On the possibility of basing Cryptography on the assumption that $\mathcal{P} \neq \mathcal{NP}$

Oded Goldreich[*]
Department of Computer Science
Weizmann Institute of Science
Rehovot, ISRAEL.
oded@wisdom.weizmann.ac.il

Shafi Goldwasser[†]
Laboratory for Computer Science
Mass. Institute of Technology
Cambridge, MA02139.
shafi@theory.lcs.mit.edu

February 1998

Abstract

Recent works by Ajtai and by Ajtai and Dwork bring to light the old (general) question of whether it is at all possible to base the security of cryptosystems on the assumption that $\mathcal{P} \neq \mathcal{NP}$. We discuss this question and in particular review and extend a two-decade old result of Brassard regarding this question. Our conclusion is that the question remains open.

Keywords: Cryptography, $\mathcal{P} \neq \mathcal{NP}$, promise problems, smart reductions.

[*]Work done while visiting LCS, MIT.

[†]DARPA grant DABT63-96-C-0018.

1 Introduction

An old folklore rooted in Brassard's paper [7] states that "cryptography" cannot be based on NP-hard problems. However, what Brassard has actually showed [7, Thm. 2, Item (2)ii] can be stated as follows

Brassard's Claim: *Consider a public-key encryption scheme with a* deterministic *encryption algorithm, and suppose that the set of valid public-keys is in* co$\mathcal{NP}$*. Then if retrieving the plaintext from the* (ciphertext, public-key) *pair is NP-Hard then* $\mathcal{NP} = \mathrm{co}\mathcal{NP}$*.*

Our concern in this note is with the restricting preconditions of the above claim.[1] Namely, the encryption algorithm is postulated to be deterministic and the set of valid public-keys for it forms a co$\mathcal{NP}$-set. These preconditions are satisfied in certain encryption schemes, and in particular in the schemes known at the time the claim was made (e.g., plain RSA), but are *not* satisfied in probabilistic encryption schemes such as the Goldwasser–Micali scheme [12] and the Blum–Goldwasser scheme [5] (as well as in the recent "lattice-based" schemes of [3, 11]). We mention that probabilistic encryption is essential to security as defined in [12].

Thus, Brassard's Claim does not rule out the possibility of "basing cryptography" (or even public-key encryption) on the assumption that $\mathcal{P} \neq \mathcal{NP}$ (even if $\mathcal{NP} \neq \mathrm{co}\mathcal{NP}$, as we do believe). Consequently, the following is an important open problem.

Open Problem: *Can one construct a secure encryption scheme based on the assumption that* $\mathcal{P} \neq \mathcal{NP}$*?*

This question is experiencing a rebirth in light of recent attempts to proceed towards this goal. In a pioneering work [1], Ajtai has constructed a one-way function assuming that the Shortest Vector Problem is hard to approximate to within a factor of n^c (in worst case), where $c > 11$ and n denotes the dimention.[2] The fundamental aspect of Ajtai's work, is the reduction of a (non-parametrized) worst-case problem to an average-case one. Consequently, Ajtai and Dwork [3] proposed a public-key encryption scheme whose security is reduced to the Unique Shortest Vector Problem, where uniqueness is again upto a large polynomail in the dimention. Interestingly, the trapdoor permutation suggested in [11] relies on the conjectured difficulty of the Closest Vector Problem. All these are relevant to the above open problem since the Closest Vector Problem is known to be NP-hard to approximate to within any constant factor and is quasi-NP-hard to approximate to within a $2^{\log^{0.999} n}$ factor [4]. Furthermore, the Shortest Vector Problem has been recently shown by Ajtai [2] to be NP-Hard (under randomized reductions). Even more recently, Micciancio [14] has proven that it is NP-Hard (again under randomized reductions) to approximate the Shortest Vector Problem to within any constant factor smaller than $\sqrt{2}$. The approximation factors mentioned in the above two types of results are very far apart, and our own work [10] points out difficulties in trying to bridge the gap. Still, the above effords renew the interest in the Open Problem (as a negative answer to the latter deems these efforts to be futile).

In this note we present some extensions of Brassard's Claim. On one hand, these extensions do cover some probabilistic encryption public-key encryption schemes (such as the Goldwasser-Micali scheme [12] and the Blum–Goldwasser scheme [5]). But, on the other hand, these extensions fall very short of providing an answer to the above Open Problem.

[1] In our discussion, we ignore the known fact that worst-case hardness of retrieving the plaintext is an inadequate (i.e., much too weak) notion of security of encryption schemes.

[2] The constant has been recently reduced to $c > 5$ by Cai and Nerurkar [8].

2 Background — Promise problems and smart reductions

A promise problem [9] is a pair of disjoint subsets of $\{0,1\}^*$. The first subset represents YES-instances, the second NO-instances, and their union is called the promise. Thus, the standard decision problem for a language $L \subseteq \{0,1\}^*$ can be casted as a promise problem $(L, \overline{L})$.

