Stream-Based Cipher Feedback Mode in Wireless Error Channel

Yang Xiao, *Senior Member, IEEE*, Hsiao-Hwa Chen, *Senior Member, IEEE*, Xiaojiang Du, *Member, IEEE*, and Mohsen Guizani, *Fellow, IEEE*

***Abstract*—Block ciphers encrypt a fixed size block of plaintext at a time to produce a block of ciphertext. Stream ciphers encrypt stream data, such as voice or Telnet traffic, one bit or more bits at a time. The cipher feedback mode is a stream cipher implemented by a block cipher via multiple stages, and in each stage one bit or a number of bits of plaintext are encrypted at a time. In this paper, we study error performance of the stream-based cipher feedback mode in an unreliable wireless channel in terms of throughput. We model performance of the cipher feedback mode in terms of the probability that part of or the whole ciphertext can not be successfully decrypted, and the throughput by adopting the cipher feedback mode. We explicitly derive the optimal number of stages in the cipher feedback mode to achieve the optimal throughput, given an error rate in a wireless network. We also prove that for the cipher feedback mode, the whole ciphertext is successfully decrypted if and only if the whole ciphertext is successfully transmitted.**

***Index Terms*—Cipher feedback mode, encryption, Data Encryption Standard (DES), performance analysis, triple DES.**

I. Introduction

In this paper, we study the performance of a stream cipher mode called Cipher Feedback Mode (CFM) [2], [4]. Stream ciphers are used to encrypt one bit or multiple bits/bytes at a time. A stream cipher normally generates a keystream XORed with plaintext to generate the ciphertext. The plaintext is recovered at receiver by generating the same keystream to decrypt the received ciphertext.

The CFM was originally derived from block ciphers such as the Data Encryption Standard (DES) [1]. However, the CFB is not limited only to the DES scheme, and it can be used in any block cipher in general such as Triple DES and Skipjack. Block ciphers encrypt a fixed size block of plaintext at a time to produce a block of ciphertext. For example, the DES has a block size of 64 bits and Triple DES has a block size of 168 bits.

In this paper, we provide an error analysis for the CFM in a wireless error channel. We analytically model throughput and the probability that part of or the whole ciphertext can not be successfully decrypted by the CFM. We solve the optimality problem defined as follows. Given a bit error rate in a wireless network and the number of bits to encrypt each time, we want to obtain the optimal throughput with the optimal number of stages.

Stream ciphers have been widely used in wireless networks, e.g., IEEE 802.11 [5] and voice encryption in wireless telephones [6], [7]. In [6], stream ciphers were used in voice encryption in wireless telephones/messaging. In [7], stream ciphers were used in CDMA wireless networks. CFM potentially can be used in wireless networks, e.g., CFM in Telnet over wireless networks or voice encryption in wireless telephones/messaging. Furthermore, stream cipher can be used in image encryption over wireless channels. With the advancement of future wireless services, especially for resource limited devices, it is believed that CFM will be found a wide range of applications in wireless networks due to its simplicity.

The rest of the paper is outlined as follows. In Section II, we briefly introduce the DES. We explain the CFM in Section III. An application of the CFM for Telnet traffic is described in Section IV. We provide an error analysis and optimality analysis in Sections V and VI, respectively. Section VII provides performance evaluation, followed by the conclusion of this paper in Section VIII.

II. Brief Introduction of DES

The Data Encryption Standard (DES) [1] was the first official US government cipher intended for unclassified usage, and in fact it is the most widely used cryptography scheme today. The DES is defined in the Federal Information Processing Standard (FIPS) 46-3 [1] and is also defined in the ANSI standard X3.92. Based on the algorithm Lucifer developed by IBM in the early 1970s, the DES, a symmetric algorithm, was designed by IBM and adopted by National Security Agency (NSA) (later renamed as NIST) in 1976, adopting 64-bit block size, a 56-bit key, and 16-round Feistel cipher. Similar to any symmetric scheme, the algorithm is assumed to be known to everybody, but the key is only shared by sender and receiver. The decryption uses the same key to convert the ciphertext back to the plaintext. In each of 16 rounds, the fundamental operations of permutation and substitution are mixed so that every bit of the ciphertext depends on every bit of the data plus every bit of the key. ANSI X9.17 can be used for managing the keys used by the DES.

