5.166 global_cardinality_no_loop

DESCRIPTION LINKS GRAPH

Origin

Derived from `global_cardinality` and `tree`.

Constraint

`global_cardinality_no_loop`(`NLOOP`, `VARIABLES`, `VALUES`)

Synonym

`gcc_no_loop`.

Arguments

```
NLOOP     : dvar
VARIABLES : collection(var−dvar)
VALUES    : collection(val−int, noccurrence−dvar)
```

Restrictions

`NLOOP` ≥ 0
`NLOOP` $\leq |$`VARIABLES`$|$
`required`(`VARIABLES`, `var`)
$|$`VALUES`$| > 0$
`required`(`VALUES`, [`val`, `noccurrence`])
`distinct`(`VALUES`, `val`)
`VALUES.noccurrence` ≥ 0
`VALUES.noccurrence` $\leq |$`VARIABLES`$|$

Purpose

`VALUES`$[i]$`.noccurrence` $(1 \leq i \leq |$`VALUES`$|)$ is equal to the number of variables `VARIABLES`$[j]$`.var` $(j \neq i, 1 \leq j \leq |$`VARIABLES`$|)$ that are assigned value `VALUES`$[i]$`.val`.
The number of assignments of the form `VARIABLES`$[i]$`.var` $= i$ $(i \in [1, |$`VARIABLES`$|])$ is equal to `NLOOP`.

Example

$$\left(\begin{array}{l} 1, \langle 1, 1, 8, 6\rangle, \\ \left\langle\begin{array}{ll} \texttt{val}-1 & \texttt{noccurrence}-1, \\ \texttt{val}-5 & \texttt{noccurrence}-0, \\ \texttt{val}-6 & \texttt{noccurrence}-1 \end{array}\right\rangle \end{array}\right)$$

The `global_cardinality_no_loop` constraint holds since:

- Values 1, 5 and 6 are respectively assigned to the set of variables $\{$`VARIABLES`$[2]$`.var`$\}$ (i.e., 1 occurrence of value 1), $\{\}$ (i.e., no occurrence of value 5) and $\{$`VARIABLES`$[4]$`.var`$\}$ (i.e., 1 occurrence of value 6). Note that, due to the definition of the constraint, the fact that `VARIABLES`$[1]$`.var` is assigned to 1 is not counted.
- In addition the number of assignments of the form `VARIABLES`$[i]$`.var` $= i$ $(i \in [1, 4])$ is equal to `NLOOP` $= 1$.

Typical

$|$`VARIABLES`$| > 1$
`range`(`VARIABLES.var`) > 1
$|$`VALUES`$| > 1$
$|$`VARIABLES`$| > |$`VALUES`$|$

Symmetry

Items of `VALUES` are permutable.

Arg. properties

- Functional dependency: `NLOOP` determined by `VARIABLES`.
- Functional dependency: `VALUES.noccurrence` determined by `VARIABLES` and `VALUES.val`.

Usage

Within the context of the `tree` constraint the `global_cardinality_no_loop` constraint allows to model a minimum and maximum degree constraint on each vertex of our trees.

Algorithm

The flow algorithm that handles the original `global_cardinality` constraint [342] can be adapted to the context of the `global_cardinality_no_loop` constraint. This is done by creating an extra *value* node representing the loops corresponding to the roots of the trees.

See also

related: `tree` *(graph partitioning by a set of trees with degree restrictions)*.

root concept: `global_cardinality` *(assignment of a* `variable` *to its position is ignored)*.

specialisation: `global_cardinality_low_up_no_loop` *(*`variable` *replaced by* `fixed interval`*)*.

Keywords

constraint arguments: pure functional dependency.

constraint type: value constraint.

filtering: flow.

modelling: functional dependency.

For all items of VALUES:

Arc input(s)	VARIABLES
Arc generator	*SELF* ↦ collection(variables)
Arc arity	1
Arc constraint(s)	• variables.var = VALUES.val • variables.key ≠ VALUES.val
Graph property(ies)	**NVERTEX** = VALUES.noccurrence

Arc input(s)	VARIABLES
Arc generator	*SELF* ↦ collection(variables)
Arc arity	1
Arc constraint(s)	variables.var = variables.key
Graph property(ies)	**NARC** = NLOOP

Graph model Since, within the context of the first graph constraint, we want to express one unary constraint for each value we use the “For all items of VALUES” iterator. Part (A) of Figure 5.374 shows the initial graphs associated with each value 1, 5 and 6 of the VALUES collection of the **Example** slot. Part (B) of Figure 5.374 shows the two corresponding final graphs respectively associated with values 1 and 6 that are both assigned to the variables of the VARIABLES collection (since value 5 is not assigned to any variable of the VARIABLES collection the final graph associated with value 5 is empty). Since we use the **NVERTEX** graph property, the vertices of the final graphs are stressed in bold.

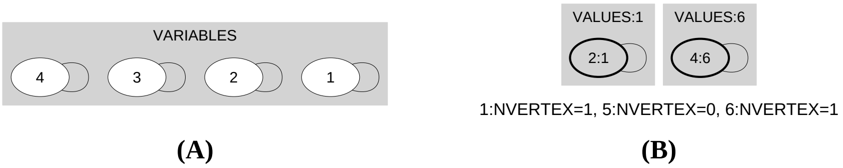

Figure 5.374: Initial and final graph of the global_cardinality_no_loop constraint