Non-asymptotic Bit Commitment Protocol Based on Noisy Channels

Valeri Korjik, IPN CINVESTAV, Mexico D.F., Mexico
E-mail: vkorjik@mail.cinvestav.mx
Kirill Morozov, State University of Telecommunications, St. Petersburg, Russia
E-mail: kirill@fem.sut.ru

Abstract – **We develop the idea to achieve the cryptographic primitive of Bit Commitment based on the existence of the Binary Symmetric Channel [1]. Our contribution (in comparison with [1]) is to extend this cryptographic protocol to a non-asymptotic case (finite length of blocks transmitted between parties). To solve this problem we derive the formulas to estimate security and reliability of this protocol and present the algorithm that optimizes the main its parameters to provide minimal complexity. Some examples are given to demonstrate that this protocol is in fact reliable and information-theoretically secure. We also discuss the problem – how to establish a noisy channel connecting parties.**

Index Terms – **Cryptographic protocols, privacy amplification, error-correcting codes, information theory.**

I. Introduction

A cryptographic protocol consists of algorithms for communications between different parties, adversaries or not. The goal of the protocol is something beyond the simple secrecy of message transmission or authentication. The list of the most important cryptographic protocols includes: secret sharing, subliminal channel, versions of digital signature, zero-knowledge proofs of knowledge, secure elections, secure multiparty computations, digital cash and others. The most simple of them that are sufficient to accomplish many complex protocols can be called *cryptographic primitives*. The two primitives known as *Bit Commitment* and *Oblivious Transfer* are elementary protocols that are sufficient to accomplish any Mental Games [2]. The current paper considers the first of these cryptographic primitives. In the case of Bit Commitment protocol, one of the parties (Alice) wants to commit to a prediction b but does not want to reveal that prediction to another party (Bob) until sometime later. Bob, on the other hand, wants to make sure that the author cannot change her mind after this party has committed to her prediction. This protocol can be achieved by the following simple algorithm:

1) Bob generates a random bit string R and sends it to Alice.
2) Alice creates the ciphertext $E_K(R,b)$ with some random key K and sends it to Bob.

(Bob can not decrypt the message, so he does not know what the bit b is.)

If Alice decides to unveil her commitment, she initiates the next protocol:

1) Alice sends the key to Bob.
2) Bob decrypts the message to reveal her the bit. He checks his random string R to verify the bit's validity.

It is easy to see that the protocol given above requires limitation on parties' computing power, otherwise Bob can find the key K before Alice decides to unveil her commitment and decrypt b using this key. Alice can also find another key K', that provides equality $E_K(R,b) = E_{K'}(R,b')$ to change her commitment to b' in place of b.

This paper (following the idea of [1]) considers a scenario where both Alice and Bob have no limitation on their computing power. It would be impossible to accomplish this protocol without another assumption. Such the extra assumption we make is that Alice and Bob are connected by Binary Symmetric Channel (BSC_p), that is a channel that will change the value of each bit with probability p as it travels from one party to the other.

II. The Derivation of the Main Non-asymptotic Relations for Security and Reliability of Bit Commitment Protocol

Let as consider first of all Bit Commitment protocol (with extension to multiple-bit string b in place of one-bit string b) proposed in [1]. Alice and Bob agree a binary linear (n, k) code C with certain minimal code distance d. There exists a BSC_p to send messages from Alice to Bob and in addition noiseless channels to exchange messages between Alice and Bob. (This assumption that might appear at first glance to be unlikely, is discussed in Section III).

After this *initialization phase* Alice and Bob have to perform the following *Protocol 1*:

1) Alice picks at random one of code words $c \in C$ and a random n-bit string m. Then Alice computes an l-bit string x such that

$$b = x \oplus h_m(c), \qquad (1)$$

where $\oplus$ is modulo two addition and $h_m(c)$ is a hash function chosen from universal$_2$ class and determined by the string m, while argument of this function is a code word c. This function maps c to an l-bit string, where l is the length of the string b intended to be committed. (In a particular case when b is one-bit string, a string x can be removed from (1) because it is always possible to choose such string m to get either 0 or 1 for $b = h_m(c)$).
2) Alice sends c over BSC_p and announces both m and x to Bob over noiseless channel.
3) Bob stores c', m and x, where c' is the received version of c corrupted by channel noises.

