5.26 among_low_up

DESCRIPTION LINKS GRAPH AUTOMATON

Origin [41]

Constraint among_low_up(LOW, UP, VARIABLES, VALUES)

Arguments

LOW	:	int
UP	:	int
VARIABLES	:	collection(var−dvar)
VALUES	:	collection(val−int)

Restrictions

LOW ≥ 0
LOW $\leq$ |VARIABLES|
UP ≥ 0
UP $\leq$ |VARIABLES|
UP $\geq$ LOW
required(VARIABLES, var)
required(VALUES, val)
distinct(VALUES, val)

Purpose Between LOW and UP variables of the VARIABLES collection are assigned a value of the VALUES collection.

Example $(1, 2, \langle 9, 2, 4, 5\rangle, \langle 0, 2, 4, 6, 8\rangle)$

The among_low_up constraint holds since between 1 and 2 values (i.e., in fact 2 values) of the collection of values $\langle 9, 2, 4, 5\rangle$ belong to the set of values $\{0, 2, 4, 6, 8\}$.

All solutions Figure 5.68 gives all solutions to the following non ground instance of the among_low_up constraint: $V_1 \in [1, 2]$, $V_2 \in [8, 9]$, $V_3 \in [5, 6]$, $V_4 \in [2, 3]$, among_low_up($\mathbf{3}, \mathbf{4}, \langle V_1, V_2, V_3, V_4\rangle, \langle \mathbf{0}, \mathbf{2}, \mathbf{4}, \mathbf{6}, \mathbf{8}\rangle$).

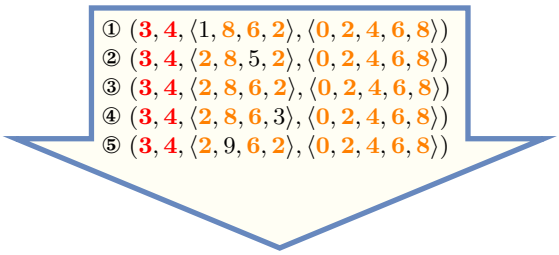

Figure 5.68: All solutions corresponding to the non ground example of the among_low_up constraint of the **All solutions** slot, where at least three variables (LOW $= \mathbf{3}$, UP $= \mathbf{4}$) are assigned a value from $\{\mathbf{0}, \mathbf{2}, \mathbf{4}, \mathbf{6}, \mathbf{8}\}$

Typical

$\texttt{LOW} < |\texttt{VARIABLES}|$
$\texttt{UP} > 0$
$\texttt{LOW} < \texttt{UP}$
$|\texttt{VARIABLES}| > 1$
$|\texttt{VALUES}| > 1$
$|\texttt{VARIABLES}| > |\texttt{VALUES}|$
$\texttt{LOW} > 0 \vee \texttt{UP} < |\texttt{VARIABLES}|$

Symmetries

- Items of VARIABLES are permutable.
- Items of VALUES are permutable.
- LOW can be decreased to any value ≥ 0.
- UP can be increased to any value $\leq |\texttt{VARIABLES}|$.
- An occurrence of a value of VARIABLES.var that belongs to VALUES.val (resp. does not belong to VALUES.val) can be replaced by any other value in VALUES.val (resp. not in VALUES.val).

Arg. properties

- Contractible wrt. VARIABLES when $\texttt{UP} = 0$.
- Contractible wrt. VARIABLES when $\texttt{UP} = |\texttt{VARIABLES}|$.
- Aggregate: LOW(+), UP(+), VARIABLES(union), VALUES(sunion).

Algorithm

The among_low_up constraint is entailed if and only if the following two conditions hold:

1. The number of variables of the VARIABLES collection assigned a value of the VALUES collection is greater than or equal to LOW.
2. The number of variables of the VARIABLES collection that can potentially be assigned a value of the VALUES collection is less than or equal to UP.

Used in

among_seq, cycle_card_on_path, interval_and_count, sliding_card_skip0.

See also

assignment dimension added: interval_and_count *(assignment dimension corresponding to intervals added)*.

generalisation: among *(*interval *replaced by* variable*)*, sliding_card_skip0 *(full sequence replaced by maximal sequences of non-zeros)*.

system of constraints: among_seq.

