5.340 same_partition

DESCRIPTION LINKS GRAPH

Origin Derived from same.

Constraint same_partition(VARIABLES1, VARIABLES2, PARTITIONS)

Type VALUES : collection(val−int)

Arguments

VARIABLES1 : collection(var−dvar)
VARIABLES2 : collection(var−dvar)
PARTITIONS : collection(p − VALUES)

Restrictions

|VALUES| ≥ 1
required(VALUES, val)
distinct(VALUES, val)
|VARIABLES1| = |VARIABLES2|
required(VARIABLES1, var)
required(VARIABLES2, var)
required(PARTITIONS, p)
|PARTITIONS| ≥ 2

Purpose For each integer i in [1, |PARTITIONS|], let $N1_i$ (respectively $N2_i$) denote the number of variables of VARIABLES1 (respectively VARIABLES2) that take their value in the i^{th} partition of the collection PARTITIONS. For all i in [1, |PARTITIONS|] we have $N1_i = N2_i$.

Example

$$\left(\begin{array}{l}\langle 1,2,6,3,1,2\rangle,\\ \langle 6,6,2,3,1,3\rangle,\\ \langle \mathtt{p}-\langle 1,3\rangle, \mathtt{p}-\langle 4\rangle, \mathtt{p}-\langle 2,6\rangle\rangle\end{array}\right)$$

The different values of the collection ⟨1, 2, 6, 3, 1, 2⟩ are respectively associated with the partitions p − ⟨1, 3⟩, p − ⟨2, 6⟩, p − ⟨2, 6⟩, p − ⟨1, 3⟩, p − ⟨1, 3⟩, and p − ⟨2, 6⟩. Therefore partitions p − ⟨1, 3⟩ and p − ⟨2, 6⟩ are respectively used 3 and 3 times. Similarly, the different values of the collection ⟨6, 6, 2, 3, 1, 3⟩ are respectively associated with the partitions p − ⟨2, 6⟩, p − ⟨2, 6⟩, p − ⟨2, 6⟩, p − ⟨1, 3⟩, p − ⟨1, 3⟩, and p − ⟨1, 3⟩. As before partitions p − ⟨1, 3⟩ and p − ⟨2, 6⟩ are respectively used 3 and 3 times. Consequently the same_partition constraint holds. Figure 5.675 illustrates this correspondence.

Typical

|VARIABLES1| > 1
range(VARIABLES1.var) > 1
range(VARIABLES2.var) > 1
|VARIABLES1| > |PARTITIONS|
|VARIABLES2| > |PARTITIONS|

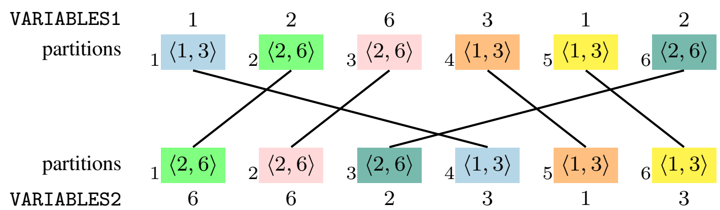


Figure 5.675: Illustration of the correspondence between the items of the `VARIABLES1` and of the `VARIABLES2` collections of the **Example** slot

Symmetries

- Arguments are permutable w.r.t. permutation (`VARIABLES1`, `VARIABLES2`) (`PARTITIONS`).
- Items of `VARIABLES1` are permutable.
- Items of `VARIABLES2` are permutable.
- Items of `PARTITIONS` are permutable.
- Items of `PARTITIONS.p` are permutable.
- An occurrence of a value of `VARIABLES.var` can be replaced by any other value that also belongs to the same partition of `PARTITIONS`.

Arg. properties

Aggregate: `VARIABLES1`(`union`), `VARIABLES2`(`union`), `PARTITIONS`(`id`).

Used in

`k_same_partition`.

