5.64 change_partition

DESCRIPTION LINKS GRAPH

Origin Derived from change.

Constraint change_partition(NCHANGE, VARIABLES, PARTITIONS)

Type
```
VALUES : collection(val−int)
```

Arguments
```
NCHANGE    : dvar
VARIABLES  : collection(var−dvar)
PARTITIONS : collection(p − VALUES)
```

Restrictions
```
|VALUES| ≥ 1
required(VALUES, val)
distinct(VALUES, val)
NCHANGE ≥ 0
NCHANGE < |VARIABLES|
required(VARIABLES, var)
required(PARTITIONS, p)
|PARTITIONS| ≥ 2
```

Purpose NCHANGE is the number of times that the following constraint holds: X and Y do not belong to the same partition of the collection PARTITIONS, where X and Y correspond to consecutive variables of the collection VARIABLES.

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

In the example we have the following two changes:

- One change between values 2 and 1 (since 2 and 1 respectively belong to the third and the first partition),
- One change between values 1 and 6 (since 1 and 6 respectively belong to the first and the third partition).

Consequently the change_partition constraint holds since its first argument NCHANGE is assigned to 2.

Typical
```
NCHANGE > 0
|VARIABLES| > 1
range(VARIABLES.var) > 1
|VARIABLES| > |PARTITIONS|
```

Symmetries

- Items of VARIABLES can be reversed.
- 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

Functional dependency: NCHANGE determined by VARIABLES and PARTITIONS.

Usage

This constraint is useful for the following problem: Assume you have to produce a set of orders, each order belonging to a given family. In the context of the **Example** slot we have three families that respectively correspond to values $1, 3$, to value 4 and to values $2, 6$. We would like to sequence the orders in such a way that we minimise the number of times two consecutive orders do not belong to the same family.

Algorithm

[30].

See also

common keyword: change *(number of changes in a sequence of* variables *with respect to a* binary constraint*)*.

used in graph description: in_same_partition.

Keywords

characteristic of a constraint: partition.

constraint arguments: pure functional dependency.

constraint type: timetabling constraint.

final graph structure: acyclic, bipartite, no loop.

modelling: number of changes, functional dependency.

Arc input(s)	VARIABLES
Arc generator	$PATH \mapsto$ collection(variables1, variables2)
Arc arity	2
Arc constraint(s)	in_same_partition(variables1.var, variables2.var, PARTITIONS)
Graph property(ies)	**NARC**= NCHANGE
Graph class	• ACYCLIC • BIPARTITE • NO_LOOP

Graph model Parts (A) and (B) of Figure 5.161 respectively show the initial and final graph associated with the **Example** slot. Since we use the **NARC** graph property, the arcs of the final graph are stressed in bold.

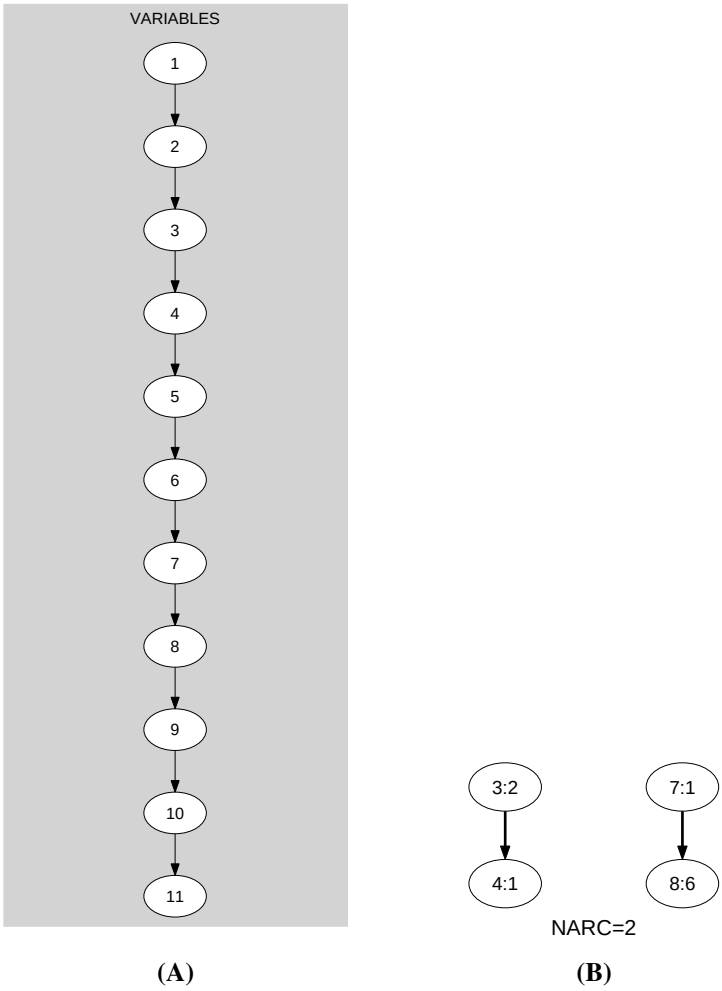

Figure 5.161: Initial and final graph of the `change_partition` constraint