5.284 not_in

DESCRIPTION LINKS GRAPH AUTOMATON

Origin Derived from in.

Constraint not_in(VAR, VALUES)

Arguments

VAR	:	dvar
VALUES	:	collection(val−int)

Restrictions

required(VALUES, val)
distinct(VALUES, val)

Purpose Enforce VAR to be assigned a value different from the values of the VALUES collection.

Example $(2, \langle 1, 3 \rangle)$

The constraint not_in holds since the value of its first argument VAR $= 2$ does not occur within the collection $\langle 1, 3 \rangle$.

Typical $|\text{VALUES}| > 1$

Symmetries

- Items of VALUES are permutable.
- One and the same constant can be added to VAR as well as to the val attribute of all items of VALUES.

Arg. properties

Contractible wrt. VALUES.

Remark Entailment occurs immediately after posting this constraint and removing all values in VALUES from VAR.

Systems notMember in **Choco**, rel in **Gecode**.

Used in group.

See also **negation:** in.

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

constraint arguments: unary constraint.

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

constraint type: value constraint.

filtering: arc-consistency, entailment.

modelling: excluded, domain definition.

Derived Collection	col(VARIABLES−collection(var−dvar), [item(var − VAR)])
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**= 0

Graph model Figure 5.611 shows the initial graph associated with the **Example** slot. Since we use the **NARC** = 0 graph property the corresponding final graph is empty.

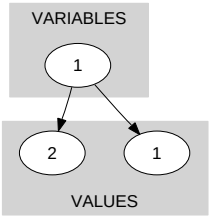

Figure 5.611: Initial graph of the not_in constraint (the final graph is empty)

Signature Since 0 is the smallest number of arcs of the final graph we can rewrite **NARC** = 0 to **NARC** ≤ 0. This leads to simplify $\overline{\textbf{NARC}}$ to $\underline{\textbf{NARC}}$.

Automaton

Figure 5.612 depicts the automaton associated with the not_in constraint. Let $\texttt{VAL}_i$ be the val attribute of the i^{th} item of the VALUES collection. To each pair $(\texttt{VAR}, \texttt{VAL}_i)$ corresponds a 0-1 signature variable S_i as well as the following signature constraint: $\texttt{VAR} = \texttt{VAL}_i \Leftrightarrow S_i$.

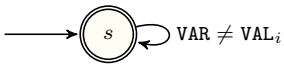

Figure 5.612: Automaton of the not_in constraint

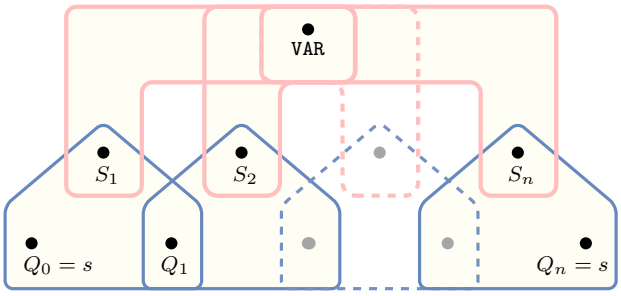

Figure 5.613: Hypergraph of the reformulation corresponding to the automaton of the not_in constraint