5.21 alldifferent_same_value

DESCRIPTION LINKS GRAPH AUTOMATON

Origin Derived from alldifferent.

Constraint alldifferent_same_value(NSAME, VARIABLES1, VARIABLES2)

Synonyms alldiff_same_value, alldistinct_same_value.

Arguments

NSAME	:	dvar
VARIABLES1	:	collection(var−dvar)
VARIABLES2	:	collection(var−dvar)

Restrictions

$\texttt{NSAME} \geq 0$
$\texttt{NSAME} \leq |\texttt{VARIABLES1}|$
$|\texttt{VARIABLES1}| = |\texttt{VARIABLES2}|$
required(VARIABLES1, var)
required(VARIABLES2, var)

Purpose All the values assigned to the variables of the collection VARIABLES1 are pairwise distinct. NSAME is equal to number of constraints of the form $\texttt{VARIABLES1}[i].\texttt{var} = \texttt{VARIABLES2}[i].\texttt{var}$ $(1 \leq i \leq |\texttt{VARIABLES1}|)$ that hold.

Example $(2, \langle 7, 3, 1, 5\rangle, \langle 1, 3, 1, 7\rangle)$

The alldifferent_same_value constraint holds since:

- All the values 7, 3, 1 and 5 are distinct,
- Among the four expressions $7 = 1$, $3 = 3$, $1 = 1$ and $5 = 7$ exactly 2 conditions hold.

All solutions Figure 5.49 gives all solutions to the following non ground instance of the alldifferent_same_value constraint: $\texttt{U}_1 \in [2, 4]$, $\texttt{U}_2 \in [1, 2]$, $\texttt{U}_3 \in [1, 4]$, $\texttt{U}_4 \in [2, 4]$, $\texttt{V}_1 \in [2, 3]$, $\texttt{V}_2 = 2$, $\texttt{V}_3 \in [0, 1]$, $\texttt{V}_4 \in [0, 3]$, alldifferent_same_value(3, $\langle \texttt{U}_1, \texttt{U}_2, \texttt{U}_3, \texttt{U}_4\rangle, \langle \texttt{V}_1, \texttt{V}_2, \texttt{V}_3, \texttt{V}_4\rangle$).

Typical

$\texttt{NSAME} < |\texttt{VARIABLES1}|$
$|\texttt{VARIABLES1}| > 2$

Symmetries

- Items of VARIABLES1 and VARIABLES2 are permutable (*same permutation used*).
- All occurrences of two distinct values in VARIABLES1.var or VARIABLES2.var can be swapped; all occurrences of a value in VARIABLES1.var or VARIABLES2.var can be renamed to any unused value.

Arg. properties

Functional dependency: NSAME determined by VARIABLES1 and VARIABLES2.

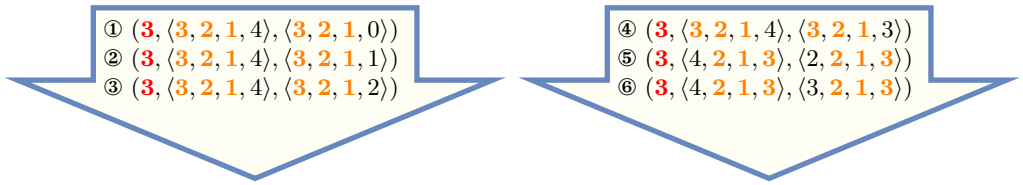


Figure 5.49: All solutions corresponding to the non ground example of the `alldifferent_same_value` constraint of the **All solutions** slot, where identical values at a same position in both collections are coloured in orange

Usage

When all variables of the second collection are initially bound to distinct values the `alldifferent_same_value` constraint can be explained in the following way:

- We interpret the variables of the second collection as the previous solution to a problem where all variables have to be distinct.
- We interpret the variables of the first collection as the current solution to find, where all variables should again be pairwise distinct.

The variable `NSAME` measures the `distance` of the current solution from the previous solution. This corresponds to the number of variables of `VARIABLES2` that are assigned to the same previous value.

See also

root concept: `alldifferent`.

Keywords

characteristic of a constraint: sort based reformulation, automaton, automaton with array of counters.

constraint type: proximity constraint.

modelling: functional dependency.

