Secure Commitment Against A Powerful Adversary

A security primitive based on average intractability

(EXTENDED ABSTRACT) *

Rafail Ostrovsky† Ramarathnam Venkatesan‡ Moti Yung§

Abstract

Secure commitment is a primitive enabling information hiding, which is one of the most basic tools in cryptography. Specifically, it is a two-party partial-information game between a "committer" and a "receiver", in which a secure envelope is first implemented and later opened. The committer has a bit in mind which he commits to by putting it in a "secure envelope". The receiver cannot guess what the value is until the opening stage and the committer can not change his mind once committed.

In this paper, we investigate the feasibility of bit commitment when one of the participants (either committer or receiver) has an unfair computational advantage. That is, we consider commitment to a *strong receiver* with a large computational power (requiring that despite his power he can not "open" the secret commitment) or commitment by a *strong committer* (requiring that despite his power he can not change the value of the committed bit). We allow the strong party to use its computational resources and investigate the underlying complexity assumptions necessary for the feasibility of these primitives.

We show how to base commitment by a *strong committer* on *any* hard on the average problem. In fact, this is the first application of average case completeness to hiding information in a security primitive. We also show how commitment to a *strong receiver* with *information theoretic security* can be implemented based on *any* one-way function.

In addition, we show that commitment to a strong receiver is *complete* for *all* partial information games between weak and strong players. That is, given any implementation of the commitment protocol to a strong receiver, any partial-information game between a weak and a strong player can be implemented based solely on such a protocol.

* To appear in Symposium on Theoretical Aspects of Computer Science (STACS) 92, February 13-15, Paris, France.

† MIT Laboratory for Computer Science, 545 Technology Square, Cambridge MA 02139, USA. Supported by IBM Graduate Fellowship. Part of this work done while at IBM T.J. Watson Research Center.

‡ Bell Communications Research, 2M-344, 445 South St, Morristown NJ 07960, USA. Part of the work done at Boston University supported by NSF-CCR9015276.

§ IBM Research, T.J. Watson Research Center, Yorktown Heights, NY 10598 USA.

1 Introduction

Secure protocols can be viewed as partial information games among mutually distrustful players (see, e.g., [GMW2, Co]). Many of these games can be based on a very simple game, called *bit-commitment* (BC) (see, e.g., [B1, B2, BM, BCC, BCY, BMO, EGL, GMW1, IY, SRA]). Here, we investigate the interplay between the computational power of the players in the commitment protocol and the complexity assumptions needed for its feasibility. A *strong* player has unlimited computing power; we often specify the exact needed power. A *weak* player is limited to polynomial time computations.

Different computational resources of the participants imply different notions of the security of the commitment. We say that bit commitment protocol is *computationally secure* if polynomially bounded receiver can not deduce the value of the committed bit before the reveal stage, however if receiver is given sufficient computational resources, he can discover the value of the committed bit. In contrast, we say that bit commitment protocol is *information-theoretically secure* if even with infinite resources, receiver can not gain any information about the bit before the reveal stage.

For commitment to a weak player, earlier Naor [N] exhibited a computationally secure bit-commitment protocol using any one-way function; when both players are weak (called the *symmetric* case), this is the best possible since such a protocol implies a one-way function [ImLu]. For the strong committer case, we relax this assumption much further, by basing it on any hard-on-average problems in PSPACE. This is the first application of Levin's theory of average case completeness to playing partial-information games. In fact, let C be any class inside PSPACE with a complete problem which (1) has an interactive proof whose prover is also in C (2) is hard-on-average. Then, assuming (1) and (2) the above (i.e. computationally secure) bit commitment protocol could be implemented from committer in C to receiver for whom complexity class C is hard on the average.

In the opposite direction, (i.e. for the commitment to a strong player) the goal is to construct an *information-theoretically* secure bit commitment protocol. (That is, to prevent the strong receiver from gaining any information about the committed secret despite his superior resources.) Previous implementations used a trapdoor permutation [GN], or a variety of specific algebraic assumptions, (e.g. [B2, BCY, BMO]). We improve this to *any* one-way function.

