## 5.112   decreasing_valley

**Origin**          Derived from valley and decreasing.

**Constraint**      decreasing_valley(VARIABLES)

**Argument**         VARIABLES : collection(var−dvar)

**Restrictions**     $|\text{VARIABLES}| > 0$
                    required(VARIABLES, var)

**Purpose**         A variable $V_k$ $(1 < k < m)$ of the sequence of variables $\text{VARIABLES} = V_1, \ldots, V_m$ is a *valley* if and only if there exists an $i$ $(1 < i \leq k)$ such that $V_{i-1} > V_i$ and $V_i = V_{i+1} = \cdots = V_k$ and $V_k < V_{k+1}$.
                    When considering all the valleys of the sequence VARIABLES from left to right enforce all valleys to be decreasing, i.e. the altitude of each valley is less than or equal to the altitude of its preceding valley when it exists.

**Example**          $(\langle 1, 7, 6, 8, 3, 7, 3, 3, 5, 4 \rangle)$

The decreasing_valley constraint holds since the sequence 1 7 **6** 8 **3** 7 3 **3** 5 2 contains three valleys, in bold, that are decreasing.



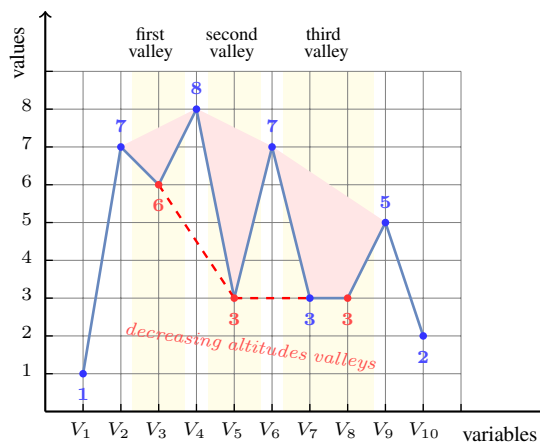Figure 5.248: Illustration of the **Example** slot: a sequence of ten variables $V_1$, $V_2$, $V_3$, $V_4$, $V_5$, $V_6$, $V_7$, $V_8$, $V_9$, $V_{10}$ respectively fixed to values 1, 7, 6, 8, 3, 7, 3, 3, 5, 2 and its corresponding three valleys, in red, respectively located at altitudes 6, 3 and 3

**Typical**

$|\mathtt{VARIABLES}| \geq 7$
$\mathtt{range}(\mathtt{VARIABLES.var}) > 1$
$\mathtt{valley}(\mathtt{VARIABLES.var}) \geq 3$

**Symmetry**

One and the same constant can be added to the `var` attribute of all items of `VARIABLES`.
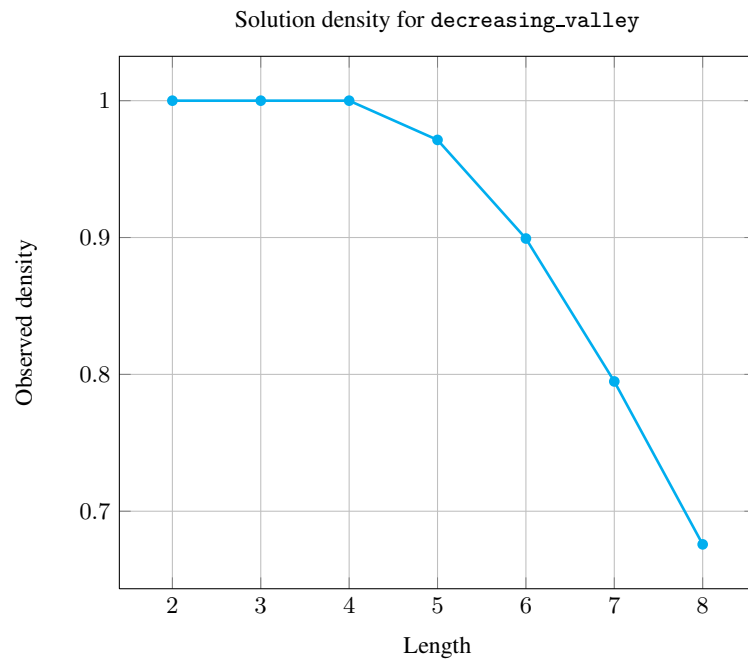
**Arg. properties**

- Prefix-contractible wrt. `VARIABLES`.
- Suffix-contractible wrt. `VARIABLES`.