To simplify the discussion we extend the definition of standard complexity classes to promise problem. For example, a promise problem $\Pi = (\Pi_{\mathrm{YES}}, \Pi_{\mathrm{NO}})$ is said to be in $\mathcal{NP}$ if there exists a polynomial-time recognizable (witness) relation R so that

- For every $x \in \Pi_{\mathrm{YES}}$ there exists a $y \in \{0,1\}^*$ such that $(x,y) \in R$ (and $|y| = \mathrm{poly}(|x|)$).
- For every $x \in \Pi_{\mathrm{NO}}$ and every $y \in \{0,1\}^*$, $(x,y) \notin R$.

As explained in [9] (see also [10]), the fact that a promise problem in $\mathcal{NP} \cap \mathrm{co}\mathcal{NP}$ (resp., $\mathcal{AM} \cap \mathrm{co}\mathcal{AM}$) is NP-hard *via arbitrary Cook reductions* does not seem to imply that $\mathcal{NP} = \mathrm{co}\mathcal{NP}$ (resp., $\mathrm{co}\mathcal{NP} \subseteq \mathcal{AM}$). However, such a conclusion does hold in case NP-hardness is proven by a restricted type of Cook-reductions, called *smart reductions*, defined by Grollmann and Selman.

Definition 1 (smart reduction [13]): *A* smart reduction *of a promise problem A to a promise problem B is a polynomial-time* (possibly randomized) *Cook-reduction that on input which satisfies the promise of A only makes queries which satisfy the promise of B. Otherwise the reduction is called* non-smart.[3]

We note that any many-to-one/Karp (possibly randomized) reduction is smart. On the other hand, Even *et. al.* used a non-smart reduction when presenting an NP-hard promise problem in $\mathcal{NP} \cap \mathrm{co}\mathcal{NP}$ (cf. [9, Thm. 4]). Their usage of a non-smart reduction seems essential in light of the result of Grollmann and Selman proved [13, Thm. 2] reproduced next.

Theorem 2 [13, Thm. 2]: *Suppose that a $\mathcal{NP}$-complete language has a deterministic smart reduction to a promise problem in $\mathcal{NP} \cap \mathrm{co}\mathcal{NP}$. Then $\mathcal{NP} = \mathrm{co}\mathcal{NP}$.*

Proof: Given any $\mathrm{co}\mathcal{NP}$-language L, we use the smart (deterministic) reduction to the promise problem Π in order to construct an NP-proof system for L. The NP-witness corresponding to an input $x \in L$, is an augmented transcript of an accepting computation of the reduction (i.e., the oracle-machine). The transcript includes queries to the Π-oracle and presumed answers of this oracle, and is augmented by NP-witnesses to the correctness of the answers provided. These NP-witnesses exist for both YES and NO-instances of Π, since $\Pi \in \mathcal{NP} \cap \mathrm{co}\mathcal{NP}$. Here is where we use the hypothesis that the reduction is smart – this hypothesis guarantees that all queries satisfy the promise (and so have NP-witnesses). ■

3 Extending Brassard's Claim

Our extension of Brassard's claim, allows a probabilistic encryption algorithm and make no requirement on the set of public-keys. We first assume that the encryption algorithm allows errorless decryption. Furthermore, the following proposition refers only to deterministic reductions (and can be easily extended to randomized ones with the weaker conclusion of $\mathrm{co}\mathcal{NP} \subseteq \mathcal{AM}$).

[3]Unfortunately, the term "non-smart" is somewhat misleading – to be non-smart (in an essential way) and yet work the reduction must be quite "clever". A term like "safe" or "honest" may have been more suitable than smart; however "honest" is taken and using "safe" may be confusing when talking about cryptography.

Proposition 3 *Let E be a* (probabilistic) *encryption algorithm for a public-key encryption scheme, and suppose that for every public-key, e, the set of possible encryptions of 0 is disjoint from the set of the possible encryptions of 1. Then $\mathcal{NP} = \text{co}\mathcal{NP}$ if any of the following two holds:*

1. *The following promise problem is NP-hard via* smart *reductions:*

   > *The* YES-*instances are pairs (e, c) where c is in the support of $E_e(0)$, and the* NO-*instances are pairs (e, c) where c is in the support of $E_e(1)$.*

   where $E_e(\sigma)$ is a random variable representing the output of the encryption algorithm E when given the message σ and using e as the encryption-key.

2. *The above promise problem is NP-Hard and the promise is in* $\text{co}\mathcal{NP}$ (i.e., the set of pairs (e, c) where c is neither in the support of $E_e(0)$ nor in the support of $E_e(1)$ is in $\mathcal{NP}$).

The condition in Item 2 relaxes the condition in Brassard's Claim. The condition in Item 1 makes further relaxation but also imposes a (quite limiting) restriction on the reduction. The fact that Item 2 implies $\mathcal{NP} = \text{co}\mathcal{NP}$ is actually a special case of [9, Thm. 6]. (We stress again that none of the items provides an adequate notion of security, as both refer to the worst-case complexity of distinguishing $E_e(0)$ from $E_e(1)$, rather to an average case complexity.)

