5.319 peak

DESCRIPTION LINKS AUTOMATON

Origin Derived from inflexion.

Constraint peak(N, VARIABLES)

Arguments

N	:	dvar
VARIABLES	:	collection(var−dvar)

Restrictions

$\mathtt{N} \geq 0$
$2 * \mathtt{N} \leq \max(|\mathtt{VARIABLES}| - 1, 0)$
required(VARIABLES, var)

Purpose

A variable V_k $(1 < k < m)$ of the sequence of variables VARIABLES $= V_1, \ldots, V_m$ is a *peak* if and only if there exists an i (with $1 < i \leq k$) such that $V_{i-1} < V_i$ and $V_i = V_{i+1} = \cdots = V_k$ and $V_k > V_{k+1}$. N is the total number of peaks of the sequence of variables VARIABLES.

Example

$(2, \langle 1, 1, 4, 8, 6, 2, 7, 1\rangle)$
$(0, \langle 1, 1, 4, 4, 4, 6, 7, 7\rangle)$
$(4, \langle 1, 5, 4, 9, 4, 6, 2, 7, 6\rangle)$

The first peak constraint holds since the sequence 1 1 4 8 6 2 7 1 contains two peaks that respectively correspond to the variables that are assigned to values 8 and 7.

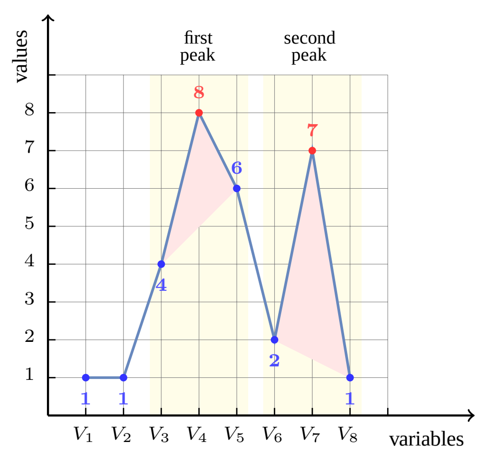


Figure 5.662: Illustration of the first example of the **Example** slot: a sequence of eight variables V_1, V_2, V_3, V_4, V_5, V_6, V_7, V_8 respectively fixed to values 1, 1, 4, 8, 6, 2, 7, 1 and its corresponding two peaks ($\mathtt{N} = 2$)

All solutions

Figure 5.663 gives all solutions to the following non ground instance of the `peak` constraint: $\mathtt{N} \in [1,2]$, $\mathtt{V}_1 \in [1,2]$, $\mathtt{V}_2 = 2$, $\mathtt{V}_3 \in [1,2]$, $\mathtt{V}_4 \in [1,2]$, $\mathtt{V}_5 \in [2,3]$, $\mathtt{peak}(\mathtt{N}, \langle \mathtt{V}_1, \mathtt{V}_2, \mathtt{V}_3, \mathtt{V}_4, \mathtt{V}_5 \rangle)$.

① $(\mathbf{1}, \langle 1, \mathbf{2}, 1, 1, 2 \rangle)$
② $(\mathbf{1}, \langle 1, \mathbf{2}, 1, 1, 3 \rangle)$
③ $(\mathbf{1}, \langle 1, \mathbf{2}, 1, 2, 2 \rangle)$
④ $(\mathbf{1}, \langle 1, \mathbf{2}, 1, 2, 3 \rangle)$
⑤ $(\mathbf{1}, \langle 1, \mathbf{2}, \mathbf{2}, 1, 2 \rangle)$
⑥ $(\mathbf{1}, \langle 1, \mathbf{2}, \mathbf{2}, 1, 3 \rangle)$

Figure 5.663: All solutions corresponding to the non ground example of the `peak` constraint of the **All solutions** slot where each peak is coloured in orange

Typical

$|\mathtt{VARIABLES}| > 2$
$\mathrm{range}(\mathtt{VARIABLES.var}) > 1$

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

- Functional dependency: `N` determined by `VARIABLES`.
- Contractible wrt. `VARIABLES` when $\mathtt{N} = 0$.

Usage

Useful for constraining the number of *peaks* of a sequence of domain variables.