**Counting**

| Length $(n)$ | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|
| Solutions | 9 | 64 | 625 | 7553 | 105798 | 1666878 | 29090469 |

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



Solution density for `decreasing_valley`

Solution density for `decreasing_valley`
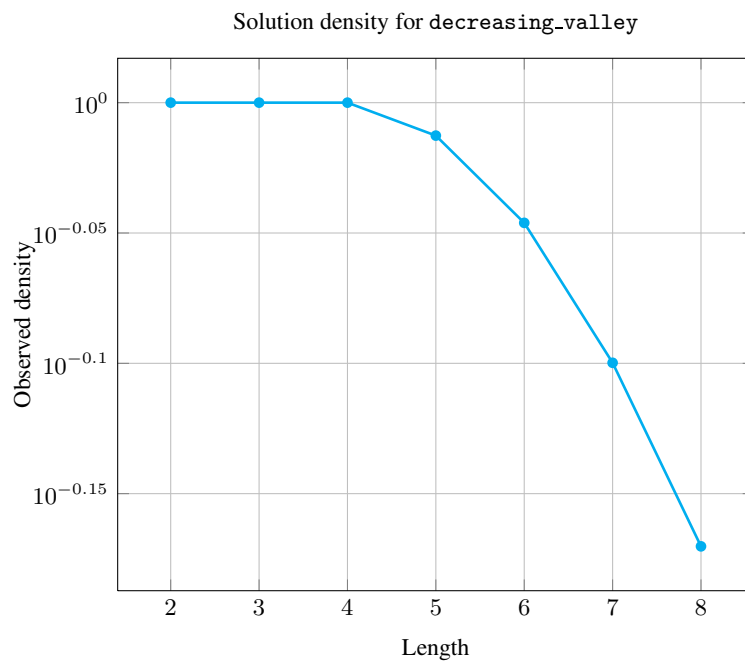
| | |
|---|---|
| **See also** | **implied by:** `all_equal_valley`. |
| | **related:** `increasing_valley`, `valley`. |
| **Keywords** | **characteristic of a constraint:** automaton, automaton with counters, automaton with same input symbol. |
| | **combinatorial object:** sequence. |
| | **constraint network structure:** sliding cyclic(1) constraint network(2). |
| **Cond. implications** | `decreasing_valley(VARIABLES)` with `valley(VARIABLES.var)` $> 0$ **implies** `not_all_equal(VARIABLES)`. |

**Automaton**

Figure 5.249 depicts the automaton associated with the `decreasing_valley` constraint. To each pair of consecutive variables $(\mathtt{VAR}_i, \mathtt{VAR}_{i+1})$ of the collection `VARIABLES` corresponds a signature variable $S_i$. The following signature constraint links $\mathtt{VAR}_i$, $\mathtt{VAR}_{i+1}$ and $S_i$: $(\mathtt{VAR}_i < \mathtt{VAR}_{i+1} \Leftrightarrow S_i = 0) \wedge (\mathtt{VAR}_i = \mathtt{VAR}_{i+1} \Leftrightarrow S_i = 1) \wedge (\mathtt{VAR}_i > \mathtt{VAR}_{i+1} \Leftrightarrow S_i = 2)$.

STATES SEMANTICS

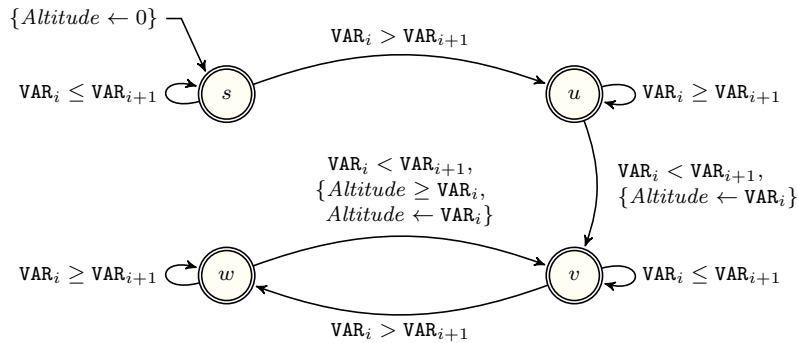| | | |
|---|---|---|
| $s$ | : initial stationary or increasing mode | $(\{= \mid >\}^*)$ |
| $u$ | : decreasing (before first potential valley) mode | $(< \{< \mid =\}^*)$ |
| $v$ | : increasing (after a valley) mode | $(> \{> \mid =\}^*)$ |
| $w$ | : decreasing (after a valley) mode | $(< \{< \mid =\}^*)$ |



Figure 5.249: Automaton for the `decreasing_valley` constraint (note the conditional transition from state $w$ to state $v$ testing that the counter *Altitude* is greater than or equal to $\mathtt{VAR}_i$ for enforcing that all valleys from left to right are in decreasing altitude)
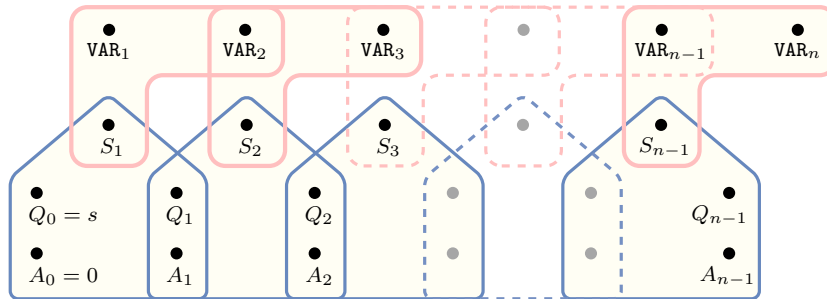


Figure 5.250: Hypergraph of the reformulation corresponding to the automaton of the `decreasing_valley` constraint where $A_i$ stands for the value of the counter *Altitude* (since all states of the automaton are accepting there is no restriction on the last variable $Q_{n-1}$)