5.413 used_by_interval

DESCRIPTION LINKS GRAPH

Origin

Derived from `used_by`.

Constraint

`used_by_interval(VARIABLES1, VARIABLES2, SIZE_INTERVAL)`

Arguments

```
VARIABLES1    : collection(var-dvar)
VARIABLES2    : collection(var-dvar)
SIZE_INTERVAL : int
```

Restrictions

|VARIABLES1| $\geq$ |VARIABLES2|
required(VARIABLES1, var)
required(VARIABLES2, var)
SIZE_INTERVAL > 0

Purpose

Let N_i (respectively M_i) denote the number of variables of the collection VARIABLES1 (respectively VARIABLES2) that take a value in the interval [SIZE_INTERVAL $\cdot\, i$, SIZE_INTERVAL $\cdot\, i$ + SIZE_INTERVAL $- 1$]. For all integer i we have $M_i > 0 \Rightarrow N_i \geq M_i$.

Example

$(\langle 1,9,1,8,6,2\rangle, \langle 1,0,7,7\rangle, 3)$

In the example, the third argument SIZE_INTERVAL $=$ 3 defines the following family of intervals $[3\cdot k, 3\cdot k+2]$, where k is an integer. Consequently the values of the collection VARIABLES2 $= \langle 1,0,7,7\rangle$ are respectively located within intervals $[0,2]$, $[0,2]$, $[6,8]$, $[6,8]$. Therefore intervals $[0,2]$ and $[6,8]$ are respectively used 2 and 2 times.

Similarly, the values of the collection VARIABLES1 $= \langle 1,9,1,8,6,2\rangle$ are respectively located within intervals $[0,2]$, $[9,11]$, $[0,2]$, $[6,8]$, $[6,8]$, $[0,2]$. Therefore intervals $[0,2]$, $[6,8]$ and $[9,11]$ are respectively used 3, 2 and 1 times.

Consequently, the `used_by_interval` constraint holds since, for each interval associated with the collection VARIABLES2 $= \langle 1,0,7,7\rangle$, its number of occurrences within VARIABLES1 $= \langle 1,9,1,8,6,2\rangle$ is greater than or equal to its number of occurrences within VARIABLES2:

- Interval $[0,2]$ occurs 3 times within $\langle 1,9,1,8,6,2\rangle$ and 2 times within $\langle 1,0,7,7\rangle$.
- Interval $[6,8]$ occurs 2 times within $\langle 1,9,1,8,6,2\rangle$ and 2 times within $\langle 1,0,7,7\rangle$.

Typical

|VARIABLES1| > 1
range(VARIABLES1.var) > 1
|VARIABLES2| > 1
range(VARIABLES2.var) > 1
SIZE_INTERVAL > 1
SIZE_INTERVAL $<$ range(VARIABLES1.var)
SIZE_INTERVAL $<$ range(VARIABLES2.var)

Symmetries

- 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.

Arg. properties

- Contractible wrt. VARIABLES2.
- Extensible wrt. VARIABLES1.
- Aggregate: VARIABLES1(union), VARIABLES2(union), SIZE_INTERVAL(id).

Reformulation

The used_by_interval($\langle \texttt{var} - U_1\ \texttt{var} - U_2, \dots, \texttt{var} - U_{|\texttt{VARIABLES1}|}\rangle$, $\langle \texttt{var} - V_1\ \texttt{var} - V_2, \dots, \texttt{var} - V_{|\texttt{VARIABLES2}|}\rangle$, SIZE_INTERVAL) constraint can be expressed by introducing $|\texttt{VARIABLES1}| + |\texttt{VARIABLES2}|$ *quotient* variables
$U_i = \texttt{SIZE\_INTERVAL} \cdot P_i + R_i$, $R_i \in [0, \texttt{SIZE\_INTERVAL} - 1]$ $(i \in [1, |\texttt{VARIABLES1}|])$,
$V_i = \texttt{SIZE\_INTERVAL} \cdot Q_i + S_i$, $S_i \in [0, \texttt{SIZE\_INTERVAL} - 1]$ $(i \in [1, |\texttt{VARIABLES2}|])$,
in term of a conjunction of $|\texttt{VARIABLES2}|$ reified constraints of the form:
$\sum_{1 \leq j \leq |\texttt{VARIABLES1}|}(Q_i = P_j) \geq \sum_{1 \leq j \leq |\texttt{VARIABLES2}|}(Q_i = Q_j)$ $(i \in [1, |\texttt{VARIABLES2}|])$.

Used in

k_used_by_interval.

See also

implied by: same_interval.

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

specialisation: used_by (variable/constant *replaced by* variable).

system of constraints: k_used_by_interval.

Keywords

characteristic of a constraint: sort based reformulation.

constraint arguments: constraint between two collections of variables.

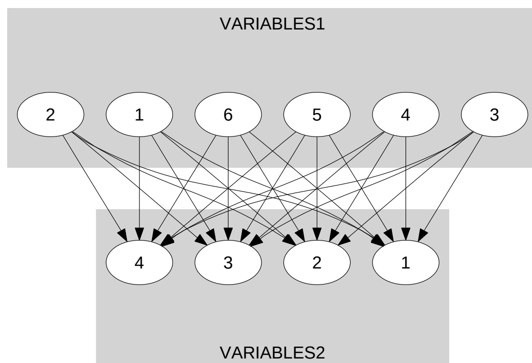
modelling: inclusion, interval.

Arc input(s)	VARIABLES1 VARIABLES2
Arc generator	$PRODUCT \mapsto$ collection(variables1, variables2)
Arc arity	2
Arc constraint(s)	variables1.var/SIZE_INTERVAL = variables2.var/SIZE_INTERVAL
Graph property(ies)	• for all connected components: $\mathbf{NSOURCE} \geq \mathbf{NSINK}$ • $\mathbf{NSINK} = \|$VARIABLES2$\|$

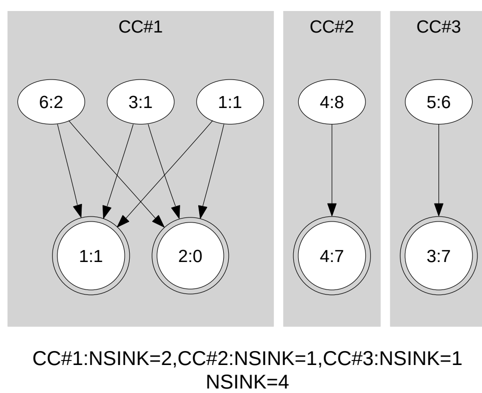
Graph model Parts (A) and (B) of Figure 5.785 respectively show the initial and final graph associated with the **Example** slot. Since we use the **NSOURCE** and **NSINK** graph properties, the source and sink vertices of the final graph are stressed with a double circle. Since there is a constraint 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. Note that the vertex corresponding to the variable that takes value 9 was removed from the final graph since there is no arc for which the associated equivalence constraint holds. The used_by_interval constraint holds since:

- For each connected component of the final graph the number of sources is greater than or equal to the number of sinks.
- The number of sinks of the final graph is equal to |VARIABLES2|.

Signature Since the initial graph contains only sources and sinks, and since sources of the initial graph cannot become sinks of the final graph, we have that the maximum number of sinks of the final graph is equal to |VARIABLES2|. Therefore we can rewrite $\mathbf{NSINK} =$ |VARIABLES2| to $\mathbf{NSINK} \geq$ |VARIABLES2| and simplify $\underline{\mathbf{NSINK}}$ to $\overline{\mathbf{NSINK}}$.



(A)



(B)

Figure 5.785: Initial and final graph of the `used_by_interval` constraint