5.37 atleast_nvalue

DESCRIPTION LINKS GRAPH

Origin [341]

Constraint `atleast_nvalue(NVAL, VARIABLES)`

Synonym `k_diff`.

Arguments

`NVAL` : `dvar`
`VARIABLES` : `collection(var−dvar)`

Restrictions

`required(VARIABLES, var)`
$\texttt{NVAL} \geq 0$
$\texttt{NVAL} \leq |\texttt{VARIABLES}|$
$\texttt{NVAL} \leq \texttt{range}(\texttt{VARIABLES.var})$

Purpose

The number of distinct values taken by the variables of the collection `VARIABLES` is greater than or equal to `NVAL`.

Example

$(2, \langle 3, 1, 7, 1, 6 \rangle)$
$(4, \langle 3, 1, 7, 1, 6 \rangle)$
$(5, \langle 3, 1, 7, 0, 6 \rangle)$

The first `atleast_nvalue` constraint holds since the collection $\langle 3, 1, 7, 1, 6 \rangle$ involves at least 2 distinct values (i.e., in fact 4 distinct values).

All solutions

Figure 5.105 gives all solutions to the following non ground instance of the `atleast_nvalue` constraint: $\texttt{NVAL} \in [\mathbf{3}, \mathbf{4}]$, $\texttt{V}_1 \in [1, 2]$, $\texttt{V}_2 = 3$, $\texttt{V}_3 \in [3, 4]$, $\texttt{V}_4 \in [2, 3]$, $\texttt{atleast\_nvalue}(\texttt{NVAL}, \langle \texttt{V}_1, \texttt{V}_2, \texttt{V}_3, \texttt{V}_4 \rangle)$.

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

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

Figure 5.105: All solutions corresponding to the non ground example of the `atleast_nvalue` constraint of the **All solutions** slot

Typical

$\texttt{NVAL} > 0$
$\texttt{NVAL} < |\texttt{VARIABLES}|$
$\texttt{NVAL} < \texttt{range}(\texttt{VARIABLES.var})$
$|\texttt{VARIABLES}| > 1$

Symmetries

- NVAL can be decreased to any value ≥ 0.
- Items of VARIABLES are permutable.
- All occurrences of two distinct values of VARIABLES.var can be swapped; all occurrences of a value of VARIABLES.var can be renamed to any unused value.

Arg. properties

Extensible wrt. VARIABLES.

Remark

The atleast_nvalue constraint was first introduced by J.-C. Régin under the name k_diff in [341]. Later on the atleast_nvalue constraint was introduced together with the atmost_nvalue constraint by C. Bessière *et al.* in an article [62] providing filtering algorithms for the nvalue constraint.

Algorithm

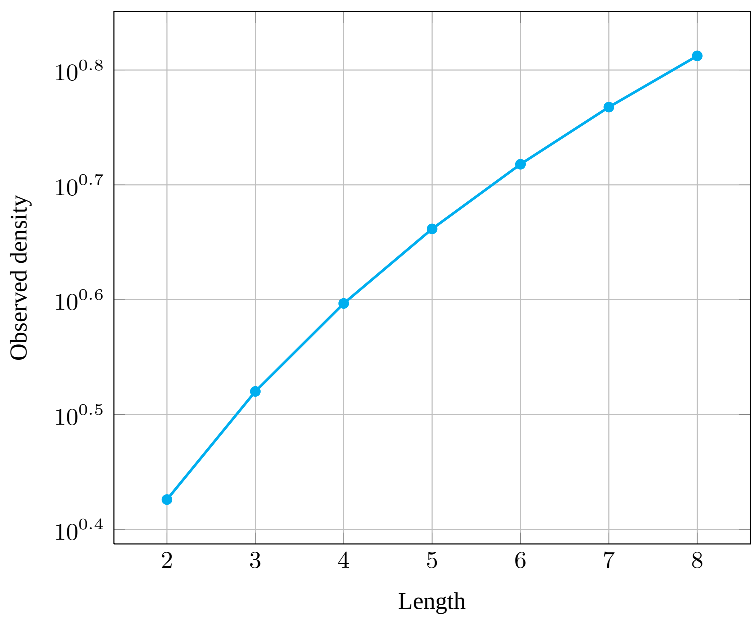
[62] provides a sketch of a filtering algorithm enforcing arc-consistency for the atleast_nvalue constraint. This algorithm is based on the maximal matching in a bipartite graph.

