Hindawi Publishing Corporation
Journal of Applied Mathematics
Volume 2014, Article ID 529736, 10 pages
http://dx.doi.org/10.1155/2014/529736

Research Article

Biclique Cryptanalysis on the Full Crypton-256 and mCrypton-128

Junghwan Song, Kwanhyung Lee, and Hwanjin Lee

Department of Mathematics, Hanyang University, Seoul 133-791, Republic of Korea

Correspondence should be addressed to Kwanhyung Lee; kwanist@hanyang.ac.kr

Received 8 November 2013; Accepted 21 January 2014; Published 27 March 2014

Academic Editor: Jongsung Kim

Biclique cryptanalysis is an attack which reduces the computational complexity by finding a biclique which is a kind of bipartite graph. We show a single-key full-round attack of the Crypton-256 and mCrypton-128 by using biclique cryptanalysis. In this paper, 4-round bicliques are constructed for Crypton-256 and mCrypton-128. And these bicliques are used to recover master key for the full rounds of Crypton-256 and mCrypton-128 with the computational complexities of $2^{253.78}$ and $2^{126.5}$, respectively. This is the first known single-key full-round attack on the Crypton-256. And our result on the mCrypton-128 has superiority over known result of biclique cryptanalysis on the mCrypton-128 which constructs 3-round bicliques in terms of computational time complexity.

1. Introduction

The block cipher Crypton is one of candidates for the Advanced Encryption Standard (AES) in 1998 [1]. The cipher has been revised to Crypton V1.0 in FSE'99 [2]. Crypton is a 12-round and 128-bit block cipher that supports key sizes up to 256 bits. A miniversion of Crypton, mCrypton, is a 64-bit block cipher with three key size versions (64 bits, 96 bits, and 128 bits) [3]. mCrypton is a 64-bit lightweight block cipher designed to be used in low-cost and resource-constrained applications. Both of them have been designed based on the block cipher square [4]. The cipher has been designed to be resistant to differential and linear cryptanalysis. Therefore it has been assumed that the above two ciphers also have the property of resisting those attacks.

However, a related-key impossible differential attack on 9 rounds of Crypton-256 has been shown by Wei et al. in 2011 [5]. For mCrypton, a related-key rectangle attack on 8 rounds of mCrypton-128 has been shown by Park in 2009 [6]. In 2011, Mala et al. showed a related-key impossible differential attack on 9 rounds of mCrypton-96 and mCrypton-128 [7]. The summary of attacks on Crypton-256 and mCrypton-128 is described in Tables 1 and 2, respectively.

In ASIACRYPT 2011, Bogdanov et al. introduce a biclique cryptanalysis, which is a meet-in-the-middle attack with a biclique and the attack is efficient compared to brute force key search. They show two techniques of constructing bicliques for AES in [8]. One is from independent related-key differentials, which is called independent biclique and the other is from interleaving related-key differentials.

The biclique attack by using independent related-key differentials consists of two parts. The first part constructs an independent-biclique and the second is called matching with precomputations. In Section 2, we describe an overview of the steps of biclique cryptanalysis. The detailed technique to recover the 256-bit master key with computational complexity in $2^{253.78}$ is presented in Section 4. And in Section 5, the 128-bit key is recovered with computational complexity in $2^{126.5}$.

2. Biclique Cryptanalysis

In the biclique cryptanalysis, the biclique, which is a kind of a bipartite graph improve the computational efficiency of computation. First we will briefly describe biclique. The block cipher is considered as the composition of two subciphers: $e = f \circ g$. Consider the subcipher f maps an internal state S to the ciphertext $C : f_K(S) = C$, where K is a secret key of e. The subcipher f maps 2^d internal states $\{S_0, \ldots, S^d_{2-1}\}$ to 2^d

Table 1: Summary of the attacks of Crypton-256.

Rounds	Attack	Complexities		References
		Data	Time	
6	Square	2^{32}	2^{56}	[9]
6	Imp. Diff.	2^{91}	2^{124}	[10]
8	Stochastic	2^{112}	2^{112}	[11]
8	Trunc. Diff.	2^{126}	$2^{126.2}$	[12]
9	Rel. Imp. Diff.	2^{105}	2^{243}	[5]
12	Biclique	2^{100}	$2^{253.78}$	This paper

Rel.: related key, Imp.: impossible, Diff.: differential, Trunc.: truncated.

Table 2: Summary of the attacks of mCrypton-128.

Rounds	Attack	Complexities		References
		Data	Time	
8	Rel. Rec.	2^{46}	2^{46}	[6]
9	Rel. Imp. Diff.	$2^{59.7}$	$2^{66.7}$	[7]
12	Biclique	2^{48}	$2^{126.56}$	[13]
12	Biclique	2^{52}	$2^{126.5}$	This paper

Rel.: related key, Imp.: impossible, Diff.: differential, Rec.: rectangle.

ciphertexts $\{C_0, \ldots, C^d_{2-1}\}$ with 2^{2d} keys $\{K_{\langle i,j\rangle}\}$, which are components of the following $2^d \times 2^d$ matrix:

$$\left[K_{\langle i,j\rangle}\right] = \begin{bmatrix} K_{\langle 0,0\rangle} & K_{\langle 0,1\rangle} & \cdots & K_{\langle 0,2^d-1\rangle} \\ \vdots & & & \\ K_{\langle 2^d-1,0\rangle} & K_{\langle 2^d-1,1\rangle} & \cdots & K_{\langle 2^d-1,2^d-1\rangle} \end{bmatrix}. \quad (1)$$

This 3-tuple $\{(C_i, S_j, K_{\langle i,j\rangle})\}$ is called a *d-dimensional biclique*, if

$$C_i = f_{K_{\langle i,j\rangle}}\left(S_j\right) \quad \forall i, j \in \left\{0, \ldots, 2^d - 1\right\}. \quad (2)$$

In other words, as illustrated in Figure 1, a biclique is a complete bipartite graph with $\{S_j\}$ and $\{C_i\}$ as the two parts of vertices connected to 2^{2d} edges, where each edge has degree 2^d.

Now we introduce the biclique cryptanalysis.

2.1. Attack Procedure. The biclique attack procedure consists of the following phases.

Key Partitioning. The key space is partitioned into 2^{k-2d} groups of 2^{2d} keys each, where k is the bit length of the secret key. Each key in the set is indexed as an element of a $2^d \times 2^d$ matrix: $[K_{\langle i,j\rangle}]$.

