Vol.2013-CSEC-61 No.32
Vol.2013-IOT-21 No.32
2013/5/10

Mutable S-box の評価と改良

中林赳明† 齋藤僚† 長瀬智行†

本研究では，ユーザが使用する暗号鍵によって出力値が変化するように，AES の S-box を発展させた，New Mutable S-box を提案した．New Mutable S-box は，AES の S-box よりも差分／線形攻撃法への耐性があることを確認している．また，プログラム上で AES に実装し，暗号における安全性評価の基本となる乱数性の評価を行った結果として，基の AES と同等の乱数性を持つことが確認されている．しかし，弱鍵の存在が発見され，脆弱性となるため，暗号としては不十分であることが判明した．そこで，本研究では，弱鍵の排斥を可能とする構造の Mutable S-box の提案と評価を行い，発見されている脆弱性を改善している．

Improvement and Evaluation of Mutable S-box

TAKEAKI NAKABAYASHI† RYO SAITO† TOMOYUKI NAGASE†

This report examines the security strength of a mutable S-box which is based on modified AES S-box. The structure of the AES S-box has been expanded and modified to improving the complexity of the S-Box's structure and to obtain appropriate non-linearity results. The mutable S-box is evaluated looking for vulnerability to differential cryptanalysis and especially to the linear cryptanalysis. The outcomes of the mutable S-box are also analyzed based on a statistical test for randomness to measure the unpredictability level of the output values.

1. まえがき

近年，クラウドコンピューティングや電子商取引など，インターネットを利用したサービスが急速に普及している．それに伴い，ネットワーク上でのセキュリティが重要視されている．ネットワーク上でのセキュリティが確立していない場合，悪意のある攻撃者がデータの盗聴や，改竄などが容易となってしまうためである．そこで，セキュリティを確立するため，情報を暗号化することが主流である．暗号化とは,悪意のある攻撃者に盗聴されたくないデータを，鍵と暗号アルゴリズムを用いて，もとのデータを予測できない複雑なデータに変換する手法である．

暗号の手法には，公開鍵暗号方式と共通鍵暗号方式の 2 つに大きく分けることができる．公開鍵暗号方式とは，暗号化に用いる鍵と復号化に用いる鍵が異なり，暗号化に用いる鍵を不特定多数者に公開する暗号方式である．共通鍵暗号方式とは，暗号化と復号化に用いる鍵が同じ暗号方式である．そのため，複数の相手と通信をする場合，保持する鍵が多くなる欠点がある．しかし，公開鍵暗号方式と比べ，処理が高速である利点もある．共通鍵暗号方式に代表される暗号アルゴリズムには，DES (Data Encryption) [1]や AES (Advanced Encryption Standard) [2][3]や MISTY [4][5]などがある．

現在，共通鍵暗号方式の解読方法で最も効果的であるものとして，差分解読法 [6]と線形解読法 [7]があり，これらは非線形変換処理部における入出力値の偏りを利用し，鍵を推測する解読法である．

当研究室では,代表的な共通鍵暗号への攻撃方法である，差分／線形攻撃法による攻撃を受けやすい AES の S-box 部を改良した，Mutable S-box (M_S-box) [8]を提案した．更に，M_S-box を改良した New Mutable S-box (N_M_S-box) [9]が提案されている．しかし，N_M_S-box に関しては，安全性における評価が十分ではないため，脆弱性が存在する可能性がある．

本研究では，提案された N_M_S-box へ，今まで試されていない安全性評価を行い，脆弱性が見つかった場合，N_M_S-box の改良を行う．また，改良後の評価，AES への実装後の評価を行う．

2. AES で用いられている S-box

S-box(Substitution box)とは，共通鍵暗号において非線形変換処理として用いられる換字表であり，暗号解読において攻撃対象とされやすい箇所となっている．