The DES protects unclassified information against a number of passive and active attacks on, for examples, electronic funds transfer, privacy protection of personal information, personal authentication, password protection, and access control [1].

The development of DES was controversial initially, with classified design elements, a relatively short key length, and suspicions about a NSA backdoor. The DES consequently came under intense academic scrutiny, motivated by the modern understanding of block ciphers and their cryptanalysis.

Manuscript received November 15, 2007; revised February 25, 2008 and May 26, 2008; accepted August 6, 2008. The associate editor coordinating the review of this letter and approving it for publication was W. Zhuang.

Y. Xiao is with the Dept. of Computer Science, Univ. of Alabama, AL 35487 USA (e-mail: yangxiao@ieee.org).

H. H. Chen is with the Dept. of Engineering Science, National Cheng Kung Univ., Taiwan (e-mail: hshwchen@ieee.org).

X. Du is with the Dept. of Computer Science, North Dakota State Univ., ND 58105, USA (e-mail: dxj@ieee.org).

M. Guizani is with the Dept. of Computer Science, Western Michigan Univ., MI 49008, USA (e-mail: mguizani@ieee.org).

Digital Object Identifier 10.1109/TWC.2009.071278

III. Cipher Feedback Mode

The Cipher Feedback (CFB) mode [2] is one mechanism to implement a stream cipher by a block cipher. A stream cipher is one that encrypts a stream data, such as voice, video, or Telnet traffic, one bit/byte at a time. The autokeyed Vigenere cipher and the Vernam cipher are two examples of the stream ciphers. The CFB is not limited to the DES scheme, and it can be used in any block cipher in general such as Triple DES and Skipjack.

A. Encryption

In the CFM mode, a plaintext (P) is divided into M units as $P = P_1||P_2||P_3||\dots||P_M$, and each unit has s bits ($s = 1, 2, \dots, 64$), as shown in Fig. 1a. In other words, s bits of data are encrypted at a time. An initialization vector (IV) of length L ($L = 64$ in Fig. 1) is used as the initial input block of the DES encryption for P_1, and the (left) most significant s bits of the output block of the DES encryption are XORed with the s-bit plaintext unit (P_1) such that the cipher text C_1 is produced. Let F denote the function of obtaining the (left) most significant s bits, and K is the key of DES. We have $C_1 = F(DES_K(IV)) \oplus P_1$.

In general, the initial input block is an IV, and then the input block, called the shift register, is changed each time. The input block H_j ($j = 1, \dots, M$) is encrypted with DES and an output block is produced. The (left) most significant s bits of the output block from the DES encryption are XORed with the s-bit plaintext unit P_j. For $j = 1, \dots, M$, we have

$$C_j = F\big[DES_K(H_j)\big]\oplus P_j \tag{1}$$

$$H_1 = IV \tag{2}$$

It is noted from (1) that the unused ($64 - s$) bits of the DES output block are discarded. The next input block is created by discarding the (left) most significant s bits of the previous input block, shifting the remaining ($64 - s$) bits to the left and then inserting s bits of cipher text just produced in the encryption operation. Let LShift(H, s) denote the function of left shifting H with s shifts such that the (left) most significant s bits are discarded. For $j = 2, \dots, M$, we have

$$H_j = L\text{Shift}\big(H_{j-1}, s\big)\oplus C_{j-1} \tag{3}$$

This process continues until the entire plain text message (P) has been encrypted.

B. Decryption

Decryption algorithm is shown in Fig. 1b, in which s bits of data are decrypted at a time. The initial input block is the same as encryption listed in (2) and (3). The DES encryption is still used to encrypt the input block H_j ($j = 1, \dots, M$) to the output block. The (left) most significant s bits of the output bock of the DES encryption are XORed with the s-bit ciphertext unit C_j to produce the plaintext block P_j. Therefore, for $j = 1, \dots, M$, we have

$$P_j = F\Big[DES_K(H_j)\Big]\oplus C_j \tag{4}$$

We can easily prove (4) as follows: $F(DES_K(H_j)) \oplus C_j = F(DES_K(H_j)) \oplus [F(DES_K(H_j)) \oplus P_j] = P_j$.

It is noted in (4) that the unused ($64 - s$) bits of the DES output block are discarded. The next input block is created by discarding the (left) most significant s bits of the previous input block, shifting the remaining ($64 - s$) bits to the left and then inserting s bits of cipher text just produced in the encryption operation. This process continues until the entire cipher text message ($C = C_1||C_2||C_3||\dots||C_M$) has been decrypted.

