Information Theoretic Reductions among Disclosure Problems

Gilles BRASSARD [†]

Département d'I.R.O.
Université de Montréal

Claude CREPEAU [‡]

Department of Computer Science [*]
MIT

Jean-Marc ROBERT

Département de Génie Electrique
Ecole Polytechnique de Montréal

Abstract

Alice disposes of some number of secrets. She is willing to disclose one of them to Bob. Although she agrees to let him choose which secret he wants, she is not willing to allow him to gain any information on more than one secret. On the other hand, Bob does not want Alice to know which secret he wishes. An *all-or-nothing disclosure* is one by which, as soon as Bob has gained any information whatsoever on one of Alice's secrets, he has wasted his chances to learn anything about the other secrets. We assume that Alice is honest when she claims to be willing to disclose one secret to Bob (i.e. she is not about to send junk). The only cheating Alice is susceptible of trying is to figure out which secret is of interest to Bob. We address the following question from an information theoretic point of view: *what is the most elementary disclosure problem*? The main result is that the general all-or-nothing disclosure of secrets is equivalent to a much simpler problem, which we call the two-bit problem.

1. INTRODUCTION

Alice disposes of some number of secrets. She is willing to disclose one of them to Bob. Although she agrees to let him choose which secret he wants, she is not willing to allow him to gain any information on more than one secret. On the other hand, Bob does not want Alice to know which secret he wishes. This is a useful building block in crypto-protocols. For instance, it can be used to easily implement a multi-party mental Poker protocol similar to that of [C1], i.e.: safe against player coalitions. An *all-or-nothing disclosure* is one by which, as soon as Bob has gained any information whatsoever on one of Alice's secrets, he has wasted his chances to learn anything about the other secrets. In particular, it must be impossible for Bob to gain joint information on several secrets, such as their exclusive-or. Notice that this is crucial, because it is well-known in classical cryptography that the exclusive-or of two plaintext English messages allows easy recovery of them both, just as a running stream Vigenère would [D].

We assume that Alice is honest when she claims to be willing to disclose one secret to Bob (i.e. she is not about to send junk). The only cheating Alice is susceptible of trying is to figure out which secret is of interest to Bob. Although equally worthwhile, we do *not* address here the problem of *verifiable* secrets [1], because it is too much application dependent and because it does not make sense in our information theoretic setting in which both parties could have unlimited computing power.

Let us stress that the major novelty consists in Bob's choosing which secret he obtains. This is interesting whenever the secrets are not anonymous: although Bob does not know their contents, he knows their individual purpose [2]. Consider for instance the following situation: an international spy disposes of a large corpus of various state secrets. He sells them by the piece to whoever is willing to pay the price. In his catalogue, each secret is advertised with a tantalizing title, such as "where is Abu Nidal". He would not accept to give away two secrets for the price of one, or even partial information on more than one secret. On the other hand, you (the potential buyer) would not pay for a randomly chosen secret, but are reluctant to let him know which secret you wish to acquire, because his knowledge of your specific interests could be a valuable secret for him to sell to someone else (under the title: "who is looking for terrorists", for instance). Let us point out that this problem was addressed and solved more that 15 years ago *by quantum physical means*, when the number of secrets is at most three, in Wiesner's original Quantum Cryptography paper [W].

Under cryptographic assumptions, a practical computationally secure protocol for this problem has been proposed in [BCR]. It is reviewed at the end of this paper. Notice that, for its particular application in [C2] (mental poker) this protocol forces Alice to cooperate (i.e. the secrets are verifiable). Here, we address the following question from an information theoretic point of view: *what is the most elementary disclosure problem?* It turns out that the general all-

[†] Supported in part by NSERC grant A4107.

[‡] Supported in part by an NSERC postgraduate scholarship.

[*] Research conducted at the Université de Montréal

[1] That is, preventing that Bob unknowingly obtains a falsified secret should Alice fail to cooperate honestly.