AES における S-box の構造を下図に示す．

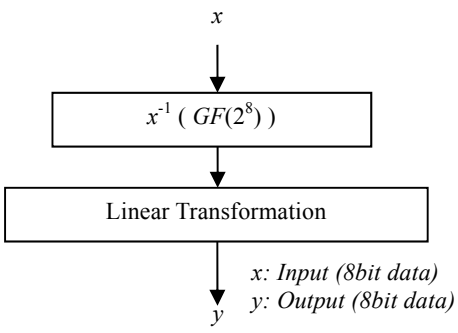


図 1 AES で用いられている S-box の構造

Figure 1 The structure of AES S-box

図 1 で示したとおり，AES で用いられている S-box は，

† 弘前大学
Hirosaki University

$GF(2^8)$上での乗算の逆元演算からなる非線形変換処理とアフィン変換からなる線形変換処理により構成されている．8 次の正方行列と 8×1 行列は，出力データの bit 列がなるべく拡散するようにと選ばれた値となっている．

3. Mutable S-box について

M_S-box は参考文献[8]で提案されており，参考論文[9]では，M_S-box の安全性と処理速度を向上させた N_M_S-box が提案されている．そのため，参考文献[8]の M_S-box を従来 M_S-box と呼ぶことにする．

3.1. 従来 M_S-box のアルゴリズム

従来 M_S-box の基本構造は，図 2 に示すように，AES で用いられている S-box の構成に“Splitting Process”，ガロア体 $GF(2^5)$ 上及び $GF(2^3)$ 上での乗算の逆元演算，“Combining Process”の処理を加えることによって構成されている．

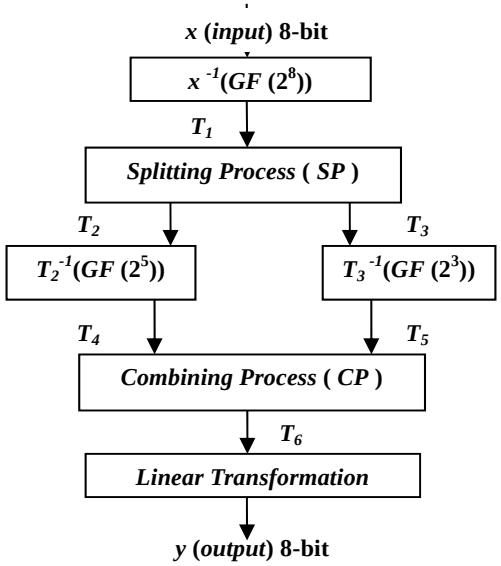


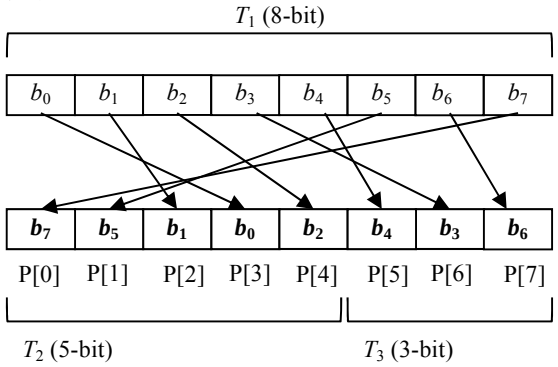
図 2　従来 M_S-box の構成
Figure 2　The structure of the proposed M_S-box

従来 M_S-box では，まず入力データ x (8-bit) に対して，ガロア体 $GF(2^8)$ 上での乗算の逆元を算出し，これを T_1 とする．次に，SP にて，T_2 (5-bit) と T_3 (3-bit)に分割する．分割した T_2 と T_3 に対して，各々にガロア体 $GF(2^5)$ 及び $GF(2^3)$ における乗算の逆元 T_4 (5-bit)と T_5 (3-bit) を求める．その後，CP にて連結された T_6 (8-bit) に対して，線形変換処理としてアフィン変換を行い，従来 M_S-box の出力値 y (8-bit) を算出している．