C. Discussion

With encryption and decryption of the CFM mode, we see that the CFM mode is an implementation of a stream cipher with a block cipher. A stream cipher is useful for stream data. It is noted that in both encryption and decryption of the CFM mode as shown in Fig. 1, only DES encryption is used, but not the DES decryption. In fact, the DES is for encrypting the input block so that different s-bit outputs are produced.

IV. RFC: TELNET Encryption: DES 64 BIT Cipher Feedback

One application of the CFM mode is found in Telnet in RFC 2952 [4], in which $s = 64$. Let $V[i]$ be the initial 64-bit vector, $V[n]$ and $D[n]$ denote the n-th 64-bit vector and the n-th chunk of 64 bits of plaintext to encrypt (decrypt), respectively. Letting $O[n]$ denote the n-th chunk of 64 bits of encrypted (decrypted) data, we have

$$V[0] = DES_K\big(V[i]\big) \tag{5}$$

$$O[n] = D[n] \oplus V[n] \tag{6}$$

$$V[n+1] = DES_K\big(O[n]\big) \tag{7}$$

V. Error Analysis for the CFM

Assume that we want to calculate bit error rate (BER) in a wireless network with a Gaussian channel model. The error rate is denoted as R_b, which can be calculated via previous frames or packets exchanges. For the CFM, if one bit error happens, it may influence several transmission units as long as the error bit(s) remain in the shift register of the receiver. Let $P_u(s)$ denote the probability that one unit of ciphertext is corrupted. We have

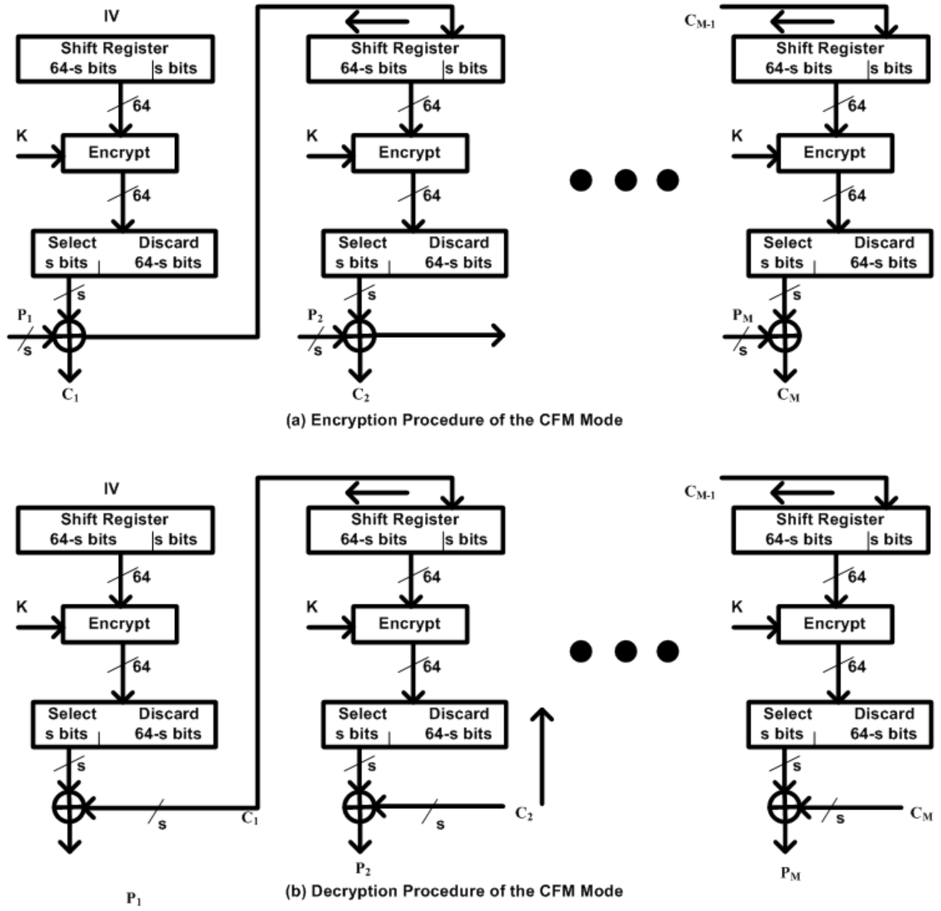
$$P_u(s) = 1 - \big(1 - R_b\big)^s \tag{8}$$

