# 128ビットブロック暗号 CLEFIA

# 自己評価書

Version 1.0

ソニー株式会社

平成 22 年 1 月 29 日

## 変更履歴

Jan 29, 2010　version 1.0

# 目 次

# 第1章　はじめに

本書では，128 ビットブロック暗号 CLEFIA の自己評価結果について記述する．

暗号技術は日々進歩しており，新しい攻撃法や設計法，実装法が盛んに研究されている．また近年，暗号機能は，以前よりさらに広い分野の機器やアプリケーションに搭載されるようになってきており，高い安全性と高い性能を兼ね備えた暗号技術を低コストで実装するニーズはますます高まってきている．

我々は最新の攻撃技術や設計技術に基づいて，高い安全性を保ちつつ，ハードウェア，ソフトウェア問わず高い性能を併せもつことを目標とし，CLEFIA の設計を行った．

**CLEFIA の設計思想**

**安全性**　ブロック暗号に対しては非常に多くの暗号攻撃法が知られている．特に，差分攻撃法 [10] や線形攻撃法 [43] に代表される汎用的な攻撃法については，その安全性を定量的に示すことが，その暗号に対する信頼性を得る上で必須と考えられる．CLEFIA では，複数の拡散行列を用いて差分攻撃法および線型攻撃法への耐性を高める新しい設計技法「拡散行列切り替え法」[63–65] を採用し，これらの攻撃法に対する定量的な安全性評価を示すことを目指した．さらに，これ以外の既知の攻撃法についても網羅的に取り上げ，それぞれについて配慮した設計を行っている．

また，共通鍵ブロック暗号に対する攻撃法は日々進化しており，CLEFIA の設計にあたっては，電子政府推奨暗号リスト (以下,「現リスト」という) に記載されているブロック暗号の設計時点以降のさまざまな攻撃法の進歩も考慮した [9, 14, 15, 18]．特に，近年，関連鍵攻撃の進展がめざましく，AES などのシンプルな鍵スケジュールをもつブロック暗号への適用が進んでいる．CLEFIA では，鍵スケジュール部に対しても安全性を評価できる設計とし，関連鍵攻撃の適用を困難とする設計を目指した．

**実装性能**　暗号技術は多くのアプリケーションにさまざまな実装環境で搭載されるようになってきており，幅広いプラットフォームで実装できることが望まれる．この点で米国政府標準暗号 AES は非常に優れた特性を

もっており，CLEFIA は AES より優位性をもつことを目標として設計を行った．AES 以降，さまざまな 128 ビットブロック暗号が設計されているが，ソフトウェア，ハードウェアともに高い性能をもち，AES を上回る特長をもつ方式は数少ない．我々は最新の攻撃技術や設計技術に基づいて，高い安全性を保ちつつ，コンパクト性と高速性を両立させることを目標とし，CLEFIA の設計を行った．CLEFIA はソフトウェアで AES と同等の性能を出すことが可能となっており，特にハードウェアにおいては，AES を超える顕著な実装性能を達成している．

**既存暗号に対する優位性**　CLEFIA は 2007 年に国際会議 Fast Software Encryption (FSE 2007) で採択され [66]，アルゴリズムが発表されて以来，多くの解析結果が発表されているが [73, 78, 79, 83, 86]，現在までにフルラウンドの CLEFIA に対する安全性上の懸念点は指摘されていない．一方，192 ビット鍵および 256 ビット鍵の AES については，関連鍵攻撃という特殊なシナリオ下であるものの，期待される安全性を満たしていないことが明らかになっている [14, 15]．CLEFIA はこのような攻撃に対する耐性を高めた設計となっており，アルゴリズムに対する信頼性を得る上で優位性があると考えられる．

CLEFIA のソフトウェア実装性能は，2.4GHz AMD Athlon 64 プロセッサで 12.9 cycles/byte，1.48 Gbps を達成している．これは現リストに記載されたブロック暗号技術の最も高速なグループに属する．

ハードウェア実装性能は，0.09$\mu$m CMOS 標準セルライブラリを使用した場合に ハードウェア規模 5Kgate 以下で実装することが可能であり，これは現リストに記載されたブロック暗号技術の最も小型実装が可能なグループに属する．また，より高速性を追求した実装では，約 6Kgate で 1.6Gps，約 12Kgate で 3Gbps を超える高速性を達成することが可能で，単位ゲート数あたりの処理速度で見ると，現リストに記載されたブロック暗号技術を超える高い優位性を示している．

**電子政府での利用について**　CRYPTREC では，電子政府推奨暗号リスト素案を作成するにあたって，電子政府システムにおける利用において十分に安全と判断されたレベルの暗号技術を選択するために安全性評価を行っており，この安全性評価において，共通鍵暗号技術における評価基準として，以下の条件のいずれかを満たすことを求めている [27].

- 現時点の最良のアルゴリズム解読技術を適用しても，秘密鍵の総当たりである $2^{128}$ 以上の計算量が必要と判断されるもの． 特に，差分攻撃や線形攻撃などの代表的な攻撃法について，安全性が確認されているもの.

- 世界的に広く使用され，かつ多くの研究者から十分な評価研究を受けているが，実運用における安全性上の大きな問題点が指摘されておらず，現時点で安全と判断されるもの． この場合，$2^{100}$ 以上の解読計算量を目安とした．

CLEFIA の安全性は，3 章に示すように，上記の第 1 項目を満たしていることから，電子政府で使用するものとして妥当であると考えられる．実装性能に関しては，4 章に示すように，ソフトウェア実装，ハードウェア実装いずれにおいても高速に実装可能であることから，高い実装性能が求められる電子政府システムでの利用に適していると考えられる．また，ハードウェア実装における小型実装性能にも優れており，高い実装制約のある環境でのアプリケーションやシステムにも適していると考えられる．

# 第2章　設計理論

この章ではブロック暗号 CLEFIA の設計理論について説明する．

CLEFIA は (1) 安全性，(2) 速度，(3) 実装コストの 3 つの基準をバランスよく実現することを考慮して設計されている．この目的を達成するために，以下に示すいくつかの設計技術が CLEFIA の設計に貢献している．

1. 差分攻撃や線形攻撃 [10, 43] への耐性を高めるための技術「拡散行列切り替え法 (DSM)」を適用した初めてのブロック暗号
2. コンパクトな F 関数を実現するため 4 系列の一般化 Feistel 構造を採用
3. あるクラスの攻撃への耐性を高めるため 2 種類の S-box を採用
4. ソフトウェア実装及びハードウェア実装の際, 軽量な部品のみで実装が可能な構成
5. データ処理部と鍵スケジューリング部を共有して実装することが可能な構成
6. 関連鍵攻撃に高い耐性を持つ新しい鍵スケジューリングアルゴリズム

以下の節では，ここに挙げた特徴について詳細に説明していく．

## 2.1　データ処理部

本節では CLEFIA のデータ処理部の設計方針について説明する．

### 2.1.1　基本構造

CLEFIA は，Feistel 構造を拡張した一般化 Feistel 構造を採用している．従来の Feistel 構造ではデータ系列が 2 系列であるのに対し，一般化 Feistel 構造は 3 系列以上のデータ系列を持つ．一般化 Feistel 構造はデータ系列における F 関数の入出力の接続位置によって様々な種類があるが，CLEFIA では Zheng らによって定義された 4 系列の Type-2 一般化 Feistel 構造を

採用している [88]．図 2.1 に 4 系列の Type-2 構造を示す．ブロック長が 128 ビットであることから，各系列のデータサイズは 32 ビットとなる．4 系列の Type-2 構造は 1 ラウンドに 2 つの F 関数をもち，一方の F 関数は 1 列目のデータ系列，もう一方の F 関数は 3 列目のデータ系列に適用される．

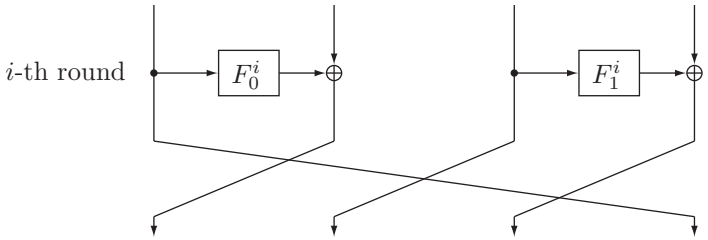


図 2.1: 4 系列 Type-2 一般化 Feistel 構造 (1 ラウンド分)

Type-2 構造は以下の特徴を持つ．

- 2 つの F 関数を同時に処理することが可能
- F 関数のサイズが通常の Feistel 構造に比べて小さい
- 通常の Feistel 構造に比べて多くのラウンドが必要

最初の特徴は特にハードウェア実装の性能を高めることに貢献し，2 つ目の特徴はソフトウェア実装とハードウェア実装の双方に大きな利点をもたらす．3 つ目の特徴は，小さい F 関数により拡散速度が遅くなることによる 4 系列構造の短所である．しかし拡散行列切り替え法 (DSM) と呼ばれる新しい技術を導入することによって，この問題を回避している．この結果，CLEFIA は主に最初の 2 つの特徴のメリットを受けることができる．

一般化 Feistel 構造に関する先駆的な研究は，Zheng, Matsumoto, Imai [88] によって着手され，さらに Nyberg [52] によって行われた．ブロック暗号 RC6 も Type-2 構造をベースとした構造を採用しており，このアルゴリズムの性能向上に寄与している [59]．安全性については，Moriai と Vaudenay が一般化 Feistel 構造の疑似ランダム特性について評価している [49]．さらに Knudsen と Wagner により integral 攻撃 [36] について，Kim らにより不能差分攻撃 [34] についての議論が行われている．

### 2.1.2 F 関数

CLEFIA の F 関数はいわゆる SP 型の F 関数であり，ラウンド鍵の加算後，Substitution 層と Permutation(または Diffusion) 層の順に演算され

る [71]．このタイプの F 関数は，Camellia [3] や Twofish [62] をはじめとした多くのブロック暗号設計に利用されている．CLEFIA は Substitution 層に 4 つの 8 ビットの S-box，Permutation 層に $4 \times 4$ の拡散行列を用いている．この F 関数はテーブル参照の手法を用いてソフトウェアで効率的に実装することが可能である [22]．

### 2.1.3 鍵ホワイトニング

CLEFIA はデータ処理部の最初と最後に鍵ホワイトニングを行っている [71]．各ホワイトニング処理は 128 ビットデータの半分，つまり 4 系列のデータ系列のうち，2 つのデータ系列上しか適用していない．これは 2 系列でもデータ処理部へ十分な鍵情報 (エントロピー) が提供されているためである．このことを説明するために，一般化 Feistel 構造におけるラウンド鍵の等価変形を考える．図 2.2 は F 関数外の鍵加算層を明示的に記述した 2 つの一般化 Feistel 構造を上下に並べて示している．この 2 つの構造は等価である．この図から全てのデータ系列へのホワイトニングは必ず半分のデータ列へのホワイトニングへ変換できる．その逆も同様である．このことから，CLEFIA では鍵加算の計算コストを削減するために半分のデータ列へのホワイトニングを採用している．

### 2.1.4 拡散行列

CLEFIA は，差分攻撃や線形攻撃への耐性を高めるために，異なる 2 つの拡散行列 $M_0$，$M_1$ を採用している．この概念は拡散行列切り替え法 (DSM) と呼ばれ，初めて 2004 年に Shirai と Shibutani によって提案され，さらに Shirai と Preneel によって研究が進められた．しかしこれらの研究は従来の Feistel 構造を対象とするものであった [63–65]．我々はこの技術を Type-2 一般化 Feistel 構造にも適用できるように拡張を行った．DSM の適用は，CLEFIA における固有の技術の 1 つになっている．DSM を用いた場合，近傍ラウンド中の差分消失や線形マスク消失を防ぐことができるため，active S-box として保証される数が増加する．

このメカニズムを説明するために，下記の定義を導入する．

**定義 2.1.** $x \in \{0,1\}^{pl}$ の表現を $x = (x_0 x_1 \dots x_{p-1})$，$x_i \in \{0,1\}^l$ とする．バンドル重み $w_l(x)$ は以下のように定義される．

$$w_l(x) = \#\{i \mid 0 \le i \le p-1, x_i \neq 0\}$$

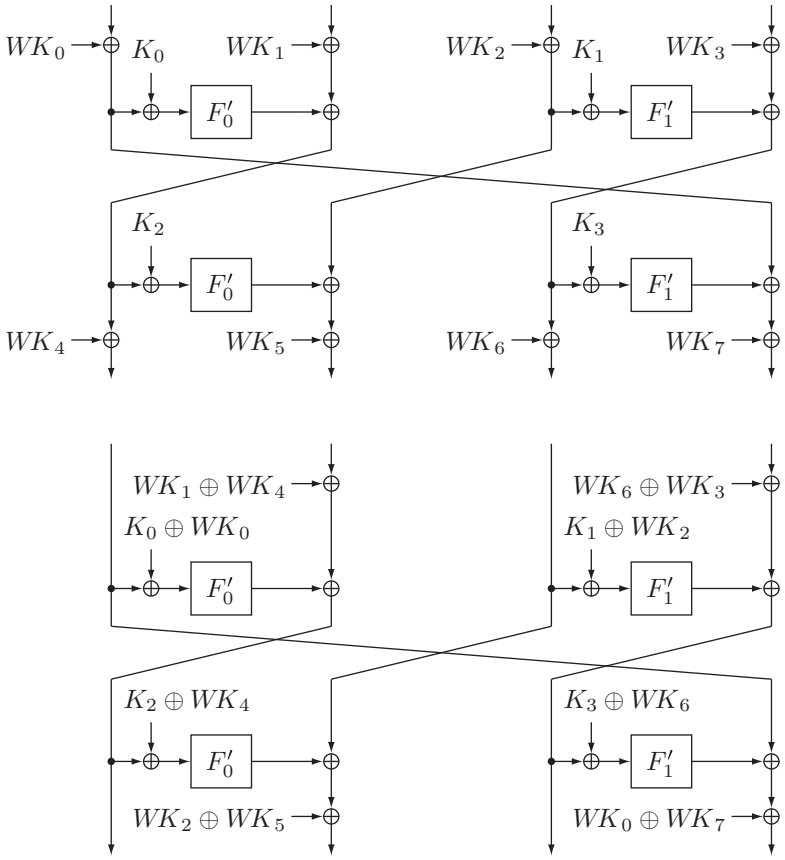


図 2.2: 等価構造

**定義 2.2.** $P : \{0,1\}^{pl} \to \{0,1\}^{ql}$ とする．$P$ の分岐数は以下のように定義される．

$$\mathcal{B}_l(P) = \min_{a \neq 0}\{w_l(a) + w_l(P(a))\}$$

DSM を使うためには，ある分岐数条件を満たす少なくとも 2 つの行列が必要である．CLEFIA では GF($2^8$) の要素を持つ 2 つの $4 \times 4$ 行列 $M_0$, $M_1$ が以下の条件を満たしている．

$$\mathcal{B}_8(M_0) = \mathcal{B}_8(M_1) = 5$$

この分岐数は，このサイズの行列において最適な値である．さらに結合した行列 $M_0|M_1$ と ${}^tM_0^{-1}|{}^tM_1^{-1}$ の分岐数も 5 である．この分岐数も最適な値である．

$$\mathcal{B}_8(M_0|M_1) = \mathcal{B}_8({}^tM_0^{-1}|{}^tM_1^{-1}) = 5$$

図 2.3 に示したように $M_0$ 及び $M_1$ を F 関数内に配置した場合，これらの 2 つの行列の相乗効果により，よい拡散特性を保持することが期待できる．この図では Type–2 Feistel 構造のデータ系列をひねらない形式で表現しているため，F 関数の位置を適切な場所に移動している．この手法を拡散行列切り替え法 (Diffusion Switching Mechanism, DSM) と呼ぶ [63–65]．この DSM の詳細なメカニズムと効果については 2.3 節で解説する．

