

5.217 `length_last_sequence`

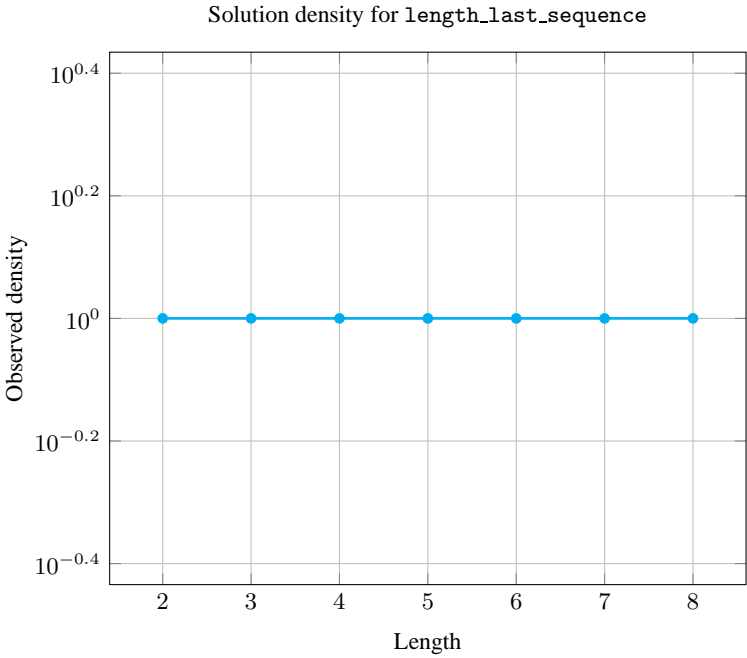
	DESCRIPTION	LINKS	AUTOMATON
Origin	Inspired by stretch_path		
Constraint	<code>length_last_sequence(LEN, VARIABLES)</code>		
Arguments	LEN : dvar VARIABLES : collection (var — dvar)		
Restrictions	$LEN \geq 0$ $LEN \leq VARIABLES $ required (VARIABLES, var)		
Purpose	LEN is the length of the maximum sequence of variables that take the same value that contains the last variable of the collection VARIABLES (or 0 if the collection is empty).		
Example	<div> $(1, \langle 4, 4, 4, 5, 5, 4 \rangle)$ $(6, \langle 4, 4, 4, 4, 4, 4 \rangle)$ $(5, \langle 2, 4, 4, 4, 4, 4 \rangle)$ </div> <p>The first <code>length_last_sequence</code> constraint holds since the sequence associated with the last value of the collection <code>VARIABLES</code> = $\langle 4, 4, 4, 5, 5, 4 \rangle$ spans over a single variable.</p>		
Typical	$LEN < VARIABLES $ $ VARIABLES > 1$		
Symmetry	All occurrences of two distinct values of <code>VARIABLES.var</code> can be swapped ; all occurrences of a value of <code>VARIABLES.var</code> can be renamed to any unused value.		
Arg. properties	Functional dependency : LEN determined by VARIABLES.		
Reformulation	<p>Without loss of generality let assume that the collection <code>VARIABLES</code> = $\langle V_1, V_2, \dots, V_n \rangle$ has more than one variable. By introducing $2 \cdot n - 1$ 0-1 variables, the <code>length_last_sequence(LEN, VARIABLES)</code> constraint can be expressed in term of $2 \cdot n - 1$ reified constraints and one arithmetic constraint (i.e., a sum_ctr constraint). We first introduce $n - 1$ variables that are respectively set to 1 if and only if two given consecutive variables of the collection <code>VARIABLES</code> are equal:</p> $B_{n-1,n} \Leftrightarrow V_{n-1} = V_n,$ $B_{n-2,n-1} \Leftrightarrow V_{n-2} = V_{n-1},$ $\dots\dots\dots$ $B_{1,2} \Leftrightarrow V_1 = V_2.$ <p>We then introduce n variables A_n, A_{n-1}, \dots, A_1 that are respectively associated to the different sliding sequences ending on the last variable of the sequence $V_1 V_2 \dots V_n$. Variable A_i is set to 1 if and only if $V_n = V_{n-1} = \dots = V_i$:</p>		

