## 5.170 graph_crossing

DESCRIPTION LINKS GRAPH

**Origin** N. Beldiceanu

**Constraint** graph_crossing(NCROSS, NODES)

**Synonyms** crossing, ncross.

**Arguments**

| | | |
|---|---|---|
| NCROSS | : | dvar |
| NODES | : | collection(succ−dvar, x−int, y−int) |

**Restrictions**

$\texttt{NCROSS} \geq 0$
required(NODES, [succ, x, y])
$\texttt{NODES.succ} \geq 1$
$\texttt{NODES.succ} \leq |\texttt{NODES}|$

**Purpose** NCROSS is the number of proper intersections between line segments, where each line segment is an arc of the directed graph defined by the arc linking a node and its unique successor.

**Example**

$$\left(2, \left\langle \begin{array}{lll} \texttt{succ}-1 & \texttt{x}-4 & \texttt{y}-7, \\ \texttt{succ}-1 & \texttt{x}-2 & \texttt{y}-5, \\ \texttt{succ}-1 & \texttt{x}-7 & \texttt{y}-6, \\ \texttt{succ}-2 & \texttt{x}-1 & \texttt{y}-2, \\ \texttt{succ}-3 & \texttt{x}-2 & \texttt{y}-2, \\ \texttt{succ}-2 & \texttt{x}-5 & \texttt{y}-3, \\ \texttt{succ}-3 & \texttt{x}-8 & \texttt{y}-2, \\ \texttt{succ}-9 & \texttt{x}-6 & \texttt{y}-2, \\ \texttt{succ}-10 & \texttt{x}-10 & \texttt{y}-6, \\ \texttt{succ}-8 & \texttt{x}-10 & \texttt{y}-1 \end{array} \right\rangle\right)$$

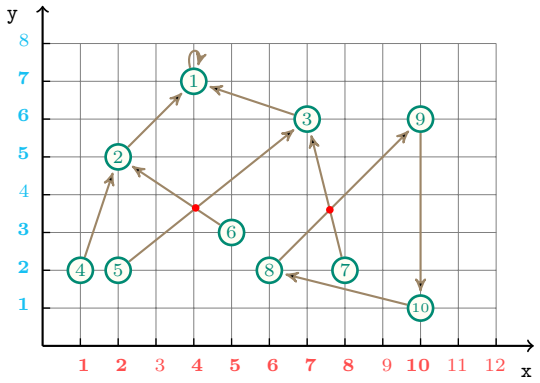
Figure 5.366 shows the line segments associated with the NODES collection. One can note the following line segments intersection:

- Arcs $8 \rightarrow 9$ and $7 \rightarrow 3$ cross,
- Arcs $5 \rightarrow 3$ and $6 \rightarrow 2$ cross also.

Consequently, the graph_crossing constraint holds since its first argument NCROSS is set to 2.

**Typical**

$|\texttt{NODES}| > 1$
$\texttt{range}(\texttt{NODES.succ}) > 1$
$\texttt{range}(\texttt{NODES.x}) > 1$
$\texttt{range}(\texttt{NODES.y}) > 1$




NODES

| | | | |
|---|---|---|---|
| 1: | succ – 1 | x – 4 | y – 7 |
| 2: | succ – 1 | x – 2 | y – 5 |
| 3: | succ – 1 | x – 7 | y – 6 |
| 4: | succ – 2 | x – 1 | y – 2 |
| 5: | succ – 3 | x – 2 | y – 2 |
| 6: | succ – 2 | x – 5 | y – 3 |
| 7: | succ – 3 | x – 8 | y – 2 |
| 8: | succ – 9 | x – 6 | y – 2 |
| 9: | succ – 10 | x – 10 | y – 6 |
| 10: | succ – 8 | x – 10 | y – 1 |

Figure 5.366: Illustration of the **Example** slot: a graph covering with 2 line segments intersections in red (NCROSS = 2)

**Symmetries**

- Attributes of NODES are permutable w.r.t. permutation (succ) (x, y) (*permutation applied to all items*).
- One and the same constant can be added to the x attribute of all items of NODES.
- One and the same constant can be added to the y attribute of all items of NODES.

**Arg. properties**

Functional dependency: NCROSS determined by NODES.

**Usage**

This is a general crossing constraint that can be used in conjunction with one graph covering constraint such as cycle, tree or map. In many practical problems ones want not only to cover a graph with specific patterns but also to avoid too much crossing between the arcs of the final graph.

**Remark**

We did not give a specific crossing constraint for each graph covering constraint. We feel that it is better to start first with a more general constraint before going in the specificity of the pattern that is used for covering the graph.

**See also**

**common keyword:** crossing *(line segments intersection)*, cycle, map, tree *(graph constraint,graph partitioning constraint)*, two_layer_edge_crossing *(line segments intersection)*.