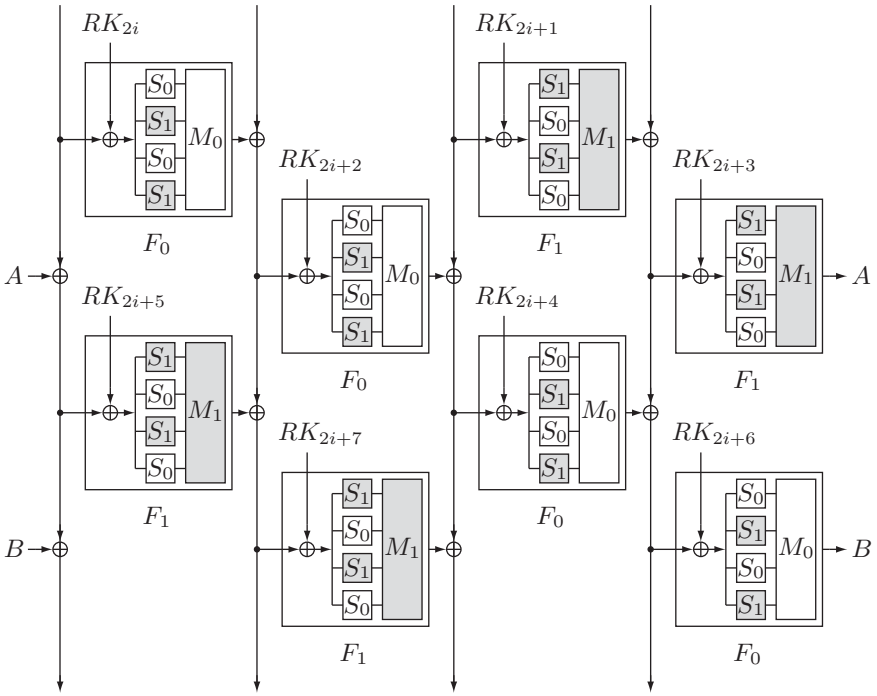


図 2.3: $M_0$, $M_1$, $S_0$ 及び $S_1$ の配置

表 2.1 は DSM の効果を示すため，CLEFIA において保証されている active S-box の数を示している．数値は重みベースの評価手法を用いた計算機シミュレーションによって求めている．

表中の ‘Normal’ は DSM を使わず，全ての F 関数に単一の最適拡散行列を用いた場合の一般化 Feistel 構造に対して保証された active S-box の数を示している．‘DSM(D)’ は最適な分岐数を持った行列 $M_0$, $M_1$ を選び，DSM を適用した場合の保証された差分 active S-box の数を示している．同様に ‘DSM(L)’ は対応ラウンドの保証された線形 active S-box の数を示すものである．この表の結果から $r \geq 3$ で DSM の効果を確認することができ，保証数は ‘Normal’ の結果に比べて 20%–40% 程度増加することがわかる．

表 2.1: active S-box の保証数

| $r$ | Normal | DSM (D) | DSM (L) | $r$ | Normal | DSM (D) | DSM (L) |
|---|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 14 | 25 | 34 | 34 |
| 2 | 1 | 1 | 1 | 15 | 26 | 36 | 36 |
| 3 | 2 | 2 | 5 | 16 | 30 | 38 | 39 |
| 4 | 6 | 6 | 6 | 17 | 32 | 40 | 42 |
| 5 | 8 | 8 | 10 | 18 | 36 | 44 | 46 |
| 6 | 12 | 12 | 15 | 19 | 36 | 46 | 48 |
| 7 | 12 | 14 | 16 | 20 | 37 | 50 | 50 |
| 8 | 13 | 18 | 18 | 21 | 38 | 52 | 52 |
| 9 | 14 | 20 | 20 | 22 | 42 | 55 | 55 |
| 10 | 18 | 22 | 23 | 23 | 44 | 56 | 58 |
| 11 | 20 | 24 | 26 | 24 | 48 | 59 | 62 |
| 12 | 24 | 28 | 30 | 25 | 48 | 62 | 64 |
| 13 | 24 | 30 | 32 | 26 | 49 | 65 | 66 |

但し，DSM を一般化 Feistel 構造に導入することで，2 つのデメリットが生じている．1 つ目は一般化 Feistel 構造の involution 性が部分的にこわれること，2 つ目は 2 つの行列を実装するための追加コストが見込まれることである．しかしここに挙げたデメリットの効率実装への影響は限定的である．involution 性の問題については，暗号化と復号の際のデータのスワップ順序のみを変更すればよい．また行列のサイズはそれほど大きくないので，2 つの行列を使うことによるペナルティも限定的である．

次に CLEFIA と同様に 8 ビットの S-box と Feistel 構造を採用しているブロック暗号 Camellia 及び Twofish と比較することで，DSM の効果について考える．

CLEFIA と Camellia は同じ分岐数 5 を持つ拡散行列を用いた Feistel 構造とみなせる．Camellia は [3] によると，差分 active S-box の数はラウンド 9, 10, 11 に対して 18, 21, 22 である．一方，線形 active S-box の数は $FL/FL^{-1}$ のない Camellia のラウンド 9, 10, 11 に対して 18, 20, 22 である．これらの値は単体の行列を使った CLEFIA より多いが，DSM を使った CLEFIA よりも少ない．つまり，CLEFIA は，2 つの拡散行列を用いた DSM を利用することによって差分攻撃や線形攻撃により高い耐性を持っているといえる．一般化 Feistel 構造は，小さい拡散行列のために拡散性が低いが，DSM はこの欠点を大きな修正コストをかけずに相殺している．

Twofish もまた最大分岐数 5 の 4×4 行列をラウンド関数に持つ．Twofish

の設計者によると，12 ラウンドで 20 個の active S-box を保証している [62]. この結果も CLEFIA よりも少ないことがわかる.

従って，CLEFIA の拡散性能は，よく用いられる active S-box の評価法から Camellia や Twofish の拡散性能に比べて優れていることが期待できる.

#### 2 つの拡散行列の選択

2 つの行列はこれまで述べたように，最適な分岐数条件を満たす必要がある．しかし，この条件を満たす行列は非常に多く存在するため，我々はハードウェア実装のコストの観点から行列を選択した.

候補となる行列は $4 \times 4$ の circulant 行列もしくは Hadamard 行列である．Hadamard 行列はブロック暗号 Anubis で用いられており [4]，$m \times m$ Hadamard 行列の各要素 $h_{i,j}$ は，ある集合 $(a_0, \ldots, a_{m-1})$ に対して $h_{i,j} = a_{i \oplus j}$ として定義される．我々はハミング重みの低い全ての circulant 行列と Hadamard 行列を調べ，ハードウェア実装が効率的にできる，つまり，XOR ゲートの数が少ない，最適な行列を見つけることができた．その結果，CLEFIA で用いる 2 つの行列 $M_0$ 及び $M_1$ は Hadamard 行列と決定した.

### 2.1.5 S-box

CLEFIA では Serpent や Camellia と同様に複数の S-box を採用している [1,3]．CLEFIA の S-box の選択理由は，以下の効果を期待したものである.

1. 既存の攻撃に対する十分な耐性
2. 効率的なハードウェア実装の実現性

まず, 安全性の観点から 2 種類の S-box を採用することを決定してから，実装特性を考慮に入れて具体的な 2 つの S-box を選んだ．我々は 2 種類の S-box を採用することにより，安全性に関して以下の効果を期待している.

- バイト単位での飽和攻撃 [19] への耐性を高める
- XSL 攻撃 [18] をはじめとした代数攻撃への耐性を高める

この理由についてはこの節の後半で述べる．CLEFIA は 2 つの異なるタイプの 8 ビット S-box $S_0$，$S_1$ を採用している．これらの 2 つの S-box は以下のように分類される.

表 2.2: $S_0$ 及び $S_1$ のセキュリティパラメータ

| | $S_0$ | $S_1$ |
|---|---|---|
| 最大差分確率 | $2^{-4.67}$ | $2^{-6.00}$ |
| 最大線形確率 | $2^{-4.38}$ | $2^{-6.00}$ |
| 最小ブール代数次数 | 6 | 7 |
| GF($2^8$) 上の多項式で表現した時の最小項数 | 244 | 252 |

- $S_0$ : ランダムに選択した 4 ビット S-box に基づく 8 ビット S-box
- $S_1$ : GF($2^8$) 上の逆元関数に基づく 8 ビット S-box

2 つの S-box を具体的にどのように選択したか，および安全性への影響については以降の節にて説明する．

#### 4 ビット S-box に基づく S-box

S-box $S_0$ は 4 ビット S-box に基づいて構成されている．この S-box は 4 つの 4 ビット S-box から構成され，全ての 4 ビット S-box は，原始多項式 $x^4 + x + 1$ で定義される GF($2^4$) 上の $2 \times 2$ 行列によって接続されている．この行列の分岐数は 3，つまり，最適な拡散行列となっている．さらに 4 つの 4 ビット S-box は AES のカウンターモードによって生成されたランダムなビット列から選択されている．$S_0$ のいくつかのセキュリティパラメータを表 2.2 に示す．

#### GF($2^8$) 上の逆元関数に基づく S-box

S-box $S_1$ は GF($2^8$) 上の逆元関数に基づいて構成されている．既約多項式として $x^8 + x^4 + x^3 + x^2 + 1$ を用いている．補間攻撃 [28] への耐性を高めるために，逆元演算の前後にアフィン変換を付加している．$S_1$ のいくつかのセキュリティパラメータを表 2.2 に示す．

#### バイト単位での飽和攻撃への耐性を高めた設計

2 つの異なる S-box を用いる 1 つ目の効果として，まず S-box の出力値の衝突を避けることができる点が挙げられる．$X_i \in \{0,1\}^8$ $(0 \leq i \leq 255)$ を 256 個の 8 ビット変数とし，$X_i$ が満たす条件によって，4 つのグループに分類する．分類された $X_i$ $(0 \leq i \leq 255)$ を以下のように呼ぶことにする．

- Const (C)：もし $\forall i, j \;\; X_i = X_j$ を満たす場合
- All (A)： もし $\forall i, j \;\; i \neq j \Leftrightarrow X_i \neq X_j$ を満たす場合
- Balance (B)：もし $\bigoplus_i X_i = 0$ を満たす場合
- Unknown (U)：不明の場合

次に F 関数が 1 つの 8 ビットの鍵加算と 1 つの 8 ビット S-box を用いた substitution 層を含むような単純なモデルを考えてみる (図 2.4 左).

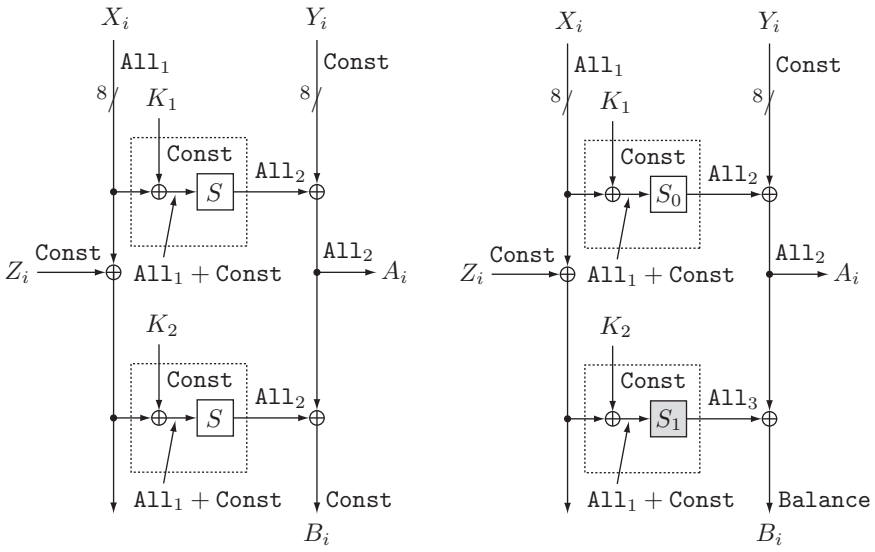


図 2.4: 飽和特性の例

まず $X_i$ が All，$Y_i$ 及び $Z_i$ が Const と仮定する．この仮定は特に CLEFIA のような一般化 Feistel 構造の場合，妥当である．

この時，$B_i$ は下記のように表現できる．

$$B_i = S(X_i \oplus K_1) \oplus S(X_i \oplus Z_i \oplus K_2) \oplus Y_i$$

一般に 2 つの S-box の出力 All は XOR されるので，$B_i$ は Balance であることが期待できる．しかし，ある条件のもとでは $B_i$ は Const になる．定数である $Z_i$ が，$Z_i = K_1 \oplus K_2$ の関係をもつ場合，S-box の出力は常に衝突し，その結果，$B_i = Y_i$ となる．この状況は $p = 1/256$ の確率で起こる．

しかし，CLEFIA では 2 つの S-box が下記の条件

$$\text{For any } c_1, c_2, \; \exists x \;\; S_0(x) \neq S_1(x \oplus c_1) \oplus c_2$$

を満たし，上述した飽和特性のキャンセルを避けることができる．図 2.4 の右図のように $S_0$，$S_1$ とすると，XOR される 2 つの `All` は上記条件によりキャンセルされないため，$B_i$ は S-box の特性により `Const` にならない．

ここでは単純なモデルでの例を示したが，実際の暗号は複雑な行列を用いている．しかし実際の暗号においても同一の S-box を用いているなら，同様の状況がおこりうると考えている．従って我々は，その上述した弱い特性を避けるために，2 つの S-box を採用し，かつ 2 つの F 関数中での S-box の並びを変更している．

### 代数攻撃への耐性を高めた設計

代数攻撃に関するこれまでの結果 [18, 28, 37] から，特定の代数関数のみ，例えばガロア体上の逆元関数のみに頼ることは代数攻撃に対する安全性の観点からいうとよい方法ではない．代数攻撃への耐性を高めるために，ブロック暗号の設計者らはこれまでいくつかのアイデアを提案している．例えば，ランダムな S-box を使う方法 [1,31]，異なるサイズの S-box を組み合わせる方法 [47]，ランダムな 4 ビット S-box から構成する方法 [4,29] などがあり，いくつかは大きな実装コストを必要とする．CLEFIA の設計では，実装コストに大きなペナルティを払うことなく代数攻撃の耐性を高める新しい対策法を適用している．その対策法は，2 つの異なる 8 ビット S-box を用意し組み合わせて使うことである．

CLEFIA で採用している 2 つの 8 ビット S-box は

- $S_0$：ランダムに選択した 4 ビット S-box に基づく 8 ビット S-box
- $S_1$：GF($2^8$) 上の逆元関数に基づく 8 ビット S-box

ランダムに選択した 8 ビット S-box は，CLEFIA にとってハードウェア実装のコストが大きすぎるため，対象としなかった．上述した 2 つの S-box を採用する方が，ハードウェアの実装効率の点から有利である．

逆元関数ベースの S-box は差分及び線形確率の観点から最適であることが知られている一方，GF(2) 上，及び GF($2^8$) 上での単純な代数関係が報告されている．逆元関数ベースの S-box のみを用いると，GF($2^8$) 上の XSL 攻撃への耐性を考慮する必要がある．この場合，GF($2^8$) 上の XSL 攻撃は GF(2) 上の攻撃 [50] に比べて安全性の潜在的脅威になる可能性がある．さらに，Daemen と Rijmen は plateau trails を持つ逆元ベースの S-box に関する新しい結果を示している [21]．これらが，我々が CLEFIA に逆元関数ベースの 8 ビット S-box のみを用いなかった理由である．

一方，4 ビット S-box ベースの 8 ビット S-box は差分及び線形特性に関して最適ではないが，ハードウェア実装において小型に実装できる点に魅

力がある．4 ビット S-box ベースの 8 ビット S-box は GF(2) 上で単純な関係があることが知られており，GF($2^8$) 上の単純な quadratic relation については存在しない．この点をふまえて XSL 攻撃の計算量の評価をすると，このタイプの S-box の GF(2) 上の XSL 攻撃への耐性は逆元関数ベースの S-box に比べて低い [18]．以上が，我々が CLEFIA に逆元関数ベースの 4 ビット S-box ベースの 8 ビット S-box のみを用いなかった理由である．

さらに文献から S-box 選択の傾向を見てみると，ある期間，多くのブロック暗号設計者が逆元関数に基づく 8 ビット S-box を用いていたことがわかる．例えば AES/Rijndael, Camellia, Misty, Hierocrypt-3 等がそうである．その後，4 ビット S-box ベースの 8 ビット S-box が用いられる傾向があり，Whirlpool[1]や Anubis, FOX が挙げられる [3–5, 22, 29, 47, 53]．我々のアプローチは，今述べた S-box 選択の傾向とは異なっている．

CLEFIA では，全ての S-box の半分は逆元関数に基づく 8 ビット S-box，残りの半分は 4 ビット S-box に基づく 8 ビット S-box である．この設計は GF(2) 上，及び GF($2^8$) 上の XSL 攻撃に対して高い耐性をもたせる．さらに，ランダムな 8 ビット S-box のみを選択した場合に比べて実装コストのペナルティがない．

