5.142 element_matrix

DESCRIPTION LINKS GRAPH AUTOMATON

Origin CHIP

Constraint element_matrix(MAX_I, MAX_J, INDEX_I, INDEX_J, MATRIX, VALUE)

Synonyms elem_matrix, matrix.

Arguments

MAX_I	:	int
MAX_J	:	int
INDEX_I	:	dvar
INDEX_J	:	dvar
MATRIX	:	collection(i−int, j−int, v−int)
VALUE	:	dvar

Restrictions

MAX_I ≥ 1
MAX_J ≥ 1
INDEX_I ≥ 1
INDEX_I $\leq$ MAX_I
INDEX_J ≥ 1
INDEX_J $\leq$ MAX_J
required(MATRIX, [i, j, v])
increasing_seq(MATRIX, [i, j])
MATRIX.i ≥ 1
MATRIX.i $\leq$ MAX_I
MATRIX.j ≥ 1
MATRIX.j $\leq$ MAX_J
|MATRIX| = MAX_I $*$ MAX_J

Purpose The MATRIX collection corresponds to the two-dimensional matrix MATRIX[1..MAX_I, 1..MAX_J]. VALUE is equal to the entry MATRIX[INDEX_I, INDEX_J] of the previous matrix.

Example

$$\left(4, 3, 1, 3, \left\langle \begin{array}{lll} \mathtt{i}-1 & \mathtt{j}-1 & \mathtt{v}-4, \\ \mathtt{i}-1 & \mathtt{j}-2 & \mathtt{v}-1, \\ \mathtt{i}-1 & \mathtt{j}-3 & \mathtt{v}-7, \\ \mathtt{i}-2 & \mathtt{j}-1 & \mathtt{v}-1, \\ \mathtt{i}-2 & \mathtt{j}-2 & \mathtt{v}-0, \\ \mathtt{i}-2 & \mathtt{j}-3 & \mathtt{v}-8, \\ \mathtt{i}-3 & \mathtt{j}-1 & \mathtt{v}-3, \\ \mathtt{i}-3 & \mathtt{j}-2 & \mathtt{v}-2, \\ \mathtt{i}-3 & \mathtt{j}-3 & \mathtt{v}-1, \\ \mathtt{i}-4 & \mathtt{j}-1 & \mathtt{v}-0, \\ \mathtt{i}-4 & \mathtt{j}-2 & \mathtt{v}-0, \\ \mathtt{i}-4 & \mathtt{j}-3 & \mathtt{v}-6 \end{array} \right\rangle, 7\right)$$

The element_matrix constraint holds since its last argument $\texttt{VALUE} = 7$ is equal to the v attribute of the k^{th} item of the MATRIX collection such that $\texttt{MATRIX}[k].i = \texttt{INDEX\_I} = 1$ and $\texttt{MATRIX}[k].j = \texttt{INDEX\_J} = 3$.

Typical

$\texttt{MAX\_I} > 1$
$\texttt{MAX\_J} > 1$
$|\texttt{MATRIX}| > 3$
$\texttt{maxval}(\texttt{MATRIX.i}) > 1$
$\texttt{maxval}(\texttt{MATRIX.j}) > 1$
$\texttt{range}(\texttt{MATRIX.v}) > 1$

Symmetry

All occurrences of two distinct values in MATRIX.v or VALUE can be swapped; all occurrences of a value in MATRIX.v or VALUE can be renamed to any unused value.

Reformulation

The $\texttt{element\_matrix}(\texttt{MAX\_I}, \texttt{MAX\_J}, \texttt{INDEX\_I}, \texttt{INDEX\_J}, \texttt{MATRIX}, \texttt{VALUE})$ constraint can be expressed in term of MAX_I $\texttt{element}(\texttt{INDEX\_J}, \texttt{LINE}_i, \texttt{VAR}_i)$ $(i \in [1, \texttt{MAX\_I}])$, where $\texttt{LINE}_i$ corresponds to the i-th line of the matrix MATRIX and of one $\texttt{element}(\texttt{INDEX\_I}, \langle \texttt{VAR}_1, \texttt{VAR}_2, \ldots, \texttt{VAR}_{\texttt{MAX\_I}} \rangle, \texttt{VALUE})$ constraint.

