Annals of Mathematics and Artificial Intelligence 28 (2000) 35–38

Perspectives

Compute-intensive methods in artificial intelligence

Bart Selman

Department of Computer Science, Cornell University, Ithaca, NY 14853, USA
E-mail: selman@cs.cornell.edu

In order to deal with the inherent combinatorial nature of many tasks in artificial intelligence, domain-specific knowledge has been used to control search and reasoning or to eliminate the need for general inference altogether. However, the process of acquiring domain knowledge is an important bottleneck in the use of such “knowledge-intensive” methods. Compute-intensive methods, on the other hand, use extensive search and reasoning strategies to limit the need for detailed domain-specific knowledge. The idea is to derive much of the needed information from a relatively compact formalization of the domain under consideration. Up until recently, such general reasoning strategies were much too expensive for use in applications of interesting size but recent advances in reasoning and search methods have shown that compute-intensive methods provide a promising alternative to knowledge-intensive methods.

In the 70’s and 80’s the success of knowledge-intensive approaches to problem solving eclipsed earlier work on compute-intensive weak methods. However, in recent years, compute-intensive methods have made a surprising comeback. One of the most prominent examples is the success of IBM’s Deep Blue in defeating Gary Kasparov in the 1997 ACM Challenge match. Deep Blue derives its strength mainly from highly optimized search [7,14]. Another dramatic development in the compute-intensive approach was the recent computer proof resolving the Robbins algebra problem [12,13]. The Robbins problem is a well-known problem in Boolean algebra, and was open for over sixty years. The computer proof was found by applying powerful search techniques guided by general search tactics. Several aspects of the computer proof could be called “creative” by mathematicians’ standards. Deep Blue’s performance and the resolution of Robbin’s theorem are good examples of a *qualitative* change in performance of compute-intensive approaches compared to just a few years ago.

In my own work, I have focused on compute-intensive methods in a range of domains, such as planning and diagnosis. BlackBox [9,10,25] is a planner that challenges the widespread belief in the AI community that planning is not amenable to general theorem-proving techniques. BlackBox shows that general propositional satisfiability algorithms can outperform specialized planning systems on a number of key benchmark problems. At the core of BlackBox lies a powerful model finding procedure. This procedure was developed as part of work on stochastic model finding methods for propositional theories, such as GSAT and WalkSat [18,22,24]. Such stochastic model finding procedures have significantly extended the range and size of constraint and satisfiability problems that can be solved effectively.

It has now become feasible to solve problem instances with tens of thousands of variables and up to several million constraints. Being able to tackle problem encodings of this size leads to a qualitative difference in the kinds of tasks that one might consider for general search and reasoning methods.

I predict that this is just the beginning of a shift to compute-intensive methods in AI. Given the tremendous difficulty of supplying a program with certain kinds of domain-specific knowledge, we may in fact be better off attacking the inherent combinatorics head on. I do not mean to suggest that simply faster machines and implementations will be sufficient. Further research on search and reasoning procedures and on problem encodings will certainly be needed. Also, there are still many examples where general methods quickly become ineffective. Nevertheless, we now have concrete evidence that the battle against combinatorics can be won, at least in certain interesting cases. The challenge is to continue pushing the compute-intensive approach into areas that have previously been considered off-limits.

An interesting parallel development is taking place in the area of natural language understanding. In this area, we see an analogous shift to data-intensive, statistical techniques. The use of statistical methods to infer linguistic structure from large text corpora can again be viewed as promising way of avoiding the difficult task of explicitly coding such information by hand.

Below follows a list of research directions and challenges.

Reasoning from first-principles: Compilation, approximation, and abstraction. Compute-intensive methods operate from "first-principles", in that little or no domain-specific search control knowledge is used. This kind of reasoning or search generally requires substantial computational resources. In order to build practical systems, it may therefore be necessary to shift as much as possible of the computational cost to an off-line pre-processing phase. Such pre-processing may involve various forms of compilation, approximation, and abstraction. Although several interesting formalisms for knowledge-based approximation, compilation, and abstraction have been developed, much work remains to be done, in particular to show a clear payoff of these methods in realistic applications [2,20,21].

