

5.40 `between_min_max`

| | DESCRIPTION | LINKS | GRAPH | AUTOMATON |
|----------------------|---|-------|-------|-----------|
| Origin | Used for defining <code>cumulative_convex</code> . | | | |
| Constraint | <code>between_min_max(VAR, VARIABLES)</code> | | | |
| Arguments | VAR : <code>dvar</code> VARIABLES : <code>collection(var-dvar)</code> | | | |
| Restrictions | <code>required(VARIABLES, var)</code> $ \text{VARIABLES} > 0$ | | | |
| Purpose | VAR is greater than or equal to at least one variable of the collection VARIABLES and less than or equal to at least one variable of the collection VARIABLES. | | | |
| Example | <code>(3, ⟨1, 1, 4, 8⟩)</code> The <code>between_min_max</code> constraint holds since its first argument 3 is greater than or equal to the minimum value of the values of the collection <code>⟨1, 1, 4, 8⟩</code> and less than or equal to the maximum value of <code>⟨1, 1, 4, 8⟩</code> . | | | |
| Typical | $ \text{VARIABLES} > 1$ <code>range(VARIABLES.var) > 1</code> | | | |
| Symmetries | <ul style="list-style-type: none"> Items of VARIABLES are <code>permutable</code>. VAR can be <code>set</code> to any value of VARIABLES.var. | | | |
| Reformulation | By introducing two extra variables MIN and MAX, the <code>between_min_max(VAR, VARIABLES)</code> constraint can be expressed in term of the following conjunction of constraints: <code>minimum(MIN, VARIABLES)</code> , <code>maximum(MAX, VARIABLES)</code> , $\text{VAR} \geq \text{MIN}$, $\text{VAR} \leq \text{MAX}$. | | | |
| Used in | <code>cumulative_convex</code> . | | | |
| See also | implied by: <code>maximum</code> , <code>minimum</code> . | | | |
| Keywords | characteristic of a constraint: <code>automaton</code> , <code>automaton without counters</code> , <code>reified automaton constraint</code> . constraint network structure: <code>centered cyclic(1) constraint network(1)</code> . | | | |

Derived Collection

$$\text{col}(\text{ITEM} \rightarrow \text{collection}(\text{var} \rightarrow \text{dvar}), [\text{item}(\text{var} - \text{VAR})])$$
Arc input(s)

ITEM VARIABLES

Arc generator $PRODUCT \mapsto \text{collection}(\text{item}, \text{variables})$ **Arc arity**

2

Arc constraint(s) $\text{item.var} \geq \text{variables.var}$ **Graph property(ies)****NARC** ≥ 1 **Graph class**

- ACYCLIC
- BIPARTITE
- NO_LOOP

Arc input(s)

ITEM VARIABLES

Arc generator $PRODUCT \mapsto \text{collection}(\text{item}, \text{variables})$ **Arc arity**

2

Arc constraint(s) $\text{item.var} \leq \text{variables.var}$ **Graph property(ies)****NARC** ≥ 1 **Graph class**

- ACYCLIC
- BIPARTITE
- NO_LOOP

Graph model

Parts (A) and (B) of Figure 5.73 respectively show the initial and final graph associated with the second graph constraint of the **Example** slot. Since we use the **NARC** graph property, the two arcs of the final graph are stressed in bold. The constraint holds since 3 is greater than 1 and since 3 is less than 8.

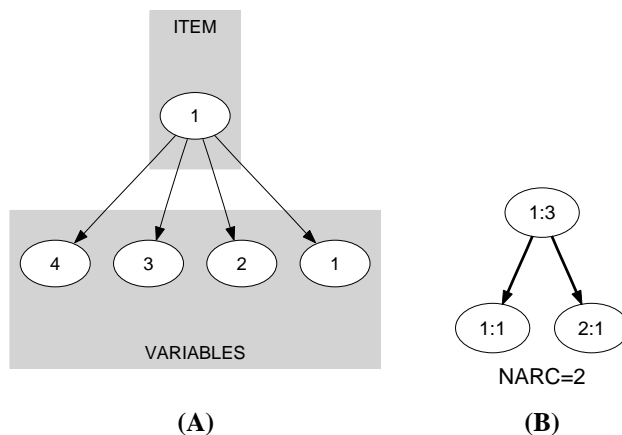


Figure 5.73: Initial and final graph of the between_min_max constraint

Automaton

Figure 5.74 depicts the automaton associated with the `between_min_max` constraint. To each pair $(\text{VAR}, \text{VAR}_i)$, where VAR_i is a variable of the collection `VARIABLES` corresponds a signature variable S_i . The following signature constraint links VAR , VAR_i and S_i : $(\text{VAR} < \text{VAR}_i \Leftrightarrow S_i = 0) \wedge (\text{VAR} = \text{VAR}_i \Leftrightarrow S_i = 1) \wedge (\text{VAR} > \text{VAR}_i \Leftrightarrow S_i = 2)$.

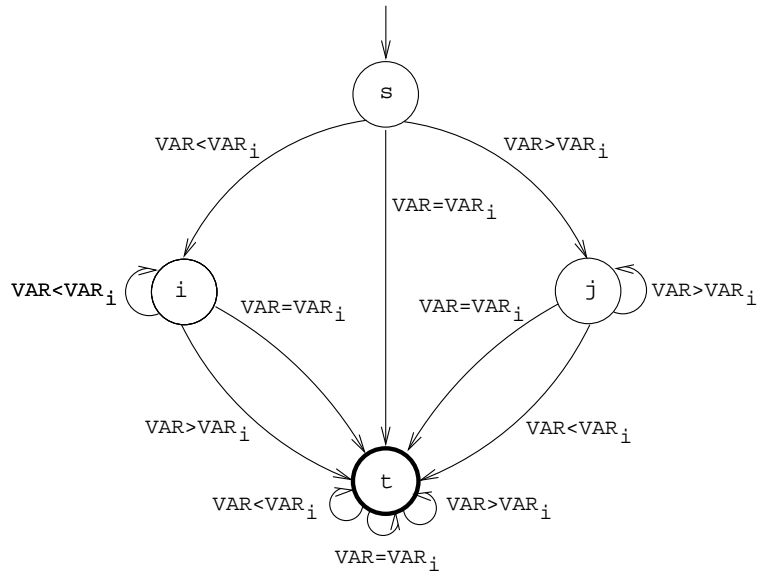


Figure 5.74: Automaton of the `between_min_max` constraint

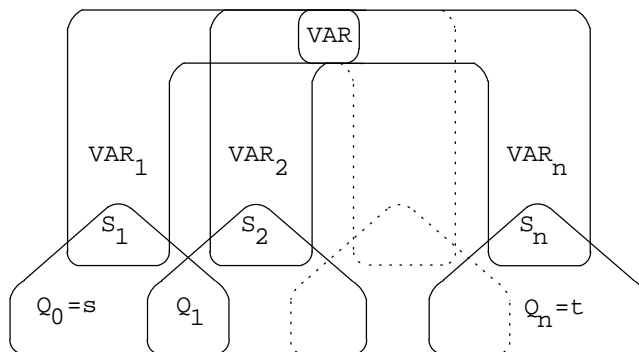


Figure 5.75: Hypergraph of the reformulation corresponding to the automaton of the `between_min_max` constraint