## 5.49 balance_tree

DESCRIPTION LINKS GRAPH

**Origin** derived from `balance` and `tree`

**Constraint** `balance_tree(BALANCE, NODES)`

**Arguments**

BALANCE : dvar
NODES : collection(index−int, succ−dvar)

**Restrictions**

BALANCE $\geq 0$
BALANCE $\leq$ max(0, |NODES| − 2)
required(NODES, [index, succ])
NODES.index $\geq 1$
NODES.index $\leq$ |NODES|
distinct(NODES, index)
NODES.succ $\geq 1$
NODES.succ $\leq$ |NODES|

**Purpose** Consider a digraph $G$ described by the NODES collection. Partition $G$ into a set of vertex disjoint trees in such a way that each vertex of $G$ belongs to a single tree. BALANCE is equal to the difference between the number of vertices of the largest tree and the number of vertices of the smallest tree.

**Example**

$$\left(\begin{array}{l}4,\left\langle\begin{array}{ll}\texttt{index}-1 & \texttt{succ}-1,\\ \texttt{index}-2 & \texttt{succ}-5,\\ \texttt{index}-3 & \texttt{succ}-5,\\ \texttt{index}-4 & \texttt{succ}-7,\\ \texttt{index}-5 & \texttt{succ}-1,\\ \texttt{index}-6 & \texttt{succ}-1,\\ \texttt{index}-7 & \texttt{succ}-7,\\ \texttt{index}-8 & \texttt{succ}-5\end{array}\right\rangle\end{array}\right)$$

$$\left(\begin{array}{l}2,\left\langle\begin{array}{ll}\texttt{index}-1 & \texttt{succ}-1,\\ \texttt{index}-2 & \texttt{succ}-1,\\ \texttt{index}-3 & \texttt{succ}-1,\\ \texttt{index}-4 & \texttt{succ}-2,\\ \texttt{index}-5 & \texttt{succ}-6,\\ \texttt{index}-6 & \texttt{succ}-6\end{array}\right\rangle\end{array}\right)$$

In the first example we have two trees involving respectively the set of vertices $\{1, 2, 3, 5, 6, 8\}$ and the set $\{4, 7\}$. They are depicted by Figure 5.110. Since BALANCE $= 6 - 2 = 4$ is the difference between the number of vertices of the largest tree (i.e., 6) and the number of vertices of the smallest tree (i.e., 2) the corresponding `balance_tree` constraint holds.

**All solutions** Figure 5.111 gives all solutions to the following non ground instance of the `balance_tree` constraint: BALANCE $= 0$, $S_1 \in [1, 2]$, $S_2 \in [1, 2]$, $S_3 \in [4, 5]$, $S_4 \in [2, 4]$, $S_5 \in [4, 5]$, $S_6 \in [5, 6]$, `balance_tree`(BALANCE, $\langle 1\ S_1, 2\ S_2, 3\ S_3, 4\ S_4, 5\ S_5, 6\ S_6\rangle$).

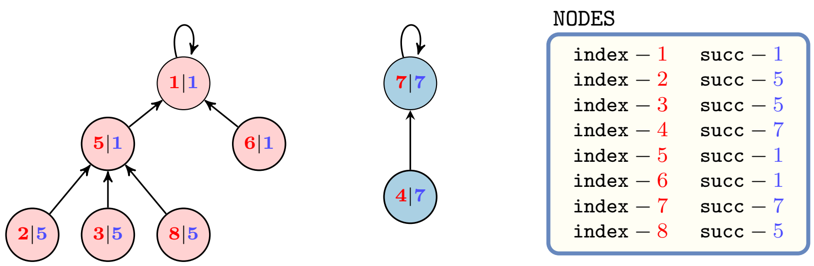



Figure 5.110: The two trees associated with the first example of the **Example** slot, respectively containing 6 and 2 vertices, therefore BALANCE $= 6 - 2 = 4$; each vertex contains the information index|succ where succ is the index of its father in the tree (by convention the father of the root is the root itself).

