- Nested SAT: An ordering of the clauses must exist with the property that if a clause C preceeds another clause C' then no variable from C except the first and the last (w.r.t to an ordering of the variables) may be contained in C'.
- READ–2: No variable may occur more than twice in a formula.

It is well known that for inputs from these classes the satisfiability problem is solvable in time proportional to the length of the formula, see [4, 3]. The following remark formalizes the relationship between S_2 and these classes:

Remark 1 *For any one of the classes 2-SAT, HORN, nested SAT and READ-2 there is a boolean function function whose complement can be expressed in S_2 but that cannot be expressed in 2-SAT, HORN, nested SAT and READ–2, respectively.*

References

1. S. Cook. The Complexity of Theorem Proving Procedures. *Proc. 3rd Ann. ACM Symp. on Theory of Computing*, pages 151–158, 1971.
2. G. Gallo and M.G. Scutella. Polynomially Solvable Satisfiability Problems. *Information Processing Letters*, 29(5):221–227, 1988.
3. H. Kleine Büning and T. Lettman. *Aussagenlogik: Deduktion und Algorithmen.* B. G. Teubner, Stuttgart, 1994.
4. D. E. Knuth. Nested satisfiability. *Acta Informatica*, 28:1–6, 1990.

Strong Lower Bounds on the Approximability of some NPO PB-Complete Maximization Problems

Viggo Kann*

Department of Numerical Analysis and Computing Science,
Royal Institute of Technology, S-100 44 Stockholm, Sweden

Abstract. The approximability of several NP maximization problems is investigated and strong lower bounds for the studied problems are proved. For some of the problems the bounds are the best that can be achieved, unless P = NP.
For example we investigate the approximability of MAX PB 0 – 1 PROGRAMMING, the problem of finding a binary vector x that satisfies a set of linear relations such that the objective value $\sum c_i x_i$ is maximized, where c_i are binary numbers. We show that, unless P = NP, MAX PB 0 – 1 PROGRAMMING is not approximable within the factor $n^{1-\epsilon}$ for any $\epsilon > 0$, where n is the number of inequalities, and is not approximable within $m^{1/2-\epsilon}$ for any $\epsilon > 0$, where m is the number of variables.
Similar hardness results are shown for other problems on binary linear systems, some problems on the satisfiability of boolean formulas and the longest induced circuit problem.

Introduction

proximation of NP-complete optimization problems is a very interesting and ive area of research. Since all NP-complete problems are reducible to each er one could suspect that they should have similar approximation properties, this is not at all the case.
The range of approximability of NP-complete problems stretches from prob- that are approximable within every constant in polynomial time, e.g. the psack problem [8], to problems that are not approximable within n^ε for some 0, where n is the size of the input instance, unless P = NP. A problem that is n to be this hard to approximate is the minimum independent dominating problem (minimum maximal independence number) [7].
Even optimization problems whose objective function is bounded by a poly- ial in the size of the input may be hard to approximate. Krentel defined a of optimization problems called OPTP[log n], that consists of all NP op- ation problems that are polynomially bounded [12]. This class, which we call NPO PB, can be divided into two classes, MAX PB and MIN PB, con- g maximization and minimization problems respectively [11]. Berman and

ail: viggo@nada.kth.se, supported by grants from TFR.

Schnitger started to investigate the approximability of MAX PB problems and proved that there are MAX PB-complete problems, i.e. MAX PB problems to which every MAX PB problem can be reduced using an approximation preserving reduction [4]. Several problems are now known to be MAX PB-complete [9]. Later, some minimization problems were shown to be MIN PB-complete [10], and recently Crescenzi, Kann, Silvestri and Trevisan proved that any MIN PB-complete problem is NPO PB-complete and that any MAX PB-complete problem is NPO PB-complete [6]. The classes of MIN PB-complete, MAX PB-complete and NPO PB-complete problems thus coincide.

For every NPO PB-complete problem there is a constant $\alpha > 0$ such that the problem is not approximable within n^α, where n is the size of the input instance, unless P = NP. For some problems, for example minimum independent dominating set, this constant can be chosen arbitrarily close to 1, which means that these problems are incredible hard to approximate.

