

5.141 element\_lesseq

	DESCRIPTION	LINKS	GRAPH	AUTOMATON
Origin	[301]			
Constraint	element_lesseq(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 less than or equal to one of the entries (i.e., the value attribute) of the table TABLE.			
Example	$\left( \begin{array}{c} \langle \text{index} - 3 \text{ value} - 1 \rangle, \\ \text{index} - 1 \quad \text{value} - 6, \\ \langle \text{index} - 2 \quad \text{value} - 9, \\ \text{index} - 3 \quad \text{value} - 2, \rangle \\ \text{index} - 4 \quad \text{value} - 9 \end{array} \right)$			
	The element_lesseq constraint holds since ITEM[1].value = 1 is less than or equal to TABLE[ITEM[1].index].value = TABLE[3].value = 2.			
Typical	TABLE  > 1 range(TABLE.value) > 1			
Symmetries	<ul style="list-style-type: none"><li>• Items of TABLE are <b>permutable</b>.</li><li>• All occurrences of two distinct values in ITEM.value or TABLE.value can be <b>swapped</b>; all occurrences of a value in ITEM.value or TABLE.value can be <b>renamed</b> to any unused value.</li></ul>			
Usage	Used for modelling variable subscripts in linear constraints [301].			
Reformulation	By introducing an extra variable VAL, the element_lesseq(⟨index - INDEX value - VALUE⟩, TABLE) constraint can be expressed in term of an <b>elem</b> (⟨index - INDEX value - VAL⟩, TABLE) constraint and of an inequality constraint VALUE ≤ VAL.			

See also	<b>common keyword:</b> <code>element</code> , <code>element_greatereq</code> , <code>element_product</code> ( <i>array constraint</i> ). <b>implied by:</b> <code>elem</code> .
Keywords	<b>characteristic of a constraint:</b> <code>automaton</code> , <code>automaton without counters</code> , <code>reified automaton constraint</code> . <b>constraint arguments:</b> <code>binary constraint</code> . <b>constraint network structure:</b> <code>centered cyclic(2) constraint network(1)</code> . <b>constraint type:</b> <code>data constraint</code> . <b>filtering:</b> <code>linear programming</code> , <code>arc-consistency</code> . <b>modelling:</b> <code>array constraint</code> , <code>table</code> , <code>variable subscript</code> , <code>variable indexing</code> .
Cond. implications	<code>element_lesseq</code> (ITEM, TABLE) with <code>minval</code> (ITEM.value) > 0 and TABLE.value > 0 <b>implies</b> <code>bin_packing_capa</code> (BINS : TABLE, ITEMS : ITEM).

Arc input(s)	ITEM TABLE
Arc generator	<i>PRODUCT</i> $\mapsto$ collection(item, table)
Arc arity	2
Arc constraint(s)	<ul style="list-style-type: none"><li>• item.index = table.index</li><li>• item.value <math>\leq</math> table.value</li></ul>
Graph property(ies)	<u>NARC</u> = 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 *less than or equal to* constraint.

Parts (A) and (B) of Figure 5.325 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.

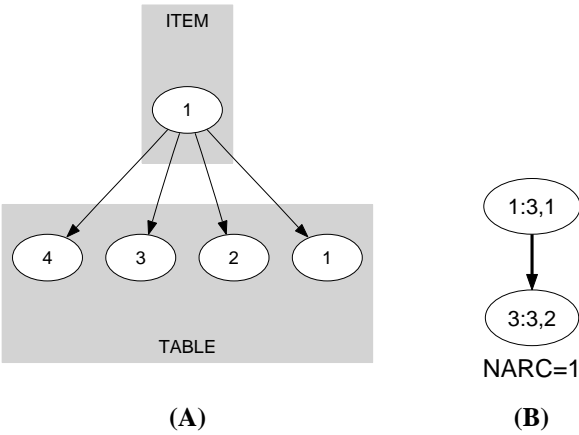
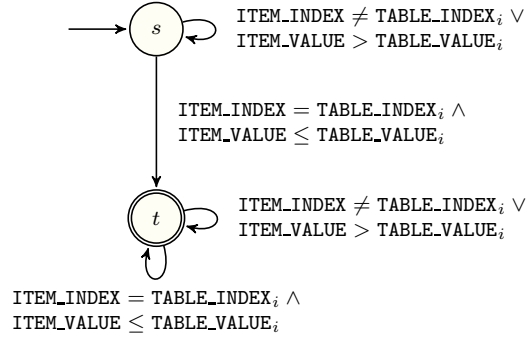
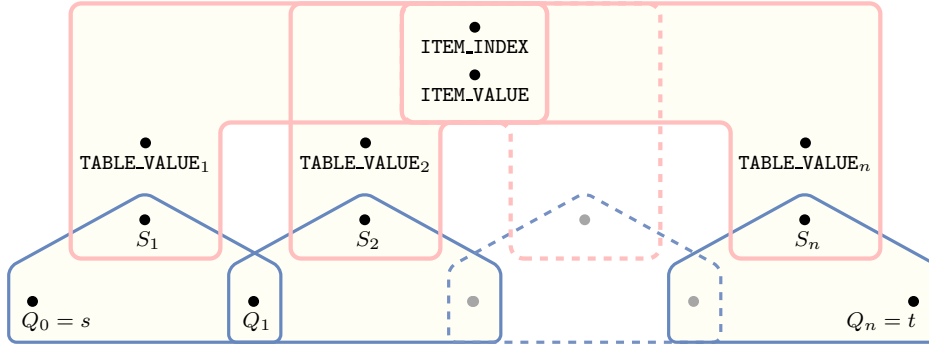


Figure 5.325: Initial and final graph of the *element\_lesseq* constraint

**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  $\text{NARC} = 1$  to  $\text{NARC} \geq 1$  and simplify NARC to NARC.

**Automaton**

Figure 5.326 depicts the automaton associated with the `element_lesseq` constraint. Let `INDEX` and `VALUE` respectively be the index and the value attributes of the unique item of the `ITEM` collection. Let `INDEXi` and `VALUEi` respectively be the index and the value attributes of the  $i^{th}$  item of the `TABLE` collection. To each quadruple  $(\text{INDEX}, \text{VALUE}, \text{INDEX}_i, \text{VALUE}_i)$  corresponds a 0-1 signature variable  $S_i$  as well as the following signature constraint:  $((\text{INDEX} = \text{INDEX}_i) \wedge (\text{VALUE} \leq \text{VALUE}_i)) \Leftrightarrow S_i$ .

Figure 5.326: Automaton of the `element_lesseq` constraintFigure 5.327: Hypergraph of the reformulation corresponding to the automaton of the `element_lesseq` constraint