① $(\mathbf{0}, \langle 1_1, 1_2, 4_3, 2_4, 4_5, 5_6\rangle)$
② $(\mathbf{0}, \langle 1_1, 1_2, 4_3, 4_4, 5_5, 5_6\rangle)$
③ $(\mathbf{0}, \langle 1_1, 1_2, 5_3, 2_4, 4_5, 5_6\rangle)$
④ $(\mathbf{0}, \langle 1_1, 1_2, 5_3, 2_4, 5_5, 5_6\rangle)$
⑤ $(\mathbf{0}, \langle 2_1, 2_2, 4_3, 2_4, 4_5, 5_6\rangle)$
⑥ $(\mathbf{0}, \langle 2_1, 2_2, 4_3, 4_4, 5_5, 5_6\rangle)$
⑦ $(\mathbf{0}, \langle 2_1, 2_2, 5_3, 2_4, 4_5, 5_6\rangle)$
⑧ $(\mathbf{0}, \langle 2_1, 2_2, 5_3, 2_4, 5_5, 5_6\rangle)$

Figure 5.111: All solutions corresponding to the non ground example of the balance_tree constraint of the **All solutions** slot; the index attribute is displayed as indices of the succ attribute and all vertices of a same tree are coloured by the same colour.

**Typical** |NODES| $> 2$

**Symmetry** Items of NODES are permutable.

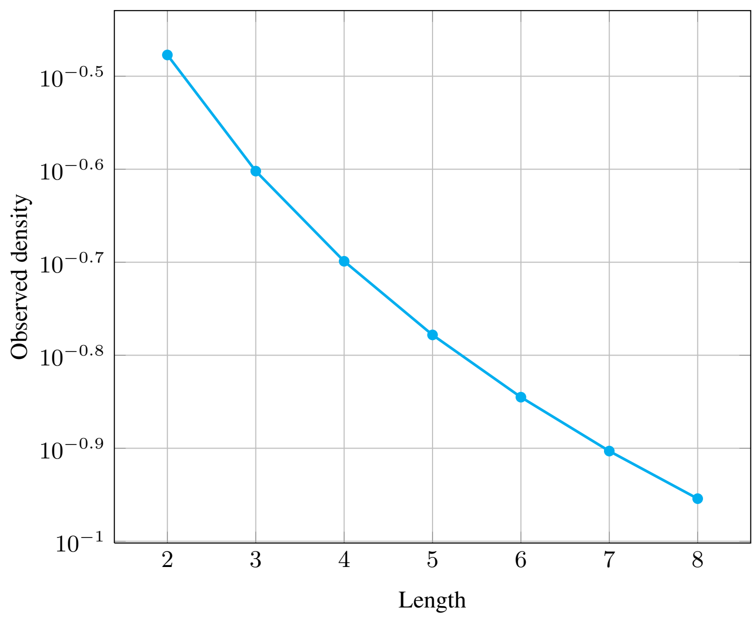
**Arg. properties** Functional dependency: BALANCE determined by NODES.

**Counting**

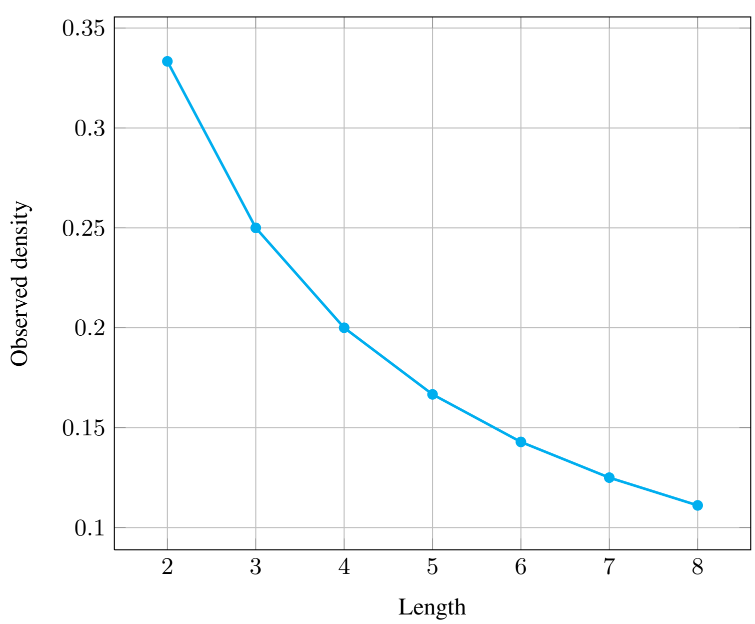
| Length ($n$) | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|
| Solutions | 3 | 16 | 125 | 1296 | 16807 | 262144 | 4782969 |