The problems known to be this hard to approximate are mainly minimization problems. Only a few maximization problems are known to have such an extreme nonapproximability bound, and they are all problems on graphs where one looks for a maximum induced connected subgraph [13]. The problem MIN DISTINGUISHED ONES, where one look for a satisfying boolean variable assignment containing as few true variables as possible from some distinguished set of variables, is NPO PB-complete and not approximable within $n^{1-\epsilon}$, where n is the number of distinguished variables [10]. The corresponding maximization problem is also NPO PB-complete, but no strong lower bound on the approximability is known. One could ask whether minimization problems in some sense can be harder to approximate than maximization problems.

In this paper we will, however, show that this is not true. We will show that several maximization problems, for example MAX DISTINGUISHED ONES, have nonapproximability bounds similar to $n^{1-\epsilon}$. We will do this by constructing approximation preserving reductions from either MIN INDEPENDENT DOMINATING SET or LONGEST INDUCED PATH and use the fact that these two problems have strong lower bounds on the approximability. We conclude that a convenient way to establish both NPO PB-completess results and strong lower bounds is to reduce from MIN INDEPENDENT DOMINATING SET or LONGEST INDUCED PATH. Note that our results do not make use of the quite complicated machinery of interactive proofs and the PCP model that recently have been used for showing approximation hardness of several optimization problems, see for example [3].

In the appendix all problems treated in the text are defined.

1.1 Preliminaries

Definition 1. An NP *optimization problem* A is a fourtuple $(I, sol, m, goal)$ such that

1. I is the set of the *instances* of A and it is recognizable in polynomial time.
2. Given an instance x of I, $sol(x)$ denotes the set of *feasible solutions* of x. These solutions are short, that is, a polynomial p exists such that, for any $y \in sol(x)$, $|y| \leq p(|x|)$. Moreover, it is decidable in polynomial time whether, for any x and for any y such that $|y| \leq p(|x|)$, $y \in sol(x)$.
3. Given an instance x and a feasible solution y of x, $m(x, y)$ denotes the positive integer *measure* of y (often also called the value of y). The function m is computable in polynomial time and is also called the *objective* function.
4. $goal \in \{\max, \min\}$.

The *class* NPO is the set of all NP optimization problems. The goal of an NPO problem with respect to an instance x is to find an *optimum solution*, i.e. a feasible solution y such that $m(x, y) = goal\{m(x, y') : y' \in sol(x)\}$. In the following *opt* will denote the function mapping an instance x to the measure of an optimum solution.

An NPO problem is said to be *polynomially bounded* if a polynomial q exists such that, for any instance x and for any solution y of x, $m(x, y) \leq q(|x|)$. The *class* NPO PB is the set of all polynomially bounded NPO problems. NPO PB = MAX PB $\cup$ MIN PB where MAX PB is the set of all maximization problems in NPO PB and MIN PB is the set of all minimization problems in NPO PB.

Given an instance x of an NPO problem and a feasible solution y of x, we define the *performance ratio of y with respect to x* as $R(x, y) = m(x, y)/opt(x)$ for minimization problems and $opt(x)/m(x, y)$ for maximization problems.

Definition 2. Let A be an NPO problem and let T be an algorithm that, for any instance x of A, returns a feasible solution $T(x)$. Given an arbitrary function $r : N \rightarrow (1, \infty)$, we say that T is an *$r(n)$-approximate algorithm for A* if, for any instance x, the performance ratio of the feasible solution $T(x)$ with respect to x verifies the inequality $R(x, T(x)) \leq r(|x|)$.

Several polynomial time approximation preserving reductions have been defined in the literature. The PTAS-reduction [6], which preserves the performance ratio very well, is suitable for defining complete problems in approximation classes. A problem $A \in$ NPO is NPO-*complete* if, for any $B \in$ NPO, there is a PTAS-reduction from B to A. Similarly, a problem $A \in$ NPO PB is NPO PB-*complete* if, for any $B \in$ NPO PB, there is a PTAS-reduction from B to A. In the same way MAX PB-complete and MIN PB-complete problems can be defined.