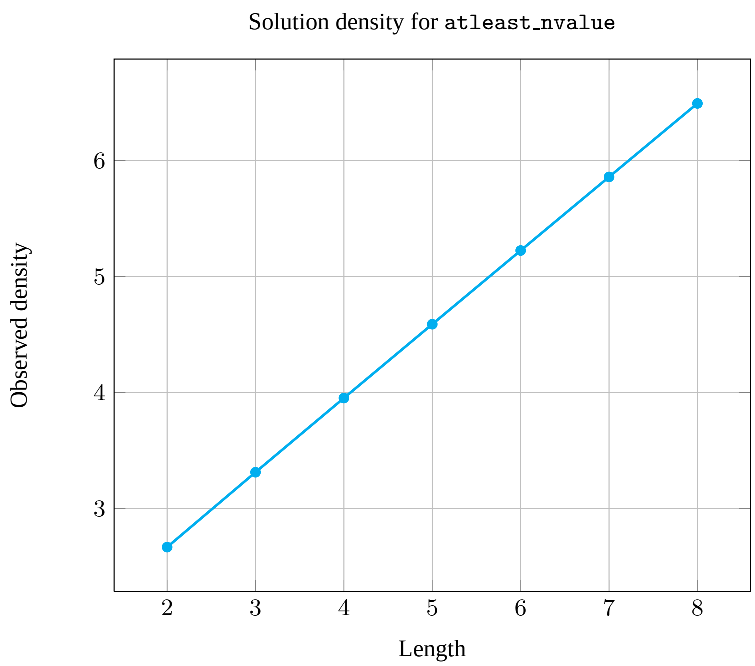
Counting

Length (n)	2	3	4	5	6	7	8
Solutions	24	212	2470	35682	614600	12286024	279472266

Number of solutions for atleast_nvalue: domains $0..n$

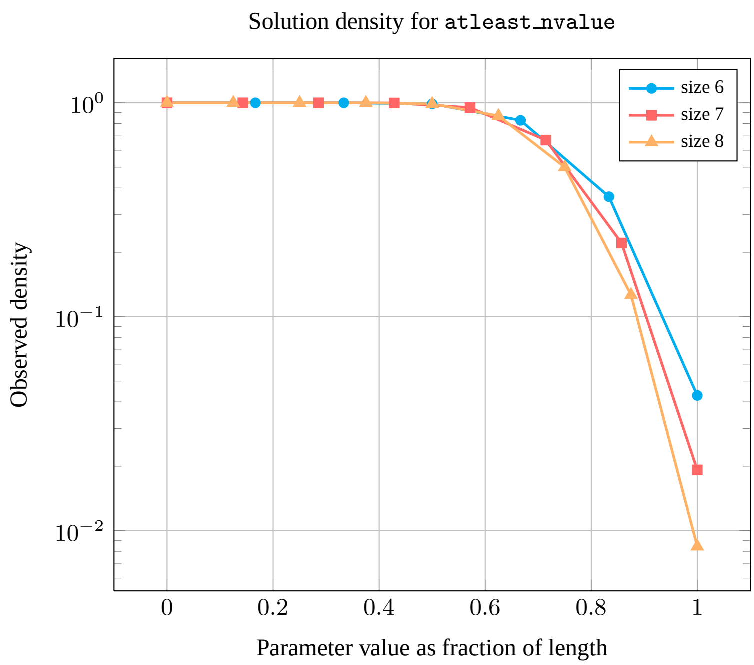
Solution density for atleast_nvalue

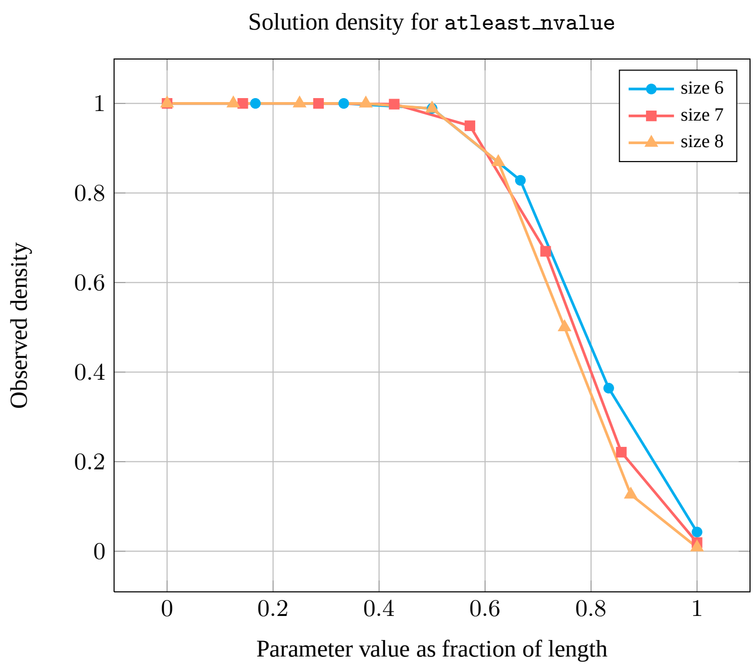




Length (n)		2	3	4	5	6	7	8
Total		24	212	2470	35682	614600	12286024	279472266
Parameter value	0	9	64	625	7776	117649	2097152	43046721
	1	9	64	625	7776	117649	2097152	43046721
	2	6	60	620	7770	117642	2097144	43046712
	3	-	24	480	7320	116340	2093616	43037568
	4	-	-	120	4320	97440	1992480	42550704
	5	-	-	-	720	42840	1404480	37406880
	6	-	-	-	-	5040	463680	21530880
	7	-	-	-	-	-	40320	5443200
	8	-	-	-	-	-	-	362880

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





See also **comparison swapped:** `atmost_nvalue`.

implied by: `and`, `equivalent`, `imply`, `nand`, `nor`, `nvalue` ($\geq$ NVAL *replaced by* $=$ NVAL), `nvisible_from_end`, `nvisible_from_start`, `or`, `size_max_seq_alldifferent`, `size_max_starting_seq_alldifferent`, `xor`.

uses in its reformulation: `not_all_equal`.