Biclique Constructing. For each group of keys, build a structure of 2^d ciphertexts $\{C_0, \ldots, C_{2^d-1}\}$, 2^d intermediate states

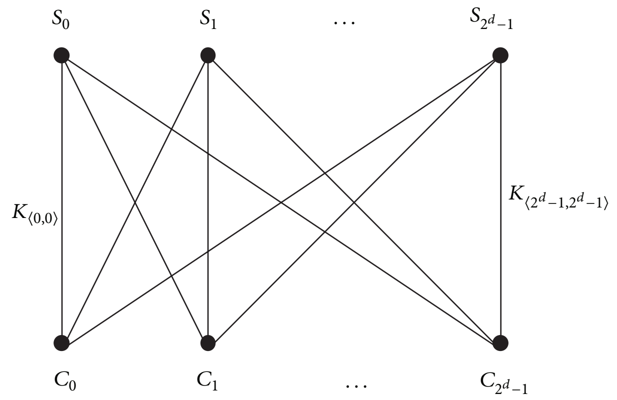


Figure 1: d-dimensional biclique.

$\{S_0, \ldots, S_{2^d-1}\}$, and $[K_{\langle i,j\rangle}]$ such that for all $i, j \in \{0, 1, \ldots, 2^d - 1\}$ the relation (2) is satisfied.

Data Collecting. An adversary obtains the plaintexts $\{P_i\}$ from the ciphertexts $\{C_i\}$ through the decryption oracle.

Key Testing. The secret key, which is an adversary try to recover, maps the plaintext P_i to the intermediate state S_j. From this fact, an adversary checks the following equation:

$$\exists i, j : P_i \xrightarrow[g]{K_{\langle i,j\rangle}} S_j, \quad (3)$$

which proposes a key candidate. Note that $P_i \xrightarrow[g]{K_{\langle i,j\rangle}} S_j$ implies that each P_i is encrypted to S_j with key $[K_{\langle i,j\rangle}]$ (i.e., $S_j = g_{K_{\langle i,j\rangle}}(P_i)$). If there is no right key satisfying (3) in the selected key group, then another key group is chosen and repeats the above process.

2.2. Biclique Construction by Independent Related-Key Differentials. In biclique cryptanalysis, there are two methods to construct a biclique. One is using independent related-key differentials and the other is using interleaving related-key differential trails. In this paper, we focus on the first of two methods, to construct biclique as described in [8].

Suppose that a secret key $K_{\langle 0,0\rangle}$ maps an intermediate state S_0 to a ciphertext C_0. Then we consider the following two types of 2^d related-key differentials with respect to $S_0 \xrightarrow[f]{K_{\langle 0,0\rangle}} C_0$.

Δ_i-Differentials. This is a related-key differential trail where the input difference is 0 and the output difference is Δ_i under a key difference Δ_i^K:

$$0 \xrightarrow[f]{\Delta_i^K} \Delta_i \quad \text{with } \Delta_0^K = 0,\ \Delta_0 = 0. \quad (4)$$

∇_j*-Differentials.* This is a related-key differential trail where the input difference is ∇_j and the output difference is 0 under a key difference ∇_j^K:

$$\nabla_j \xrightarrow[f]{\nabla_j^K} 0 \quad \text{with } \nabla_0^K = 0,\ \nabla_0 = 0. \tag{5}$$

The 3-tuple $(S_0, C_0, K_{\langle 0,0\rangle})$ conforms to both sets of differentials at the same time. If the two key differential trails, Δ_i-differentials and ∇_j-differentials, do not share active nonlinear components, then the tuple also conforms to 2^{2d} combined (Δ_i, ∇_j)-differentials:

$$\nabla_j \xrightarrow[f]{\Delta_i^K \oplus \nabla_j^K} \Delta_i \quad \text{for } i, j \in \left\{0, \dots, 2^d - 1\right\}. \tag{6}$$

This combined (Δ_i, ∇_j)-differentials is derived from property of S-box switch [14] and sandwich attack [15]. By using the combined differentials, an adversary reduces the computational complexity. The construction of a biclique requires less than $2 \cdot 2^d$ computations of f.

2.3. Matching with Precomputations. The technique of matching with precomputations is an efficient method to check (3) in biclique cryptanalysis procedure. Let v be some selected bytes of an internal state between $\{P_i\}$ and $\{S_j\}$. The flow of matching with precomputation procedure is as the following. First, an adversary computes and stores in memory the following for all i, j:

$$\begin{gathered} \forall i = 0, 1, \dots, 2^d - 1, \quad P_i \xrightarrow{K_{\langle i,0\rangle}} \overrightarrow{v}_i, \\ \forall j = 0, 1, \dots, 2^d - 1, \quad \overleftarrow{v}_j \xrightarrow{K_{\langle 0,j\rangle}} S_j. \end{gathered} \tag{7}$$

Then for particular i and j, which is not in stored memory, the adversary checks the matching at v by recomputing only those parts of the cipher which differ from the stored one.

3. Description the Crypton and mCrypton

In this section, we describe Crypton and mCrypton, briefly.

3.1. Description of Crypton. Crypton is a 128-bit block cipher supports key sizes up to 256 bits. The standard number of rounds is 12. Let us represent the 128-bit block A as a 4 × 4 matrix of bytes:

$$A = \begin{pmatrix} a_{0,3} & a_{0,2} & a_{0,1} & a_{0,0} \\ a_{1,3} & a_{1,2} & a_{1,1} & a_{1,0} \\ a_{2,3} & a_{2,2} & a_{2,1} & a_{2,0} \\ a_{3,3} & a_{3,2} & a_{3,1} & a_{3,0} \end{pmatrix}. \tag{8}$$

Crypton uses component functions, γ, π, τ, and σ.

Nonlinear Substitution γ. γ_o and γ_e are bytewise nonlinear substitutions which are applied to odd rounds and even rounds, respectively.

Bit Permutation π. π_o and π_e are linear transformations for odd rounds and even rounds, respectively. The two bit permutations mix each byte column of 4 × 4 byte array using four masking bytes m_i.

We denote "$\cdot$" and "$\oplus$" bitwise logical operations for AND and XOR, respectively. π_o is given as follows:

$$B_{i,j} = \oplus_{k=0}^{3} \left(A_{k,j} \cdot m_{(i+j+k) \bmod 4}\right), \tag{9}$$

and π_e is given as shown below:

$$B_{i,j} = \oplus_{k=0}^{3} \left(A_{k,j} \cdot m_{(i+j+k+2) \bmod 4}\right). \tag{10}$$

Byte Transposition τ. τ is a byte transposition; it simply moves the byte at (i, j) position to (j, i) position; that is, $B = \tau(A) \Leftrightarrow b_{i,j} = a_{j,i}$.

