5.179 in_interval_reified

DESCRIPTION LINKS

Origin Reified version of in_interval.

Constraint in_interval_reified(VAR, LOW, UP, B)

Synonyms dom_reified, in_reified.

Arguments

VAR : dvar
LOW : int
UP : int
B : dvar

Restrictions

LOW $\leq$ UP
B ≥ 0
B ≤ 1

Purpose Enforce the following equivalence, VAR $\in$ [LOW, UP] $\Leftrightarrow$ B.

Example $(3, 2, 5, 1)$

The in_interval_reified constraint holds since:

- Its first argument VAR $= 3$ is greater than or equal to its second argument LOW $= 2$ and less than or equal to its third argument UP $= 5$ (i.e., $3 \in [2, 5]$).
- The corresponding Boolean variable B is set to 1 since condition $3 \in [2, 5]$ holds.

Typical

VAR $\neq$ LOW
VAR $\neq$ UP
LOW $<$ UP

Symmetries

- An occurrence of a value of VAR that belongs to [LOW, UP] (resp. does not belong to [LOW, UP]) can be replaced by any other value in [LOW, UP] (resp. not in [LOW, UP]).
- One and the same constant can be added to VAR, LOW and UP.

Reformulation The in_interval_reified constraint can be reformulated in terms of linear constraints. For convenience, we rename VAR to x, LOW to l, UP to u, and B to y. The constraint is decomposed into the following conjunction of constraints:

$$\begin{aligned} x \geq l \quad &\Leftrightarrow y_1, \\ x \leq u \quad &\Leftrightarrow y_2, \\ y_1 \wedge y_2 &\Leftrightarrow y \,. \end{aligned}$$

We show how to encode these constraints with linear inequalities. The first constraint, i.e., $x \geq l \Leftrightarrow y_1$ is encoded by posting one of the following three constraints:

$$\begin{cases} \text{a)} & \text{if } \underline{x} \geq l: & y_1 = 1, \\ \text{b)} & \text{if } \overline{x} < l: & y_1 = 0, \\ \text{c)} & \text{otherwise}: & x \geq (l - \underline{x}) \cdot y_1 + \underline{x} \ \wedge \ x \leq (\overline{x} - l + 1) \cdot y_1 + l - 1. \end{cases}$$

On the one hand, cases a) and b) correspond to situations where one can fix y_1, no matter what value will be assigned to x. On the other hand, in case c), y_1 can take both values 0 or 1 depending on the value assigned to x. As shown by Figure 5.404, all possible solutions for the pair of variables (x, y_1) satisfy the following two linear inequalities $x \geq (l - \underline{x}) \cdot y_1 + \underline{x}$ and $x \leq (\overline{x} - l + 1) \cdot y_1 + l - 1$. The first inequality discards all points that are above the line that goes through the two extreme solution points $(\underline{x}, 0)$ and $(l, 1)$, while the second one removes all points that are below the line that goes through the two extreme solution points $(l - 1, 0)$ and $(\overline{x}, 1)$.

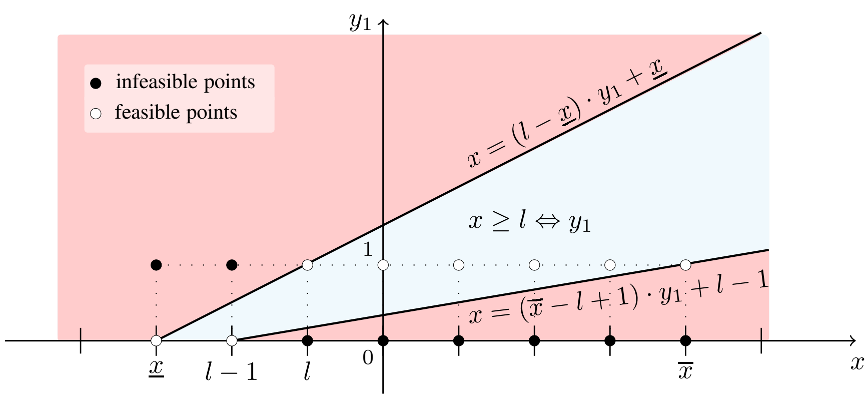

Figure 5.404: Illustration of the reformulation of the reified constraint $x \geq l \Leftrightarrow y_1$ with two linear inequalities

The second constraint, i.e., $x \leq u \Leftrightarrow y_2$ is encoded by posting one of the following three constraints:

$$\begin{cases} \text{d)} & \text{if } \overline{x} \leq u: & y_2 = 1, \\ \text{e)} & \text{if } \underline{x} > u: & y_2 = 0, \\ \text{f)} & \text{otherwise}: & x \leq (u - \overline{x}) \cdot y_2 + \overline{x} \ \wedge \ x \geq (\underline{x} - u - 1) \cdot y_2 + u + 1. \end{cases}$$

On the one hand, cases d) and e) correspond to situations where one can fix y_2, no matter what value will be assigned to x. On the other hand, in case f), y_2 can take both value 0 or 1 depending on the value assigned to x. As shown by Figure 5.405, all possible solutions for the pair of variables (x, y_2) satisfy the following two linear inequalities $x \leq (u - \overline{x}) \cdot y_2 + \overline{x}$ and $x \geq (\underline{x} - u - 1) \cdot y_2 + u + 1$. The first inequality discards all points that are above the line that goes through the two extreme solution points $(\overline{x}, 0)$ and $(u, 1)$, while the second one removes all points that are below the line that goes through the two extreme solution points $(u + 1, 0)$ and $(\underline{x}, 1)$.

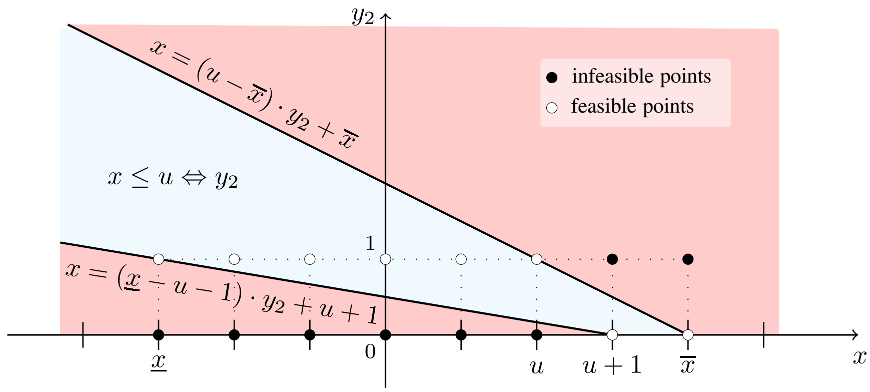

Figure 5.405: Illustration of the reformulation of the reified constraint $x \leq u \Leftrightarrow y_2$ with two linear inequalities

The third constraint, i.e., $y_1 \wedge y_2 \Leftrightarrow y$ is encoded as:

$$\begin{cases} \text{g)} & y \geq y_1 + y_2 - 1, \\ \text{h)} & y \leq y_1, \\ \text{i)} & y \leq y_2. \end{cases}$$

Case g) handles the implication $y_1 \wedge y_2 \Rightarrow y$, while cases h) and i) take care of the other side $y \Rightarrow y_1 \wedge y_2$.

See also

specialisation: `in_interval`.

uses in its reformulation: `alldifferent` *(bound consistency preserving reformulation)*.

Keywords

characteristic of a constraint: reified constraint.

constraint arguments: binary constraint.

constraint type: predefined constraint, value constraint.

filtering: arc-consistency.