5.275 next_element

DESCRIPTION LINKS GRAPH AUTOMATON

Origin N. Beldiceanu

Constraint next_element(THRESHOLD, INDEX, TABLE, VAL)

Arguments

THRESHOLD	:	dvar
INDEX	:	dvar
TABLE	:	collection(index−int, value−dvar)
VAL	:	dvar

Restrictions

INDEX ≥ 1
INDEX $\leq$ |TABLE|
THRESHOLD $<$ INDEX
required(TABLE, [index, value])
|TABLE| > 0
TABLE.index ≥ 1
TABLE.index $\leq$ |TABLE|
distinct(TABLE, index)

Purpose INDEX is the smallest entry of TABLE strictly greater than THRESHOLD containing value VAL.

Example

$$\left(\begin{array}{c} \\ 2, 3, \left\langle \begin{array}{ll} \text{index} - 1 & \text{value} - 1, \\ \text{index} - 2 & \text{value} - 8, \\ \text{index} - 3 & \text{value} - 9, \\ \text{index} - 4 & \text{value} - 5, \\ \text{index} - 5 & \text{value} - 9 \end{array} \right\rangle, 9 \\ \\ \end{array}\right)$$

The next_element constraint holds since 3 is the smallest entry located after entry 2 that contains value 9.

Typical

|TABLE| > 1
range(TABLE.value) > 1

Usage Originally introduced for modelling the fact that a nucleotide has to be consumed as soon as possible at cycle INDEX after a given cycle represented by variable THRESHOLD.

See also **related:** minimum_greater_than *(identify an element in a table)*, next_greater_element *(allow to iterate over the values of a table)*.

Keywords **characteristic of a constraint:** minimum, automaton, automaton without counters, reified automaton constraint, derived collection.

constraint network structure: centered cyclic(3) constraint network(1).

constraint type: data constraint.

modelling: table.

Derived Collection	col(ITEM−collection(index−dvar, value−dvar), [item(index − THRESHOLD, value − VAL)])
Arc input(s)	ITEM TABLE
Arc generator	*PRODUCT* ↦ collection(item, table)
Arc arity	2
Arc constraint(s)	• item.index < table.index • item.value = table.value
Graph property(ies)	**NARC** > 0
Sets	SUCC ↦ [source, variables − col(VARIABLES−collection(var−dvar), [item(var − TABLE.index)])]
Constraint(s) on sets	minimum(INDEX, variables)

Graph model Parts (A) and (B) of Figure 5.575 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 arcs of the final graph are stressed in bold.

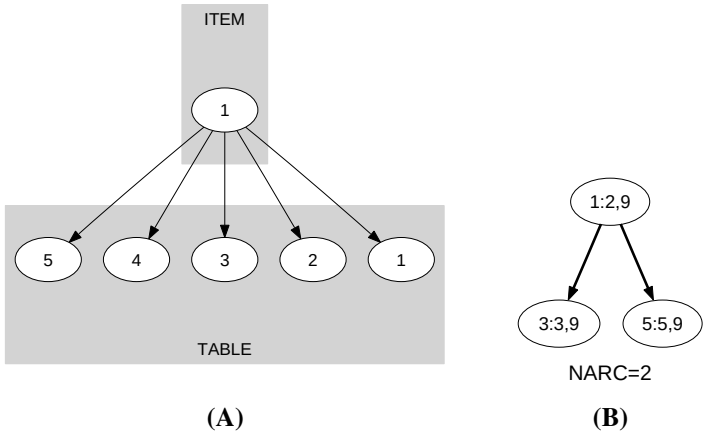

Figure 5.575: Initial and final graph of the next_element constraint

Automaton

Figure 5.576 depicts the automaton associated with the next_element constraint. Let $\mathtt{I}_k$ and $\mathtt{V}_k$ respectively be the index and the value attributes of the k^{th} item of the TABLE collections. To each quintuple (THRESHOLD, INDEX, VAL, $\mathtt{I}_k$, $\mathtt{V}_k$) corresponds a signature variable S_k as well as the following signature constraint:

