

5.288 **nvalue_on_intersection**

	DESCRIPTION	LINKS	GRAPH
Origin	Derived from common and nvalue .		
Constraint	<code>nvalue_on_intersection(NVAL, VARIABLES1, VARIABLES2)</code>		
Arguments	<div>NVAL : dvar VARIABLES1 : collection(var–dvar) VARIABLES2 : collection(var–dvar)</div>		
Restrictions	<div>required(VARIABLES1, var) required(VARIABLES2, var) NVAL ≥ 0 NVAL ≤ VARIABLES1 NVAL ≤ VARIABLES2 NVAL ≤ range(VARIABLES1.var) NVAL ≤ range(VARIABLES2.var)</div>		
Purpose	NVAL is the number of distinct values that both occur in the VARIABLES1 and VARIABLES2 collections.		
Example	<div>(2, ⟨1, 9, 1, 5⟩, ⟨2, 1, 9, 9, 6, 9⟩)</div> <p>Note that the two collections ⟨1, 9, 1, 5⟩ and ⟨2, 1, 9, 9, 6, 9⟩ share two values in common (i.e., values 1 and 9). Consequently the <code>nvalue_on_intersection</code> constraint holds since its first argument NVAL is set to 2.</p>		
Typical	<div>NVAL > 0 NVAL < VARIABLES1 NVAL < VARIABLES2 NVAL < range(VARIABLES1.var) NVAL < range(VARIABLES2.var) VARIABLES1 > 1 VARIABLES2 > 1</div>		
Symmetries	<ul style="list-style-type: none">Arguments are permutable w.r.t. permutation (NVAL) (VARIABLES1, VARIABLES2).Items of VARIABLES1 are permutable.Items of VARIABLES2 are permutable.All occurrences of two distinct values in VARIABLES1.var or VARIABLES2.var can be swapped; all occurrences of a value in VARIABLES1.var or VARIABLES2.var can be renamed to any unused value.		

Arg. properties

- [Functional dependency](#): NVAL determined by VARIABLES1 and VARIABLES2.
- [Contractible](#) wrt. VARIABLES1 when NVAL = 0.
- [Contractible](#) wrt. VARIABLES2 when NVAL = 0.

See also

common keyword: [alldifferent_on_intersection](#), [common](#), [same_intersection](#) (*constraint on the intersection*).

root concept: [nvalue](#).

Keywords

constraint arguments: pure functional dependency.

constraint type: counting constraint, constraint on the intersection.

final graph structure: connected component.

modelling: number of distinct values, functional dependency.

Arc input(s)	VARIABLES1 VARIABLES2
Arc generator	<i>PRODUCT</i> \mapsto <i>collection</i> (variables1,variables2)
Arc arity	2
Arc constraint(s)	variables1.var = variables2.var
Graph property(ies)	<i>NCC</i> = NVAL

Graph model

Parts (A) and (B) of Figure 5.620 respectively show the initial and final graph associated with the **Example** slot. Since we use the *NCC* graph property we show the connected components of the final graph. The variable NVAL is equal to this number of *connected components*. Note that all the vertices corresponding to the variables that take values 5, 2 or 6 were removed from the final graph since there is no arc for which the associated equality constraint holds.

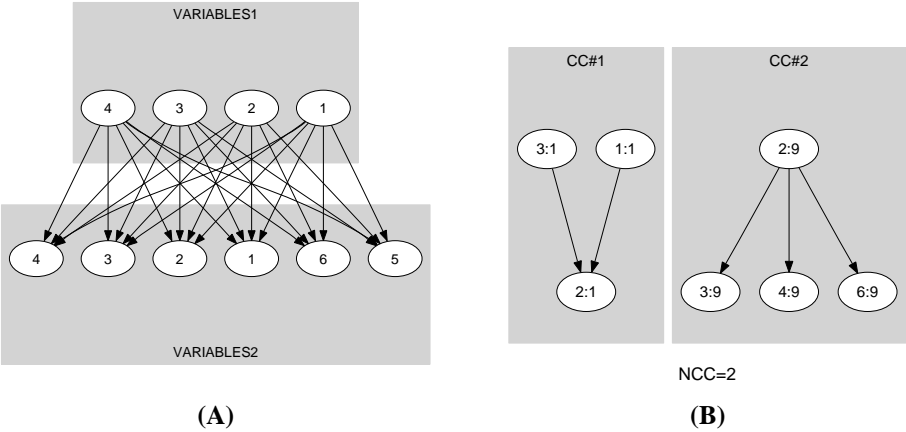


Figure 5.620: Initial and final graph of the *nvalue_on_intersection* constraint