To get the later result we use another security primitive, the Oblivious Transfer (OT) protocol, introduced by Rabin [R]. This is a protocol by which one party sends a bit to a receiver, the bit gets there with probability 1/2 and the sender does not know the result of the transfer. We first show that the existence of the following three protocols is equivalent:

1. *BC* from weak to strong
2. *OT* from weak to strong
3. *OT* from strong to weak

That is, given an implementation of any one of these three protocols, we show how to implement the others *without any additional assumptions*. Thus, bit-commitment from a weak to strong player is "as hard" as any other protocol between weak and strong player (since OT is complete [K]). The corresponding result for the symmetric case is unknown and is unlikely to be proven using "black box" reductions [IR]. Finally, we use the above reduction and our recent result that "OT from weak to strong" can be based on any one-way function [OVY]) to get the bit commitment to a strong receiver based on *any* one-way function.

1.1 Preliminaries

The model we consider for two-party protocols is the standard system of communicating probabilistic machines [GMR]. In this section, we describe a few disclosure primitives and relations among them.

We start with an informal definition of Bit-commitment: BC may be thought of as a way for player S (the Sender) to commit a bit b to player R (the Receiver) in such a way that the bit may be revealed to R at a later point in time. Before b is revealed (but even after b has been committed), no information about b is revealed to R. When b is revealed, it is guaranteed to be the same as the value to which it was originally committed.

Oblivious Transfer (OT) is a two-party protocol introduced by Rabin [R]. Rabin's OT assumes that S possesses a value x, after the transfer R gets x with probability $\frac{1}{2}$ and it knows whether or not it got it (*equal-opportunity requirement*). A does not know whether B got the value (*oblivious-ness requirement*). A similar notion of 1-2-OT (one out of two OT) was introduced by [EGL]. In 1-2-OT, player S has two bits b_0 and b_1 and R has a selection bit i. After the transfer, R gets only b_i, while S does not know the value of i. Equivalently, R may get a random bit in $\{b_0, b_1\}$, or the game can be played on strings rather then bits. Further, there are many other flavors of OT [C, BCR, K, CK] all of which are information-theoretically equivalent. That is, given any one of these protocols, one can implement the other ones. Thus, by "OT" we can refer to any one of them.

The following notations will be used. By $(\texttt{weak}\xrightarrow{\text{BC}}\texttt{strong})$, we denote BC from a polynomially-bounded player to an infinitely-powerful one. We use $(\texttt{strong}\xrightarrow{\text{BC}}\texttt{weak})$, $(\texttt{strong}\xrightarrow{\text{OT}}\texttt{weak})$, $(\texttt{weak}\xrightarrow{\text{OT}}\texttt{strong})$ with similar meanings.

We must stress, that our results hold for the insecure communication environment. This should be contrasted with the work of [BGW, CCD, RB, BG, K, CK] where they assume right from the start that some form of OT already exists, or that secure channels exist. Instead, we concentrate on the two party scenario where secure channels do not help and investigate the required complexity assumptions for achieving BC.

1.2 Previous and related work

Our main primitive is BC, used as a basic building block in many different settings [B1, B2, BM, BCC, BCY, BMO, GMW1, K, N, Ost, SRA]. As was noted earlier, in the symmetric case BC and one-way functions are equivalent [BM, ILL, H, N]. We consider any hard on average problems (in PSPACE) as a base for the BC primitive.

The second primitive we apply is Oblivious Transfer. Rabin [R] defined and implemented OT for honest parties based on the intractability of factoring; Fischer, Micali and Rackoff [FMR] improved this result to be robust against cheaters. Other variations of OT were studied and shown to be information theoretically equivalent. Yao [Y] used OT (based on factoring) to construct secure circuit evaluation. Goldreich, Micali and Wigderson [GMW2] based OT for symmetric case (which also extends the asymmetric case of $(\texttt{strong}\xrightarrow{\text{OT}}\texttt{weak})$) on the existence of any trapdoor permutation, and used it for multiparty circuit evaluation. Thus, secure circuit evaluation for poly-bounded players was made possible, assuming one-way trapdoor permutations exists. OT was also shown to be complete for secure circuit evaluation [K]. OT was also used to implement non-interactive and bounded-interaction zero-knowledge proof systems for NP [KMO]. This paper investigates the connection of asymmetric OT and asymmetric BC.

