

**5.284 npair**

	DESCRIPTION	LINKS	GRAPH
<b>Origin</b>	Derived from <code>nvalue</code> .		
<b>Constraint</b>	<code>npair(NPAIRS, PAIRS)</code>		
<b>Arguments</b>	<code>NPAIRS</code> : <code>dvar</code> <code>PAIRS</code> : <code>collection(x-dvar, y-dvar)</code>		
<b>Restrictions</b>	$NPAIRS \geq \min(1,  PAIRS )$ $NPAIRS \leq  PAIRS $ <code>required</code> (PAIRS, [x, y])		
<b>Purpose</b>	NPAIRS is the number of distinct pairs of values assigned to the pairs of variables of the collection PAIRS.		
<b>Example</b>	$\left( 2, \left\langle \begin{array}{cc} x-3 & y-1, \\ x-1 & y-5, \\ x-3 & y-1, \\ x-3 & y-1, \\ x-1 & y-5 \end{array} \right\rangle \right)$		
	The <code>npair</code> constraint holds since its first argument $NPAIRS = 2$ is set to the number of distinct pairs $\langle x-3 \ y-1 \rangle$ and $\langle x-1 \ y-5 \rangle$ of its second argument PAIRS.		
<b>Typical</b>	$NPAIRS > 1$ $NPAIRS <  PAIRS $ $ PAIRS  > 1$ <code>range</code> (PAIRS.x) > 1 <code>range</code> (PAIRS.y) > 1		
<b>Symmetries</b>	<ul style="list-style-type: none"> <li>Items of PAIRS are <code>permutable</code>.</li> <li>Attributes of PAIRS are <code>permutable</code> w.r.t. permutation <math>(x, y)</math> (<i>permutation applied to all items</i>).</li> <li>All occurrences of two distinct tuples of values of NPAIRS can be <code>swapped</code>; all occurrences of a tuple of values of NPAIRS can be <code>renamed</code> to any unused tuple of values.</li> </ul>		
<b>Arg. properties</b>	<ul style="list-style-type: none"> <li><code>Functional dependency</code>: NPAIRS determined by PAIRS.</li> <li><code>Contractible</code> wrt. PAIRS when <math>NPAIRS = 1</math> and <math> PAIRS  &gt; 0</math>.</li> <li><code>Contractible</code> wrt. PAIRS when <math>NPAIRS =  PAIRS </math>.</li> </ul>		
<b>Remark</b>	This is an example of a <i>number of distinct values</i> constraint where there is more than one attribute that is associated with each vertex of the final graph.		

**See also**

**related:** `nclass` (pair of variables replaced by variable  $\in$  partition),  
`nequivalence` (pair of variables replaced by variable mod constant),  
`ninterval` (pair of variables replaced by variable/constant).  
**specialisation:** `nvalue` (pair of variables replaced by variable).

**Keywords**

**characteristic of a constraint:** pair.  
**constraint arguments:** pure functional dependency.  
**constraint type:** counting constraint, value partitioning constraint.  
**final graph structure:** strongly connected component, equivalence.  
**modelling:** number of distinct equivalence classes, functional dependency.

<b>Arc input(s)</b>	PAIRS
<b>Arc generator</b>	<i>CLIQUE</i> $\mapsto$ collection(pairs1, pairs2)
<b>Arc arity</b>	2
<b>Arc constraint(s)</b>	<ul style="list-style-type: none"> <li>• pairs1.x = pairs2.x</li> <li>• pairs1.y = pairs2.y</li> </ul>
<b>Graph property(ies)</b>	<u>NSCC</u> = NPAIRS

**Graph model**

Parts (A) and (B) of Figure 5.595 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 pair of values that is assigned to some pairs of variables of the PAIRS collection. In our example we have the following pairs of values:  $\langle x - 3 \ y - 1 \rangle$  and  $\langle x - 1 \ y - 5 \rangle$ .

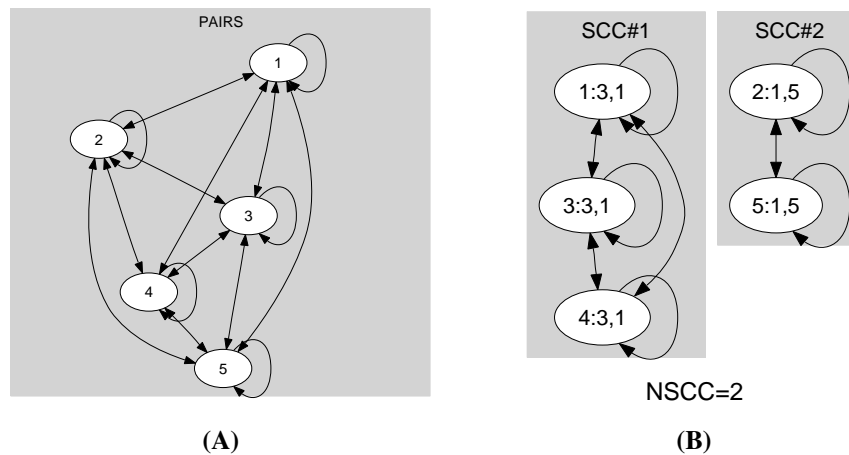


Figure 5.595: Initial and final graph of the npair constraint

20030820

1819