Probabilistically Estimating Backbones and Variable Bias: Experimental Overview ⋆

Eric I. Hsu, Christian J. Muise, J. Christopher Beck, and Sheila A. McIlraith

Department of Computer Science, University of Toronto
{eihsu,cjmuise,sheila}@cs.toronto.edu, jcb@mie.utoronto.ca

Abstract. Backbone variables have the same assignment in all solutions to a given constraint satisfaction problem; more generally, *bias* represents the proportion of solutions that assign a variable a particular value. Intuitively such constructs would seem important to efficient search, but their study to date has been from a mostly conceptual perspective, in terms of indicating problem hardness or motivating and interpreting heuristics. Here we summarize a two-phase project where we first measure the ability of both existing and novel probabilistic message-passing techniques to directly estimate bias and identify backbones for the Boolean Satisfiability (SAT) Problem. We confirm that methods like Belief Propagation and Survey Propagation–plus Expectation Maximization-based variants–do produce good estimates with distinctive properties. The second phase demonstrates the use of bias estimation within a modern SAT solver, exhibiting a correlation between accurate, stable, estimates and successful backtracking search. The same process also yields a family of search heuristics that can dramatically improve search efficiency for the hard random problems considered.

1 Introduction

Probabilistic message-passing algorithms like Survey Propagation (SP) and Belief Propagation (BP), plus variants based on Expectation Maximization (EM), have proved very successful for random SAT and CSP problems [2–4]. This success would appear to result from the ability to implicitly sample from the space of solutions and thus estimate variable bias: the percentages of solutions that have a given variable set true or false. However, such bias estimation ability has never been measured directly, and its actual usefulness to heuristic search has also escaped systematic study. Similarly, backbones, or variables that must be set a certain way in any solution to a given problem, have also drawn a good deal of recent interest [5–9], but they have not been directly targeted for discovery within arbitrary problems. Since backbones must have 100% positive or negative bias, bias determination generalizes the task of backbone identification.

Isolating the performance of probabilistic techniques when applied to bias estimation improves our understanding of both the estimators and of bias itself, ultimately directing the design of a complete problem-solving system. Thus the first stage of our study compares the basic accuracy of six message-passing techniques and two control methods when applied to hard, random, satisfiable SAT problems as stand-alone bias estimators. The second stage assesses how such comparisons translate when we move the

⋆ This article summarizes a more detailed description that is available as a technical report [1].

algorithms into the realm of full-featured search, by embedding them as variable/value ordering heuristics within the MiniSat solver [10]. While it is intuitive that bias should relate to how we set variables during search, it is not obvious that bias should be a key to efficiency in the presence of modern features like restarts and clause learning.

2 Definitions

Definition 1 (SAT instance) *A (CNF)* **SAT instance** *is a set C of m clauses, constraining a set V of n Boolean variables. Each clause $c \in C$ is a disjunction of literals built from the variables in V. An assignment $X \in \{0,1\}^n$ to the variables satisfies the instance if it makes at least one literal true in each clause.*

Definition 2 (Bias, Survey) *For a satisfiable SAT instance $\mathcal{F}$, the* **estimated bias distribution** *θ_v of a variable v attempts to represent the fraction of solutions to $\mathcal{F}$ wherein v appears positively or negatively. Thus it consists of a* **positive bias** *θ_v^+ and a* **negative bias** *θ_v^-, where $\theta_v^+, \theta_v^- \in [0,1]$ and $\theta_v^+ + \theta_v^- = 1$. A vector of bias distributions, one for each variable in a theory, will be called a* **survey***, denoted $\Theta(\mathcal{F})$.*

Equivalently, it can be useful to think of a variable's bias as the probability of finding the variable set positively or negatively when randomly sampling from the space of satisfying assignments. Less formally, the "**strength**" of a bias distribution indicates the margin by which it favors one value over the other.

3 Probabilistic Methods for Estimating Bias

We compare six distinct message-passing techniques for measuring variable bias: Belief Propagation (BP), EM Belief Propagation-Local/Global (EMBP-L and EMBP-G), Survey Propagation (SP), and EM Survey Propagation-Local/Global (EMSP-L and EMSP-G). On receiving a SAT instance $\mathcal{F}$, each of the propagation methods begins by formulating an initial survey at random. Each algorithm proceeds to successively refine its estimates over multiple iterations. An iteration consists of a single pass through all variables, where the bias for each variable is updated with respect to the other variables' biases, according to the characteristic rule for a method. If no variable's bias has changed between two successive iterations, the process ends with convergence; otherwise a method terminates by timeout or some other parameter. EM-type methods are "convergent", or guaranteed to converge naturally, while regular BP and SP are not [4].

