Parallel Approximability of MAXNP Problems

Maria Serna   **Fatos Xhafa**

Departament de Llenguatges i Sistemes Informàtics
Universitat Politècnica de Catalunya

Mòdul C6 - Campus Nord
Jordi Girona Salgado, 1-3
08034-Barcelona, Spain

Technical Report **LSI-95-40-R**

Parallel Approximability of MAXNP Problems*

Maria Serna[†] Fatos Xhafa[†]

Abstract

We study the relationship between the computationally defined class NCX of all optimization problems that are approximable within constant ratio in NC and syntactically defined classes MAXSNP and MAXNP. A simple observation shows that any problem in MAXSNP is constant approximable in NC. We show that MAXNP is also contained in NCX.

1 Introduction

An important result of the theory of approximation is that NP-hard problems behave differently regarding their degree of approximation. Nowadays, we know that there are problems approximable in sequential within any constant factor of the optimum (e.g., KNAPSACK), problems for which there are algorithms that approximate them within some but not all constant factors (e.g., Max CUT) and problems that are not approximable even within a polynomial factor (e.g., Max CLIQUE). As a matter of fact, the first approximation results [Gra66, Joh74, SG76] were of constant ratio. These problems were the first to constitute the class APX of all NPO problems approximable in sequential with relative error bounded by a constant. This is a computationally defined class, in that, in order to show that a certain problem belongs to the class we should exhibit a polynomial time algorithm that finds an approximate solution whose measure is within constant factor of the optimum, or show that it is reducible via an approximation preserving reduction to a problem in the class. Many problems, such as Max CUT, Max 3SAT, have been shown to be complete for this class under PTAS-reductions [ADP80, PM81, ACP94]. On the other hand, Papadimitriou and Yannakakis [PY91] used Fagin's syntactic definition of the class NP and introduced the classes MAXNP and MAXSNP to give answer to the question: is there a subclass of problems in APX which could be proved constant approximable in a generic way. It turned out that

*This research was supported by the ESPRIT BRA Program of the European Community under contract No. 7141, Project ALCOM II.

†Departament de Llenguatges i Sistemes Informàtics, Universitat Politècnica de Catalunya Pau Gargallo 5, 08028-Barcelona, Email: {mjserna,fatos}@goliat.upc.es

this question was closely related to whether several natural problems do not have Polynomial Time Approximation Schemes. Those classes are syntactically defined. Papadimitriou and Yannakakis proved that any problem in MAXSNP/MAXNP class can be approximated in polynomial time with a constant ratio so both MAXSNP and MAXNP are subclasses of APX. In the parallel setting we have an analogous situation. The parallel counterpart of the class APX has been treated in [SX95] where the class of problems that are approximable within constant ratio in NC, denoted NCX, was considered and there were shown also complete problems for this class under a logspace reduction, called NCAS reduction. Here we are interested in studying the approximability of MAXNP class in the parallel setting. The parallel approximability of MAXSNP problems is somehow more straightforward. This can be explained by observing that all the reductions used in [PY91] to prove MAXSNP-complete problems under L-reductions can be done also in logspace. Thus, for example, Max CUT is MAXSNP-complete under L+logspace reduction, and since L-reduction preserve approximability and Max CUT is constant approximable[1] in NC it follows that any problem in MAXSNP is constant approximable in NC. Here we show that the class MAXNP is also a subset of NCX, that is every MAXNP problem has a parallel approximation with a constant ratio. To show this we consider Max GSAT(B) problem. The algorithm used to approximate it in NC with a constant factor goes through a randomized step. First, we show that we can find an approximate solution in RNC whose measure is within a constant factor of the optimum, and then we argue that such a solution can be find also deterministically in NC. This idea is then generalized for any problem in MAXNP. It has been pointed out to us that similar techniques are used also in [CS93, BCCS95].

2 Preliminaries and Notations

When studing the parallel approximability of a problem, a crucial parameter is the approximation ratio achivable by an NC algorithm[2].

Approximation Ratio An algorithm $\mathcal{A}$ has approximation ration r, for a maximization problem (resp. minimization problem), if, for every instance of the problem it always finds a feasible solution of value at least OPT/r (resp. at most $rOPT$), where OPT is the value of the optimal solution.

Syntactic MAXSNP and MAXNP Classes [PY91] are extensions of Fagin's syntactic characterization of NP [Fag74], which states that NP consists of all predicates on structures G that can be expressed in the form $\exists S \phi(G, S)$ where S is also a structure and ϕ is first order predicate. More-

[1]Max CUT is trivially 1/2-approximable in NC.

[2]An NC algorithm finds a solution to a given problem in time polylogarithmic in instance size and uses a polynomial number of processors.

over, ϕ can be written as $\phi = \forall \overline{x}\ \exists \overline{y}\ \psi(\overline{x}, \overline{y}, G, S)$ where ψ is quantifier free. In particular the problems of NP whose predicate ϕ can be expressed as $\phi = \forall \overline{x}\ \psi(\overline{x}, G, S)$ form the subclass SNP (Strict NP).