3.2. Splitting Process

従来 M_S-box の入出力パターンが暗号鍵によって変化するようにしている処理部が SP であり，従来 M_S-box の中心となる処理部である．SP では，暗号鍵より生成されるサブ鍵 (SubKey) から算出される 2 つのパラメータ α と N を用いて並び替え処理と分割処理を行っている．ここで，α を Initial point (並び替え開始位置) ，N を Kernel value とする．

サブ鍵より 2 つのパラメータ α と N を算出する手順は，まず 15-bit データであるサブ鍵を上位から α (3-bit) ，β (6-bit) ，γ (6-bit) に分割する．この時点で α の値は決定される．その後，β のパリティ (各 bit の排他的論理和) を算出し，その結果が 1 であれば β の上位 3-bit，0 であれば β の下位 3-bit を X とする．ここで，X が{0,0,0} である場合，$(R \bmod 6)+2$ の演算結果を X に上書きしている．R はラウンド数であり，2 ラウンド目のラウンド関数が実行されている場合，R は 2 となる．この処理によって X の値が算出される．



Label	**Compute of Bit Number**	Bit Number
P[0]	$N^5 = 2^5 = 7$	b_7
P[1]	$N^6 = 2^6 = 5$	b_5
P[2]	$N^7 = 2^7 = 1$	b_1
P[3]	**Initial Point**	b_0
P[4]	$N^1 = 2^1 = 2$	b_2
P[5]	$N^2 = 2^2 = 4$	b_4
P[6]	$N^3 = 2^3 = 3$	b_3
P[7]	$N^4 = 2^4 = 6$	b_6

図 3　SP での処理例　(α=3，N=2 の場合)
Figure 3 Splitting process SP for (α=3，N=2)

X の値を算出した際の処理と同様の処理を行い，γ から Y を算出する．そして，この X と Y を元にして，ガロア体 $GF(2^3)$ 上での X と Y の乗算を行い，演算結果を N とする．ここで，N が {0,0,1}である場合，$(R \bmod 6)+2$ の演算結果を N に上書きしている．これによって，β (6-bit) 及び γ (6-bit) から N の値が算出され，N の値の範囲が $2 \le N < 8$ となる．ここで，N の値が $2 \le N < 8$ の範囲内に収まるようにしたのは，これから説明する並び替え処

Vol.2013-CSEC-61 No.32
Vol.2013-IOT-21 No.32
2013/5/10

理において並び替え位置を $GF(2^3)$ 上のべき乗演算 N^m ($0 < m < 8$) によって算出しているからである．もし，N の値が {0,0,0} や{0,0,1} も含んでしまうと，べき乗演算 N^m の結果が 0 もしくは 1 で固定されてしまい，並び替え処理が正常に行われないことになる．上記の手法で算出された α と N を元に並び替え処理及び分割処理を行っている．具体的には，並び替え後のデータを格納する配列に P[0]から P[7]までラベル付けを行い，P[α]に b_0 を， P[$(\alpha + m) \bmod 8$] に $b_N{}^m$ を格納することによって，並び替えを行っている．最後に，並び替えられた 8-bit データを上位 5-bit 及び下位 3-bit に分割している．

図 3 に *SP* の処理例 (α=3，N=2 の場合) を示す．*SP* によって分割されたデータに対して，それぞれ GF (2^5) 上及び GF (2^3) 上での乗算の逆元演算を行い，*CP* によってそれぞれ上位 5-bit，下位 3-bit として結合される．*CP* によって結合された 8-bit データに対して，AES で使用されている S-box における線形変換処理を行い，従来 M_S-box の出力としている．

3.3. N_M_S-box のアルゴリズム

N_M_S-box の構造を図 4 に示す．N_M_S-box は，従来 M_S-box を，より安全性と処理速度の向上を目的に提案された．まず，安全性向上のために，ガロア体 GF (2^8) 上での乗算の逆元演算を 3 重構造にし，間に，サブ鍵 16bit の右 8bit と左 8bit の XOR 加算をそれぞれ取り入れることで，入出力パターンを増やしている．

