5.45 balance_interval

DESCRIPTION LINKS GRAPH AUTOMATON

Origin Derived from balance.

Constraint balance_interval(BALANCE, VARIABLES, SIZE_INTERVAL)

Arguments

BALANCE	:	dvar
VARIABLES	:	collection(var−dvar)
SIZE_INTERVAL	:	int

Restrictions

BALANCE ≥ 0
BALANCE $\leq$ max$(0, |$VARIABLES$| - 2)$
required(VARIABLES, var)
SIZE_INTERVAL > 0

Purpose Consider the largest set $\mathcal{S}_1$ (respectively the smallest set $\mathcal{S}_2$) of variables of the collection VARIABLES that take their value in a same interval [SIZE_INTERVAL$\cdot k$, SIZE_INTERVAL$\cdot k +$ SIZE_INTERVAL $- 1$], where k is an integer. BALANCE is equal to the difference between the cardinality of $\mathcal{S}_2$ and the cardinality of $\mathcal{S}_1$.

Example $(3, \langle 6, 4, 3, 3, 4\rangle, 3)$

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. Values 6,4,3,3 and 4 are respectively located within intervals $[6, 8]$, $[3, 5]$, $[3, 5]$, $[3, 5]$ and $[3, 5]$. Therefore intervals $[6, 8]$ and $[3, 5]$ are respectively used 1 and 4 times. The balance_interval constraint holds since its first argument BALANCE is assigned to the difference between the maximum and minimum number of the previous occurrences (i.e., $4 - 1$).

Typical

$|$VARIABLES$| > 2$
SIZE_INTERVAL > 1
SIZE_INTERVAL $<$range(VARIABLES.var)

Symmetries

- Items of VARIABLES are permutable.
- An occurrence of a value of VARIABLES.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

Functional dependency: BALANCE determined by VARIABLES and SIZE_INTERVAL.

Usage An application of the balance_interval constraint is to enforce a *balanced assignment* of interval of values, no matter how many distinct interval of values will be used. In this case one will *push down* the maximum value of the first argument of the balance_interval constraint.

See also **specialisation:** balance (variable/constant *replaced by* variable).

Keywords

application area: assignment.

characteristic of a constraint: automaton, automaton with array of counters.

constraint arguments: pure functional dependency.

constraint type: value constraint.

final graph structure: equivalence.

modelling: interval, balanced assignment, functional dependency.

Arc input(s)	VARIABLES
Arc generator	$CLIQUE \mapsto$ collection(variables1, variables2)
Arc arity	2
Arc constraint(s)	variables1.var/SIZE_INTERVAL = variables2.var/SIZE_INTERVAL
Graph property(ies)	RANGE_NSCC= BALANCE
Graph class	EQUIVALENCE

Graph model The graph property RANGE_NSCC constraints the difference between the sizes of the largest and smallest strongly connected components.

Parts (A) and (B) of Figure 5.103 respectively show the initial and final graph associated with the **Example** slot. Since we use the RANGE_NSCC graph property, we show the largest and smallest strongly connected components of the final graph.

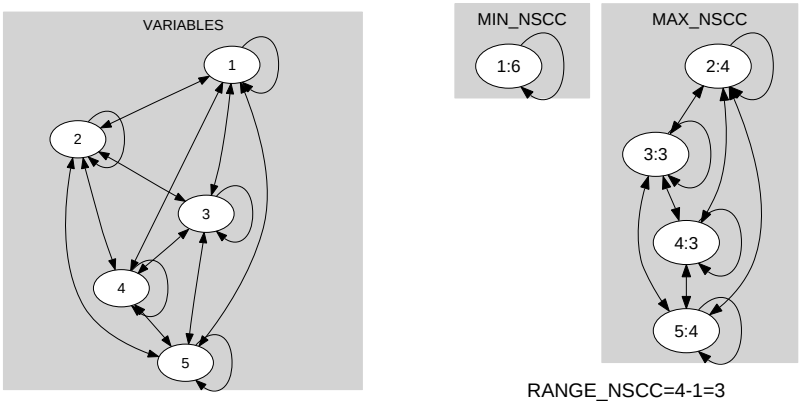

Figure 5.103: Initial and final graph of the balance_interval constraint

Automaton

Figure 5.104 depicts the automaton associated with the `balance_interval` constraint. To each item of the collection `VARIABLES` corresponds a signature variable S_i that is equal to 1.

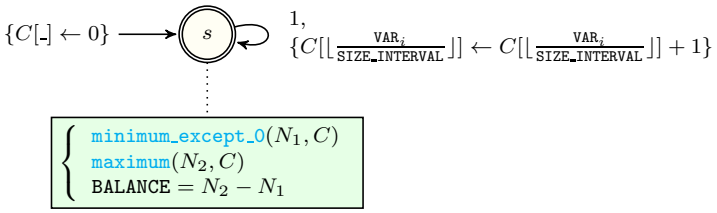

Figure 5.104: Automaton of the `balance_interval` constraint