First International Congress on Tools for Teaching Logic
Local Search Algorithms for SAT in a Course in Logic for Computer Science

Ramón Béjar, Alba Cabiscol and Felip Manyà
Departament d'Informàtica i Enginyeria Industrial
Universitat de Lleida
Jaume II, 69, E-25001 Lleida, Spain
{ramon,alba,felip}@eup.udl.es

1 Introduction

Nowadays, the undergraduate curriculum in computer science of most universities includes an introductory course in logic. In this kind of course, the syllabus usually includes the syntax and semantics of propositional and …rst-order logic, as well as some proof systems such as natural deduction, resolution and tableaux. In some cases, there are also some lecture devoted to explain some basic concepts on logic programming and some practical work using a Prolog interpreter.

One of the most important problems studied in an introductory course in logic is the propositional satis…ability problem (SAT); i.e., the problem of deciding if there is a truth assignment for the variables in a propositional formula in conjunctive normal form (CNF formula) that makes the formula true according to the usual rules of interpretation. SAT is important for computer science because it was the …rst problem shown to be NP-hard [1], and a good algorithm for SAT would entail a good algorithm for other NP-hard problems. As Mitchell et al. [4] say, SAT is also of special concern to arti…cial intelligence because of its direct relationship to deductive reasoning (i.e., given a collection of base facts ¡, a sentence Á may be deduced i¤ ¡ [f:Ág is not satis…able). Many other forms of reasoning including default reasoning, diagnosis, planning and image interpretation, also make direct appeal to satis…ability.

In a typical course in logic for computer science, students learn some complete algorithm for SAT such as the Davis-Putnam (DP) procedure [2]. In this paper we propose to complement the study of complete satis…ability solvers with the introduction of a lecture on local search algorithms for SAT. On the one hand, this topic can be very helpful for understanding concepts such as

satis…ability, completeness, incompleteness, local search algorithm, random algorithm and NP-hard problem. On the other hand, students see more closely the relationship between logic and computer science.

2 Organization of the lecture

In this section we describe a possible organization of a lecture on local search algorithms for SAT addressed to undergraduate computer science students. We divide the organization of the lecture into three parts. For each part, we give the objectives we want to achieve, what we believe students should learn and an sketch of the development of the lecture. In the following, we assume that students know the terminology of propositional CNF formulas, the concept of satis…ability and a complete algorithm for SAT.

Part 1: Why to use local search algorithms for SAT.

The objective of this part is to present instances of SAT that are di¢cult (or even that are not possible) to solve using complete algorithms for SAT in order to justify the use of incomplete local search algorithms for SAT. In this part students should learn what is the phase transition phenomenon and how to generate random 3-SAT instances.

First of all, we introduce the generation of random instances for the 3-SAT problem. We explain that a generator of such instances has as parameters the number of clauses (C) and the number of propositional variables (V). Given C and V, an instance is produced by generating C clauses. Each clause is produced by choosing uniformly at random three literals with di¤erent propositional variable from the set of literals that can be formed with the variables of V. Then, we present the phase transition phenomenon that occurs for the random 3-SAT problem. We explain that Mitchell et al. [4] reported results from experiments on testing the satis…ability of random 3-SAT instances with the DP procedure. They observed that (i) there is a sharp phase transition from satis…able to unsatis…able instances for a value of the ratio of the number of clauses to the number of variables. At lower ratios, most of the instances are under-constrained and are thus satis…able. At higher ratios, most of the instances are over-constrained and are thus unsatis…able. The value of that ratio where 50% of the instances are satis…able is referred to as the threshold; and (ii) there is an easy-hard-easy pattern in the computational di¢culty of solving problem instances as that ratio is varied; the hard instances occur in the area near the threshold. Nowadays, there is strong experimental evidence that the value of the threshold is around 4:25, but there are no analytical results. Only lower and upper bounds on the location of the threshold are known [3]. Finally, we point out that the existing complete algorithms for SAT have not been able to solve such hard instances for $V \geq 500$, but they have been solved easily using the local search algorithms we will explain in this lecture.

Part 2: Description of local search procedures for SAT.

The objective of this part is to introduce the concepts and terminology of local search methods and to present how they can be used to …nd satisfying assignments. Students should learn the notion of incomplete algorithm, the ideas behind two local search algorithms for SAT (GSAT [6] and WalkSAT [5]) and strategies to escape from local minima.

