5.202 inverse_within_range

DESCRIPTION LINKS GRAPH

Origin Derived from inverse.

Constraint inverse_within_range(X, Y)

Synonyms inverse_in_range, inverse_range.

Arguments

```
X : collection(var−dvar)
Y : collection(var−dvar)
```

Restrictions

```
required(X, var)
required(Y, var)
```

Purpose

If the i^{th} variable of the collection X is assigned to j and if j is greater than or equal to 1 and less than or equal to the number of items of the collection Y then the j^{th} variable of the collection Y is assigned to i.

Conversely, if the j^{th} variable of the collection Y is assigned to i and if i is greater than or equal to 1 and less than or equal to the number of items of the collection X then the i^{th} variable of the collection X is assigned to j.

Example $(\langle 9, 4, 2\rangle, \langle 9, 3, 9, 2\rangle)$

Since the second item of X is assigned to 4, the fourth item of Y is assigned to 2. Similarly, since the third item of X is assigned to 2, the second item of Y is assigned to 3. Figure 5.452 illustrates the correspondence between X and Y.

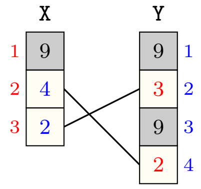

Figure 5.452: Correspondence between the items of $\mathtt{X} = \langle 9, 4, 2\rangle$ and the items of $\mathtt{Y} = \langle 9, 3, 9, 2\rangle$: on the X side values between 1 and $|\mathtt{Y}| = 4$ are shown in blue, on the Y side values between 1 and $|\mathtt{X}| = 3$ are shown in red.

Typical

$|\mathtt{X}| > 1$
range(X.var) > 1
$|\mathtt{Y}| > 1$
range(Y.var) > 1

Symmetry

Arguments are permutable w.r.t. permutation $(\mathtt{X}, \mathtt{Y})$.

Usage

Consider an integer value m and a sequence of n variables S from which you have to select a subsequence S' such that:

- All variables of S' have to be assigned to distinct values from $[1, m]$,
- All variables not in S' have to be assigned a value, not necessarily distinct, outside $[1, m]$.

As for the `inverse` constraint we may want to create explicitly a *value variable* for each value in [1,m] in order to state some specific constraints on the *value variables* or to use a heuristics involving the original variables of S as well as the *value variables*. The purpose of the `inverse_within_range` constraint is to link the variables of S with the *value variables*.

See also

common keyword: `inverse_set` *(channelling constraint)*.

specialisation: `inverse` *(the 2 collections have not necessarly the same number of items)*.

Keywords

constraint type: graph constraint.

final graph structure: bipartite, no loop, symmetric.

heuristics: heuristics.

modelling: channelling constraint, dual model.

Arc input(s)	X Y
Arc generator	*SYMMETRIC_PRODUCT*↦collection(s1, s2)
Arc arity	2
Arc constraint(s)	s1.var = s2.key
Graph class	● BIPARTITE ● NO_LOOP ● SYMMETRIC