[2] In order to get a computationally secure scheme under cryptographic assumptions, it would otherwise suffice to use a variation on oblivious transfer (attributed to Oded Goldreich in [BPT]) that allows "Alice to transfer to Bob exactly one out of two recognizable messages" so that neither has control over which message will be received.

0272-5428/86/0000/0168$01.00 © 1986 IEEE

or-nothing disclosure of secrets is equivalent to the *two-bit problem* (described below). This result does not depend on computational complexity cryptographic assumptions.

2. THE MOST ELEMENTARY DISCLOSURE PROBLEM

It is exactly as hard to all-or-nothing disclose one t-bit secret among n than it is to disclose one bit among two. This result is obtained by a chain of reductions that allows the collapse of an apparent hierarchy of disclosure problems. Here is a list of problems that turn out to be information-theoretically equivalent, that is even if either or both party(ies) had unlimited computing power.

> The *two-bit problem* (2BP): Alice disposes of two secret bits and she is willing to disclose one of them to Bob, at his choosing. Bob must not be allowed to learn more than one bit of information on Alice's bits, but Alice will not be upset if Bob succeeds in gaining any (deterministic) one-bit function of these two bits, such as their exclusive-or. If Bob plays fair and obtains the physical bit of his choice, Alice does not know which of her two bits she disclosed.
>
> The *all-or-nothing two-bit problem* (AN2BP): Alice disposes of two secret bits and she is willing to disclose one of them to Bob, at his choosing. Nothing Bob can do will give him more than one of these *physical* bits: as soon as he obtains any information on one of them, he looses all hopes to gain any information on the other. Alice does not know which of her two bits she disclosed.
>
> The *all-or-nothing n-bit problem* (ANNBP): This is identical to the previous problem, except that Alice owns n secret bits rather than 2. She wishes to all-or-nothing disclose one of them to Bob, at Bob's choosing.
>
> The *all-or-nothing disclosure of secrets* (ANDOS): Described previously.

We shall now sketch how to efficiently transform any protocol for 2BP into one for AN2BP, any protocol for AN2BP into one for ANNBP, and any protocol for ANNBP into one for ANDOS. The most interesting reduction is the last one, so that we will be rather brief about the first two. More details will be provided in the final paper. The first reduction (2BP $\Rightarrow$ AN2BP) allows an exponentially small probability of undetected cheating, but no amounts of computing power could increase this probability. The other two reductions, however, are information-theoretically perfect in the sense that any fool-proof solution to AN2BP would yield a fool-proof solution to the general ANDOS problem. An information-theoretic secure solution to any of these problems, including the elementary 2BP, would be of considerable interest. Under cryptographic assumptions, we review the computationally secure solution of [BCR] in section 4.

2.1. 2BP $\Rightarrow$ AN2BP

Assume the availability of a protocol for 2BP. Let Alice dispose of two secret bits a and b, of which she is willing to *all-or-nothing* disclose one to Bob.

protocol 1

> Let m be an even integer, used as safety parameter. Alice randomly chooses a subset $X \subseteq \{1, 2, \cdots, m\}$ of size $m/2$. Let u and v be the smallest positive integers within X and outside X, respectively. Alice randomly chooses a total of $2m-2$ bits r_i and s_j, $1 \le i \le m$, $i \ne u$, and $1 \le j \le m$, $j \ne v$. She sets the bits r_u and s_v such that $a = \oplus\{r_i \mid i \in X\}$ and $b = \oplus\{s_i \mid i \notin X\}$, where "$\oplus$" denotes the exclusive-or. She uses the 2BP protocol to disclose Bob one of r_1 or s_1, one of r_2 or s_2, $\cdots$, and one of r_m or s_m. Only then does she give X to Bob. If he systematically asked for all r_i's (resp. all s_i's), he can easily reconstruct a (resp. b). □

