

5.111 decreasing_peak

	DESCRIPTION	LINKS	AUTOMATON
Origin	Derived from peak and decreasing .		
Constraint	decreasing_peak(VARIABLES)		
Argument	VARIABLES : collection (var-dvar)		
Restrictions	$ VARIABLES > 0$ required (VARIABLES, var)		
Purpose	<p>A variable V_k ($1 < k < m$) of the sequence of variables $VARIABLES = V_1, \dots, V_m$ is a <i>peak</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 $V_k > V_{k+1}$.</p> <p>When considering all the peaks of the sequence $VARIABLES$ from left to right enforce all peaks to be decreasing, i.e. the altitude of each peak is less than or equal to the altitude of its preceding peak when it exists.</p>		
Example	<div>$((1, 7, 7, 4, 3, 7, 2, 2, 5, 4))$</div>		

The decreasing_peak constraint holds since the sequence 1 7 7 4 3 7 2 2 5 4 contains three peaks, in bold, that are decreasing.

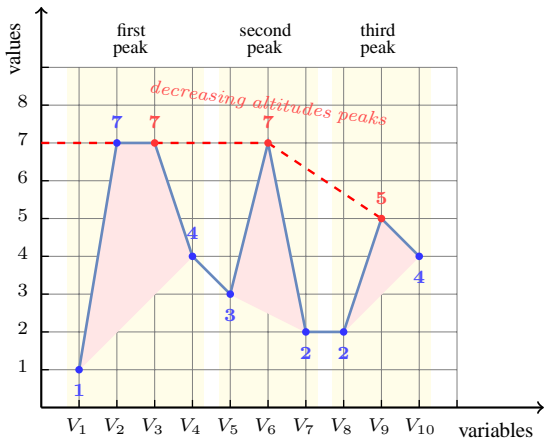


Figure 5.262: Illustration of the **Example** slot: a sequence of ten variables $V_1, V_2, V_3, V_4, V_5, V_6, V_7, V_8, V_9, V_{10}$ respectively fixed to values 1, 7, 7, 4, 3, 7, 2, 2, 5, 4 and its corresponding three peaks, in red, respectively located at altitudes 7, 7 and 5

Typical

```
|VARIABLES| ≥ 7
range(VARIABLES.var) > 1
peak(VARIABLES.var) ≥ 3
```