Proposition 3 [6]. *Any* MIN PB-*complete problem is* NPO PB-*complete and any* MAX PB-*complete problem is* NPO PB-*complete.*

The approximability for problems that are not approximable within a constant is usually described as a function of the size of the problem instance, or more precisely, as a function of some size parameter, like the number of nodes or edges in an input graph. The PTAS-reduction does not preserve size parameters, so this reduction cannot be used when investigating the approximability (or nonapproximability) of problems that are very hard to approximate, like NPO PB-complete problems. For such problems it is not relevant whether the reduction increases the performance ratio by a constant factor. We will use the

S-reduction, defined in [10], which is a reduction that guarantees that the performance ratio is preserved within *some* constant factor, but has full control over the increase of the size if the problem instance.

Definition 4. ([10]) Let A and B be two NPO problems. A is said to be *S-reducible* to B with size amplification a if there exist three functions f, g, a, and a positive constant c such that:

1. for any $x \in I_A$, $f(x,r) \in I_B$ is computable in time polynomial in $|x|$,
2. for any $x \in I_A$, for any $y \in sol_B(f(x))$, $g(x,y) \in sol_A(x)$ is computable in time polynomial in $|x|$ and $|y|$,
3. $a : R^+ \to R^+$ is monotonously increasing, positive and computable,
4. for any $x \in I_A$, for any $y \in sol_B(f(x))$, $R_A(x, g(x,y)) \leq R_B(f(x), y)$,
5. for any $x \in I_A$, $|f(x)| \leq a(|x|)$.

Proposition 5 [10]. *Given two* NPO *problems F and G, if there is an S-reduction from F to G with size amplification $a(n)$ and G is approximable within some monotonously increasing positive function $u(n)$ of the size of the input instance, then F is approximable within $c \cdot u(a(n))$. Conversely, if F is not approximable within $c \cdot u(a(n))$, then G is not approximable within $u(n)$.*

For constant and polylogarithmic approximable problems the S-reduction preserves approximability within a constant for any polynomial size amplification, since $c \log^k(n^p) = p^k c \log^k n = O(\log^k n)$. For n^c approximable problems it only does this for size amplification $O(n)$, since $c \cdot (O(n))^c = O(n^c)$.

2 Lower Bounds

In this section we will prove lover bounds on the approximability of the following NPO PB-complete problems: Max PB 0 − 1 Programming [4], Max Number of Satisfiable Formulas [14], Max Distinguished Ones [14], Max Ones [14], Max C Bin Sat$^{\mathcal{R}_1;\mathcal{R}_2}$ (maximum constrained binary satisfiable linear subsystem) [1], Max Bin Irrelevant Sat$^{\mathcal{R}}$ (maximum irrelevant binary variables in linear system) [2], and Longest Induced Circuit.

In the references above the problems are defined and are also shown to be Max PB-complete. The problems are therefore NPO PB-complete by Proposition 3. Formal definitions of the problems can be found in the appendix.

We first show that Max PB 0 − 1 Programming is hard to approximate. This result was obtained as a side-effect in the proof of Theorem 5 in [6]. We will then modify this proof to prove hardness results for the other problems.

Theorem 6. Max PB 0 − 1 Programming *is not approximable within $n^{1-\varepsilon}$ for any $\varepsilon > 0$, where n is the number of inequalities, and is not approximable within $m^{1/2-\varepsilon}$ for any $\varepsilon > 0$, where m is the number of variables.*

Proof. Halldórsson has proved that, unless P = NP, Min Independent Dominating Set is not approximable within $n^{1-\varepsilon}$ for any $\varepsilon > 0$, where n is the sum of the number of nodes and edges in the graph [7]. We will use this fact to show that Max PB 0 − 1 Programming is hard to approximate.

We will construct a reduction from Min Independent Dominating Set to Max PB 0 − 1 Programming using the following idea. The objective function, i.e. the number of nodes in the independent dominating set is encoded by introducing an order of the nodes in the solution. The order is encoded by a squared number of 0 − 1 variables in the programming problem, see Fig. 1. A solution of size 1 shall correspond to the 0 − 1 programming objective value n, and a solution of size p shall correspond to an objective value of $\lfloor n/p \rfloor$.