Key Addition σ. σ_K is a bitwise key XOR with key K. Let K_i be the ith encryption round key derived from a user key K using the key schedule.

The block cipher Crypton can be described as $\phi_e \circ \rho_e K_{12} \circ \rho_o K_{11} \circ \cdots \circ \rho_e K_2 \circ \rho_o K_1 \circ \sigma K_0$, where odd round function $\rho_o K$ and even round function $\rho_e K$ are defined by $\rho_o K = \sigma_K \circ \tau \circ \pi_o \circ \gamma_o$ and $\rho_e K = \sigma_K \circ \tau \circ \pi_e \circ \gamma_e$. Linear transformation $\phi_e = \tau \circ \pi_e \circ \tau$ is used after the last round.

3.2. Description of mCrypton. mCrypton is a 12-round and 64-bit block cipher with three key size options (64 bits, 96 bits, and 128 bits). Since mCrypton is based on Crypton, the main concepts of description are very similar to ones of Crypton. The round function of mCrypton also consists of four steps as follows.

Nonlinear Substitution γ. It consists of nibblewise substitutions on a 4 × 4 array using four 4-bit S-boxes, S_i, $(0 \le i \le 3)$.

Bit Permutation π. It mixes each column 4 × 4 array A using column permutation π_i for each column i $(0 \le i \le 3)$:

$$\pi(A) = \left(\pi_0\left(A_c[0]\right)\pi_1\left(A_c[1]\right)\pi_2\left(A_c[2]\right)\pi_3\left(A_c[3]\right)\right), \tag{11}$$

where $A_c[i]$ are the ith column of A.

Each π_i is defined by

$$b = \pi_i(a) \Longleftrightarrow b_j - \bigoplus_{k=0}^{3} \left(m_{i+j+k \bmod 4} \cdot a_k\right), \tag{12}$$

where a column $a = (a_0, a_1, a_2, a_3)^t$ and a column $b = (b_0, b_1, b_2, b_3)^t$.

Byte Transposition τ. It moves the nibble at the (i, j)th position to the (j, i)th position; that is, $B = \tau(A) \Leftrightarrow b_{ji} = a_{ij}$. So $\tau^{-1} = \tau$.

Key Addition σ. $B = \sigma_K(A)$ is defined by $B_r[i] = A_r[i] \oplus K[i]$ $(0 \le i \le 3)$, where $K = (K[0], K[1], K[2], K[3])$ is a round key.

Like Crypton, mCrypton also can be described as

$$E_K = \phi \circ \rho_{K_{12}} \circ \rho_{K_{11}} \circ \cdots \circ \rho_{K_2} \circ \rho_{K_1} \circ \sigma_{K_0}, \tag{13}$$

where $\phi = \tau \circ \pi \circ \tau$.

In this paper, we focus on the 128-bit key version of the mCrypton that is composed of 12 rounds.

Table 3: Indices of expanded keys $E_e[i]$ of Crypton-256 associated with each round.

Round	i
0	0, 1, 2, 3
1	4, 5, 6, 7
2	1, 2, 3, 0
3	7, 4, 5, 6
4	2, 3, 0, 1
5	6, 7, 4, 5
6	3, 0, 1, 2
7	5, 6, 7, 4
8	0, 1, 2, 3
9	$4^*, 5^*, 6^*, 7^*$
10	$1^*, 2^*, 3^*, 0^*$
11	7, 4, 5, 6
12	2, 3, 0, 1

Note: the $*$ represents key space.

4. Biclique Cryptanalysis of Crypton-256

In this section, we describe a biclique attack with dimension 8 ($d = 8$) on the full 12-round Crypton-256. We recover secret key by constructing biclique using independent related-key differentials.

4.1. Key Partitioning and Constructing Biclique for 4 Rounds. We describe how to partition key groups of Crypton-256 in this section. Key schedule of Crypton-256 expands master key, and then all of the round keys are uniquely determined by expanded keys. Therefore, if an expanded key $E_e[i]$ is recovered, the mater key $K[i]$ $(0 \le i \le 7)$ is derived. Indices of 32-bit expanded keys $E_e[i]$ used for generating round keys in each round are listed in Table 3.

The base keys $K_{\langle 0,0\rangle}$ are all 2^{240} 32-byte values with two bytes fixed to 0 ($K_e[38]$ and $K_e[42]$, which is derived from $E_e[6]$ and $E_e[3]$, resp.), but the remaining 30 bytes changes over all values:

$$K_e[38] = (E_e[6]^{\lll b^4})^{\lll 24} \oplus 0x7784368e$$

$$K_e[42] = (E_e[3]^{\lll b^2})^{\lll 8} \oplus 0xb317c51c$$

			0
	0		

We Find second byte of $E_e[0]$ and $E_e[3]$ and fourth byte of $E_e[6]$ and $E_e[7]$ give construction of biclique. Therefore the set of keys $\{K_{\langle i,j\rangle}\}$ which is considering combined (Δ_i, ∇_j)-differentials with respect to the base key $K_{\langle 0,0\rangle}$, is determined by all possible i and j in the following positions:

			j
			j
	i		
	i		

Now, we explain how to construct biclique for 4 rounds of Crypton-256 with dimension 8 ($d = 8$). Let f be the subcipher from Round 9 to final round of Crypton-256. Let the key $K_{\langle 0,0\rangle}$ maps an intermediate state S_0 to a ciphertext C_0, $C_0 = f_{K_{\langle 0,0\rangle}}(S_0)$. Consider previously explained two related-key differentials.

Δ_i-*Differentials*. The Δ_i-differentials are derived from the difference Δ_i^K where the difference of the expanded key is i in the following positions:

Δ_i^K(Round 9 and 10) $=$

	i		
	i		

∇_j-*Differentials*. The ∇_j-differentials are derived from the difference ∇_j^K where the difference of the expanded key is j in the following positions:

∇_j^K(round 9 and 10) $=$

			j
			j

Both Δ_i-differentials and ∇_j-differentials are depicted in Figure 2. Since those two differentials do not share active S-boxes, one can easily obtain the following differentials with respect to the $(P_0, S_0, K_{\langle 0,0\rangle})$:

$$\forall i, j, \quad S_0 \oplus \nabla_j \xrightarrow[f]{K_{\langle 0,0\rangle}\oplus\Delta_i^K\oplus\nabla_j^K} P_0 \oplus \Delta_i. \qquad (14)$$

Hence we can confirm a construction of biclique with dimension 8.

