A Probabilistic Error-Correcting Scheme

Scott Decatur* Oded Goldreich† Dana Ron‡

June 25, 1997

Abstract

In the course of research in Computational Learning Theory, we found ourselves in need of an error-correcting encoding scheme for which few bits in the codeword yield no information about the plain message. Being unaware of a previous solution, we came-up with the scheme presented here. Since this scheme may be of interest to people working in Cryptography, we thought it may be worthwhile to "publish" this part of our work within the Cryptography community.

Clearly, a scheme as described above cannot be deterministic. Thus, we introduce a probabilistic coding scheme which, in addition to the standard coding theoretic requirements, has the feature that any constant fraction of the bits in the (randomized) codeword yields no information about the message being encoded. This coding scheme is also used to obtain efficient constructions for the *Wire-Tap Channel* Problem.

KEYWORDS: Error Correcting Codes, Privacy, Wire-Tap Channel, Computational Learning Theory.

*DIMACS Center, Rutgers University, Piscataway, NJ 08855.
E-mail: sed@dimacs.rutgers.edu. Part of this work was done while the author was at the Laboratory for Computer Science, MIT, supported by a grant from the Reed Foundation.

†Department of Computer Science and Applied Mathematics, Weizmann Institute of Science, Rehovot, ISRAEL. E-mail: oded@wisdom.weizmann.ac.il. On sabbatical leave at LCS, MIT.

‡Laboratory for Computer Science, MIT, 545 Technology Sq., Cambridge, MA 02139.
E-mail: danar@theory.lcs.mit.edu. Supported by an NSF postdoctoral fellowship.

1 Introduction

We believe that the following problem may be relevant to research in Cryptography:

> Provide an error-correcting encoding scheme for which few bits in the codeword yield no information about the plain message.

Certainly, no deterministic encoding may satisfy this requirement, and so we are bound to seek probabilistic error-correcting encoding schemes. Specifically, in addition to the standard coding theoretic requirements (i.e., of correcting upto a certain threshold number of errors), we require that obtaining less than a threshold number of bits in the (randomized) codeword yield no information about the message being encoded.

Below we present such a probabilistic encoding scheme. In particular, the scheme can (always) correct a certain constant fraction of errors, and has the property that fewer than a certain constant fraction of bits (in the codeword) yield no information about the encoded message. Thus, using this encoding scheme over an insecure channel tampered by an adversary who can read and modify (only) a constant fraction of the transmitted bits, we establish correct and private communication between the legitimate endpoints.

The new coding scheme is also used to obtain *efficient constructions* for the *Wire-Tap Channel* Problem (cf., [9]). Related work has been pointed out to us recently by Claude Crépeau. These include [4, 7, 1, 3]. In particular, the seemingly stronger version of the problem, considered in this work, was introduced by Csiszár and Körner [4]. Maurer has shown that this version of the problem can be reduced to the original one by using bi-directional communiaction [7]. Crépeau (private comm., April 1997) has informed us that, using the techniques in [1, 3], one may obtain an alternative efficient solution to the Wire-Tap Channel Problem again by using bi-directional communiaction.

Our own motivation to study the problem had to do with Computational Learning Theory. Indeed, the solution was introduced and used in our work on *computational sample complexity* [5].

2 Formal Setting

Theorem 1 (main result): *There exist constants* $c_{\mathrm{rate}}, c_{\mathrm{err}}, c_{\mathrm{sec}} < 1$ *and a pair of probabilistic polynomial-time algorithms,* (E, D)*, so that*

1. Constant Rate*:* $|E(x)| = |x|/c_{\mathrm{rate}}$*, for all* $x \in \{0,1\}^*$*.*

2. Linear Error Correction*: for every* $x \in \{0,1\}^*$ *and every* $e \in \{0,1\}^{|E(x)|}$ *which has at most* $c_{\mathrm{err}} \cdot |E(x)|$ *ones,*

$$\mathrm{Prob}(D(E(x) \oplus e) = x) \;=\; 1$$

   where $\alpha \oplus \beta$ *denotes the bit-by-bit exclusive-or of the strings* α *and* β*. Algorithm* D *is deterministic.*

