DL Systems Comparison

Ian R. Horrocks
Department of Computer Science
University of Manchester
Manchester M13 9PL, UK
horrocks@cs.man.ac.uk

Peter F. Patel-Schneider
Bell Labs Research
Murray Hill, NJ, U.S.A.
pfps@research.bell-labs.com

1 Introduction

The tradeoff between expressiveness and tractability in Description Logics (DLs) has long been recognised [9]. For DL system designers and implementors there are several possible approaches to this problem [3]. These include:

- (severely) constraining expressiveness so that (Tbox) reasoning can be performed in polynomial time (e.g., Classic [11]);
- using tractable but incomplete reasoning algorithms (e.g., Loom [10]);
- using sound and complete reasoning algorithms for logics with high worst case complexities on the grounds that the pathological cases which give rise to the theoretical complexity results are unlikely to arise in practice (e.g., Kris [1]).

In all cases there is a need for empirical evaluation in order to determine how the systems perform with respect to realistic Knowledge Bases (KBs) [6].

Such evaluation is even more important for the new generation of DL systems which use highly optimised tableaux algorithms to provide sound and complete reasoning for very expressive description languages [5, 8, 12]. Empirical evaluation is essential to the development of these systems in order to test the effectiveness of various optimisation techniques.

For all DL systems the evaluation process would be facilitated by the existence of a *benchmark suite*: a range of test problems which could be used to measure a system's performance and compare it with that of other systems.

2 Objectives

DL'98 is hosting a DL systems comparison session in which systems are compared on the basis of a wide range of benchmarks. This first round of the comparison will serve mostly as a mechanism for the development and evaluation of a benchmark suite, and it is not intended to be the last word on a possible ranking.

The aims of the comparison are to:

- reach a consensus on a *corpus of benchmarks* for DL systems, which may become an agreed standard.
- bring people together with a common interest in implementation work on various DLs.
- give the implementors of reasoners for the considered logics the opportunity to demonstrate their systems and to make them better known.

3 Comparing DL Systems

A fair comparison is very difficult, since different computers are used, systems are implemented in different programming languages, and the expressiveness of the different logics varies considerably. We hope to overcome these problems to a certain extent by using different families of parameterised knowledge bases. The time to process these KBs is expected to be exponential in the parameter for most reasoners, and the test methodology is to determine the largest parameter value that a system can handle within a given time limit. Consequently, the results can only be slightly improved by using a faster computer or a better programming language.

Moreover, we believe that expressiveness and efficiency are not the only qualities of a reasoner—a given reasoner may be prefered if it is: very comfortable to use; very small and simple, and thus more secure; written is a standard language and thus easily portable; able to deal with many logics; proven to be correct and complete; efficient in the use of space during computation; tailored for specific applications; *et cetera*.

4 The Test Procedure

The benchmark suite currently consists of four kinds of test: concept satisfiability tests, artificial Tbox classification tests, realistic Tbox classification tests and synthetic Abox tests. The test data, along with more information on the test procedure, can be obtained at *ftp://mighp0.cs.man.ac.uk/pub/theses/horrocks/dl98/dl98-test.tar.gz.*

4.1 Concept Satisfiability Tests (Tableaux'98)

This group of tests measures the performance of the system when computing the coherence (satisfiability) of large concept expressions without reference to a Tbox. The ideas behind this group were borrowed from the Comparison of Theorem Provers for Modal Logics at Tableaux'98 [2] and the tests use test data developed by Alain Heuerding and Stefan Schwendimann [7].

The test consists of 9 classes of concept (e.g. k_branch), in both coherent and incoherent forms. For each class of concept, 21 examples of supposedly exponentially increasing difficulty are automatically generated from a basic pattern which incorporates features intended to make the concept's coherence hard to compute.

The test methodology is to ascertain the number of the largest concept of each type whose coherence the system is able to compute within 100 seconds of CPU time. For example, if the coherence of the first concept is computed in 10s, that of the second in 50s and that of the third in 120s, then the result of the test is 2. If the system is able to compute the coherence of the largest concept in less than 100s then the result is 21. The correctness of the system is also tested by checking that the answers are as expected.

4.2 Artificial Tbox Classification Tests

These three groups of tests measure the performance of the system when classifying an artificially generated Tbox.

The first group uses Tboxes generated from the large concept expressions described in section 4.1 by recursively naming all sub-concepts. The test methodology is similar to that for the large concept expressions with the result being the number of the largest Tbox which the system was able to classify within 500 seconds of CPU time. The correctness of the system is tested by checking the coherence of a special test concept which relies on all the other concepts defined in the Tbox.

The other two groups of tests use slightly modified versions of synthetic and randomly generated Tboxes developed at DFKI during an earlier comparison of DL systems [6]. The result of these tests is the CPU time required to classify the Tbox, with an upper limit of 1,000 seconds. If the time limit is exceeded, the result is shown as >1000. For these tests a more thorough correctness test is performed by checking that the concept hierarchy computed by the system corresponds to a reference hierarchy.