Since we deal with an asymmetric two-party model, let us point out what was considered in this model in addition to zero-knowledge proof systems of Goldwasser, Micali and Rackoff [GMR]. Note that this model represents naturally interaction between a small user and an all-powerful organization which may possess very large computational power. One such case is the context of zero-knowledge arguments of Brassard, Crépeau and Chaum [BCC], which assume an all-powerful verifier from which information has to be hidden. (Here we note that their protocols can be executed by polynomial time parties with cryptographic applications in mind while our results concentrate on allowing one party to have infinite power and use it in the computation. Recently, investigating the symmetric case, new results which reduce complexity assumptions in the practical context of [BCC] were also achieved [NOVY].) Another setting similar to ours is the model of using a powerful oracle to compute a value while keeping the real argument secret, [AFK, FO] where the oracle indeed uses its power.

2 Bit-commitment from strong to weak

In a $\texttt{strong}\xrightarrow{\text{BC}}\texttt{weak}$ protocol, if an infinitely-powerful "committer" (or Sender) tries to cheat by changing the value of the committed bit, the probabilistic polynomial-time "receiver" can catch this with overwhelming probability (over receiver's coin tosses). The actual work to be performed by the sender to execute the protocol is stated in the theorems below. Of course, if the receiver breaks the assumption, the value of the committed bit will be available before decomittal.

We first give a bit-commitment protocol based on an *average case complete* [L, VL, G, ImLe] problem. Randomized NP (RNP) consists of problems from NP under samplable distributions. For convenience we fix one such problem, namely Graph Coloration Problem (GCP) (see below). If there is any NP problem which is hard on average under any samplable (i.e., generatable in polynomial time) distribution, then so is this complete problem under random inputs. Thus, if a one-way function exists, then this complete problem is hard-on-average but the reverse implication that some complete (and thus hard-on-average) problem implies a one-way functions is open.

Let x be generated according to a distribution μ. An algorithm $A(x)$ is polynomial on average if it runs in time $(|x|r(x))^{O(1)}$, where $\mathbf{E}_\mu r(x) < 1$. Intuitively, $r(x)$ is a randomness test that takes small values on "typical" strings and large values on "rare" or "atypical" x. So, A can run longer on some rare inputs. Also, ignoring polynomial (in k) factors, an algorithm can take $2^{O(k)}$ time, with probability (over inputs) at most 2^{-k}. Let AP be the class of NP problems under samplable distributions which can be solved in polynomial on average time. A problem under a distribution μ is called *hard-on-average* if it is not in AP. In general, we may consider any complexity class instead of NP for defining AP. It is not hard to show (See the Corollary in [L] and [VL] for discussions) that a hard-on-average problem yields a problem with polynomial fraction of hard instances.

Lemma 1 *Unless $RNP = AP$, there is a protocol for committing a bit by a strong sender to a weak receiver, where the Sender needs only be a $(NP \cup co-NP)$ machine.*

Proof (Sketch): The following can be deduced from [N, GL]:

[N]: Assume there is pseudo-random generator (unpredictable for the receiver) that can be computed by the committer and which can be checked (given its seed) by the receiver. Then, there is a bit-commitment protocol from the committer to receiver.

[GL]: (List Decoding)Let $f(x) = y$ be polynomial time computable. Let $G(y, r) \in \{\pm 1\}$ be an algorithm that predicts the inner product $b(x, r)$ with a correlation $\mathbf{E}_r G(y, r)(-1)^{b(x,r)} = \varepsilon$. Then, there is an algorithm $A(y)$ that in $1/\varepsilon^{O(1)}$ time outputs a list L containing $1/\varepsilon^2$ strings such that $x \in L$.