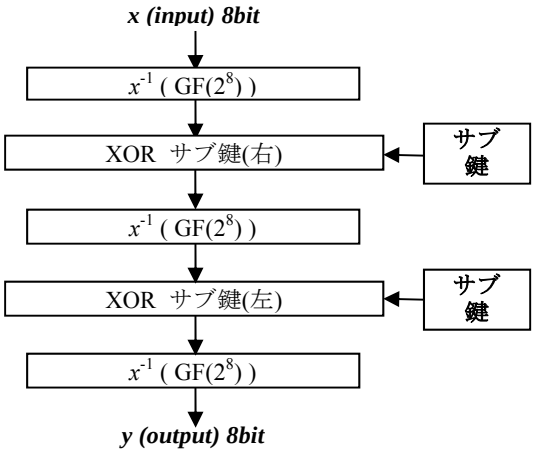


図 4 N_M_S-box の構造

Figure 4 The structure of N_M_S-box

さらに，演算には，換字処理と論理演算のみを用いているため，高速処理が実現できる．ハードウェア実装に関しては，アルゴリズムより，ガロア体逆元演算回路が 1 種類と XOR 回路のみの構成となっているため，問題はないとしている．

4. 従来 M_S-box と N_M_S-box の安全性評価

従来 M_S-box と N_M_S-box の評価を行うにあたり，差分／線形攻撃法の指標[10]を用いた．可変型 S-box における指標としては，MADP(最大平均差分確率)と MALHP(最大平均線系確率)が用いられる．まず，差分確率 (DP) ，線形確率 (LP) の式を以下に示す．

$$DP(\Delta x, \Delta y) = \frac{\#\{x | F(x) \oplus F(x \oplus \Delta x) = \Delta y\}}{2^n}$$

$$LP(\Gamma x, \Gamma y) = \left(2 \frac{\#\{x | x \bullet \Gamma x = F(x) \bullet \Gamma y\}}{2^n} - 1 \right)^2$$

(x: 入力, y: 出力, n: 入力 bit 数,#{x}: x の個数)

サブ鍵ごとの全ての入出力パターンで，DP／LP の平均を取ったものが，ADP (平均差分確率)／ALP (平均線形確率)となり，ADP／ALP 中で最大の確率が MADP／MALHP となる．ADP／ALP 及び MADP／MALHP の式を以下に示す．

$$ALP(\Gamma x, \Gamma y) = \frac{1}{2^t} \sum_k \left(2 \frac{\#\{x | x \bullet \Gamma x = F_k(x) \bullet \Gamma y\}}{2^n} - 1 \right)^2$$

(k:サブ鍵の総数,t:サブ鍵の bit 数)

$$MADP = \max_{\Delta x (\neq 0), \Delta y} ADP(\Delta x, \Delta y)$$

$$MALHP = \max_{\Gamma x, \Gamma y (\neq 0)} ALP(\Gamma x, \Gamma y)$$

N_M_S-box では，各サブ鍵における DP／LP を評価せずに，MADP／MALHP の評価を行っていた．そのため，各サブ鍵における DP／LP を本研究では評価した．評価結果は，サブ鍵が，ある値の場合に DP／LP が共に 100%になることが判明した．これは，AES S-box，従来 M_S-box には存在していない. DP／LP が 100%となるサブ鍵の本数は，65535 本のうち 255 本と少ないが，DP／LP が 100%であると，差分／線形特性が必ず発見されるため，少ない本数であっても，脆弱性に繋がる．

