The borg Algorithm Portfolio (ver. PB-11.04.03)

Bryan Silverthorn and **Risto Miikkulainen**
Department of Computer Science
The University of Texas at Austin
{bsilvert,risto}@cs.utexas.edu

Summary of Approach

The borg solver is an *algorithm portfolio* (Huberman, Lukose, and Hogg 1997). It uses empirical information about solvers and problem instances to allocate computational resources among multiple subsolvers, attempting to maximize the time spent on those well suited to each instance. Portfolio methods such as SATzilla (Xu et al. 2008) have proved increasingly effective in satisfiability; an earlier version of borg was successful at the 2010 pseudo-Boolean solver competition.

An algorithm portfolio must decide which solvers to run and for how long to run them. These decisions rely entirely on expectations about solver behavior. The borg solver attempts to to learn predictable aspects of solver behavior given data on the successes and failures of solvers on many problem instances. This version, borg-pb-11.04.03, builds on the work of Silverthorn and Miikkulainen (2010) toward generative models of solver behavior. It assumes latent class structure in solver runs (i.e., that solver runs can be grouped by similarity according to their probability of success versus time) and in problem instances (i.e., that problem instances can be grouped by similarity according to the run classes that solvers exhibit on them). It uses standard methods (EM) to learn the parameters of its model, then trains a logistic regression classifier to predict the probabilities of class membership. The features used are statistics of the constraint graph and clause coefficients.

Solver scheduling decisions are made by computing an optimal discrete schedule over the weighted matrices of solver sucess probabilities predicted by the model, conditioned on the classifier's output and on the sequence of past executions. The portfolio chooses to pause, resume, or start a solver run at a series of discrete decision points.

Portfolio Composition

Portfolio methods rely entirely on the performance of the solvers they employ, and are possible only because of the extensive engineering and research involved in making those solvers effective. This version of borg-pb considered 11 subsolvers, including various parameterizations. Table 1 lists these solvers and their primary authors.

Software

The source code, data sets, and toolchain used in borg are publicly available at http://www.cs.utexas.edu/~bsilvert/. Anyone with an interest in experimenting with a portfolio method in their domain is encouraged to contact the authors.

Acknowledgments

This research was supported in part by the NSF under grants EIA-0303609, IIS-0757479, and IIS-0915038, and by the THECB under grant 003658-0036-2007.

References

Huberman, B.; Lukose, R.; and Hogg, T. 1997. Economics Approach to Hard Computational Problems. *Science*.

Silverthorn, B., and Miikkulainen, R. 2010. Latent Class Models for Algorithm Portfolio Methods. In *AAAI*.

Xu, L.; Hutter, F.; Hoos, H. H.; and Leyton-Brown, K. 2008. SATzilla: Portfolio-based Algorithm Selection for SAT. *JAIR*.

Name	Author(s)
pbct-0.1.2-linear	Anders Franzén and Roberto Bruttomesso
sat4j-pb-v20101225	Daniel Le Berre and Anne Parrain
sat4j-pb-v20101225-cutting	Daniel Le Berre and Anne Parrain
bsolo_pb10-l1	Vasco Manquinho and José Santos
bsolo_pb10-l2	Vasco Manquinho and José Santos
bsolo_pb10-l3	Vasco Manquinho and José Santos
wbo1.4a	Vasco Manquinho, Joao-Marques Silva, and Jordi Planes
wbo1.4b-fixed	Vasco Manquinho, Joao-Marques Silva, and Jordi Planes
clasp-1.3.7	Martin Gebser, Benjamin Kaufmann, André Neumann, and Torsten Schaub
scip-2.0.1-clp	Stefan Heinz and Michael Winkler
scip-2.0.1-spx	Stefan Heinz and Michael Winkler

Table 1: Subsolvers used by borg-pb-11.04.03.