

5.279 no_valley

	DESCRIPTION	LINKS	AUTOMATON
Origin	Derived from valley .		
Constraint	<code>no_valley(VARIABLES)</code>		
Argument	<code>VARIABLES : collection(var-dvar)</code>		
Restrictions	$ VARIABLES > 0$ <code>required(VARIABLES, var)</code>		
Purpose	<p>A variable V_k ($1 < k < m$) of the sequence of variables $VARIABLES = V_1, \dots, V_m$ is a <i>valley</i> if and only if there exists an i ($1 < i \leq k$) such that $V_{i-1} > V_i$ and $V_i = V_{i+1} = \dots = V_k$ and $V_k < V_{k+1}$. The total number of valleys of the sequence of variables $VARIABLES$ is equal to 0.</p>		
Example	<div style="border: 1px solid black; padding: 5px; display: inline-block;"> $((1, 1, 4, 8, 8, 2))$ </div>		

The `no_valley` constraint holds since the sequence 1 1 4 8 8 2 does not contain any valley.

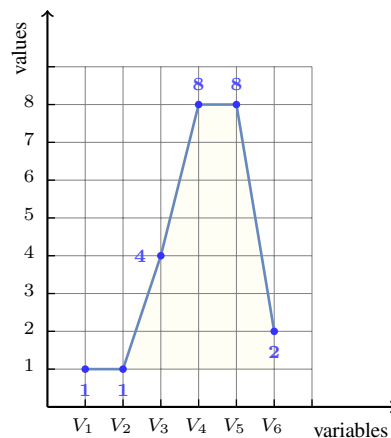


Figure 5.583: Illustration of the **Example** slot: a sequence of five variables $V_1, V_2, V_3, V_4, V_5, V_6$ respectively fixed to values 1, 1, 4, 8, 8, 2 without any valley

Typical	$ VARIABLES > 3$ <code>range(VARIABLES.var) > 1</code>
----------------	---

Symmetries

- Items of VARIABLES can be [reversed](#).
- One and the same constant can be [added](#) to the `var` attribute of all items of VARIABLES.

