

---

## Notes for Lecture 15

### 1 Hardness of Approximation

We know that a number of important optimization problems are NP-hard to solve exactly. Today we begin the study of the complexity of finding *approximate* solutions.

There is a fundamental difficulty in proving hardness of approximation results. All the NP-completeness proofs for graph problems before 1990 can be essentially described as follows: we start from the computation of a generic non-deterministic Turing machine, then we encode its computation as a 3SAT formula, using the construction of Cook's theorem, and then we reduce 3SAT to the problem of interest (the reduction may be presented as a sequence of reductions involving several intermediate problems, but it can always be thought of as a direct reduction from 3SAT) by encoding variables and clauses of the formula as sub-graphs connected in a proper way. The computation of a Turing machine is very sensitive to small changes, and it seems impossible to generate an inapproximability gap starting from a fragile model and applying "local" reductions. The only inapproximability results that can be proved with such reductions are for problems that remain NP-hard even restricted to instances where the optimum is a small constant. For example, in the Metric Min  $k$ -Center problem it is NP-hard to decide whether the optimum has cost 1 or 2, and so no algorithm can have a performance ratio smaller than 2 unless  $P = NP$  [HS85]. Similarly, in the Coloring problem it is NP-hard to decide whether the optimum has cost 3 or 4, and so no algorithm has performance ratio smaller than  $4/3$  unless  $P = NP$ , and Garey and Johnson [GJ76] show that the gap can be "amplified" to  $k$  versus  $2k - 4$  for constant  $k$ , ruling out also algorithms with performance ratio smaller than 2. Most interesting problems, however, become trivial when restricted to inputs where the optimum is a constant.

To prove more general inapproximability results it seemed necessary to first find a machine model for NP in which accepting computations would be "very far" from rejecting computations. Before such a model was discovered, an important piece of work on inapproximability was due to Papadimitriou and Yannakakis, who showed that, assuming that Max 3SAT does not have a PTAS,<sup>1</sup> then several other problems do not have a PTAS [PY91]. Berman and Schnitger [BS92] proved that if Max 2SAT does not have a PTAS then, for some  $c > 0$ , the Independent Set problem cannot be approximated within a factor  $n^c$ .

The modern study of inapproximability was made possible by the discovery, due to Feige et al. [FGL<sup>+</sup>96] in 1990, that probabilistic proof systems could give a robust model for NP that could be used to prove an inapproximability result for the Independent Set problem.<sup>2</sup> A year later, Arora et al. [AS98, ALM<sup>+</sup>98] proved the PCP Theorem, a very strong characterization of NP in terms of proof systems, and showed how to use the PCP

---

<sup>1</sup>PTAS, short for *Polynomial Time Approximation Scheme*, is an algorithm that given an instance  $x$  and a parameter  $\epsilon$  runs in time polynomial in  $n$ , but with an arbitrary dependency on  $\epsilon$ , and returns a  $(1 + \epsilon)$ -approximate solution.

<sup>2</sup>Another, less influential, connection between probabilistic proof checking and inapproximability was discovered around the same time by Condon [Con93].

Theorem to prove that Max 3SAT does not have a PTAS. Using the reductions of [PY91] (and others [PY93, BP89]), the PCP Theorem gave inapproximability results for several other problems.

## 2 Probabilistically Checkable Proofs

Probabilistically checkable proofs (PCPs) provide a “robust” characterization of the class **NP**. When we reduce a generic **NP** problem  $L$  to 3SAT using Cook’s theorem, we give a way to transform an instance  $x$  into a 3CNF formula  $\varphi_x$  so that if  $x \in L$  then  $\varphi_x$  is satisfiable, and if  $x \notin L$  then  $\varphi_x$  is not satisfiable. Following the proof of Cook’s theorem, however, we see that it is always easy (even when  $x \notin L$ ) to construct an assignment that satisfies all the clauses of  $\varphi_x$  except one.

Using the PCP Theorem one can prove a stronger version of Cook’s theorem, that states that, in the above reduction, if  $x \in L$  then  $\varphi_x$  is satisfiable, and if  $x \notin L$  then there is no assignment that satisfies even a  $1 - \epsilon$  fraction of clauses of  $\varphi_x$ , where  $\epsilon > 0$  is a fixed constant that does not depend on  $x$ . This immediately implies that Max 3SAT does not have a PTAS (unless  $\mathbf{P} = \mathbf{NP}$ ), and that several other problems do not have a PTAS either (unless  $\mathbf{P} = \mathbf{NP}$ ), using the reductions of Papadimitriou and Yannakakis [PY91] and others.

