T-79.511 Special Course on Cryptology / Zero Knowledge: Comments on "A first look at the $\mathbf{NP} = \mathbf{PCP}(\log n, 1)$ theorem"

Petteri Kaski

October 23, 2001

1 Introduction

The purpose of this document is to give comments and constructive criticism on the manuscript [4] describing the "PCP Theorem" [1, 3], which characterizes $\mathbf{NP}$ as the class of languages $L \subseteq \{0,1\}^*$ whose instances $x \in L$ admit polynomial-length membership proofs $y \in \{0,1\}^*$ verifiable by a probabilistic polynomial time Turing machine using $O(\log |x|)$ random bits and $O(1)$ bits from the proof string y.

First some general comments. The manuscript succeeds well in conveying the key ideas and constructions that underlie the PCP Theorem. The constructions are given in sufficient detail to provide the reader familiar with basic notions of algebra and computational complexity a strong intuition of the theorem. Furthermore, omission of detailed analysis is justified since it would be impossible to present in a short survey. All in all, the manuscript is well-written, although (easily correctable) minor inconsistencies can be found.

The main reference used is a revised version of Sanjeev Arora's thesis [1], which gives a self-contained treatment of the PCP Theorem. The article [3] (see also [2]), although arguably less self-contained, contains a more recent proof of the PCP Theorem, including more accurate definitions in particular for verifier composition and encoding/decoding schemes. (It would be nice to mention this article in the references of the manuscript.)

2 Comments and suggestions on specific topics

Optimality of the characterization. The characterization $\mathbf{NP} = \mathbf{PCP}(\log n, 1)$ is optimal in the sense that if $\mathbf{NP} \subseteq \mathbf{PCP}(o(\log n), o(\log n))$, then $\mathbf{P} = \mathbf{NP}$ [2]. This would be nice to mention in the manuscript.

Furthermore, the PCP Theorem can be stated in a form where the number of queries is a fixed constant for all languages in $\mathbf{NP}$ (assuming a soundness error of at most 1/2) [3, Theorem 1.2.1]. Definitions 1.1 and 1.3 in the manuscript imply only that a language-dependent constant exists.

Gap/promise problems. The definition of MAX-3SAT(ϵ) could be written more clearly. For example: "Call a 3SAT-instance *less than δ-satisfiable* if every truth assignment satisfies less than a δ-fraction of its clauses. The problem MAX-3SAT(ϵ) is to decide the satisfiablity of a 3SAT instance ϕ that is guaranteed to be either (a) satisfiable or (b) less than $(1-\epsilon)$-satisfiable." (In particular, if neither (a) nor (b) holds, then the behaviour of the decision algorithm may be arbitrary.)

Definition 1.2. Exactly one of the occurences of c should be replaced by $1-c$ for the sentence in part (ii) to be consistent with the notation. Also, given that random access is allowed to the proof, the proof string length should probably be explicitly upper bounded by a polynomial in the input length.

Theorem 1.5. *The "only if" direction.* An acceptance probability of less than $1-\epsilon$ in the nonmembership case is sufficient by Definitions 1.2–1.3 to guarantee $L \in \mathbf{PCP}(\log n, 1)$, so why is it necessary to reduce it to 1/2? Also, strictly speaking, the "only if" part of the proof only proves $\mathbf{NP} \subseteq \mathbf{PCP}(\log n, 1)$. (The reverse inclusion naturally holds without the **NP**-hardness assumption, as stated later in Theorem 1.6 of the manuscript.)

The "if" direction. The subscript "c" is unspecified. With Definitions 1.2 and 1.3 it cannot be assumed that at least half of the $\phi_{\psi,r}$ contain a clause that is not satisfied.

Definition 2.1. It would assist the reader if the definition would explicitly state that the size of the "symbol" alphabet is in general a function of the input size, and that the "symbols" can w.l.o.g. be assumed to be encoded by bit strings whose length depends on the symbol alphabet size.

Definition 2.2. The constant c is undefined. The role of integer p is unclear, namely, is it a parameter to the subroutine or a fixed constant. The expression "if all the z_i's are codewords" should be replaced by "if all the z_i's are $\delta/3$-close" for consistency.