Cond. implications

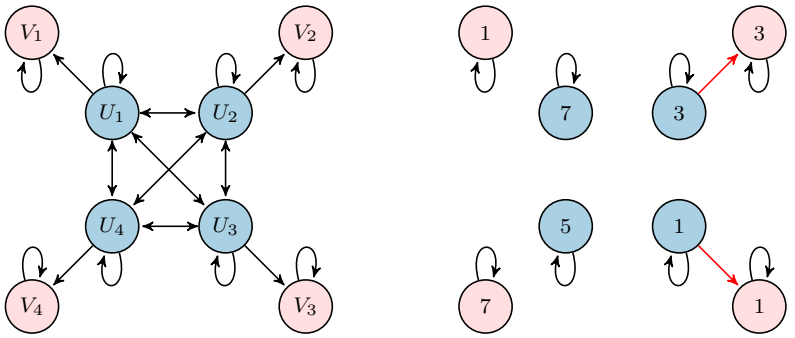
`alldifferent_same_value`(`NSAME`, `VARIABLES1`, `VARIABLES2`)
with $2 * \texttt{NSAME} = |\texttt{VARIABLES1}|$
implies `differ_from_exactly_k_pos`(`K`, `VECTOR1`, `VECTOR2`).

Arc input(s)	VARIABLES1 VARIABLES2
Arc generator	$PRODUCT(CLIQUE, LOOP, =) \mapsto$ collection(variables1, variables2)
Arc arity	2
Arc constraint(s)	variables1.var = variables2.var
Graph property(ies)	• MAX_NSCC ≤ 1 • NARC_NO_LOOP = NSAME

Graph model The arc generator $PRODUCT(CLIQUE, LOOP, =)$ is used in order to generate all the arcs of the initial graph:

- The arc generator $CLIQUE$ creates all links between the items of the first collection VARIABLES1,
- The arc generator $LOOP$ creates a loop for each item of the second collection VARIABLES2,
- Finally the arc generator $PRODUCT(=)$ creates an arc between items located at the same position in the collections VARIABLES1 and VARIABLES2.

Part (A) of Figure 5.50 gives the initial graph associated with the **Example** slot. Variables of collection VARIABLES1 are coloured, while variables of collection VARIABLES2 are kept in white. Part (B) represents the final graph associated with the **Example** slot. In this graph each vertex constitutes a strongly connected component and the number of arcs that do not correspond to a loop is equal to 2 (i.e., NSAME).



(A) (B)

Figure 5.50: (A) Initial and (B) final graph of the alldifferent_same_value$(2, \langle U_1, U_2, U_3, U_4\rangle, \langle V_1, V_2, V_3, V_4\rangle)$ constraint with $U_1 = 7$, $U_2 = 3$, $U_3 = 1$, $U_4 = 5$ and $V_1 = 1$, $V_2 = 3$, $V_3 = 1$, $V_4 = 7$ (in Part (B) arcs in red correspond to the arcs counted by the argument NSAME)

Automaton

Figure 5.51 depicts the automaton associated with the `alldifferent_same_value` constraint. Let $\texttt{VAR1}_i$ and $\texttt{VAR2}_i$ respectively denote the i^{th} variables of the `VARIABLES1` and `VARIABLES2` collections. To each pair of variables $(\texttt{VAR1}_i, \texttt{VAR2}_i)$ corresponds a signature variable $\texttt{S}_i$. The following signature constraint links $\texttt{VAR1}_i$, $\texttt{VAR2}_i$ and $\texttt{S}_i$: $\texttt{VAR1}_i = \texttt{VAR2}_i \Leftrightarrow \texttt{S}_i$.

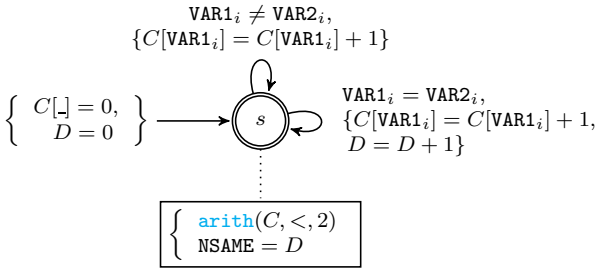


Figure 5.51: Automaton of the `alldifferent_same_value` constraint