

5.308 soft_same_var

	DESCRIPTION	LINKS	GRAPH
Origin	[384]		
Constraint	<code>soft_same_var(C, VARIABLES1, VARIABLES2)</code>		
Synonym	<code>soft_same.</code>		
Arguments	C : <code>dvar</code> VARIABLES1 : <code>collection(var-dvar)</code> VARIABLES2 : <code>collection(var-dvar)</code>		
Restrictions	$C \geq 0$ $C \leq VARIABLES1 $ $ VARIABLES1 = VARIABLES2 $ <code>required(VARIABLES1, var)</code> <code>required(VARIABLES2, var)</code>		
Purpose	C is the minimum number of values to change in the VARIABLES1 and VARIABLES2 collections so that the variables of the VARIABLES2 collection correspond to the variables of the VARIABLES1 collection according to a permutation.		
Example	$\left(\begin{array}{c} \text{var} - 9, \\ \text{var} - 9, \\ 4, \left\langle \begin{array}{c} \text{var} - 9, \\ \text{var} - 9, \end{array} \right\rangle, \\ \text{var} - 9, \\ \text{var} - 1 \\ \text{var} - 9, \\ \text{var} - 1, \\ \left\langle \begin{array}{c} \text{var} - 1, \\ \text{var} - 1, \end{array} \right\rangle \\ \text{var} - 1, \\ \text{var} - 8 \end{array} \right)$		
	As illustrated by Figure 5.550, there is a correspondence between two pairs of values of the collections $\langle 9, 9, 9, 9, 9, 1 \rangle$ and $\langle 9, 1, 1, 1, 1, 8 \rangle$. Consequently, we must unset at least $6 - 2$ items (6 is the number of items of the VARIABLES1 and VARIABLES2 collections). The <code>soft_same_var</code> constraint holds since its first argument C is set to $6 - 2$.		
Symmetries	<ul style="list-style-type: none"> Arguments are <code>permutable</code> w.r.t. permutation (C) (VARIABLES1, VARIABLES2). Items of VARIABLES1 are <code>permutable</code>. Items of VARIABLES2 are <code>permutable</code>. All occurrences of two distinct values in VARIABLES1.var or VARIABLES2.var can be <code>swapped</code>; all occurrences of a value in VARIABLES1.var or VARIABLES2.var can be <code>renamed</code> to any unused value. 		

Usage	A soft same constraint.
Algorithm	[384, page 80].
See also	hard version: same . implies: soft_used_by_var .
Keywords	constraint arguments : constraint between two collections of variables. constraint type : soft constraint, relaxation, variable-based violation measure. filtering : minimum cost flow.

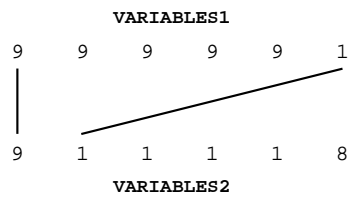


Figure 5.550: Correspondence between collection $\langle 9, 9, 9, 9, 9, 1 \rangle$ and collection $\langle 9, 1, 1, 1, 1, 8 \rangle$

Arc input(s)	VARIABLES1 VARIABLES2
Arc generator	<i>PRODUCT</i> \mapsto <code>collection(variables1, variables2)</code>
Arc arity	2
Arc constraint(s)	<code>variables1.var = variables2.var</code>
Graph property(ies)	<u>NSINK_NSOURCE</u> = $ VARIABLES1 - C$

Graph model

Parts (A) and (B) of Figure 5.551 respectively show the initial and final graph associated with the **Example** slot. Since we use the NSINK_NSOURCE graph property, the source and sink vertices of the final graph are stressed with a double circle. The `soft_same_var` constraint holds since the cost 4 corresponds to the difference between the number of variables of `VARIABLES1` and the sum over the different connected components of the minimum number of sources and sinks.

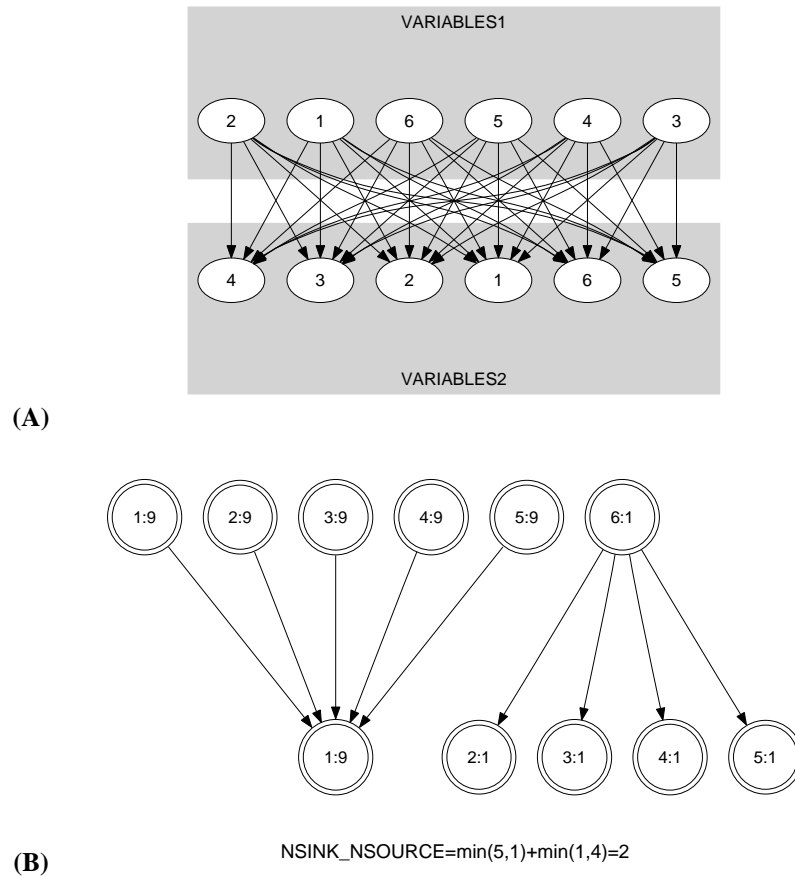


Figure 5.551: Initial and final graph of the `soft_same_var` constraint

20050507

1587