If Alice wants to unveil her commitment b, she initiates the next *Protocol 2*:

1) Alice sends the code word c to Bob over noiseless channel.
2) Bob reveals commitment b using (1) knowing x, m and c. He computes the Hamming distance $d_H(c,c')$ between c and c' and if $d_H(c,c') \le l_0$, where l_0 is a certain threshold, then Bob accepts b, otherwise he rejects.

The performance evaluation of this protocol can be presented by the following values:

1) The amount of Shannon's information I_0 leaking to Bob about commitment b just after the receiving c', m and x, taking into account his knowledge of the code C and BSC_p.

   It is more reasonable in fact to use another criterion of security, namely, the probability P_c of correct optimal decoding b by Bob based on his knowledge C, c', m, x and hash function $h_m(...)$. This probability can be upper bounded using Fano's inequality [3]

$$l + (1-P_c)\log_2 \frac{1-P_c}{2^l - 1} + P_c \log_2 P_c \le I_0 \qquad (2)$$

   In particular, when b is one-bit commitment, the relation (2) is transformed to the following

$$1 + (1-P_c)\log_2 (1-P_c) + P_c \log_2 P_c \le I_0 . \qquad (3)$$

2) The probability P_{ch} to occur $d_H(\tilde{c},c') \le l_0$, where $\tilde{c} \in C$, $\tilde{c} \ne c$. This means that Alice has been operating successfully to cheat Bob.
3) The probability P_{fr} to occur $d_H(c,c') > l_0$. This means that bit commitment b is rejected by Bob, while Alice is honest in her execution of the protocol.

Our purpose is to find the relations which connect the values introduced above to each other and with parameters of code C and BSC_p. Eventually we want to show that this protocol "works" in reality, that is, it provides reasonable security and reliability for finite number n of channel uses. But before of doing this we want to show (following [1] but extending this approach to the case $l > 1$) that this protocol "works" in asymptotic case, as $n \to \infty$.

We can upper bound the amount of Shannon's information I_0 leaking to Bob about the commitment b given c', m, x and with the knowledge a fact that $c \in C$, as follows [4]

$$I_0 \le \frac{2^{-(n-r-t-l)}}{\ln 2} \qquad (4)$$

where t is Renyi information that Bob gains of c' received on BSC_p,
$r = n - k$ is the number of check symbols of code C.

Let us consider the exponent in (4). We can take $t \approx n(1-H(p))$, where $H(...)$ is the entropy function, as $n \to \infty$ with high probability [4], and put $l = \alpha n$, where $0 < \alpha < 1$. (So we can consider α as the *rate of protocol*). Substituting these variables into (4) we get the following conditions to be exponential decreasing I_0 to zero, as $n \to \infty$.

$$k - n(1-H(p)) - n\alpha > 0 ,$$

that is equivalently to the following inequality

$$R_C = \frac{k}{n} > C_p + \alpha , \qquad (5)$$

where R_C is the *code rate* of chosen code C. (This fact does not contradict our intuition, because otherwise Bob could correct all errors in c', due to Shannon's theorem [3].)

We are going to show next that there exists such threshold l_0, that Bob is able to accept Protocol 2 with high probability, if Alice is honest, that is, she sends at the step 1 of Protocol 2 the same code word c as she did at the step 2 of Protocol 1, and to reject with high probability Protocol 2 if Alice sends to him another code word $\tilde{c} \ne c$ at the step 1 of Protocol 2. Moreover we will assume (in line with [1]) that dishonest Alice can send not necessary the code word c but any n-bit string w at the step 2 of Protocol 1. (Bob is unable to verify this fraud because w is sent through noisy channel).