#### $S_0$ と $S_1$ の位置

CLEFIA は DSM を採用しており，2 つの異なる F 関数が存在するため，2 つの S-box を組み合わせる仕組みを導入するのに適している．F 関数 $F_0$ では 4 つの S-box の並びが $S_0, S_1, S_0, S_1$ の順序であるのに対して，F 関数 $F_1$ では $S_1, S_0, S_1, S_0$ の順序である．CLEFIA では 4 ビットベースの S-box と逆元関数ベースの S-box は同じ数だけ存在しており，一般化 Feistel 構造におけるデータ系列の任意のバイトには両方の S-box が交互に適用される (図 2.3 参照)．従って，この構成はバイトごとの飽和攻撃と XSL 攻撃への耐性を持つことが期待される．$F_0$ と $F_1$ 両方に 2 つの $S_0$ と 2 つの $S_1$ が存在することは，実装時にリソースの共有化をする場合に有利である．

## 2.2 鍵スケジューリング部

本節では CLEFIA の鍵スケジューリング部の設計方針について説明する．CLEFIA の鍵スケジュール部の特徴は下記である．

[1]国際標準規格 ISO/IEC 10118-3 に採用されているハッシュ関数である．

1. 中間鍵 $L$ は鍵 $K$ から CLEFIA のデータ処理部の置換関数によって生成される．このことから，関連鍵攻撃に強い耐性があることが期待できる．
2. 等価なラウンド鍵を排除するために，$L$ はラウンド鍵として用いられる
3. $K \oplus L$ は各ラウンドのラウンド鍵として用いられ，一方向性 $K \rightarrow K \oplus L$，つまり $K \oplus L$ から $K$ を求めることが難しいというメリットをもたらす．
4. $L$ を生成する置換関数は比較的重い処理だが，ラウンド鍵と中間値 $L$ の生成コストは軽量である．
5. ここに挙げた特徴は，いずれの鍵長の鍵スケジュール部に対しても共通していえることである．

上記特徴の詳細は，この節にて説明する．

### 2.2.1 128 ビット鍵での $GFN_{4,12}$ の利用

$GFN_{4,12}$ は鍵スケジュール部と鍵ホワイトニングのない 12 ラウンドの CLEFIA である．$GFN_{4,12}$ に対するラウンド鍵は固定の定数である．$GFN_{4,12}$ は 128 ビット鍵 CLEFIA の鍵スケジューリングステップで用いられている．我々は $GFN_{4,12}$ がよい差分伝播特性をもっていると考えている．というのは，$GFN_{4,12}$ の出力差分を制御することは，仮に攻撃者が入力差分を制御できたとしても，非常に難しいからである．$GFN_{4,12}$ が鍵スケジューリング部に適切に用いられるなら，関連鍵攻撃が非常に難しいブロック暗号を構成することができる．

これまでの評価結果から，12 ラウンドの CLEFIA に対して 28 個の差分 active S-box，30 個の線形 active S-box が存在し，$S_0$ による最大の $DP_{max}$ は $2^{-4.67}$，最大の $LP_{max}$ は $2^{-4.38}$ である．このことから，差分特性確率や線形特性確率はそれぞれ $28 \times 4.67 = 130.76$，$30 \times 4.38 = 131.40$ となり，$2^{-128}$ より大きな確率を持たない．この議論は，differential や linear hull についてではなく，差分特性や線形特性についてのみ述べているので，$GFN_{4,12}$ においてよい differential や linear hull がないとは結論できない．しかし，CLEFIA は $DP_{max} = LP_{max} = 2^{-6}$ である S-box $S_1$ も用いているため，実際の特性確率のマージンはここでの見積もりに比べて大きいことが期待できる．特性確率のマージンに関する詳細な議論は Daemen と Rijmen の論文 [20] を参照されたい．

表 2.3: 8 系列の一般化 Feistel 構造の active S-box 最小数

| 段数 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| active | 0 | 1 | 2 | 6 | 8 | 12 | 14 | 21 | 24 | 29 | 34 | 39 |

### 2.2.2 192/256 ビット鍵での $GFN_{8,10}$ の利用

$GFN_{8,10}$ は 10 ラウンドの 8 データ系列の一般化 Feistel 構造であり，各データ系列のビット幅は 32 ビットである．$GFN_{8,10}$ に対するラウンド鍵は鍵長ごとに決められた固定の定数である．$GFN_{8,10}$ の入出力データ長は 256 ビットである．$GFN_{8,10}$ は 192/256 ビット鍵 CLEFIA の鍵スケジューリング部で用いられる．$GFN_{8,10}$ が鍵スケジューリング部に適切に用いられるなら，関連鍵攻撃が非常に難しいブロック暗号を構成することができる．

表 2.3 の評価結果から，$GFN_{8,10}$ に対して少なくとも 29 個の差分 active S-box が存在する．従って $2^{29\times(-4.67)} = 2^{-135.43}$ となることから，$2^{-128}$ より大きい差分特性確率は存在しない．

### 2.2.3 $K$ と $L$ の混合

CLEFIA の 128 ビット鍵スケジュールでは，128 ビットの中間値 $L$ が $K$ から $GFN_{4,12}$ を用いて生成され，ラウンド鍵生成ステップにおいて $K$ と $L$ が混ぜ合わせられる．この方法は $K$ と $L$ の全数探索に対して有効である．$GFN_{4,12}$ の置換により，$K$ は $L$ の全てのビットに依存するため，$L$ の部分情報のみからは $K$ の 1 ビットも推測することが難しい．逆の場合も同じである．従って，$K$ と $L$ が適切にラウンド鍵を生成するために与えられるなら，このような攻撃者に対して暗号強度を高めることができる．

192 ビット及び 256 ビットの鍵スケジュールの場合も同様である．2 つの 128 ビットの中間値 $L_L, L_R$ が $GFN_{8,10}$ によって鍵 $K_L, K_R$ から生成されるため，192 ビット及び 256 ビット鍵の場合も同じ効果が期待できる．

### 2.2.4 *DoubleSwap* 関数

CLEFIA のラウンド鍵生成過程において，中間値 $L$, $L_L$ 及び $L_R$ は *DoubleSwap* 関数を用いて 2 ラウンドごとに更新される．この処理はラウンド鍵間の単純な関係を壊すためである．さらに *DoubleSwap* 関数は巡回シフトと比べて効率のよいハードウェア実装が実現できる．

### 2.2.5 実装方法の柔軟性

CLEFIA は 128 ビット, 192 ビット及び 256 ビット鍵の鍵スケジューリング部とデータ処理部を共有化できるように設計されている．全ての鍵スケジューリングアルゴリズムは CLEFIA のデータ処理部に基づく $GFN_{4,12}$, $GFN_{8,10}$ を用いている．従って，全ての鍵スケジュールアルゴリズムの部品を共有化することによって効率的なハードウェア実装が可能である．

### 2.2.6 定数

各鍵長それぞれの鍵スケジュールアルゴリズムでラウンド定数が用いられる．各定数のサイズは 32 ビットで，各値は 1 つの 16 ビットの初期値から生成される [71]．さらに，これらの定数は最初の 16 ビットの定数を単純なビット演算を繰り返すことによって連続的に生成することができる．従って，ハードウェア実装において，これらの値を動的に生成する実装をすれば，定数を保持するコストを大幅に削減することができる．

## 2.3 拡散行列切り替え法 (DSM)

ここでは，DSM を利用した Type-2 一般化 Feistel 構造の active S-box 数の下界に関する理論的結果および探索アルゴリズムを紹介する [67][2]．

### 2.3.1 対象とする構造

はじめに，$d$ 本のデータ系列（ただし $d \geq 2$ とする）を扱う Type-2 一般化 Feistel 構造について示す．ここでは，$d = 2$ としたときに，通常の Feistel 構造と一致する一般化 Feisetl 構造のクラスを一般化 Feistel 構造と呼ぶものとする．本書では Type-2 と呼ばれる一般化 Feistel 構造を扱う．各種の一般化 Feistel 構造の暗号学的な性質については文献 [34, 49] で研究されている．

$n$ をブロック長，$d$ を $d|n$ を満たす整数とし，$P_0, \ldots, P_{d-1}$ を $n/d$ ビット平文ワード，$C_0, \ldots, C_{d-1}$ を $n/d$ ビット暗号文ワードとする．Type-2 一般化 Feistel 構造は一ラウンド中に複数の F 関数を持ち，系列数 $d$ は偶数である． $F_i^j(x, y)$ により $j$ ラウンド目における左から $i$ 番目の F 関数を表すとする．このとき，Type-2 一般化 Feistel 構造は以下のように定義される．

[2] 2.3 節の著作権は電子情報通信学会に帰属する．©2008 IEICE

*Step 1.* $X_0 \leftarrow P_0, \ldots, X_{d-1} \leftarrow P_{d-1}$
*Step 2.* $i = 1$ から $r$ に対して以下を実行：
  *Step 2.1* $j = 0$ から $d/2 - 1$ に対して以下を実行：
    *Step 2.1.1* $X_{2j+1} \leftarrow X_{2j+1} \oplus F_i^j(RK_i^j, X_{2j})$
  *Step 2.2* $tmp \leftarrow X_{d-1}$,
    $X_j \leftarrow X_{j-1}$ ( for $j = d-1$ to 1),
    $X_0 \leftarrow tmp$
*Step 3.* $C_0 \leftarrow X_0, \ldots, C_{d-1} \leftarrow X_{d-1}$

上記の $RK_i$ $(1 \leq i \leq r)$ は鍵スケジュールから供給されるラウンド鍵であり，ここではその生成法についての定義は行わない．また，一般性を失うことなく，ここでは最終段にデータ系列入れ替え処理を含む構成とする．図 2.5 は $d = 8$ の Type-2 一般化 Feistel 構造のラウンド関数を図示したものである．

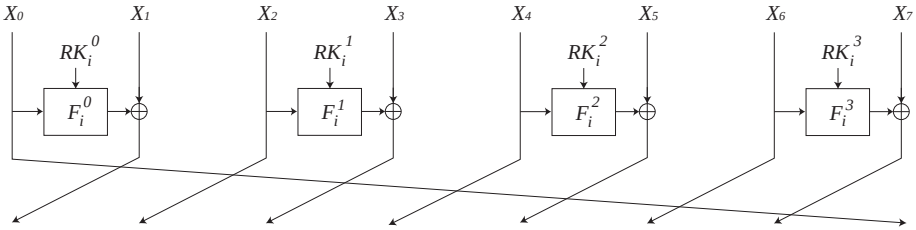


図 2.5: Type-2 一般化 Feistel 構造のラウンド関数 $(d = 8)$

本書では，Feistel 構造で利用する F 関数の形態を，一般的な形の一つである SP 型 F 関数であるものとする [30]．$l$ を S-box の入出力サイズとし，$m$ を拡散正方行列のサイズとすると，$lm$ ビットのラウンド鍵 $RK$ および入力データ $X$ を入力として取り，$Y$ を出力する SP 型 F 関数は次のように表わされる：

*Step 1.* $T \leftarrow RK \oplus X$
*Step 2.* $T = T_0 \mid T_1 \mid \cdots \mid T_{m-1},\ T_i \in \{0,1\}^l$ とする
  $T_i \leftarrow S(T_i)$ $(i = 0$ から $m-1$ まで$)$
*Step 3.* $Y = Y_0 \mid Y_1 \mid \cdots \mid Y_{m-1},\ Y_i \in \{0,1\}^l$ とする
  ${}^t(Y_0, Y_1, \cdots, Y_{m-1}) = M\ {}^t(T_0, T_1, \cdots, T_{m-1})$

上において $A|B$ は 2 つのデータ $A$ と $B$ の結合を表す．$S(\cdot)$ は $l$ の全単射 S-box を表し，$M$ はある体 GF$(2^l)$ 上で定義される正則な $m \times m$ 行列を表す．以降では，$M_i^j$ を用いて $F_i^j$ において使われる拡散行列 $M$ を表現するものとする．図 2.6 は $m = 4$ の場合の SP 型 F 関数 $F_i^j$ を示す．

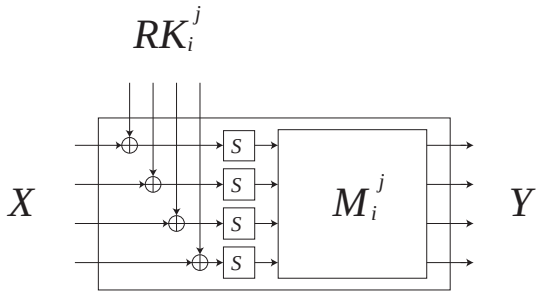


図 2.6: F 関数 $F_j^i$

上記の定義により，ブロック長 $n$ は $d$, $l$, $m$ により $n = dlm$ で表わされる．

### 2.3.2 基本概念

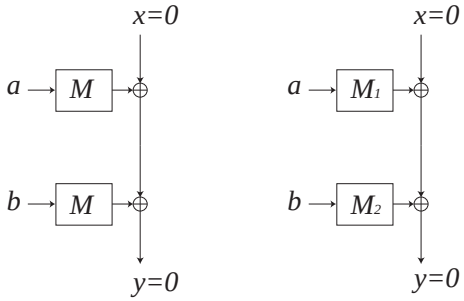


図 2.7: DSM の概念

DSM の基本概念を図 2.7 を利用して説明する．$M$ を正則な $m \times m$ 行列とし，$\boldsymbol{a}, \boldsymbol{b} \in \{\{0,1\}^l\}^m$ を $m$ 次元ベクトルとする．図 2.7 の左側では同じ行列 $M$ から出力された 2 つの値がデータ系列に XOR されている．ここで $x$ と $y$ とがともに 0 であると仮定すると，$w_l(\boldsymbol{a}) + w_l(\boldsymbol{b})$ の最小重みは，$w_l(\boldsymbol{a}) + w_l(\boldsymbol{b}) = 2$ となる．なぜなら任意の $M$ に対して $M(\boldsymbol{a}+\boldsymbol{b}) = \boldsymbol{0}$ を満たすベクトル $\boldsymbol{a} = \boldsymbol{b}, w_l(\boldsymbol{a}) = 1$ が存在してしまうからである．では図 2.7 の右側にあるように，2 つの異なる行列 $M_1$ と $M_2$ が利用されると $w_l(\boldsymbol{a}) + w_l(\boldsymbol{b}) \geq \mathcal{B}_l([M_1|M_2])$ となる．ここで $[A|B]$ は行列 $A$ と $B$ の結合によって得られる $m \times 2m$ の行列とする．定義 2.2 より分岐数 $\mathcal{B}_l([M_1|M_2])$ は最大で $m+1$ であり，その条件を満たす行列は最適拡散と呼ばれる [22,65]．もしも SP 型 F 関数のように行列の直前に S-box を置いた場合には，$w_l(\boldsymbol{a}) + w_l(\boldsymbol{b})$ はこの場合の active S-box 数となって

いる．この観測結果より，上記の分岐数の条件を満たしていれば，右側の行列の選択方針の方が局所的に多くの active S-box 数を保証できる可能性があることがわかる．DSM ではこの性質を Feistel 構造全体に適用し下界の向上を行っている．

### 2.3.3 DSM を利用した Type-2 一般化 Feistel 構造

次に，この DSM 技術を Type-2 一般化 Feistel 構造に適用する．例えば，$d = 6$ とした際の Type-2 構造について，データ入れ替えによるねじれを戻した形で示したものが図 2.8 である．DSM を使うためには，$F_i^j$ と $F_{i+2}^{j-1}$ 中の二つの行列 $M_i^j$ と $M_{i+2}^{j-1}$ とが，すべての可能な $i$ と $j$ の組に対して以下の DSM の分岐数に対する条件を満足しなくてはならない．ここでは，$M$ や $F$ の添え字のうち右上のものは $d/2$ を法とする mod 演算が適用されている．例えば $d/2 = 0$，$-1 = d/2 - 1$ となる．

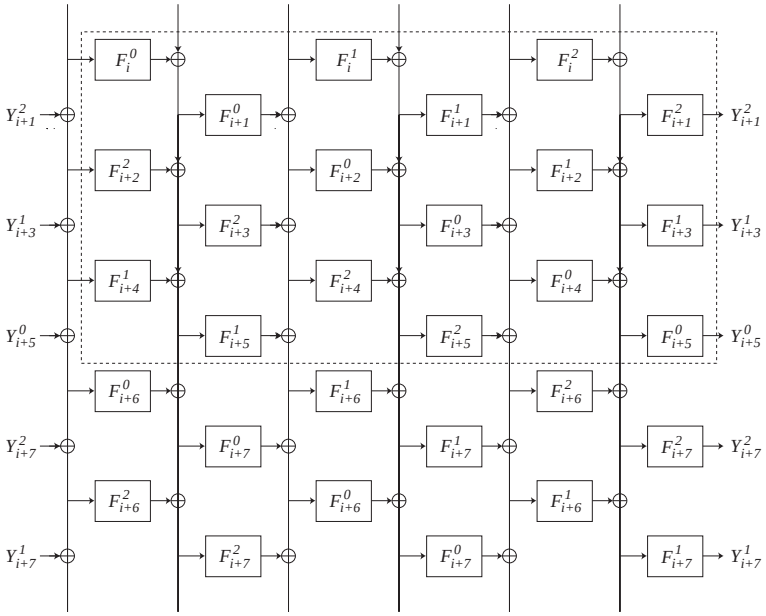

図 2.8: Type-2 一般化 Feistel 構造 ($d = 6$, ねじれなし)

ここで、$B_1^D, B_2^D$ および $B_2^L$ を以下のように定義する．

**定義 2.3.**

$$B_1^D = \min_{1 \le i \le r,\ 0 \le j < d/2} (\mathcal{B}_l(M_i^j)) \ ,$$

$$B_2^D = \min_{1 \le i \le r-2,\ 0 \le j < d/2} (\mathcal{B}_l([M_i^j \mid M_{i+2}^{j-1}]) \ .$$

$$B_2^L = \min_{1\le i\le r-2,\ 0\le j<d/2} (\mathcal{B}_l([{}^t(M_i^j)^{-1} \mid {}^t(M_{i+2}^{j-1})^{-1}])) \ .$$

この定義により $B_1^D \ge B_2^D$ であることがわかる.

