ANALYSIS OF POLYNOMIAL APPROXIMATION ALGORITHMS FOR CONSTRAINT EXPRESSIONS

Karl J. Lieberherr*
Institut für Informatik
ETH Zurich, 8092 Zurich

Stephen A. Vavasis
Dept. of Mathematics, Princeton University
Princeton, NJ 08544

Extended Abstract
(Proofs omitted)

Abstract

The generalized maximum satisfiability problem contains a large class of interesting combinatorial optimization problems. Since most of them are NP-complete we analyze fast approximation algorithms.

Every generalized ψ-satisfiability problem has a polynomial ε_ψ-approximate algorithm for a naturally defined constant ε_ψ, $0 \le \varepsilon_\psi < 1$ which is determined here explicitly for several ψ. It is shown that ε_ψ can be approximated by the Soviet Ellipsoid Algorithm. The fraction ε_ψ is known to be best-possible in the sense that the following set is NP-complete: The ψ-formulas S which have an assignment satisfying the fraction $\tau' > 1-\varepsilon_\psi$ (τ' rational) of <u>all</u> clauses in S.

Among other results we also show that for many ψ, local search algorithms fail to be ε_ψ-approximate algorithms. In some cases, local search algorithms can be arbitrarily far from optimal.

1. Introduction

The performance of fast approximation algorithms for the maximum ψ-satisfiability problem is investigated. This problem class contains e.g. the following NP-complete problems which are discussed in [Garey/Johnson (1979)]: MAX CUT, EXACT COVER, SET SPLITTING, NOT-ALL-EQUAL SAT, ONE-IN-THREE SAT. An instance of a maximum ψ-satisfiability problem consists of a sequence of constraints and the problem is to find an

---

*) Currently on leave from Princeton University. This research is supported by NSF grant MCS80-04490.

assignment which satisfies as many as possible. These constraint satisfaction problems appear in many practical applications like time table scheduling, minimizing PLA's, decoding messages, designing statistical experiments, etc.

Maximization problems of the type described above are naturally formulated as maximum ψ-satisfiability problems [Schaefer (1978), Lieberherr (1982)]. ψ is a finite set of logical relations $R_1,\ldots,R_m$ which are used to express the constraints. A ψ-formula S with n variables is a finite sequence of clauses each of the form $R_i(x_i,\ldots,x_{r_i})$. r_i is the rank of R_i and $x_1,\ldots,x_{r_i}$ are a subset of the n variables of S. The maximum ψ-satisfiability problem consists of finding, for any ψ-formula S, an assignment to the variables of S satisfying the maximum number of clauses.

It is well-known that for most ψ it is NP-equivalent to find the optimal assignment for a given ψ-formula [Schaefer (1978)]. Therefore it is justified to analyze polynomial heuristics and to determine how close to the optimal solution they come [Garey/Johnson (1979)].

As shown in [Lieberherr (1982)] each maximum ψ-satisfiability problem has an associated constant τ_ψ, which has the following meaning: The fraction τ_ψ of the clauses can be satisfied in polynomial time in any ψ-formula S. However if it can be decided in polynomial time whether at least the fraction $\tau'>\tau_\psi$ (τ' rational, $\tau'<1$) of the clauses can be satisfied, then the maximum ψ-satisfiability problem can be solved in polynomial time. Therefore, if the maximum ψ-satifisfiability problem is NP-equivalent, then the set of ψ-formulas which have an assignment satisfying the fraction $\tau'>\tau_\psi$ (τ' rational, $\tau'<1$) of the clauses is NP-complete. In this case τ_ψ is called a P-optimal threshold.

It is now possible to answer the following question: Is there a polynomial ε_ψ-approximate algorithm for the maximum ψ-satisfiability problem? I.e. is there a polynomial algorithm which for any instance is guaranteed to find an assignment within ε_ψ of the optimal assignment for some constant $\varepsilon_\psi<1$? (For a general definition of an ε-approximate algorithm see e.g. [Papadimitriou/Steiglitz (1981)]. It is well-known that there are maximization problems which do not have a polynomial ε-approximate algorithm for any $\varepsilon<1$ (see e.g. [Sahni/Gonzales (1976)]),

however the maximum ψ-satisfiability problem has a polynomial $(1-\tau_\psi)$-approximate algorithm (MAXMEAN* in [Lieberherr (1982)]). This follows directly from the definition of τ_ψ. It is easy to prove that $\tau_\psi>0$ if ψ does not contain the empty relation.

