

5.29 among_var

	DESCRIPTION	LINKS	GRAPH
Origin	Generalisation of among		
Constraint	<code>among_var(NVAR, VARIABLES, VALUES)</code>		
Arguments	NVAR : dvar VARIABLES : collection (var-dvar) VALUES : collection (val-dvar)		
Restrictions	$NVAR \geq 0$ $NVAR \leq VARIABLES $ required (VARIABLES, var) required (VALUES, val)		
Purpose	NVAR is the number of variables of the collection VARIABLES that are equal to one of the variables of the collection VALUES.		
Example	<code>(3, <4, 5, 5, 4, 1>, <1, 5, 8, 1>)</code> The <code>among_var</code> constraint holds since exactly 3 values of the collection of variables <code><4, 5, 5, 4, 1></code> occurs within the collection <code><1, 5, 8, 1></code> .		
Typical	$ VARIABLES > 1$ $ VALUES > 1$ $ VARIABLES > VALUES $		
Symmetries	<ul style="list-style-type: none"> Items of VARIABLES are permutable. Items of VALUES are permutable. All occurrences of two distinct values in VARIABLES.var or VALUES.val can be swapped; all occurrences of a value in VARIABLES.var or VALUES.val can be renamed to any unused value. An occurrence of a value of VARIABLES.var that belongs to VALUES.val (resp. does not belong to VALUES.val) can be replaced by any other value in VALUES.val (resp. not in VALUES.val). 		
Arg. properties	<ul style="list-style-type: none"> Functional dependency: NVAR determined by VARIABLES and VALUES. Contractible wrt. VARIABLES when $NVAR = 0$. Contractible wrt. VARIABLES when $NVAR = VARIABLES$. Aggregate: $NVAR(+)$, $VARIABLES(\text{union})$, $VALUES(\text{union})$. 		
Systems	<code>among</code> in Choco , <code>count</code> in Gecode , <code>amongvar</code> in JaCoP .		

See also

implied by: [among](#).

related: [common](#).

specialisation: [among](#) (*variable replaced by constant within list of values VALUES*).

uses in its reformulation: [min_n](#).

Keywords

constraint arguments: pure functional dependency.

constraint type: counting constraint.

final graph structure: acyclic, bipartite, no loop.

modelling: functional dependency.

Arc input(s)	VARIABLES VALUES
Arc generator	<i>PRODUCT</i> \mapsto <code>collection(variables, values)</code>
Arc arity	2
Arc constraint(s)	<code>variables.var = values.val</code>
Graph property(ies)	NSOURCE = NVAR
Graph class	<ul style="list-style-type: none"> • ACYCLIC • BIPARTITE • NO_LOOP

Graph model

Parts (A) and (B) of Figure 5.67 respectively show the initial and final graph associated with the **Example** slot. Since we use the **NSOURCE** graph property, the source vertices of the final graph are stressed with a double circle. Since the final graph has only 3 sources the variables NVAR is fixed to 3.

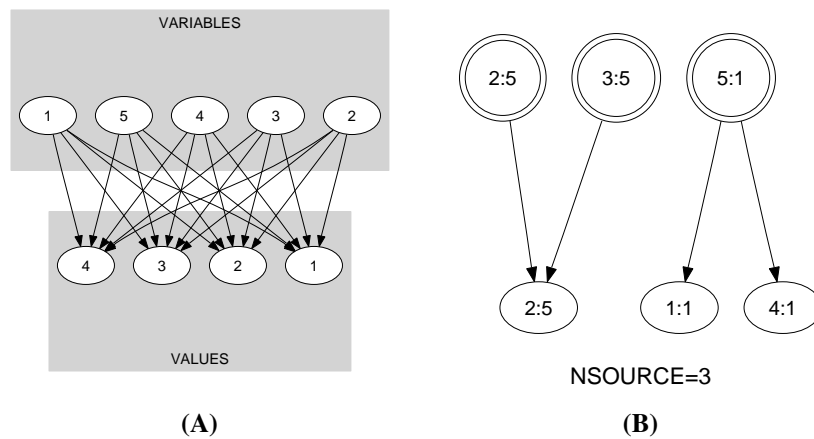


Figure 5.67: Initial and final graph of the `among_var` constraint

20090418

611