Let us consider the case (without the loss of generality) when Alice sends to Bob at the step 2 of Protocol 1 such n-bit string w that the nearest code word $c \in C$ occurs at Hamming distance s from w, that is $d_H(c,w) = s$. It results in the following inequality (as a consequence the Hamming distance property)

$$d_H(w,\tilde{c}) \ge d - s , \qquad (6)$$

where $\tilde{c} \ne c$ is any code word.
If Alice announces either the code words c or $\tilde{c}$ at the step 1 of Protocol 2, the average Hamming distances between the string c' received by Bob and these code words will be the following, respectively

$$d_H(c,c') = s(1-p) + (n-s)p = np + s(1-2p) \qquad (7)$$

$$d_H(\tilde{c},c') \ge (d-s)(1-p) + (n-d+s)p = np + (1-2p)(d-s) \qquad (8)$$

We can let without the loss of generality that $d = np\gamma, \gamma > 0$ and $s = \beta n$, $0 < \beta < 1$.
Then the relations (7) and (8) can be rewritten as follows

$$d_H(c,c') = np + \beta n(1-2p) \qquad (9)$$

$$d_H(\tilde{c},c') \ge np + n(p\gamma - \beta)(1-2p) \qquad (10)$$

Since Bob assumes that Protocol is honest so that Alice sends c at the step 2 of Protocol 1, he can find the aver-

age Hamming distance $d_H(c,c') = np$ and then select the threshold $l_0 = n(p+\varepsilon)$, $\varepsilon > 0$ to provide the probability of false rejection of honest Protocol as small as desired, as $n \to \infty$. We can see from (9) and (10) that $d_H(c,c')$ increases, while $d_H(\tilde{c},c')$ decreases, when β increases and, $d_H(c,c') \approx d_H(\tilde{c},c')$ if $\beta = p\gamma/2$. This implies a fact that it is sufficient for Bob to reject all code words $\tilde{c}$ for $\beta \le p\gamma/2$, because he rejects c if $\beta > p\gamma/2$ and thus dishonest Alice is unable to open the commitment of both strings corresponding to c and $\tilde{c}$. As a result of this remark we obtain the following sufficient inequality that a false protocol initiated by dishonest Alice will be rejected by Bob for any $s = \beta n$

$$\varepsilon < \frac{p\gamma}{2}(1-2p) \tag{11}$$

It easy to see that this inequality is true for any symbol error probability ($0 < p < 1/2$) taken on BSC_p connecting Alice and Bob. Moreover this inequality does not contradict to the previous inequality (5). In fact, if we let that a chosen code C satisfies Varshamov-Gilbert bound [5], it results in the following inequality for large n

$$H(p\gamma) < H(p) - \alpha, \tag{12}$$

that does not contradict to inequality (11).

Thus, we have shown that Bit Commitment Protocol given above "works" in any BSC_p. What we are going to do next is to consider a non-asymptotic case, when we expect to find the optimal symbol error probability p, which provides the minimal complexity of the Protocol for some given security and reliability.

If the linear (n, k) code is shared by Alice and Bob as well as some BSC_p connecting them, Bob can select such threshold l_0, that provides a certain probability P_{fr} of *false rejection* of honest protocol using the following relation

$$P_{fr} = \sum_{i=l_0+1}^{n} \binom{n}{i} p^i (1-p)^{n-i} \tag{13}$$

Assuming that dishonest Alice can send some string w, such that the nearest code word $c \in C$ occurs at the Hamming distance s from w, and taking into account the inequality (6), we may estimate the probabilities that the announcements about either the code word c or another code word $\tilde{c}$ sent by Alice will be accepted by Bob as follows, respectively

$$P(c) = \sum_{i=0}^{s} \binom{s}{i} (1-p)^i p^{s-i} \cdot \sum_{j=0}^{l_0-i} \binom{n-s}{j} p^j (1-p)^{n-s-j} \tag{14}$$

$$P(\tilde{c}) \le \sum_{i=0}^{d-s} \binom{d-s}{i} (1-p)^i p^{d-s-i} \cdot \sum_{j=0}^{l_0-i} \binom{n-d+s}{j} p^j (1-p)^{n-d+s-j} \tag{15}$$