If we consider the **Example** slot we get the following element constraints:

- $\texttt{element}(3, \langle 4, 1, 7 \rangle, 7)$,
- $\texttt{element}(3, \langle 1, 0, 8 \rangle, 8)$,
- $\texttt{element}(3, \langle 3, 2, 1 \rangle, 1)$,
- $\texttt{element}(3, \langle 0, 0, 6 \rangle, 6)$,
- $\texttt{element}(1, \langle 7, 8, 1, 6 \rangle, 7)$.

Systems

nth in **Choco**, element in **Gecode**.

See also

common keyword: elem, element *(array constraint)*.

Keywords

characteristic of a constraint: automaton, automaton without counters, reified automaton constraint, derived collection.

constraint arguments: ternary constraint.

constraint network structure: centered cyclic(3) constraint network(1).

constraint type: data constraint.

filtering: arc-consistency.

modelling: array constraint, matrix.

Derived Collection

$$\texttt{col}\left(\begin{array}{l}\texttt{ITEM}-\texttt{collection}(\texttt{index\_i}-\texttt{dvar},\texttt{index\_j}-\texttt{dvar},\texttt{value}-\texttt{dvar}),\\ [\texttt{item}(\texttt{index\_i}-\texttt{INDEX\_I},\texttt{index\_j}-\texttt{INDEX\_J},\texttt{value}-\texttt{VALUE})]\end{array}\right)$$

Arc input(s) ITEM MATRIX

Arc generator $PRODUCT \mapsto$ `collection(item, matrix)`

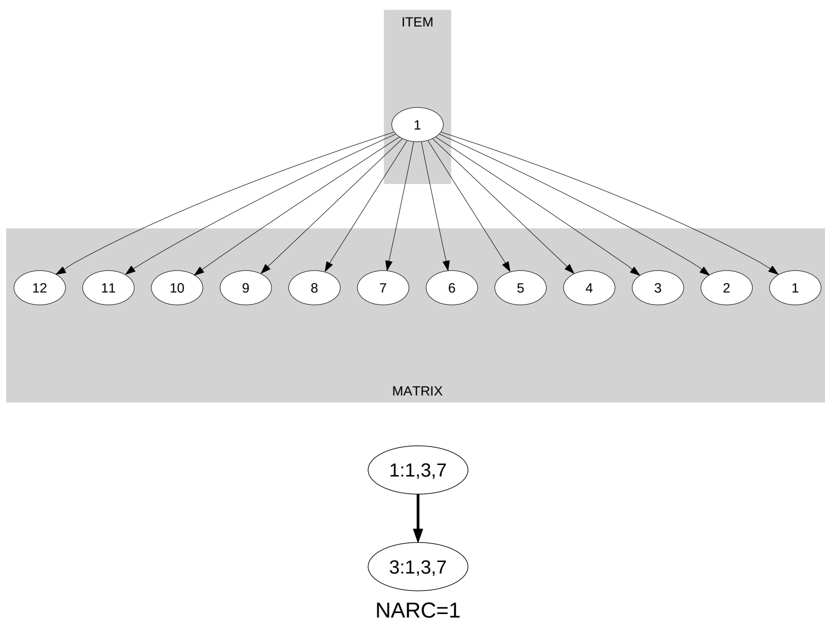
Arc arity 2

Arc constraint(s)
- `item.index_i = matrix.i`
- `item.index_j = matrix.j`
- `item.value = matrix.v`

Graph property(ies) $\mathbf{NARC} = 1$

Graph model Similar to the `element` constraint except that the arc constraint is updated according to the fact that we have a two-dimensional matrix.