It is in general an open problem whether there are polynomial ε'-approximate algorithms for $\varepsilon'<1-\tau_\psi$ or whether there is even a polynomial approximation scheme for the maximum ψ-satisfiability problem ([Huang/Lieberherr (1981)], see [Papadimitriou/Steiglitz (1981)] for the definition of a polynomial approximation scheme).

The algorithms which we analyze in this paper have the nice property of being "P-optimal". The basic theme of a theory of P-optimal approximation algorithms is the question of determining a threshold that can be satisfied efficiently and showing that satisfying more than the threshold is hard. An algorithm that guarantees to satisfy the threshold is said to be P-optimal. Intuitively, these P-optimal algorithms do the "best possible" of what can be done efficiently. The concept of P-optimal approximation is applicable to any NP-complete problem.

2. Generalized Satisfiability

In this section we define first the generalized satisfiability problem, following [Schaefer (1978)]. However the same concept has been used in artificial intelligence and related areas (see [Freuder (1978)]) under the names "constraint expressions" and "networks of constraints". In the last part of the section we summarize several useful facts about maximum ψ-satisfiability that are special cases of results published in [Lieberherr (1982)].

Let $\psi=\{R_1,\ldots,R_m\}$ be any finite set of logical relations. A logical relation is defined to be any subset of $\{0,1\}^r$ for some integer $r\geq 1$. The integer r is called the rank of the relation. Define a ψ-formula to be any sequence of clauses, each of the form $R_i(\zeta_1,\zeta_2,\ldots)$, where $\zeta_1,\zeta_2,\ldots$ are distinct, non-negated variables, whose number matches the rank of $R_i, i\in\{1,\ldots,m\}$. The ψ-satisfiability problem is the problem of deciding whether a given ψ-formula is satisfiable. The main result in [Schaefer (1978)] characterizes the complexity of the ψ-satisfiability problem for every finite set ψ of logical relations. An interesting feature of this characterization is, that for any such ψ, the ψ-satisfiability problem is either polynomial-time decidable or NP-complete.

The maximum ψ-satisfiability problem consists of finding, for any formula S, an assignment to the variables of S satisfying the maximum number of clauses. The difficulty of approximating the maximum ψ-satisfiability problem is the subject of this paper.

It turns out, that the following question deserves further attention: Given a set ψ of relations, which fraction τ_ψ of the clauses can be satisfied for every ψ-formula S? In other words, we are trying to solve the minimax problem:

$$\tau_\psi = \inf_{\substack{\text{all } \psi\text{-formulas}\\ S}} \; \max_{\substack{\text{all assignments}\\ J \text{ for } S}} \; \frac{\text{SATISFIED}(S,J)}{\text{CLAUSES}(S)} .$$

(SATISFIED(S,J) is the number of satisfied clauses in formula S under assignment J. CLAUSES(S) is the number of clauses in formula S.) The following theorem which is proven in [Lieberherr (1982)] (in generalized form) shows that the constant τ_ψ is best possible in an algorithmic sense.

Theorem 2.1

Let ψ be a finite set of relations such that the ψ-satisfiability problem is NP-complete.

1.1 There is a polynomial algorithm MAXMEAN* that satisfies the fraction τ_ψ of the clauses.

1.2 For any rational $\tau' > \tau_\psi$ the set of ψ-formulas that have an assignment satisfying at least the fraction τ' of the clauses is NP-complete.

The above theorem claims that algorithm MAXMEAN* (given below) is relative P-optimal.