We define PCPs by considering a probabilistic modification of the definition of **NP**. We consider probabilistic polynomial time verifiers  $V$  that are given an input  $x$  and “oracle access” to a witness string  $w$ . We model the fact that  $V$  is a probabilistic algorithm by assuming that  $V$ , besides the input  $x$  and the witness  $w$ , takes an additional input  $R$ , that is a sequence of random bits. Then  $V$  performs a deterministic computation based on  $x$ ,  $w$  and  $R$ . For fixed  $w$  and  $x$ , when we say “ $V^w(x)$  accepts” we mean “the event that  $V$  accepts when given oracle access to witness  $w$ , input  $x$ , and a uniformly distributed random input  $R$ .” When we refer to the “probability that  $V^w(x)$  accepts,” we take the probability over the choices of  $R$ .

We say that a verifier is  $(r(n), q(n))$ -restricted if, for every input  $x$  of length  $n$  and for every  $w$ ,  $V^w(x)$  makes at most  $q(n)$  queries into  $w$  and uses at most  $r(n)$  random bits.

We define the class  $\mathbf{PCP}[r(n), q(n)]$  as follows. A language  $L$  is in  $\mathbf{PCP}[r(n), q(n)]$  if there is an  $(r(n), q(n))$ -restricted verifier  $V$  such that if  $x \in L$ , then there is  $w$  such that  $V^w(x)$  accepts with probability 1 and if  $x \notin L$  then for every  $w$  the probability that  $V^w(x)$  accepts is  $\leq 1/2$ .

We also consider the following more refined notation. We say that a language  $L$  is in  $\mathbf{PCP}_{c(n), s(n)}[r(n), q(n)]$  if there is an  $(r(n), q(n))$ -restricted verifier  $V$  such that if  $x \in L$ , then there is  $w$  such that  $V^w(x)$  accepts with probability at least  $c(n)$ , and if  $x \notin L$  then for every  $w$  the probability that  $V^w(x)$  accepts is at most  $s(n)$ . Of course, the definition makes sense only if  $0 \leq s(n) < c(n) \leq 1$  for every  $n$ . The parameter  $c(n)$  is called the *completeness* of the verifier and the parameter  $s(n)$  is called the *soundness error*, or simply the *soundness* of the verifier.

Note that if  $r(n) = O(\log n)$  then the proof-checking can be *derandomized*, that is,  $V$  can be simulated by a polynomial time deterministic verifier that simulates the computation of  $V$  on each of the  $2^{r(n)} = n^{O(1)}$  possible random inputs and then computes the probability that  $V^w(x)$  accepts, and then accepts if and only if this probability is one. It then follows

that, for example,  $\mathbf{PCP}[O(\log n), O(\log n)] \subseteq \mathbf{NP}$ . The PCP Theorem shows a surprising converse.

**Theorem 1 (PCP Theorem)**  $\mathbf{NP} = \mathbf{PCP}[O(\log n), O(1)]$ .

The theorem was proved in [AS98, ALM<sup>+</sup>98], motivated by a relation between PCP and approximation discovered in [FGL<sup>+</sup>96], and based on much previous work. In the notes for the past lectures we discussed the historical context

### 3 PCP and the Approximability of Constraint Satisfaction Problem

In the Max 3SAT problem we are given a 3CNF boolean formula, that is, a boolean formula in conjunctive normal form (AND-of-OR of literals, where a literal is either a variable or the negation of a variable) such that each term in the conjunction is the OR of at most three literals. The goal is to find an assignment that satisfies the largest possible number of terms.

In the Max qCSP problem, where  $q$  is a positive integer, we are given a system of boolean constraints defined over boolean variables such that every constraint involves at most  $q$  variables. The goal is to find an assignment that satisfies as many constraints as possible. Note that Max 3SAT is a special case of Max 3CSP. (View each clause as a constraint.)