これらの定義を使って，Type-2 構造に対する差分および線形 active S-box 数の下限に関する証明を示す.

### Type-2 一般化 Feistel 構造の差分 active S-box

$X_i^j$ と $K_i^j$ を F 関数 $F_i^j$ への入力とし，$D_i^j$ を使って $F_i^j$ の差分 active S-box 数を表す．Type-2 一般化 Feistel 構造に非ゼロ差分データが入力された場合には，以下の性質が利用可能である.

**性質 2.1.** 2 つの連続するラウンド中には，最低でも一つの $F$ 関数において差分 *active S-box* をもつものが存在する.

この性質は構造の全単射性より説明される.

**性質 2.2.** もしも $D_i^j = 0$ ならば，$D_{i-1}^{j+1} = D_{i+1}^j$ を満たす．また，もしも $D_i^j \ne 0$ ならば $D_i^j + D_{i-1}^{j+1} + D_{i+1}^j \ge B_1^D$ を満たす.

この性質は次の等式から導かれる.

$$F_i^j(K_i^j, X_i^j) = X_{i-1}^{j+1} \oplus X_{i+1}^j.$$

**性質 2.3.** もし，$D_i^j \ne 0$ または $D_{i+2}^{j-1} \ne 0$ ならば，$D_i^j + D_{i+2}^{j-1} + D_{i-1}^{j+1} + D_{i+3}^{j-1} \ge B_2^D$ である.

この性質は次の等式から導かれる.

$$F_i^j(K_i^j, X_i^j) \oplus F_{i+2}^{j-1}(K_{i+2}^{j-1}, X_{i+2}^{j-1}) = X_{i-1}^{j+1} \oplus X_{i+3}^{j-1}$$

これらの性質により，以下が得られる.

**定理 2.1.** $d \ge 4$ とする．*SP* 型の $F$ 関数を利用した $d$ 系列の *Type-2* 一般化 *Feistel* 構造では任意の連続する 6 ラウンドにおいて $B_1^D + B_2^D$ 個の差分 *active S-box* が保証される.

証明. ここでは，$a$ ラウンド目から始まる連続する 6 ラウンドに注目する．証明を理解しやすくするため，図 2.9 のように，6 ラウンドに含まれる $3d$ 個の F 関数を整列した箱に並べて示す．箱の幅は $d$ である．同一のラウンド内にある F 関数は，同じ行にある箱に対応し，隣のラウンドにある F 関数は隣の列に格納される．図 2.8 において破線で囲まれた領域は $d = 6$ の場合の $i$ 番目のラウンドから 6 ラウンド分を表したものである.

性質 2.1 により，3 ラウンド目と 4 ラウンド目の F 関数の中には最低一つの非ゼロ差分が入力されている F 関数が存在する．図 2.9 では $(a+2)$ ラウンド目と $(a+3)$ ラウンド目である.

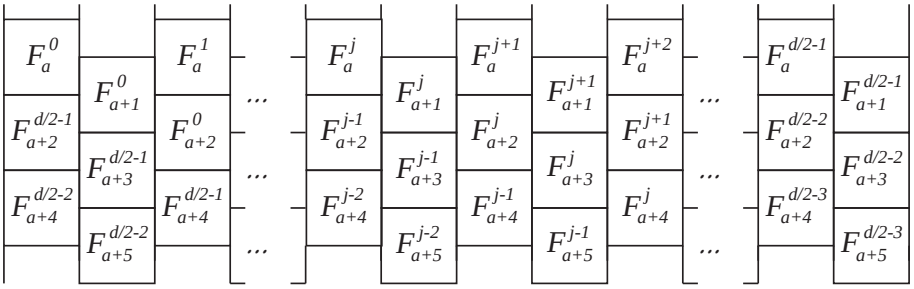


図 2.9: Type-2 一般化 Feistel 構造 (箱形式)

**CASE 1** (第 3 ラウンド目に非ゼロ差分が存在する場合．)
第 3 ラウンド目に存在するある非ゼロ差分が $D^j_{a+2} \neq 0$ であったと仮定する。性質 2.2 と 2.3 により,

$$D^j_{a+2} + D^{j+1}_{a+1} + D^j_{a+3} \geq B^D_1 \ , \tag{2.1}$$

$$D^j_{a+2} + D^{j-1}_{a+4} + D^{j+1}_{a+1} + D^{j-1}_{a+5} \geq B^D_2 \ . \tag{2.2}$$

**1A** もし，$D^{j-1}_{a+3} \neq 0$ であれば性質 2.3 より，$D^j_{a+1} + D^{j-1}_{a+3} + D^{j+1}_a + D^{j-1}_{a+4} \geq B^D_2$ が成り立つ．このことと式 (2.1) を合わせて，$\sum_{i=a}^{a+5} \sum_{j=0}^{d/2-1} D^j_i \geq B^D_1 + B^D_2$ が得られる．

**1B** もし，$D^j_{a+3} \neq 0$ であれば性質 2.2 より，$D^j_{a+3} + D^{j+1}_{a+2} + D^j_{a+4} \geq B^D_1$ が成り立つ．このことと (2.2) を合わせて，$\sum_{i=a}^{a+5} \sum_{j=0}^{d/2-1} D^j_i \geq B^D_1 + B^D_2$ が得られる．

**1C** もし，$D^{j-1}_{a+3} = D^j_{a+3} = 0$ であれば，$D^{j-1}_{a+3} = 0$ の条件と性質 2.2 より，$D^{j-1}_{a+4} = D^j_{a+2} \neq 0$ が成り立つ．性質 2.2 を $D^j_{a+4}$ に使えば,

$$D^{j-1}_{a+4} + D^j_{a+3} + D^{j-1}_{a+5} \geq B^D_1 \tag{2.3}$$

が得られる．式 (2.1) と (2.3) には $D^j_{a+3}$ が重複してあらわれているが, $D^j_{a+3} = 0$ とすでにわかっているため無視することができる． 従って $\sum_{i=a}^{a+5} \sum_{j=0}^{d/2-1} D^j_i \geq 2 \times B^D_1 \geq B^D_1 + B^D_2$ が得られる．

また $D^j_{a+2}$ 以外の場所に非ゼロ差分があった場合にも同様に証明が可能である。

**CASE 2** (第 4 ラウンド目に非ゼロ差分が存在する場合．)
この場合は，構造の対称性を考慮すれば CASE 1 と同様に下界を証明できる． □

### Type-2 一般化 Feistel 構造の線形 active S-box

差分の場合と同様に，$F_i^j$ に対する線形 active S-box の数を $L_i^j$ で表す．Type-2 一般化 Feistel 構造に非ゼロの入力マスク値が与えられた時に，以下の性質が成り立つ：

**性質 2.4.** 2 つの連続するラウンド中には，最低でも一つの $F$ 関数において線形 *active S-box* をもつものが存在する．

この性質は構造の全単射性より説明される．

**性質 2.5.** 任意の *3* つ組，$L_i^j, L_{i+1}^j, L_{i+2}^{j-1}$ に対して，

- $L_i^j = L_{i+1}^j = L_{i+2}^{j-1} = 0$，または
- $L_i^j + L_{i+1}^j + L_{i+2}^{j-1} \geq B_2^L$（ただし，3 つの項のうち 2 つ以上の項は非ゼロ）

上記の性質を使って，以下の定理を示す．

**定理 2.2.** $d \geq 4$ とする．*SP* 型の $F$ 関数を利用した $d$ 系列の *Type-2* 一般化 *Feistel* 構造では任意の連続する 6 ラウンドにおいて $2 \times B_2^L$ 個の線形 *active S-box* が保証される．

証明. 定理 2.1 と同様に，$a$ ラウンド目から始まる連続する 6 ラウンドにおける active S-box の保証数を示す．
性質 2.4 は第 3 ラウンド目および第 4 ラウンド目に最低一つの非ゼロ入力線形マスクを持つ F 関数が存在することを保証する．この場合，$(a+2)$ 番目もしくは $(a+3)$ 番目のラウンドである．

**CASE 1** (第 3 ラウンド目に非ゼロ線形マスクが存在する場合．)
第 3 ラウンド目に存在するある非ゼロ線形マスクが $L_{a+2}^j \neq 0$ であったと仮定する。性質 2.5 より，$L_{a+1}^j + L_{a+2}^j + L_{a+3}^{j-1} \geq B_2^L$ が得られる．ここで，上記式内の 3 つの項のそれぞれが 0 ではないと仮定すると，以下が得られる．

- $L_a^j + \boldsymbol{L_{a+1}^j} + L_{a+2}^{j-1} \geq B_2^L$
- $L_a^{j+1} + L_{a+1}^{j+1} + \boldsymbol{L_{a+2}^j} \geq B_2^L$
- $\boldsymbol{L_{a+3}^{j-1}} + L_{a+4}^{j-1} + L_{a+5}^{j-2} \geq B_2^L$

上記で述べた 3 つの非ゼロの項を太字で強調している．これら 3 つの式においては，重複する項は存在しない．性質 2.5 により一つの式に含まれる 3 つの項のうち最低 2 つ以上の項は非ゼロである，従って上記 3 つの式のうち 2 つは必ず成立している．その結果，$\sum_{i=a}^{a+5}\sum_{j=0}^{d/2-1} L_i^j \geq 2 \times B_2^L$ が得られる．

また $L_{a+2}^j$ 以外の場所に非ゼロ差分があった場合にも同様に証明が可能である。

**CASE 2** (第 4 ラウンド目に非ゼロ線形マスクが存在する場合．)
この場合は，構造の対称性を考慮すれば CASE 1 と同様に下界を証明できる． □

### 2.3.4 計算機による評価

本節では，一般化 Feistel 構造の下界を示すための別のアプローチを示す．既存の探索手法を一般化 Feistel 構造に合うように手法を改善する [64].

### 2.3.5 探索アルゴリズムの基本方針

active S-box 数を数えるための基本的な探索アルゴリズムは [64] において示されている．

1. すべての可能な重みの数値 $D_i^j$ (または $L_i^j$), $(1 \leq i \leq r)$ の組み合わせについて以下を行う．
   - $D_i^j$ (または $L_i^j$) の組に対して矛盾がないかどうかをチェックする．チェックは前節までにのべた構造の持つ性質を利用する．もしも，矛盾が存在するならば，その組候補を棄却する．そうでなければ，$D_i^j$ $(1 \leq i \leq r)$ の和を計算して保存する．
2. 和の最小数をその構造の下界として出力する．

2.3.3 節で紹介した性質を重み値の誤った組み合わせを排除するために用いている．例えば，Type-2 一般化 Feistel 構造の性質を用いると，$B_2^D = 5$ かつ $D_i^j \neq 0$ の場合には性質 2.1 により $D_i^j + D_{i-1}^{j+1} + D_{i+1}^j < 5$ にはなり得ないため，この候補は棄却される．

実際の探索アルゴリズムは，以下のように行う．$\mathcal{ST}_R$ を評価対象の $R$ 段の一般化構造を表すものとし，$NF_i$ をはじめの $i$ ラウンド目までに含まれる F 関数の総数であるとする．そして，$\mathcal{F}_1, \mathcal{F}_2, \cdots, \mathcal{F}_{NF_R}$ を使って，

$\mathcal{F}_{di+j} = F_i^j$ という形で別名を割り当てる．さらに，$\mathcal{D}_i$ と $\mathcal{L}_i$ をそれぞれ F 関数 $\mathcal{F}_i$ に対する差分 active S-box 数および線形 active S-box 数とする．基本の最小 active S-box 数の探索アルゴリズムは表 2.4 のようになる．

表 2.4: 基本探索アルゴリズム

| |
|---|
| INPUT: $R$ (ラウンド数), $\mathcal{ST}_R$ (評価対象構造) |
| OUTPUT: $\mathcal{ST}_R$ の保証された active S-box 数 |
| メイン: |
| Step 1. グローバル変数を次のように設定 $LB = \infty$ |
| Step 2. $Func(1)$ を呼び出す |
| Step 3. $LB$ を出力する |
| |
| $Func(x)$ |
| Step 4. もし $x = NF_R + 1$ ならば以下を実行する: |
| もし $LB > \sum_{p=1}^{NF_R} \mathcal{D}_p$ ならば $LB \leftarrow \sum_{p=1}^{NF_R} \mathcal{D}_p$ とする |
| Step 5. もし $x \neq NF_R + 1$ ならば，$j = 0$ から $m$ に対して以下を実行する: |
| $\mathcal{D}_x = j$ と設定し，その構造に対するすべての性質が |
| 満足されているかどうかをチェック |
| チェック結果が OK であれば，$Func(x + 1)$ を呼び出す |

上記において $Func(x)$ は再帰呼び出し関数である．線形 active S-box 数を探索する際には，$\mathcal{D}_i$ を $\mathcal{L}_i$ と置き換え，適用する性質を線形マスクのものに変更する．

この探索アルゴリズムは小さなパラメータセットに対してのみ現実的な時間内で完了することを確認している．我々の実験によれば，10 ラウンドの $m = 4, d = 4$ とした Type-2 一般化 Feistel 構造でも一日以上の探索時間がかかってしまう．この莫大な計算コストは，より大きなサイズの構造を評価する際には障害となってしまう．

### 2.3.6 改善された探索アルゴリズム

基本の探索アルゴリズムに対して追加の枝切り手法を導入することにより高速化を図る．改善された探索アルゴリズムを表 2.5 に示す．基本アルゴリズムと改良アルゴリズムの主要な違いは，改良アルゴリズムではすでに得られているより少ないラウンドに対する最少 active S-box 数を利用する点にある．

表 2.5: 改良探索アルゴリズム

INPUT: $R$ (ラウンド数), $\mathcal{ST}_R$ (評価対象構造)
OUTPUT: $\mathcal{ST}_R$ の保証された active S-box 数

Main:
Step 1. グローバル変数を次のように設定 $LB_i = \infty$ $(1 \leq i \leq R)$
Step 2. $i = 1$ から $R$ に対して以下を実行する:
$Func(1, i)$ を呼び出す
Step 3. $LB_R$ を出力する

$Func(x, r)$
Step 4. もし $x = NF_r + 1$ ならば，以下を実行する:
もし $LB_r > \sum_{p=1}^{NF_r} \mathcal{D}_p$ ならば $LB_r \leftarrow \sum_{p=1}^{NF_r} \mathcal{D}_p$ とする．
Step 5. もし $x \neq NF_r + 1$ ならば，$j = 0$ から $m$ に対して以下を実行する:
$\mathcal{D}_x = j$ と設定し，その構造に対するすべての性質が
満足されているかどうかをチェック
チェック結果が OK であれば，以下を実行する:
Step 5.1. もし $x \notin \{NF_k | 1 \leq k \leq r-1\}$ ならば,
$Func(x + 1, r)$ を呼び出す
Step 5.2. もし $x \in \{NF_k | 1 \leq k \leq r-1\}$ ならば,
$z$ を $x = NF_z$ を満たす整数とし,
$\sum_{p=1}^{NF_z} \mathcal{D}_p + LB_{r-z} \leq LB_r$ である場合に限り,
$Func(x + 1, r)$ を呼び出す

Step 5.2. において，もしその時点で確定したはじめの $z$ 個の F 関数に含まれる active S-box 数と，残されたラウンドで保証されている下界との和が既に，現時点で得られている全ラウンドに対する下界を超えている場合には探索は中断される．なぜなら，それ以上の探索を行っても下界を更新することはありえないからである．この早期中断による枝切り方法により，探索コストが大幅に低減される．我々の実装結果では 50 段の $m = 4$, $d = 4$ とした Type-2 一般化 Feistel 構造に対する結果を数十秒で得ることができた．この改良により，より大きな構造に対する評価を行うことができるようになった．

# 第3章　安全性評価

この章では CLEFIA の安全性について述べる．CLEFIA の安全性を評価するため，ブロック暗号に対して現在知られているあらゆる攻撃を検討した．CLEFIA に対してそれぞれの攻撃が適用できるかをチェックした後，その攻撃により CLEFIA が何段まで攻撃可能か評価することで，それぞれの攻撃に対する CLEFIA の耐性を詳細に評価している．CLEFIA に対して検討を行った 20 種類の攻撃を下記に示す.