This protocol allows Bob to attempt cheating with a non-zero probability of success: if he chooses randomly to read $m/2$ of the r_i's and $m/2$ of the s_i's, he obtains both a and b only if he guessed X correctly, which happens with probability $1/\binom{m}{m/2} \approx 2^{-m}\sqrt{\pi m/2}$. With such a strategy, however, he has an overwhelming probability to get absolutely no information on neither a nor b. Another strategy would be to ask for $r_i \oplus s_i$ for some i; but this is dumb because it irrevocably wastes his chances to learn one of a or b, depending of whether $i \in X$. The analysis of this protocol becomes significantly harder if Bob attempts biased questions on r_i and s_i, such as their conjunction, for some values of i. In this case, use of Bernshtein's Law of Large Numbers [K] allows us to prove that, no matter which m bits he requests from the 2BP protocol, Bob only has an exponentially small chance of getting more than an exponentially small advantage on both a and b, simultaneously. The details are quite messy; they can be found in the final version of this paper.

2.2. AN2BP $\Rightarrow$ ANNBP

Assume the availability of a protocol for AN2BP. Let Alice dispose of n secret bits $b_1, b_2, \cdots, b_n$, of which she is willing to all-or-nothing disclose one to Bob.

protocol 2

> Alice randomly chooses $n-2$ bits $r_1, r_2, \cdots, r_{n-2}$. She then uses the AN2BP protocol to all-or-nothing disclose Bob one of b_1 or r_1, one of $b_2 \oplus r_1$ or $r_1 \oplus r_2$, $\cdots$, one of $b_{n-2} \oplus r_{n-3}$ or $r_{n-3} \oplus r_{n-2}$, and one of $b_{n-1} \oplus r_{n-2}$ or $b_n \oplus r_{n-2}$. If Bob wishes secret b_1, he simply asks for it in the first instance of the AN2BP protocol. If Bob wishes secret b_s, for $2 \le s \le n-1$, he asks

for r_1, $r_1 \oplus r_2$, $\cdots$, $r_{s-2} \oplus r_{s-1}$ and $b_s \oplus r_{s-1}$ in the first s instances. He then computes b_s as the exclusive-or of these s bits. It is just as easy for him to get b_n. □

The point here is that once Bob decides to ask for $b_s \oplus r_{s-1}$, for some s, he looses track of $r_{s-1} \oplus r_s$, hence r_s, so that the answers to all further questions become meaningless. Notice that this would *not* be so if a 2BP protocol had been used instead of an AN2BP protocol. Indeed, Bob could then have asked for b_1 and $(b_{n-1} \oplus r_{n-2}) \oplus (b_n \oplus r_{n-2}) = b_{n-1} \oplus b_n$, for instance, thus getting two bits of information on Alice's n secrets.

2.3. ANNBP ⇒ ANDOS

Consider any function $f: \{0,1\}^m \rightarrow \{0,1\}^t$. The set $I \subseteq \{1, 2, \cdots, m\}$ is said to *bias* f if knowledge of the bits $<x[i]>_{i \in I}$ in a string x of length m is susceptible to yielding information on $f(x)$, that is if there exists an assignment of Boolean values to the bits $<x[i]>_{i \in I}$ and some output value $z \in \{0,1\}^t$ such that

$$\#\{x \in \{0,1\}^m \mid f(x) = z \text{ and the } I\text{-bits of } x \text{ respect the given assigment}\} \neq 2^{m-t-\#I}.$$

The *information support* of f is defined as $\{I \subseteq \{1, 2, \cdots, m\} \mid I \text{ biases } f\}$. Finally, the function f is called an *(m,t)-zigzag* if its information support does not contain two disjoint sets.

Intuitively, a zigzag has the following crucial property: there does not exist two *disjoint* sets $I, J \subseteq \{1, 2, \cdots, m\}$ and two (possibly) distinct input strings $x, y \in \{0,1\}^m$ such that asking for the I-bits of x and the J-bits of y yields information on both $f(x)$ and $f(y)$. Therefore, in order to obtain information on $f(x)$ and $f(y)$, it is necessary to query some input bits $x[i]$ and $y[i]$ for the same i. This is precisely what we need to turn an ANNBP protocol into an ANDOS protocol.