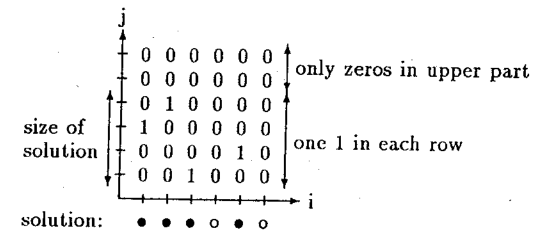

Fig. 1. The idea of the reduction from Min Independent Dominating Set to Max PB 0 − 1 Programming. The variable $x_i^j = 1$ if and only if v_i is the jth node in the solution. There is at most one 1 in each column and in each row.

Given an instance of Min Independent Dominating Set, i.e. a graph with nodes $V = \{v_1, \ldots, v_m\}$ and edges E, construct m^2 variables x_i^j, $1 \leq i, j \leq m$, n variables y_p, $1 \leq p \leq n$, and the following inequalities:

$$\forall i \in [1..m] \quad \sum_{j=1}^{m} x_i^j \leq 1, \quad \text{(at most one 1 in each column)} \tag{1}$$

$$\forall j \in [1..m] \quad \sum_{i=1}^{m} x_i^j \leq 1, \quad \text{(at most one 1 in each row)} \tag{2}$$

$$\forall j \in [1..m-1] \quad \sum_{i=1}^{m} x_i^j - \sum_{i=1}^{m} x_i^{j+1} \geq 0, \quad \text{(only zeros in upper part)} \tag{3}$$

$$\forall (v_i, v_j) \in E \quad \sum_{k=1}^{m} x_i^k + \sum_{k=1}^{m} x_j^k \leq 1, \quad \text{(independence)} \tag{4}$$

$$\forall i \in [1..m] \quad \sum_{k=1}^{m} x_i^k + \sum_{\substack{j:(v_i,v_j)\in E \\ k\in[1..m]}} x_j^k \geq 1, \quad \text{(domination)} \tag{5}$$

$$\forall i \in [1..n] \quad y_i \leq \sum_{k=1}^{m} x_k^{\lfloor n/i \rfloor}, \quad \text{(objective variables)} \tag{6}$$

$$y_i \geq \sum_{k=1}^{m} x_k^{\lfloor n/i \rfloor}. \tag{7}$$

The objective function is defined as $\sum_{p=1}^{n} y_p$.

One can now verify that an independent dominating set of size s will exactly correspond to a solution of the $0-1$ programming problem with objective value $\lfloor n/s \rfloor$ and vice versa.

Suppose that the minimum independent dominating set has size M, then the performance ratio s/M for the independent dominating set problem will correspond to the performance ratio $\lfloor n/M \rfloor / \lfloor n/s \rfloor = s/M(1 \pm m/n)$ for the $0-1$ programming problem, where m/n is the relative error due to the floor operation. By choosing n large enough the relative error can be made arbitrarily small, but for proving the theorem it is enough to choose $n = m$ to obtain a reduction that preserves the approximability within a factor of 2.

The reduction is obviously an S-reduction with amplification $O(m+|E|)$ with respect to the number of inequalities and amplification $O(m^2)$ with respect to the number of variables. The theorem now follows from Proposition 5 together with the fact that MIN INDEPENDENT DOMINATING SET is not approximable within $n^{1-\varepsilon}$ for any $\varepsilon > 0$, where n is the sum of the number of nodes and edges in the graph [7]. □

Theorem 7. MAX NUMBER OF SATISFIABLE FORMULAS *is not approximable within $n^{1-\varepsilon}$ for any $\varepsilon > 0$, where n is the number of formulas, unless* P = NP.

