Speeding up constrained path solvers with a reachability propagator

Student: **Luis Quesada** Supervisors: **Peter Van Roy and Yves Deville**

Université catholique de Louvain
Place Sainte Barbe, 2, B-1348 Louvain-la-Neuve, Belgium
{luque, pvr, yde}@info.ucl.ac.be

We present a propagator which we call *Reachability* that implements a generalized reachability constraint on a directed graph g. Given a source node $source$ in g, we can identify three parts in the *Reachability* constraint: (1) the relation between each node of g and the set of nodes that it reaches, (2) the association of each pair of nodes $\langle source, i\rangle$ with its set of cut nodes, and (3) the association of each pair of nodes $\langle source, i\rangle$ with its set of bridges.

Formally, this constraint can be defined as follows:

$$Reachability(g, source, rn, cn, be) \equiv \forall_{i \in N}. \begin{array}{l} rn(i) = Reach(g, i) \wedge \\ cn(i) = CutNodes(g, source, i) \wedge \\ be(i) = Bridges(g, source, i) \end{array} \tag{1}$$

where g is a graph whose set of nodes is a subset of N, $source$ is a node of g, $rn(i)$ is the set of nodes that i reaches in g (defined by $Reach(g, i)$), $cn(i)$ is the set of nodes appearing in all paths from source to i in g (defined by $CutNodes(g, source, i)$), and $be(i)$ is the set of edges appearing in all paths from $source$ to i in g (defined by $Bridges(g, source, i)$) [1].

Reachability has been implemented using a message passing approach on top of the multi-paradigm programming language Oz [Moz04]. The pruning rules of *Reachability* have been defined using the notion of graph variable [DDD05]. In [QVD05a,QVD05b], we discuss the implementation of *Reachability* in detail and its suitability for finding simple paths with mandatory nodes in directed graphs [2].

References

[DDD05] G. Dooms, Y. Deville, and P. Dupont. CP(Graph):introducing a graph computation domain in constraint programming. In *CP2005 Proceedings*, 2005.

[Moz04] Mozart Consortium. The Mozart Programming System, version 1.3.0, 2004. Available at *http://www.mozart-oz.org/*.

[QVD05a] Luis Quesada, Peter Van Roy, and Yves Deville. Reachability: a constrained path propagator implemented as a multi-agent system. In *CLEI2005 Proceedings*, 2005.

[QVD05b] Luis Quesada, Peter Van Roy, and Yves Deville. The reachability propagator. Research Report INFO-2005-07, Université catholique de Louvain, Louvain-la-Neuve, Belgium, 2005. Available at *http://www.info.ucl.ac.be/~luque/SPMN/paper.pdf*.

[1] Any node in N is a cut node between i and j if there is no path going from i to j. Similarly, any edge in $N \times N$ is a bridge between i and j if there is no path going from i to j.

[2] The problem of finding a simple path containing a set of mandatory nodes is not trivially reducible to Hamiltonian path.