

5.64 change\_partition

|              | DESCRIPTION   | LINKS | GRAPH |
|--------------|---|-------|-------|
| Origin       | Derived from <a href="#">change</a> .   |       |       |
| Constraint   | change_partition(NCHANGE, VARIABLES, PARTITIONS)  |       |       |
| Type         | VALUES : <a href="#">collection</a> (val-int)   |       |       |
| Arguments    | NCHANGE : <a href="#">dvar</a><br>VARIABLES : <a href="#">collection</a> (var-dvar)<br>PARTITIONS : <a href="#">collection</a> (p-VALUES)   |       |       |
| Restrictions | VALUES  ≥ 1<br><a href="#">required</a> (VALUES, val)<br><a href="#">distinct</a> (VALUES, val)<br>NCHANGE ≥ 0<br>NCHANGE <  VARIABLES <br><a href="#">required</a> (VARIABLES, var)<br><a href="#">required</a> (PARTITIONS, p)<br> PARTITIONS  ≥ 2  |       |       |
| Purpose      | NCHANGE is the number of times that the following constraint holds: <i>X</i> and <i>Y</i> do not belong to the same partition of the collection PARTITIONS, where <i>X</i> and <i>Y</i> correspond to consecutive variables of the collection VARIABLES.  |       |       |
| Example      | $\left( \begin{array}{l} 2, \langle 6, 6, 2, 1, 3, 3, 1, 6, 2, 2, 2 \rangle, \\ \langle p - \langle 1, 3 \rangle, p - \langle 4 \rangle, p - \langle 2, 6 \rangle \rangle \end{array} \right)$ <p>In the example we have the following two changes:</p> <ul style="list-style-type: none"><li>• One change between values 2 and 1 (since 2 and 1 respectively belong to the third and the first partition),</li><li>• One change between values 1 and 6 (since 1 and 6 respectively belong to the first and the third partition).</li></ul> <p>Consequently the <code>change_partition</code> constraint holds since its first argument NCHANGE is assigned to 2.</p> |       |       |
| Typical      | NCHANGE > 0<br> VARIABLES  > 1<br><a href="#">range</a> (VARIABLES.var) > 1<br> 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       | <i>PATH</i> $\mapsto$ collection(variables1, variables2)  |
| Arc arity           | 2   |
| Arc constraint(s)   | in_same_partition(variables1.var, variables2.var, PARTITIONS)   |
| Graph property(ies) | <b>NARC</b> = NCHANGE   |
| Graph class         | <ul style="list-style-type: none"><li>• ACYCLIC</li><li>• BIPARTITE</li><li>• NO_LOOP</li></ul>   |
| Graph model         | Parts (A) and (B) of Figure 5.178 respectively show the initial and final graph associated with the <b>Example</b> slot. Since we use the <b>NARC</b> graph property, the arcs of the final graph are stressed in bold. |

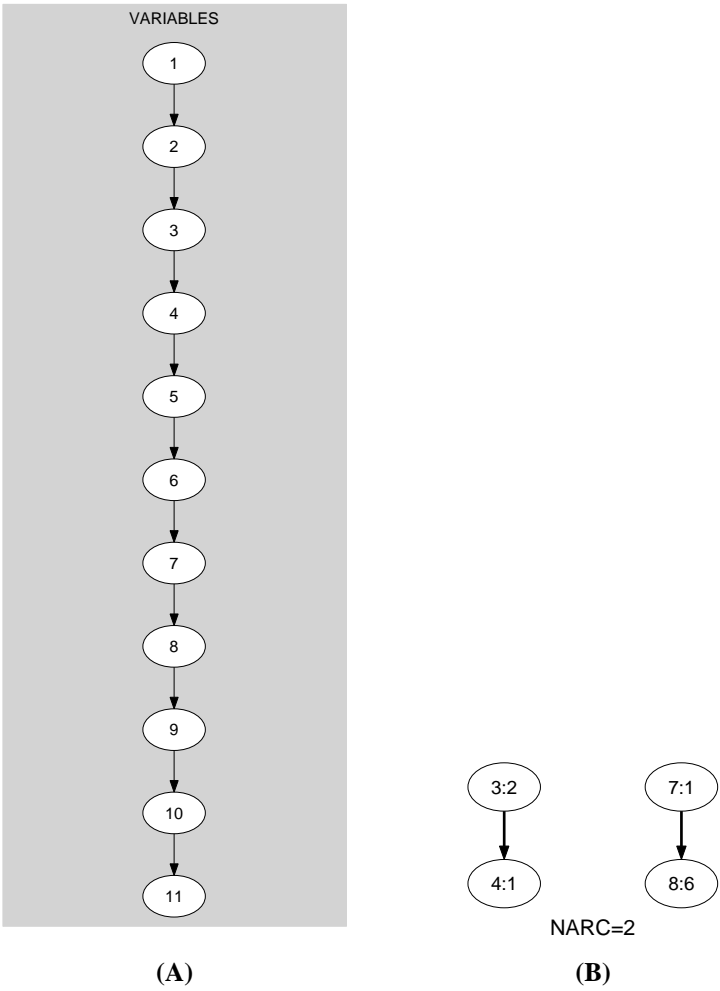


Figure 5.178: Initial and final graph of the change\_partition constraint