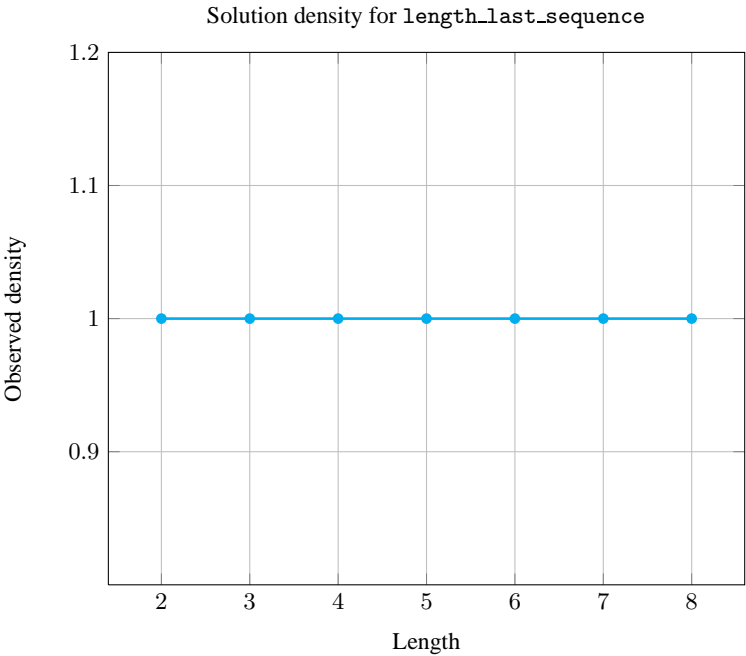
$A_n = 1,$
 $A_{n-1} \Leftrightarrow B_{n-1,n} \wedge A_n,$
 $A_{n-2} \Leftrightarrow B_{n-2,n-1} \wedge A_{n-1},$
.....
 $A_1 \Leftrightarrow B_{1,2} \wedge A_2.$
Finally we state the following arithmetic constraint:
 $LEN = A_n + A_{n-1} + \cdots + A_1.$

Counting

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

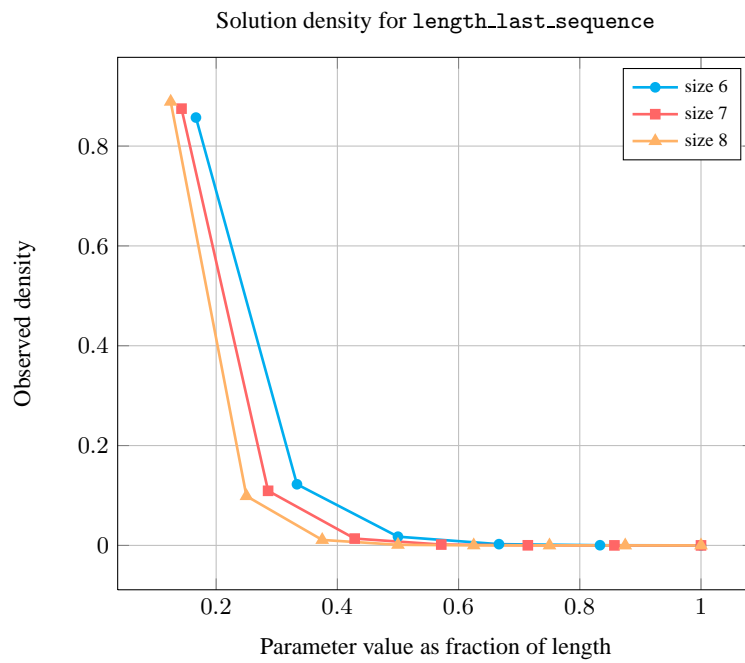
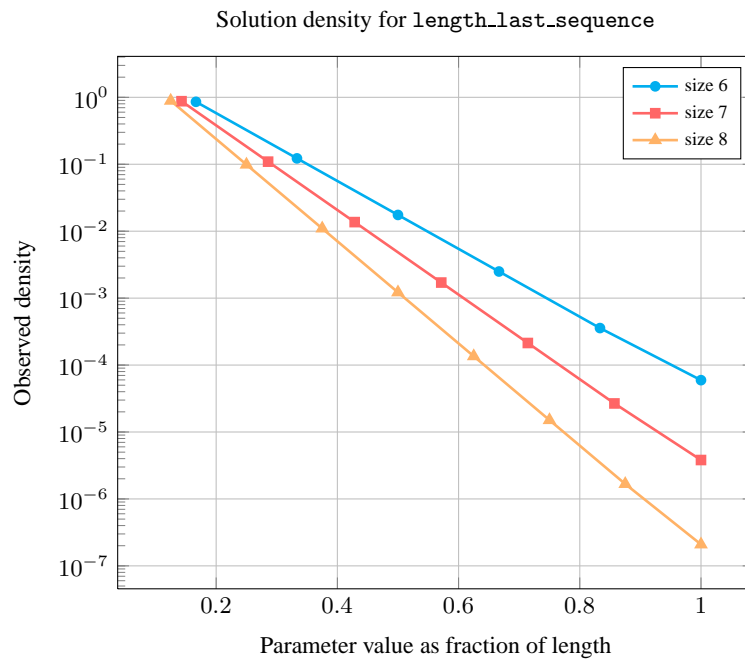
Number of solutions for length_last_sequence: domains 0..*n*