First of all, we introduce the concepts of incomplete algorithm, local search method and local minima. Secondly, we describe in detail the procedure GSAT. We explain that GSAT performs a local search of the space of truth assignments by starting with a randomly generated assignment, and then repeatedly changing ("‡ipping") the assignment of a variable that leads to the largest decrease in the total number of unsatis…ed clauses. Thirdly, we illustrate a typical search of GSAT for a random 3-SAT instance by plotting the number of unsatis…ed clauses as a function of the number of ‡ips performed, and discuss the problem of local minima. We see that the mechanism to escape from local minima used in GSAT is to restart from a new randomly generated solution. Fourthly, we describe in detail the procedure WalkSAT and explain that WalkSAT applies random walks as a strategy to escape from local minima. We show that a typical search of WalkSAT outperforms a typical search of GSAT in hard instances of the phase transition. Finally, we present tabu search as another strategy to escape from local minima.

Part 3: Experimental investigation.

The objective of this part is to present how to solve combinatorial problems using satis…ability solvers and to show that local search algorithms for SAT outperform the existing complete algorithms for SAT on several classes of problems. Students should learn how to encode combinatorial problems as SAT instances, when it is appropriate to use incomplete algorithms for SAT and how to conduct an experimental investigation for evaluating and comparing algorithms.

First of all, we explain that some combinatorial problems can be e¢ciently solved by reducing them to SAT and then determining the satis…ability of the SAT instance generated with a satis…ability solver. We discuss the importance of de…ning appropriate encodings, as well as the need of fast satis…ability solvers equipped with suitable data structures for representing CNF formulas. We illustrate the question of encodings with the graph coloring problem. Secondly, we describe the design of an experimental investigation we conducted in order to evaluate and compare several satis…ability solvers (DP, GSAT and WalkSAT), and discuss the experimental results obtained for hard instances of the phase transition and for the graph coloring problem. Finally, we give some bibliographical references, and some URL's that allow us to access to publicly available implementations of both complete and incomplete algorithms for SAT, and to collections of combinatorial problems encoded as SAT instances.

3 Discussion

To our best knowledge, standard books on logic for computer science do not include a chapter on local search algorithms for SAT. However, we have observed that this novel topic increases the motivation and interest of computer science students. We believe that this is so because students see more closely the relationship between logic and computer science. Another factor is that students see that propositional logic can be used as a powerful problem solving system.

In our opinion, there are several reasons for introducing a lecture on local search algorithms for SAT in a course in logic for computer science: (i) it helps students to understand logical concepts such as satis…ability, completeness, incompleteness and decidability; (ii) it helps students to understand computational concepts such as local search algorithm, random algorithm and NP-hard problem; (iii) students deal with computationally di¢cult instances, understand the computational limits of NP-hardness and learn a possible way of overcoming computational hardness; (iv) students learn how to encode and solve combinatorial problems using propositional logic; (v) student learn how to conduct an experimental investigation for evaluating and comparing algorithms; and (vi) students apply notions that have studied in other courses (complexity, data structures, programming languages).

Acknowledgements

This research was partially supported by the projects ALFA Tools for Teaching Logic and CICYT TIC96-1038-C04-03. The …rst author was supported by a doctoral fellowship of the Comissionat per a Universitats i Recerca (1998FI00326).

References

[1] S. Cook. The complexity of theorem-proving procedures. In Proc. 3rd Annual ACM Symposium on Theory of Computing, pages 151–158, 1971.

[2] M. Davis, G. Logemann, and D. Loveland. A machine program for theorem-proving. Communications of the ACM, 5:394–397, 1962.

[3] L. M. Kirousis, E. Kranakis, and D. Krizanc. Aproximating the unsatis…ability threshold of random formulas. In Proc. 4th Annual European Symposium on Algorithms, ESA'96, pages 27–38, 1996.

[4] D. Mitchell, B. Selman, and H. Levesque. Hard and easy distributions of SAT problems. In Proc. 10th National Conference on Arti…cial Intelligence, AAAI'92, pages 459–465, 1992.

[5] B. Selman, H. A. Kautz, and B. Cohen. Noise strategies for improving local search. In Proc. 12th National Conference on Arti…cial Intelligence, AAAI'94, pages 337–343, 1994.

[6] B. Selman, H. Levesque, and D. Mitchell. A new method for solving hard satis…ability problems. In Proc. 10th National Conference on Arti…cial Intelligence, AAAI'92, pages 440–446, 1992.