Let A_j ($j = 1, \dots, M$) denote the event that the shift-register j has errors caused by error propagation in the previous stages, and B_j ($j = 1, \dots, M$) denote the event that C_j has transmission errors. Let $\Pr(A_j)$ and $\Pr(B_j)$ denote the corresponding probabilities, respectively. As stated earlier, L is the length of IV. For $j = 1, \dots, M$, we have

$$\Pr(A_j) = \begin{cases} 0, & j = 1 \\ P_u[(j-1)s], & 0 < (j-1)s \le L \\ P_u(\lceil \frac{L}{s} \rceil\, s), & (j-1)s > L \end{cases} \tag{9}$$

$$\Pr(B_j) = P_u(s) \tag{10}$$

Let D_j ($j = 1, \dots, M$) denote the event that the ciphertext C_j can not be successfully decrypted, and D denote the event that the whole ciphertext C can not be successfully decrypted. To calculate $Pr(D_j)$ is difficult due to the following reasons:



(a) Encryption Procedure of the CFM Mode

(b) Decryption Procedure of the CFM Mode

Fig. 1. Illustration of the CFM mode.

1) If there are errors in both the shift register and transmitted ciphertext, there is a probability that the final ciphertext can be successfully decrypted due to the operation of Xor;
2) Encryption and selection add more complexity since it is difficult to see the distribution of errors after the encryption. However, we can obtain $P = Pr(D)$ later.

A_2 is true if and only if B_1 is true. A_3 is true if and only if B_1 is true and $s < L$, or B_2 is true. Let Q_j denote the event that the first $L - s$ bits of register j have error. Therefore, for $j = 2, \dots, M$, we have $A_1 = \phi$, $A_j = Q_j \cup B_{j-1}$, and $Q_j \subseteq \cup_{k=1,\dots,j-2} B_k$. From the above discussion, we have $\cup_{j=1,\dots,M} A_j = \cup_{j=1,\dots,M-1} B_j$. We have $\Pr\left(\cup_{j=1,\dots,M} A_j\right) = \Pr\left(\cup_{j=1,\dots,M-1} B_j\right) = P_u[(M-1)\,s]$. Since $\cup_{j=1,\dots,M} A_j$ and B_M are independent, we have $\Pr(\cup_{j=1,\dots,M}(A_j \cup B_j)) = \Pr(\cup_{j=1,\dots,M} A_j \cup B_M) = \Pr\left(\cup_{j=1,\dots,M} A_j\right) + \Pr\left(B_M\right) - \Pr\left(\cup_{j=1,\dots,M} A_j\right)\Pr\left(B_M\right) = \Pr\left(\cup_{j=1,\dots,M} A_j\right) + P_u(s) - \Pr\left(\cup_{j=1,\dots,M} A_j\right) P_u(s) = P_u(Ms)$.

However, we cannot claim that $\left(\cup_{j=1,\dots,M}(A_j \cup B_j)\right) = D$ is true due to the reason that if there are errors in both the shift register and transmitted ciphertext, there is a probability that the final ciphertext can be successfully decrypted due to the operation of Xor.

For two arbitrary events X and Y, the claim "If X is true, then Y is true" is denoted as "$X \Rightarrow Y$" or "$Y \Leftarrow X$"; the claim "X is true if only if Y is true" is denoted as "$X \Leftrightarrow Y$"; let $\overline{X}$ denote the reversed event of X; if "$X \Rightarrow Y$", then we have "$\overline{X} \Leftarrow \overline{Y}$".

Lemma 1: for $j = 2, \dots, M$, we have

$$\overline{D_j}\prod\nolimits_{i=1}^{j-1}\overline{D_i} \Leftrightarrow \overline{B_j}\prod\nolimits_{i=1}^{j-1}\overline{B_i} \tag{11}$$

$$D_j\prod_{i=1}^{j-1}\overline{D_i} \Leftrightarrow B_j\prod_{i=1}^{j-1}\overline{B_i}. \tag{12}$$

Proof: We prove the above equations by natural induction.

