5.78   common_partition

DESCRIPTION   **LINKS**   **GRAPH**

Origin   Derived from common.

Constraint

$$\texttt{common\_partition} \left(\begin{array}{l} \texttt{NCOMMON1,} \\ \texttt{NCOMMON2,} \\ \texttt{VARIABLES1,} \\ \texttt{VARIABLES2,} \\ \texttt{PARTITIONS} \end{array} \right)$$

Type

VALUES	:	collection(val−int)

Arguments

NCOMMON1	:	dvar
NCOMMON2	:	dvar
VARIABLES1	:	collection(var−dvar)
VARIABLES2	:	collection(var−dvar)
PARTITIONS	:	collection(p − VALUES)

Restrictions

$|\texttt{VALUES}| \geq 1$
required(VALUES, val)
distinct(VALUES, val)
$\texttt{NCOMMON1} \geq 0$
$\texttt{NCOMMON1} \leq |\texttt{VARIABLES1}|$
$\texttt{NCOMMON2} \geq 0$
$\texttt{NCOMMON2} \leq |\texttt{VARIABLES2}|$
required(VARIABLES1, var)
required(VARIABLES2, var)
required(PARTITIONS, p)
$|\texttt{PARTITIONS}| \geq 2$

Purpose

NCOMMON1 is the number of variables of the VARIABLES1 collection taking a value in a partition derived from the values assigned to the variables of VARIABLES2 and from PARTITIONS.
NCOMMON2 is the number of variables of the VARIABLES2 collection taking a value in a partition derived from the values assigned to the variables of VARIABLES1 and from PARTITIONS.

Example

$$\left(\begin{array}{l} 3, 4, \langle 2, 3, 6, 0 \rangle, \\ \langle 0, 6, 3, 3, 7, 1 \rangle, \\ \langle \texttt{p} - \langle 1, 3 \rangle, \texttt{p} - \langle 4 \rangle, \texttt{p} - \langle 2, 6 \rangle \rangle \end{array} \right)$$

In the example, the last argument PARTITIONS defines the partitions $\texttt{p} - \langle 1, 3 \rangle$, $\texttt{p} - \langle 4 \rangle$ and $\texttt{p} - \langle 2, 6 \rangle$. As a consequence the first three items of collection $\langle 2, 3, 6, 0 \rangle$ respectively correspond to the partitions $\texttt{p} - \langle 2, 6 \rangle$, $\texttt{p} - \langle 1, 3 \rangle$, and $\texttt{p} - \langle 2, 6 \rangle$. Similarly the items of collection $\langle 0, 6, 3, 3, 7, 1 \rangle$ (from which we remove items 0 and 7 since they do

not belong to any partition) respectively correspond to the partitions $\texttt{p} - \langle 2, 6 \rangle$, $\texttt{p} - \langle 1, 3 \rangle$, $\texttt{p} - \langle 1, 3 \rangle$, and $\texttt{p} - \langle 1, 3 \rangle$. The `common_partition` constraint holds since:

- Its first argument $\texttt{NCOMMON1} = 3$ is the number of partitions associated with the items of collection $\langle 2, 3, 6, 0 \rangle$ that also correspond to partitions associated with $\langle 0, 6, 3, 3, 7, 1 \rangle$.

- Its second argument $\texttt{NCOMMON2} = 4$ is the number of partitions associated with the items of collection $\langle 0, 6, 3, 3, 7, 1 \rangle$ that also correspond to partitions associated with $\langle 2, 3, 6, 0 \rangle$.

Typical

$|\texttt{VARIABLES1}| > 1$
$\texttt{range}(\texttt{VARIABLES1.var}) > 1$
$|\texttt{VARIABLES2}| > 1$
$\texttt{range}(\texttt{VARIABLES2.var}) > 1$
$|\texttt{VARIABLES1}| > |\texttt{PARTITIONS}|$
$|\texttt{VARIABLES2}| > |\texttt{PARTITIONS}|$

Symmetries

- Arguments are permutable w.r.t. permutation $(\texttt{NCOMMON1}, \texttt{NCOMMON2})$ $(\texttt{VARIABLES1}, \texttt{VARIABLES2})$ $(\texttt{PARTITIONS})$.

- 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 `VARIABLES1.var` can be replaced by any other value that also belongs to the same partition of `PARTITIONS`.

- An occurrence of a value of `VARIABLES2.var` can be replaced by any other value that also belongs to the same partition of `PARTITIONS`.

Arg. properties

- Functional dependency: `NCOMMON1` determined by `VARIABLES1`, `VARIABLES2` and `PARTITIONS`.

- Functional dependency: `NCOMMON2` determined by `VARIABLES1`, `VARIABLES2` and `PARTITIONS`.

See also

specialisation: `common` *(*`variable` $\in$ `partition` *replaced by* `variable`*).*

used in graph description: `in_same_partition`.

Keywords

characteristic of a constraint: partition.

constraint arguments: constraint between two collections of variables, pure functional dependency.

final graph structure: acyclic, bipartite, no loop.

modelling: functional dependency.

Arc input(s)	VARIABLES1 VARIABLES2
Arc generator	*PRODUCT*↦collection(variables1, variables2)
Arc arity	2
Arc constraint(s)	in_same_partition(variables1.var, variables2.var, PARTITIONS)
Graph property(ies)	• **NSOURCE**= NCOMMON1 • **NSINK**= NCOMMON2
Graph class	• ACYCLIC • BIPARTITE • NO_LOOP

Graph model   Parts (A) and (B) of Figure 5.184 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 the graph has only 3 sources and 4 sinks the variables NCOMMON1 and NCOMMON2 are respectively equal to 3 and 4. Note that the vertices corresponding to the variables that take values 0 or 7 were removed from the final graph since there is no arc for which the associated in_same_partition constraint holds.
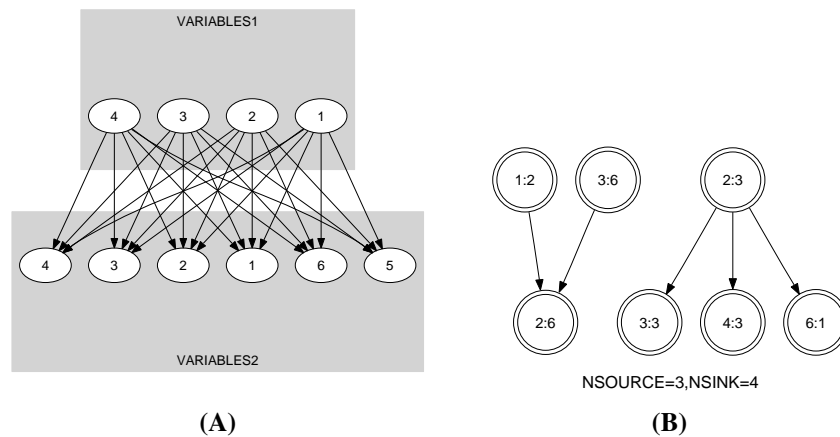

(A)   **(B)**

Figure 5.184: Initial and final graph of the common_partition constraint