5.70 clause_or

DESCRIPTION LINKS AUTOMATON

Origin Logic

Constraint clause_or(POSVARS, NEGVARS, VAR)

Synonym clause.

Arguments

POSVARS	:	collection(var−dvar)
NEGVARS	:	collection(var−dvar)
VAR	:	dvar

Restrictions

$|\texttt{POSVARS}| + |\texttt{NEGVARS}| > 0$
required(POSVARS, var)
$\texttt{POSVARS.var} \geq 0$
$\texttt{POSVARS.var} \leq 1$
required(NEGVARS, var)
$\texttt{NEGVARS.var} \geq 0$
$\texttt{NEGVARS.var} \leq 1$
$\texttt{VAR} \geq 0$
$\texttt{VAR} \leq 1$

Purpose Given a first collection of 0-1 variables $\texttt{POSVARS} = U_1, U_2, \ldots, U_p$, a second collection of 0-1 variables $\texttt{NEGVARS} = V_1, V_2, \ldots, V_n$, and a variable VAR, enforce $\texttt{VAR} = (U_1 \vee U_2 \vee \cdots \vee U_p) \vee (\neg V_1 \vee \neg V_2 \vee \cdots \vee \neg V_n)$.

Example $(\langle 0, 0\rangle, \langle 0\rangle, 1)$

Typical $|\texttt{POSVARS}| + |\texttt{NEGVARS}| > 1$

Symmetries

- Items of POSVARS are permutable.
- Items of NEGVARS are permutable.

Arg. properties

- Extensible wrt. POSVARS when $\texttt{VAR} = 1$.
- Extensible wrt. NEGVARS when $\texttt{VAR} = 1$.

Remark The clause_or constraint is called clause in **Gecode** (http://www.gecode.org/).

Systems reifiedOr in **Choco**, clause in **Choco**, clause in **Gecode**.

See also **common keyword:** clause_and, or *(Boolean constraint)*.

Keywords

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

constraint network structure: Berge-acyclic constraint network.

constraint type: Boolean constraint.

filtering: arc-consistency.

modelling: disjunction.

Automaton

Figure 5.190 depicts the automaton associated with the clause_or constraint:

- To the argument VAR of the clause_or constraint corresponds the first signature variable.
- To each variable of the argument POSVARS corresponds a signature variable.
- Finally, to each variable $\texttt{VAR}_i$ of the argument NEGVARS corresponds a signature variable that is the negation of $\texttt{VAR}_i$.

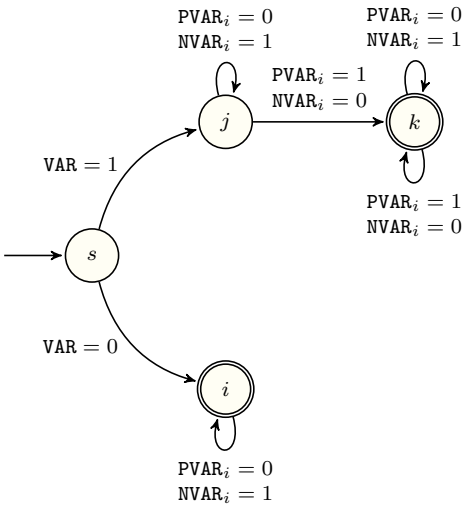

Figure 5.190: Automaton of the clause_or constraint ($\texttt{PVAR}_i$ and $\texttt{NVAR}_i$ respectively denote variables of POSVARS and NEGVARS)

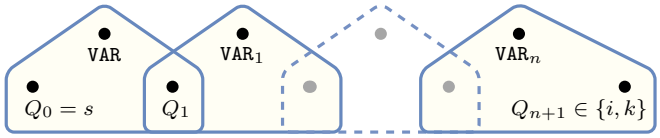

Figure 5.191: Hypergraph of the reformulation corresponding to the automaton of the clause_or constraint ($\texttt{VAR}_1, \ldots, \texttt{VAR}_n$ denotes $\texttt{PVAR}_1, \ldots, \texttt{PVAR}_{|\texttt{POSVARS}|}, 1 - \texttt{NVAR}_1, \ldots, 1 - \texttt{NVAR}_{|\texttt{NEGVARS}|}$)