We first prove the case when $j = 2$. From Fig. 1b, we have $\overline{D_1} \Leftrightarrow \overline{B_1}$, i.e., $D_1 \Leftrightarrow B_1$. Therefore, we have $D_2\overline{D_1} \Leftrightarrow D_2\overline{B_1} \Leftrightarrow B_2\overline{B_1}$, and $\overline{D_2}\,\overline{D_1} \Leftrightarrow \overline{D_2}\,\overline{B_1} \Leftrightarrow \overline{B_2}\,\overline{B_1}$.

Assume that the case for $M - 1 \geq j = k > 2$ is true, i.e.,

$$\overline{D_k}\prod_{i=1}^{k-1}\overline{D_i} \Leftrightarrow \overline{B_k}\prod_{i=1}^{k-1}\overline{B_i} \tag{13}$$

$$D_k\prod_{i=1}^{k-1}\overline{D_i} \Leftrightarrow B_k\prod_{i=1}^{k-1}\overline{B_i} \tag{14}$$

When $j = k + 1$, we have

$$\begin{aligned}&\overline{D_{k+1}}\prod_{i=1}^{k}\overline{D_i} \Leftrightarrow \overline{D_{k+1}}\left(\overline{D_k}\prod_{i=1}^{k-1}\overline{D_i}\right)\\ &\Leftrightarrow \overline{D_{k+1}}\left(\overline{B_k}\prod_{i=1}^{k-1}\overline{B_i}\right) \Leftrightarrow \overline{B_{k+1}}\prod_{i=1}^{k}\overline{B_i}\end{aligned} \tag{15}$$

and

$$\begin{aligned}&D_{k+1}\prod_{i=1}^{k}\overline{D_i} \Leftrightarrow D_{k+1}\left(\overline{D_k}\prod_{i=1}^{k-1}\overline{D_i}\right)\\ &\Leftrightarrow D_{k+1}\left(\overline{B_k}\prod_{i=1}^{k-1}\overline{B_i}\right) \Leftrightarrow B_{k+1}\prod_{i=1}^{k}\overline{B_i}\end{aligned} \tag{16}$$

■

Theorem 1: For cipher feedback mode, the whole ciphertext is successfully decrypted if and only if the whole ciphertext is successfully transmitted.

Proof: This is to prove $\overline{D} \Leftrightarrow \overline{B}$. Since $D = \cup_{j=1}^{M}\left(D_j\prod_{i=1}^{j-1}\overline{D_i}\right)$ and $B = \cup_{j=1}^{M}\left(B_j\prod_{i=1}^{j-1}\overline{B_i}\right)$, based on Lemma 1, we have $D \Leftrightarrow B$, i.e., $\overline{D} \Leftrightarrow \overline{B}$. ■

Based on Theorem 1, we have

$$P = Pr(D) = Pr(B) = P_u(Ms) \tag{17}$$

VI. Optimality Study

Assume that a stop-and-wait protocol is adopted. In other words, the sender sends an encrypted frame/packet to the receiver, and waits for an acknowledgement frame/packet from the receiver before sending the next frame/packet.

A. Problem Definition

Therefore, our optimization problem can be defined as follows.

The Optimization Problem: given a bit error rate R_b in a wireless network and the number of (s) bits to encrypt each time, we want to obtain the optimal throughput with the number of stages, i.e., M.

B. Optimal Analysis

Let R, H, L_{ACK}, T_P, and $T_{\text{Propagation}}$ denote the transmission rate in bps (bits per second), the frame/packet overhead in bits including headers (Medium Access Control header, IP header, TPC/UDP header) and trailer, the length of the acknowledgement frame/packet in bits, the transmission time for the physical header, and the propagation delay, respectively. The normalized throughput (T) is obtained as the ratio between successful transmission data and the total data, and can be written as follows:

$$T = \frac{O_1 M\left[(1-R_b)^s\right]^M}{M + O_2}, \tag{18}$$

where

$$O_1 = \left(1-R_b\right)^{2H+L_{ACK}} \tag{19}$$

$$O_2 = \frac{2H + L_{ACK} + 2RT_P + 2RT_{\text{Propagation}}}{s} \tag{20}$$

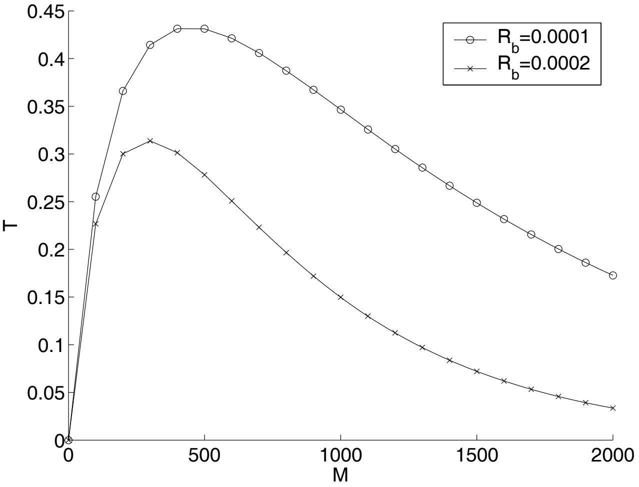


Fig. 2. Normal throughput versus M.

We can calculate the first order directive of the normalized throughput and let it to be zero, i.e., $\frac{\partial T}{\partial M} = 0$. We obtain the optimal number of stages (M) of the CFM explicitly as the integer value of the following equations, or

$$M_{optimal} = \frac{-O_2\ln\left(1-R_b\right)^s - \sqrt{\left[O_2\ln\left(1-R_b\right)^s\right]^2 - 4O_2\ln\left(1-R_b\right)^s}}{2\ln\left(1-R_b\right)^s} \tag{21}$$

VII. Performance Evaluation

In this section, we illustrate some numerical results for the performance of CFM under different parameters.

A. Optimality of M

Figs. 2 and 3 use the following parameters: $L_{ACK} = 0$, $H = 24 \times 8$, $2RT_P + 2RT_{\text{Propagation}} = 1600$, and $s = 8$. Fig. 2 shows the normalized throughput T over the number of stages M with two different BER values. As illustrated in the figure, an optimal T value exists, given a BER value. Furthermore, as BER increases, both the optimal M value and optimal T value decrease. Fig. 3 shows the throughput T over BER under the optimal M value and other M values. As illustrated in the figure, an optimal M value does provide the best throughput. We observe that the optimal throughput is an upper-bound or envelop of the throughput with other chosen M values. Furthermore, as BER increases, the optimal throughput decreases.

Fig. 4 shows the optimal M value over BER. As illustrated in Fig. 4, the optimal M value decreases as the BER increases.

B. Effect of s

Fig. 5 uses the following parameters: $L_{ACK} = 0$, $H = 24 \times 8$, $2RT_P + 2RT_{\text{Propagation}} = 1600$, and $R_b = 0.0001$. Fig. 5 shows the optimal M value versus s, the number of bits to encrypt at a time. As illustrated in Fig. 5, the optimal M value decreases as the value of s increases.

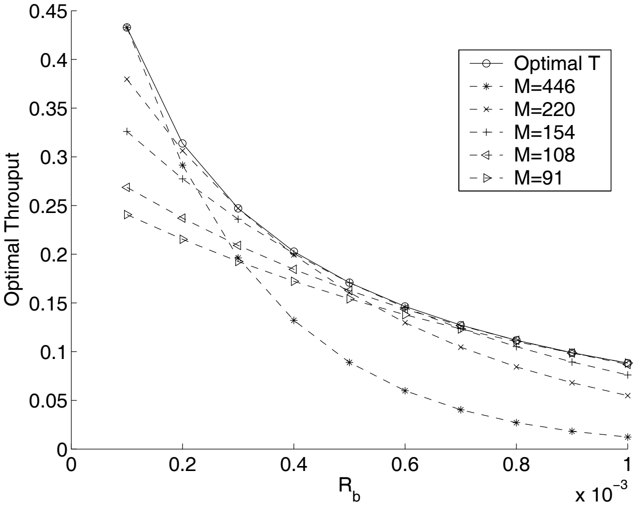


Fig. 3. Optimal throughput versus BER.