Thus a designer of this protocol has to fix the parameter P_{fr} which meets the desired reliability and the parameter P_c which meets the security of protocol and has to select the other parameters k, l_0, p, n in such a way to provide a proper resistance against Alice's fraud in the Protocol 2. In addition the minimal possible complexity of execution of the protocol should be provided. (This complexity first of all can be estimated by code length n). There may be also another purpose to optimize the parameters, if we want to maximize the rate of protocol $\alpha = l/n$ for some given block length n. Note that the amount of Renyi information t in (4) can be evaluated for non-asymptotic case as follows [4]

$$t = n(1 + \log_2(p^2 + (1-p)^2)). \tag{16}$$

The problem how to select the main parameters of Bit Commitment protocol can be solved by the following algorithm:

1) Fix P_{fr}, P_c, l, n and p.
2) Find the threshold l_0 from (13) given P_{fr}, p and n.
3) Find I_0 from (2) or (3) given P_c.
4) Evaluate the minimal possible value $k = n - r$, where r can be found from (4).
5) Find the minimal Hamming distance d of (n, k) linear code using the bound for BCH codes [5].
6) Compute $P(c)$ and $P(\tilde{c})$ by (14), (15) for $s = 0,1,\ldots,[d/2]$ given n, p, s and l_0.
7) Take a decision whether $P(c)$ and $P(\tilde{c})$ are enough or not to protect the Protocol 2 against Alice's fraud. If "Yes", then decrease block length n. If "Not", then try to optimize p to provide a positive answer. In the case if no one of p provides the desired security against Alice's fraud, then increase "n".

Example. Let us take $l = 1$, $p = 0.1$, $n = 1023$, $P_{fr} = 10^{-4}$, $P_c = 0.50001$. Then we can evaluate (following algorithm above): $I_0 = 2.88\ 10^{-10}$, $t = 730$ bits, $l_0 = 140$, $k = 758$, $d = 55$. The results of the evaluations $P(c)$ and $P(\tilde{c})$ as the functions of $s = 0,1,\ldots,[d/2]$ are given in Table I.

TABLE I

THE PROBABILITIES $P(c)$ AND $P(\tilde{c})$ AS THE FUNCTIONS OF $s = 0,1,\ldots,[d/2]$ GIVEN p = 0.1, n = 1023, $P_{fr} = 10^{-4}$, $P_c = 0.50001$, $l_0 = 140$, $k = 758$, $d = 55$

s	0	1	2	3	4
$P(c)$	0.9999	0.9999	0.9999	0.9998	0.9998
$P(\tilde{c})$	0.2754	0.3043	0.3345	0.3656	0.3977
s	5	6	7	8	9
$P(c)$	0.9997	0.9996	0.9995	0.9993	0.9991
$P(\tilde{c})$	0.4304	0.4634	0.4967	0.5299	0.5629
s	10	11	12	13	14
$P(c)$	0.9988	0.9985	0.9980	0.9975	0.9968
$P(\tilde{c})$	0.5953	0.6271	0.6579	0.6877	0.7162
s	15	16	17	18	19
$P(c)$	0.9960	0.9950	0.9937	0.9922	0.9903
$P(\tilde{c})$	0.7433	0.7690	0.7932	0.8157	0.8366
s	20	21	22	23	24
$P(c)$	0.9881	0.9854	0.9823	0.9785	0.9741
$P(\tilde{c})$	0.8558	0.8735	0.8895	0.9040	0.9170
s	25	26	27		
$P(c)$	0.9689	0.9629	0.9560		
$P(\tilde{c})$	0.9286	0.9389	0.9480		

We can see from this table that it does not meet the requirements to be secure against Alice's fraud. In fact, if Alice selects $s = \lfloor d/2 \rfloor = 27$, she will be able to decept any code word c or $\tilde{c}$ of her choice with high enough probability ≈ 0.95. Before moving on to a more appropriated choice of parameters it is reasonable to give a stronger definition for the protocol to be secure against Alice's fraud.