Assume now the availability of a protocol for ANNBP. Let Alice dispose of n t-bit secrets $x_1, x_2, \cdots, x_n$. Let $f: \{0,1\}^m \rightarrow \{0,1\}^t$ be a publicly known zigzag, for some integer m.

protocol 3

For each $i \leq n$, Alice randomly chooses some $y_i \in \{0,1\}^m$ such that $f(y_i) = x_i$. Let b_{ij} be y_i's j^{th} bit for $1 \leq i \leq n$ and $1 \leq j \leq m$. For each value of j, Alice uses the ANNBP protocol to allow Bob to learn one of $<b_{ij}>_{1 \leq i \leq n}$. If Bob decides to read b_{sj} for some fixed s, and each $1 \leq j \leq m$, this gives him y_s and hence his selected secret $x_s = f(y_s)$. □

The only cheating Bob could attempt is to select $b_{i_1 j_1}$ and $b_{i_2 j_2}$, where $i_1 \neq i_2$, in order to learn something on both y_{i_1} and y_{i_2}. Because of the ANNBP protocol, this would imply that $j_1 \neq j_2$. Therefore, since f is a zigzag, this could not give him information on both x_{i_1} and x_{i_2}. We have hence achieved an efficient ANDOS protocol under the conditions that the ANNBP protocol be efficient, m be polynomial in t, f be efficient to compute, and uniformly distributed random inverses of f be efficient to select. Here, we shall limit our attention to linear zigzags, that is functions f that can be represented by a $t \times m$ Boolean matrix F such that $f(y) = Fy$, both y and $f(y)$ being given as column vectors, and all arithmetic being done modulo 2. The following theorem characterizes linear zigzags.

Theorem. Let F be the Boolean matrix associated with some function $f: \{0,1\}^m \rightarrow \{0,1\}^t$. For any non empty $X \subseteq \{1, 2, \cdots, t\}$, let $I_X \in \{0,1\}^m$ be defined as the bitwise exclusive-or of the rows of F indexed by members of X. The function f is a linear zigzag if and only if for any two non-empty $X, Y \subseteq \{1, 2, \cdots, t\}$, the bitwise conjunction of I_X and I_Y is not identically zero. (Note that the special case $X = Y$ implies that I_X cannot be identically zero for any non-empty X, hence F must have full row rank.)

Proof (sketch).

Necessary: if the condition fails with $X = Y$, the exclusive-or of the X-bits of $f(x)$ is always zero, hence the empty set is in the information support of f, so that f cannot be a zigzag; if it fails with $X \neq Y$, the exclusive-or of the X-bits of $f(x)$ and the exclusive-or of the Y-bits of $f(y)$ can both be obtained by asking disjoint questions on bits of x and y.

Sufficient: this is a consequence of the xor-lemma of [BBR], which says in essence that any partial information on $f(x)$ obtained by knowledge of specific bits of x automatically gives complete knowledge on some exclusive-or of the bits of $f(x)$. □

This characterisation allows to prove by induction the existence of a linear (m,t)-zigzag whenever $m = 3^{\lceil log_2 t \rceil}$. For any integer $k \geq 0$, recursively define the $2^k \times 3^k$ matrix F_k as $F_0 = [1]$, and

$$F_{k+1} = \begin{bmatrix} F_k & 0_k & F_k \\ 0_k & F_k & F_k \end{bmatrix}$$

where 0_k is the identically zero $2^k \times 3^k$ matrix. For any t, let $k = \lceil log_2 t \rceil$. Any t rows of F_k defines a linear $(3^k,t)$-zigzag. For instance,

$$F_2 = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 \end{bmatrix}$$

defines a (9,4)-zigzag. It is clear that m is polynomial in t ($m \in O(t^{1.59})$), this zigzag is easy to compute, and random inverses are easy to select. This completes the information-theoretic reduction from any efficient solution for the two-bit problem to an efficient solution for the all-or-nothing disclosure of secrets problem.