Research on abstraction and reformulation may also shed new light on human problem solving abilities. The success of compute-based methods in reaching expert-level performance on certain tasks revives the intriguing question of how human cognition is able to avoid the apparent inherent combinatorics of the underlying problem domains. Humans appear to employ clever abstractions, problem reformulations, or perhaps heuristic rules to dramatically change or reduce the search space. However, exactly how this is done is still very much an open research issue.

The nature of hard computational problems and its connections to statistical physics. During the past ten years, we have obtained a much better understanding of the nature of computationally hard problems, mainly by studying distributions of random problem instances [3,4,11,15,16]. A better understanding of more structured instances, as they occur in actual applications, is needed. This is a rapidly emerging area of research incorporating concepts and methods from statistical physics and combinatorics.

Robustness, uncertainty, and the brittleness of knowledge representation schemes. Subtle differences in problem representations can lead to large differences in our ability to solve the underlying problem. Moreover, many problem encodings used in, for example, planning and reasoning, lead to solutions that are quite brittle when it comes to small changes in the original problem. A *principled* approach for designing "good" problem representations is needed [23].

Experimentation with close ties to algorithmic and theoretical work. Worst-case and theoretical average-case results for search and reasoning procedures often tell us surprisingly little about their performance on real-world problems. In general, detailed experimentation is necessary to analyze search and reasoning strategies. Such studies often give rise to new algorithms, and lead to new theoretical models for explaining the empirically observed phenomena.

Challenge applications. AI planning is a good example of a challenge domain for compute-intensive methods, given that it is a notoriously hard combinatorial search problem. The recent work on reformulating planning in terms of a large constraint satisfaction problem has brought a novel perspective on AI planning, and there is still much room for improvement in this area. Other promising areas of application of compute-intensive methods are, for example, in scheduling, diagnosis, and machine learning.

The integration of machine learning techniques with reasoning and search methods. Over the years, the area of machine learning and that of reasoning and search have developed more or less independently. Recently, we have seen some encouraging results that suggest that learning techniques can be used to boost the performance of reasoning and search methods. The central idea behind this line of research is that of using machine learning methods to discover useful features of the combinatorial search space underlying the reasoning or search task [1,26]. For example, in our recent work, we have integrated learning techniques into the BlackBox planning approach. By incorporating first-order rule learning strategies into the BlackBox system, we have obtained promising results on learning from a large set of previously solved, smaller planning problems [5,6]. We hope that this work will result in a general method for speeding up problem solvers by learning from past experience.

References

[1] J.A. Boyan and A.W. Moore, Learning evaluation functions for global optimization and Boolean satisfiability, in: *Proc. 15th National Conference on Artificial Intelligence (AAAI)*, Madison, WI (1998).

[2] M. Cadoli and F.M. Donini, A survey on knowledge compilation, AI Communications (1998).

[3] P. Cheeseman, B. Kanefsky, M. Taylor and M. William, Where the really hard problems are, in: *Proc. IJCAI-91* (1991) pp. 331–336.

[4] T. Hogg, B.A. Huberman and C.P. Williams, Phase transitions in problem solving, Artificial Intelligence (Special Issue) 81 (1996).

[5] Y. Huang, B. Selman and H. Kautz, Control knowledge in planning: benefits and tradeoffs, in: *Proc. of the 16th Natl. Conf. on Artificial Intelligence (AAAI-99)*, Orlando, FL (1999).

[6] Y. Huang, B. Selman and H. Kautz, Learning declarative control rules for constraint-based planning, Under review for Machine Learning (2000).

[7] G. Kasparov, The day that I sensed a new kind of intelligence, Time, May 26 (1997) 66–67. See also Time, March 25 (1996) 55.