次に，新たな検証として，サブ鍵による入力値と出力値の偏りを検証した．検証結果として，N_M_S-box では，あるサブ鍵を用いた場合，入力値と出力値が同じになることを発見した．サブ鍵が $[FFFE]_{16}$ の場合，入力値 $[37]_{16}$ を入力すると，出力値 $[37]_{16}$ が出力され，$[67]_{16}$ を入力した場合， $[67]_{16}$ を出力し，サブ鍵によっては 0～2 組の入力=出力が存在する．平均すると，サブ鍵 1 本につき，入力=出力が 1 組存在する．また，入力=出力の組が 1～2 となるサブ鍵の値は不規則である．S-box 部で入力=出力となることは，換字処理が行われないことになるため，脆弱性に繋がる可能性が非常に高い．また，この偏りも，AES S-box や，従来 M_S-box では存在していない．

以上のことから，MADP／MALHP の数値や処理速度に関して，従来 M_S-box よりも優れていたが，このままでは脆弱性が存在するため，従来 M_S-box や AES S-box との性能を比較する意味が無いといえる．よって，本研究では，N_M_S-box の発見された脆弱性を改善する構造を提案する．

5. N_M_S-box の改良構造の提案

N_M_S-box の評価により，幾つかの脆弱性が確認できた．従って，ここでは，N_M_S-box を改良したものを提案する．N_M_S-box は，シンプルな構造であるため，計算速度が高速である．そのため，発見された脆弱性が無い構造で，尚且つ，シンプルな構造を目指した．改良を行った構造を図 5 に示す．サブ鍵 8bit を用い，ガロア体上逆元演算を 2 重に減らし，アフィン変換をサブ鍵との XOR の直前に行う構造にしている．DP／LP が 100%となる原因はアフィン変換を行わないことで発生する問題であったことが判明したため，N_M_S-box では行わないアフィン変換を行う構造にした．また，図 3 の構造の場合であっても，入力=出力となるサブ鍵(以下弱鍵と呼ぶ)が存在するが，N_M_S-box では不可能であった弱鍵の排斥を可能にしている構造である．

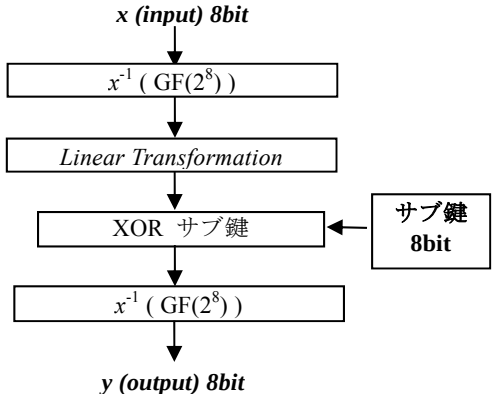


図 5　提案 Mutable S-box の構造
Figure 5 The structure of the proposed S-box

6. 提案構造において発生する弱鍵の偏り

弱鍵とは，脆弱性に繋がるサブ鍵のことである．サブ鍵により，S-box を可変にすることで，弱鍵は存在してしまう．N_M_S-box には，不規則に弱鍵が多数存在した．N_M_S-box をベースに設計する上で，従来通りの不規則に弱鍵が存在するのでは排斥アルゴリズムが困難になるため，規則的に弱鍵が存在するような設計を考慮し，提案した．ここでの規則的というのは，サブ鍵 8bit における各 bit の 1 の個数が偶数の場合，弱鍵となることである．サブ鍵 8bit の 1 の個数の遇数，奇数については表 1 に例を示す．パリティチェックで 1 の個数が偶数と判明したサブ鍵の場合，入力 8bit と出力 8bit が同じになるため，弱鍵と判断出来る．

実際に弱鍵の排斥を行うアルゴリズムは，サブ鍵上位 7bit のパリティを検証し，奇数パリティであれば下位 1bit を 0，偶数パリティであれば下位 1bit を 1 としてサブ鍵を生成する．