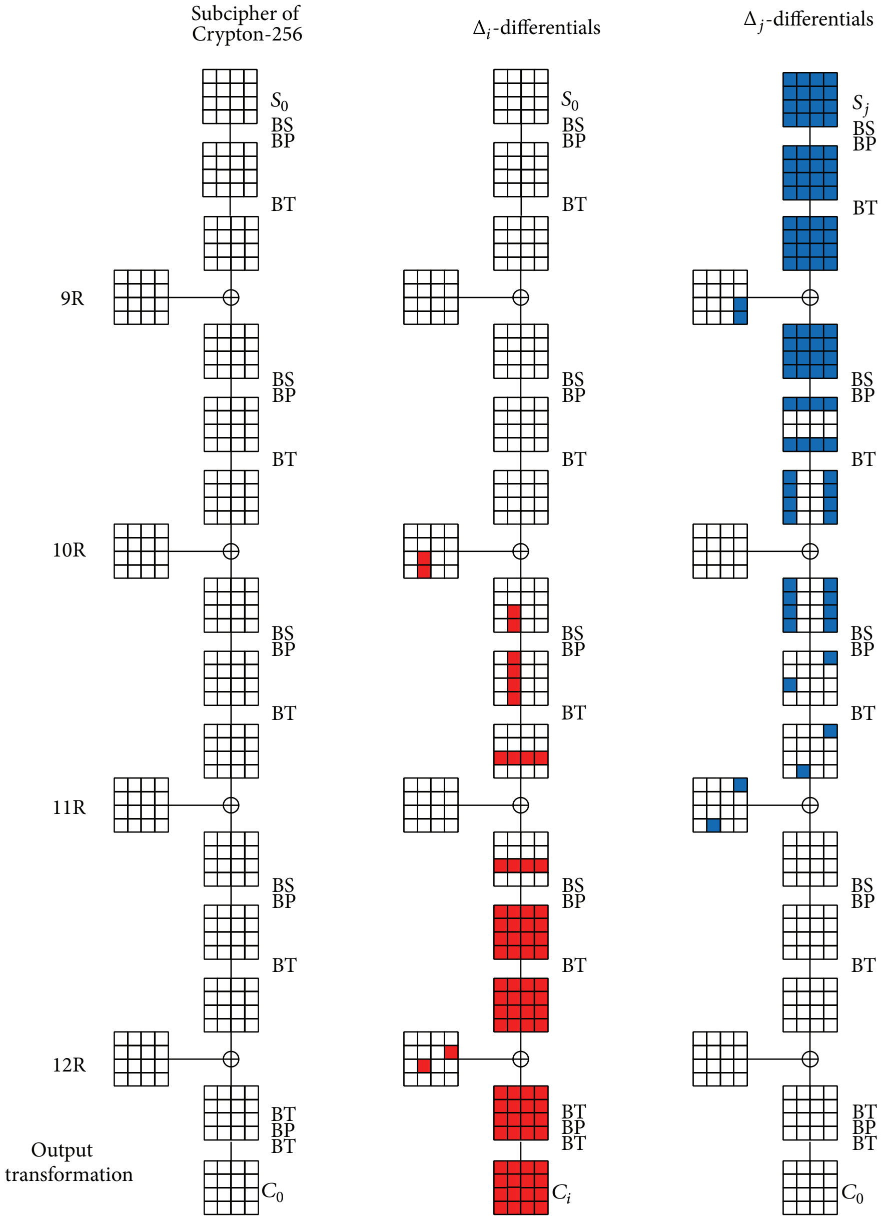


FIGURE 2: 4-round biclique of Crypton-256.

4.2. Key Recovery for the Crypton-256. We describe the key recovery procedure using constructed 4-round biclique for the full Crypton-256. For further explanation, let g be a composition of g_1 and g_2, $g = g_2 \circ g_1$. Then Crypton-256, E, is the composition of the subciphers as follows:

$$E : P \xrightarrow[g_1]{} V \xrightarrow[g_2]{} S \xrightarrow[f]{} C, \tag{15}$$

where g_1 is the subcipher from Round 0 to 4, and g_2 is the subcipher from Round 5 to 8 of Crypton-256. Assume that the plaintext P_i corresponding to each ciphertext C_i in a constructed 4-round biclique is obtained by a decryption oracle.

The adversary finds a candidate key in the following *key testing* step by computing the only 1 byte of intermediate variable v:

$$P_i \xrightarrow[g_1]{K_{\langle i,j\rangle}} \overrightarrow{v} \overset{?}{=} \overleftarrow{v} \xleftarrow[g_2]{K_{\langle i,j\rangle}} S_j. \tag{16}$$

One can perform key recovery procedure by the following steps, precomputation and recomputations.

Precomputation. This step is a preparation phase for an efficient meet-in-the-middle attack. As in Section 2.3, one computes and stores (7) with 2^d encryptions and 2^d decryptions. In Crypton-256, we consider an intermediate matching variable byte v in the output of Round 4 as the byte in the following position:

			v

In precomputation step, first we consider forward direction, from an initial round to Round 4. For all $i = 0, \ldots, 2^8 - 1$, the adversary computes v of the output in Round 4 from P_i with $K_{\langle i,0\rangle}$. And one stores it as $\overrightarrow{v}$ with the intermediate states and subkeys in memory. On the other hand, in backward direction, let us consider subcipher of Crypton-256 from Round 5 to 8. For all $j = 0, \ldots, 2^8 - 1$, one computes v from S_j with $K_{\langle 0,j\rangle}$ and stores it as $\overleftarrow{v}$ with the intermediate states and subkeys in memory. And then we check (16) for every i, j by recomputing those variables which differ from the bytes stored in memory, considering forward and backward directions.

Backward Recomputation. In this step, we explain how to recompute difference between $\overleftarrow{v} \overset{K_{\langle i,j\rangle}}{\longleftarrow} S_j$ and stored one, $\overleftarrow{v}_j \overset{K_{\langle 0,j\rangle}}{\longleftarrow} S_j$. This difference is influenced by the key difference between $K_{\langle i,j\rangle}$ and $K_{\langle 0,j\rangle}$. By key schedule of Crypton-256, the difference in the subkey of Round 8 is two bytes of 16 bytes. The bytes to be recomputed, which include 29 S-boxes, are illustrated in Figure 3.

