Systematic and nonsystematic search

Matthew L. Ginsberg*
CIRL
University of Oregon
Eugene, Oregon 97403
ginsberg@cs.uoregon.edu

There seems to be a great deal of confusion here regarding the question of whether or not systematic search methods can be applied to problems of interesting size. Tsang, for example, says:

> Complete algorithms alone have little chance of solving realistic CSPs [constraint satisfaction problems]. For example, the problem of assigning 30 jobs to 10 machines (satisfying certain constraints) has a search space of 10^{30} nodes.

So what? Neither a systematic nor a nonsystematic approach needs to search the entire space in order to solve a satisfiable problem. Perhaps what Tsang is trying to say is simply that there is no point in trying to search enormous search spaces in their entirety. Assuming that $P \neq NP$, it's hard to argue with that.

The upshot of this observation seems to be that if a large problem is unsatisfiable, finding a proof of unsatisfiability is likely to be impossible for *any* approach. One can conclude from this that the principal claimed advantage of systematic methods, that they can eventually report a problem as unsolvable, is illusory.

Of course, that's simply not true. There are many large problems that are clearly unsatisfiable. Can United Airlines schedule all of its daily flights using only 17 aircraft? Surely not – and we don't need to examine the entire space of possible schedules in order to prove it. So we can probably say the following:

Observation 1 *Some – but not all – large unsatisfiable problems are unlikely to be solvable using systematic search methods. All unsatisfiable problems of any size are unsolvable using nonsystematic methods.*

Contrary to the nonsystematic community's propaganda, unsatisfiable problems are important, and can sometimes – not always – be solved using systematic methods.

The claimed advantage of the nonsystematic methods comes from their superior performance on *satisfiable* problems. There has been a great deal written and said about this [Langley, 1992, Minton *et al.*, 1990, Selman *et al.*, 1992], and I won't repeat it here. A reasonable summary appears to be that the symbolic approaches (min-conflicts, GSAT and so on) uniformly outperform nonsymbolic methods (simulated annealing, neural nets, etc.) and that:

Observation 2 *When solving a satisfiable problem, it is important to be able to follow local gradients.*

All of the claimed performance advantage of the stochastic or other nonsystematic methods appears to be based in their ability to move locally through the search space; systematic methods typically lack this property.[1] But nowhere is it suggested that nonsystematicity in and of itself is an attractive property; how could it be?

In spite of their inability to follow local gradients, systematic approaches are competitive with nonsystematic ones on a wide variety of problems; job-shop scheduling is a typical example [Harvey and Ginsberg, 1995, Smith and Cheng, 1993, Smith, 1992]. One explanation for the performance of the systematic approach is that some measure of systematicity is needed to escape the local minima that arise in realistic problems.

When solving a mix of problems including both satisfiable and unsatisfiable instances, it is clear that systematicity and the ability to follow local gradients are both important. Given the job-shop scheduling results, it appears that both can be important even if we restrict our attention to satisfiable problems alone.

One member of the panel (Dechter), reaching the same conclusion, suggests that the algorithms run in parallel or that we strive to "understand what problem features make a given algorithm run more effectively ... yielding a case-based integration." She suggests that algorithm development should perhaps be

*This work has been supported by the Air Force Office of Scientific Research under contract 92-0693 and by ARPA/Rome Labs under contracts F30602-91-C-0036 and F30602-93-C-00031.

[1] Dechter's suggestion in this panel that "Backtracking is a local search procedure that can be viewed as selecting the ... maximum gradient descent of some function" misses the point. We want to follow local gradients of a function that measures solution quality, not one that happens to mimic our particular search procedure.

regarded as less important than this integration, saying that, "The important problem in my opinion is ... how to exploit identified class-superior algorithms within one meta-framework."

This seems to me to miss the mark completely. Far more sensible would be for the constraint-satisfaction community to focus at least some of its effort on the development of new systematic techniques that follow local gradients. Both dynamic backtracking [Ginsberg, 1993] (misdescribed by Dechter as integrating "backjumping and possibly other ideas") and partial-order dynamic backtracking [Ginsberg and McAllester, 1994] are preliminary attempts in this direction. The work is much more difficult than simple and continued experimentation on ever-larger randomly generated problems, but seems far more likely to lead to results of practical importance.

References

[Ginsberg and McAllester, 1994] Matthew L. Ginsberg and David A. McAllester. GSAT and dynamic backtracking. In *Proceedings of the Fourth International Conference on Principles of Knowledge Representation and Reasoning*, Bonn, Germany, 1994.

[Ginsberg, 1993] Matthew L. Ginsberg. Dynamic backtracking. *Journal of Artificial Intelligence Research*, 1:25–46, 1993.

[Harvey and Ginsberg, 1995] William D. Harvey and Matthew L. Ginsberg. Limited discrepancy search. In *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence*, 1995. Submitted.

[Langley, 1992] Pat Langley. Systematic and nonsystematic search strategies. In *Artificial Intelligence Planning Systems: Proceedings of the First International Conference*, pages 145–152. Morgan Kaufmann, 1992.

[Minton *et al.*, 1990] Steven Minton, Mark D. Johnston, Andrew B. Philips, and Philip Laird. Solving large-scale constraint satisfaction and scheduling problems using a heuristic repair method. In *Proceedings of the Eighth National Conference on Artificial Intelligence*, pages 17–24, 1990.

[Selman *et al.*, 1992] Bart Selman, Hector Levesque, and David Mitchell. A new method for solving hard satisfiability problems. In *Proceedings of the Tenth National Conference on Artificial Intelligence*, pages 440–446, 1992.

[Smith and Cheng, 1993] Stephen F. Smith and Cheng-Chung Cheng. Slack-based heuristics for constraint satisfaction scheduling. In *Proceedings of the Eleventh National Conference on Artificial Intelligence*, pages 139–144, 1993.

[Smith, 1992] Douglas R. Smith. Transformational approach to scheduling. Technical Report KES.U.92.2, Kestrel Institute, 1992.