

5.238 longest_increasing_sequence

	DESCRIPTION	LINKS	AUTOMATON
Origin	constraint on sequences		
Constraint	longest_increasing_sequence(L, VARIABLES)		
Synonym	size_longest_increasing_sequence.		
Arguments	L : dvar VARIABLES : collection(var-dvar)		
Restrictions	L ≥ 0 L < range(VARIABLES.var) required(VARIABLES, var)		
Purpose	<p>L is the largest difference between the first and the last value of the maximum increasing sequences of the collection VARIABLES.</p> <p>A sequence of consecutive variables X_i, X_{i+1}, \dots, X_j ($1 \leq i \leq j \leq \text{VARIABLES}$) of the collection of variables VARIABLES is a <i>maximum increasing sequence</i> if all the following conditions simultaneously apply:</p> <ul style="list-style-type: none">• $X_i \leq X_{i+1} \leq \dots \leq X_j$,• $i = 1$ or $X_{i-1} > X_i$,• $i = \text{VARIABLES}$ or $X_j > X_{j+1}$.		
Example	<div><div>(7, <10, 8, 8, 6, 4, 9, 11, 8>)</div><div>(0, <10, 8, 7, 5, 4, 3, 1, 0>)</div></div> <p>Figure 5.526 gives a graphical representation of the first example of the Example slot with its two maximum increasing sequences in red of respective size 0 and 7. The corresponding longest_increasing_sequence constraint holds since its first argument L is fixed to the maximum size 7.</p>		
Typical	L > 0 VARIABLES > 1 nval(VARIABLES.var) > 2		
Symmetry	One and the same constant can be added to the var attribute of all items of VARIABLES.		
Arg. properties	Functional dependency: L determined by VARIABLES.		
Counting			

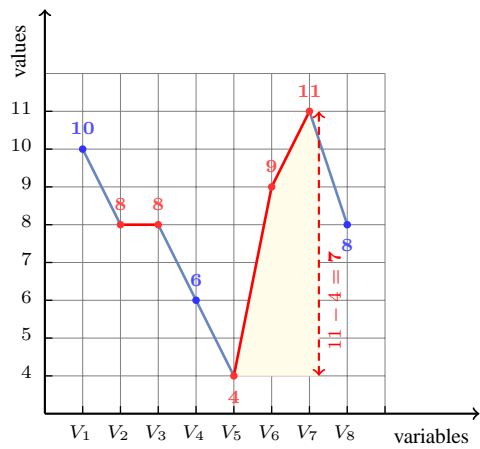
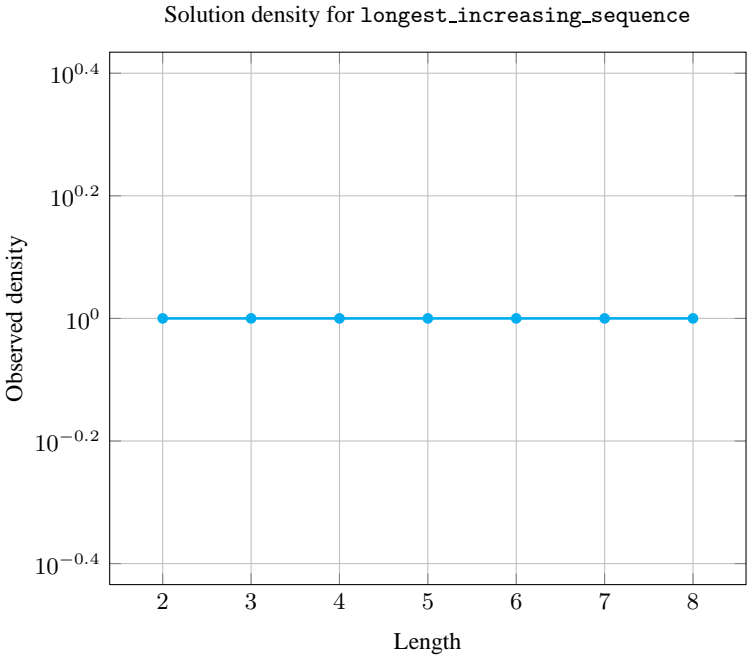
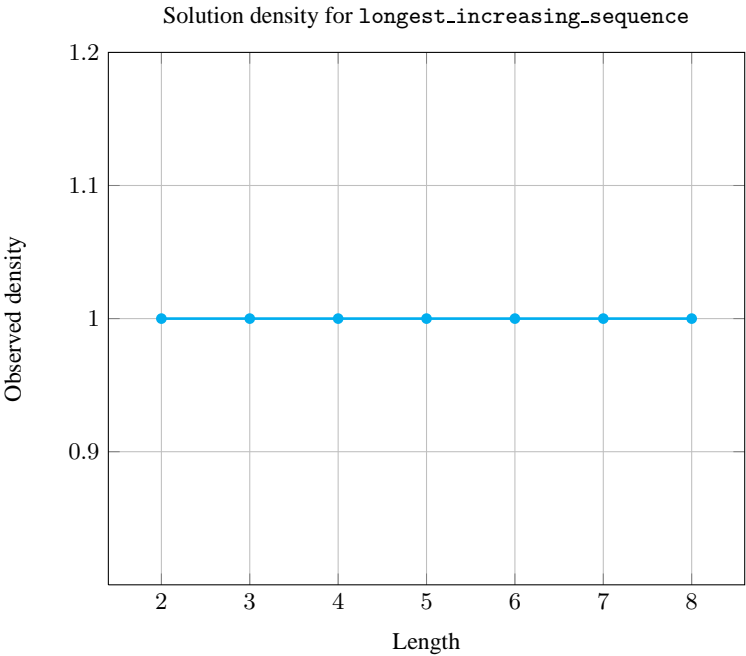


