contiene los 9 números correspondientes a la novena subcuadricula

■ Dominios:

- `dimension_cuadricula = 1..cuadricula`

  Este dominio es útil para poder acceder a todas las celdas de la cuadrícula grande, pues es el que demarca que los números que estén en el tablero solo sean del 1 al 9.

■ Restricciones:

- Una primera restricción que solo sirve para poder copiar los datos del tablero inicial del Sudoku consiste en que si alguna celda en el tablero de la variable de entrada contiene un número mayor que 0, este mismo número se copie en la misma ubicación en el tablero de salida, si no es así, es decir, hay una celda con 0, se omite su valor:

$$\forall i, j \in \{1, 9\} \mid entrada(i, j) > 0 \therefore salida(i, j) = entrada(i, j)$$

- Luego, una restricción importante es para garantizar que los números de todas las filas sean distintos:

$$\forall i, j \in \{1, 9\} \mid$$

$$x_{1,1} \neq x_{1,2} \neq x_{1,3} \neq ... \neq x_{1,9}$$

$$\wedge$$

$$x_{2,1} \neq x_{2,2} \neq x_{2,3} \neq ... \neq x_{2,9}$$

$$\wedge$$

$$...$$

$$x_{9,1} \neq x_{9,2} \neq x_{9,3} \neq ... \neq x_{9,9}$$

- Luego, una restricción importante es para garantizar que los números de todas las columnas sean distintos:

  $$\forall i, j \in \{1, 9\} \mid$$

  $$x_{1,1} \neq x_{2,1} \neq x_{3,1} \neq ... \neq x_{9,1}$$

  $$\wedge$$

  $$x_{1,2} \neq x_{2,2} \neq x_{3,2} \neq ... \neq x_{2,9}$$

  $$\wedge$$

  $$...$$

  $$x_{1,9} \neq x_{2,9} \neq x_{3,9} \neq ... \neq x_{9,9}$$

  Donde $x_{i,j}$ es el número de la celda en la posición (i, j).

- Luego, usando las variables de:

  `subcuadricula_uno`

  hasta

  `subcuadricula_nueve`

  Se impusieron las restricciones que permiten que en cada subcuadrícula no se repitan los números del 1 al 9. Por cada variable de subcuadricula (que contiene los numeros de dicha subcuadricula) se impuso la restricción de

  `alldifferent`

  Para la

  `subcuadricula_uno`

La restricción sería así:

$$x_{1,1} \neq x_{1,2} \neq x_{1,3}$$

$$\wedge$$

$$x_{2,1} \neq x_{2,2} \neq x_{2,3}$$

$$\wedge$$

$$x_{3,1} \neq x_{3,2} \neq x_{3,3}$$

Para la

`subcuadricula_dos`

La restricción sería así:

$$x_{1,4} \neq x_{1,5} \neq x_{1,6}$$

$$\wedge$$

$$x_{2,4} \neq x_{2,5} \neq x_{2,6}$$

$$\wedge$$

$$x_{3,4} \neq x_{3,5} \neq x_{3,6}$$

Para la

`subcuadricula_tres`

La restricción sería así:

$$x_{1,7} \neq x_{1,8} \neq x_{1,9}$$

$$\wedge$$

$$x_{2,7} \neq x_{2,8} \neq x_{2,9}$$

$$\wedge$$

$$x_{3,7} \neq x_{3,8} \neq x_{3,9}$$

Para la

`subcuadricula_cuatro`

La restricción sería así:

$$x_{4,1} \neq x_{4,2} \neq x_{4,3}$$

$$\wedge$$

$$x_{5,1} \neq x_{5,2} \neq x_{5,3}$$

$$\wedge$$

$$x_{6,1} \neq x_{6,2} \neq x_{6,3}$$

Para la

`subcuadricula_cinco`

La restricción sería así:

$$x_{4,4} \neq x_{4,5} \neq x_{4,6}$$

$$\wedge$$

$$x_{5,4} \neq x_{5,5} \neq x_{5,6}$$

$$\wedge$$

$$x_{6,4} \neq x_{6,5} \neq x_{6,6}$$

Para la

`subcuadricula_seis`

La restricción sería así:

$$x_{4,7} \neq x_{4,8} \neq x_{4,9}$$

$$\wedge$$

$$x_{5,7} \neq x_{5,8} \neq x_{5,9}$$

$$\wedge$$

$$x_{6,7} \neq x_{6,8} \neq x_{6,9}$$

Para la

`subcuadricula_siete`

La restricción sería así:

$$x_{7,1} \neq x_{7,2} \neq x_{7,3}$$

$$\wedge$$

$$x_{8,1} \neq x_{8,2} \neq x_{8,3}$$

$$\wedge$$

$$x_{9,1} \neq x_{9,2} \neq x_{9,3}$$

Para la

`subcuadricula_ocho`

La restricción sería así:

$$x_{7,4} \neq x_{7,5} \neq x_{7,6}$$

$$\wedge$$

$$x_{8,4} \neq x_{8,5} \neq x_{8,6}$$

$$\wedge$$

$$x_{9,4} \neq x_{9,5} \neq x_{9,6}$$

Para la

`subcuadricula_nueve`

La restricción sería así:

$$x_{7,7} \neq x_{7,8} \neq x_{7,9}$$

$$\wedge$$

$$x_{8,7} \neq x_{8,8} \neq x_{8,9}$$

$$\wedge$$

$$x_{9,7} \neq x_{9,8} \neq x_{9,9}$$

3. Secuencia Mágica

- Variables:

- `n`

  La variable (ingresada por el usuario mediante el IDE o proveída por un archivo de datos .dzn), representa la longitud de la secuencia mágica.

- Dominios:

  - `dominio = 0..n-1`

    Este dominio caracteriza dos cosas: el tamaño de la secuencia mágica (el tamaño del arreglo que la representa) y los posibles valores que toman los números dentro de la secuencia mágica.

- Restricciones:

  - Llamemos *ocurre* a un predicado que recibe una lista l y un número x, así, $ocurre(l, x)$ arroja el número de ocurrencias del número x en la lista l. Otro predicado puede ser *posicion* que dada una lista l y un número x, retorna la posición del número x en la lista l (indexando desde 0). La restricción principal del problema consiste en que la secuencia mágica se caracteriza porque el número i ocurre exactamente x_i veces en la secuencia. Así, la restricción podría modelarse:

    $$\forall x \in l, ocurre(l, posicion(l, x)) = x$$

    Esto quiere decir que el número de la posición de x en la lista l debe ocurrir x veces en la lista l.

  - Una restricción redundante adicional, dicta que la suma de todos los números en la secuencia debe sumar el número n (longitud de la secuencia):

    $$\sum_{i=0}^{n-1} x_i = n$$

    Donde x_i es el i-ésimo elemento de la secuencia mágica.

  - Otra restricción adicional, dice que la suma de cada elemento de la secuencia multiplicada por $i - 1$ debe ser igual a 0:

    $$\sum_{i=0}^{n-1} x_i * (i - 1) = 0$$

    Donde x_i es el i-ésimo elemento de la secuencia mágica.