1. 差分攻撃 (Differential Cryptanalysis)
2. 線形攻撃 (Linear Cryptanalysis)
3. 差分線形攻撃 (Differential-Linear Cryptanalysis)
4. Boomerang 攻撃 (Boomerang Attack)
5. 拡張 Boomerang 攻撃 (Amplified Boomerang Attack)
6. Rectangle 攻撃 (Rectangle Attack)
7. Truncated 差分攻撃 (Truncated Differential Cryptanalysis)
8. Truncated 線形攻撃 (Truncated Linear Cryptanalysis)
9. 不能差分攻撃 (Impossible Differential Cryptanalysis)
10. 飽和攻撃 (Saturation Cryptanalysis)
11. 衝突攻撃 (Collision Attack)
12. 高階差分攻撃 (Higher Order Differential Cryptanalysis)
13. 補間攻撃 (Interpolation Cryptanalysis)
14. XSL 攻撃 (XSL Attack)
15. カイ二乗攻撃 ($\chi^2$ Cryptanalysis)
16. スライド攻撃 (Slide Attack)

17. 関連暗号攻撃 (Related-Cipher Cryptanalysis)
18. 関連鍵攻撃 (Related-Key Cryptanalysis)
19. 関連鍵 Boomerang 攻撃 (Related-Key Boomerang Cryptanalysis)
20. 関連鍵 Rectangle 攻撃 (Related-Key Rectangle Cryptanalysis)

これらの攻撃に対する評価結果を以下の節に順に示す．3.1 節では CLEFIA のデータ処理部に関する安全性について，3.2 節では鍵スケジュール部を含む CLEFIA の安全性について述べている．

この結果，現時点では 128 ビット鍵の CLEFIA に対しては (18 段中)12 段まで，192 ビット鍵の CLEFIA に対しては (22 段中)13 段まで，256 ビット鍵の CLEFIA に対しては (26 段中)14 段まで，いずれも不能差分攻撃によって攻撃可能であることが確認されている．しかしながら，いずれの鍵長においても仕様通りの CLEFIA については，現時点の最良の解読技術を適用しても，秘密鍵の全数探索の計算量より少ない計算量で攻撃することはできない．

## 3.1 暗号解析 I — データ処理部

本節では，CLEFIA のデータ処理部の安全性を評価するため，下記の攻撃について検討を行う．

1. 差分攻撃
2. 線形攻撃
3. 差分線形攻撃
4. Boomerang 攻撃
5. 拡張 Boomerang 攻撃
6. Rectangle 攻撃
7. Truncated 差分攻撃
8. Truncated 線形攻撃
9. 不能差分攻撃
10. 飽和攻撃
11. 衝突攻撃
12. 高階差分攻撃
13. 補間攻撃
14. XSL 攻撃
15. カイ二乗攻撃

### 3.1.1 差分攻撃

差分攻撃は Biham と Shamir によって提案された，ブロック暗号に対する汎用的な攻撃である [10,11]．差分攻撃に対するブロック暗号の安全性を評価する方法には，以下の 2 通りがある．

1. ランダム置換 (random permutations) との識別に利用可能な差分 (differential) が存在しないことを示す
2. ランダム置換 (random permutations) との識別に利用可能な差分特性 (differential characteristic) が存在しないことを示す

これまでのところ，多くの暗号に対して 1 つ目の手法により安全性を評価することは困難であることが知られている．SPN 構造の最大差分確率を評価する手法が Hong らによって示されているが [26]，CLEFIA の全体構造は SPN 構造ではないため，AES や FOX のようにこの理論を用いて評価することはできない [22,29].

我々はもう 1 つのアプローチ，すなわち差分特性確率を評価する手法を採用する．これは active S-box の数を計算することで評価できることが知られており，この評価手法は AES や Camellia などでも用いられている [3,22]．最大差分確率と最大差分特性確率の間にどれほどのギャップがあるかはこれまで明らかになっていないが，Daemen と Rijmen によって両者の関係が詳細に議論されており [20]，これによると，ある統計的な仮定のもとで，両者には統計的な関連が存在している．よって，差分特性ベースのアプローチは差分攻撃に対する安全性を評価する合理的な手法の一つであると考えることができる．

**active S-box**

一般に，ブロック暗号に含まれる差分 active S-box の数を保証する方法には 2 種類ある．一つは証明等で示された active S-box 数の下限を用いる方法，もう一つは探索アルゴリズムにより active S-box 数の下限を評価する方法である．我々は両方を確認し，どちらのアプローチがより厳密な下限を得られるかを確かめた．その結果，探索ベースで求められる下限は証明により示される下限よりも厳密であることがわかった．よって，我々は計算機による探索で得られた結果を CLEFIA の安全性評価に用いる．

表 2.1 に，計算機実験により得られた CLEFIA の最小 active S-box 数を示す．ここでは ‘DSM(D)’ の列に着目する．この列の ‘$r$’ 段目に示される数値は $r$ 段の CLEFIA に含まれる差分 active S-box の最小数を示している．

**差分攻撃に対する安全性評価に用いる S-box の差分確率**

差分攻撃に対する安全性を評価するためには S-box の差分確率が必要である．CLEFIA には 2 種類の S-box $S_0$ と $S_1$ があり，それぞれの最大差分確率は $DP_{max}^{S_0} = 2^{-4.67}$, $DP_{max}^{S_1} = 2^{-6.0}$ となっている．設計者の観点から CLEFIA の差分攻撃に対する安全性を評価するには，CLEFIA の全ての S-box が (差分攻撃に対して弱い=高い最大差分確率を有する)$S_0$ であると仮定して評価すべきと考えられる．

#### 差分攻撃

表 2.1 より 12 段 CLEFIA に最小 28 個の差分 active S-box が存在することと $S_0$ の最大差分確率 $DP_{max}^{S_0}$ が $2^{-4.67}$ であることより，12 段 CLEFIA の最大差分特性確率は $DCP_{max}^{\text{12-round}} \leq 2^{28\times(-4.67)} = 2^{-130.76}$ のように示せる．これは，攻撃者が差分攻撃に利用可能な 12 段差分特性が存在しないことを意味する．さらに，実際の最大差分特性確率 $DCP_{max}$ は，次の 2 つの理由によりこの見積もりよりも小さい値であることが期待される．1 つ目の理由は，すべての 28 個の active S-box が同時に差分確率 $2^{-4.67}$ をとるような差分特性経路を構築するのは極めて困難であること，2 つ目の理由は，CLEFIA はより小さな最大差分確率をとる $S_1$ も用いていることである．よって，攻撃者が CLEFIA をランダム置換と識別するのに利用可能な 12 段差分経路を見つけるのは困難と考えられる．これにより，最も効率のよい鍵回復攻撃を考慮に入れても，仕様通りの CLEFIA は差分攻撃に対して十分な耐性を持つと考える．

### 3.1.2 線形攻撃

線形攻撃は松井によって提案された，ブロック暗号に対する汎用的な攻撃である [43]．線形攻撃に対する安全性を評価するには，差分攻撃に対する安全性評価と同様の手法，すなわち，線形 active S-box 数と S-box の最大線形確率を用いて評価することができる．

表 2.1 の ‘DSM(L)’ で示された列の ‘$r$’ 段目に示される数値は $r$ 段の CLEFIA に含まれる線形 active S-box の最小数を示している．CLEFIA の S-box $S_0$ と $S_1$ の最大線形確率は，それぞれ $LP_{max}^{S_0} = 2^{-4.38}$, $LP_{max}^{S_1} = 2^{-6.00}$ である．我々は線形攻撃に対する安全性評価においても，差分攻撃の時と同様，全ての S-box が $S_0$ であると仮定して評価を行う．表 2.1 より 12 段 CLEFIA に最小 30 個の差分 active S-box が存在することと S-box の特性より，12 段 CLEFIA の最大線形特性確率は $LCP_{max}^{\text{12-round}} \leq 2^{30\times-4.38} = 2^{-131.40}$ のように示せる．すべての 30 個の active S-box が同時に最大の線形確率 $2^{-4.38}$ をとるような線形特性経路を構築するのは極めて困難であること，CLEFIA はより小さな最大線形確率 $2^{-6.00}$ をとる $S_1$ も用いていることより，攻撃者が CLEFIA とランダム置換を識別するのに利用可能な 12 段線形経路を見つけるのは困難と考えられる．

### 3.1.3 差分線形攻撃

差分線形攻撃 (Differential-Linear cryptanalysis) は Langford と Hellman によって提案された，ブロック暗号に対する汎用的な攻撃である [41].

この攻撃では差分特性と線形特性の両方を利用する．攻撃に利用する差分特性の確率を $p$，線形特性の確率を $q$ とすると，差分線形攻撃の攻撃計算量は $p^2q^2$ のオーダーとなる．差分・線形特性ベースでの解析では，差分 active S-box を 2 個含む差分特性確率 $2^{2\times(-4.67)} = 2^{-9.34}$ の 3 段差分特性と，線形 active S-box を 10 個含む線形特性確率 $2^{10\times(-4.38)} = 2^{-43.8}$ の 5 段線形特性を組み合わせた 8 段 distinguisher が最も攻撃者に有利な組み合わせとなる．しかしながら，この distinguisher を用いた攻撃計算量は差分攻撃や線形攻撃において最も効率のよい distinguisher を用いた場合の攻撃計算量より多い．よって，仕様通りの CLEFIA は差分線形攻撃に対して十分な耐性を持つと考えられる．

### 3.1.4 Boomerang 攻撃

Boomerang 攻撃は Wagner によって提案された適応的選択平文・暗号文攻撃である [82]．この攻撃では比較的短い差分特性のペアを quartet という特殊な構造において利用する．Boomerang 攻撃のアイデアは，確率の低い，長い差分経路ではなく，確率の高い，短い 2 本の差分経路を組み合わせるというものである．

$n$ をブロックサイズのビット長，$k$ を鍵サイズのビット長とする．CLEFIA の暗号化アルゴリズム $E : \{0,1\}^n \times \{0,1\}^k \to \{0,1\}^n$ を $E_0$ と $E_1$ のカスケード構造 $E = E_1 \circ E_0$ として表し，$E_0$ に確率 $p$ の差分経路 $\alpha \to \beta$，$E_1$ に確率 $q$ の差分経路 $\gamma \to \delta$ が存在するとする．Boomerang 攻撃では $E_0$ において平文ペア $(P_0, P_1)$, $(P_2, P_3)$ に関して差分経路 $(\alpha \to \beta)$ を用い，$E_1$ において暗号文ペア $(C_0, C_2)$, $(C_1, C_3)$ に関して差分経路 $(\gamma \to \delta)$ を用いる．以下のプロセスで攻撃を行う．

- $P_0 \oplus P_1 = \alpha$ を満たす平文ペア $(P_0, P_1)$ に対し，対応する暗号文のペアを求め，$(C_0, C_1)$ とする．
- $C_2 = C_0 \oplus \delta$, $C_3 = C_1 \oplus \delta$ を満たす暗号文ペア $(C_2, C_3)$ に対し，対応する平文のペアを求め，$(P_2, P_3)$ とする．
- $P_2 \oplus P_3 = \alpha$ を満たすかどうかをチェックする．

ランダム置換であれば，$P_2 \oplus P_3 = \alpha$ の条件が成り立つ確率は $2^{-n}$ である．しかしながら，$E$ では，$(P_0, P_1)$ が差分経路 $(\alpha \to \beta)$ に従うペア (right pair) である確率は $p$，$(C_0, C_2)$ および $(C_1, C_3)$ がともに right pair である確率は $q^2$ となる．もしこれらのペアがすべて right pair であれば下記が成り立ち，

$$E_1^{-1}(C_2) \oplus E_1^{-1}(C_3) = \beta = E_0(P_2) \oplus E_0(P_3)$$

確率 $p$ で $P_2 \oplus P_3 = \alpha$ が成り立つ．これらを満たす平文・暗号文を quartet と呼ぶ．よってこの quartet が Boomerang 攻撃に必要な条件を満たす確率は $(pq)^2$ となる．従って，128 ビットブロック暗号 CLEFIA に Boomerang 攻撃が適用できる条件は

$$pq > 2^{-64}$$

となる．

我々は CLEFIA に対して 2 種類の 9 段 boomerang distinguisher (distinguisher I，distinguisher II) を見つけている．表 3.1 に示すように，CLEFIA は distinguisher I，II を用いて 9 段までランダム置換と識別可能である．表 3.1 のケース III は distinguisher I を 1 段伸ばしたものであるが，10 段ではランダム置換と識別できないことが分かる．これらの boomerang distinguisher を用いた鍵回復攻撃は 9 段以上に伸ばせたとしても 2, 3 段が限界と考えられ，仕様通りの CLEFIA は Boomerang 攻撃に対して十分な安全性をもっていると考えられる．

表 3.1: Boomerang Distinguisher

| ケース | $E_1 \circ E_0$ | | $E$ |
|---|---|---|---|
| I | $(E_0, E_1)$<br>$(p, q)$ | (3.5-round, 5.5-round)<br>$(\leq (2^{-4.68})^2, \leq (2^{-4.67})^8)$ | 9-round<br>$(pq)^2 \leq 2^{-93.40}$ |
| II | $(E_0, E_1)$<br>$(p, q)$ | (4.5-round, 4.5-round)<br>$(\leq (2^{-4.68})^6, \leq (2^{-4.67})^6)$ | 9-round<br>$(pq)^2 \leq 2^{-112.08}$ |
| III | $(E_0, E_1)$<br>$(p, q)$ | (3.5-round, 6.5-round)<br>$(\leq (2^{-4.68})^2, \leq (2^{-4.67})^{12})$ | 10-round<br>$(pq)^2 \leq 2^{-130.76}$ |

上記の distinguisher で用いられている差分特性を以下に示す．distinguisher I における $E_0$ では，図 3.1 に示す 3.5 段差分特性が使われている．但し，図中の 32 ビットデータ $a$, $b$, $c$ はそれぞれ $w_8(a) = 1$, $w_8(b) = 4$, $w_8(c) = 4$ を満たす非ゼロの値である．

distinguisher I の $E_1$ では，図 3.2 (左) に示す 5.5 段差分特性が使われている．但し，図中の 32 ビットデータ $d$, $e$, $f$, $g$, $h$ はそれぞれ $w_8(d) = 1$, $w_8(e) = 4$, $w_8(f) = 4$, $w_8(g) = 1$, $w_8(h) = 4$ を満たす非ゼロの値である．またケース III の $E_1$ での差分特性は，distinguisher I の $E_1$ の差分特性に 1 段追加することで得られる (図 3.2 (右))．