There is a nice graphical representation of this construction that explains why we call such functions "zigzags". The figure in the Appendix shows a (27,8)-zigzag. The nodes on the left correspond to the 27 input bits, whereas the nodes on the right correspond to the 8 output bits. Each non-input node should be thought of as computing the exclusive-or of its two inputs. Given output x, this suggests a systematic way to randomly select an input y such that $x = f(y)$: peel off level by level, from right to left.

The optimality question for zigzags is still open: could there exist an $(m,2^k)$-zigzag for $m < 3^k$? One very crude lower bound on m for the existence of linear (m,t)-zigzags can be obtained by a reduction from the theory of binary linear error-correcting codes [MS]. Indeed, the matrix for any linear (m,t)-zigzag must be the generator matrix of an $[m, t]$ binary linear code such that the minimal distance between any two codewords is at least t (the converse fails in general). Therefore, from Griesmer's bound [G], we get $m \geq 3t-2-\beta(t-1)$, where $\beta(x)$ is the number of 1-bits in the binary representation of x. There could nonetheless exist a more efficient non-linear zigzag.

After reading a first draft of this paper, Oded Goldreich and Silvio Micali discovered a different characterization for zigzags. Consider a $t \times m$ matrix F and some set $I \subseteq \{1, 2, \cdots, m\}$. Let F_I denote the $t \times \#I$ submatrix consisting of the I-columns of F and let F_I' denote the $t \times (m-\#I)$ submatrix consisting of F with its I-columns removed. The function defined by matrix F is an (m,t)-zigzag if and only if, for every $I \subseteq \{1, 2, \cdots, m\}$, at least one of F_I or F_I' has full rank. More intuitively, $f: \{0, 1\}^m \rightarrow \{0, 1\}^t$ is an (m,t)-zigzag if and only if "for every set of indices $I \subseteq \{1, 2, \cdots, m\}$ either (for every b) x's bits indexed by I are not determined given that $f(x) = b$ or this holds for x's bits indexed by $\{1, 2, \cdots, m\} - I$" [GrM].

Umesh Vazirani has suggested a different approach for the ANNBP ⇒ ANDOS reduction [V]. His insight was to use again the idea behind the 2BP ⇒ AN2BP reduction of Section 2.1. This leads to a more economical protocol for ANDOS, at the cost of introducing an exponentially small probability of undetected cheating even if the ANNBP protocol had been perfect.

3. RELATED RESULTS (sketch)

There are two intermediate problems between ANNBP and ANDOS when Alice disposes of n t-bit secrets, one of which she is willing to disclose to Bob. In ANDOS, she does not want Bob to get information on more than one secret. One natural intermediate problem is for her to tolerate this kind of cheating, as long as Bob does not get *physical* bits from more than one secret. The other intermediate problem is more restrictive: Alice wants to make sure that if Bob ever gets a physical bit from some secret, he cannot get information on any other secret (therefore, he could get information on more than one secret only if he were willing to give up knowledge of physical bits altogether).

We have simple solutions for both these problems; if we redefine the notion of zigzag accordingly, we have a (t,t)-*neo*1-zigzag and a $(2t-1,t)$-*neo*2-zigzag for the reduction of ANNBP to each of these two intermediate problems. Both these reductions are proven optimal among all linear schemes. As a consequence of the lower bound mentioned at the end of the previous section, the all-or-nothing disclosure of secrets problem is strictly harder than both intermediate problems, assuming we only consider linear reductions from the all-or-nothing n-bit problem.

4. OUTLINE OF THE SCHEME OF [BCR]

Because the proceedings of the CRYPTO conference may not be widely distributed this year (1986), let us describe the quadratic residuosity based computationally secure ANDOS protocol given in [BCR]. We assume here that the reader has some number theoretic background, being familiar with the notation $\mathbb{Z}_m^*$, the notions of quadratic residues and Jacobi symbols, and the quadratic residuosity assumption (QRA) [GwM]. We also assume the reader is familiar with the principle of zero-knowledge interactive proofs [GMR, GHY, BC].

