

1 NQueen MiniZinc

1.1 Modellazione

Sia $n > 3$ la dimensione del problema.

$$X = \{Q_1, \dots, Q_n\} \quad (1)$$

Ogni variabile Q_i rappresenta la posizione della regina sull' i -esima colonna. Perciò viene naturale definire i domini.

$$D_i = \{x \mid 1 \leq x \leq n\} \quad (2)$$

Questi sono i domini per l' i -esima variabile.

$$\begin{aligned} \forall i, j \mid Q_i, Q_j \in X \wedge i \neq j \\ C_{(i,j)} = (\{Q_i, Q_j\}, \{(x, y) \mid x \in D_i \wedge y \in D_j \wedge x \neq y \wedge |x - y| \neq |i - j|\}) \end{aligned} \quad (3)$$

I vincoli sono fatti in questo modo. Per ogni coppia di variabile (quindi per ogni colonna) le regine non possono stare sulla stessa riga ($x \neq y$) e sulla stessa diagonale (il fatto che non sono sulla stessa colonna è implicito).

Quindi per verificare che due regine non siano sulla stessa diagonale abbiamo che il valore assoluto della differenza sulla posizione delle regine sulle righe deve essere diverso dal valore assoluto della differenza sulla posizione delle regine sulle colonne.

Da notare però che ho dei vincoli duplicati, per esempio nel caso $n = 5$ avrò il vincolo $C_{(1,2)} = C_{(2,1)}$.

Avere vincoli in più non è necessariamente un problema. Però quando verrà implementato con MiniZinc questi vincoli in più daranno problemi per l'efficienza.

Quindi ecco una formulazione equivalente dove elimino i vincoli duplicati:

$$\begin{aligned} \forall i, j \mid Q_i, Q_j \in X \wedge j < i \\ C_{(i,j)} = (\{Q_i, Q_j\}, \{(x, y) \mid x \in D_i \wedge y \in D_j \wedge x \neq y \wedge |x - y| \neq i - j\}) \end{aligned} \quad (4)$$

1.2 Istanziamento

Per il caso $n = 5$ abbiamo che le variabili sono:

$$X = \{Q_1, Q_2, Q_3, Q_4, Q_5\} \quad (5)$$

I domini sono:

$$\forall i \mid 1 \leq i \leq n = 5, D_i = \{1, 2, 3, 4, 5\} \quad (6)$$

Invece i vincoli sono:

$$\left\{ \begin{array}{l} C_{(2,1)} = (\{Q_2, Q_1\}, \{(2,1), (3,1), (4,1), (5,1), (1,3), (2,4), (3,5)\}) \\ C_{(3,1)} = (\{Q_3, Q_1\}, \{(2,1), (3,1), (4,1), (5,1), (1,4), (2,5)\}) \\ C_{(4,1)} = (\{Q_4, Q_1\}, \{(2,1), (3,1), (4,1), (5,1), (1,5)\}) \\ C_{(5,1)} = (\{Q_5, Q_1\}, \{(2,1), (3,1), (4,1), (5,1)\}) \\ C_{(3,2)} = (\{Q_3, Q_2\}, \{(1,2), (3,2), (4,2), (5,2), (2,4), (3,5)\}) \\ C_{(4,2)} = (\{Q_4, Q_2\}, \{(1,2), (3,2), (4,2), (5,2), (2,5)\}) \\ C_{(5,2)} = (\{Q_5, Q_2\}, \{(1,2), (3,2), (4,2), (5,2)\}) \\ C_{(4,3)} = (\{Q_4, Q_3\}, \{(1,3), (2,3), (4,3), (5,3), (3,5)\}) \\ C_{(5,3)} = (\{Q_5, Q_3\}, \{(1,3), (2,3), (4,3), (5,3)\}) \\ C_{(5,4)} = (\{Q_5, Q_4\}, \{(1,4), (2,4), (3,4), (5,4)\}) \end{array} \right.$$

1.3 Implementazione MiniZinc

Si può trovare l'implementazione della modellazione descritta in questo documento nella stesso repository nel file model.mzn. Per le costanti guardare data.dzn.