

Local Search With Maximal Independent Sets

Joel M. Gompert

Constraint Systems Laboratory
University of Nebraska-Lincoln
Email: jgompert@cse.unl.edu

When solving Constraint Satisfaction Problems (CSPs), it is desirable to find multiple solutions, or to find solutions that are robust, allowing us to modify the values of variables without breaking the solution. Furthermore, we would like to be able to represent these multiple solutions in a compact manner. We present a method for improving the performance of, and solutions returned by, stochastic local search using *maximal independent sets*¹ of the constraint graph. Given an independent set, this information can be used to significantly speed up the process of solving a CSP by reducing the search space. The CSP is partitioned into two sets of variables I and $\bar{I}$, where I is a maximal independent set of the constraint graph. In this way, search is concentrated only on the variables of $\bar{I}$, reducing the search space by a factor exponential in the size of I . Also this technique can provide multiple solutions in a compact form with no extra cost, since if we find a set of valid domain values for each variable in I , every element of the Cartesian product of these sets is a solution to the CSP. We focus on exploiting this information in local search. This technique is limited to low-density graphs, since dense graphs are less likely to contain a large independent set. The resulting solutions are robust with respect to the variables in the independent set. We compare the technique with WalkSAT, as defined for CSPs by [2], on low-density random CSP instances generated according to model B, with 16 variables, domain size 8, tightness 0.3 and density 0.1. The average CPU run-time for WalkSAT was 96.5 seconds, while the average for WalkSAT-IS was 0.36 seconds. Also, while WalkSAT returns one solution, the average number of solutions returned by WalkSAT-IS per instance was 47,536,305 solutions. This is a dramatic improvement in performance, as well as in the number of solutions returned. This technique falls in the category of algorithms exploiting backdoor variables [3]. A related technique is the cycle-cutset technique [2]. Related work on SAT problems can be found in [4]. Future work will involve a wider range of experiments. Also we plan to perform a comparison with the cycle-cutset technique.

Acknowledgments. This research is supported by NSF CAREER Award #0133568.

References

1. Boppana, R., Halldórsson, M.M.: Approximating maximum independent sets by excluding subgraphs. In: 2nd Scandinavian Workshop on Algorithm Theory. Volume 447. (1990) 13–25
2. Dechter, R.: Constraint Processing. Morgan Kaufmann (2003)
3. Williams, R., Gomes, C.P., Selman, B.: Backdoors to typical case complexity. In: Artificial Intelligence. (2003)
4. Kautz, H., McAllester, D., Selman, B.: Exploiting variable dependency in local search. Poster session abstract, IJCAI (1997)

¹ We used the CliqueRemoval algorithm [1], which executes in linear-time.