

From CATS to SAT: Modeling Empirical Hardness to Understand and Solve Hard Computational Problems

*Kevin Leyton-Brown
University of British Columbia*

From combinatorial auctions to supply chains and beyond, researchers in multiagent resource allocation frequently find themselves confronted with hard computational problems. This tutorial will focus on empirical hardness models, a machine learning methodology that can be used to predict how long an algorithm will take to solve a problem before it is run.

My coauthors and I first developed this line of research in our work on the Combinatorial Auction Test Suite (CATS), when investigating whether "realistic" combinatorial auction problems were always computationally easier than the hardest artificial distributions.

We did eventually figure out how to tune our instance generators to create harder instances. Along the way, however, we also developed a host of other methods that I will survey in this tutorial. These included ways of accurately predicting an algorithm's runtime on an unseen instance, determining which instance properties most affect an algorithm's performance, and building algorithm portfolios that can dramatically outperform their constituent algorithms.

After satisfying ourselves that empirical hardness models are a useful way of tackling combinatorial auction problems, we sought to demonstrate their effectiveness on a more widely-studied NP-complete problem, and hence turned to SAT. I will also describe some of the techniques we developed for this second problem domain, including the direct prediction of satisfiability status, the construction of hierarchical models, and the inclusion of incomplete local search algorithms.

I will conclude by describing SATzilla, an algorithm portfolio constructed from 19 state-of-the-art complete and incomplete SAT solvers, which won 5 medals at the 2007 SAT competition.