BP estimates bias via Pearl's Belief Propagation, also known as the Sum-Product algorithm [11, 12]. **SP** (Survey Propagation) extends BP to measure not only the probabilities of a variable being positively or negatively *constrained* in a solution, but also the probability that it could have been set either way [2]. Such extra sensitivity changes the dynamics between iterations, but in the final survey any mass for the third "joker state" is evenly divided between the positive and the negative for the purposes of estimating bias. Both methods can be re-formulated within the Expectation Maximization framework [13], producing the four EM-based methods. **EMBP-L** and **EMBP-G** use the two-state model of BP and differ in exploiting variational approximations [14] based on local (generalized arc-) consistency and global consistency, respectively. Similarly, **EMSP-L** and **EMSP-G** apply local or global consistency to the three-state model represented by SP. Global methods embody a tighter bound on the variational approximation, but in general take longer to compute than local methods. For experimental comparison, we created two non-probabilistic control methods. **LC** ("Literal Count") greatly simplifies the backbone-inspired heuristic at the core of a highly successful system for refuting *unsatisfiable* SAT instances [8]. **CC** ("Clause Count") is an even simpler baseline method that just counts the number of clauses containing a given variable as a positive literal, and the number wherein it appears negatively. The relative weights of these two counts determine the variable's estimated bias distribution.

4 Summary of Experiments

The first phase of experiments compared the eight methods as stand-alone bias estimators for randomly generated problems whose true biases were found via exhaustive model counting. Figure 1 depicts basic root-mean-squared error, with the global EM-based methods performing the best, along with the controls and then regular BP. Local EM-based methods and regular SP comprise a second band of less accurate methods, providing bias estimates that were off by .36 up to .48, depending on the number of backbone variables in a problem.

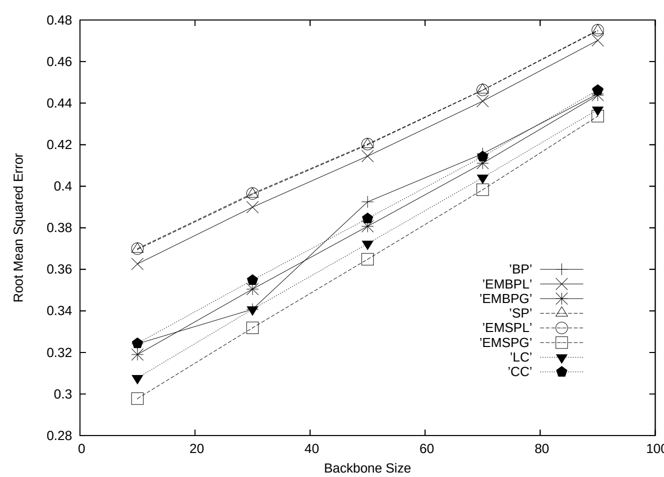

Fig. 1. RMS error over 500 instances of increasing backbone size, $n = 100$.

However, a second pattern arises across experiments emphasizing strong biases and estimates: "backbone identification rate" and "rank of first wrong bias" (see longer presentation [1]). The former measures the proportion of actual backbone variables that an estimator biases toward the correct polarity. The latter examines the variables to which a method assigns the strongest biases, and finds the highest-ranking estimate that was toward the wrong polarity. For both of these measures, BP actually performs the best, followed by EMSP-G, LC, SP, EMBP-G, the local EM methods, and CC.

These two sets of measures (basic accuracy and accuracy on strong biases) turned out to be the most important predictors of success in the second phase of study. Here, the methods were embedded as variable-ordering heuristics within the MiniSat solver [10]. Part of this process meant determining a good way to use surveys: based on our experimental experiences, the results reflect the intuitive "succeed-first" approach of setting the most strongly-biased variable in a survey in the direction of its stronger bias.

The overall runtime results in Figure 2 exhibit a correlation between good bias estimates and significant improvement in search efficiency. The graph consists average runtimes broken down to show the proportion that was devoted to computing surveys.

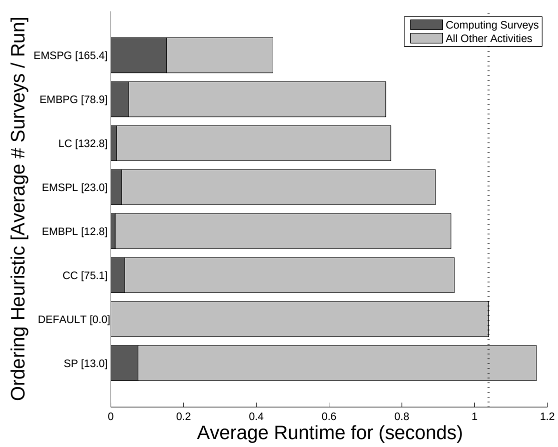

Fig. 2. Total/Survey runtimes averaged over 100 random problems, $n = 250$ and $\alpha = 4.11$.

