

### 5.310 `orth_link_ori_siz_end`

	DESCRIPTION	LINKS	GRAPH
<b>Origin</b>	Used by several constraints between orthotopes		
<b>Constraint</b>	<code>orth_link_ori_siz_end(ORTHOTOPE)</code>		
<b>Argument</b>	<code>ORTHOTOPE</code> : <code>collection(ori-dvar, siz-dvar, end-dvar)</code>		
<b>Restrictions</b>	$ ORTHOTOPE  > 0$ <code>require_at_least(2, ORTHOTOPE, [ori, siz, end])</code> $ORTHOTOPE.siz \geq 0$ $ORTHOTOPE.ori \leq ORTHOTOPE.end$		
<b>Purpose</b>	Enforce for each item of the ORTHOTOPE collection the constraint $ori + siz = end$ .		
<b>Example</b>	$(\langle ori - 2 \quad siz - 2 \quad end - 4, ori - 1 \quad siz - 3 \quad end - 4 \rangle)$ <p>The <code>orth_link_ori_siz_end</code> constraint holds since the two items <math>\langle ori - 2 \quad siz - 2 \quad end - 4 \rangle</math> and <math>\langle ori - 1 \quad siz - 3 \quad end - 4 \rangle</math> respectively verify the conditions <math>2 + 2 = 4</math> and <math>1 + 3 = 4</math>.</p>		
<b>Typical</b>	$ ORTHOTOPE  > 1$ $ORTHOTOPE.siz > 0$		
<b>Symmetries</b>	<ul style="list-style-type: none"> <li>Items of ORTHOTOPE are <a href="#">permutable</a>.</li> <li>One and the same constant can be <a href="#">added</a> to the <code>ori</code> and <code>end</code> attributes of all items of ORTHOTOPE.</li> <li>One and the same constant can be <a href="#">added</a> to the <code>siz</code> and <code>end</code> attributes of all items of ORTHOTOPE.</li> </ul>		
<b>Arg. properties</b>	<ul style="list-style-type: none"> <li><b>Functional dependency:</b> <code>ORTHOTOPE.ori</code> determined by <code>ORTHOTOPE.siz</code> and <code>ORTHOTOPE.end</code>.</li> <li><b>Functional dependency:</b> <code>ORTHOTOPE.siz</code> determined by <code>ORTHOTOPE.ori</code> and <code>ORTHOTOPE.end</code>.</li> <li><b>Functional dependency:</b> <code>ORTHOTOPE.end</code> determined by <code>ORTHOTOPE.ori</code> and <code>ORTHOTOPE.siz</code>.</li> <li><b>Contractible</b> wrt. ORTHOTOPE.</li> </ul>		
<b>Usage</b>	Used in the <b>Arc constraint(s)</b> slot for defining some constraints like <a href="#">diffn</a> , <a href="#">place_in_pyramid</a> or <a href="#">orths_are_connected</a> .		
<b>Used in</b>	<a href="#">diffn</a> , <a href="#">orth_on_the_ground</a> , <a href="#">orth_on_top_of_orth</a> , <a href="#">orths_are_connected</a> , <a href="#">two_orth_are_in_contact</a> , <a href="#">two_orth_column</a> , <a href="#">two_orth_do_not_overlap</a> , <a href="#">two_orth_include</a> .		

**Keywords**

**constraint arguments:** pure functional dependency.

**constraint type:** decomposition.

**geometry:** orthotope.

**modelling:** functional dependency.

<b>Arc input(s)</b>	ORTHOTOPE
<b>Arc generator</b>	$\text{SELF} \mapsto \text{collection}(\text{orthotope})$
<b>Arc arity</b>	1
<b>Arc constraint(s)</b>	$\text{orthotope.ori} + \text{orthotope.siz} = \text{orthotope.end}$
<b>Graph property(ies)</b>	$\text{NARC} =  \text{ORTHOTOPE} $

**Graph model**

Parts (A) and (B) of Figure 5.631 respectively show the initial and final graph associated with the **Example** slot. Since we use the **NARC** graph property, the loops of the final graph are stressed in bold.

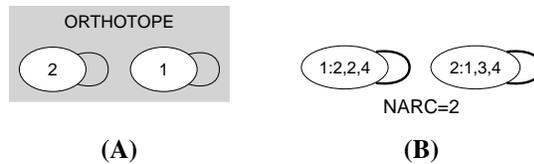


Figure 5.631: Initial and final graph of the `orth_link_ori_siz_end` constraint

**Signature**

Since we use the *SELF* arc generator on the ORTHOTOPE collection the number of arcs of the initial graph is equal to  $|\text{ORTHOTOPE}|$ . Therefore the maximum number of arcs of the final graph is also equal to  $|\text{ORTHOTOPE}|$ . For this reason we can rewrite the graph property  $\text{NARC} = |\text{ORTHOTOPE}|$  to  $\text{NARC} \geq |\text{ORTHOTOPE}|$  and simplify  $\overline{\text{NARC}}$  to  $\overline{\text{NARC}}$ .

20030820

1931