Algorithm MAXMEAN*
Input: ψ-formula S.
Output: Assignment which satisfies at least τ_ψ clauses.

```
 maxassignment := 0;
 loop
    compute k such that
     $\max_{0 \le k' \le n} \text{mean}_{k'}(S) = \text{mean}_k(S)$
    {$\text{mean}_k(S)$ is the average number of satisfied clauses among all
    assignments having exactly k ones.  $\text{mean}_k(S)$ is a polynomial in k
    which can be efficiently computed}
    for all variables x in S do
       if $\text{mean}_{k-1}(S_{x=1}) > \text{mean}_k(S_{x=0})$
       then J[x] := 1; k := k-1; S := $S_{x=1}$
       else J[x] := 0; S := $S_{x=0}$
       {$\text{mean}_{-1}(S)=\text{mean}_0(S), \text{mean}_{n+1}(S)=\text{mean}_n(S)$}
   h := SATISFIED(S,J);
   if h>maxassignment then
   maxassignment := h else exit;
   rename all variables in S which are assigned 1 by J;
 end
```

3. Absolute P-optimality of MAXMEAN*

This section improves the results in [Lieberherr (1982)]. Here we show that it is hard to decide whether one can improve on MAXMEAN*. In [Lieberherr (1982)] it was only shown that it is hard to find an assignment which is better than the assignment found by MAXMEAN*. The new insight we gained is that in this context decision and search problems have about the same complexity. Namely we show that the following two problems are polynomially related to each other.

1. Is there an assignment for a given ψ-formula which satisfies more than
$$\text{maxmean}(S) = \max_{0 \le k \le n} \text{mean}_k(S)$$
clauses?
2. Find an assignment for a given ψ-formula which satisfies more than maxmean(S) clauses (if such an assignment exists).

This relationship holds since the following two problems are polynomially related.

1. Is there an assignment for a given ψ-formula which is better than the assignment $J_{ALL\ 0}$ (which assigns 0 to all variables).
2. Find an assignment which is better than the assignment $J_{ALL\ 0}$ (if such an assignment exists).

This question is related to the absolute P-optimality of algorithm MAXMEAN*. We first need a few definitions.

Let {S} be the set of instances of an NP-equivalent maximum ψ-satisfiability problem. A polynomial time computable function g: {S} → R (rational numbers) is said to be absolute P-optimal, if the following two conditions hold

a) there is a polynomial algorithm M which finds for every formula x in {S} an assignment J so that $SATISFIED(x,J) \geq g(x)$ and

b) the set of instances x in {S} which have an assignment such that $SATISFIED(x,J) > g(x)$ is NP-complete (by a Turing reduction).

The function g has the following intuitive meaning: There is a polynomial algorithm that finds an assignment which is at least as "good" as g(x), but to do "better" is NP-complete. Algorithm M is said to be absolute P-optimal with respect to function g.

The renaming of a variable with respect to value γ is a substitution of $e(x,\gamma)=(\gamma-x) \bmod 2$ for variable x. The default is "with respect to value 1". Therefore the renaming of variable x is 1-x.

Let J be an assignment for formula S. The renaming R of formula S with respect to J is a substitution of e(x,J(x)) for all variables x in S. The resulting formula R(S) is called the renamed formula S (with respect to J).

Let $Q(x_1,\ldots,x_n)$ be a relation and let J be an assignment for $x_1,\ldots,x_n$. The renamed relation Q with respect to J is the relation L(Q,J) defined by

$$L(Q,J)(e(x_1,J(x_1)),\ldots,e(x_n,J(x_n))) \leftrightarrow Q(x_1,\ldots,x_n) \ .$$

By definition

$$Q(J(x_1),\ldots,J(x_n))=L(Q,J)(0,\ldots,0) \ .$$

A set of relations ψ is said to be closed under renaming if all relations that can be generated from relations in ψ by renaming, are in ψ.

Theorem 3.1

Let ψ be a finite set of relations which is closed under renaming. Assume that the maximum ψ-satisfiability problem is NP-equivalent. Then the set of ψ-formulas which have an assignment satisfying more clauses than the assignment $J_{ALL\ 0}$ (which assigns 0 to all variables) is NP-complete (by a Turing reduction).

Lemma 3.2

Let ψ be a finite set of relations.
Let $\Omega(S)$ be a polynomial decision algorithm for deciding whether there is a better assignment than the assignment $J_{ALL\ 0}$ for a given ψ-formla S. Then there is a polynomial algorithm for finding an assignment satisfying more clauses than $J_{ALL\ 0}$.