Length (<i>n</i>)		2	3	4	5	6	7	8
Total		9	64	625	7776	117649	2097152	43046721
Parameter value	1	6	48	500	6480	100842	1835008	38263752
	2	3	12	100	1080	14406	229376	4251528
	3	-	4	20	180	2058	28672	472392
	4	-	-	5	30	294	3584	52488
	5	-	-	-	6	42	448	5832
	6	-	-	-	-	7	56	648
	7	-	-	-	-	-	8	72
	8	-	-	-	-	-	-	9

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



See also [common keyword: length_first_sequence](#) (*counting constraint, sequence*).

Keywords [characteristic of a constraint](#): automaton, automaton with counters.

combinatorial object: sequence.

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

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

constraint type: value constraint, counting constraint.

filtering: glue matrix.

modelling: functional dependency.

Automaton

Figure 5.491 depicts the automaton associated with the `length_last_sequence` 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 : $VAR_i = VAR_{i+1} \Leftrightarrow S_i$.

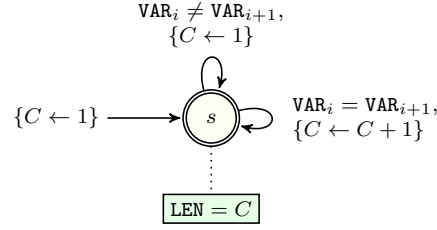


Figure 5.491: Automaton of the `length_last_sequence` constraint when $|VARIABLES| \geq 2$

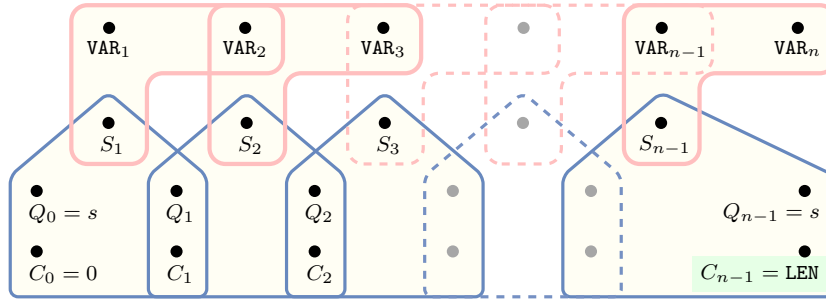


Figure 5.492: Hypergraph of the reformulation corresponding to the automaton of the `length_last_sequence` constraint

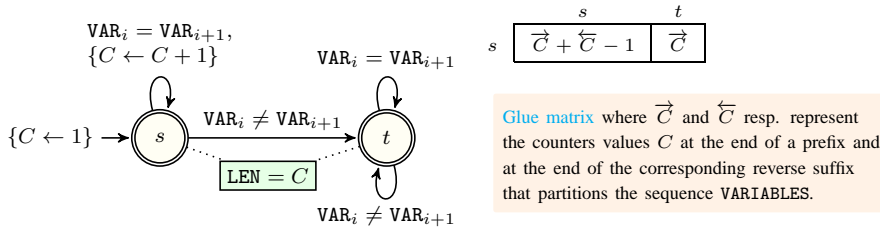


Figure 5.493: Automaton of the reverse of the `length_last_sequence` constraint (i.e., the `length_first_sequence` constraint) when $|VARIABLES| \geq 2$ and corresponding glue matrix between `length_last_sequence` and its reverse `length_first_sequence`