

## 5.10 alldifferent\_interval

	DESCRIPTION	LINKS	GRAPH	AUTOMATON
<b>Origin</b>	Derived from <a href="#">alldifferent</a> .			
<b>Constraint</b>	<code>alldifferent_interval(VARIABLES, SIZE_INTERVAL)</code>			
<b>Synonyms</b>	<code>alldiff_interval</code> , <code>alldistinct_interval</code> .			
<b>Arguments</b>	VARIABLES : <code>collection</code> (var-dvar) SIZE_INTERVAL : <code>int</code>			
<b>Restrictions</b>	<code>required(VARIABLES, var)</code> SIZE_INTERVAL > 0			
<b>Purpose</b>	Enforce all variables of the collection VARIABLES to belong to distinct intervals. The intervals are defined by $[SIZE\_INTERVAL \cdot k, SIZE\_INTERVAL \cdot k + SIZE\_INTERVAL - 1]$ where $k$ is an integer.			
<b>Example</b>	<code>((2, 4, 10), 3)</code> <p>In the example, the second argument <code>SIZE_INTERVAL = 3</code> defines the following family of intervals <math>[3 \cdot k, 3 \cdot k + 2]</math>, where <math>k</math> is an integer. Since the three variables of the collection VARIABLES take values that are respectively located within the three following distinct intervals <math>[0, 2]</math>, <math>[3, 5]</math> and <math>[9, 11]</math>, the <code>alldifferent_interval</code> constraint holds.</p>			
<b>Typical</b>	<code> VARIABLES  &gt; 2</code> <code>SIZE_INTERVAL &gt; 1</code>			
<b>Symmetries</b>	<ul style="list-style-type: none"> <li>Items of VARIABLES are <a href="#">permutable</a>.</li> <li>A value of VARIABLES.var that belongs to the <math>k</math>-th interval, of size SIZE_INTERVAL, can be renamed to any unused value of the same interval.</li> <li>Two distinct values of VARIABLES.var that belong to two distinct intervals, of size SIZE_INTERVAL, can be <a href="#">swapped</a>.</li> </ul>			
<b>See also</b>	<b>implied by:</b> <a href="#">all_min_dist</a> . <b>specialisation:</b> <code>alldifferent</code> (variable/constant replaced by variable).			
<b>Keywords</b>	<b>characteristic of a constraint:</b> <code>all different</code> , <code>automaton</code> , <code>automaton with array of counters</code> . <b>constraint type:</b> <code>value constraint</code> . <b>filtering:</b> <code>arc-consistency</code> . <b>final graph structure:</b> <code>one_succ</code> . <b>modelling:</b> <code>interval</code> .			

<b>Arc input(s)</b>	VARIABLES
<b>Arc generator</b>	<code>CLIQUE</code> $\mapsto$ <code>collection</code> (variables1, variables2)
<b>Arc arity</b>	2
<b>Arc constraint(s)</b>	$\text{variables1.var}/\text{SIZE\_INTERVAL} = \text{variables2.var}/\text{SIZE\_INTERVAL}$
<b>Graph property(ies)</b>	<code>MAX_NSICC</code> $\leq 1$
<b>Graph class</b>	<code>ONE_SUCC</code>

**Graph model**

Similar to the `alldifferent` constraint, but we replace the binary *equality* constraint of the `alldifferent` constraint by the fact that two variables are respectively assigned to two values that belong to the same interval. We generate a *clique* with a *belong to the same interval* constraint between each pair of vertices (including a vertex and itself) and state that the size of the largest strongly connected component should not exceed 1.

Parts (A) and (B) of Figure 5.12 respectively show the initial and final graph associated with the **Example** slot. Since we use the `MAX_NSICC` graph property we show one of the largest strongly connected component of the final graph.

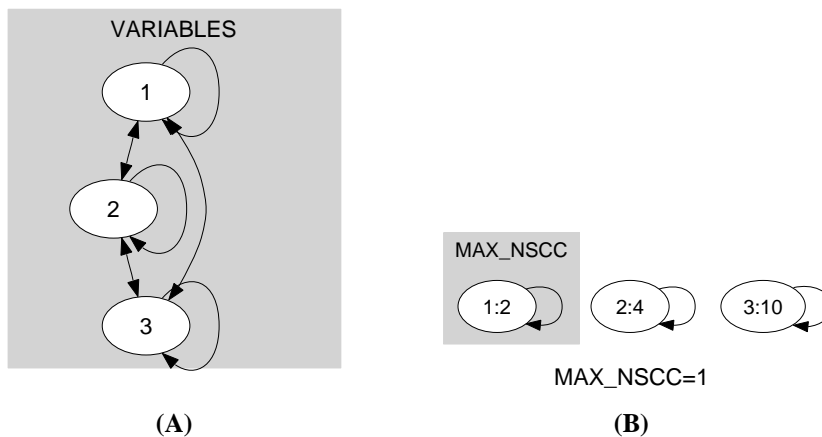


Figure 5.12: Initial and final graph of the `alldifferent_interval` constraint

**Automaton**

Figure 5.13 depicts the automaton associated with the `alldifferent_interval` constraint. To each item of the collection `VARIABLES` corresponds a signature variable  $S_i$  that is equal to 1. For each interval  $[SIZE\_INTERVAL \cdot k, SIZE\_INTERVAL \cdot k + SIZE\_INTERVAL - 1]$  of values the automaton counts the number of occurrences of its values and finally imposes that the values of an interval are taken at most once.

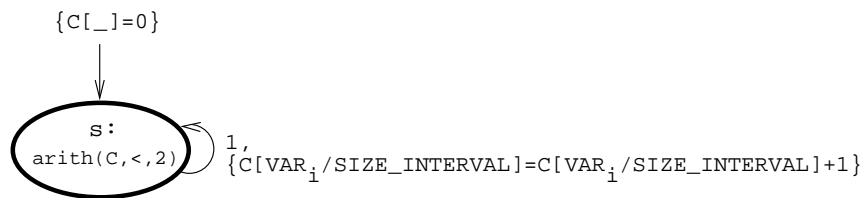


Figure 5.13: Automaton of the `alldifferent_interval` constraint

20030820

421