$$((\mathtt{I}_k \leq \mathtt{THRESHOLD}) \wedge (\mathtt{I}_k < \mathtt{INDEX}) \wedge (\mathtt{V}_k = \mathtt{VAL})) \Leftrightarrow S_k = 0 \wedge$$
$$((\mathtt{I}_k \leq \mathtt{THRESHOLD}) \wedge (\mathtt{I}_k < \mathtt{INDEX}) \wedge (\mathtt{V}_k \neq \mathtt{VAL})) \Leftrightarrow S_k = 1 \wedge$$
$$((\mathtt{I}_k \leq \mathtt{THRESHOLD}) \wedge (\mathtt{I}_k = \mathtt{INDEX}) \wedge (\mathtt{V}_k = \mathtt{VAL})) \Leftrightarrow S_k = 2 \wedge$$
$$((\mathtt{I}_k \leq \mathtt{THRESHOLD}) \wedge (\mathtt{I}_k = \mathtt{INDEX}) \wedge (\mathtt{V}_k \neq \mathtt{VAL})) \Leftrightarrow S_k = 3 \wedge$$
$$((\mathtt{I}_k \leq \mathtt{THRESHOLD}) \wedge (\mathtt{I}_k > \mathtt{INDEX}) \wedge (\mathtt{V}_k = \mathtt{VAL})) \Leftrightarrow S_k = 4 \wedge$$
$$((\mathtt{I}_k \leq \mathtt{THRESHOLD}) \wedge (\mathtt{I}_k > \mathtt{INDEX}) \wedge (\mathtt{V}_k \neq \mathtt{VAL})) \Leftrightarrow S_k = 5 \wedge$$
$$((\mathtt{I}_k > \mathtt{THRESHOLD}) \wedge (\mathtt{I}_k < \mathtt{INDEX}) \wedge (\mathtt{V}_k = \mathtt{VAL})) \Leftrightarrow S_k = 6 \wedge$$
$$((\mathtt{I}_k > \mathtt{THRESHOLD}) \wedge (\mathtt{I}_k < \mathtt{INDEX}) \wedge (\mathtt{V}_k \neq \mathtt{VAL})) \Leftrightarrow S_k = 7 \wedge$$
$$((\mathtt{I}_k > \mathtt{THRESHOLD}) \wedge (\mathtt{I}_k = \mathtt{INDEX}) \wedge (\mathtt{V}_k = \mathtt{VAL})) \Leftrightarrow S_k = 8 \wedge$$
$$((\mathtt{I}_k > \mathtt{THRESHOLD}) \wedge (\mathtt{I}_k = \mathtt{INDEX}) \wedge (\mathtt{V}_k \neq \mathtt{VAL})) \Leftrightarrow S_k = 9 \wedge$$
$$((\mathtt{I}_k > \mathtt{THRESHOLD}) \wedge (\mathtt{I}_k > \mathtt{INDEX}) \wedge (\mathtt{V}_k = \mathtt{VAL})) \Leftrightarrow S_k = 10 \wedge$$
$$((\mathtt{I}_k > \mathtt{THRESHOLD}) \wedge (\mathtt{I}_k > \mathtt{INDEX}) \wedge (\mathtt{V}_k \neq \mathtt{VAL})) \Leftrightarrow S_k = 11.$$

The automaton is constructed in order to fulfil the following conditions:

- We look for an item of the TABLE collection such that $\mathtt{INDEX}_i > \mathtt{THRESHOLD}$ and $\mathtt{INDEX}_i = \mathtt{INDEX}$ and $\mathtt{VALUE}_i = \mathtt{VAL}$,
- There should not exist any item of the TABLE collection such that $\mathtt{INDEX}_i > \mathtt{THRESHOLD}$ and $\mathtt{INDEX}_i < \mathtt{INDEX}$ and $\mathtt{VALUE}_i = \mathtt{VAL}$.

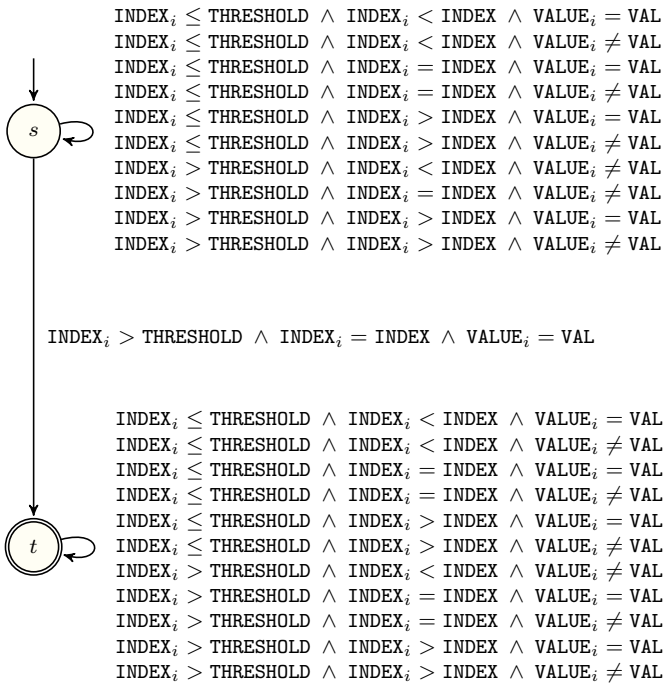

Figure 5.576: Automaton of the `next_element` constraint

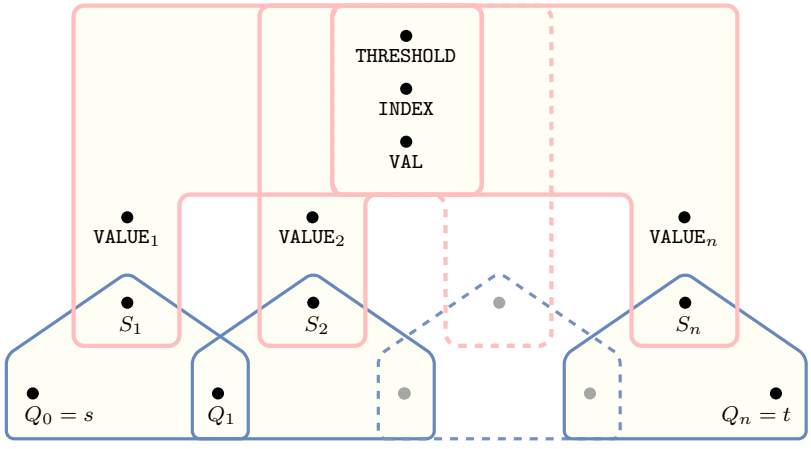

Figure 5.577: Hypergraph of the reformulation corresponding to the automaton of the `next_element` constraint