Proof: In Item 1 we follow the structure of the proof of Theorem 2. Relying on the hypothesis that the reduction is smart and that the supports of $E_e(0)$ and $E_e(1)$ are disjoint, we can prove the validity of each oracle answer by supplying an NP-witness (i.e., the randomness used by the encryption algorithm). In Item 2 we use the hypothesis that the promise is in $\text{co}\mathcal{NP}$ in order to single out queries which violate the promise and prove our claim by supplying an NP-witness. Finally, we note that all these NP-proofs can be concatenated into a single NP-proof and so $L \in \mathcal{NP}$ (where L is an arbitrary $\text{co}\mathcal{NP}$ languages to which the reduction is applied). ■

Next, we consider encryption schemes which may err. Namely, with some bounded probability a valid ciphertext could be generated which could be decrypted both as a 0 and as a 1. We assume, however, that the encryption scheme errs in a way that the receiver may detect that an error in decoding may occurred (i.e., that the received ciphertext is a valid encryption of two different messages).

Proposition 4 *Let E be a* (probabilistic) *encryption algorithm for a public-key encryption scheme. Then $\text{co}\mathcal{NP} \subseteq \mathcal{AM}$, if either of the following two conditions hold:*

1. *The following problem is NP-hard via smart reductions*

   > *The* YES *instances are pairs (e, c) where c is in the support of $E_e(0)$ but not in the support of $E_e(1)$, and the* NO *instances are pairs (e, c) where c is in the support of $E_e(1)$ but not in the support of $E_e(0)$.*

2. *The above promise problem is NP-Hard, and the set of pairs (e, c) where c is in the support of either $E_e(0)$ or $E_e(1)$* (or both) *is in $\text{co}\mathcal{NP}$.*

Proof: We follow the structure of the proof of Theorem 3. Specifically, in Item 1 we are essentially in the same situation as in Item 1 of Theorem 3. For Item 2 we merely need to show that the set of pairs violating the promise is in $\mathcal{NP}$. But this is easy as this set is the union of two NP-sets: (1) the set of pairs being a valid encryption of both 0 and 1 (by supplying the coins used in encryption); and (2) the set of pairs not being a valid encryption of either bits (by hypothesis). ■

Final warning: It seems that many proofs of security are established via reductions which are not smart (e.g., the reduction to distinguishing encryptions of different messages may produce strings which are not encryptions of any message).

References

[1] M. Ajtai. Generating Hard Instances of Lattice Problems. In *28th STOC*, pages 99–108, 1996.

[2] M. Ajtai. The Shortest Vector Problem in L_2 is NP-Hard for Randomized Reductions. Unpublished manuscript, May 1997.

[3] M. Ajtai and C. Dwork. A Public-Key Cryptosystem with Worst-Case/Average-Case Equivalence, In *29th STOC*, pages 284–293, 1997.

[4] S. Arora, L. Babai, J. Stern and Z. Sweedyk. The hardness of approximate optima in lattices, codes, and systems of linear equations. *Journal of Computer and System Sciences*, Vol. 54, pages 317–331, 1997.

[5] M. Blum and S. Goldwasser. An Efficient Probabilistic Public-Key Encryption Scheme which hides all partial information. In *Crypto84*, LNCS (196) Springer-Verlag, pages 289–302.

[6] R. Boppana, J. Håstad, and S. Zachos. Does Co-NP Have Short Interactive Proofs? *IPL*, 25, May 1987, pp. 127-132.

[7] G. Brassard. Relativized Cryptography. In *20th FOCS*, pages 383–391, 1979.

[8] J. Cai and A.P. Nerurkar. An improved Worst-Case to Average-Case connection for lattice problems. In *38th FOCS*, pages 468–477, 1997.

[9] S. Even, A.L. Selman, and Y. Yacobi. The Complexity of Promise Problems with Applications to Public-Key Cryptography. *Inform. and Control*, Vol. 61, pp. 159–173, 1984.

[10] O. Goldreich and S. Goldwasser. On the Limits of Non-Approximability of Lattice Problems. In *30th STOC*, to appear (1998).

[11] O. Goldreich, S. Goldwasser and S. Halevi. Public-Key Cryptosystems from Lattice Reduction Problems. In *Crypto97*, Springer LNCS, Vol. 1294, pp. 112–131.

[12] S. Goldwasser and S. Micali. Probabilistic Encryption. *JCSS*, Vol. 28, No. 2, pages 270–299, 1984.

[13] J. Grollmann and A.L. Selman. Complexity Measures for Public-Key Cryptosystems. *SIAM J. Comput.*, Vol. 17, No. 2, pages 309–335, 1988.

[14] D. Micciancio. On the Inapproximability of the Shortest Vector in a Lattice within some constant factor. Preliminary version MIT/LCS/TM-574, February 1998.