Let $x_1, x_2, \cdots, x_n$ be Alice's t-bit secrets, and let b_{ij} be x_i's j^{th} bit for $1 \leq i \leq n$ and $1 \leq j \leq t$. Initially, Alice randomly selects two distinct large primes p and q together with a quadratic non-residue y modulo $m = pq$ whose Jacobi symbol is +1. For each secret bit b_{ij}, she selects a random $x_{ij} \in \mathbb{Z}_m^*$ and computes $z_{ij} = x_{ij}^2 y^{b_{ij}} \textbf{ mod } m$. Notice that z_{ij} is a quadratic residue if and only if $b_{ij} = 0$. Finally, Alice gives Bob both m and y, keeping p and q secret, together with all the z_{ij}'s. According to QRA, this does not enable Bob to obtain in polynomial time any information on Alice's actual secrets.

If Bob wanted to know bit b_{ij} for one specific i and j, and if Alice were willing to cooperate, the following protocol comes to mind: Bob chooses a random $r \in \mathbb{Z}_m^*$ and a random bit a, he computes the question $q = z_{ij} r^2 y^a \textbf{ mod } m$ and he asks Alice for the quadratic residuosity of q. Clearly, $b_{ij} = a$ if and only if q is a quadratic residue. On the other hand, regardless of i and j, q is a completely random element of $\mathbb{Z}_m^*$ and thus Alice has no idea as to which of her secret bits she gave away. One might be tempted to "solve" ANDOS by allowing Bob to ask t such questions, one for each bit of the secret he wants. There are three severe flaws with this idea:

- Bob could ask for t bits taken from distinct secrets.
- Bob could obtain in one question the exclusive-or of severals bits. For instance, he could ask the question

$q = z_{ij} z_{kj} r^2 y^a \textbf{ mod } m$ and therefore learn $b_{ij} \oplus b_{kj}$. As pointed out in the introduction, this would most probably enable him to obtain two complete secrets by asking for their exclusive-or, assuming the actual secrets are in plaintext English.

- More subtly, despite the previous claim, this would open the door for Alice to cheat as well! Indeed, she could lie from the beginning and give Bob a quadratic *residue* for her y. In this case, the questions asked by the unsuspecting Bob would keep the same quadratic character as the corresponding z's, allowing Alice to figure out Bob's interests.

In order to solve these difficulties, it is imperative that both Alice and Bob convince the other of her/his good faith: Alice must show that the information she posted initially is genuine and Bob must convince Alice that his questions are honest. This is where zero-knowledge interactive protocols come into play. The third problem above is solved by Alice using zero-knowledge interactive protocols of [GHY] and [GMR] to convince Bob that m has only two prime factors and that y is a quadratic non-residue modulo m, respectively. In a context of *verifiable* secret, this is where Alice would also convince Bob that the secrets hidden by the z_{ij}'s respect whichever conditions befit the application (a specific example is given in [C2]).

The first two problems above are harder to control. Although we have found several solutions, we only sketch here our favourite. Let σ be a permutation of $\{1, 2, \ldots, n\}$. A σ-packet P_σ consists of one question for each bit of each secret in the following way: $P_\sigma = <q_{kj} \mid 1 \le k \le n, 1 \le j \le t>$ such that each $q_{kj} = z_{ij} r_{kj}^2 y^{a_{kj}} \textbf{ mod } m$, where r_{kj} is a random element of $\mathbf{Z}_m^*$, a_{kj} is a random bit and $i = \sigma^{-1}(k)$. Moreover, a σ-packet is valid if Bob knows the corresponding σ, r_{kj}'s and a_{kj}'s (notice that any collection of nt elements of $\mathbf{Z}_m^*$ is a σ-packet for every permutation σ, and any valid packet looks like any other collections of random elements of $\mathbf{Z}_m^*$ to Alice).