We have two possibilities to proceed. The first corresponds to "flipping a coin over telephone" based on Bit Commitment protocol [6]. Then dishonest Alice could win at each "flipping" with the knowledge of Bob's "flipping" selecting either c or $\tilde{c}$. But the probability for each trial to be accepted is

$$P_a = \tfrac{1}{2}(P(c) + P(\tilde{c})).$$

If this value is close to 1/2 (or smaller) then dishonest Alice has no advantage over honest one. Thus a designer of Bit Commitment protocol has to provide

$$\max_{0 \le s \le [d/2]} \tfrac{1}{2}(P(c) + P(\tilde{c}))$$

at most close to 1/2. These values computed for different n and the optimal p which minimize P_a are presented in Table II.

We will discuss these results in Section III.

The second possibility is to use this protocol to commit to Bob by Alice the bit, which determines one of two known unique files. In this case dishonest Alice can be interested to get not too small both probability $P(c)$ and $P(\tilde{c})$ for some $s = 0,1,\ldots,[d/2]$. Hence a designer of this protocol should provide the smallest probability $P(c) \approx P(\tilde{c})$ for s = [d/2] taken over all p.

Table II

THE PROBABILITIES $P_a = \max_{0 \le s \le [d/2]} \frac{1}{2}(P(c) + P(\tilde{c}))$ AND THE VALUES CORRESPONDING TO THEM OF OPTIMAL p COMPUTED FOR DIFFERENT n AND RESTRICTIONS:

$P_{fr} = 10^{-4}$, $P_c = 0.50001$

#	n	P_{opt}	$P_a = \max_{0 \le s \le [d/2]} \frac{1}{2}(P(c) + P(\tilde{c}))$
1	1023	0.213	0.73069
2	2047	0.187	0.67586
3	4095	0.179	0.49995
4	8191	0.187	0.49995
5	16383	0.169	0.49995

These probabilities computed for different n and for the optimal values p corresponding to them are presented in Table III. We will discuss these results in Section III.

TABLE III

THE PROBABILITIES $P(c) \approx P(\tilde{c})$ FOR $s = [d/2]$ AND CORRESPONDING TO THEM VALUES OF OPTIMAL p COMPUTED FOR DIFFERENT n AND RESTRICTIONS:

$P_{fr} = 10^{-4}$, $P_c = 0.5001$

#	n	P_{opt}	$P(c) \approx P(\tilde{c})$ for $s = [d/2]$
1	1023	0.213	0.738
2	2047	0.187	0.682
3	4095	0.179	0.268
4	8191	0.187	0.020
5	16383	0.169	$4.4 \cdot 10^{-5}$

III. DISCUSSION OF THE MAIN RESULTS AND OPEN PROBLEMS

We have presented non-asymptotic formulas to estimate reliability and security of Bit Commitment protocol based on noisy channels proposed in [1]. Using these relations and the algorithm given above we can find all the parameters of Bit Commitment protocol in order to provide the desired level of reliability and security. Tables II and III show that this protocol "works" in non-asymptotic case. In fact in the case of "flipping coin over telephone" this table gives the following results:

$$n = 4095, P_c = 0.5001, P_{fr} = 10^{-4},$$

$$\max_s \tfrac{1}{2}(P(c) + P(\tilde{c})) = 0.49995.$$

This means that Bob can "open" commitment just after the completion of Protocol 1 with the probability 0.50001 of correct bit receiving, that is practically the same as it could be got after his random guess before execution of this protocol. If Alice is honest party Bob

will reject Bit Commitment protocol with small enough probability 10^{-4}. On the other hand, if Alice tries to cheat and sends either c or $\tilde{c}$ of her choice to win in each of "flippings" she has practically no advantage over her honest behavior because the probability in each "flipping" to be accepted is at most 0.49995.

It is worth noting that for each n there is the optimal value of the symbol error probability p that provides minimum P_a.

Protocol "works" also in the case when Alice commits to Bob the bit to select one of two possible files. In fact, it results from Table III, that the probability to "open" commitment of Alice's choice without rejection of protocol by Bob is at most $4.4 \cdot 10^{-5}$ for n = 16383, $P_c = 0.5001$, $P_{fr} = 10^{-4}$. This means that dishonest Alice has practically no chances to cheat Bob.