3. Partial Secrecy*: Loosely speaking, a substring containing* $c_{\mathrm{sec}} \cdot |E(x)|$ *bits of* $E(x)$ *does not yield information on* x*. Namely, let* I *be a subset of* $\{1, ..., |\alpha|\}$*, and let* α_I *denote the substring of* α *corresponding to the bits at locations* $i \in I$*. Then for every* $n \in \mathcal{N}$*,* $m = n/c_{\mathrm{rate}}$*,* $x, y \in \{0,1\}^n$*,* $I \in \{J \subset \{1, ..., m\} : |J| \leq c_{\mathrm{sec}} \cdot m\}$*, and* $\alpha \in \{0,1\}^{|I|}$*,*

$$\mathrm{Prob}(E(x)_I = \alpha) \;=\; \mathrm{Prob}(E(y)_I = \alpha)$$

   Furthermore, $E(x)_I$ *is uniformly distributed over* $\{0,1\}^{|I|}$*.*

In addition, on input x*, algorithm* E *uses* $O(|x|)$ *coin tosses.*

Items 1 and 2 are standard requirements of Coding Theory, firstly met by Justesen [6]. What is non-standard in the above is Item 3. Indeed, Item 3 is impossible if one insists that the encoding algorithm (i.e., E) is deterministic.

2.1 Proof of Theorem 1

Using a "nice" error correcting code, the key idea is to encode the information by first augmenting it by a sufficiently long random padding. To demonstrate this idea, consider an $2n$-by-m matrix M defining a constant-rate/linear-error-correction (linear) code. That is, the string $z \in \{0,1\}^{2n}$ is encoded by $z \cdot M$. Further suppose that the submatrix defined by the last n rows of M and any $c_{\mathrm{sec}} \cdot m$ of its columns is of full-rank (i.e., rank $c_{\mathrm{sec}} \cdot m$). Then, we define the following probabilistic coding, E, of strings of length n. To encode $x \in \{0,1\}^n$, we first uniformly select $y \in \{0,1\}^n$, let $z = xy$ and output $E(x) = z \cdot M$. Clearly, the error-correction features of M are inherited by E. To see that the secrecy requirement holds consider any sequence of $c_{\mathrm{sec}} \cdot m$ bits in $E(x)$. The contents of these bit locations is the product of z by the corresponding columns in M; that is, $z \cdot M' = x \cdot A + y \cdot B$, where M' denotes the submatrix corresponding to these columns in M, and A (resp., B) is the matrix resulting by taking the first (resp., last) n rows of M'. By hypothesis B is full rank, and therefore $y \cdot B$ is uniformly distributed (and so is $z \cdot M'$ regardless of x).

What is missing in the above is a specific construction satisfying the hypothesis as well as allowing efficient decoding. Such a construction can be obtained by mimicking Justesen's construction [6]. Recall that Justesen's Code is obtained by composing two codes: Specifically, an *outer* linear code over an n-symbol alphabet is composed with an *inner* random linear code.[1] The outer code is obtained by viewing the message as the coefficients of a polynomial of degree $t-1$ over a field with $\approx 3t$ elements, and letting the codeword consists of the values of this polynomial at all field elements. Using the Berlekamp-Welch Algorithm [2], one can efficiently retrieve the information from a codeword provided that at most t of the symbols (i.e., the values at field elements) were corrupted. We obtain a variation of this outer-code as follows: Given $x \in \{0,1\}^n$, we set $t \stackrel{\mathrm{def}}{=} 2n/\log_2(2n)$, and view x as a sequence of $\frac{t}{2}$ elements in GF($3t$).[2] We uniformly select $y \in \{0,1\}^n$ and view it as another sequence of $\frac{t}{2}$ elements in GF($3t$). We consider the degree $t-1$ polynomial defined by these t elements, where x corresponds to the high-order coefficients and y to the low order ones. Clearly, we preserve the error-correcting features of the original outer code. Furthermore, any $t/2$ symbols of the codeword yield no information about x. To see this, note that the values of these $t/2$ locations are obtained by multiplying a t-by-$t/2$ Vandermonde with the coefficients of the polynomial. We can rewrite the product as the sum of two products the first being the product of a $t/2$-by-$t/2$ Vandermonde with the low order coefficients. Thus, a uniform distribution on these coefficients (represented by y) yields a uniformly distributed result (regardless of x).

Next, we obtain an analogue of the inner code used in Justesen's construction. Here the aim is to encode information of length $\ell \stackrel{\mathrm{def}}{=} \log_2 3t$ (i.e., the representation of an element in GF($3t$)) using codewords of length $O(\ell)$.