ただし，弱鍵を排斥するため，サブ鍵 8bit の鍵の総数 256 本から 128 本へと減少してしまう．

表 1 サブ鍵 8bit のパリティチェック
Table 1 Parity Check of the sub-key of 8-bit

偶数パリティのサブ鍵	奇数パリティのサブ鍵
00000000	00000001
00000011	00000010
…	…
111111111	111111110

7. 提案構造の評価

提案 M_S-box に対する MADP／MALHP の結果を以下に示す．

表 2　MADP／MALHP 評価結果
Table 2 Evaluation results of MADP／MALHP

	MADP	MALHP
AES S-box	2^{-6}	2^{-6}
従来 M_S-box	$2^{-6.42}$	$2^{-6.67}$
提案 M_S-box	$2^{-7.45}$	$2^{-7.36}$

表 2 より，表の数字は各種解読法の成立する確率を示しており，確率が低ければ低いほど安全性が高いとされている．AES の S-box に対して，MADP が約 2.7 倍，MALHP が約 2.6 倍の安全性向上がみられる．従来 M_S-box に比べても，MADP で約 2.0 倍，MALHP で約 1.6 倍の安全性向上となっている．N_M_S-box では，第 4 章で脆弱性が発見されているので，ここでは比較していない．

8. 提案 M_S-box の AES への実装

今回は AES のプログラムに対して，SubBytes 内の処理を提案 M_S-box に変更することで，プログラム上での実装を行った．サブ鍵の決定方法に関しては，ラウンド鍵を用いて図 5 のように行うこととする．

初めに，ラウンド鍵 128bit を用意し，それぞれ 8bit ずつに分割し K_i (i は 0～15) とする．次に，K_i 以外の XOR を計算し出力する．出力された値を Y と置き，Y と K_i の値を XOR することで生成される 8bit をサブ鍵として使用する．AES のサブ鍵 128bit の場合，S-box の入出力は 8bit であり，1 回のラウンドあたり提案 M_S-box を 16 個使用することになる．そのため，図 6 の方法だと，16 個全てにおいて異なるサブ鍵を用いることができる．

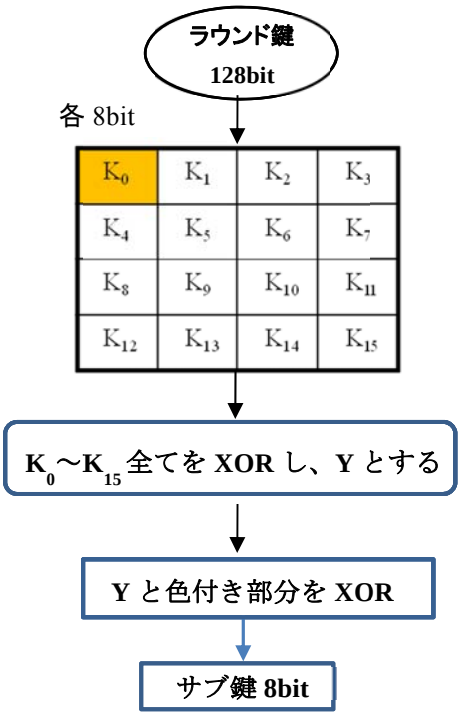


図 6 ラウンド鍵 128bit からのサブ鍵 8bit の生成
Figure 6 Generating of 8-bit sub-key from 128-bit round key

実装後，通常の AES と AES-MS のプログラムを比べると非常に大きな違いが存在する．それは，AES の S-box は換字表を用いて SubBytes 変換ができるため，SubBytes 変換でのプログラムが単純であり，速度の向上がみられる．一方，提案 M_S-box では，サブ鍵を生成し，更に弱鍵の排斥を行うため，大きな遅延が予想される．しかし，脆弱性が改善された換字表の予想が困難である Mutable S-box であるため，安全性の向上が見込まれる．そこで，AES と提案 M_S-box を実装した AES(提案 AES-MS)では，処理速度にどの程度の違いがあるのか，プログラム上で計測した．表 4 は S-box 単体での比較で，表 5 が AES 実装後の比較である．表の結果は，S-box 単体において，10 億回分の平均を取ったものであり，AES 実装後の速度評価では暗号化と復号化をそれぞれ 50 万回分行い，平均を取ったものである．