**Theorem 2** *The PCP Theorem implies that there is a constant  $q$  such that there is no  $a$ -approximate algorithm for Max qCSP with  $a < 2$ , unless  $\mathbf{P} = \mathbf{NP}$ .*

PROOF: Let  $L \in \mathbf{PCP}[O(\log n), q]$  be an  $\mathbf{NP}$ -complete problem, where  $q$  is a constant, and let  $V$  be the  $(O(\log n), q)$ -restricted verifier for  $L$ . We describe a reduction from  $L$  to Max qCSP.

Given an instance  $z$  of  $L$ , our plan is to construct a Max qCSP instance  $I_z$  with  $m = |z|^{O(1)}$  constraints such that

$$\begin{aligned} z \in L &\Rightarrow I_z \text{ is satisfiable} \\ z \notin L &\Rightarrow \text{opt}_{\text{Max qCSP}}(I_z) \leq \frac{m}{2} \end{aligned} \tag{1}$$

Once (1) is proved, the theorem follows.

We enumerate all random inputs  $R$  for  $V$ . The length of each string is  $r(|z|) = O(\log |z|)$ , so the number of such strings is polynomial in  $|z|$ . For each  $R$ ,  $V$  chooses  $q$  positions  $i_1^R, \dots, i_q^R$  and a Boolean function  $f_R : \{0, 1\}^q \rightarrow \{0, 1\}$  and accepts iff  $f_R(w_{i_1^R}, \dots, w_{i_q^R}) = 1$ .

We want to simulate the possible computation of the verifier (for different random inputs  $R$  and different witnesses  $w$ ) as a Boolean formula. We introduce Boolean variables  $x_1, \dots, x_\ell$ , where  $\ell$  is the length of the witness  $w$ .

For every  $R$  we add the constraint  $f_R(x_{i_1^R}, \dots, x_{i_q^R}) = 1$ . This completes the description of  $I_z$ .

Now, if  $z \in L$ , then there is a witness  $w$  that is accepted by  $V$  with probability 1. Consider the assignment  $x_i \leftarrow w_i$ , where  $w_i$  is the  $i$ -th bit of  $w$ : then such an assignment satisfies all the constraints of  $I_z$ .

If  $I_z$  has an assignment  $x_i \leftarrow a_i$  that satisfies more than  $m/2$  constraints, then the witness  $w$  defined as  $w_i := a_i$  is accepted with probability more than  $1/2$  by  $V$ , which implies that  $z \in L$ . So if  $z \notin L$  then  $I_z$  has no assignment that satisfies more than  $m/2$  constraints.  $\square$

**Theorem 3** *The PCP Theorem implies that is an  $\epsilon_1 > 0$  such that there is no polynomial time  $(1 + \epsilon_1)$ -approximate algorithm for Max-3SAT, unless  $\mathbf{P} = \mathbf{NP}$ .*

PROOF: Given an instance  $I$  of Max qCSP (where  $q$  is the constant in the PCP Theorem) over variables  $x_1, \dots, x_n$  and with  $m$  constraints, we show how to construct an instance  $\varphi_I$  of Max 3SAT with  $m'$  clauses such that

$$\begin{aligned} I \text{ is satisfiable} &\Rightarrow \varphi_I \text{ is satisfiable} \\ \text{opt}_{\text{Max qCSP}}(I) \leq \frac{m}{2} &\Rightarrow \text{opt}_{\text{Max 3SAT}}(\varphi_I) \leq (1 - \epsilon_1)m' \end{aligned} \quad (2)$$

Once (2) is proved, the theorem follows.

For every constraint  $f(x_{i_1}, \dots, x_{i_q}) = 1$  in  $I$ , we first construct an equivalent qCNF of size  $\leq 2^q$ . Then we “convert” clauses of length  $q$  to clauses length 3, which can be done by introducing additional variables, as in the standard reduction from  $k$ SAT to 3SAT (for example  $x_2 \vee x_{10} \vee x_{11} \vee x_{12}$  becomes  $(x_2 \vee x_{10} \vee y_R) \wedge (\bar{y}_R \vee x_{11} \vee x_{12})$ ). Overall, this transformation creates a formula  $\varphi_I$  with at most  $q2^q$  3CNF clauses for each constraint in  $I$ , so the total number of clauses in  $\varphi_I$  is at most  $q \cdot 2^q \cdot m$ .

Let us now see the relation between the optimum of  $\varphi_z$  as an instance of Max 3SAT and the optimum of  $I$  as an instance of Max qCSP.