Keywords

constraint type: counting constraint, value partitioning constraint.

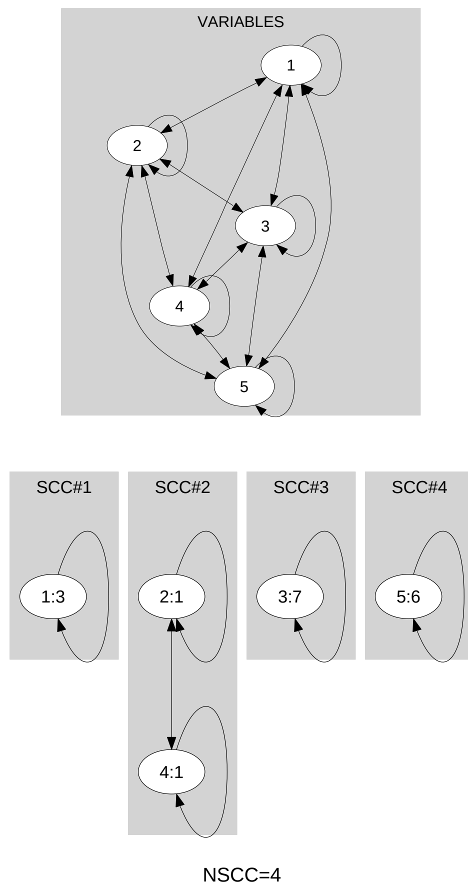
filtering: bipartite matching, arc-consistency.

final graph structure: strongly connected component, equivalence.

modelling: number of distinct equivalence classes, number of distinct values.

Arc input(s)	VARIABLES
Arc generator	*CLIQUE*$\mapsto$collection(variables1, variables2)
Arc arity	2
Arc constraint(s)	variables1.var $=$ variables2.var
Graph property(ies)	**NSCC**$\geq$ NVAL
Graph class	EQUIVALENCE

Graph model Parts (A) and (B) of Figure 5.106 respectively show the initial and final graph associated with the first example of the **Example** slot. Since we use the **NSCC** graph property we show the different strongly connected components of the final graph. Each strongly connected component corresponds to a specific value that is assigned to some variables of the VARIABLES collection. The 4 following values 1, 3, 6 and 7 are used by the variables of the VARIABLES collection.



(A)

(B)

Figure 5.106: Initial and final graph of the `atleast_nvalue` constraint