Thus, if $|y| = n$ and $b(x, r)$ can be predicted with probability (over r) $1/2 \pm 1/n^c$, x can be computed in $n^{O(1)}$ time. Notice the absence of samplability requirement over x. This yields a hard-core bit based on a hard-on-average problem. Let f be the function checking the relation GCP which takes a edge-colored (with 4 colors) digraph and outputs the uncolored digraph, the number of edges of each color, and the list of all 3-node induced colored subgraphs with nodes relabeled 1,2,3 ; then $b(x, r)$ is hard-to-predict from y, r unless $RNP = AP$. Now, using the constructions of [H, ILL] the committer can generate pseudo-random bits. $\Box$.

Next we show the optimal conditions for commitment from strong to weak.

Theorem 1 *There exists a bit-commitment protocol from an infinitely-powerful sender to a weak receiver, based on any complete problem for any complexity class in PSPACE which is hard on the average.*

The proof has two steps described in the following proposition and lemma: first, we exhibit a complete problem in $RPSPACE$, second, we use analogous construction to Lemma 1, basing a generator on this complete problem. We also argue that this is the hardest language to base commitment on.

Let u be a machine with some fixed polynomial space bound, where $u(p, x, b) = (p, x)$ if the program p accepts x and $b = 1$ or p rejects x and $b = 0$. Otherwise $u(p, x, b) = 0000...00$. The problem of inverting u on an arbitrary input is equivalent to the halting problem for $PSPACE$. Let (μ, u) be the problem of inverting u when its inputs are randomly distributed under the distribution μ. By $RPSPACE$ we mean the class of all such pairs (μ, R) where $\mu \in P$ and $R \in PSPACE$. We define completeness similar to as in [L]. Let λ be the uniform distribution over all strings with $x \in \{0,1\}^n$, $\lambda(\{x\}) = \frac{2^{-n}}{n(n+1)}$.

Proposition 1 *(λ, u) is complete for $RPSPACE$.*

Proof (Sketch): Given an instance x of a problem (μ, R), the reduction in [L] produces an instance y for (λ, u). In our case u runs in polynomial space. $\Box$.

That is, (λ, u) is hard on the average unless every problem in $PSPACE$ under every polynomial time computable distribution has a polynomial on average algorithm. Note that this is weaker than the assertion that for example, Graph Coloration is hard-on average.

Let $\bar{x} = x_1 \circ x_2 \circ \cdots x_k$, $\bar{p} = p_1 \circ p_2 \circ \cdots p_k$, $\bar{b} = b_1 \circ b_2 \circ \cdots b_k$, and $u^*(\bar{p}, \bar{x}, \bar{b}) = \bar{p}, \bar{x}$. Then u^* is hard-to-invert for some $k = |x_i|^{O(1)}$ if u is.

If a bit $b(x)$ can not be predicted with probability p, one can amplify the unpredictability using independent $x_i, i := 1 \cdots n/p^2$ at random and taking the Xor of $b(x_i)$. We now obtain an unpredictable bit as follows. Let $e(x)$ be an encoding of x so that x can be uniquely decoded from any y in the Hamming Sphere of radius $0.05|e(x)|$ centered at $e(x)$. Then for $f(x) = y$, $b(x, i) = i$-th bit of $e(x)$ is hard to predict given y on constant fraction of i's, if x is hard-to-predict from y.

We note that assumption in the next lemma (a special case of the next lemma was independently shown in [K2]) can not be further weakened to any class larger than $PSPACE$ since any language provable by a prover to a polynomial-time verifier must be in $PSPACE$

as was first observed by P. Feldman; (in particular, proving "or opening" the language induced by the commitment protocol and value).

Lemma 2 *Unless RPSPACE=AP, there exists a bit commitment protocol from a (PSPACE) sender to a weak receiver.*

Generalizing the above lemma even further, we show that for any complexity class C inside PSPACE, if there is an interactive proof of membership for a complete language in C by the prover who is also in C, and if C is hard on the average, then a bit-commitment protocol can be constructed, in which the prover need not be more powerful then C.