Proof. We modify the construction of the proof of Theorem 6 in the follow[ing] way. Given a graph with nodes $V = \{v_1, \ldots, v_m\}$ and edges E, construct [the] variables x_i^j, $1 \le i,j \le m$, m variables y_p, $1 \le p \le m$, and the m boole[an] formulas $\Phi_p = y_p \wedge \varphi$, $1 \le p \le m$ where φ is the conjunction of the follow[ing] formulas:

$$\forall i \in [1..m], 1 \le j < k \le m \qquad \overline{x_i^j} \vee \overline{x_i^k},$$

$$\forall i \in [1..m], 1 \le j < k \le m \qquad \overline{x_j^i} \vee \overline{x_k^i},$$

$$\forall j \in [1..m-1] \left(\bigvee_{k=1}^{m} x_k^{j+1}\right) \Rightarrow \left(\bigvee_{k=1}^{m} x_k^{j}\right),$$

$$\forall (v_i, v_j) \in E, \forall k, l \in [1..m] \qquad \overline{x_i^k} \vee \overline{x_j^l},$$

$$\forall i \in [1..m] \qquad \bigvee_{k=1}^{m} x_i^k \vee \bigvee_{\substack{j:(v_i,v_j)\in E \\ k\in[1..m]}} x_j^k,$$

$$\forall p \in [1..m] \qquad y_p \equiv \bigvee_{k=1}^{m} x_k^{\lfloor m/i \rfloor}.$$

It is clear that the boolean formulas (8–13) restrict the variables in ex[actly] the same way as the inequalities (1–7) and that the number of satisfied [for]mulas $y_p \wedge \varphi, 1 \le p \le m$ is the same as the sum $\sum_{p=1}^{m} y_p$ in MAX PB [0 – 1] PROGRAMMING. Therefore MAX NUMBER OF SATISFIABLE FORMULAS [is not] approximable within $n^{1-\varepsilon}$ for any $\varepsilon > 0$, where n is the number of for[mulas,] unless P = NP.

Theorem 8. MAX DISTINGUISHED ONES *is not approximable within $n^{1-\varepsilon}$ for any $\varepsilon > 0$, where n is the number of distinguished variables, and* MAX ONES *is not approximable within $n^{1/3-\varepsilon}$ for any $\varepsilon > 0$, where n is the number of variables, unless* P = NP.

Proof. First we prove the result for MAX DISTINGUISHED ONES. This time we modify the construction of the proof of Theorem 7. We first construct a formula with clauses of unbounded size and then rewrite the clauses as 3-SAT clauses.

Let $y_p, 1 \le p \le m$ be the distinguished variables. Apart from the $m^2 + m$ variables x_i^j and y_p we will need m variables t_p, $1 \le p \le m$, and m variables z_p, $1 \le p \le m$. We define these variables by

$$\forall p \in [1..m] \quad t_p \equiv \bigvee_{k=1}^{m} x_k^p, \; z_p \equiv \bigvee_{k=1}^{m} x_p^k. \tag{14}$$

We then can reformulate the formulas (10), (12), and (13) as, respectively,

$$\forall j \in [1..m-1] \quad \overline{t_{j+1}} \vee t_j, \tag{15}$$

$$\forall i \in [1..m] \; z_i \vee \bigvee_{j:(v_i,v_j)\in E} z_j, \tag{16}$$

$$\forall i \in [1..m] \quad y_i \equiv t_{\lfloor m/i \rfloor}. \tag{17}$$

We can now rewrite the equivalences in (14) and (17) as clauses using the [re]writing rule $u \equiv v \longrightarrow (u \vee \overline{v}) \wedge (\overline{u} \vee v)$. All clauses now consist of 2 literals [ex]cept (14) and (16). We use the standard method of rewriting these as clauses [with] 3 literals. In this process we introduce $O(m^2 + |E|)$ new variables.

The conjunction of all these clauses is equivalent to the formula φ in the proof [o]f Theorem 7, and the number of true distinguished variables in a variable as[sig]nment will exactly correspond to the number of satisfied formulas in the other [pro]of. Since the number of distinguished variables is m, MAX DISTINGUISHED [ON]ES is not approximable within $m^{1-\varepsilon}$ for any $\varepsilon > 0$, unless P = NP.