表 3 S-box 1 回あたりの実行時間 (μs)
Table 3 Processing time of S-box (μs)

AES S-box	**0.04418**
従来 M_S-box	**1.14286**
提案 M_S-box	**0.10841**

表 3 より，処理速度は AES の S-box よりも約 2.5 倍低下している．しかし，従来 M_S-box と比較すると，約 10.5 倍処理速度が向上している．AES S-box と比べ，処理速度が低下した要因として考えられるのは，弱鍵の排斥を行うアルゴリズムを追加し，ガロア体逆元演算とサブ鍵との XOR 演算をしているためである．また，従来 M_S-box と比べ処理速度が向上した要因として考えられるのは，従来 M_S-box では SP 部や CP 部において処理が複雑であるためである．

表 4 提案 M_S-box 実装後の実行時間(ms)
Table 4 Comparison of the processing time of the S-box

AES	**0.07759**
提案 AES_MS	**0.11615**

表 4 より，実装した際の処理速度低下は従来 AES より約 1.5 倍である．しかし，実際に AES に組み込み，使用する際には問題がない処理速度であるといえる．

9. 乱数検定による安全性評価

暗号における乱数性というものは必須条件であり，乱数性がないものはその偏りから簡単に解読されてしまう危険性がある．AES に提案 M_S-box を実装したことで，暗号化後の暗号文に対する乱数性に何らかの偏りが生じたのならば，根本から提案 M_S-box の処理を見直さなければならないことになる．そのため，提案 M_S-box に対する安全性評価として乱数性の安全を証明する必要がある．

今回乱数検定に用いたのが，NIST Special Publication 800-22[11]である．この乱数検定は，表 5 のような 15 種類の検定法で構成されている．

表 5 NIST に含まれる検定名
Table 5 Tests based on NIST

	検定名
1	一次元度数検定
2	ブロック単位の頻度検定
3	累積和検定
4	連の検定
5	ブロック単位の最長列検定
6	2 値行列ランク検定
7	離散フーリエ変換検定
8	重なりあいのないテンプレート
9	重なりあいのあるテンプレート
10	Maurer のユニバーサル統計検定
11	近似エントロピー検定
12	ランダム偏差検定
13	種々のランダム偏差検定
14	系列検定
15	線形複雑度検定

乱数検定では，0 と 1 の乱数列を対象に，各検定法によって，*p*-value という値が算出される．この値を用いた判定基準[11]を以下に示す．

(1) *p*-value が 0.01 以上になる割合（*P*）
(2) *p*-value の一様性（*U*）

(1) は，乱数列の個数を *m* としたとき，*p*-value が 0.01 以上

になる割合が（式 1）の範囲にある場合，良い擬似乱数生成器であると判定される．

$$0.99 \pm 3\sqrt{\frac{0.99 \times 0.01}{m}} \quad (1)$$

(2) は，区間[0,1)を 10 分割し，各区間に属する *p*-value の個数が均等であるかどうかを x^2 分布によって検定する．具体的には，1≦ i ≦10 について，F_i を区間[(i-1)/10,i/10)に属する *p*-value の個数とする時，（式 2）を計算し，*p-value*=igamc(9/2,x^2/2)を計算する．*p-value*≧0.0001 のとき，良い擬似乱数生成器であると判定される．

$$\chi^2 = \sum_{i=1}^{10} \frac{(F_i - m/10)^2}{m/10} \quad (2)$$

これらの判定は 100 万 bit を 1000 本で行い，良い乱数生成器と判定された割合をもとに，合否を決めている．

10. 乱数性の測定結果

AES と従来 M_S-box については乱数性が確認されているので，今回は AES-MS について乱数検定を行う． AES には鍵長 128bit, 192bit, 256bit の 3 パターンがあるが, 128bit において乱数性検定を行った．