[8] H. Kautz and B. Selman, An empirical evaluation of knowledge compilation, in: *Proc. of the 12th Natl. Conf. on Artificial Intelligence (AAAI-94)*, Seattle, WA (1994) pp. 155–161.

[9] H. Kautz and B. Selman, Pushing the envelope: planning, propositional logic, and stochastic search, in: *Proc. of the 13th Natl. Conf. on Artificial Intelligence (AAAI-96)*, Portland, OR (1996) pp. 1194–1201.

[10] H. Kautz and B. Selman, Unifying SAT-based and graph-based planners, in: *Proc. of the 15th Intl. Joint Conf. on Artificial Intelligence (IJCAI-99)* (1999).

[11] S. Kirkpatrick and B. Selman, Critical behavior in the satisfiability of random Boolean expressions, Science 264 (1994) 1297–1301.

[12] G. Kolata, Computer math proof shows reasoning power, New York Times, December 10 (1996).

[13] W. McCune, Solution of the Robbins problem, Journal of Automated Reasoning 19(3) (1997) 263–276.

[14] D. McDermott, Yes, computers can think, New York Times, May 14 (1997). See ftp://ftp.cs.yale.edu/pub/mcdermott/papers/deepblue.txt.

[15] D. Mitchell, B. Selman and H. Levesque, Hard and easy distribution of SAT problems, in: *Proc. 10th Natl. Conf. on Artificial Intelligence (AAAI-92)*, San Jose, CA (1992) pp. 459–465.

[16] R. Monasson, R. Zecchina, S. Kirkpatrick, B. Selman and L. Troyansky, Determining computational complexity from characteristic 'phase transitions', Nature 400(8) (1999) 133–137.

[17] B. Selman, Near-optimal plans, tractability, and reactivity, in: *Proc. of the 4th Int. Conf. on Knowledge Representation and Reasoning (KR-94)*, Bonn, Germany (1994) pp. 521–529.

[18] B. Selman, Stochastic search and phase transitions: AI meets physics, in: *Proc. of the 14th International Joint Conference on Artificial Intelligence (IJCAI-95)*, Montreal, Canada (1995).

[19] B. Selman, R. Brooks, T. Dean, E. Horvitz, T. Mitchell and N. Nilsson, Challenge problems for artificial intelligence, in: *Proc. of the 13th Natl. Conf. on Artificial Intelligence (AAAI-96)*, Portland, OR (1996) pp. 193–224.

[20] B. Selman and H. Kautz, Knowledge compilation using Horn approximations, in: *Proc. of the 9th Natl. Conf. on Artificial Intelligence (AAAI-91)*, Boston, MA (1991) pp. 904–909.

[21] B. Selman and H. Kautz, Knowledge compilation and theory approximation, Journal of the ACM 43(2) (1996) 193–224.

[22] B. Selman, H. Kautz and B. Cohen, Local search strategies for satisfiability testing, in: DIMACS Series in Discrete Mathematics and Theoretical Computer Science, Vol. 26 (Amer. Math. Soc., Providence, RI, 1993) pp. 521–532.

[23] B. Selman, H. Kautz and D. McAllester, Computational challenges in propositional reasoning and search, in: *Proc. IJCAI-97*, Nagoya, Japan (1997).

[24] B. Selman, H. Levesque and D. Mitchell, GSAT: A new method for solving hard satisfiability problems, in: *Proc. 10th Natl. Conf. on Artificial Intelligence (AAAI-92)*, San Jose, CA (1992) pp. 440–446.

[25] D. Weld, Recent advances in AI planning, Artificial Intelligence Magazine 20 (1999) 93–123.

[26] W. Zhang and T.G. Dietterich, High-performance job-shop scheduling with a time-delay TD(lambda) network, Advances in Neural Information Processing Systems 8 (1996) 1024–1030.