In order to formulate the problem as a MAX ONES problem instance we use [the i]dea in [14] and create copies of the distinguished variables to make each such [var]iable more valuable than all the nondistinguished variables together. □

[Th]eorem 9. MAX C BIN SAT$^{\mathcal{R}_1;\mathcal{R}_2}$ *is not approximable within $n^{1-\varepsilon}$ where n is [the n]umber of optional relations, and not within $n^{1/2-\varepsilon}$ where n is the number [of va]riables, for any $\varepsilon > 0$, and for any $\mathcal{R}_1, \mathcal{R}_2 \in \{=, \ge, >, \ne\}$, unless* P = NP. [MA]X BIN IRRELEVANT SAT$^{\mathcal{R}}$ *is not approximable within $n^{1/3-\varepsilon}$ for any $\varepsilon > 0$, [and] for any $\mathcal{R} \in \{=, \ge, >, \ne\}$, where n is the sum of the number of variables [and] the number of relations, unless* P = NP.

[The] proof is by S-reduction from the MAX PB $0-1$ PROGRAMMING instances [const]ructed in the proof of Theorem 6, and will appear in the full version.

[The]orem 10. LONGEST INDUCED CIRCUIT *is not approximable within $n^{1-\varepsilon}$ for [any] $\varepsilon > 0$, where n is the number of nodes, unless* P = NP.

Proof. We reduce from the problem LONGEST INDUCED PATH that is known to be NPO PB-complete [4] and not approximable within $n^{1-\varepsilon}$ for any $\varepsilon > 0$, where n is the number of nodes, unless P = NP [13].

Given an input graph to LONGEST INDUCED PATH, $G = (V, E)$. We extend this graph by adding one new node w_i for each node $v_i \in V$ and add edges (w_i, v_i) for $1 \le i \le |V|$. We also add one special node w_0 that is connected to all new nodes, i.e. we have edges (w_0, w_i) for $1 \le i \le |V|$.

Every induced path in G (say, starting in node v_s and ending in node v_e) can now be extended to an induced circuit in the new graph by adding the nodes w_s, w_e and w_0. The circuit will have length 4+(length of induced path).

On the other hand, every induced circuit in the new graph that contains the node w_0 will give us an induced path in G containing 4 edges less than the circuit. An induced circuit that does not contain the special node cannot contain any w node, so by removing any node in the circuit we will get an induced path in G containing 2 edges less than the circuit. □

3 Summary and Discussion

The following table summarizes the nonapproximability results in the paper. For each result we give the name of the problem, the nonapproximability exponent b and the size parameter n, saying that the problem is not approximable within $n^{b-\varepsilon}$ for any $\varepsilon > 0$, unless P = NP.

problem	b	size parameter n
MAX PB 0 − 1 PROGRAMMING	1	inequalities
	1/2	variables
MAX NUMBER OF SATISFIABLE FORMULAS	1	formulas
MAX DISTINGUISHED ONES	1	distinguished variables
	1/2	variables
MAX ONES	1/3	variables
MAX C BIN SAT$^{\mathcal{R}_1;\mathcal{R}_2}$	1	optional relations
	1/2	variables
MAX BIN IRRELEVANT SAT$^{\mathcal{R}}$	1/3	variables+relations
LONGEST INDUCED CIRCUIT	1	nodes

In all the treated maximization problems except LONGEST INDUCED CIRCUIT it is NP-complete to decide whether there exists a solution. Since an approximation algorithm must be able to in polynomial time return some solution, we have chosen to include a trivial solution in the input instance.

Another way would be to extend the space of solutions with a special solution with objective value 1. This would make it possible to easily show approximation hardness of some of the problems. For example, we can reduce 3-SAT to MAX NUMBER OF SATISFIABLE FORMULAS by constructing n copies of the formula. If the formula is satisfiable the objective value is n, otherwise it is 1. An algorithm approximating the number of satisfied formulas within $n^{1-\varepsilon}$ would therefore solve the NP-hard 3-SAT problem. We thank Magnús Halldórsson for this remark.

Thus the exact formulation of the problems with respect to trivial solutions is of importance. Since the problem should be to find a *good* solution and not to find *any* solution, we think that our definition is the natural definition when studying the approximability of these optimization problems.

