

5.63 common_interval

	DESCRIPTION	LINKS	GRAPH
Origin	Derived from <code>common</code> .		
Constraint	$\text{common_interval} \left(\begin{array}{l} \text{NCOMMON1,} \\ \text{NCOMMON2,} \\ \text{VARIABLES1,} \\ \text{VARIABLES2,} \\ \text{SIZE_INTERVAL} \end{array} \right)$		
Arguments	<pre> NCOMMON1 : dvar NCOMMON2 : dvar VARIABLES1 : collection(var-dvar) VARIABLES2 : collection(var-dvar) SIZE_INTERVAL : int </pre>		
Restrictions	<pre> NCOMMON1 ≥ 0 NCOMMON1 ≤ VARIABLES1 NCOMMON2 ≥ 0 NCOMMON2 ≤ VARIABLES2 required(VARIABLES1, var) required(VARIABLES2, var) SIZE_INTERVAL > 0 </pre>		
Purpose	<p>NCOMMON1 is the number of variables of the collection of variables VARIABLE1 taking a value in one of the intervals derived from the values assigned to the variables of the collection VARIABLE2: To each value v assigned to a variable of the collection VARIABLE2 we associate the interval $[\text{SIZE_INTERVAL} \cdot \lfloor v/\text{SIZE_INTERVAL} \rfloor, \text{SIZE_INTERVAL} \cdot \lfloor v/\text{SIZE_INTERVAL} \rfloor + \text{SIZE_INTERVAL} - 1]$. NCOMMON2 is the number of variables of the collection of variables VARIABLE2 taking a value in one of the intervals derived from the values assigned to the variables of the collection VARIABLE1: To each value v assigned to a variable of the collection VARIABLE1 we associate the interval $[\text{SIZE_INTERVAL} \cdot \lfloor v/\text{SIZE_INTERVAL} \rfloor, \text{SIZE_INTERVAL} \cdot \lfloor v/\text{SIZE_INTERVAL} \rfloor + \text{SIZE_INTERVAL} - 1]$.</p>		
Example	$\left(\begin{array}{l} 3, 2, \langle 8, 6, 6, 0 \rangle, \\ \text{var} - 7, \\ \text{var} - 3, \\ \langle \text{var} - 3, \rangle, 3 \\ \text{var} - 3, \\ \text{var} - 3, \\ \text{var} - 7 \end{array} \right)$		

In the example, the last argument `SIZE_INTERVAL = 3` defines the following family of intervals $[3 \cdot k, 3 \cdot k + 2]$, where k is an integer. As a consequence the items of

collection $\langle 8, 6, 6, 0 \rangle$ respectively correspond to intervals $[6, 8]$, $[6, 8]$, $[6, 8]$ and $[0, 2]$. Similarly the items of collection $\langle 7, 3, 3, 3, 3, 7 \rangle$ respectively correspond to intervals $[6, 8]$, $[3, 5]$, $[3, 5]$, $[3, 5]$, $[3, 5]$, $[6, 8]$. The `common_interval` constraint holds since:

- Its first argument `NCOMMON1 = 3` is the number of intervals associated with the items of collection $\langle 8, 6, 6, 0 \rangle$ that also correspond to intervals associated with $\langle 7, 3, 3, 3, 3, 7 \rangle$.
- Its second argument `NCOMMON2 = 2` is the number of intervals associated with the items of collection $\langle 7, 3, 3, 3, 3, 7 \rangle$ that also correspond to intervals associated with $\langle 8, 6, 6, 0 \rangle$.

Typical

```
|VARIABLES1| > 1
range(VARIABLES1.var) > 1
|VARIABLES2| > 1
range(VARIABLES2.var) > 1
SIZE_INTERVAL > 1
```

Symmetries

- Arguments are [permutable](#) w.r.t. permutation $(NCOMMON1, NCOMMON2)$ $(VARIABLES1, VARIABLES2)$ $(SIZE_INTERVAL)$.
- Items of `VARIABLES1` are [permutable](#).
- Items of `VARIABLES2` are [permutable](#).
- An occurrence of a value of `VARIABLES1.var` that belongs to the k -th interval, of size `SIZE_INTERVAL`, can be [replaced](#) by any other value of the same interval.
- An occurrence of a value of `VARIABLES2.var` that belongs to the k -th interval, of size `SIZE_INTERVAL`, can be [replaced](#) by any other value of the same interval.

See also

[specialisation](#): `common` (variable/constant *replaced by variable*).

Keywords

[constraint arguments](#): constraint between two collections of variables.

[final graph structure](#): acyclic, bipartite, no loop.

[modelling](#): interval.

Arc input(s)	VARIABLES1 VARIABLES2
Arc generator	<i>PRODUCT</i> \mapsto <code>collection(variables1, variables2)</code>
Arc arity	2
Arc constraint(s)	$\text{variables1.var}/\text{SIZE_INTERVAL} = \text{variables2.var}/\text{SIZE_INTERVAL}$
Graph property(ies)	<ul style="list-style-type: none"> • NSOURCE = NCOMMON1 • NSINK = NCOMMON2
Graph class	<ul style="list-style-type: none"> • ACYCLIC • BIPARTITE • NO_LOOP

Graph model

Parts (A) and (B) of Figure 5.134 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 2 sinks the variables NCOMMON1 and NCOMMON2 are respectively equal to 3 and 2. Note that the vertices corresponding to the variables that take values 0 or 3 were removed from the final graph since there is no arc for which the associated arc constraint holds.

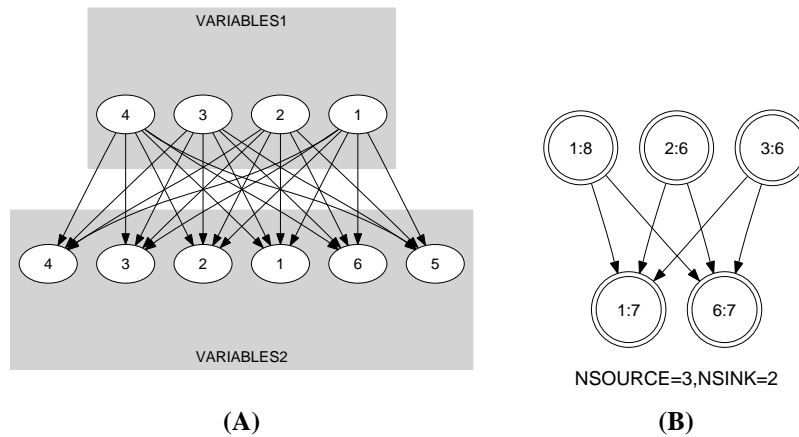


Figure 5.134: Initial and final graph of the `common_interval` constraint

20030820

645