Figure 5.526: Illustration of the first example of the **Example** slot: a sequence of eight variables $V_1, V_2, V_3, V_4, V_5, V_6, V_7, V_8$ respectively fixed to values 10, 8, 8, 6, 4, 9, 11, 8 and its two maximum increasing sequences in red of respective size $8 - 8 = 0$ and $11 - 4 = 7$.

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

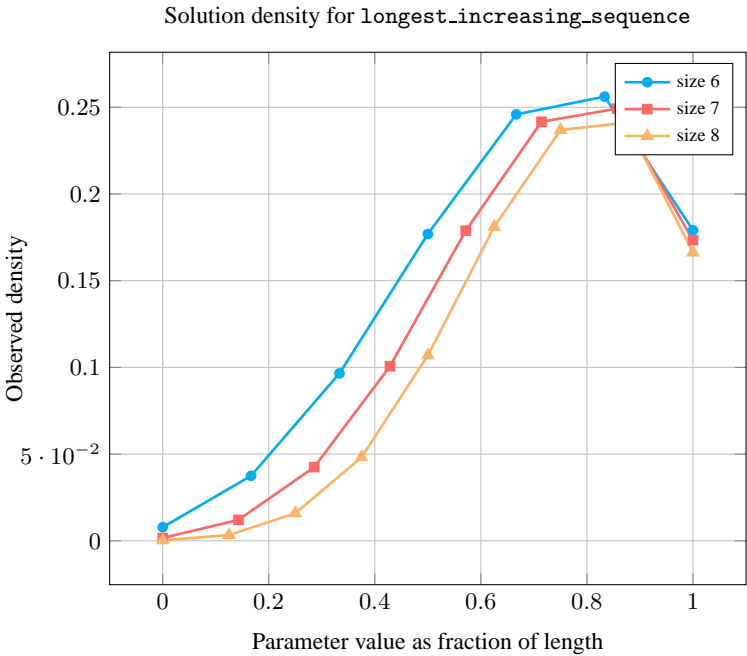
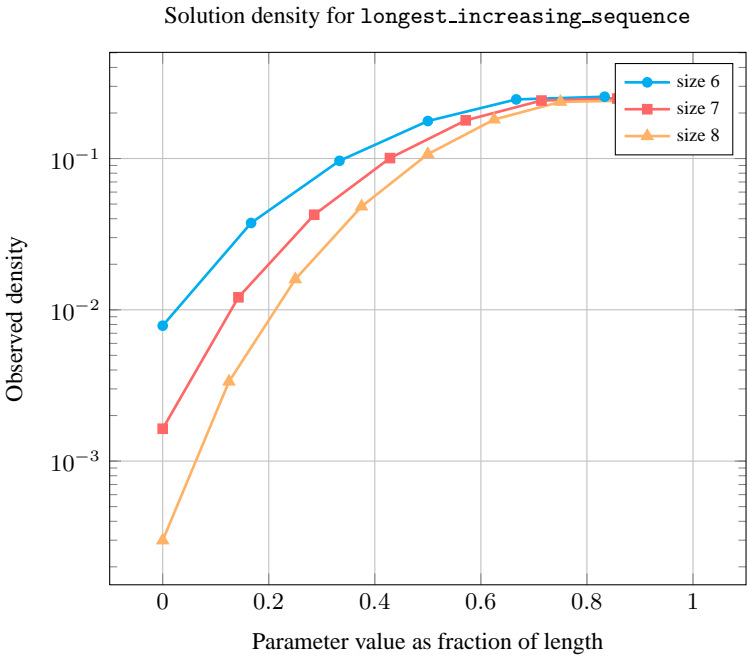
Number of solutions for longest_increasing_sequence: domains 0.. n





