Lessons Learned from Benchmarking in the Automated Planning Community

Malte Helmert[1]

1 INTRODUCTION

Automated planning is a classical area of artificial intelligence research that is concerned with the problem of planning a course of action for an agent (or set of agents) acting in a complex environment. In the frequently studied case of *classical planning*, e. g. in the propositional STRIPS formalism [1], the problem is that of finding a sequence of actions that achieves the goal of a single agent in a deterministic, static, fully observable world, given only a logical description of the initial state, goal, and preconditions and effects of actions. Automated planning is an active research area, represented at general AI conferences like IJCAI, AAAI and ECAI as well as in the annual ICAPS (*International Conference on Automated Planning and Scheduling*) conference series.

2 BENCHMARKING IN PLANNING

Up to the late nineties, most classical planning systems used their own, slightly or largely incompatible problem representations, which made it hard to compare them. This changed with the introduction of the PDDL modeling language [2], originally designed for the first International Planning Competition (IPC), which was organized by Drew McDermott as part of the AIPS 1998 conference. Between 1998 and 2008, the IPC has been repeated as a regular biennial event, which has helped establish PDDL as the problem representation used by essentially all classical planners today. Moreover, the benchmark problems used for the evaluation of planners at the IPC (which are growing with every edition of the competition, since an important aspect of the competition is to challenge planners on new, previously unseen problems) have become the "canonical" set of benchmarks for empirical evaluations of planning system. For example, all 46 papers with empirical evaluations of classical planners published in the ICAPS conference series in the period 1999–2005 use IPC benchmarks at least as part of their evaluation, and since 2003 the overwhelming majority uses IPC benchmarks *exclusively*.

3 LESSONS LEARNED

Few planning researchers would deny that the rise of a common input representation and benchmark set for classical planning has had a profound impact on the research community. There are both good and bad aspects to this, and no two researchers would completely agree on their assessment of these pros and cons and on what conclusions should be drawn from the experience. Hence, the following list of "lessons learned" should be regarded as highly subjective.

- *Common benchmarks push the state of the art.* The importance of benchmarks that allow directly comparing the approaches of different research teams cannot be overstated. Healthy competition can greatly accelerate the development of efficient algorithms.
- *Clear and common metrics are important.* In many cases, it is hard to decide which of two approaches performs better on a given problem since there are different, incomparable aspects to consider. Nevertheless, it is important to define clear, quantifiable metrics that make an unequivocal judgment. Trade-offs and design decisions in the development of such metrics should be highlighted; different metrics are needed for different purposes. Still, it is necessary to commit to a *small* set of core metrics, or else clear conclusions cannot be drawn.
- *Algorithm evaluators should not be algorithm designers.* Ideally, the evaluation of a system or algorithm should be performed by someone who has no stake in it, since it is often all too easy to come up with an experiment that makes a bad approach look good. A ubiquitous set of benchmarks with a commonly accepted evaluation methodology is one way to achieve this goal.
- *Overfitting is an issue, as is having a moving target.* No evaluation metric is perfect, and a research communities must continuously verify that it is still measuring the right thing. Algorithm designers quickly adapt their methods to optimize precisely what is measured, whether or not this captures the underlying research goal. At the same time, there are strong benefits to keeping evaluation metrics stable in order to reliably track progress over time.
- *Competitions are not scientific experiments.* Good engineering of poor ideas can often outperform bad engineering of good ideas. Understanding *why* an approach performs well is at least as important as knowing *that* it performs well. Benchmarking has its place within proper scientific methodology, but it can only be part of the story.
- *Maintaining benchmarks is work that must be incentivized.* Designing and maintaining a useful set of benchmarks is a significant amount of work that is often hard to justify because it does not produce the kind of scientific output (conference papers, journal articles, etc.) on which academics are evaluated. Research communities must find ways to actively encourage working on infrastructure for benchmarking and evaluation.

REFERENCES

[1] Tom Bylander, 'The computational complexity of propositional STRIPS planning', *AIJ*, **69**(1–2), 165–204, (1994).

[2] Drew McDermott, Malik Ghallab, Adele Howe, Craig Knoblock, Ashwin Ram, Manuela Veloso, Daniel Weld, and David Wilkins, 'PDDL – The Planning Domain Definition Language – Version 1.2', Technical Report CVC TR-98-003, Yale Center for Computational Vision and Control, (1998).

[1] Albert-Ludwigs-Universität Freiburg, Germany, helmert@informatik.uni-freiburg.de