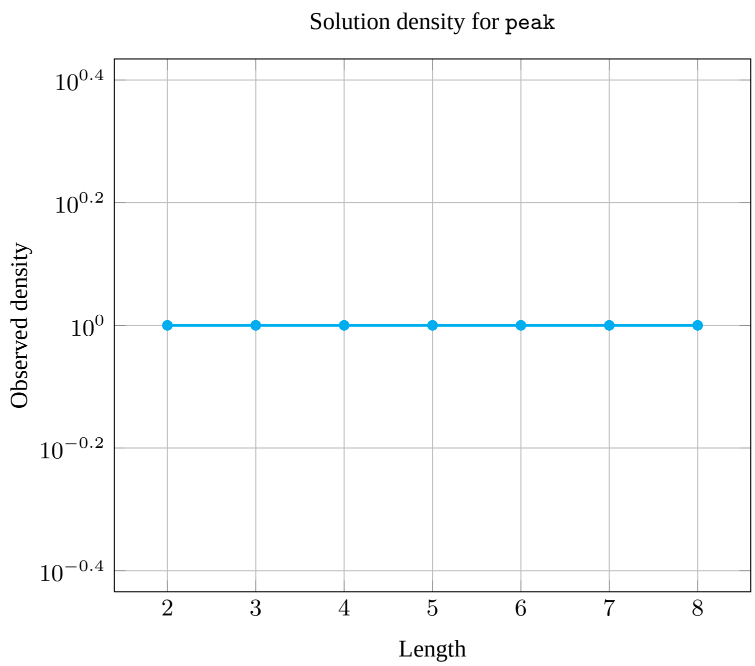
Remark

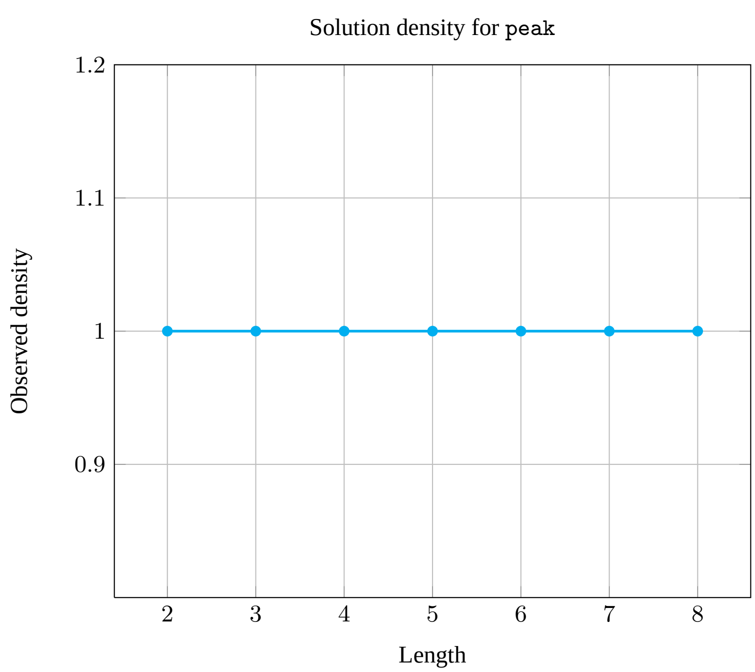
Since the arity of the arc constraint is not fixed, the `peak` constraint cannot be currently described with the graph-based representation. However, this would not hold anymore if we were introducing a slot that specifies how to merge adjacent vertices of the final graph.

Counting

Length (n)	2	3	4	5	6	7	8
Solutions	9	64	625	7776	117649	2097152	43046721

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

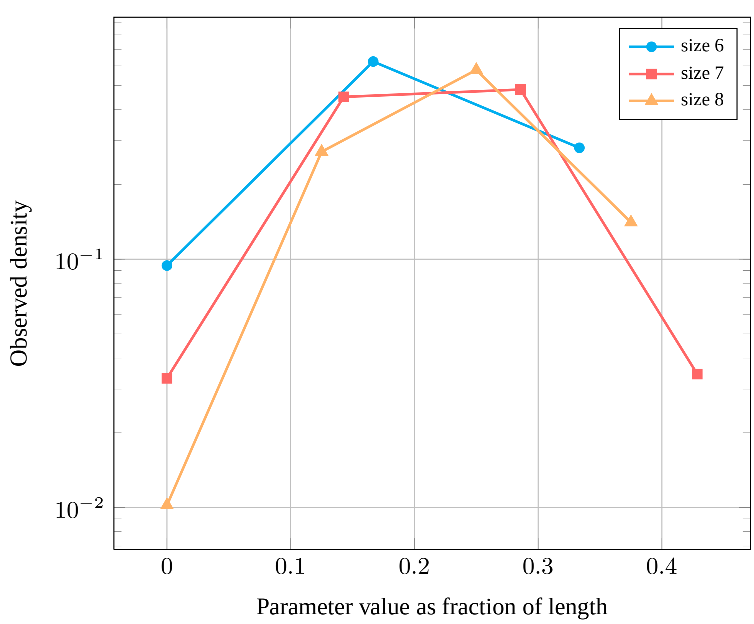

Solution density for peak
Observed density
$10^{0.4}$
$10^{0.2}$
10^{0}
$10^{-0.2}$
$10^{-0.4}$
2
3
4
5
6
7
8
Length