Length (<i>n</i>)		2	3	4	5	6	7	8
Total		9	64	625	7776	117649	2097152	43046721
Parameter value	0	6	20	70	252	924	3432	12870
	1	2	18	122	750	4412	25382	144314
	2	1	16	161	1398	11361	89132	685090
	3	-	10	162	1942	20816	211106	2074365
	4	-	-	110	2024	28930	375084	4603682
	5	-	-	-	1410	30134	506766	7792840
	6	-	-	-	-	21072	522648	10197174
	7	-	-	-	-	-	363602	10379696
	8	-	-	-	-	-	-	7156690

Solution count for longest_increasing_sequence: domains 0..*n*



See also

common keyword: longest_decreasing_sequence,
min_dist_between_inflexion(sequence).

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.

filtering: glue matrix.

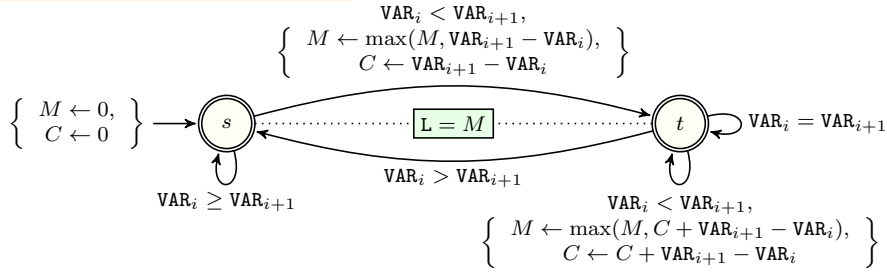
modelling: functional dependency.

Automaton

Figure 5.527 depicts the automaton associated with the `longest_increasing_sequence` constraint.

STATES SEMANTICS

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



Blue matrix where \vec{M} , \vec{C} and \overleftarrow{M} , \overleftarrow{C} resp. represent the counters values M , C at the end of a prefix and at the end of the corresponding reverse suffix that partitions the sequence `VARIABLES`.

	$s (\{< =\}^*)$	$t (> \{> =\}^*)$
$s (\{> =\}^*)$	$\max(\vec{M}, \overleftarrow{M})$	$\max(\vec{M}, \overleftarrow{M})$
$t (< \{< =\}^*)$	$\max(\vec{M}, \overleftarrow{M})$	$\max(\vec{M}, \vec{C} + \overleftarrow{C}, \overleftarrow{M})$

Figure 5.527: Automaton of the `longest_increasing_sequence` constraint and its glue matrix (note that the reverse of the `longest_increasing_sequence` constraint is the `longest_decreasing_sequence` constraint)

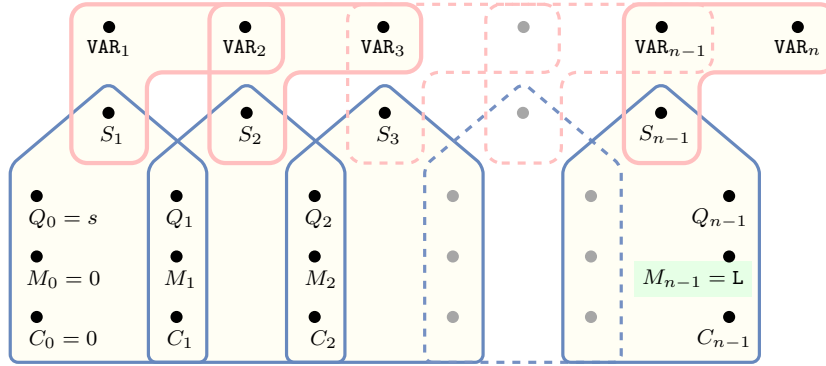


Figure 5.528: Hypergraph of the reformulation corresponding to the automaton of the `longest_increasing_sequence` constraint