Number of solutions for balance_tree: domains $0..n$

Solution density for `balance_tree`

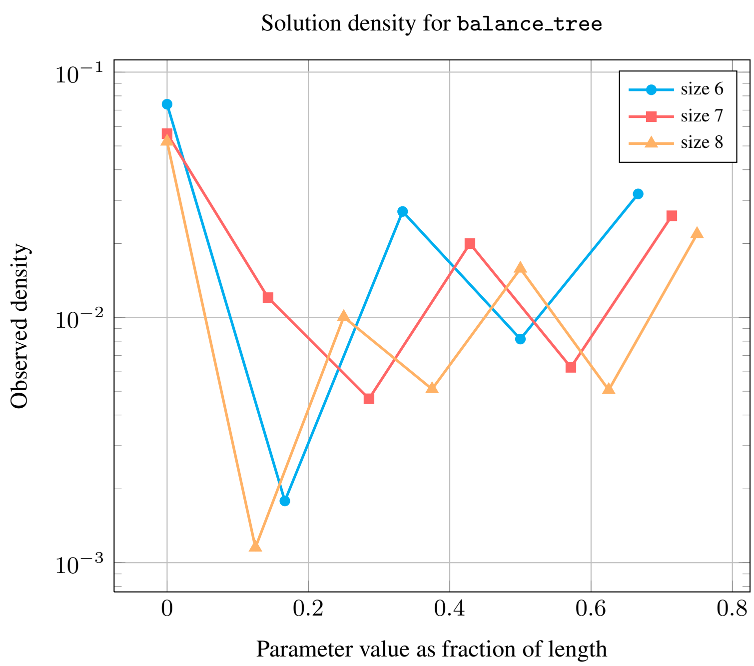


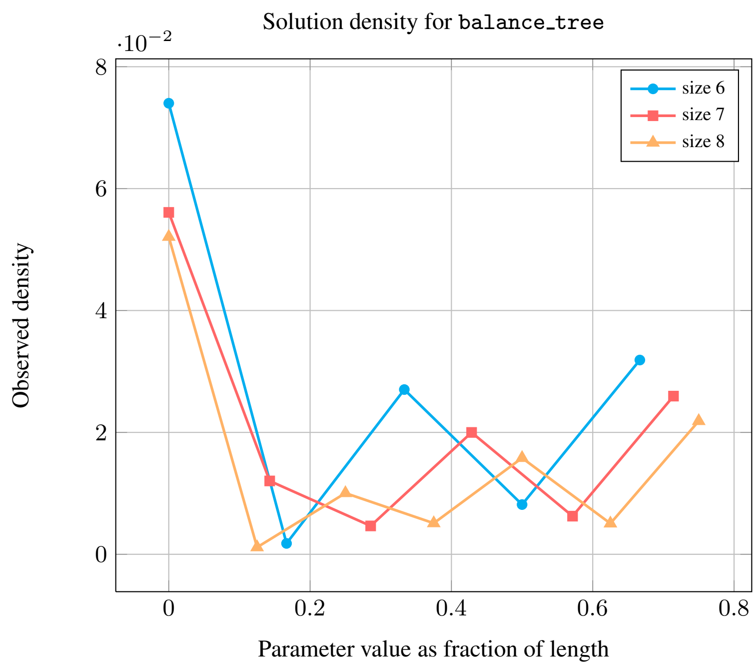

Solution density for `balance_tree`



| Length ($n$) | | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| Total | | 3 | 16 | 125 | 1296 | 16807 | 262144 | 4782969 |
| Parameter value | 0 | 3 | 10 | 77 | 626 | 8707 | 117650 | 2242193 |
| | 1 | - | 6 | 12 | 260 | 210 | 25242 | 49616 |
| | 2 | - | - | 36 | 90 | 3180 | 9765 | 432264 |
| | 3 | - | - | - | 320 | 960 | 41930 | 219520 |
| | 4 | - | - | - | - | 3750 | 13125 | 680456 |
| | 5 | - | - | - | - | - | 54432 | 217728 |
| | 6 | - | - | - | - | - | - | 941192 |

Solution count for `balance_tree`: domains $0..n$






**See also**

**implied by:** `balance_path`.

**related:** `balance` *(equivalence classes correspond to vertices in same tree rather than variables assigned to the same value)*, `tree` *(do not care how many trees but how balanced the trees are)*.

**Keywords**

**constraint type:** graph constraint, graph partitioning constraint.

**filtering:** DFS-bottleneck.

**final graph structure:** connected component, tree, one_succ.

**modelling:** functional dependency.

**Cond. implications**

balance_tree(BALANCE, NODES)
  with BALANCE $> 0$
  and BALANCE $\leq$ |NODES|