Symmetry

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

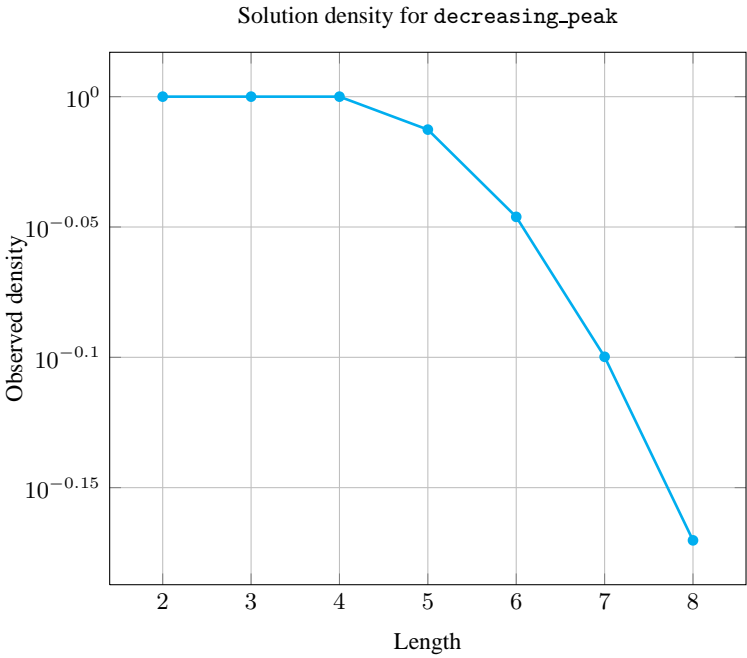
Arg. properties

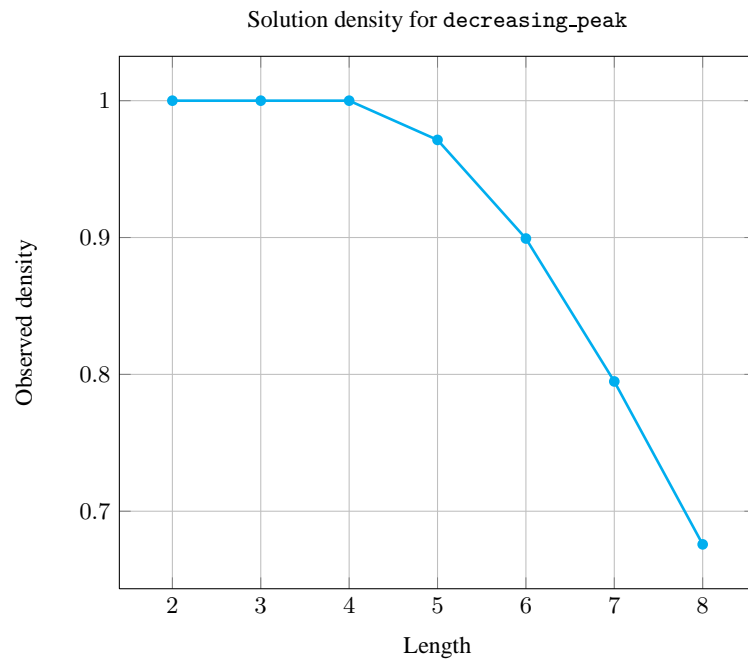
- Prefix-contractible wrt. VARIABLES.
- Suffix-contractible wrt. VARIABLES.

Counting

Length (<i>n</i>)	2	3	4	5	6	7	8
Solutions	9	64	625	7553	105798	1666878	29090469

Number of solutions for decreasing_peak: domains 0..*n*



**See also**

implied by: [all_equal_peak](#).

related: [increasing_peak](#), [peak](#).

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_peak(VARIABLES)`
 with `peak(VARIABLES.var) > 0`
implies `not_all_equal(VARIABLES)`.

Automaton

Figure 5.263 depicts the automaton associated with the `decreasing_peak` constraint. To each pair of consecutive variables $(\text{VAR}_i, \text{VAR}_{i+1})$ of the collection `VARIABLES` corresponds a signature variable S_i . The following signature constraint links VAR_i , VAR_{i+1} and S_i : $(\text{VAR}_i < \text{VAR}_{i+1} \Leftrightarrow S_i = 0) \wedge (\text{VAR}_i = \text{VAR}_{i+1} \Leftrightarrow S_i = 1) \wedge (\text{VAR}_i > \text{VAR}_{i+1} \Leftrightarrow S_i = 2)$.

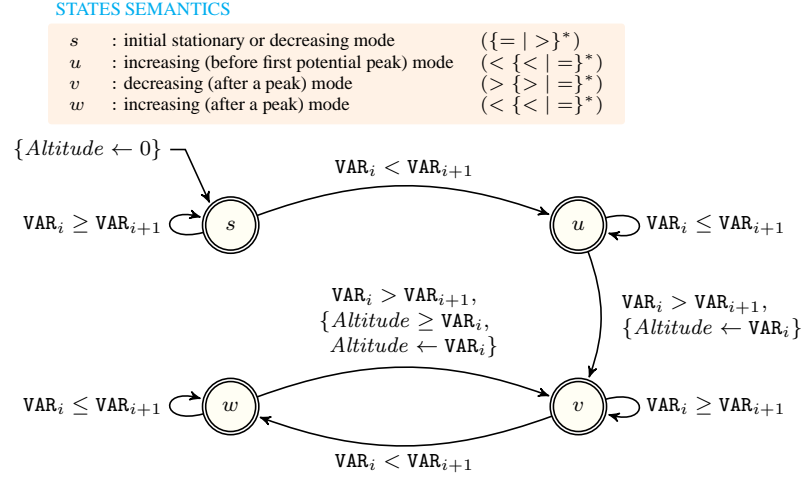


Figure 5.263: Automaton for the `decreasing_peak` constraint (note the conditional transition from state w to state v testing that the counter *Altitude* is greater than or equal to VAR_i for enforcing that all peaks from left to right are in decreasing altitude)

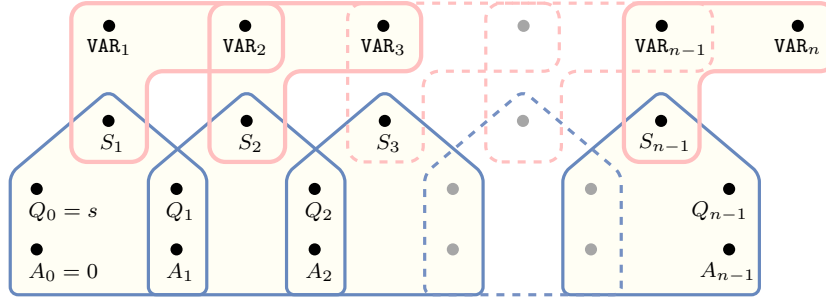


Figure 5.264: Hypergraph of the reformulation corresponding to the automaton of the `decreasing_peak` constraint where A_i stands for the value of the counter *Altitude* (since all states of the automaton are accepting there is no restriction on the last variable Q_{n-1})