Parts (A) and (B) of Figure 5.328 respectively show the initial and final graph associated with the **Example** slot. Since we use the $\mathbf{NARC}$ graph property, the unique arc of the final graph is stressed in bold.



(A)

(B)

Figure 5.328: Initial and final graph of the `element_matrix` constraint

Signature Because of the first condition of the arc constraint the final graph cannot have more than one arc. Therefore we can rewrite $\mathbf{NARC} = 1$ to $\mathbf{NARC} \geq 1$ and simplify $\overline{\mathbf{NARC}}$ to $\underline{\overline{\mathbf{NARC}}}$.

Automaton Figure 5.329 depicts the automaton associated with the `element_matrix` constraint. Let $\mathtt{I}_k$, $\mathtt{J}_k$ and $\mathtt{V}_k$ respectively be the `i`, the `j` and the `v` k^{th} attributes of the `MATRIX` collection. To each sextuple $(\mathtt{INDEX\_I}, \mathtt{INDEX\_J}, \mathtt{VALUE}, \mathtt{I}_k, \mathtt{J}_k, \mathtt{V}_k)$ corresponds a 0-1 signature variable S_k as well as the following signature constraint: $((\mathtt{INDEX\_I} = \mathtt{I}_k) \wedge (\mathtt{INDEX\_J} = \mathtt{J}_k) \wedge (\mathtt{VALUE} = \mathtt{V}_k)) \Leftrightarrow S_k$.

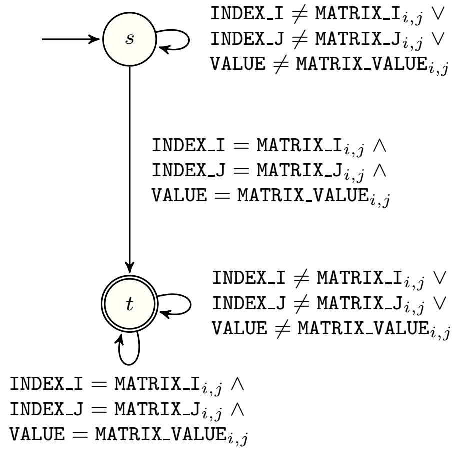


Figure 5.329: Automaton of the `element_matrix` constraint

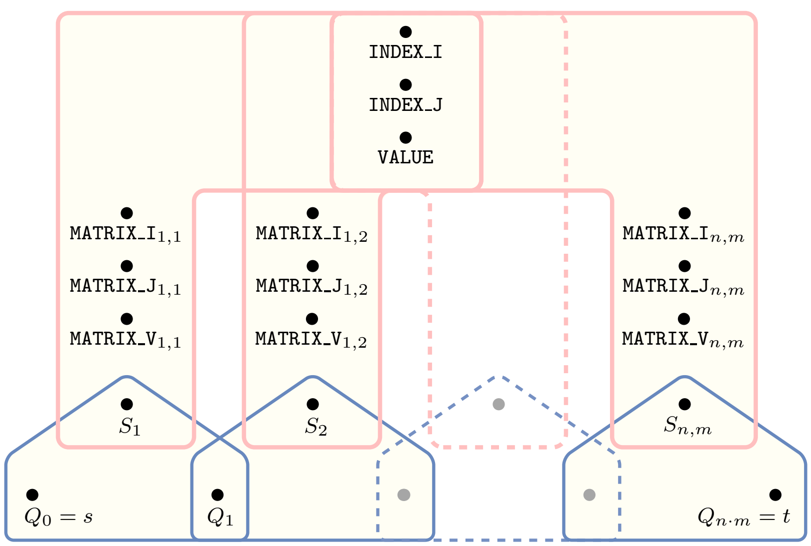


Figure 5.330: Hypergraph of the reformulation corresponding to the automaton of the `element_matrix` constraint where n and m respectively stands for `MAX_I` and `MAX_J`