**implies** ordered_atleast_nvector(NVEC : BALANCE, VECTORS : NODES).

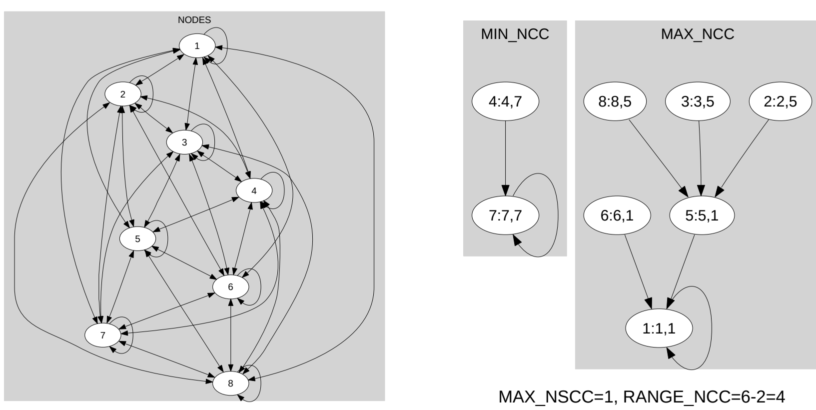
| | |
|---|---|
| **Arc input(s)** | NODES |
| **Arc generator** | *CLIQUE* $\mapsto$ collection(nodes1, nodes2) |
| **Arc arity** | 2 |
| **Arc constraint(s)** | nodes1.succ = nodes2.index |
| **Graph property(ies)** | • **MAX_NSCC** $\leq 1$<br>• **RANGE_NCC** = BALANCE |

**Graph model** In order to express the binary constraint that links two vertices one has to make explicit the identifier of the vertices. This is why the balance_tree constraint considers objects that have two attributes:

- One fixed attribute index that is the identifier of the vertex,
- One variable attribute succ that is the successor of the vertex.

We use the graph property **MAX_NSCC** $\leq 1$ in order to specify the fact that the size of the largest strongly connected component should not exceed one. In fact each root of a tree is a strongly connected component with a single vertex.

Parts (A) and (B) of Figure 5.112 respectively show the initial and final graph associated with the first example of the **Example** slot. Since we use the **RANGE_NCC** graph property, we show the connected components of the final graph. The constraint holds since all the vertices belong to a tree and since BALANCE = **RANGE_NCC** $6 - 2 = 4$.




**(A)** **(B)**

Figure 5.112: Initial and final graph of the balance_tree constraint