Forward Recomputation. Recomputing difference, between $P_i \xrightarrow{K_{\langle i,j\rangle}} \overrightarrow{v}$ and stored one, $P_i \xrightarrow{K_{\langle i,0\rangle}} \overrightarrow{v}_i$, is influenced by the key difference between $K_{\langle i,j\rangle}$ and $K_{\langle i,0\rangle}$. By the key schedule, the difference in the subkey of Round 8 is two bytes of 16 bytes. The bytes to be recomputed, which include 10 S-boxes, are depicted in Figure 4.

By these recomputations of two directions, the adversary would make sure whether corresponding key $K_{\langle i,j\rangle}$ satisfies (16). If it satisfies (16), the adversary should check matching the whole bytes at output of Round 4 (input of Round 5) for $K_{\langle i,j\rangle}$, P_i, and S_j. If the adversary cannot find the right key, then one should choose another key group and repeat the above procedures.

4.3. Complexities. Let C_{biclique} be the complexity of constructing a biclique. In our cryptanalysis, it is at most $2^{d+1}(= 2^9)$ 8-round computations, where $n = 256$ and $d = 8$. Let C_{precomp} be the complexity of the precomputation for the matching in (16). And C_{recomp} is the complexity of the recomputation of the byte v. Approximately 2.438 byte substitution operations (39 S-boxes) are required in recomputation. C_{falsepos} is the complexity caused by false positives, which have to be matched on other byte positions. Since the matching in (16) is performed on a single byte, C_{falsepos} is less than $2^{2d-8}(= 2^8)$ computations. Therefore, the total complexity of the biclique cryptanalysis on the full Cryption-256 is as follows:

Table 4: Each round keys of mCrypton-128 from Round 9 to 12.

Round	Round keys
9	$(U[6] << 11) \oplus S(U[5] << 11)$ $(U[7] << 11) \oplus S(U[5] << 11)$ $(U[0] << 11) \oplus S(U[5] << 11)$ $(U[1] << 11) \oplus S(U[5] << 11)$
10	$(U[3] << 11) \oplus S(U[2] << 11)$ $(U[4] << 19) \oplus S(U[2] << 11)$ $(U[5] << 14) \oplus S(U[2] << 11)$ $(U[6] << 11) \oplus S(U[2] << 11)$
11	$(U[0] << 14) \oplus S(U[7] << 11)$ $(U[1] << 19) \oplus S(U[7] << 11)$ $(U[2] << 14) \oplus S(U[7] << 11)$ $(U[3] << 11) \oplus S(U[7] << 11)$
12	$(U[5] << 14) \oplus S(U[4] << 19)$ $(U[6] << 19) \oplus S(U[4] << 19)$ $(U[7] << 14) \oplus S(U[4] << 19)$ $(U[0] << 14) \oplus S(U[4] << 19)$

$$C_{\text{Total}} = 2^{(n-2d)} \left[C_{\text{biclique}} + C_{\text{precomp}} + C_{\text{recomp}} + C_{\text{falsepos}}\right], \tag{17}$$

where C_{biclique}: $2^{(8+1)} \times (4/12) \leq 2^8$, C_{precomp}: $2^8 \times (8/12) \leq 2^8$, C_{recomp}: $2^{(2\cdot 8)} \times (2.438/12) \leq 2^{13.701}$, and C_{falsepos}: $2^{(2\cdot 8-8)} = 2^8$.

Consequentially, the total complexity is

$$C_{\text{Total}} : 2^{240} \times \left(2^8 + 2^8 + 2^{13.7} + 2^8\right) = 2^{253.78}. \tag{18}$$

Although the Δ_i-differential affects all bytes of the ciphertext, only two bytes have 8-bit difference and the remaining bytes have only 6-bit difference. So, 28-bit ciphertext has no difference. As a result, the data complexity does not exceed 2^{100}.

5. Biclique Cryptanalysis of mCrypton-128

In this section, we describe a biclique cryptanalysis with dimension 8 ($d = 8$) on the full mCrypton-128. We recover secret key by constructing a 4-round biclique using independent related-key differentials.

5.1. Key Partitioning and Constructing Biclique for 4 Rounds. By the key schedule of mCrypton-128 in Table 4, all of the round keys are uniquely determined by the master key $U[i]$. We find that some bits of $U[3]$, $U[4]$, $U[1]$, and $U[2]$ give construction of a biclique. The base keys $K_{\langle 0,0\rangle}$ are all 2^{112} 32

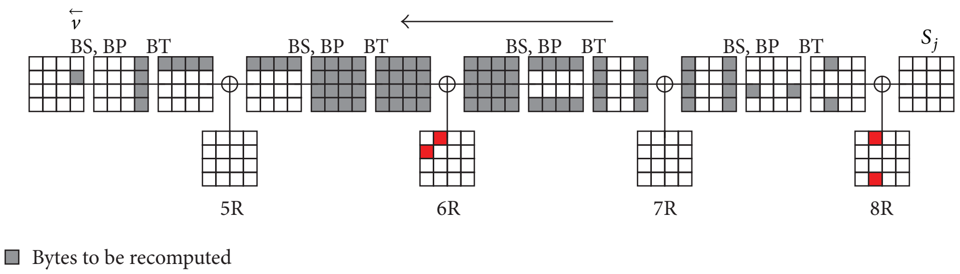


Figure 3: Recomputation in the backward direction of Crypton-256.