This Lemma shows that in this context decision and search problems have a polynomially related complexity.

The proof of the following corollary shows the connection between the problem of improving the assignment $J_{ALL\ 0}$ and the absolute P-optimality of MAXMEAN*.

Corollary 3.3

If the maximum ψ-satisfiability problem is NP-equivalent and ψ is closed under renaming, then algorithm MAXMEAN* is absolute P-optimal with respect to the function maxmean(S).

4. Closed Form Analysis of MAXMEAN*

In this section we determine the P-optimal thresholds τ_ψ for various ψ by explicit formulas. It is conjectured that τ_ψ is in general an algebraic number but no efficient method is known to determine (efficiently) an irreducible polynomial which defines τ_ψ ([Demyanov/Malozemov (1974)], [Charalambous/Conn (1978)]).

If ψ contains only one relation then the computation of τ_ψ involves only a maximization of a real polynomial over [0,1] and is therefore straight-forward. We refer the reader to [Lieberherr/Specker (1982)] for examples, how to compute τ_ψ by explicit formulas if ψ contains several relations. In the next section we will show that τ_ψ can be approximated efficiently by the Soviet Ellipsoid Algorithm for linear programming.

4.1 Hypergraph coloring

In this and the next subsection we allow the variables to assume c values 0,1,...,c-1 where $c \geq 2$. So far we have only considered logical variables (c=2) in this paper.

Let $R_m(x_1,x_2,\ldots,x_m)$ be the relation that holds, iff $\{0,1,\ldots,c-1\} \subseteq \{x_1,x_2,\ldots,x_m\}$. Let $\psi_m=\{R_m\}, m \geq 3$. A ψ_m-formula S has the following interpretation as a hypergraph coloring problem. Each variable of S corresponds to a node of the hypergraph. Each clause of S corresponds to a hyperedge and the clause expresses that each possible color is assigned to at least one node on the hyperedge. For c=2 this is the regular graph coloring problem (see section 4.2).

Theorem 4.2

$$\tau_{\psi_m} = \frac{c!}{c^m} S_2(m,c)$$

is a P-optimal threshold, if $c \geq 3$ and $m \geq 3$.
τ_{ψ_m} is determined in [Erdös/Kleitman (1968)] ($S_2(m,c)$ are the Stirling numbers of the second kind).

4.2 Graph coloring

Let R(x,y) be the relation that holds, if $x \neq y$. Let $\psi=\{R\}$ and allow c values 0,1,...,c-1 to be assigned to the variables. This maximum ψ-satisfiability problem is a generalization of the graph coloring problem. The c values are intended to be colors. Each clause in a ψ-formula corresponds to an edge and requires that the endpoints have different colors.

Theorem 4.3

$$\tau_\psi = \frac{c-1}{c}$$

is a P-optimal threshold, if $c \geq 3$.

([Vitanyi (1981)] and [Sahni/Gonzales (1976)] also determine τ_ψ and give an efficient algorithm to satisfy at least the fraction τ_ψ.

4.3 Exact Cover

We analyze a similar special case of the exact cover problem. Let $R_m(x_1,x_2,\ldots,x_m)$ be the relation that holds iff exactly one of the m variables is 1. Let $\psi_m=\{R_m\}$.

Theorem 4.4

$$\tau_{\psi_m} = \left(\frac{m-1}{m}\right)^{m-1}$$

is a P-optimal threshold, if $m \geq 3$.

5. Efficient Approximation of Performance Bound

The results of this section show that the computation of the performance bound of MAXMEAN* can be done efficiently. This completes our result in [Lieberherr (1982)] that MAXMEAN* can be generated efficiently.

In the last section we expressed the performance bound of MAXMEAN* by giving explicit formulas for the solution of the minimax problems. In general this is not possible since Galois theory proves that the roots of high degree polynomials cannot be expressed by radicals.

However linear programming allows us to compute the performance bound numerically in an efficient way. The Soviet Ellipsoid Algorithm (see e.g. [Papadimitriou/Steiglitz (1981)] will provably solve the problem in polynomial time. The clue to this result is to consider the minimax problem from a different point of view which yields a linear programming interpretation.