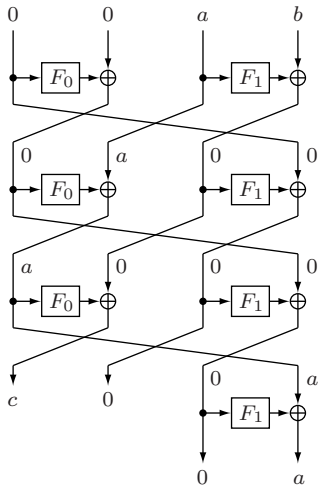


図 3.1: Distinguisher I の $E_0$ 内 3.5 段差分特性

### 3.1.5 拡張 Boomerang 攻撃

拡張 Boomerang 攻撃 (Amplified Boomerang attack) は適応的選択平文・暗号文攻撃である Boomerang 攻撃を選択平文攻撃に拡張した攻撃である [32]．この拡張のアイデアは，平文 (入力) 差分 $\alpha$ をとる多くの平文の暗号化を行い，その中から Boomerang 攻撃に必要な条件を満たす quartet を見つけるというものである．

文献 [32] には，$n$ をブロックサイズとすると，$N$ 個の平文ペアから $N^2 2^{-(n+1)} p^2 q^2$ 個の right quartet が得られることが示されている．よって 9 段 CLEFIA では $N = 2^{111.30}$ 個の平文ペアから 1 個の right quartet が得られることが期待される．これらの平文ペアを集めるには $2^8 \times 4$ 個の平文ペアからなる $2^{92.30}$ 個の structure が必要である．よって，CLEFIA に対する拡張 Boomerang 攻撃には $2^{92.30} \times 2^8 \times 4 = 2^{102.30}$ 個の選択平文が必要となる．

拡張 Boomerang 攻撃シナリオでは，9 段までランダム置換との識別が可能である．鍵回復攻撃ではこれに加えて 2, 3 段伸ばすのが限界と考えられるため，仕様通りの CLEFIA は拡張 Boomerang 攻撃に対して十分な安全性をもっていると考えられる．

### 3.1.6 Rectangle 攻撃

Rectangle 攻撃は拡張 Boomerang 攻撃において，差分値 $\beta, \gamma$ を 1 つの値に固定せず，すべての可能性のある値を利用して攻撃計算量を改善する

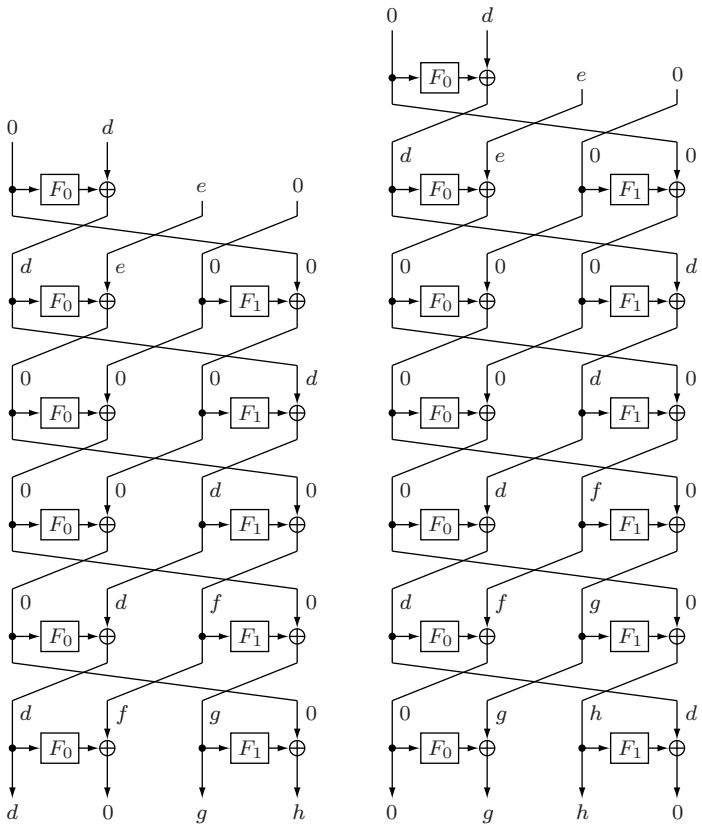


図 3.2: Distinguisher I の $E_1$ 内 5.5 段差分特性 (左) と III の $E_1$ 内 6.5 段差分特性 (右)

ものである．これにより，right quartet になる確率が向上し，入力差分 $\alpha$ をもつ $N$ 個の平文ペアから $N^2 2^{-n} \hat{p}^2 \hat{q}^2$ 個の right quartet が得られることが期待できる．但し， $\hat{p}, \hat{q}$ は下記のように定義できる．

$$\hat{p} = \sqrt{\sum_{\beta} \Pr^2[\alpha \to \beta]}, \qquad \hat{q} = \sqrt{\sum_{\gamma} \Pr^2[\gamma \to \delta]}. \tag{3.1}$$

これにより，表 3.2 の Case III が 10 段の distinguisher として攻撃に利用できる可能性が高くなる．表 3.2 の Case III の distinguisher の差分特性確率はランダム置換との識別のしきい値 $2^{-128}$ よりわずかに小さいが，Rectangle 攻撃において確率が増幅される可能性があるからである．しかしながら鍵回復攻撃はこれに加えて数段までしか適用できないと考えられるため，仕様通りの CLEFIA は Rectangle 攻撃に対して十分な安全性をもっていると考えられる．

### 3.1.7 Truncated 差分攻撃

Truncated 差分攻撃は Knudsen によって提案された，ブロック暗号に対する汎用的な攻撃である [37]．Truncated 差分とは，ある一部の差分情報のみが予測可能な差分値である．CLEFIA はバイト単位での処理を基本とした設計となっており，各バイトの差分値がゼロの場合 '0' を， 非ゼロの場合 '1' を割り当てた truncated 差分攻撃を適用するのが自然である．このアプローチは E2 や Camellia など多くのブロック暗号の安全性評価に適用されている [30, 46, 48, 72]．従来，$F$ 関数が SP 構造をもつ Feistel 暗号の Truncated 差分攻撃に対する安全性を体系的に評価する手法は未解決であった．これが CLEFIA の Truncated 差分攻撃に対する安全性評価を困難にしていた．

しかしながら，E2 や Camellia は CLEFIA と同じ Feistel 構造をもっているため，E2 や Camellia に関する結果から学ぶことができる．Truncated 差分攻撃に関する E2 と Camellia の大きな差は，E2 が SPS 構造の $F$ 関数をもち，Camellia が SP 構造の $F$ 関数を持っていることである．E2 と Camellia に関する最良の結果は，E2 における 7 段の truncated differential，$FL/FL^{-1}$ なしの Camellia における 9 段の truncated differential である [30, 46, 48, 72]．Camellia において 2 層目の S 層がないことが，Truncated 差分攻撃に対する耐性の差につながっていると考えられる．

$F$ 関数が SPS 構造の変形 CLEFIA を CLEFIA+S と呼ぶことにする．さらに，$F$ 関数内において，拡散行列切り替え法を使わず，すべて同じ拡散行列を利用する変形 CLEFIA+S を CLEFIA+S-D と呼ぶことにする．我々は，計算機シミュレーションにより，10 段以上の CLEFIA+S-D には攻撃に利用可能な truncated differential が存在しないことを確認した．もし，CLEFIA+S-D に 9 段 truncated differential が存在したとしても，E2 と Camellia の差から，仕様通りの CLEFIA-D は Truncated 差分攻撃に対して耐性があることが期待できる．さらに拡散行列切り替え法 DSM によって耐性が増すことが期待できる．

これらは Truncated 差分攻撃に対して得られる部分的な解析であるため，仕様通りの CLEFIA に対してより説得力のある評価方法が望まれる．

### 3.1.8 Truncated 線形攻撃

Truncated 線形攻撃は Camellia の設計者らにより安全性評価の過程で示された，ブロック暗号に対する汎用的な攻撃である [2]．差分攻撃と線形攻撃の双対性 [44] により，Truncated 差分攻撃と同様の手法で Truncated 線形攻撃に対する安全性を評価することができる．従って，仕様通りの CLEFIA は DSM を用いない場合でも Truncated 線形攻撃に対して耐性

を持っていると考えられる．さらに，DSM を用いることでより耐性が高まると期待できる．

### 3.1.9 不能差分攻撃

不能差分とは，差分確率が 0 の差分経路，すなわち決して起こりえない差分経路のことである．このような不能差分を用いて，正しくない鍵候補を排除し，正しい鍵を見つけることが可能である [8].

CLEFIA には以下の 9 段の不能差分パスが存在する [70].

- $(0, \alpha, 0, 0) \overset{9r}{\not\rightarrow} (0, \alpha, 0, 0)$
- $(0, 0, 0, \alpha) \overset{9r}{\not\rightarrow} (0, 0, 0, \alpha)$

但し，32 ビットデータ $\alpha \in \{0,1\}^{32}$ は非ゼロの値である．図 3.3 に 1 つ目の 9 段不能差分パスを示す．この図で ‘+’ は非ゼロの差分値，‘*’ は未知の差分値を示す．2 つ目の不能差分パスはこの不能差分パスのすべての差分値のビット位置を巡回シフトすることで得られる．

CLEFIA 設計者らにより，この 9 段不能差分パスを用いた不能差分攻撃が示されている [70]．表 3.2 に攻撃可能段数および攻撃計算量を示す．

表 3.2: CLEFIA に対する不能差分攻撃 [70]

| 段数 | 鍵長 | 鍵ホワイトニング | 既知平文数 | 計算量 |
|---|---|---|---|---|
| 10 | 128, 192, 256 | 有 | $2^{101.7}$ | $2^{102}$ |
| 11 | 192, 256 | 有 | $2^{103.5}$ | $2^{188}$ |
| 12 | 256 | 無 | $2^{103.8}$ | $2^{252}$ |

[70] において不能差分攻撃に対する安全性評価が示された後，新たな評価結果が数多く発表されている [73, 78, 79, 83, 86].

Wang らは，[70] と同じ 9 段不能差分パスを用い，テーブル参照の利用や鍵空間の削減によって鍵導出計算量を削減し，128 ビット鍵 CLEFIA を 12 段まで，192 ビット鍵 CLEFIA を 13 段まで，256 ビット鍵 CLEFIA を 14 段まで攻撃可能とした [83].

角尾らは不能差分パス探索手法を改良し，CLEFIA に用いられている DSM 拡散行列の性質を利用して以下のような 9 段不能差分パスを発見した [79].

- $(0, \alpha_{in}, 0, 0) \overset{9r}{\not\rightarrow} (0, \alpha_{out}, 0, 0)$

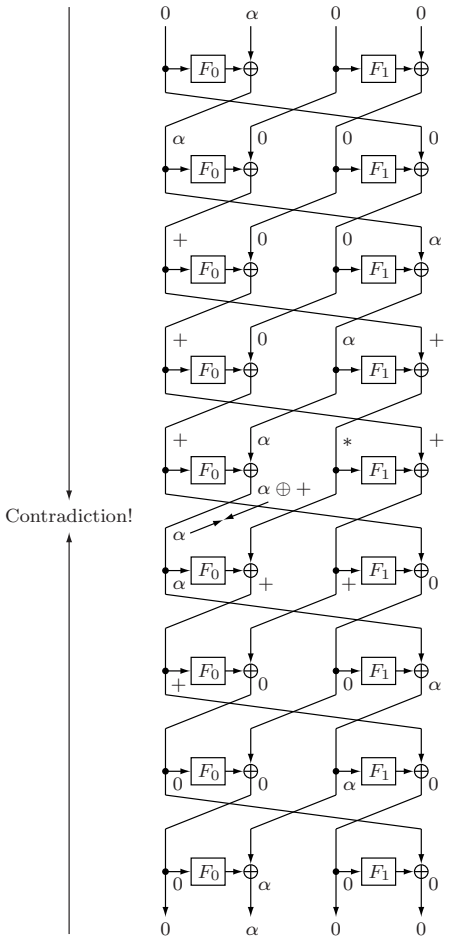


図 3.3: 9 段不能差分特性

- $(0,0,0,\alpha_{in}) \overset{9r}{\not\rightarrow} (0,0,0,\alpha_{out})$

但し，32 ビットデータ $\alpha_{in}$, $\alpha_{out}$ は表 3.3 に示す値をとる． 表 3.3 中の a, b は任意の非ゼロの 8 ビットデータである．

角尾らはこの 9 段不能差分パスを用い，かつ S-box の差分値分布の利用やホワイトニング鍵の移動 (等価変形) などにより鍵導出計算量を削減し，128 ビット鍵 CLEFIA を 12 段まで，192 ビット鍵 CLEFIA を 13 段まで，256 ビット鍵 CLEFIA を 14 段まで攻撃可能とした [79]．さらにこの攻撃は，辻原ら [78] により，32 ビットデータ $\alpha_{in}$, $\alpha_{out}$ が表 3.4 に示す値をとりうる不能差分パスを用いることで，より少ない計算量で多くの鍵ビット数が回復できるように改良されている．但し，表 3.4 の a, b, c は任意の非ゼロの 8 ビットデータである．

表 3.3: $\alpha_{in}$, $\alpha_{out}$ としてとりうる差分値 [79]

| $\alpha_{in}$ | $\alpha_{out}$ |
|---|---|
| (0,0,0,a) | (0,0,b,0), (0,b,0,0), (b,0,0,0) |
| (0,0,a,0) | (0,0,0,b), (0,b,0,0), (b,0,0,0) |
| (0,a,0,0) | (0,0,0,b), (0,0,b,0), (b,0,0,0) |
| (a,0,0,0) | (0,0,0,b), (0,0,b,0), (0,b,0,0) |

表 3.4: $\alpha_{in}$, $\alpha_{out}$ としてとりうる差分値 [78]

| $\alpha_{in}$ | $\alpha_{out}$ |
|---|---|
| (0,0,0,a) | (0,0,b,c), (0,b,0,c), (b,0,0,c) |
| (0,0,a,0) | (0,0,b,c), (0,b,c,0), (b,0,c,0) |
| (0,a,0,0) | (0,b,0,c), (0,b,c,0), (b,c,0,0) |
| (a,0,0,0) | (b,0,0,c), (b,0,c,0), (b,c,0,0) |
| (0,0,b,c) | (0,0,0,a), (0,0,a,0) |
| (0,b,0,c) | (0,0,0,a), (0,a,0,0) |
| (b,0,0,c) | (0,0,0,a), (a,0,0,0) |
| (0,b,c,0) | (0,0,a,0), (0,a,0,0) |
| (b,0,c,0) | (0,0,a,0), (a,0,0,0) |
| (b,c,0,0) | (0,a,0,0), (a,0,0,0) |

2010 年 1 月現在，各鍵長の CLEFIA に対して，最も少ない計算量で最も長い段数まで攻撃可能な不能差分攻撃は，辻原ら [78] によるものである．この結果を表 3.5 に示す.

また，これと独立して Sun らが不能差分攻撃の結果を発表していたが [73]，その後著者らにより取り下げられている.

また，Inscrypt 2008 にて Zhang らにより，角尾ら [79] によって示された 9 段不能差分攻撃を用いて，ホワイトニング鍵なしの 128 ビット鍵 CLEFIA が 14 段まで攻撃可能という発表がされたが [86]，CLEFIA 設計者が Pre-proceedings に記された計算量の見積もりに誤りがあることを著者に指摘したところ [85]，Springer より出版された Proceedings [87] では，計算量見積もりの記述が削除され，結論で「我々の攻撃シナリオが成功するかどうかは，検証を待っているところである」という記載に変更されている.