**Keywords**

**constraint arguments:** pure functional dependency.

**constraint type:** graph constraint, graph partitioning constraint.

**geometry:** geometrical constraint, line segments intersection.

**modelling:** functional dependency.

**Arc input(s)** NODES

**Arc generator** $CLIQUE(<) \mapsto \texttt{collection}(\texttt{n1}, \texttt{n2})$

**Arc arity** 2

**Arc constraint(s)**

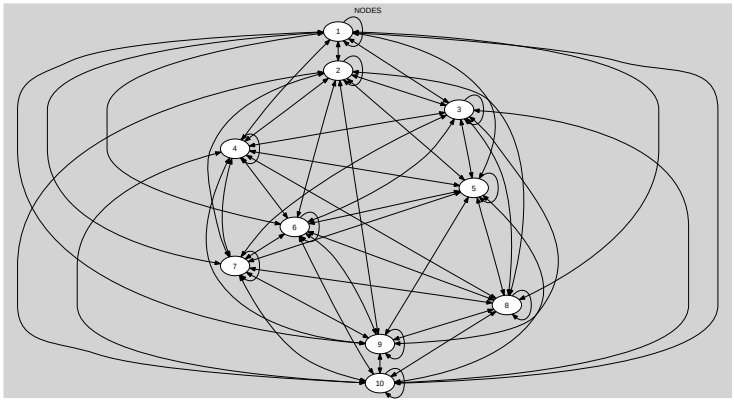
- $\max(\texttt{n1.x}, \texttt{NODES}[\texttt{n1.succ}].\texttt{x}) \geq \min(\texttt{n2.x}, \texttt{NODES}[\texttt{n2.succ}].\texttt{x})$
- $\max(\texttt{n2.x}, \texttt{NODES}[\texttt{n2.succ}].\texttt{x}) \geq \min(\texttt{n1.x}, \texttt{NODES}[\texttt{n1.succ}].\texttt{x})$
- $\max(\texttt{n1.y}, \texttt{NODES}[\texttt{n1.succ}].\texttt{y}) \geq \min(\texttt{n2.y}, \texttt{NODES}[\texttt{n2.succ}].\texttt{y})$
- $\max(\texttt{n2.y}, \texttt{NODES}[\texttt{n2.succ}].\texttt{y}) \geq \min(\texttt{n1.y}, \texttt{NODES}[\texttt{n1.succ}].\texttt{y})$
- $(\texttt{n2.x} - \texttt{NODES}[\texttt{n1.succ}].\texttt{x}) * (\texttt{NODES}[\texttt{n1.succ}].\texttt{y} - \texttt{n1.y}) - (\texttt{NODES}[\texttt{n1.succ}].\texttt{x} - \texttt{n1.x}) * (\texttt{n2.y} - \texttt{NODES}[\texttt{n1.succ}].\texttt{y}) \neq 0$
- $(\texttt{NODES}[\texttt{n2.succ}].\texttt{x} - \texttt{NODES}[\texttt{n1.succ}].\texttt{x}) * (\texttt{n2.y} - \texttt{n1.y}) - (\texttt{n2.x} - \texttt{n1.x}) * (\texttt{NODES}[\texttt{n2.succ}].\texttt{y} - \texttt{NODES}[\texttt{n1.succ}].\texttt{y}) \neq 0$
- $\texttt{sign}\left(\prod\left(\begin{array}{l}\texttt{n2.x} - \texttt{NODES}[\texttt{n1.succ}].\texttt{x},\\ \texttt{NODES}[\texttt{n1.succ}].\texttt{y} - \texttt{n1.y}\end{array}\right) - \prod\left(\begin{array}{l}\texttt{NODES}[\texttt{n1.succ}].\texttt{x} - \texttt{n1.x},\\ \texttt{n2.y} - \texttt{NODES}[\texttt{n1.succ}].\texttt{y}\end{array}\right)\right) \neq \texttt{sign}\left(\prod\left(\begin{array}{l}\texttt{NODES}[\texttt{n2.succ}].\texttt{x} - \texttt{NODES}[\texttt{n1.succ}].\texttt{x},\\ \texttt{n2.y} - \texttt{n1.y}\end{array}\right) - \prod\left(\begin{array}{l}\texttt{n2.x} - \texttt{n1.x},\\ \texttt{NODES}[\texttt{n2.succ}].\texttt{y} - \texttt{NODES}[\texttt{n1.succ}].\texttt{y}\end{array}\right)\right)$

**Graph property(ies)** NARC= NCROSS

**Graph model** Each node is described by its coordinates x and y, and by its successor succ in the final covering. Note that the co-ordinates are initially fixed. We use the arc generator $CLIQUE(<)$ in order to avoid counting twice the same line segment crossing.

Parts (A) and (B) of Figure 5.367 respectively show the initial and final graph associated with the **Example** slot. Since we use the **NARC** graph property, the arcs of the final graph are stressed in bold. Each arc of the final graph corresponds to a proper intersection between two line segments.

**(A)**




**(B)**

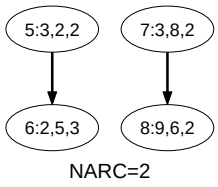



Figure 5.367: Initial and final graph of the `graph_crossing` constraint