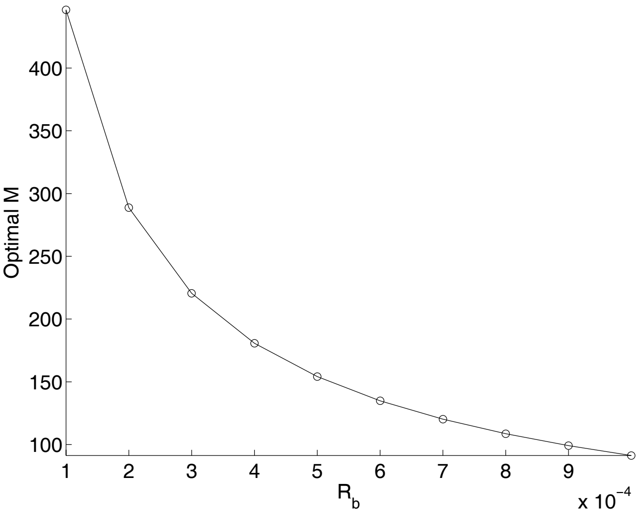


Fig. 4. Optimal M versus BER.

VIII. Conclusions

In this paper, we provided an error analysis of the stream-based cipher feedback mode in an unreliable wireless channel. We have analytically modeled throughput and probability that a part of or the whole ciphertext can not be successfully decrypted by the CFM. We have solved the following optimality problem: given a bit error rate and the number of bits to encrypt each time, we want to obtain the optimal throughput with the optimal number of stages. The following observations have been made from this paper.

- Given a BER value, an optimal T value always exists. Furthermore, as BER increases, both the optimal M value and optimal T value decrease.
- An optimal M value does provide the highest throughput. It is observed that the optimal throughput is an upper-bound or envelop of the throughput with other chosen M values.
- The optimal M value decreases as the value of s increases.

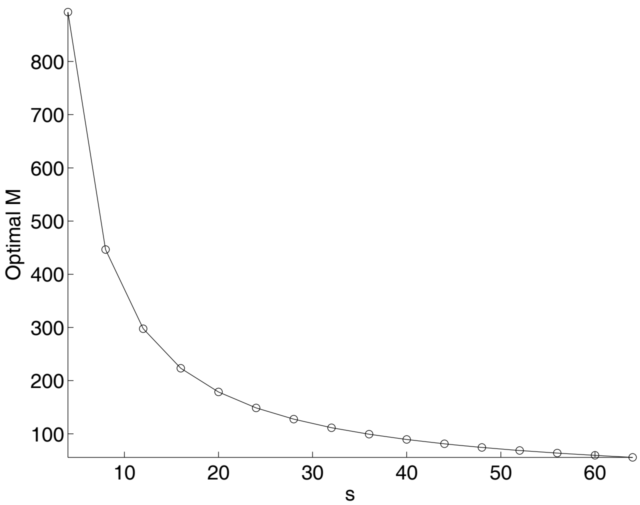


Fig. 5. Optimal M versus s.

The study of the work is beneficial to obtain design parameters for the cipher feedback mode in wireless networks. In this paper, a collision-free medium access control (MAC) protocol is used for illustration purpose. Our future works include considering more complex MAC protocols, and applying this work into sliding-window protocols in TCP/IP.

Acknowledgment

This work was supported in part by 1) the US National Science Foundation (NSF) under the grants CNS-0716211, CNS-0737325, CNS-0829827, CNS-0721907, and CNS-0709268, 2) Army Research Office under grants W911NF-07-1-0250 and W911NF-08-1-0334, and 3) Taiwan National Science Council under the grant NSC 97-2219-E-006-004.

Dr. Y. Xiao would like to thank his PhD student, Mr. Xiannuan Liang, for providing comments and suggestions for an early version of this paper.

References

[1] FIPS Publication 46-3, “Data Encryption Standard (DES)," U.S. DoC/NIST, Oct. 25, 1999.

[2] FIPS Publication 81, “DES Modes of Operation," U.S. DoC/NIST, December 1980.

[3] FIPS Publication 800-38A, “Recommendation for Block Cipher Modes of Operation: Methods and Techniques," U.S. DoC/NIST, 2001.

[4] T. Ts’o, “RFC 2952: Telnet Encryption: DES 64 bit Cipher Feedback," Request for Comments, Network Working Group, the Internet Society, 2000.

[5] IEEE 802.11-1999, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specification, Standard, IEEE, Aug. 1999.

[6] G. Rose, “A stream cipher based on linear feedback over GF(2^8)," *Book Series Lecture Notes in Computer Science*, vol. 1438/1998, pp. 135-146, July 07, 2006.

[7] N. Hamdy, et al., “MANAGE1: new stream cipher for data encryption in CDMA wireless networks," in *Proc. 2006 International Conf. Computer Eng. Syst.*, Nov. 2006.