表 6 提案 M_S-box に対する乱数検定結果表
Table 6 Tests of the random number from M_S-box

	U	P
1	○	○
2	○	○
3	○	○
4	○	○
5	○	○
6	○	○
7	○	○
8	○	○
9	○	○
10	○	○
11	○	○
12	○	○
13	○	○
14	○	○
15	○	○

U: p-value の一様性 P: p-value が 0.01 以上になる割合
○: 合格　　×: 不合格

入力値 128-bit は全て 0 のデータ，鍵 128-bit はランダムに生成し，100 万 bit を 1000 本の計 10 億 bit を生成した．結果を表 6 に示す．

表 6 より，15 種類の検定全てに合格していることがわかる．つまり，提案 M_S-box を AES に実装しても，乱数性に関しては問題ないことが言える．

11. まとめと今後の課題

本研究では，当研究室で最後に提案された N_M_S-box に対する，安全性の評価を行い，脆弱性の存在を指摘した．さらに，その脆弱性を取り除いた，M_S-box の提案を行った．提案 M_S-box では，脆弱が取り除かれた構造でありながら，AES の S-box，従来 M_S-box を上回る MADP／MALHP の数値であるため，安全性向上を図ることが出来た．また，従来 M_S-box を上回る処理速度であった．以上のことから，今回の研究は，発展性があり，大変有意義なものであった．

今後の課題としては以下のことが挙げられる．

・ハードウェア実装に関する検証
・他攻撃方法への耐性検証

ハードウェア実装では，テーブルの事前計算や，サブ鍵を求める際に必要となる値 Y の格納，弱鍵排斥アルゴリズム等により，使用メモリや回路規模の増大が考えられる．ハードウェア実装の際に，どの程度問題となるのかを検証し，対策を行う必要がある．

他攻撃方法への耐性としては，鍵関連攻撃法，補間攻撃法や SQUARE 攻撃法, 高階差分／線形攻撃法などが挙げられるが，これらに対しての耐性の検証は未だ行なっておらず，検証する必要がある．

参考文献

1) “DATA ENCRYPTION STANDARD (DES),” FIPS PUB 46-3（1999）.
2) “Announcing the ADVANCED ENCRYPTION STANDARD (AES),” FIPS PUB 197（2001）.
3) Joan Daemen, Vincent Rijmen, “Rijndael,” NISSC（2000）.
4) 松井 充, “ブロック暗号アルゴリズム MISTY,” ISEC96(167) 35-48（1996）
5) 三菱電機株式会社, “暗号技術仕様書 MISTY1,” （2001）.
6) 太田 和夫, 青木 和麻呂, ”暗号の攻撃・解読法：差分攻撃法,” 情報処理, 37(6), 521-525 （1996）.
7) 松井 充, ”暗号の攻撃・解読法：線形解読法,” 情報処理, 37(6), 516-520(1996).
8) 金子 敏信, 東京理科大学理工学部電気工学科 金子研究室, “共通鍵ブロック暗号のための強度評価ライブラリ,”（2004）.
9) NIST, “SP800-22 rev1a”(2010).
10) 宇根 正志, 太田 和夫, “共通鍵暗号を取り巻く現状と課題-DES から AES へ-,” 日本銀行金融研究所（1999）.
11) 山口 晃由, 橋山 智訓, 大熊 繁, “非線形変換テーブルの更新を行う暗号法の提案とその安全性評価,” 電子情報通信学会論文誌, Vo;86-A No.8 pp.860-871,（2003）.
12) 金子 敏信, 東京理科大学理工学部電気電子情報工学科, “共通鍵ブロック暗号の線形攻撃耐性評価報告書,” CRYPTREC, No.2001（2010）.