Towards Using Description Logics for Symbolic Shape Analysis

Lilia Georgieva[1], Patrick Maier[2]

[1] School of Math. & Comp. Sci., Heriot-Watt Univ., Edinburgh, UK, lilia@macs.hw.ac.uk
[2] Max-Planck-Institut für Informatik, Saarbrücken, Germany, maier@mpi-sb.mpg.de

Abstract

We investigate description logics as a framework for symbolic shape analysis. We propose a predicate abstraction based shape analysis, parameterized by a DL to represent the abstraction predicates. Depending on the chosen logic, sharing, reachability and separation in pointer data structures are expressible.

Our work follows the trend in symbolic shape analysis by encoding properties of pointer programs in logical formulae [1, 2, 3].

Description logics with functional atomic roles (for modeling pointer fields), nominals (for modeling program variables) and fixed points are natural and expressive languages for specifying properties of pointer programs, e. g., x points to a doubly linked list, y points to a binary tree (i. e., a DAG without sharing), or the heap cells reachable from x resp. y are separated. We propose a predicate abstraction based shape analysis, parameterized by a DL to represent the abstraction predicates, which are concept expressions. The analysis relies on DL reasoners for checking concept subsumption w. r. t. nonempty TBoxes in finite models.

Ideally, the reasoners should be complete w. r. t. the chosen DL, or even decide it. However, if there are no such reasoners (e. g., there is none handling $\mu\mathcal{ALCO}_f^{-1}$, the logic needed to express separation) the analysis allows to trade precision for complexity: subsumption queries may be approximated (e. g., by relaxing functionality restrictions) in less expressive but computationally more feasible DLs.

References

[1] A. Møller and M. Schwartzbach. The pointer assertion logic engine. In *Proc. PLDI*, pages 221–231. ACM Press, 2001.

[2] A. Podelski and T. Wies. Boolean heaps. In *Proc. SAS*, 2005. To appear.

[3] M. Sagiv, T. Reps, and R. Wilhelm. Parametric shape analysis via 3-valued logic. *TOPLAS*, 24:217–298, 2002.