The "DEFAULT" method represents regular MiniSat without bias estimation, and BP performed so badly as to not appear on the graph. When scaling to larger experiments, the relative efficiency of the best method (EMSP-G) exhibits exponential growth. For instance, when $n = 450$, it requires an average of 5 minutes per problem, while default MiniSat typically requires 30, or times out after three hours.

Of the many measures examined during the first phase of experiments, basic accuracy seems to be the most important predictor of success as a heuristic: global EM methods and controls outperform local EM methods and SP/BP, and within each of these bands the SP version is a more effective heuristic than the BP version. Exceptions to this correlation indicate the importance of two additional factors: the secondary strength-oriented accuracy measures mentioned above, and variance in accuracy. For instance, CC compares relatively well in basic accuracy, but ranks lower as a heuristic; it scored the worst with the accuracy measures that focus on strong biases. And while BP scores the best on such secondary measures and does relatively well on basic accuracy, it yields the worst heuristic. This may be due to *variance* in basic accuracy–of the various methods BP exhibited the wildest fluctuations in error by far.

5 Conclusions

The main findings of these experiments indicate that probabilistic message-passing techniques can be comparatively successful at estimating variable bias and identifying backbone variables, and that successful bias estimation has a positive effect on heuristic search efficiency within a modern solver. Secondary contributions include a novel family of EM-based bias estimators, and a series of design insights culminating in a fast solver for hard random problems.

However, many important issues remain. For instance, structured and unsatisfiable instances have not yet been considered by this bias estimation framework. This may require a finer-grained analysis of accuracy that considers variance across multiple runs with various random seeds. Further, the best way to use surveys for variable ordering cannot be settled conclusively by the limited span of branching strategies that have been studied to date. For instance, we waste some of the SP framework's power when we split the probability mass for the joker state between positive and negative bias; future branching strategies might favor variables with low joker probabilities.

References

1. Hsu, E., Muise, C., Beck, J.C., McIlraith, S.: Applying probabilistic inference to heuristic search by estimating variable bias. Technical Report CSRG-577, University of Toronto (2008)
2. Braunstein, A., Mezard, M., Zecchina, R.: Survey propagation: An algorithm for satisfiability. Random Structures and Algorithms **27** (2005) 201–226
3. Dechter, R., Kask, K., Mateescu, R.: Iterative join-graph propagation. In: Proc. of 18th Int'l Conference on Uncertainty in A.I. (UAI '02), Edmonton, Canada. (2002) 128–136
4. Hsu, E., Kitching, M., Bacchus, F., McIlraith, S.: Using EM to find likely assignments for solving CSP's. In: Proc. of 22nd National Conference on Artificial Intelligence (AAAI '07), Vancouver, Canada. (2007)
5. Monasson, R., Zecchina, R., Kirkpatrick, S., Selman, B., Troyansky, L.: Determining computational complexity from characteristic phase transitions. Nature **400**(7) (1999) 133–137
6. Kilby, P., Slaney, J., Thiébaux, S., Walsh, T.: Backbones and backdoors in satisfiability. In: Proc. of 20th National Conference on A.I. (AAAI '05), Pittsburgh, PA. (2005)
7. Singer, J., Gent, I., Smaill, A.: Backbone fragility and the local search cost peak. Journal of Artificial Intelligence Research **12** (2000) 235–270
8. Dubois, O., Dequen, G.: A backbone-search heuristic for efficient solving of hard 3-SAT formulae. In: Proc. of 17th International Joint Conference on Artificial Intelligence (IJCAI '01), Seattle, WA. (2001)
9. Zhang, W.: Configuration landscape analysis and backbone guided local search. Part I: Satisfiability and maximum satisfiability. Artificial Intelligence **158**(1) (2004) 1–26
10. Eén, N., Sörensson, N.: An extensible SAT-solver. In: Proc. of 6th International Conference on Theory and Applications of Satisfiability Testing (SAT '03), Portofino, Italy. (2003)
11. Pearl, J.: Probabilistic Reasoning in Intelligent Systems. Morgan Kaufmann, 1988.
12. Kschischang, F.R., Frey, B.J., Loeliger, H.A.: Factor graphs and the sum-product algorithm. IEEE Transactions on Information Theory **47**(2) (2001)
13. Dempster, A., Laird, N., Rubin, D.: Maximum likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistical Society **39**(1) (1977) 1–39
14. Jordan, M., Ghahramani, Z., Jaakkola, T., Saul, L.: An introduction to variational methods for graphical models. In Jordan, M., ed.: Learning in Graphical Models. MIT Press (1998)