3 Bit-commitment from weak to strong

Theorem 2 *The existence of the following three protocols is equivalent, provided that the* `strong` *player can perform $P^{\#P}$ (or stronger) computations:*

- $(\texttt{weak} \xrightarrow{\text{BC}} \texttt{strong})$
- $(\texttt{weak} \xrightarrow{\text{OT}} \texttt{strong})$
- $(\texttt{strong} \xrightarrow{\text{OT}} \texttt{weak})$

Proof sketch:
$(\texttt{weak} \xrightarrow{\text{BC}} \texttt{strong}) \iff (\texttt{strong} \xrightarrow{\text{OT}} \texttt{weak})$:

$(\Longrightarrow)$ We are given a protocol $(\texttt{weak} \xrightarrow{\text{BC}} \texttt{strong})$ and we show how to execute $(\texttt{strong} \xrightarrow{\text{OT}} \texttt{weak})$ when `strong` player has b_0, b_1 as two input random bits to transmit via 1-2-OT.

Let ω_v denote the random tape of the `weak` player (wlog, we assume it's a string of a fixed (polynomial) size l). Let C denote the transcript of the messages exchanged when the `weak` player commits a bit in $(\texttt{weak} \xrightarrow{\text{BC}} \texttt{strong})$. Let $A_b(C) \leftarrow \{\omega_v :$ the conversation is C when `weak` player's random tape string is ω_v and `weak` player later decomits bit $b\}$. If we have a fixed C in context we just write A_0 and A_1. Note that these sets (i.e., $A_b(C)$) are disjoint and we may take C to be such that these are non-empty; otherwise the `strong` player can compute which value is being committed. Also, after the conversation, the `weak` player having a fixed C, and a (consistent) $\omega_v \in A_0(C)$, can not compute a string in $A_1(C)$; otherwise his committed bit and decommitted bit need not be the same. The protocol for 1-2-OT is as follows:

- `strong` and `weak` player execute $(\texttt{weak} \xrightarrow{\text{BC}} \texttt{strong})$ protocol. Let the conversation be C, the random tape of the `weak` player be $\omega_v \in \{0,1\}^l$, and the committed bit be b'.
- For $\beta \leftarrow 0$ to 1 do:

  Set $i \leftarrow 1$;
  [`Repeat`:]

  (`strong`): sends random $h_i^\beta \in \{0,1\}^l$
  (`weak`): sends $b_i^\beta := B(\omega_v, h_i^\beta)$ (the inner product) if $\beta = b'$ and a random bit otherwise.
  (`strong`): sends "stop" and exits loop if $\exists!\ \omega_v^\beta \in A_b\ \forall j \leq i \quad B(\omega_v^\beta, h_j^\beta) = b_j^\beta$.

  $i \leftarrow i+1$;
  [goto `Repeat`]
- End-For
- Then, the `strong` player chooses a random h so that $B(\omega_v^0, h) \neq B(\omega_v^1, h)$ and sends it to the weak player.

The above step is repeated thrice. The `weak` player randomly chooses two out of the three conversations and asks the `strong` player to convince him that the **strong** player acted according to the protocol (using the fact that this could be done in $P^{\#P}$ [LFKN]). If the **strong** player fails, the **weak** player aborts. Otherwise, the remaining conversation is used as follows: Let ω_0, ω_1 be the remaining "decomittal" strings of the third, unqueried conversation. The strong player selects a random string p, $|p| = l$ and sends to the **weak** player p, and two pairs $< \gamma_i, v_i >$, $i \in \{0,1\}$, where $v_i = b_i \oplus B(p, \omega_i)$, and $\gamma_i = B(h, \omega_i)$.

This can be shown to yield α-1-2-OT (where the sender can guess the result of the transfer with a slight advantage α), which is information-theoretically equivalent to OT [CK] using polynomial-time reductions.

($\Longleftarrow$): is straightforward: the **strong** player selects two random strings and plays 1-2-string-OT with the weak player. The "selection-bit" of a weak player serves as his committal. $\square$

<u>$(\texttt{weak} \xrightarrow{\text{BC}} \texttt{strong}) \iff (\texttt{weak} \xrightarrow{\text{OT}} \texttt{strong})$</u>:

($\Longleftarrow$) : BC is known to follow from OT [C, K].