不能差分攻撃は CLEFIA 設計者および多くの外部研究者によって評価

表 3.5: CLEFIA に対する最も効率のよい不能差分攻撃 [78]

| 段数 | 鍵長 | 鍵ホワイトニング | 既知平文数 | 計算量 | メモリ |
|---|---|---|---|---|---|
| 12 | 128, 192, 256 | 有 | $2^{111.0}$ | $2^{111}$ | $2^{81}$ |
| 13 | 192, 256 | 有 | $2^{111.8}$ | $2^{155}$ | $2^{112}$ |
| 14 | 256 | 有 | $2^{112.3}$ | $2^{220}$ | $2^{113}$ |

が行われてきているが，表 3.5 の攻撃計算量が示すように，仕様通りの CLEFIA は不能差分攻撃に対して十分な安全性をもっていることを示していると考えられる．

### 3.1.10 飽和攻撃

飽和攻撃 (Saturation cryptanalysis) はもともと Daemen ら [19] によりブロック暗号 Square に対する専用の攻撃として提案されたもので，“スクエア攻撃 (Square attack)” と呼ばれていた．またこのタイプの攻撃は multiset 攻撃と呼ばれることもある．

#### バイト単位での飽和攻撃による識別攻撃

典型的な飽和攻撃は，ブロック暗号のバイト指向構造 (バイト単位での演算) を利用しており，このタイプの攻撃は AES に対しても有効である [23]. CLEFIA も強いバイト指向構造をもっているため，まず，バイト単位での飽和攻撃を検討する．

$X = \{X_i | X_i \in \{0,1\}^8\}$ $(0 \leq i < 2^8)$ を 256 個の 8 ビット値の集合とし，$X_i$ の状態を以下のように分類する．

- `Const (C)`：もし $\forall i, j \;\; X_i = X_j$ を満たす場合
- `All (A)`：　もし $\forall i, j \;\; i \neq j \Leftrightarrow X_i \neq X_j$ を満たす場合
- `Balance (B)`：もし $\bigoplus_i X_i = 0$ を満たす場合
- `Unknown (U)`：不明の場合

次に，平文 256 個があり，1 バイトのみ `All` で，残りのバイトが `Const` という条件を満たしているとする．このとき，5 段 CLEFIA の入出力には以下のような関係が存在する．

- ((C C C C) (C C C A) (C C C C) (C C C C))
  $\xrightarrow{5r}$ ((U U U U) (U U U U) (B B B B) (U U U U))

- ((C C C C) (C C C C) (C C C C) (C C C A))
  $\xrightarrow{5r}$ ((B B B B) (U U U U) (U U U U) (U U U U))

さらに，入力の (C C C A) は (C C A C), (C A C C), (A C C C) のいずれでも同じ出力状態となる．よって，5 段 CLEFIA に対して 8 種類の飽和パスが存在する．一番目の飽和パスを図 3.4 に示す．これらの飽和パスを用い，出力の該当バイトに Balance が現れるかどうかを観測することで，5 段 CLEFIA をランダム置換と識別することができる．

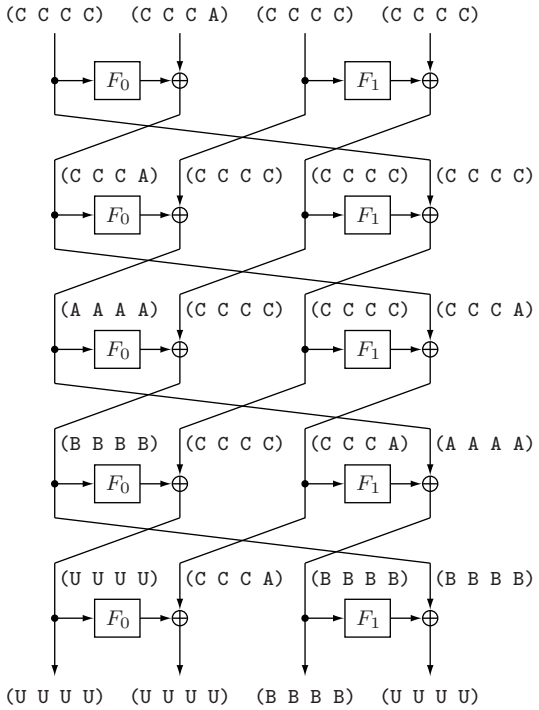


図 3.4: 5 段 CLEFIA 飽和パスの例

### 7 段 CLEFIA に対する鍵回復攻撃

上記飽和パスを用いた鍵回復攻撃を考察する．上記の 5 段飽和パスに 2 段追加し (図 3.5 参照)，ラウンド鍵 $RK_{10}$ と $RK_{13}$ を導出する．この攻撃では，暗号文の一部 $C_0^{(7)}, C_2^{(7)}, C_3^{(7)}$ とラウンド鍵 $RK_{10}, RK_{13}$ から 5 段目の出力値 32 ビットを復号できることを利用し，(B B B B) かどうかを判定する．

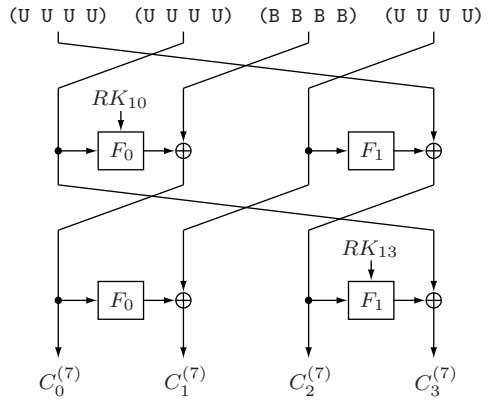


図 3.5: 7 段 CLEFIA に対する鍵回復攻撃

ラウンド鍵 $(RK_{10}, RK_{13}) \in \mathcal{K}$ を以下のように導出する．

1. ラウンド鍵 $RK_{10}$，$RK_{13}$ の値をそれぞれ $k_{guess10}$，$k_{guess13}$ と推定する．

2. それぞれの推定値に対し，

   - それぞれの暗号文に対し，以下を計算する．

     $$Z_i = F_0(k_{guess13}, C_2^{(7)}) \oplus C_3^{(7)}$$

     次に以下を計算し，

     $$Y_i = F_1(k_{guess10}, Z_i) \oplus C_0^{(7)}$$

     すべての $Y_i$ に対して和 $Y = \bigoplus_i Y_i$ を計算する．

3. もし $Y = 0$ なら，推定値 $k_{guess10}, k_{guess13}$ はラウンド鍵 $RK_{10}, RK_{13}$ の候補である．$Y \neq 0$ であれば，推定値は正しい値ではないため棄却する．

鍵候補が上記の棄却ステップにおいて生き残る確率は $2^{-32}$ である．よって，256 個平文が 3 セットあれば正しい鍵を 1 つに絞り込むことが可能である．攻撃計算量は $2^{64} \times 2^8 \times 2^8 = 2^{80}$ 回の F 関数の計算である．よって，この攻撃は 128 ビット鍵，192 ビット鍵，256 ビット鍵の 7 段 CLEFIA いずれにも適用可能である．

さらに攻撃可能段数を増やせる可能性もあるが，上記の飽和パスを使う限りでは，増加段数は限られると思われる．しかしながら，飽和攻撃の単位を 8 ビットから 32 ビットにすると，より多くの段数をもつ CLEFIA の攻撃が可能となる．これについて次に述べる．

### 32 ビットワード単位での飽和攻撃による識別攻撃

次に, 32 ビットワード単位での飽和攻撃を考察する.

$X = \{X_i | X_i \in \{0,1\}^{32}\}$ $(0 \leq i < 2^{32})$ を $2^{32}$ 個の 32 ビット値の集合とし，$X_i$ の状態をバイト単位での飽和攻撃と同様に分類する.

- Const (C)： もし $\forall i,j \;\; X_i = X_j$ を満たす場合
- All (A)： もし $\forall i,j \;\; i \neq j \Leftrightarrow X_i \neq X_j$ を満たす場合
- Balance (B)：もし $\bigoplus_i X_i = 0$ を満たす場合
- Unknown (U)：不明の場合

この分類を用いて 6 段 CLEFIA の入出力関係を以下のように書くことができる.

- (C A C C) $\xrightarrow{6r}$ (B U U U)
- (C C C A) $\xrightarrow{6r}$ (U U B U)

一番目の飽和パスを図 3.6 に示す．これらの飽和パスを用い，出力の該当ワードに Balance が現れるかどうかを観測することで，6 段 CLEFIA をランダム置換と識別することができる.

これらの 6 段飽和パスを 8 段飽和パスに拡張することができる．まず, どのように 7 段飽和パスに拡張するかについて説明する．$\mathtt{A}_{(64)}$ を All 状態の 64 ビット値とし，これを $\mathtt{A}_{(64)} = \mathtt{A}_{0(64)} \mid \mathtt{A}_{1(64)}$ のように 2 分割することを考える．これを用いると，以下のような 7 段飽和パスが得られる.

- (C C $\mathtt{A}_{0(64)}$ $\mathtt{A}_{1(64)}$) $\xrightarrow{7r}$ (B U U U)
- ($\mathtt{A}_{0(64)}$ $\mathtt{A}_{1(64)}$ C C) $\xrightarrow{7r}$ (U U B U)

これらの識別には $2^{64}$ 個の平文を必要とする．以下，この飽和パスについて説明する．1 段目の後，(C C $\mathtt{A}_{0(64)}$ $\mathtt{A}_{1(64)}$) は (C $\mathtt{A}_{0(64)}$ $\mathtt{A'}_{1(64)}$ C) となり，連続セグメント $\mathtt{A}_{0(64)}$ | $\mathtt{A'}_{1(64)}$ は All となる．(C $\mathtt{A}_{0(64)}$ $\mathtt{A'}_{1(64)}$ C) は $2^{32}$ 個の (C A C C) ストラクチャを含んでおり，第 3 番目の定数ワードはすべての $2^{32}$ 種類の値をとりうる．よって上記の 6 段飽和パスで示したように，出力において Balance の状態が保たれる.

8 段飽和パスへの拡張も同様に得ることができる．$\mathtt{A}_{(96)}$ を All 状態の 96 ビット値とし，これを $\mathtt{A}_{(96)} = \mathtt{A}_{0(96)} \mid \mathtt{A}_{1(96)} \mid \mathtt{A}_{2(96)}$ のように 3 分割することを考える．これを用いると，以下のような 8 段飽和パスが得られる.

- ($\mathtt{A}_{0(96)}$ C $\mathtt{A}_{1(96)}$ $\mathtt{A}_{2(96)}$) $\xrightarrow{8r}$ (B U U U)

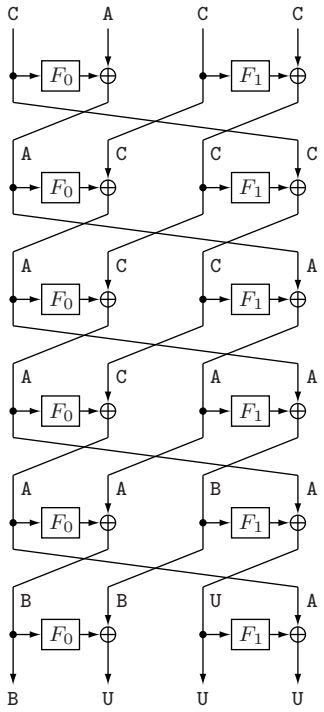


図 3.6: 6 段 CLEFIA 飽和パスの例

- $(\mathtt{A}_{0(96)}\ \mathtt{A}_{1(96)}\ \mathtt{A}_{2(96)}\ \mathtt{C}) \xrightarrow{8r} (\mathtt{U}\ \mathtt{U}\ \mathtt{B}\ \mathtt{U})$

これらの識別には $2^{96}$ 個の平文を必要とする.

次に，上記飽和パスを用いた 9 段飽和攻撃と 10 段飽和攻撃を示す.

### 9 段 CLEFIA に対する鍵回復攻撃

上記 8 段飽和パスを用い，1 段伸ばして 9 段目のラウンド鍵 $RK_{17}$ を導出する.

1. $(\mathtt{A}_{0(96)}\ \mathtt{C}\ \mathtt{A}_{1(96)}\ \mathtt{A}_{2(96)})$ の形をもつ $2^{96}$ 個の平文を入力する.
2. 出力ワード $C_2^{(9)}$ に対し，出現値の頻度をカウントし，奇数回出現した 32 ビット値のリスト $LIST$ を作る.
3. 出力ワード $C_3^{(9)}$，すべての $2^{96}$ 個の値の排他的論理和を計算し，$Y$ とする.
4. $LIST$ 中のすべての候補値 $l_i \in LIST$ とラウンド鍵 $RK_{17}$ の候補値 $k_{guess}$ に対し，以下を計算する.

$$Z = \bigoplus_i F_1(k_{guess}, l_i) \ .$$

- $Z = Y$ ならば，$k_{guess}$ は $RK_{17}$ の候補であり，$Z \neq Y$ ならば，$k_{guess}$ は $RK_{17}$ の正しい値ではないので棄却する．

鍵候補が上記の棄却ステップにおいて生き残る確率は $2^{-32}$ である．よって，256 個平文が 3 セットあれば正しい鍵を 1 つに絞り込むことが可能である．攻撃計算量は，$LIST$ 作成のための約 $2^{31}$ 回の F 関数の計算である．よって，この攻撃は 128 ビット鍵，192 ビット鍵，256 ビット鍵の 9 段 CLEFIA いずれにも適用可能である．

### 10 段 CLEFIA に対する鍵回復攻撃

上記と同じ 8 段飽和パスを用い，2 段伸ばして 9，10 段目のラウンド鍵 $RK_{17}$, $RK_{18}$ を導出する．基本的な攻撃シナリオは 3.1.10 節のバイト飽和を用いた 7 段 CLEFIA への攻撃と同じである．

ラウンド鍵 $(RK_{17}, RK_{18}) \in \mathcal{K}$ を以下のように導出する．

1. ラウンド鍵 $RK_{17}$, $RK_{18}$ の値をそれぞれ $k_{guess17}$, $k_{guess18}$ と推定する．

2. それぞれの推定値に対し，

   - それぞれの暗号文に対し，以下を計算する．

     $$Z_i = F_0(k_{guess18}, C_0^{(10)}) \oplus C_1^{(10)}$$

     次に以下を計算し，

     $$Y_i = F_1(k_{guess17}, Z_i) \oplus C_2^{(10)}$$

     すべての $Y_i$ に対して和 $Y = \bigoplus_i Y_i$ を計算する．

3. もし $Y = 0$ なら，推定値 $k_{guess17}, k_{guess18}$ はそれぞれ，ラウンド鍵 $RK_{17}, RK_{18}$ の候補である．$Y \neq 0$ であれば，推定値は正しい値ではないため棄却する．

鍵候補が上記の棄却ステップにおいて生き残る確率は $2^{-32}$ である．よって，256 個の平文が 2 セット以上あれば正しい鍵 1 つに絞り込むことが可能である．攻撃計算量は，それぞれの推定鍵 (64 ビット) に対し，$2^{32}$ 回の $F_0$ 関数の計算量と $2^{32}$ 回の $F_1$ 関数の計算量が必要であるため，およそ

$2^{128}$ 回の F 関数の計算が必要となる．よって，この攻撃は 128 ビット鍵，192 ビット鍵，256 ビット鍵の 10 段 CLEFIA いずれにも適用可能である．

段数を縮小した CLEFIA に対する飽和攻撃による鍵回復攻撃を示したが，同様の攻撃シナリオのもとでは，今後攻撃可能段数が伸びたとしても 2，3 段にとどまるであろうと考えられる．よって，仕様通りの CLEFIA は飽和攻撃に対して強い耐性をもつと考えられる．

### 3.1.11 Gilbert-Minier 衝突攻撃

Gilbert-Minier 衝突攻撃は飽和攻撃の一種であり，Gilbert と Minier によって提案された攻撃である [25]．Gilbert と Minier のオリジナル論文では，Rijndael を攻撃するために特別な 4 段 distinguisher を用いており，同じ識別攻撃をそのまま CLEFIA に適用するのは困難であると考えられる．Rindael に対して通常の飽和攻撃では 6 段まで攻撃できるのに対し，Gilbert-Minier 衝突攻撃では 7 段まで攻撃可能である．CLEFIA に対しても同様に攻撃が適用可能と思われるが，CLEFIA に対しても攻撃可能段数が 2，3 段増えるにとどまると考えられる．よって，仕様通りの CLEFIA は Gilbert-Minier 衝突攻撃に対し，強い耐性をもつと考えられる．

### 3.1.12 高階差分攻撃

高階差分攻撃は Knudsen らによって考案され [28, 37, 40]，非線形要素が次数が低いブール多項式で表現可能なブロック暗号に適用される．高階差分攻撃は，ブロック暗号のある中間ビットが $d$ 次のブール多項式で表現できた時，$(d+1)$ 階差分をとると 0 になることを利用するものである．