Next thing to do is to discuss a problem how (n, k) code and noisy BSC_p connecting Alice and Bob can be chosen. It does not matter which of the parties select (n, k) code if it is given as BCH or concatenated code, because there is a possibility to check its minimal code distance very easily. To arrange BSC_p is harder problem. The BSC_p can be formed to be sure, by some third trusted party that can guarantee its properties. But in this case it can be useless to consider the protocol described above because Bit Commitment Problem could be also solved by trustee. The best solution would be to arrange BSC_p by Alice and Bob without any outside assistance. It is impossible to permit Bob to do that, because Bob is able then to select a less noisy channel and hence to "open" Bit Commitment just after the completion of Protocol 1 using an error correction procedure with chosen before (n, k) code. It seems to be more attractive to arrange the noisy channel by Alice, because she has to be interested to provide the symbol error probability p in line with an agreement between parties. Otherwise either Bob will be able to "open" Bit Commitment just after the completion of Protocol 1 using error correction procedure or the protocol will be rejected very often by Bob. But there is unfortunately another type of attack for dishonest Alice. She can generate binary symmetric channel with symbol error probability p but having memory. Then she can perhaps select such code word $\tilde{c}$ to provide the probability $P(\tilde{c})$ greater rather than (15). This results in a requirement to verify by Bob the properties of BSC_p arranged by Alice. Because Bob knows both transmitted word c and received word c' after the completion of Protocol 2, he can verify the properties of this channel using appropriated statistical criteria. (Of course this approach should be studied in more detail).

It can be a more severe situation if Alice generates discrete noise, that looks like a random one (that is, it passes all statistical criteria) but she knows the error pattern $e = c' \oplus w$ received by Bob. Then Alice can find some code word $\tilde{c}$ that gives the desired commitment and $d_H(\tilde{c}, c') \le d_H(w, c') \le l_0$ (Such code word has to be found, otherwise Bob could recover c just after the completion of Protocol 1 using the minimal Hamming distance algorithm for decoding). The feature is in the condition that Alice should not know precisely the error pattern received by Bob! The way out can be to form BSC_p as quantum channel with low intensity of random polarized photons like it was described in [7]. The difference is only that Bob has to use Bredbard base to receive modulated pulses and parties do not agree bases at all. It results in the symbol error probability $p = 0.15$, that is suitable for Bit Commitment protocol considered above. Bob cannot improve his channel, because the choice other than Bredbard base results in a greater symbol error probability. On the other hand, Alice has no reason to change her modulation for each of pulses but she can do it for certain pulses in order to cheat Bob being announced false code word $\tilde{c}$. (This approach is a subject of our further investigations.)

Eventually there may be such situation when Alice and Bob are connected with both noiseless and noisy channels, which can be distinguished by these parties. Then Bit Commitment protocol described above can be applied for any inferred level of reliability and security.

We are going to consider next Oblivious Transfer protocol [1] for non-asymptotic case to move information-theoretical secure protocols closer to being practical.

REFERENCES

[1] C. Crepeau, Efficient Cryptographic Protocols Based on Noisy Channels. *Proc. Eurocrypt'97*, Lecture Notes in Computer Science, N 1233, pp. 306–317.

[2] O. Goldreich, S. Micali and A. Wigderson, How to play any mental game, or: A completeness theorem for protocols with honest majority. *In Proc. 19th ACM Symposium on Theory of Computing*, pp. 218-229, ACM, 1987.

[3] Fano R.M. Transmission of Information, *MIT, Cambridge, Mass.*, 1961

[4] C.H. Bennett, G. Brassard, and U.M. Maurer, Generalized privacy amplification, *IEEE Trans. on IT*, vol. 41, N 6, Nov. 1995, pp. 1915-1923.

[5] F.J. Mac Williams and N.J.A. Sloan, The theory of error-correcting codes, *New-York: North-Holland*, 1977.

[6] B. Schneier, Applied Cryptography, *John Wiley*, 1996.

[7] C.H. Bennett, F. Bessette, G. Brassard, L. Salvali, and J. Smolin, Experimental quantum cryptography, *Journal of Cryptology*, vol. 5, N 1, pp. 3-28, 1992.