Syntactic MAXNP consists of problems maxΠ, for $\Pi \in$NP, defined as: find a structure S that maximizes $|\{\overline{x} : \exists \overline{y} \psi(\overline{x}, \overline{y}, G, S)\}|$.
Syntactic MAXSNP consists of problems maxΠ, for $\Pi \in$SNP, defined as: find a structure S that maximizes $|\{\overline{x} : \psi(\overline{x}, G, S)\}|$.

L-reductions [PY91] Given two NPO problems Π and Π', it is said that Π L-reduces to Π', if there exist a pair of functions (f, g) computable in polynomial time and two constants $\alpha, \beta > 0$ such that (a) function f transforms a given instance I of Π into an instance I' of Π' satisfying $\mathrm{OPT}(I') \leq \alpha \mathrm{OPT}(I)$ and (b) function g maps solutions of instance I' of cost C' into solutions of I of cost C in such way that $|C - \mathrm{OPT}(I)| \leq \beta |C' - \mathrm{OPT}(I')|$.

Observe that whenever problem Π L-reduces to problem Π' and Π' is approximable within a constant factor r we have that problem Π is approximable within $\alpha\beta r$. In particular, when functions f, g are computable in logspace we speak of L+logspace reductions.

MAXSNP (resp. **MAXNP**) is the class of all optimization problems that are L-reducible to a problem in the class Syntactic MAXSNP (resp. Syntactic MAXNP).

***k*-wise Independence** Let $X_1, X_2, \ldots, X_n$ be n random variables with values from a set T and a natural k. It is said that they are k-wise independent if any k of them, $X_{j_1}, X_{j_2}, \ldots, X_{j_k}$, are independent. k-wise independence of random variables is useful for constructing small probability spaces with a desired property. More precisely, when k is a constant, $k > 2$, then sample spaces of polynomial size $O(n^k)$ can be constructed in NC (for $k = 2$ the sample space is $O(n)$). Thus RNC algorithms based on k-wise random variables can be derandomized yielding deterministic NC algorithms [Lub86].
We will consider the following two problems through the paper.

Max GSAT(B)
Given a set of clauses C where each clause is the disjunction of conjunctions, with each conjunction containing up to B literals, find a truth assignment to the variables that satisfies the maximum number of clauses.

Max *k*CSP
Given a collection $C_1, \ldots, C_m$ of k-ary constraints over $X = \{x_1, \ldots, x_n\}$ where each C_i has the weight w_i, Max kCSP problem is to find a truth assignment to the variables that maximizes the sum of the weights of satisfied constraints. A k-ary constraint on the set X is a pair $C = (V, P)$, where V is a subset of k variables from X and $P : \{T, F\}^k \rightarrow \{T, F\}$ is a boolean predicate over V.

3 The Relationship Between MAXNP and NCX

In order to relate the classes MAXNP and NCX we will consider Max GSAT(B) problem. The reason for that is twofold. In one hand, the analysis for this problem can easily be extended to any other problem in MAXNP, and, on the other hand, it is known from [PY91] that for any problem in the class MAXNP there is a constant B such that the problem is L-reduced to Max GSAT(B), for that B. Max GSAT(B) problem is in the class MAXNP since it can be written in the form:

$$\exists S\, \forall \overline{x}\, \exists \overline{y}\, \Psi(\overline{x}, \overline{y}, S, C),$$

where S is an assignment to the variables, $\overline{x}$ is any clause and $\overline{y}$ is some subclause of the clause $\overline{x}$ that is satisfied. We prove that Max GSAT(B) belongs to the class NCX. The proof goes through a randomized step. First, we show that we can find an approximate solution in RNC and later we argue that such a solution can be find also deterministically in NC.

Proposition 1 *There is an NC algorithm that for any given instance of Max GSAT(B) finds an assignment to the variables that satisfies at least $1/2^B$ of the clauses, runs in $O(\log n)$ time and uses $O(n^B)$ processors.*

Proof. Let us consider first the case $B = 2$. Let S denote an assignment to the variables and $X(C_i, S)$, $1 \leq i \leq m$ be the variables:

$$X(C_i, S) = \begin{cases} 1 & \text{if } S \text{ satisfies the clause } C_i, \\ 0 & \text{otherwise.} \end{cases}$$

Consider also the function $S \to m(S) = \#$*of satisfied clauses by* S and let us estimate the average value $E[m(S)]$.

$$\begin{aligned} E[m(S)] = \sum_S Pr(S)m(S) &= \sum_S Pr(S) \sum_{i=1}^{m} X(C_i, S) \\ = \sum_{i=1}^{m} \sum_S Pr(S) X(C_i, S) &= \sum_{i=1}^{m} \frac{1}{2^n} \sum_S X(C_i, S). \end{aligned} \qquad (1)$$