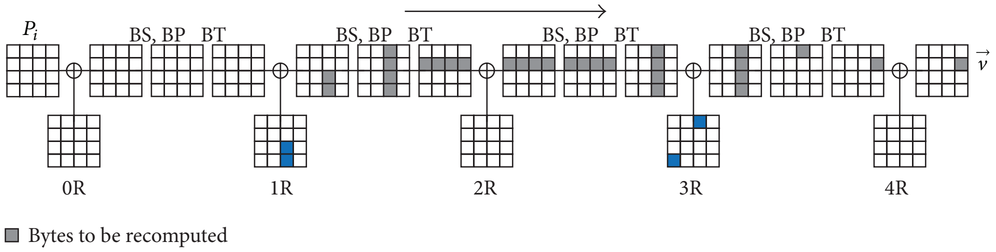
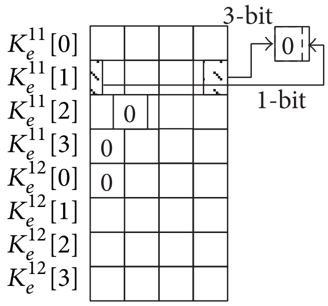
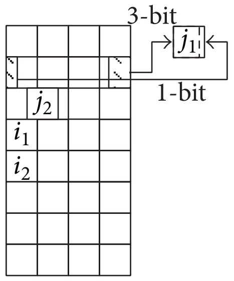


Figure 4: Recomputation in the forward direction of Crypton-256.

nibbles at Round 11 and 12 with 16 bits fixed to 0, in the following positions:



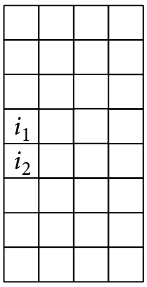
And the set of keys $\{K_{\langle i,j\rangle}\}$, which is considering combined (Δ_i, ∇_j)-differentials with respect to the base key $K_{\langle 0,0\rangle}$, is determined by all possible $i = i_1 \| i_2$ and $j = j_1 \| j_2$ in the following positions:



Now, we explain how to construct a biclique for 4 rounds of mCrypton-128. Consider the following two related-key differentials. Let f be the subcipher from Round 9 to final round of mCrypton-128. Let the key $K_{\langle 0,0\rangle}$ maps an intermediate state S_0 to a ciphertext C_0, $C_0 = f_{K_{\langle 0,0\rangle}}(S_0)$. Consider the two related-key differentials.

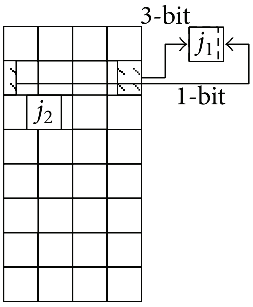
Δ_i-Differentials. The Δ_i-differentials are derived from the following difference Δ_i^K:

Δ_i^K(Round 11 and 12) =



∇_j-Differentials. The ∇_j-differentials are derived from the following difference ∇_j^K:

Δ_j^K(Round 11 and 12) =



Δ_i-differentials and ∇_j-differentials are depicted in Figure 5. We construct a 4-round biclique with dimension 8.

5.2. Key Recovery for the mCrypton-128. Let us explain the key recovery procedure using the 4-round biclique for the full round of mCrypton-128. The adversary finds the right key in

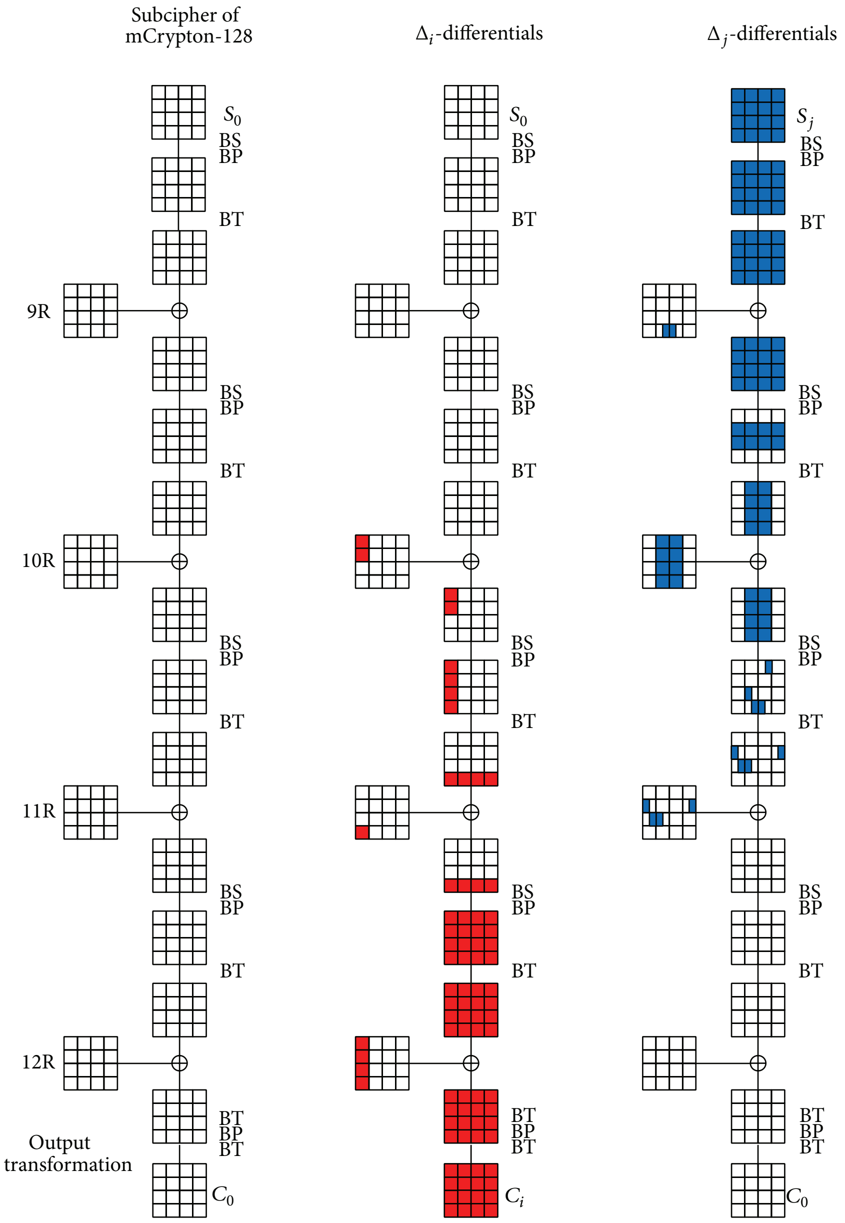


FIGURE 5: 4-round biclique of mCrypton-128.

the following *key testing* step by checking the only 1 nibble of intermediate variable v in (16).

