5.292 nvectors

DESCRIPTION LINKS GRAPH

Origin Inspired by nvector and count.

Constraint nvectors(VECTORS, RELOP, LIMIT)

Synonym npoints.

Type

VECTOR	:	collection(var−dvar)

Arguments

VECTORS	:	collection(vec − VECTOR)
RELOP	:	atom
LIMIT	:	dvar

Restrictions

$|\texttt{VECTOR}| \geq 1$
required(VECTORS, vec)
same_size(VECTORS, vec)
$\texttt{RELOP} \in [=, \neq, <, \geq, >, \leq]$

Purpose Let N be the number of distinct tuples of values taken by the vectors of the VECTORS collection. Enforce condition N RELOP LIMIT to hold.

Example

$$\left(\begin{array}{l} \texttt{vec} - \langle 5,6\rangle, \\ \left\langle \begin{array}{l} \texttt{vec} - \langle 5,6\rangle, \\ \texttt{vec} - \langle 9,3\rangle, \\ \texttt{vec} - \langle 5,6\rangle, \end{array} \right\rangle, =, 2 \\ \texttt{vec} - \langle 9,3\rangle \end{array}\right)$$

The nvectors constraint holds since the number of distinct tuples of values (i.e., tuples $\langle 5,6\rangle$ and $\langle 9,3\rangle$) occurring within the collection VECTORS is equal (i.e., RELOP is set to $=$) to its third argument LIMIT $= 2$.

Typical

$|\texttt{VECTOR}| > 1$
$|\texttt{VECTORS}| > 1$
$\texttt{RELOP} \in [=, <, \geq, >, \leq]$
$\texttt{LIMIT} > 1$
$\texttt{LIMIT} < |\texttt{VECTORS}|$

Symmetries

- Items of VECTORS are permutable.
- Items of VECTORS.vec are permutable (*same permutation used*).
- All occurrences of two distinct values of VECTORS.vec can be swapped; all occurrences of a value of VECTORS.vec can be renamed to any unused value.

Arg. properties

- Contractible wrt. VECTORS when RELOP $\in [<, \leq]$.
- Extensible wrt. VECTORS when RELOP $\in [\geq, >]$.

Reformulation The nvectors(VECTORS, RELOP , LIMIT) constraint can be expressed in term of the conjunction nvector(NV, VECTORS) $\wedge$ NV RELOP LIMIT.

See also **specialisation:** nvector *(replace a comparison with the number of distinct vectors by an equality with the number of distinct vectors)*.

Keywords **characteristic of a constraint:** vector.

constraint type: counting constraint, value partitioning constraint.

final graph structure: strongly connected component, equivalence.

modelling: number of distinct equivalence classes.

problems: domination.

Arc input(s)	VECTORS
Arc generator	$CLIQUE \mapsto$ collection(vectors1, vectors2)
Arc arity	2
Arc constraint(s)	lex_equal(vectors1.vec, vectors2.vec)
Graph property(ies)	NSCC RELOP LIMIT
Graph class	EQUIVALENCE

Graph model Parts (A) and (B) of Figure 5.625 respectively show the initial and final graph associated with the **Example** slot. Since we use the **NSCC** graph property we show the different strongly connected components of the final graph. Each strongly connected component corresponds to a tuple of values that is assigned to some vectors of the VECTORS collection. The 2 following tuple of values $\langle 5, 6\rangle$ and $\langle 9, 3\rangle$ are used by the vectors of the VECTORS collection.

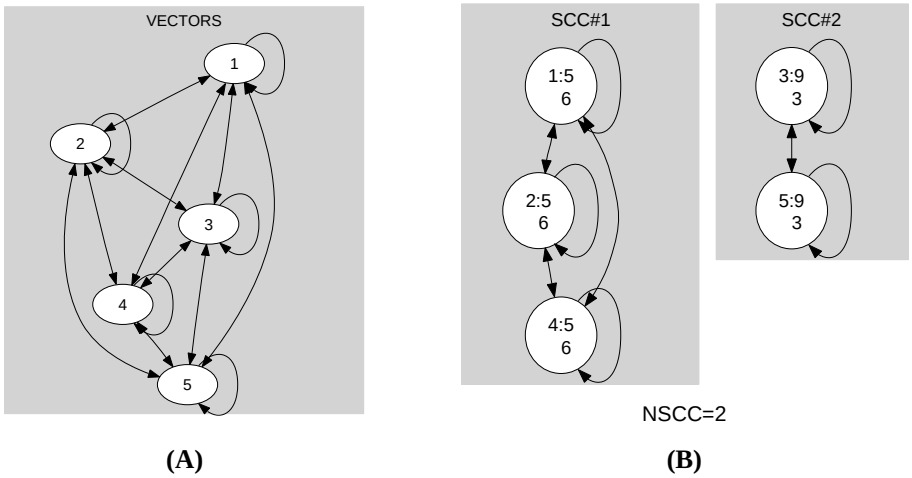

Figure 5.625: Initial and final graph of the nvectors constraint