Theorem 5.2
Let ψ be a finite set of relations $R_1,\ldots,R_m$. Then the fraction of clauses which can be satisfied in every ψ-formula with n variables (and which will be satisfied by MAXMEAN*) is the solution of a linear program with m+n+2 constraints and m+1 variables. Each number occuring in the program has $O(\log(n))$ bits.

5. Limitations of Local Search

Local search is known to be an excellent heuristic for solving hard combinatorial optimization problems (see e.g. [Papadimitriou/Steiglitz (1982)]. Our empirical observation of the good performance of local search algorithms on the maximum ψ-satisfiability problem motivated us to compare their performance with the known performance of MAXMEAN*.

Our results are negative in the sense that many local search algorithms do not guarantee to satisfy the fraction τ_ψ of the clauses in a given ψ-formula i.e. are not P-optimal. In some cases we even show that local search is arbitrary far from being optimal.

We only give a short summary of our results (for a complete description see [Lieberherr/Vavasis (1982)]). For relations of rank 2 or 3 which are closed under renaming we show that simple local search algorithms are P-optimal. For relations of rank 4 or higher we show that a large class of local search algorithms are not P-optimal. In our reasoning we use ideas from Bernstein's proof of Weierstrass' theorem.

Conclusion

MAXMEAN* is shown to be optimal in several respects: It is relative and absolute P-optimal. It is also bestpossible for the ψ-formulas with a fixed number n of variables in the sense that MAXMEAN* satisfies at least as many clauses as one can satisfy in every ψ-formula with n variables. This last property makes MAXMEAN* superior to previously proposed algorithms with the performance bound τ_ψ. Our relative optimality results depend in a crucial way on the fact that we consider all ψ-formulas. If we restrict our attention to proper subsets of the set of all ψ-formulas, then the above questions may become very hard. The reason is that e.g. the well-known, and now solved, 4-color conjecture can be put into this framework (express the 4-coloring problem as in section 4 and minimize only among formulas which correspond to planar graphs).

References

Charalambous 1978. Charalambous Conn, "An efficient method to solve the minimax problem directly," SIAM J. Num. Anal. 15, pp. 162-187 (1978).

Demyanov 1974. V.F. Demyanov V.N. Malozemov, Introduction to Minimax, John Wiley & Sons, New York (1974).

Erdös 1968. P. Erdös and D.J. Kleitman, "On coloring graphs to maximize the proportion of multicolored k-edges," Journal of Combinatorial Theory 5 (2) pp. 164-169 (Sept. 1968).

Freuder 1978. E.C. Freuder, "Synthesizing Constraint Expressions," Comm. ACM 21 (11) pp. 958-966 (1978).

Garey 1976. M.R. Garey, D.S. Johnson, and L. Stockmeyer, "Some simplified NP-complete graph problems," Theor. Comput. Sci. 1, pp. 237-267 (1976).

Huang/Lieberherr (1982) M.A. Huang, K.J. Lieberherr, "Towards an approximation scheme for generalized satisfiability", Report 292. Dep. EECS, Princeton University (1982).

Lieberherr 1982. K.J. Lieberherr and E. Specker, "Complexity of Partial Satisfaction II," Report 293, Dep. of EECS, Princeton University (1982).

Lieberherr 1982. K.J. Lieberherr, "Algorithmic extremal problems in combinatorial optimization," Journal of Algorithms, September 1982.

Lieberherr/Vavasis (1982). K.J. Lieberherr, Stephen A. Vavasis, "Limitations of Local Search," Report 302, Dep. of EECS, Princeton University (1982).

Papadimitriou 1982. C.H. Papadimitriou K. Steiglitz, Combinatorial Optimization, Prentice Hall (1982).

Sahni 1976. S. Sahni and T. Gonzales, "P-complete approximation problems," J. ACM 23 (3) pp. 555-565 (1976).

Schaefer 1978. T. Schaefer, "The complexity of satisfiability problems," Proc. 10th Annual ACM Symposium on Theory of Computing, pp. 216-226 (1978).

Vitanyi 1981. P.M.B. Vitanyi, "How well can a graph be n-colored?" Discrete Mathematics 34,,pp. 69-80 (1981).