($\Longrightarrow$): Assume the **weak** player has two bits b_0 and b_1 and he wishes to execute 1-2-OT to the **strong** player. Since we assume $(\texttt{weak} \xrightarrow{\text{BC}} \texttt{strong})$, it follows that the **strong** player can do both OT and BC to the **weak** player. So the **strong** player can commit a bit by putting it in an "envelope". The **strong** player makes envelopes with names $e_1, \cdots, e_4$ and forms the pairs $P_0 = \{e_1, e_2\}$ and $P_1 = \{e_3, e_4\}$ satisfying:

1. the contents one pair are identical, while the contents in the other pair are different.

2. there is a label $l(e_i) \in \{0,1\}$ such that it is distinct for each envelope within a pair.

The above step is repeated $2k$ times, where k is the security parameter. Subsequently, for k-size randomly chosen subset, **weak** player requests to see the contents of both pairs. If the above constraints are not verified, weak player aborts the protocol. If not, then for the remaining k pairs $(P_0^1, P_1^1), \ldots, (P_0^k, P_1^k)$ the **weak** player chooses random bits $b_0^1, b_1^1, \ldots b_0^k, b_1^k$ and chooses (using appropriate OT protocol) the contents c_0^j (for j from 1 to k) of the envelope $e_i^j \in P_0^j$ with $l(e_i) = b_0^j$ and the content c_1^j of $e_i \in P_1$ with $l(e_i) = b_1^j$. Then the **weak** player sends c_0^j, c_1^j to the **strong** player. The strong player divides c_i^j into two equal size groups, (putting into one group bits which are pairwise distinct), and sends to the weak player indices of this two groups (without specifying which group is which, of course). The **weak** player takes an Xor of the first input bit (i.e. b_0) with the corresponding b_i^j bits of the first group and Xor of the second input bit (i.e. b_1) with the second group and sends this two bits back to the **strong** player. For the set for which the **strong** player knows *all* the b_i^j, he can compute the value of the input bit, while for the other bit, with overwhelming probability the value is hidden. (Alternatively, the strong player can ask which of groups to use with which input bit, first or second). $\square$

We can conclude that:

Corollary 1 *Given a* <u>$(\texttt{weak} \xrightarrow{\text{BC}} \texttt{strong})$</u> *protocol, then any partial information game of polynomial-size between a weak and a strong ($P^{\#P}$ or stronger) player is realizable.*

Bit commitment from weak to strong: In the bit-commitment protocol from the weak player to the strong one, recall that the goal is that even an infinitely-powerful "receiver" can not guess the committed bit with probability better then $\frac{1}{2}+\epsilon$, but such that a polynomially-bounded committer can not change a committed value, unless he breaks the assumption (which is explicitly) stated in the theorem.

In [OVY] we show how OT can be implemented in the asymmetric model under general complexity assumptions. For the sake of completeness, we explain briefly the technique behind this construction in the appendix. Using the results there and applying theorem 2, we get:

Corollary 2 *Given any one-way permutation, there exists a (weak-to-strong) bit-commitment protocol from a probabilistic poly-time "committer" to an (NP or stronger) "receiver".*

Corollary 3 *Given any one-way function, there exists a (weak-to-strong) bit-commitment protocol from a probabilistic poly-time "committer" to a ($P^{\#P}$ or stronger) "receiver".*

We stress that in the above two lemmas, once committed, the value of the committed bit is protected from the receiver *information-theoretically.*

Acknowledgments

We would like to thank Gilles Brassard, Shafi Goldwasser, Silvio Micali, Moni Naor, and Noam Nisan for helpful discussions.

References

[AFK] M. Abadi, J. Feigenbaum and J. Kilian. *On Hiding Information from an Oracle* J. Comput. System Sci. 39 (1989) 21-50.

[B1] Blum M., *Applications of Oblivious Transfer*, Unpublished manuscript.

[B2] Blum, M., "Coin Flipping over the Telephone," IEEE COMPCON 1982, pp. 133-137.

[BM] Blum, M. and S. Micali, "How To Generate Cryptographically Strong Sequences of Pseudorandom Bits," FOCS 82, (Also SIAM J. Comp. 84).

[BCC] G. Brassard, D. Chaum and C. Crepeau, *Minimum Disclosure Proofs of Knowledge*, JCSS, v. 37, pp 156-189.