After the initialisation described previously, the ANDOS protocol proceeds as follows if x_i is the secret of interest to Bob.

- Bob selects a random permutation σ and forms a valid σ-packet P_σ.
- Bob gives P_σ to Alice, keeping σ secret, and convinces her that it is valid (see below).
- Bob sends $k = \sigma(i)$ to Alice as his actual request.
- Alice gives Bob the quadratic character of each q_{kj} in Bob's packet P_σ, for this specific k and each $1 \le j \le t$.
- Bob infers each of Alice's bits b_{ij} for $1 \le j \le t$, hence he obtains x_i as desired.
- If Bob wishes to obtain another secret and if Alice is willing to give (or sell) it to him, it suffices to repeat the previous 3 steps with the relevant new value for i.

It is of course crucial that Alice be convinced that Bob's packet is valid, for he could otherwise stuff it with dishonest questions and we would be back to the beginning. This is achieved by an idea very similar to those leading to the perfect zero-knowledge interactive protocol of [BC]. Let s be a safety parameter agreed upon between Alice and Bob. After giving Alice his σ-packet P_σ, Bob chooses s additional permutations $\sigma_1, \sigma_2, \cdots, \sigma_s$ of $\{1, 2, \cdots, n\}$ and he creates s additional σ_i-packets $P_1, P_2, \cdots, P_s$. He sends all these packets together with the original P_σ. At this point, Alice selects a random subset $X \subseteq \{1, 2, \cdots, s\}$ and sends it to Bob as a challenge. In order to convince her of the validity of P_σ, Bob must:

- for each $l \in X$, prove the validity of P_l to Alice by disclosing σ_l and all the random elements of $\mathbf{Z}_m^*$ and random bits used in the creation of P_l;
- for each $l \notin X$, prove to Alice that P_σ is valid if and only if P_l is valid by disclosing $\sigma_l^{-1}\sigma$ and showing that he is capable of transforming the questions in P_σ into the corresponding questions in P_l (we leave the details of this to the reader).

At the end of this subprotocol, Alice will be convinced that P_σ is valid, with a 2^{-s} probability of beeing fooled by Bob. Indeed, the only way he could convince her of the validity of an invalid P_σ would be by producing valid packets for each $l \in X$ and invalid packets for each $l \notin X$. Since he must do so before being told X, the result follows from the fact that Alice has 2^s different choices for X.

ACKNOWLEDGEMENTS

We wish to thank David Chaum, Oded Goldreich, Silvio Micali, René Peralta, Joel Seiferas and Umesh Vazirani for their valuable comments on our protocols and for suggesting improvements. Isabelle Duchesnay has suggested the international spy application. Gilles Brassard also wishes to thank Manuel Blum for his inspiring talk at the MIT Endicott House conference in June 1985.

REFEEENCES

[BBR] C. H. Bennett, G. Brassard and J.-M. Robert, "Privacy Amplification through Public Discussion", submitted to *SIAM Journal on Computing*, 1985.

[BPT] R. Berger, R. Peralta and T. Tedrick, "A Provably Secure Oblivious Transfer Protocol", *Proceedings of EUROCRYPT 84*, 1984, pp. 379-386.

[BC] G. Brassard and C. Crépeau, "Non-Transitive Transfer of Confidence: a *Perfect* Zero-Knowledge Interactive Protocol for SAT and Beyond", *these Proceedings of the 27th Annual IEEE Symposium on the Foundations of Computer Science, 1986.*

[BCR] G. Brassard, C. Crépeau and J.-M. Robert, "All-or-Nothing Disclosure of Secrets", *presented at CRYPTO 86*, 1986.

[C1] C. Crépeau, "A Secure Poker Protocol That Minimizes the Effect of Player Coalitions", *Advances in Cryptology: Proceedings of CRYPTO 85*, H. C. Williams ed., Lecture Notes in Computer Science 218, Springer-Verlag, Berlin, 1986, pp. 73-86.