[1] Our presentation of Justesen's Code is inaccurate but suffices for our purposes.

[2] Here we assume that $3t$ is a prime power. Otherwise, we use the first prime power greater than $3t$. Clearly, this has a negligible effect on the construction.

Hence, we do not need an efficient decoding algorithm, since Maximum Likelihood Decoding via exhaustive search is affordable (as $2^\ell = O(t) = O(n)$). Furthermore, any code which can be specified by $\log(n)$ many bits will do (as we can try and check all possibilities in $\mathrm{poly}(n)$-time), which means that we can use a randomized argument provided that it utilizes only $\log(n)$ random bits. For example, we may use a linear code specified by a (random) 2ℓ-by-4ℓ Toeplitz matrix.[3] Using a probabilistic argument one can show that with positive probability such a random matrix yields a "nice" code as required in the motivating discussion.[4] In the rest of the discussion, one such good Toeplitz matrix is fixed.

We now get to the final step in mimicking Justesen's construction: the composition of the two codes. Recall that we want to encode $x \in \{0,1\}^n$, and that using a random string $y \in \{0,1\}^n$ we have generated a sequence of $3t$ values in GF($3t$), denoted $x_1, ..., x_{3t}$, each represented by a binary string of length ℓ. (This was done by the outer code.) Now, using the inner code (i.e., the Toeplitz matrix) and additional $3t$ random ℓ-bit strings, denoted $y_1, ..., y_{3t}$, we encode each of the above x_i's by a 4ℓ-bit long string. Specifically, x_i is encoded by the product of the Toeplitz matrix with the vector $x_i y_i$.

Clearly, we preserve the error-correcting features of Justesen's construction [6]. The Secrecy condition is shown analogously to the way in which the Error Correction feature is established in [6]. Specifically, we consider the partition of the codeword into consecutive 4ℓ-bit long subsequences corresponding to the codewords of the inner code. Given a set I of locations (as in the secrecy requirement), we consider the relative locations in each subsequence, denoting the induced locations in the i^{th} subsequence by I_i. We classify the subsequences into two categories depending on whether the size of the induced I_i is above the secrecy threshold for the inner code or not. By a counting argument, only a small fraction of the subsequences have I_i's above the threshold. For the rest we use the Secrecy feature of the inner code to state that no information is revealed about the corresponding x_i's. Using the Secrecy feature of the outer code, we conclude that no information is revealed about x. ■

[3] A Toeplitz matrix, $T = (t_{i,j})$, satisfies $t_{i,j} = t_{i+1,j+1}$, for every i, j.

[4] The proof uses the fact that any (non-zero) linear combination of rows (columns) in a random Toeplitz matrix is uniformly distributed.

2.2 An Efficient Wire-Tap Channel Encoding Scheme

The *Wire-Tap Channel* Problem, introduced by Wyner [9], generalized the standard setting of a Binary Symmetric Channel. Recall that a Binary Symmetric Channel *with crossover probability* p, denoted BSC_p, is a randomized process which represents transmission over a noisy channel in which each bit is flipped with probability p (independently of the rest). Thus, for a string $\alpha \in \{0,1\}^n$, the random variable $\mathrm{BSC}_p(\alpha)$ equals $\beta \in \{0,1\}^n$ with probability $p^d \cdot (1-p)^{n-d}$, where d is the Hamming distance between α and β (i.e., the number of bits on which they differ). In the *Wire-Tap Channel Problem* there are two (independent) noisy channels from the sender one representing the transmission to the legitimate receiver and the second representing information obtained by an adversary tapping the legitimate transmission line and incurring some noise as well. In Wyner's work [9] the wire-tap channel introduces additional noise on top of the legitimate channel (and so may be thought of as taking place at the receiver's side). Here we consider a seemingly more difficult setting (introduced in [4]) in which the wire-tap channel is applied to the original packet being transmitted (and so may be thought of as taking place at the sender's side).

Wyner studied the information theoretic facet of the problem [9], analogously to Shannon's pioneering work on communication [8]. Below we consider the computational aspect of the problem for the special case of very noisy tapping-channel.