[BCR] G. Brassard, C. Crépeau and J.-M. Robert, *"Information Theoretic Reductions among Disclosure Problems"*, FOCS 86 pp. 168-173.

[BCY] Brassard G., C. Crépeau, and M. Yung, "Everything in NP can be proven in Perfect Zero Knowledge in a bounded number of rounds," *ICALP* 89.

[BG] Beaver D., S. Goldwasser *Multiparty Computation with Faulty Majority* FOCS 89, pp 468-47.

[BMO] Bellare, M., S. Micali and R. Ostrovsky, "The (True) Complexity of Statistical Zero Knowledge" STOC 90.

[BGW] Ben-Or M., S. Goldwasser and A. Wigderson, *Completeness Theorem for Noncryptographic Fault-tolerant Distributed Computing*, STOC 88, pp 1-10.

[CCD] D. Chaum, C. Crepeau and I. Damgard, *Multiparty Unconditionally Secure Protocols*, STOC 88, pp 11-19.

[Co] A. Condon, *Computational Models of Games*, Ph.D. Thesis, University of Washington, Seattle 1987. (MIT Press, ACM Distinguished Dissertation Series).

[C] C. Crépeau, *Equivalence between Two Flavors of Oblivious Transfer*, Crypto 87.

[CK] C. Crépeau, J. Kilian *Achieving Oblivious Transfer Using Weakened Security Assumptions* , FOCS 88.

[EGL] S. Even, O. Goldreich and A. Lempel, *A Randomized Protocol for Signing Contracts*, CACM v. 28, 1985 pp. 637-647.

[FMR] Fischer M., S. Micali, C. Rackoff *An Oblivious Transfer Protocol Equivalent to Factoring*, Manuscript.

[GHY] Z. Galil, S. Haber and M. Yung, *Cryptographic Computations and the Public-Key Model*, Crypto 87.

[FO] J. Feigenbaum and R. Ostrovsky, *A Note On One-Prover, Instance-Hiding Zero-Knowledge Proof Systems* In Proceedings of the first international symposium in cryptology in Asia, (ASIACRYPT'91), November 11-14, 1991, Fujsiyoshida, Yamanashi, Japan.

[GL] O. Goldreich and L. Levin, *Hard-core Predicate for ANY one-way function* , STOC 89.

[GMW1] O. Goldreich, S. Micali and A. Wigderson, *Proofs that Yields Nothing But their Validity*, FOCS 86, pp. 174-187.

[GMW2] O. Goldreich, S. Micali and A. Wigderson, *How to Play any Mental Poker* , STOC 87.

[GMR] S. Goldwasser, S. Micali and C. Rackoff, *The Knowledge Complexity of Interactive Proof-Systems*, STOC 85, pp. 291-304.

[GN] S. Goldwasser and N. Nisan, personal communication.

[G] Y. Gurevich, Average Case Completeness, Journ. of Comp Sys. Sci, 1991.

[H] Hastad, J., "Pseudo-Random Generators under Uniform Assumptions", *STOC 90*.

[ImLu] R. Impagliazzo and M. Luby, *One-way Functions are Essential for Complexity-Based Cryptography* FOCS 89.

[ILL] R. Impagliazzo, R., L. Levin, and M. Luby "Pseudo-Random Generation from One-Way Functions," *STOC* 89.

[ImLe] R. Impagliazzo, R., L. Levin, No better ways to generate hard NP instances than to choose uniformly at random, FOCS 90.

[IR] R. Impagliazzo and S. Rudich, *On the Limitations of certain One-Way Permutations* , STOC 89.

[IY] R. Impagliazzo and M. Yung, *Direct Minimum-Knowledge Computations*, Proc. of Crypto 87, Springer Verlag.

[K] J. Killian, *Basing Cryptography on Oblivious Transfer* , STOC 1988 pp 20-31.

[K2] J. Kilian *Interactive Proofs With Provable Security Against Honest Verifiers* CRYPTO 90, pp. 371-384.

[KMO] J. Killian, S. Micali and R. Ostrovsky *Minimum-Resource Zero-Knowledge Proofs*, FOCS 1989.