[C2] C. Crépeau, "A Zero-Knowledge Poker Protocol that Achieves Confidentiality of the Players' Strategy, *or* How to Achieve an Electronic Poker Face", *presented at CRYPTO 86*, 1986.

[D] D. E. R. Denning, *Cryptography and Data Security*, Addison-Wesley, Reading, Massachussetts, 1982.

[GHY] Z. Galil, S. Haber and M. Yung, "A Private Interactive Test of a Boolean Predicate and Minimum-Knowledge Public-Key Cryptosystems" *Proceedings of the 26th Annual IEEE Symposium on the Foundations of Computer Science*, 1985, pp. 360-371.

[GrM] O. Goldreich and S. Micali, *Personal communication*, 1986.

[GwM] S. Goldwasser and S. Micali, "Probabilistic Encryption", *Journal of Computer and System Sciences*, **28**, 1984, pp. 270-299.

[GMR] S. Goldwasser, S. Micali and C. Rackoff, "The Knowledge Complexity of Interactive Proof-Systems", *Proceedings of the 17th Annual ACM Symposium on the Theory of Computing*, 1985, pp. 291-304.

[G] J. H. Griesmer, "A Bound for Error-Correcting Codes", *IBM Journal*, **7**, 1960, pp. 532-542.

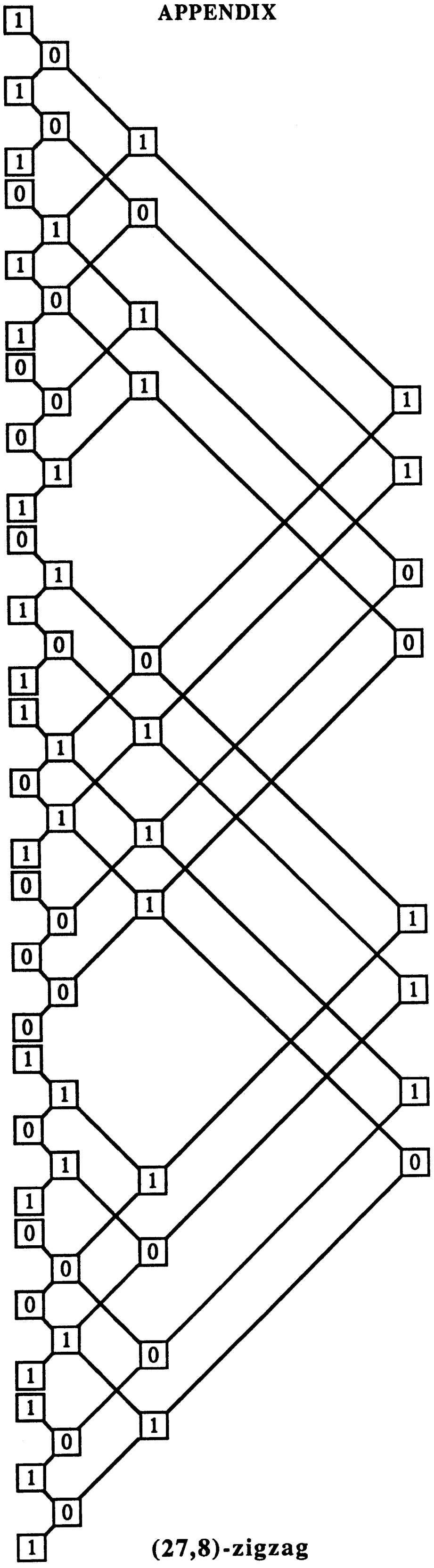
[K] E. Kranakis, *Primality and Cryptography*, John Wiley and sons, Chichester, 1986.

[MS] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes*, North-Holland, New York, 1977.

[V] U. Vazirani, *Personal communication*, 1986.

[W] S. Wiesner, "Conjugate Coding", unpublished manuscript written *ca* 1970, subsequently made available in *SIGACT/NEWS*, **15**:1, 1983, pp. 78-88.

APPENDIX



(27,8)-zigzag