References

1. E. Amaldi and V. Kann. The complexity and approximability of finding maximum feasible subsystems of linear relations. Theoretical Comput. Sci., to appear, 1995.
2. E. Amaldi and V. Kann. On the approximability of removing the smallest number of relations from linear systems to achieve feasability. Technical Report ORWP-6-94, Dep. of Mathematics, Swiss Federal Institute of Technology, Lausanne, 1994.
3. M. Bellare and M. Sudan. Improved non-approximability results. In *Proc. Twenty sixth Ann. ACM Symp. on Theory of Comp.*, pages 184–193. ACM, 1994.
4. P. Berman and G. Schnitger. On the complexity of approximating the independent set problem. *Inform. and Comput.*, 96:77–94, 1992.
5. P. Crescenzi and V. Kann. A compendium of NP optimization problems. Technical Report SI/RR-95/02, Dipartimento di Scienze dell'Informazione, Università di Roma "La Sapienza", 1995. The list is updated continuously. The latest version is available by anonymous ftp from `nada.kth.se` as `Theory/Viggo-Kann/compendium.ps.Z`.
6. P. Crescenzi, V. Kann, R. Silvestri, and L. Trevisan. Structure in approximation classes. In *Proc. of First Ann. Int. Computing and Comb. Conf.*, to appear, 1995.
7. M. M. Halldórsson. Approximating the minimum maximal independence number. *Inform. Process. Lett.*, 46:169–172, 1993.
8. O. H. Ibarra and C. E. Kim. Fast approximation for the knapsack and sum of subset problems. *J. ACM*, 22(4):463–468, 1975.
9. V. Kann. *On the Approximability of NP-complete Optimization Problems.* PhD thesis, Dep. of Numerical Analysis and Computing Science, KTH, 1992.
10. V. Kann. Polynomially bounded minimization problems that are hard to approximate. *Nordic J. Computing*, 1:317–331, 1994.
11. P. G. Kolaitis and M. N. Thakur. Logical definability of NP optimization problems. *Inform. and Comput.*, 115:321–353, 1994.
12. M. W. Krentel. The complexity of optimization problems. *J. Comput. System Sci.*, 36:490–509, 1988.
13. C. Lund and M. Yannakakis. The approximation of maximum subgraph problems. In *Proc. of 20th International Colloquium on Automata, Languages and Programming*, pages 40–51. Springer-Verlag, 1993. Lecture Notes in Comput. Sci. 700.
14. A. Panconesi and D. Ranjan. Quantifiers and approximation. *Theoretical Comput. Sci.*, 107:145–163, 1993.

Appendix: A List of NPO Problems

Here follows a list of definitions of problems mentioned in the text. A much larger list of NPO problems can be found in [5].

Since it is NP-hard to decide the existence of solutions of all maximization problems below, except LONGEST INDUCED CIRCUIT, we will demand that a trivial solution is included in each problem instance.

Min Independent Dominating Set

Instance: Graph $G = (V, E)$.
Solution: An independent dominating set for G, i.e., a subset $V' \subseteq V$ such that for all $u \in V - V'$ there is a $v \in V'$ for which $(u, v) \in E$, and such that no two nodes in V' are joined by an edge in E.
Measure: Cardinality of the independent dominating set, i.e., $|V'|$.

Max PB 0 − 1 Programming

Instance: Integer $m \times n$-matrix $A \in Z^{m \cdot n}$, integer m-vector $b \in Z^m$, nonnegative binary n-vector $c \in \{0,1\}^n$.
Solution: A binary n-vector $x \in \{0,1\}^n$ such that $Ax \geq b$.
Measure: The scalar product of c and x, i.e., $\sum_{i=1}^{n} c_i x_i$.

Max Number of Satisfiable Formulas

Instance: Set U of variables, collection C of 3CNF formulas.
Solution: A subset $C' \subseteq C$ of the formulas such that there is a truth assignment for U that satisfies every formula in C'.
Measure: Number of satisfied formulas, i.e., $|C'|$.