Keywords

characteristic of a constraint: automaton, automaton with counters.

constraint network structure: alpha-acyclic constraint network(2).

constraint type: value constraint, counting constraint.

filtering: arc-consistency, entailment.

final graph structure: acyclic, bipartite, no loop.

Cond. implications

among_low_up(LOW, UP, VARIABLES, VALUES)
with distinct(VARIABLES, var)
implies among_low_up(LOW, UP, VALUES, VARIABLES).

Arc input(s)	VARIABLES VALUES
Arc generator	$PRODUCT \mapsto$ collection(variables, values)
Arc arity	2
Arc constraint(s)	variables.var = values.val
Graph property(ies)	• NARC$\geq$ LOW • NARC$\leq$ UP
Graph class	• ACYCLIC • BIPARTITE • NO_LOOP

Graph model Each arc constraint of the final graph corresponds to the fact that a variable is assigned to a value that belong to the VALUES collection. The two graph properties restrict the total number of arcs to the interval [LOW, UP].

Parts (A) and (B) of Figure 5.69 respectively show the initial and final graph associated with the **Example** slot. Since we use the NARC graph property, the arcs of the final graph are stressed in bold.

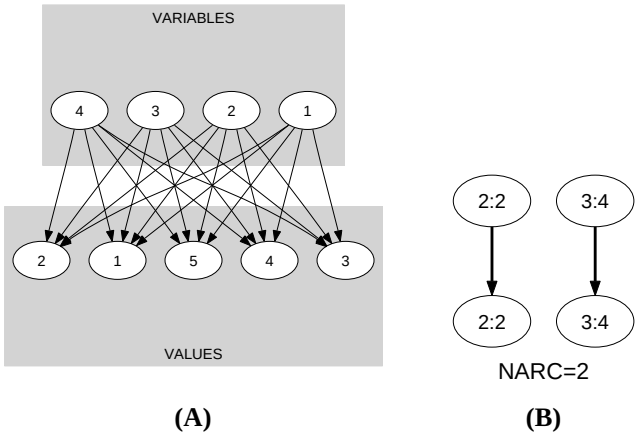

Figure 5.69: Initial and final graph of the among_low_up constraint

Automaton

Figure 5.70 depicts the automaton associated with the among_low_up constraint. To each variable $\texttt{VAR}_i$ of the collection VARIABLES corresponds a 0-1 signature variable S_i. The following signature constraint links $\texttt{VAR}_i$ and S_i: $\texttt{VAR}_i \in \texttt{VALUES} \Leftrightarrow S_i$. The automaton counts the number of variables of the VARIABLES collection that take their value in VALUES and finally checks that this number is within the interval $[\texttt{LOW}, \texttt{UP}]$.

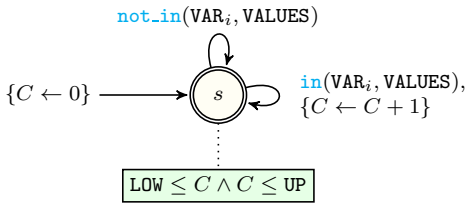

Figure 5.70: Automaton of the among_low_up constraint

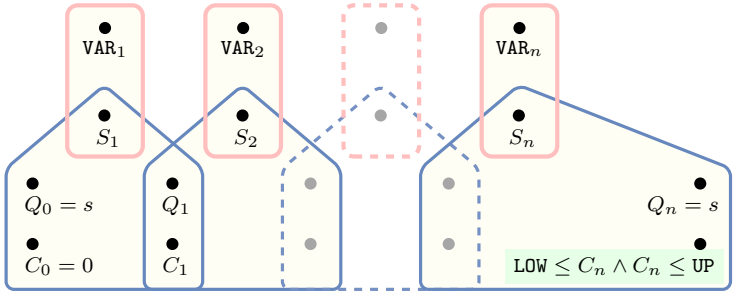

Figure 5.71: Hypergraph of the reformulation corresponding to the automaton (with one counter) of the among_low_up constraint: since all states variables $Q_0, Q_1, \ldots, Q_n$ are fixed to the unique state s of the automaton, the transitions constraints share only the counter variable C and the constraint network is Berge-acyclic