

5.23 and

| | DESCRIPTION | LINKS | AUTOMATON |
|--------------|--|-------|-----------|
| Origin | Logic | | |
| Constraint | <code>and(VAR, VARIABLES)</code> | | |
| Synonym | <code>rel.</code> | | |
| Arguments | VAR : <code>dvar</code> VARIABLES : <code>collection(var-dvar)</code> | | |
| Restrictions | $VAR \geq 0$ $VAR \leq 1$ $ VARIABLES \geq 2$ <code>required(VARIABLES, var)</code> $VARIABLES.var \geq 0$ $VARIABLES.var \leq 1$ | | |
| Purpose | Let VARIABLES be a collection of 0-1 variables $VAR_1, VAR_2, \dots, VAR_n$ ($n \geq 2$). Enforce $VAR = VAR_1 \wedge VAR_2 \dots \wedge VAR_n$. | | |
| Example | $(0, \langle 0, 0 \rangle)$ $(0, \langle 0, 1 \rangle)$ $(0, \langle 1, 0 \rangle)$ $(1, \langle 1, 1 \rangle)$ $(0, \langle 1, 0, 1 \rangle)$ | | |
| Symmetry | Items of VARIABLES are <code>permutable</code> . | | |
| Systems | <code>reifiedAnd</code> in Choco , <code>rel</code> in Gecode , <code>andbool</code> in JaCoP , <code>#/\</code> in SICStus . | | |
| See also | common keyword: <code>clause_and</code> , <code>equivalent</code> , <code>imply</code> , <code>nand</code> , <code>nor</code> , <code>or</code> , <code>xor</code> (<i>Boolean constraint</i>). implies: <code>minimum</code> . | | |
| Keywords | characteristic of a constraint: <code>automaton</code> , <code>automaton without counters</code> , <code>reified automaton constraint</code> . constraint network structure: Berge-acyclic constraint network. constraint type: Boolean constraint. filtering: arc-consistency. | | |

Automaton

Figure 5.40 depicts the automaton associated with the **and** constraint. To the first argument **VAR** of the **and** constraint corresponds the first signature variable. To each variable VAR_i of the second argument **VARIABLES** of the **and** constraint corresponds the next signature variable. There is no signature constraint.

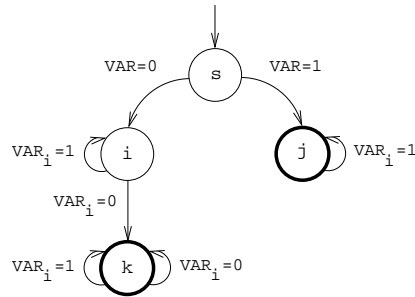


Figure 5.40: Automaton of the **and** constraint

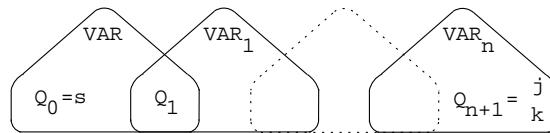


Figure 5.41: Hypergraph of the reformulation corresponding to the automaton of the **and** constraint