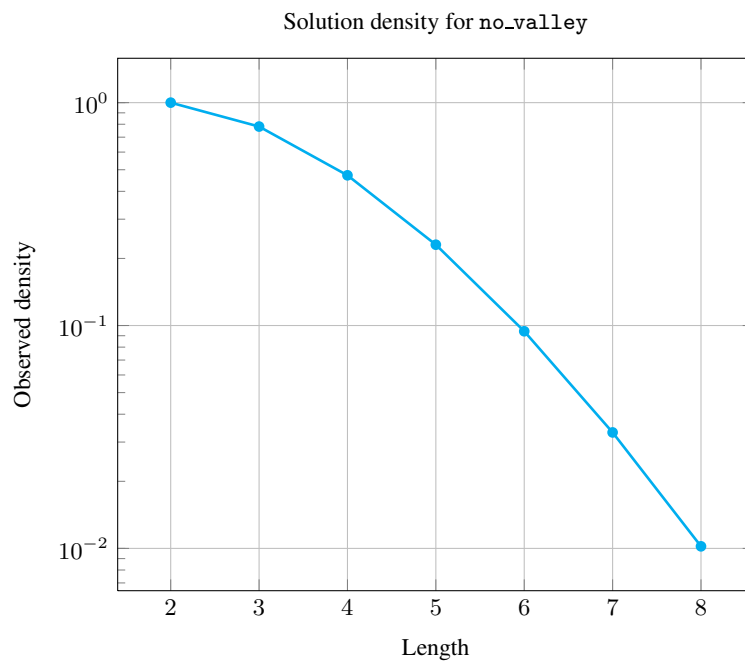
Arg. properties

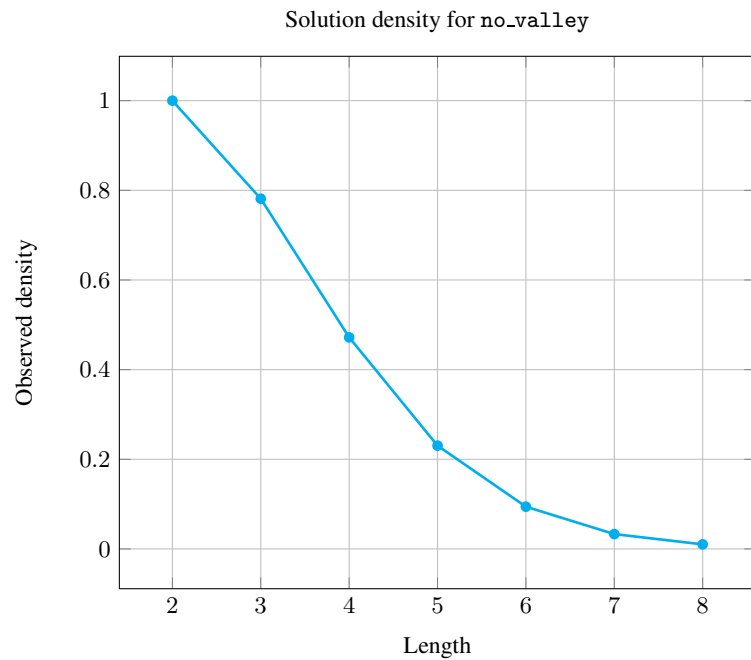
[Contractible](#) wrt. VARIABLES.

Counting

Length (n)	2	3	4	5	6	7	8
Solutions	9	50	295	1792	11088	69498	439791

Number of solutions for `no_valley`: domains $0..n$



**See also**

comparison swapped: [no_peak](#).

generalisation: [valley](#) (introduce a variable counting the number of valleys).

implied by: [decreasing](#), [global_contiguity](#), [increasing](#).

implies: [all_equal_valley_min](#).

related: [peak](#).

Keywords

characteristic of a constraint: [automaton](#), [automaton without counters](#), [automaton with same input symbol](#), [reified automaton constraint](#).

combinatorial object: [sequence](#).

constraint network structure: [sliding cyclic\(1\)](#) [constraint network\(1\)](#).

Automaton

Figure 5.584 depicts the automaton associated with the no_valley constraint. To each pair of consecutive variables (VAR_i, VAR_{i+1}) of the collection VARIABLES corresponds a signature variable S_i . The following signature constraint links VAR_i, VAR_{i+1} and S_i : $(VAR_i < VAR_{i+1} \Leftrightarrow S_i = 0) \wedge (VAR_i = VAR_{i+1} \Leftrightarrow S_i = 1) \wedge (VAR_i > VAR_{i+1} \Leftrightarrow S_i = 2)$.

STATES SEMANTICS

s : stationary/increasing mode ($\{< | =\}^*$)
 t : decreasing mode ($\{> | =\}^*$)

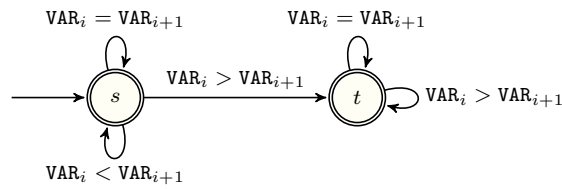


Figure 5.584: Automaton of the no_valley constraint

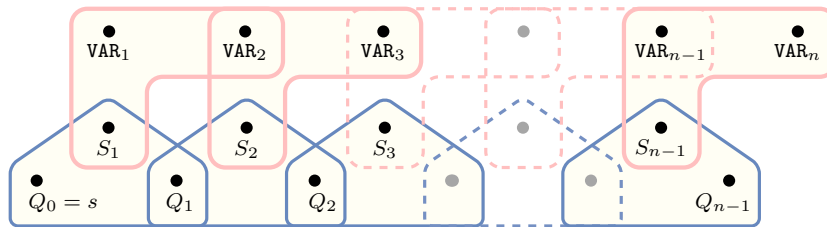


Figure 5.585: Hypergraph of the reformulation corresponding to the automaton of the no_valley constraint (since all states of the automaton are accepting there is no restriction on the last variable Q_{n-1})