If  $I$  is satisfiable, then set the  $x$  variables in  $\varphi_I$  to the same values and set the auxiliary variables appropriately, then the assignment satisfies all clauses, and  $\varphi_I$  is satisfiable.

If every assignment of  $I$  contradicts at least half of the constraints, then consider an arbitrary assignment to the variables of  $\varphi$ ; the restriction of the assignment to the  $x$  variables contradicts at least  $m/2$  constraints of  $I$ , and so at least  $m/2$  of the clauses of  $\varphi_I$  are also contradicted. The number  $m' - m/2$  is at most  $m'(1 - \epsilon_1)$  if we choose  $\epsilon_1 \leq \frac{1}{2q2^q}$ .  $\square$

Interestingly, the converse also holds: any gap-creating reduction from an  $\mathbf{NP}$ -complete problem to Max qCSP implies that the **PCP** Theorem must be true.

**Theorem 4** *If there is a reduction as in (1) for some problem  $L$  in  $\mathbf{NP}$ , then  $L \in \mathbf{PCP}[O(\log n), q]$ . In particular, if  $L$  is  $\mathbf{NP}$ -complete then the **PCP** Theorem holds.*

PROOF: We describe how to construct a verifier for  $L$ .  $V$  on input  $z$  expects  $w$  to be a satisfying assignment for  $I_z$ .  $V$  picks a constraint of at random, and checks that the assignment  $x_i \leftarrow w_i$  satisfies it. The number of random bits used by the verifier is  $\log m = O(\log |z|)$ . The number of bits of the witness that are read by the verifier is  $q$ .

$$\begin{aligned} z \in L &\Rightarrow I_z \text{ is satisfiable} \\ &\Rightarrow \exists w \text{ such that } V^w(z) \text{ always accept.} \end{aligned}$$

$$\begin{aligned} z \notin L &\Rightarrow \forall w \text{ a fraction } \frac{1}{2} \text{ of constraints of } I_z \text{ are unsatisfied by } w \\ &\Rightarrow \forall w V^w(z) \text{ rejects with probability } \geq \frac{1}{2} \end{aligned}$$

$\square$

## References

- [ALM<sup>+</sup>98] S. Arora, C. Lund, R. Motwani, M. Sudan, and M. Szegedy. Proof verification and hardness of approximation problems. *Journal of the ACM*, 45(3):501–555, 1998. Preliminary version in *Proc. of FOCS'92*. 1, 3
- [AS98] S. Arora and S. Safra. Probabilistic checking of proofs: A new characterization of NP. *Journal of the ACM*, 45(1):70–122, 1998. Preliminary version in *Proc. of FOCS'92*. 1, 3
- [BP89] M. Bern and P. Plassmann. The Steiner tree problem with edge lengths 1 and 2. *Information Processing Letters*, 32:171–176, 1989. 1
- [BS92] P. Berman and G. Schnitger. On the complexity of approximating the independent set problem. *Information and Computation*, 96:77–94, 1992. Preliminary version in *Proc. of STACS'89*. 1
- [Con93] A. Condon. The complexity of the max-word problem and the power of one-way interactive proof systems. *Computational Complexity*, 3:292–305, 1993. Preliminary version in *Proc. of STACS91*. 1
- [FGL<sup>+</sup>96] U. Feige, S. Goldwasser, L. Lovász, S. Safra, and M. Szegedy. Interactive proofs and the hardness of approximating cliques. *Journal of the ACM*, 43(2):268–292, 1996. Preliminary version in *Proc. of FOCS91*. 1, 3
- [GJ76] M.R. Garey and D.S. Johnson. The complexity of near-optimal graph coloring. *Journal of the ACM*, 23:43–49, 1976. 1
- [HS85] D.S. Hochbaum and D.B. Shmoys. A best possible heuristic for the  $k$ -center problem. *Mathematics of Operations Research*, 10(2):180–184, 1985. 1
- [PY91] C. H. Papadimitriou and M. Yannakakis. Optimization, approximation, and complexity classes. *Journal of Computer and System Sciences*, 43:425–440, 1991. Preliminary version in *Proc. of STOC'88*. 1, 2
- [PY93] C.H. Papadimitriou and M. Yannakakis. The travelling salesman problem with distances one and two. *Mathematics of Operations Research*, 18:1–11, 1993. 1