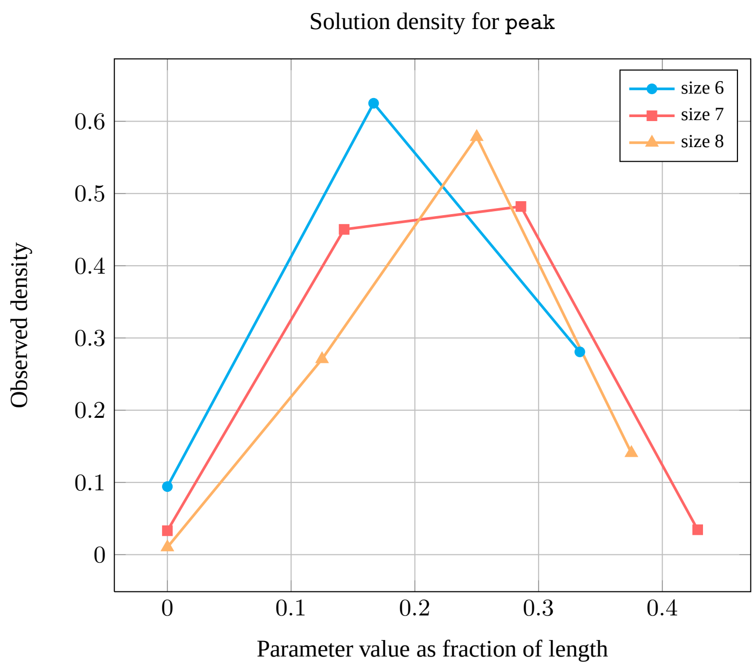

Solution density for peak
Observed density
1.2
1.1
1
0.9
2
3
4
5
6
7
8
Length

Length (n)		2	3	4	5	6	7	8
Total		9	64	625	7776	117649	2097152	43046721
Parameter value	0	9	50	295	1792	11088	69498	439791
	1	-	14	330	5313	73528	944430	11654622
	2	-	-	-	671	33033	1010922	24895038
	3	-	-	-	-	-	72302	6057270

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

Solution density for `peak`





See also

common keyword: `highest_peak`, `inflexion`, `min_dist_between_inflexion`, `min_width_peak` *(sequence)*.

comparison swapped: `valley`.

generalisation: `big_peak` *(a tolerance parameter is added for counting only big peaks)*.

related: `all_equal_peak`, `all_equal_peak_max`, `decreasing_peak`, `increasing_peak`, `no_valley`.

specialisation: `no_peak` *(the variable counting the number of peaks is set to 0 and removed)*.

Keywords

characteristic of a constraint: automaton, automaton with counters, automaton with same input symbol.

combinatorial object: sequence.

constraint arguments: reverse of a constraint, pure functional dependency.

constraint network structure: sliding cyclic(1) constraint network(2).

filtering: glue matrix.

modelling: functional dependency.

Cond. implications

- $\texttt{peak}(\texttt{N}, \texttt{VARIABLES})$
  with $\texttt{N} > 0$
  implies $\texttt{atleast\_nvalue}(\texttt{NVAL}, \texttt{VARIABLES})$
  when $\texttt{NVAL} = 2$.

- $\texttt{peak}(\texttt{N}, \texttt{VARIABLES})$
  implies $\texttt{inflexion}(\texttt{N}, \texttt{VARIABLES})$
  when $\texttt{N} = \texttt{peak}(\texttt{VARIABLES.var}) + \texttt{valley}(\texttt{VARIABLES.var})$.

Automaton Figure 5.664 depicts the automaton associated with the `peak` constraint. To each pair of consecutive variables $(\texttt{VAR}_i, \texttt{VAR}_{i+1})$ of the collection `VARIABLES` corresponds a signature variable S_i. The following signature constraint links $\texttt{VAR}_i$, $\texttt{VAR}_{i+1}$ and S_i: $(\texttt{VAR}_i < \texttt{VAR}_{i+1} \Leftrightarrow S_i = 0) \wedge (\texttt{VAR}_i = \texttt{VAR}_{i+1} \Leftrightarrow S_i = 1) \wedge (\texttt{VAR}_i > \texttt{VAR}_{i+1} \Leftrightarrow S_i = 2)$.