Max Distinguished Ones

Instance: Disjoint sets X, Z of variables, collection C of disjunctive clauses of at most 3 literals, where a literal is a variable or a negated variable in $X \cup Z$.
Solution: Truth assignment for X and Z that satisfies every clause in C.
Measure: The number of Z variables that are set to true in the assignment.

Max Ones

Instance: Set X of variables, collection C of disjunctive clauses of at most 3 literals where a literal is a variable or a negated variable.
Solution: Truth assignment that satisfies every clause in C.
Measure: The number of variables that are set to true in the assignment.

Max C Bin Sat$^{\mathcal{R}_1;\mathcal{R}_2}$

Instance: $\mathcal{R}_1, \mathcal{R}_2 \in \{=, \geq, >, \neq\}$ defining the types of relations. Systems $A_1 x_1 \mathcal{R}_1$ and $A_2 x_2 \mathcal{R}_2 b_2$ of linear relations, where A_1 and A_2 are integer matrices and b_1 and b_2 are integer vectors.
Solution: Two vectors x_1 and x_2 of binary numbers such that all relations $A_1 x_1 \mathcal{R}_1$ are satisfied.
Measure: The number of relations in $A_2 x_2 \mathcal{R}_2 b_2$ that are satisfied by x_2.

Max Bin Irrelevant Sat$^{\mathcal{R}}$

Instance: $\mathcal{R} \in \{=, \geq, >, \neq\}$ defining the type of relations. System $Ax\mathcal{R}b$ of relations, where A is an integer matrix, and b is an integer vector.
Solution: A vector x of binary numbers such that all relations $Ax\mathcal{R}b$ are satisfied.
Measure: The number of zero elements in x, i.e., $|\{i : x_i = 0\}|$.

Longest Induced Circuit

Instance: Graph $G = \langle V, E \rangle$.
Solution: A subset $V' \subseteq V$ such that the subgraph induced by V' is a circuit.
Measure: Length of the circuit, i.e., $|V'|$.

Some Typical Properties of Large AND/OR Boolean Formulas

Hanno Lefmann and Petr Savický*

Lehrstuhl Informatik II, Universität Dortmund, D-44221 Dortmund, Germany
lefmann@ls2.informatik.uni-dortmund.de
and
Institute of Computer Science, Academy of Sciences of Czech Republic, Prague, Czech Republic, savicky@uivt.cas.cz

Abstract. In this paper typical properties of large random Boolean AND/OR formulas are investigated. Such formulas with n variables are viewed as rooted binary trees chosen from the uniform distribution of all rooted binary trees with m leaves, where n is fixed and m tends to infinity. The leaves are labeled by literals and the inner nodes by the connectives AND/OR, both uniformly at random. In extending the investigation to infinite trees, we obtain a close relation between the formula size complexity of an arbitrary Boolean function f and the probability of its occurrence under this distribution, i.e., the negative logarithm of this probability differs from the formula size complexity of f only by a polynomial factor.

Introduction

this paper, we study Boolean functions determined by large random AND/OR oolean formulas with a given number n of variables. Such formulas are rooted ary trees chosen from the uniform distribution on trees with m leaves, where tends to infinity, labeled by connectives and variables. Each of the $m-1$ inner des has degree two and is labeled by AND or OR with probability 1/2 and pendently of the labeling of all the other nodes. Each leaf is labeled by a ral, a variable or its negation, from the uniform distribution on the $2n$ literals independently of the labeling of all the other nodes.

ppears that, with high probability, the function computed by the large ran- formula is in fact determined only by a small part of it. Using this, for arbitrary Boolean function f, we establish a close relation between its for- size complexity $L(f)$, which is the minimal size of an AND/OR formula essing f, and the limit probability $P(f)$ of the occurrence of f under the ibution described above, when m approaches infinity.

orem 1. *There exist positive constants $c_1, c_2 > 0$, such that for every Boolean tion f of n variables satisfying $L(f) \geq \Omega(n^3)$, the following holds:*

$$e^{-c_1 L(f) \log n} \leq P(f) \leq e^{-c_2 L(f)/n^3}.$$

s research was supported by GA CR, Grant No. 201/95/0976, and by Heinrich- tz-Stiftung while visiting Universität Dortmund, FB Informatik, LS II.