Coding-theoretic definitions. Does the constant n denote the input size in "$C \subset A^n$"? The "goodness" of a code is defined somewhat inaccurately.[1]

Lemma 2.4. The reviewer would like to suggest [2, 3] as an alternative references for the proof of the Composition Theorem.

The proof sketch could indicate explicitly that the $(Q+1)$-part encoded proof verification capability of V_2 is used. Also, the $y_{Q+1}(\gamma) = \pi_\gamma$ selection could be justified in more detail; cf. [1, p. 22] or [3].

Identification of $\{0, \ldots, N\}$ with a subset of $\mathbb{F}_q^m$. The word "isomorphism" should probably be replaced by "bijection" in the context $\xi : \{0, \ldots, N\} \to \{0, \ldots, h\}^m$?

Low degree test and interpolation. The term "hard case" is ambiguous. The interpolation algorithm is somewhat inaccurate; if $\kappa : \mathbb{F}_q \to \mathbb{F}_q^m$, then evidently $z_{s+1} \in_U \mathbb{F}_q^m$. Moreover, it is not immediately clear (at least to the reviewer) that Procedure 4.2-(ii) in [1, p. 56] (upon which the reviewer assumes the interpolation algorithm to be based) would correspond to the interpolation algorithm presented in the manuscript. In particular, Procedure 4.2-(ii) makes a random selection from the q^m curves in $P(\langle z_1, \ldots, z_s \rangle)$, the set of degree s curves in $\mathbb{F}_q^m$ whose first[2] s points are $z_1, \ldots, z_s$. Thus, it is the reviewer's understanding that the values $y_1, \ldots, y_s$ selected uniformly at random from $\mathbb{F}_q$ in the manuscript are in fixed in Procedure 4.2-(ii). Whether this affects the correctness of the algorithm remains open.

Arithmetization of a 3SAT-instance. The definition of χ_j should read "$\chi_j(c, v) = 1$ if v is the jth variable of the clause c from ϕ and $\chi_j(c, v) = 0$ otherwise." Also, it would clarify the presentation if it were explicitly indicated that

$$\prod_{j=1}^{3} \chi_j(c, v_j)(s_j(c) - f(v_j)) = 0$$

is an equation system with one equation for each choice $c, v_1, v_2, v_3 \in \{0, \ldots, h\}^m$. Furthermore, $g_f \equiv 0$ holds if and only if f encodes a satisfying truth assignment for ϕ.

3 Suggestions for discussion

Number of proof queries required to recognize NP. What is the minimum number of proof queries sufficient to recognize all languages in **NP**? Is e.g. one bit of the proof sufficient to guarantee a soundness error of at most 1/2?

Nonuniformity. How is the class $\mathbf{PCP}(r, q)$ affected if we allow the verifier to be a nonuniform (probabilistic polynomial-time) Turing machine?

Applicability of the techniques. Are the techniques applied in the proof of the PCP theorem also applicable in other contexts? For example, what other well-known complexity classes besides **NP** can be characterized using the PCP formalism by varying the random string and the number of queries allowed? (See e.g. [3] and the references therein for a partial answer; are there more recent results?)

References

[1] S. Arora. *Probabilistic Checking of Proofs and Hardness of Approximation Problems.*

[1] It is impossible that there exist two distinct codewords that have (relative) distance less than $\delta/2$ to any fixed noncodeword. (If this were possible, then an application of the triangle inequality would produce a contradiction to δ being the (relative) minimum distance of the code.) So, unique decoding is *always* guaranteed if the noncodeword is $\delta/2$-close (to a codeword).

[2] With respect to a "canonical" linear order on $\mathbb{F}_q$.

Technical Report CS-TR-476-94, Department of Computer Science, Princeton University, 1994. (Revised version of a dissertation submitted at CS Division, UC Berkeley, August 1994.)

[2] S. Arora and S. Safra. Probabilistic Checking of Proofs: A New Characterization of NP. *J. ACM* **45** (1998) no. 1, 70–122.

[3] S. Arora, C. Lund, R. Motwani, M. Sudan and M. Szegedy. Proof Verification and the Hardness of Approximation Problems. *J. ACM* **45** (1998) no. 3, 501–555.

[4] M. Kojo. T-79.511 Special Course on Cryptology / Zero Knowledge: A first look at the NP = $\mathrm{PCP}(\log n, 1)$ theorem. October 19, 2001.