Let α be the maximum number of conjunctions of any clause C_i. We can write

$$\sum_S X(C_i, S) \geq \sum_S \frac{1}{\alpha} \sum_{j=1}^{\alpha} X(C_i^j, S) = \frac{1}{\alpha} \sum_{j=1}^{\alpha} \sum_S X(C_i^j, S), \qquad (2)$$

where C_i^j is a subclause of C_i and $X(C_i^j, S)$ are defined in the same way as $X(C_i, S)$. Since C_i^j is a conjunction of two literals, it is satisfied with probability $1/4$ if the values for its literals are chosen independently (and uniformly). Thus, the last term in (2) is $1/4 \cdot 2^n$ and therefore, from (1) and (2) we deduce $E[m(S)] \geq m/4$. This implies that the assignment S is 1/4-approximate because the maximum number of satisfied clauses is at most m. Since we have used only pairwise independence of the variables, an assignment S that satisfy $m(S) \geq m/4$ can be found also deterministically in $O(\log n)$ time and using $O(n)$ processors.

The general case is proved similarly and we will have that $m(S) \geq m/2^B$ which implies the approximation ratio of $1/2^B$. Again, the time bound and the number of processors are obvious (for the last recall that our random variables are B-wise independent). □

Additionally, we observe that the analysis for Max GSAT(B) can be also applied to Max kCSP problem giving a $1/2^k$-approximation.

Corollary 1 *There is an NC algorithm that, given an instance of Max kCSP, finds an assignment to the variables that satisfies at least $1/2^k$ of clauses.*

Finally, notice that the above result works in a straightforward way for the weighted version of the problem.

Turning on the general case, i.e., for any given problem in MAXNP, we can easily derive the parallel constant approximability. Indeed, the analysis done for Max GSAT(B) is valid for any MAXNP problem. It just suffices to have in mind that S is an assignment to the variables, $\overline{x}$ is any clause and $\overline{y}$ is some subclause of the clause $\overline{x}$ that is satisfied (see definition of Max GSAT(B) and that of MAXSNP.) Alternatively, the approximability of other MAXNP problems is implied from the result of Papadimitriou et al. [PY91] that any MAXNP problem is L-reduced to Max GSAT(B), for some constant B, the fact that L-reductions preserve constant approximability and our parallel approximation for Max GSAT(B).

Theorem 1 *Every problem in MAXNP can be approximated in NC within a constant ratio.*

Open Questions

Khanna et al. [KMSV94] have shown that the class APX coincides with the closure under E-reductions of the classes MAXSNP and MAXNP. It would be interesting to investigate whether this result holds also in the parallel case. Their proof uses the machinery of PCP (Probabilistic Checking of Proofs), so in particular a PCP-free proof of the result would be of a special interest.

References

[ACP94] G. Ausiello, P. Crescenzi, and M. Protasi. Approximate Solution of NP Optimization Problems. Technical Report SI/RR - 94/03, Dipartimento di Scienze dell'Informazione. University of Rome, "La Sapienza", 1994.

[ADP80] G. Ausiello, A. D'Atri, and M. Protasi. Structure Preserving Reductions Among Convex Optimization Problems. *Journal of Computer and System Sciences*, 21:136–153, 1980.

[BCCS95] D.P. Bovet, A. Clementi, P. Crescenzi, and R. Silvestri. Parallel Approximation for Optimization Problems. Technical Report SI/RR - 95/09, Dipartimento di Scienze dell'Informazione, University of Rome, "La Sapienza", 1995.

[CS93] P. Crescenzi and R. Silvestri. Average Measure, Descriptive Complexity and Approximation of Maximization Problems. *Int. Journal on Foundations of Computer Science*, 4:15–30, 1993.

[Fag74] R. Fagin. Generalized First-Order Spectra and Polynomial Time Recognizable Sets. In *Complexity of Computations.* (R. Karp Ed.) Amer. Math. Soc., 1974.

[Gra66] R.L. Graham. Bounds for Certain Multiprocessing Anomalies. *Bell System Technology Journal*, 45:1563–11581, 1966.

[Joh74] D.S. Johnson. Approximation Algorithms for Combinatorial Problems. *Journal of Computer and System Sciences*, 9:256–278, 1974.

[KMSV94] S. Khanna, R. Motwani, M. Sudan, and U. Vazirani. On Syntactic versus Computational Views of Approximability. In *Proceedings of 35th IEEE Symposium on Foundations of Computer Science*, pages 819–830, 1994.

[Lub86] M. Luby. A Simple Parallel Algorithm for the Maximal Indipendent Set Problem. *SIAM Journal of Computing*, 15:1036–1053, 1986.

[PM81] A. Paz and S. Moran. Non-Deterministic Polynomial Optimization Problems and their Approximation. *Theoretical Computer Science*, 15:251–277, 1981.

[PY91] C.H. Papadimitriou and M. Yannakakis. Optimization, Approximation, and Complexity Classes. *Computer and System Sciences*, 43:425–440, 1991.

[SG76] S. Sahni and T. Gonzalez. P-complete Approximation Problems. *Journal of ACM*, 23:555–565, 1976.

[SX95] M. Serna and F. Xhafa. On Parallel versus Sequential Approximations. Technical Report LSI-95-18-R, Universitat Politècnica de Catalunya, Dept. de LSI, 1995.