STATES SEMANTICS

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

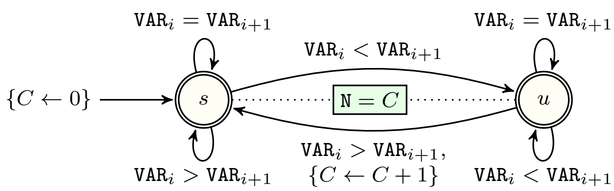


Figure 5.664: Automaton of the `peak` constraint

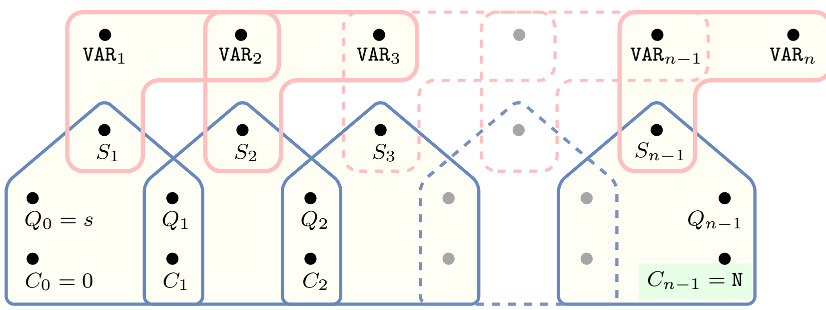


Figure 5.665: Hypergraph of the reformulation corresponding to the automaton of the `peak` constraint (since all states of the automaton are accepting there is no restriction on the last variable Q_{n-1})

Glue matrix where $\overrightarrow{C}$ and $\overleftarrow{C}$ resp. represent the counter value C at the end of a prefix and at the end of the corresponding reverse suffix that partitions the sequence `VARIABLES`.

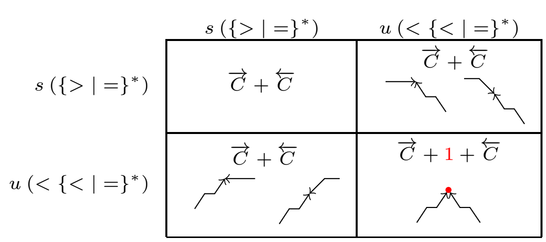


Figure 5.666: Glue matrix of the `peak` constraint

$\texttt{peak}(\texttt{N} = 2, \langle 1, 1, 4, 8, 6, 2, 7, 1 \rangle)$

	1	1	4	8		8	6	2	7	1	
	=	<	<				<	<	>	<	
i	0	1	2	3	:	4	3	2	1	0	i
$\overrightarrow{Q_i}$	s	s	u	u	:	u	u	s	u	s	$\overleftarrow{Q_i}$
$\overrightarrow{C_i}$	0	0	0	0	:	1	1	1	0	0	$\overleftarrow{C_i}$
$\overrightarrow{\texttt{N}_i}$	0	0	0	0	:	1	1	1	0	0	$\overleftarrow{\texttt{N}_i}$

$\texttt{peak}\left(\overrightarrow{\texttt{N}_3} = 0, \langle 1, 1, 4, 8 \rangle\right)$ $\texttt{peak}\left(\overleftarrow{\texttt{N}_4} = 1, \langle 8, 6, 2, 7, 1 \rangle\right)$

glue matrix entry associated with the state pair (u, u):

$$\texttt{N} = \overrightarrow{\texttt{C}_3} + 1 + \overleftarrow{\texttt{C}_4} = 0 + 1 + 1 = 2$$

Figure 5.667: Illustrating the use of the state pair (u, u) of the glue matrix for linking $\texttt{N}$ with the counters variables obtained after reading the prefix $1, 1, 4, 8$ and corresponding suffix $8, 6, 2, 7, 1$ of the sequence $1, 1, 4, 8, 6, 2, 7, 1$; note that the suffix $8, 6, 2, 7, 1$ (in pink) is proceed in reverse order; the left (resp. right) table shows the initialisation (for $i = 0$) and the evolution (for $i > 0$) of the state of the automaton and of its counter C upon reading the prefix $1, 1, 4, 8$ (resp. the suffix $1, 7, 2, 6, 8$).