See also

implies: `used_by_partition`.

soft variant: `soft_same_partition_var` *(variable-based violation measure)*.

specialisation: `same` *(*`variable` ∈ `partition` *replaced by* `variable`*)*.

system of constraints: `k_same_partition`.

used in graph description: `in_same_partition`.

Keywords

characteristic of a constraint: sort based reformulation, partition.

combinatorial object: permutation.

constraint arguments: constraint between two collections of variables.

Arc input(s)	VARIABLES1 VARIABLES2
Arc generator	$PRODUCT \mapsto$ collection(variables1, variables2)
Arc arity	2
Arc constraint(s)	in_same_partition(variables1.var, variables2.var, PARTITIONS)
Graph property(ies)	• for all connected components: **NSOURCE**=**NSINK** • **NSOURCE**= |VARIABLES1| • **NSINK**= |VARIABLES2|

Graph model

Parts (A) and (B) of Figure 5.676 respectively show the initial and final graph associated with the **Example** slot. Since we use the **NSOURCE** and **NSINK** graph properties, the source and sink vertices of the final graph are stressed with a double circle. Since there is a constraint on each connected component of the final graph we also show the different connected components. Each of them corresponds to an equivalence class according to the arc constraint. The same_partition constraint holds since:

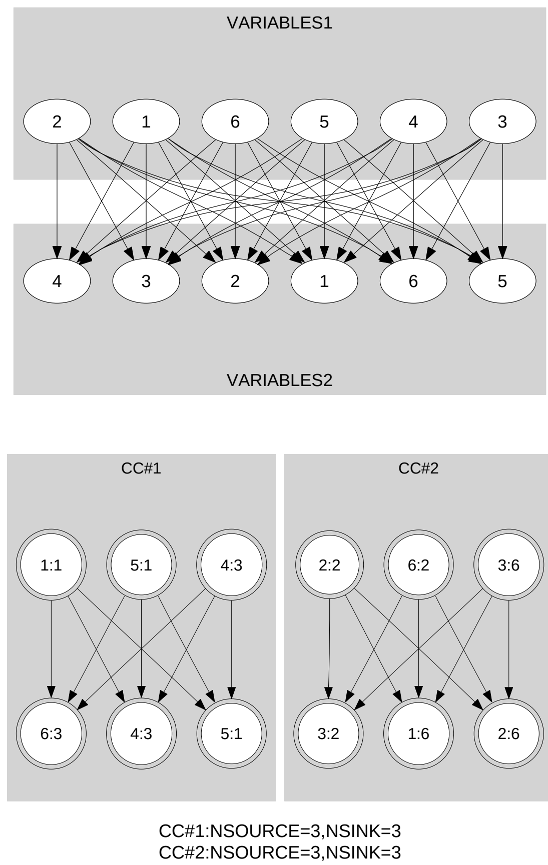
- Each connected component of the final graph has the same number of sources and of sinks.
- The number of sources of the final graph is equal to |VARIABLES1|.
- The number of sinks of the final graph is equal to |VARIABLES2|.

Signature

Since the initial graph contains only sources and sinks, and since isolated vertices are eliminated from the final graph, we make the following observations:

- Sources of the initial graph cannot become sinks of the final graph,
- Sinks of the initial graph cannot become sources of the final graph.

From the previous observations and since we use the $PRODUCT$ arc generator on the collections VARIABLES1 and VARIABLES2, we have that the maximum number of sources and sinks of the final graph is respectively equal to |VARIABLES1| and |VARIABLES2|. Therefore we can rewrite **NSOURCE** $=$ |VARIABLES1| to **NSOURCE** $\geq$ |VARIABLES1| and simplify $\underline{\overline{\mathbf{NSOURCE}}}$ to $\overline{\mathbf{NSOURCE}}$. In a similar way, we can rewrite **NSINK** $=$ |VARIABLES2| to **NSINK** $\geq$ |VARIABLES2| and simplify $\underline{\overline{\mathbf{NSINK}}}$ to $\overline{\mathbf{NSINK}}$.



(A)

(B)

Figure 5.676: Initial and final graph of the `same_partition` constraint