4.3 Realistic Tbox Classification Tests

This group of tests measures the performance of the system when classifying a realistic Tbox. The tests use 2 KBs derived from the Galen medical terminology KB [8], and six other KBs from various sources that were used in the DFKI testing. The result of these tests is the CPU time required to classify the Tbox, with an upper limit of 1,000 seconds, and correctness is again checked by comparing the computed and reference hierarchies.

Some of the KBs in this group include constructs that cannot be handled by most current systems. Variants that ignore role definitions and role domains and ranges have been constructed for the KBs that include such constructs. The variants have `-role` appended to the KB name. Other variants for specific systems have also been constructed.

4.4 Synthetic Abox Tests

These tests measure the performance of the system's Abox when realising a synthetic Abox (inferring the most specific concept in the Tbox which each individual instantiates). They were derived from the Tableaux'98 Tbox classification tests, and each of the 9 tests consists of a Tbox, an Abox and a set of Abox queries (it is intended that this will be extended to multiple tests of increasing size, as for the Tbox classification tests, but it has proved difficult to generate larger tests due to the poor performance of available systems). The result of the tests is the CPU time required to realise the Abox, with an upper limit of 1,000 seconds. Correctness is tested by the queries, which should all evaluate to true.

5 Results

The comparison was open to any DL which accepts a reasonable subset of the language specified by the KRSS Description Logic Specification document [13], and all the test data in the benchmark suite used the KRSS syntax. Submissions were received with respect to 6 DL systems: Crack, DLP, FaCT, HAM-ALC, Kris and NeoClassic. Of these, all but NeoClassic implement various supersets of the $\mathcal{ALC}$ description language [14]. No verification of any of the results has been attempted.

Only the Kris system was able to perform all of the benchmark tests. Other systems were unable to perform some of the tests due to a lack of language constructs or of Abox reasoners. This should not be taken to mean that Kris's description language is a superset of all the others in the test: many of the other systems support additional constructs, such as transitive roles, which do not currently occur in the benchmark suite. In several cases Tbox tests were performed by either ignoring or modifying unsupported constructs (notably number restrictions). In these cases, or in the case where a test ran out of time, the correctness is shown as "?".

6 Conclusion

As stated in Section 2, it was never the intention of the comparison to produce a ranking of the participating systems, and in any case the heterogeneity of the systems makes this impossible.

The comparison has succeeded in its primary objective: a provisional benchmark suite has been established and has been made available as a resource to the developers of DL systems. However this is only a beginning, and much more data is required in order to extend the suite. In particular there is an urgent need for more realistic KBs, more expressive KBs and more Aboxes of any sort.

References

[1] F. Baader and B. Hollunder. KRIS: Knowledge representation and inference system. *SIGART Bulletin*, 2(3):8–14, 1991.

[2] P. Balsiger and A. Heuerding. Comparison of theorem provers for modal logics — introduction and summary. In de Swart [4], pages 25–26.

[3] A. Borgida. Description logics are not just for the flightless-birds: A new look at the utility and foundations of description logics. Technical Report DCS-TR-295, New Brunswick Department of Computer Science, Rutgers University, 1992.

[4] H. de Swart, editor. *Automated Reasoning with Analytic Tableaux and Related Methods: International Conference Tableaux'98*, number 1397 in Lecture Notes in Artificial Intelligence. Springer-Verlag, May 1998.

[5] V. Haarslev, R. Möller, and A.-Y. Turhan. Implementing an $\mathcal{ALCRP}(\mathcal{D})$ abox reasoner: Progress report. In E. Franconi, G. De Giacomo, R. M. MacGregor, W. Nutt, C. A. Welty, and F. Sebastiani, editors, *Collected Papers from the International Description Logics Workshop (DL'98)*, 1998. To appear.

[6] J. Heinsohn, D. Kudenko, B. Nebel, and H.-J. Profitlich. An empirical analysis of terminological representation systems. *Artificial Intelligence*, 68:367–397, 1994.

[7] A. Heuerding and S. Schwendimann. A benchmark method for the propositional modal logics k, kt, s4. Technical report IAM-96-015, University of Bern, Switzerland, October 1996.

[8] I. Horrocks. The FaCT system. In de Swart [4], pages 307–312.

[9] H. J. Levesque and R. J. Brachman. Expressiveness and tractability in knowledge representation and reasoning. *Computational Intelligence*, 3(2):78–93, 1987.

[10] R. M. MacGregor. Inside the LOOM description classifier. *SIGART Bulletin*, 2(3):88–92, 1991.

[11] P. F. Patel-Schneider. The CLASSIC knowledge representation system: Guiding principles and implementation rationale. *SIGART Bulletin*, 2(3):108–113, 1991.

[12] P. F. Patel-Schneider. System description: DLP. Bell Labs Research, Murray Hill, NJ, December 1997.

[13] P. F. Patel-Schneider and B. Swartout. Description logic specification from the krss effort, June 1993.

[14] M. Schmidt-Schauß and G. Smolka. Attributive concept descriptions with complements. *Artificial Intelligence*, 48:1–26, 1991.