Precomputation. As explained in Section 4.2 for Crypton-256, in mCrypton-128, we consider an intermediate matching variable v in the output of Round 4 as the byte in the following position:

			v

In this step, we first consider forward direction, from initial round to Round 4 of mCrypton-128. For all $i = 0, \ldots, 2^8 - 1$, the adversary computes v of the output of Round 4, from P_i and $K_{\langle i,0\rangle}$. And one stores it as $\overrightarrow{v}$ with the intermediate states and subkeys in memory. On the other hand, in backward direction, we consider Rounds from 5 to 8. For all $j = 0, \ldots, 2^8 - 1$, one computes v from S_j and $K_{\langle 0,j\rangle}$ and stores it as $\overleftarrow{v}$ with the intermediate states and subkeys in memory. Then we check (16) for every i, j by recomputing those variables which differ from the variables stored in memory considering forward and backward direction.

Backward Recomputation. In backward direction, we look at how the computation $\overleftarrow{v} \xleftarrow{K_{\langle i,j\rangle}} S_j$ differs from stored one,

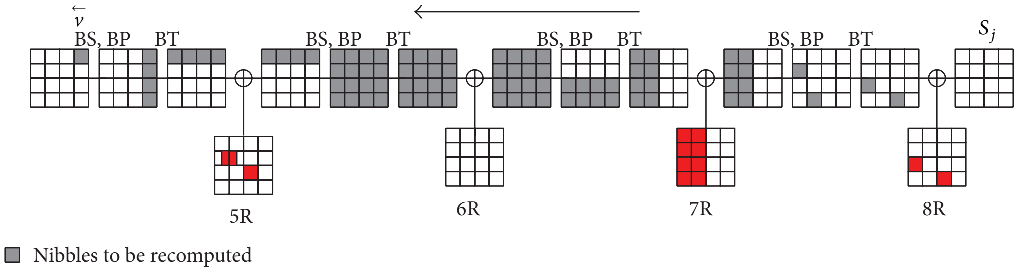


Figure 6: Recomputation in the backward direction of mCrypton-128.

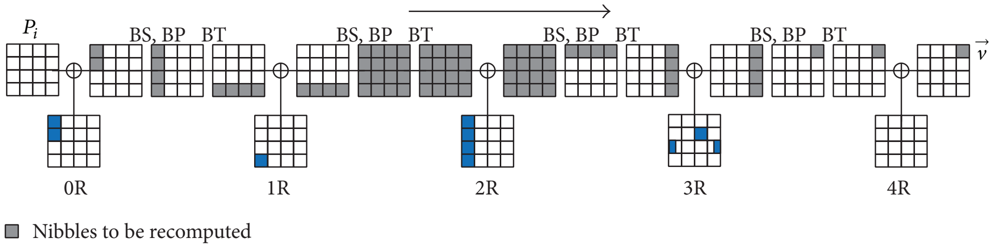


Figure 7: Recomputation in the forward direction of mCrypton-128.

$\overleftarrow{v}_j \xleftarrow{K_{\langle 0,j\rangle}} S_j$. The area to be recomputed, which includes 25 S-boxes, is illustrated in Figure 6.

Forward Recomputation. Let us figure out how the computation $P_i \xrightarrow{K_{\langle i,j\rangle}} \overrightarrow{v}$ differs from stored one, $P_i \xrightarrow{K_{\langle i,0\rangle}} \overrightarrow{v}_i$. The area to be recomputed, which includes 30 S-boxes, is depicted in Figure 7.

By those recomputations of two directions, the adversary would make sure whether corresponding key $K_{\langle i,j\rangle}$ satisfies (16). If it is satisfied (16), the candidate key is right key with high probability. Otherwise, the adversary should choose another key group and repeat the above procedures again.

5.3. Complexities. We construct a biclique for 4 rounds of mCrypton-128 where the dimension is 8. The Δ_i-differentials are based on the difference in 4-bits of $U[3]$ and $U[4]$, and ∇_j-differentials are based on the difference in 4 bits of $U[1]$ and $U[2]$. Approximately 3.4375 nonlinear substitution operations (55 S-boxes) are required in recomputation:

$$C_{\text{biclique}}: 2^{(8+1)} \times (4/12) \leq 2^8,$$
$$C_{\text{precomp}}: 2^8 \times (8/12) \leq 2^8,$$
$$C_{\text{recomp}}: 2^{(2\cdot 8)} \times (3.4375/12) \leq 2^{14.2},$$
$$C_{\text{falsepos}}: 2^{(2\cdot 8-4)} = 2^{12}.$$

Consequentially, the total complexity is

$$C_{\text{Total}}: 2^{112} \times \left(2^8 + 2^8 + 2^{14.2} + 2^{12}\right) = 2^{126.5}. \qquad (19)$$

In ciphertext, four nibbles have 4-bit difference and the remaining 12 nibbles have only 3-bit difference. Also 12 bits of ciphertext have zero difference. Hence the data complexity does not exceed 2^{52}.

6. Conclusions

We use bicliques to recover master key for the full rounds of Crypton-256 and mCrypton-128 with the computation complexity of $2^{253.78}$ and $2^{126.5}$, respectively. This is the first single-key full-round attack for the Crypton-256. And our result on the mCrypton-128 with 4-round bicliques is better than the known biclique cryptanalysis result with 3-round bicliques in terms of computational time complexity.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

[1] C. H. Lim, "CRYPTON: a new 128-bit block cipher," in *NIST AES Proposal*, 1998.

[2] C. H. Lim, "A revised version of CRYPTON: CRYPTON V1. 0," in *Fast Software Encryption*, pp. 31–45, 1999.

[3] C. H. Lim and T. Korkishko, "MCrypton—a lightweight block cipher for security of low-cost RFID tags and sensors," in *Information Security Applications*, pp. 243–258, Springer, 2006.

[4] J. Daemen, L. Knudsen, and V. Rijmen, "The block cipher square," in *Fast Software Encryption*, pp. 149–165, 1997.

[5] Y. Wei, C. Li, and B. Sun, "Related-key impossible differential cryptanalysis on crypton and crypton v1.0," in *Proceedings of the World Congress on Internet Security (WorldCIS '11)*, pp. 227–232, 2011.

[6] J. H. Park, "Security analysis of mCrypton proper to low-cost ubiquitous computing devices and applications," *International Journal of Communication Systems*, vol. 22, no. 8, pp. 959–969, 2009.

[7] H. Mala, M. Dakhilalian, and M. Shakiba, “Cryptanalysis of mCrypton—a lightweight block cipher for security of RFID tags and sensors,” *International Journal of Communication Systems*, vol. 25, no. 4, pp. 415–426, 2012.

[8] A. Bogdanov, D. Khovratovich, and C. Rechberger, “Biclique cryptanalysis of the full AES,” in *Advances in Cryptology—ASIACRYPT 2011*, pp. 344–371, Springer, Heidelberg, Germany, 2011.

