5.140 element_greatereq

Origin	[301]
Constraint	element_greatereq(ITEM, TABLE)
Arguments	ITEM : collection(index−dvar, value−dvar) TABLE : collection(index−int, value−int)
Restrictions	required(ITEM, [index, value]) ITEM.index ≥ 1 ITEM.index ≤ |TABLE| |ITEM| = 1 |TABLE| > 0 required(TABLE, [index, value]) TABLE.index ≥ 1 TABLE.index ≤ |TABLE| distinct(TABLE, index)
Purpose	ITEM[1].value is greater than or equal to one of the entries (i.e., the value attribute) of the table TABLE.
Example	$$\left(\begin{array}{l}\langle\text{index}-1\ \text{value}-8\rangle,\\ \left\langle\begin{array}{ll}\text{index}-1 & \text{value}-6,\\ \text{index}-2 & \text{value}-9,\\ \text{index}-3 & \text{value}-2,\\ \text{index}-4 & \text{value}-9\end{array}\right\rangle\end{array}\right)$$ The element_greatereq constraint holds since ITEM[1].value = 8 is greater than or equal to TABLE[ITEM[1].index].value = TABLE[1].value = 6.
Typical	|TABLE| > 1 range(TABLE.value) > 1
Symmetries	• Items of TABLE are permutable. • All occurrences of two distinct values in ITEM.value or TABLE.value can be swapped; all occurrences of a value in ITEM.value or TABLE.value can be renamed to any unused value.
Usage	Used for modelling variable subscripts in linear constraints [301].
Reformulation	By introducing an extra variable VAL, the element_greatereq(⟨index − INDEX value − VALUE⟩, TABLE) constraint can be expressed in term of an elem(⟨index − INDEX value − VAL⟩, TABLE) constraint and of an inequality constraint VALUE ≥ VAL.

See also
    common keyword: `element`, `element_lesseq`, `element_product` *(array constraint)*.

    implied by: `elem`.

Keywords
    characteristic of a constraint: automaton, automaton without counters, reified automaton constraint.

    constraint arguments: binary constraint.

    constraint network structure: centered cyclic(2) constraint network(1).

    constraint type: data constraint.

    filtering: linear programming, arc-consistency.

    modelling: array constraint, table, variable subscript, variable indexing.

Arc input(s)	ITEM TABLE
Arc generator	$PRODUCT \mapsto$ collection(item, table)
Arc arity	2
Arc constraint(s)	• item.index = table.index • item.value $\geq$ table.value
Graph property(ies)	**NARC**= 1

Graph model   Similar to the element constraint except that the *equality* constraint of the second condition of the arc constraint is replaced by a *greater than or equal to* constraint.

Parts (A) and (B) of Figure 5.305 respectively show the initial and final graph associated with the **Example** slot. Since we use the **NARC** graph property, the unique arc of the final graph is stressed in bold.
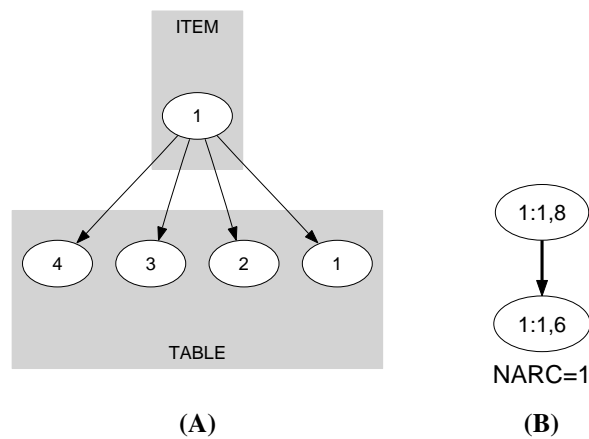

Figure 5.305: Initial and final graph of the element_greatereq constraint

Signature   Since all the index attributes of TABLE are distinct and because of the first arc constraint the final graph cannot have more than one arc. Therefore we can rewrite **NARC** $= 1$ to **NARC** ≥ 1 and simplify $\overline{\text{NARC}}$ to $\overline{\text{NARC}}$.

Automaton   Figure 5.306 depicts the automaton associated with the `element_greatereq` constraint. Let INDEX and VALUE respectively be the `index` and the `value` attributes of the unique item of the ITEM collection. Let $INDEX_i$ and $VALUE_i$ respectively be the `index` and the `value` attributes of the ith item of the TABLE collection. To each quadruple $(INDEX, VALUE, INDEX_i, VALUE_i)$ corresponds a 0-1 signature variable S_i as well as the following signature constraint: $((INDEX = INDEX_i) \wedge (VALUE \geq VALUE_i)) \Leftrightarrow S_i$.
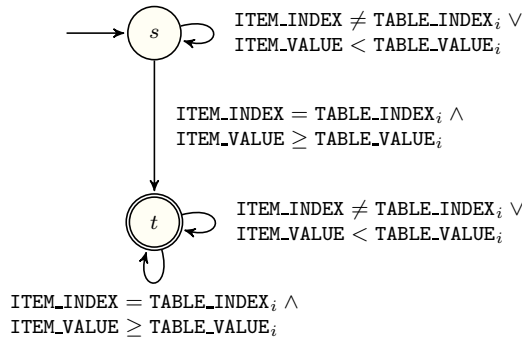

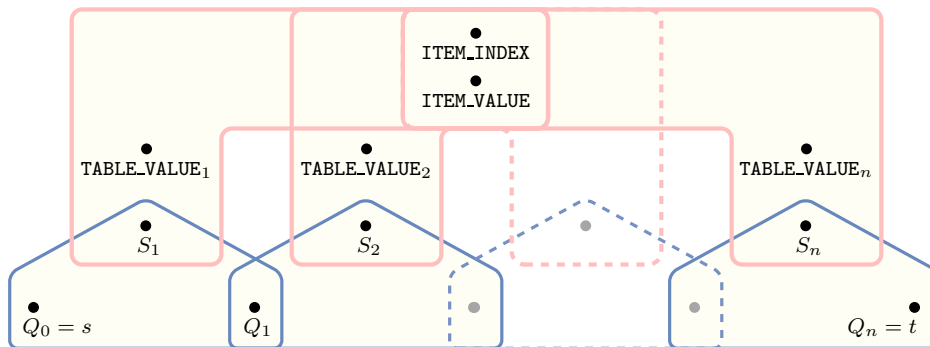Figure 5.306: Automaton of the `element_greatereq` constraint

Figure 5.307: Hypergraph of the reformulation corresponding to the automaton of the `element_greatereq` constraint