[L] L. Levin *Average Case Complete Problems* SIAM J. of Computing, 1986 VOL 15, pp. 285-286.

[LFKN] Lund, C., L. Fortnow, H. Karloff, and N. Nisan, "Algebraic Methods for Interactive Proof Systems" FOCS 90.

[N] M. Naor *"Bit Commitment Using Pseudo-Randomness"* Crypto-89 pp.123-132.

[NOVY] M. Naor, R. Ostrovsky, R. Venkatesan, M. Yung, Zero-Knowledge Arguments for NP can be Based on General Complexity Assumptions, manuscript.

[Ost] R. Ostrovsky *One-way Functions, Hard on Average Problems and Statistical Zero-knowledge Proofs* In Proceedings of 6'th Annual Structure in Complexity Theory Conference. June 30 – July 3, 1991, Chicago. pp. 51-59.

[OVY] R. Ostrovsky, R. Venkatesan, M. Yung, Fair Games Against an All-powerful Adversary, *Sequences 91*, July 1991, Positano, Italy, to appear in Springer Verlag. (Also presented at *DIMACS 1990 Cryptography Workshop*, 1-4 October 1990, Princeton.)

[R] M., Rabin *"How to exchange secrets by oblivious transfer"* TR-81 Aiken Computation Laboratory, Harvard, 1981.

[RB] T. Rabin and M. Ben-Or, *Verifiable Secret Sharing and Secure Protocols* , STOC 89.

[S] A. Shamir *IP=PSPACE* , FOCS 90.

[SRA] A. Shamir, R. Rivest and L. Adleman, *Mental Poker*, Technical Memo MIT (1979).

[VL] Venkatesan R., and L. Levin *Random Instances of a Graph Coloring Problem are Hard* STOC 88. Almost Journal version available.

[Y] A. C. Yao, *How to Generate and Exchange Secrets*, FOCS 86.

Appendix

We briefly recall our results from [OVY] on how $\texttt{strong}\stackrel{\mathrm{OT}}{\longrightarrow}\texttt{weak}$ protocols can be based on general complexity assumption. Assume that the `strong` player (the Sender S) has a secret random input bit b, which he wants the `weak` player (the Receiver R) to get with probability 1/2. R wants S not to know whether or not R received the bit.

For simplicity, let f be a strong one-way permutation (invertible in polynomial time only on a exponentially small fraction of the inputs). Below, S is given a secret input bit b at the beginning of the protocol, $B(x, y)$ denotes the dot-product mod 2 of x and y, and all $h_i \in \{0,1\}^n$ are linearly independent. The following is a "zooming" technique which can be described as gradually focusing on a value, while maintaining information-theoretic uncertainty.

- $\{R(0)\}$: R selects x' of length n at random and computes $x = f(x')$. He keeps both x' and x secret from S.

- **For i from 1 to $(n-1)$ do the following steps:**

$\{S(i)\}$: S selects at random h_i and sends it to R.

$\{R(i)\}$: R sends $c_i := B(h_i, x)$ to S.

- $\{S(n)\}$: Let x_0, x_1 be the ones which satisfy $\forall i, 1 \le i < n, B(h_i, x_{\{0,1\}}) = c_i$. S flips a random coin j, selects a random string p, $|p| = l$ and sends to R a triple $< p, x_j, v >$, where $v = b \oplus B(p, f^{-1}(x_j))$.

- $\{R(n)\}$: R checks if for his x, $x = x_j$, and if so, computes $b' = v \oplus B(p, x')$ as the resulting bit he gets from S via an "OT" protocol and outputs (x, b').

We omit the proofs of the following theorems. (The proofs involve applying the basing zooming technique based on the power of the sender and what he can interactively prove.)

Theorem 3 *There exists a protocol implementing OT from a strong (at least probabilistic NP or stronger) player to a probabilistic polynomial-time player, based on any one-way permutation.*

Theorem 4 *There exists a protocol implementing OT protocol from an all-powerful (at least probabilistic $P^{\#P}$ or stronger) player to a probabilistic polynomial-time player, given any one-way function.*