[9] C. D’halluin, G. Bijnens, V. Rijmen, and B. Preneel, “Attack on six rounds of crypton,” in *Fast Software Encryption*, pp. 46–59, 1999.

[10] J. H. Cheon, M. Kim, K. Kim, L. Jung-Yeun, and S. Kang, “Improved impossible differential cryptanalysis of rijndael and crypton,” in *Information Security and Cryptology—ICISC 2001*, pp. 39–49, Springer, Berlin, Germany, 2002.

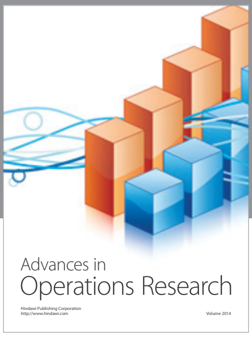
[11] M. Minier and H. Gilbert, “Stochastic cryptanalysis of crypton,” in *Fast Software Encryption*, pp. 121–133, 2001.

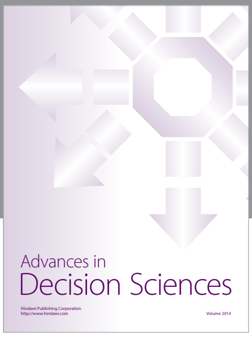
[12] J. Kim, S. Hong, S. Lee, J. H. Song, and H. Yang, “Truncated differential attacks on 8-round CRYPTON,” in *Information Security and Cryptology—ICISC 2003*, pp. 446–456, Springer, Berlin, Germany, 2004.

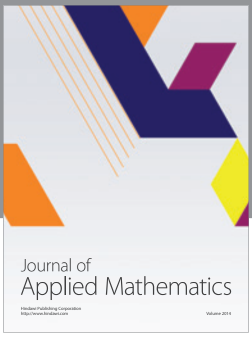
[13] K. Jeong, H. Kang, C. Lee, J. Sung, S. Hong, and J. Lim, “Weakness of lightweight block ciphers mCrypton and LED against biclique cryptanalysis,” in *Peer-to-Peer Networking and Applications*, pp. 1–17, 2013.

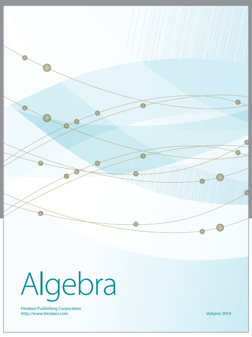
[14] A. Biryukov and D. Khovratovich, “Related-key cryptanalysis of the full AES-192 and AES-256,” in *Advances in Cryptology—ASIACRYPT 2009*, pp. 1–18, Springer, Berlin, Germany, 2009.

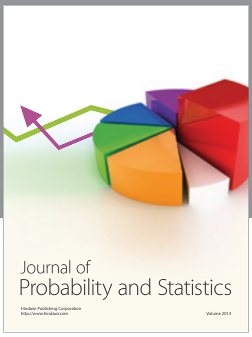
[15] O. Dunkelman, N. Keller, and A. Shamir, “A practical-time related-key attack on the KASUMI cryptosystem used in GSM and 3G telephony,” in *Advances in Cryptology—CRYPTO 2010*, pp. 393–410, Springer, Heidelberg, Germany, 2010.


Advances in
Operations Research

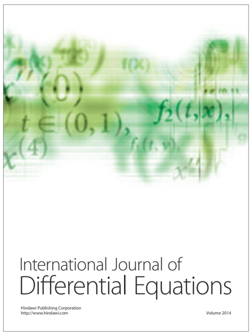

Advances in
Decision Sciences


Journal of
Applied Mathematics

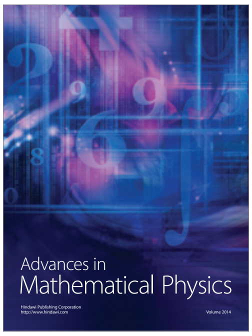

Algebra

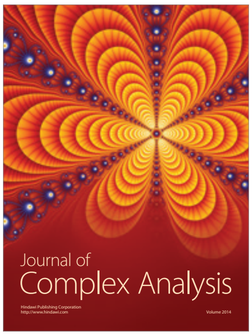

Journal of
Probability and Statistics

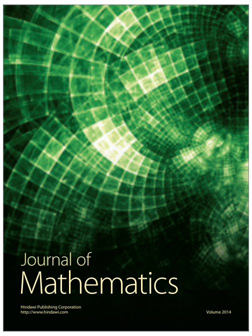
The Scientific
World Journal

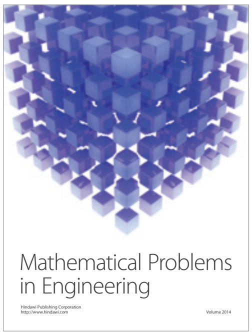

International Journal of
Differential Equations

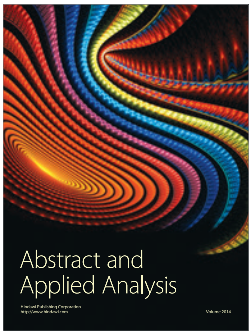
International Journal of
Combinatorics

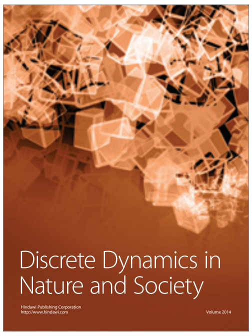

Advances in
Mathematical Physics

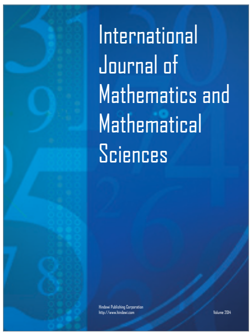

Journal of
Complex Analysis


Journal of
Mathematics


Mathematical Problems
in Engineering

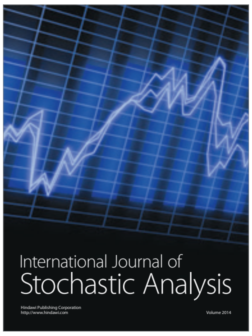

Abstract and
Applied Analysis


Discrete Dynamics in
Nature and Society


International
Journal of
Mathematics and
Mathematical
Sciences

Journal of
Discrete Mathematics

Journal of
Function Spaces


International Journal of
Stochastic Analysis

Journal of
Optimization