Theorem 2 (efficient wire-tap channel encoding): *Let (E,D) be a coding scheme as in Theorem 1 and let $\mathrm{BSC}_p(\alpha)$ be a random process which represents the transmission of a string α over a Binary Symmetric Channel with crossover probability p[5]. Then,*

1. Error Correction*: For every $x \in \{0,1\}^*$*

$$\mathrm{Prob}(D(\mathrm{BSC}_{\frac{c_{\mathrm{err}}}{2}}(E(x))) = x) \;=\; 1 - \exp(-\Omega(|x|))$$

2. Secrecy*: For every $x \in \{0,1\}^*$*

$$\sum_{\alpha \in \{0,1\}^{|E(x)|}} \left| \mathrm{Prob}(\mathrm{BSC}_{\frac{1}{2}-\frac{c_{\mathrm{sec}}}{4}}(E(x)) = \alpha) \;-\; 2^{-|E(x)|} \right|$$

is exponentially vanishing in $|x|$.

[5]The *crossover probability* is the probability that a bit is complemented in the transmission process.

Proof: Item 1 follows by observing that, with overwhelming high probability, the channel complements less than a $c_{\mathrm{err}}/2$ fraction of the bits of the codeword. Item 2 follows by representing $\mathrm{BSC}_{(1-\gamma)/2}(\alpha)$ as a two-stage process: In the first stage each bit of α is *set* (to its current value) with probability γ, independently of the other bits. In the second stage each bit which was not set in the first stage, is assigned a uniformly chosen value in $\{0,1\}$. Next, we observe that, with overwhelming high probability, at most $2\gamma|E(x)| = c_{\mathrm{sec}}|E(x)|$ bits were set in the first stage. Suppose we are in this case. Then, applying Item 3 of Theorem 1, the bits set in Stage 1 are uniformly distributed regardless of x, and due to Stage 2 the un-set bits are also random. ■

DISCUSSION: As mentioned above, the setting considered in Theorem 2 is actually due to Csiszár and Körner [4]. Clearly, a solution cannot exist unless the channel of Item 1 is more reliable than the one of Item 2. A special case of the results in [4] is that a solution always exists when the channel of Item 1 is more reliable than the one of Item 2. However, the latter result is non-constructive. In contrast, the result of Theorem 2 is constructive and efficient, but it requires a significant gap between the reliability of the two channels. In particular, the crossover probability of the channel in Item 1 (denoted $\frac{c_{\mathrm{err}}}{2}$) is typically very small (i.e., of the order of 0.01); whereas the crossover probability of the channel in Item 2 (denoted $\frac{1}{2} - \frac{c_{\mathrm{sec}}}{4}$) is typically very close to 1/2 (i.e., of the order of 0.49).

Crépeau (private comm., April 1997) has informed us that alternative solutions, which utilize bi-directional communication, may be obtained by using the techniques in [7, 1, 3]. We stress that when using bi-directional communication one can cope with an arbitrary pair of channels (and specifically the channel in the secrecy condition may be more reliable than the channel in the error-correcting condition) – see [7].

Acknowledgments

We are grateful to Moni Naor and Ronny Roth for helpful discussions. We also wish to thank Claude Crépeau for pointing out and explaining to us some related work.

References

[1] C.H. Bennett, G. Brassard, C. Crépeau, and U. Maurer. Generalized Privacy Amplification. *IEEE Transaction on Information Theory*, Volume 41, Number 6, pp. 1915-1923. November 1995.

[2] E. Berlekamp and L. Welch. Error correction of algebraic block codes. US Patent 4,633,470, 1986.

[3] C. Cachin and U.M. Maurer. Linking Information Reconciliation and Privacy Amplification, *Jour. of Cryptology*, Vol. 10, No. 2, pages 97–110, 1997.

[4] I. Csiszár and J. Körner. Broadcast channels with confidential messages. *IEEE Transactions on Info. Theory*, Vol. 24, 1978, pp. 339–348.

[5] S. Decatur, O. Goldreich and D. Ron. Computational Sample Complexity. To appear in *COLT*, 1997.

[6] J. Justesen. A class of constructive asymptotically good algebraic codes. *IEEE Trans. Inform. Theory*, 18: 652–656, 1972.

[7] U.M. Maurer. Perfect Cryptographic Security from partially independent channels. *23rd STOC*, 1991, pp. 561–571.

[8] C. E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27: 379–423, 623–656, 1948.

[9] A. D. Wyner. The wire-tap channel. *Bell System Technical Journal*, 54(8):1355–1387, Oct. 1975.