CLEFIA の S-box $S_0$ と $S_1$ をブール多項式次数はそれぞれ 6 と 7 である．詳細には，S-box $S_0$ は 4 つの小さな S-box $SS_0$, $SS_1$, $SS_2$, $SS_3$ からなる．$SS_i : \{0,1\}^4 \rightarrow \{0,1\}^4$, $SS_i(x_0, x_1, x_2, x_3) = (y_0, y_1, y_2, y_3)$ とし，$y_j$ を変数 $(x_0, x_1, x_2, x_3)$ のブール多項式で表現した場合，

$$\deg(y_j) = 3$$

がすべての $0 \leq j \leq 3$ で成り立つことを確認した．さらに，$S_0 : \{0,1\}^8 \rightarrow \{0,1\}^8$, $S_0(x_0, x_1, \ldots, x_7) = (y_0, y_1, \ldots, y_7)$ とし，$y_j$ を変数 $(x_0, x_1, \ldots, x_7)$ のブール多項式で表現した場合，

$$\deg(y_j) = 6$$

がすべての $0 \leq j \leq 7$ で成り立つことを，具体的なブール多項式表現を導いて確認した．

$S_1$ は GF($2^8$) 上の逆元関数をベースとしており，8 ビット S-box では最も高い代数次数をもつ．すなわち，$S_1 : \{0,1\}^8 \rightarrow \{0,1\}^8$, $S_1(x_0, x_1, \ldots, x_7) = (y_0, y_1, \ldots, y_7)$ とし，$y_j$ を変数 $(x_0, x_1, \ldots, x_7)$ のブール多項式で表現した場合，

$$\deg(y_j) = 7$$

がすべての $0 \leq j \leq 7$ で成り立つことを，具体的なブール多項式表現を導いて確認した．

中間ビットの次数は，少なくとも次数 6 次の非線形処理を行う S-box で処理されるたびに次数が指数関数的に増加していくことが期待される．

CLEFIA で 3 つの S-box を通過した後は $6^3 > 128$ となるため，$(d+1)$ 階差分が 0 になるようなデータを集めることは困難であると考えられる．CLEFIA に高階差分攻撃を適用できるのは限定的になると考えられ，仕様通りの CLEFIA はこの攻撃に対して十分な安全性をもっていると考えられる．

### 3.1.13 補間攻撃

補間攻撃は Jakobsen と Knudsen により [28] にて提案され，非線形要素が数学的に簡易に表現可能なブロック暗号に適用可能な攻撃である．補間攻撃の原理は，暗号文が平文の多項式または有理式で表され，その未知係数が $N$ 個のとき，$N$ 個の暗号文・平文ペアからその多項式または有理式が構成できるというものである．攻撃者がその多項式または有理表現を構成できると，鍵を知ることなくあらゆる平文から対応する暗号文へ，あるいはあらゆる暗号文から対応する平文に変換することが可能となる．補間攻撃の計算量や攻撃に必要な平文・暗号文ペア数は $N$ により決まるため，$N$ をなるべく大きくすることが重要である．$N$ が非常に大きく攻撃者が $N$ ペアの平文・暗号文を集めることが現実的でなければ，そのブロック暗号は補間攻撃に対して安全であるということになる．

CLEFIA の S-box $S_0$ と $S_1$ に対して，GF($2^8$) 上多項式で表現した際の項数を評価した．$S_0 : \mathrm{GF}(2^8) \rightarrow \mathrm{GF}(2^8)$, $S_0(x) = y$ とし，$y$ を $x$ の多項式で表したとき，あらゆる既約多項式に対しても，最小項数は 244 になることを確認した．

S-box $S_1$ に対しても，$S_1 : \mathrm{GF}(2^8) \rightarrow \mathrm{GF}(2^8)$, $S_1(x) = y$ とし，$y$ を $x$ の多項式で表したとき，あらゆる既約多項式に対しても，最小項数は 252 になることを確認した．

$S_0$, $S_1$ のいずれにおいても，項数は，GF($2^8$) 上の置換における最大値である 255 に近い値をとる．さらに，$S_0$, $S_1$ の 2 種類の S-box を用いるこ

とで，数段でそれぞれの S-box の数学的に見て簡易な構造が崩れると考えられる．

よって，補間攻撃が CLEFIA にとって脅威になるとは考えにくい．

### 3.1.14 XSL 攻撃

S-box の代数表現には，多くの興味深い非線形特性が含まれているが，ブロック暗号を GF(2) または GF($2^8$) 上の多変数多項式の連立方程式で表現すると，次数が低く，疎な overdefined な (項の数より方程式の数の方が多い) 連立方程式で表せる可能性があり，Courtois と Pieprzyk により主張されているように，これが攻撃に結びつく可能性がある [18]．

以下に，Courtois と Pieprzyk の手法 [18] に倣い，CLEFIA の簡易版 CLEFIA-I の XSL 攻撃に対する計算量を見積もる．最初の XSL 攻撃を [29] のように攻撃の目的をラウンド鍵を導出するものとする．すなわち，この攻撃シナリオでは鍵スケジュール部を考慮しない．

CLEFIA-I は，CLEFIA のすべての 4-bit S-box $SS_0$, $SS_1$, $SS_2$, $SS_3$ を恒等変換 $I$，すなわち，$I : \{0,1\}^4 \rightarrow \{0,1\}^4$, $I(x) = x$ に置き換えることで得られるアルゴリズムとする．

置き換えられた 4-bit S-box は XSL 攻撃に対する安全性の向上に寄与しないため，CLEFIA-I を攻撃するのは CLEFIA を攻撃するよりずっと容易であることに注意されたい．

Courtois と Pieprzyk [18] によると，XSL 攻撃の計算量は以下のように見積もられる．

$$T^{\omega} \text{ with } T \approx (t-\rho)^P \binom{S_{\text{inv}}}{P} \tag{3.2}$$

但し

- $T$: 連立方程式に含まれる全項数
- $\omega$: ガウス消去法の計算量指数
- $t$: $S_1$ の表現に含まれる単項の数
- $\rho$: 連立方程式に含まれる式の数
- $S_{\text{inv}}$: 攻撃で考慮される S-box の数
- $P$: 攻撃パラメータ (整数)

である．

今，$s$ ビット入出力の S-box を GF($2^s$) 上逆元関数のアフィン変換関数とすると，$\rho = 3s-1$ 個の確率 1 で成り立つ双アフィン等式と確率 $1-2^{-s}$ 個

で成り立つもう一つの等式が得られる [18]．[18] と同じ議論により，$t = 81$ および $\rho = 23$ が得られる．

次に $S_{\mathrm{inv}}$ を攻撃で考慮される全 S-box $S_1$ の総数とする．鍵長が 128 ビットの場合，CLEFIA の段数 $r$ は 18 であり，各ラウンドに F 関数が 2 つ，それぞれの $F_i$ は 2 つの S-box からなり，攻撃に 2 組の平文-暗号文ペアが必要となることから，以下が得られる．

$$S_{\mathrm{inv}} = 2 \times 2 \times 18 \times 2 = 144$$

ラウンド鍵は $K$ と $L$ を含むため，ラウンド鍵を一意に定めるのに 2 組の既知平文-暗号文ペアが必要になることに注意されたい．同様に，鍵長が 192 ビットの場合，CLEFIA の段数 $r$ は 22 であり，$S_{\mathrm{inv}} = 2 \times 2 \times 22 \times 4 = 352$，鍵長が 256 ビットの場合，CLEFIA の段数 $r$ は 26 であり，$S_{\mathrm{inv}} = 2 \times 2 \times 26 \times 4 = 416$ となる．ここではラウンド鍵は $K_L$, $K_R$, $L_L$, $L_R$ を含むため，ラウンド鍵を一意に定めるのに 4 組の既知平文-暗号文ペアが必要になることに注意されたい．

攻撃パラメータ $P$ に対してはいくつか条件があり [18]，[29] より，$P$ は以下のように与えられる．

$$P = \frac{t - \rho}{s + \frac{t'}{S_{\mathrm{inv}}}}$$

但し，我々のケースの場合，$t' = 25$ である．これより，$r = 18, 22, 26$ の場合，$P = 8$ となる．

Courtois と Pieprzyk [18] は $\omega$ の値を Coppersmith と Winograd [17] によって導かれている既知の最良の値 $\omega = 2.376$ と仮定している．[18] によると，[17] の著者はこのアルゴリズムの定数ファクタは不明としており，非常に大きな値であることも想定される．従って，このアルゴリズムが現実的な計算量で適用できるかどうかという議論がある．よって，我々は $\omega = 2.376$ と $\omega = 3$ の両方のケースで攻撃計算量の見積もりを行う．

上記の値と式 (3.2) より，$r = 18$ の CLEFIA-I に対し，(連立方程式に含まれる) 全項数は $T = 81^8 \binom{144}{8} > 2^{50+41} = 2^{91}$ と見積もられる．これより攻撃計算量は $T^{2.376} = 2^{216}$ または $T^3 = 2^{273}$ となる．$r = 22$ の CLEFIA-I に対しては，$T = 81^8 \binom{352}{8} > 2^{50+52} = 2^{102}$，よって攻撃計算量は $T^{2.376} = 2^{242}$ または $T^3 = 2^{306}$ と見積もられる．$r = 26$ の CLEFIA-I に対しては，$T = 81^8 \binom{416}{8} > 2^{50+54} = 2^{104}$，よって攻撃計算量は $T^{2.376} = 2^{247}$ または $T^3 = 2^{312}$ と見積もられる．

この計算量見積もりを表 3.6 をまとめる．これまでの議論に照らして，これらの数字の解釈は細心の注意を払う必要がある．また一方で，XSL 攻撃に必要とされる実際の計算量は現時点では明らかになっておらず，議論のあるところである [42,51]．

表 3.6: CLEFIA-I に対する XSL 攻撃の計算量見積もり

| | $\omega = 2.376$ | $\omega = 3$ |
|---|---|---|
| $r = 18$ | $2^{216}$ | $2^{273}$ |
| $r = 22$ | $2^{242}$ | $2^{306}$ |
| $r = 26$ | $2^{247}$ | $2^{312}$ |

さらに，我々の計算量見積もりでは $S_0$ を除いており，これは CLEFIA において $S_0$ が XSL 攻撃に対する安全性の向上に何の寄与もしないという極めて非現実的な仮説である．よって CLEFIA に対する実際の XSL 攻撃はこの見積もりよりもずっと困難であると考えられる．

以上で，GF(2) 上の XSL 攻撃について述べてきたが，GF($2^8$) 上の XSL 攻撃が効率的な鍵回復攻撃につながる可能性もある [18]．現時点では，[42] でも指摘されているように，このアプローチの有効性を確認することはできておらず，XSL 攻撃による効率的な鍵攻撃が存在するという具体的な主張ができるほど XSL 攻撃の計算量見積もりに正確性はない．さらに，$S_0$, $S_1$ の 2 種類の S-box により，1 種類の場合に比べ XSL 攻撃の適用が本質的に困難になっており，XSL 攻撃への耐性向上につながると考えられる．

### 3.1.15 カイ二乗攻撃

カイ二乗攻撃 ($\chi^2$ cryptanalysis) はブロック暗号解析のための統計的攻撃の一種である．

この攻撃はもともと Vaudenay [80] によって提案され，Gilbert ら [24] と Knudsen，Meier [35] によって独立に RC6 に適用された Knudsen と Meier は，[35] において，一般の鍵に対して 15 段まで，弱鍵 (weak key) に対して 17 段まで攻撃を成功させている．

カイ二乗攻撃に対する RC6 の脆弱性は，RC6 に用いられているデータ依存ローテーションにある．我々は CLEFIA のアルゴリズムには，カイ二乗攻撃に有効な相関関係 (correlation) は存在しないと考えている．よって，カイ二乗攻撃は仕様通りの CLEFIA には有効でないと考える．

## 3.2 暗号解析 II — 鍵スケジュール部

本節では，鍵スケジュール部を含む CLEFIA の安全性について評価する．

### 3.2.1 スライド攻撃

スライド攻撃は Biryukov と Wagner によって提案された，ブロック暗号の鍵スケジュール部解析のための汎用的な技術である [12]．スライド攻撃への対抗策として，各段で独立なラウンド定数を用いるという方法が知られている．CLEFIA のアルゴリズムでは，仕様に示されているように，ラウンド定数が用いられており，スライド攻撃に対する十分な耐性があると考えられる．

### 3.2.2 関連暗号攻撃

関連暗号攻撃 (Related-cipher attack) は Wu によって提案された，ブロック暗号の鍵スケジュール部解析のための汎用的な技術である [84]．鍵長に依存して段数が変わるブロック暗号を考える．このとき，各段のラウンド鍵は追加段のラウンド鍵を除き，どの鍵長に対しても同一であるとする．このとき，これらの異なる段数をもつブロック暗号を ‘関連している (related)’ といい，ラウンド鍵が同じであれば，長い段数の暗号は短い段数の暗号化結果を用いて簡単に解析できてしまう．192 ビット鍵の CLEFIA と 256 ビット鍵の CLEFIA は鍵スケジュール部アルゴリズムが類似しているため，この攻撃のリスクがある．この関連暗号攻撃を避けるため，CLEFIA ではそれぞれの鍵長に対して異なるラウンド定数を用いている [71]．これはスライド攻撃への対策と同じである．これにより，CLEFIA は関連暗号攻撃に対して十分な耐性があると考えられる．

### 3.2.3 関連鍵攻撃

関連鍵攻撃 (Related-key cryptanalysis) は Biham [7] によって提案された攻撃法である．この攻撃は関連する鍵を用いた複数の暗号化処理から得られる情報を利用する．このコンセプトを Kelsey らが [33] において，関連鍵差分 (related-key differential) という形で利用している．この関連鍵差分は，2 つの関連鍵のもとでの 2 回の暗号化処理において，差分値がどのように展開されるかというものである．

関連鍵差分 (related-key differential) は以下の式の成立確率が十分に高く (あるいはゼロに) なるような平文差分 $\Delta P$，暗号文差分 $\Delta C$，鍵差分

$\Delta K$ の三つ組である．

$$\Pr_{P,K}[E_K(P) \oplus E_{K\oplus\Delta K}(P \oplus \Delta P) = \Delta C]$$

128 ビット鍵 CLEFIA では，鍵スケジュール部において中間鍵 $L$ は $GFN_{4,12}$ により生成される．$GFN_{4,12}$ は 12 ラウンドの 4 系列の Type-2 一般化 Feistel 構造である．表 2.1 に示すように，$GFN_{4,12}$ には少なくとも 28 個の active S-box が含まれるので，その最大差分特性確率は $DCP_{max} \leq 2^{28\times(-4.67)} = 2^{-130.76}$ となる．これより，どのような $\Delta K$ と $\Delta L$ に対しても，$(\Delta K \rightarrow \Delta L)$ の差分特性確率は $2^{-128}$ より小さくなる，すなわち，攻撃に有効な差分経路 $(\Delta K \rightarrow \Delta L)$ は存在しないことが期待される．これは，$\Delta L$ のすべての情報が必要な場合，どのような関連鍵差分 $(\Delta P, \Delta C, \Delta K)$ の確率も $2^{-128}$ より小さいことを示唆している．なぜなら $L$ のすべてのビットは連続する 2 段でラウンド鍵として利用されるからである．$\Delta L$ の一部の word が未知であるような $(\Delta K \rightarrow \Delta L)$ が利用されることもあるかもしれないが，このような distinguisher は未知の word が少なくとも 3 段ですべての word に伝播することから，攻撃への効果は限定的と考える．また，我々の知る限り，確率がゼロの関連鍵差分を用いた攻撃は知られていない．

192 ビットまたは 256 ビット鍵 CLEFIA では，鍵スケジュール部において中間鍵 $(L_L, L_R)$ は $GFN_{8,10}$ により生成される．$GFN_{8,10}$ は 10 ラウンドの 8 系列の Type-2 一般化 Feistel 構造である．$GFN_{8,10}$ のラウンド鍵は鍵長によって決まる固定定数である．表 2.3 に示すように，$GFN_{8,10}$ には少なくとも 29 個の active S-box が含まれるので，$2^{-128}$ より大きな確率をもつ差分特性は存在しない．すなわち，いかなる $(\Delta K_L, \Delta K_R)$ と $(\Delta L_L, \Delta L_R)$ に対しても，$((\Delta K_L, \Delta K_R) \rightarrow (\Delta L_L, \Delta L_R))$ の差分特性確率は $2^{-128}$ より小さくなる．従って，$(\Delta L_L, \Delta L_R)$ のすべてのビット値が必要とされる場合，いかなる関連鍵差分 $(\Delta P, \Delta C, (\Delta K_L, \Delta K_R))$ の確率も $2^{-128}$ を上回らない．なぜなら，$L_L$ と $L_R$ のすべてのビットは連続する少なくとも 6 段でラウンド鍵として利用されるからである．$(\Delta L_L, \Delta L_R)$ の一部の word が未知であるような $((\Delta K_L, \Delta L_R) \rightarrow (\Delta L_L, \Delta L_R))$ が利用されることもあるかもしれないが，このような distinguisher は 128 ビット鍵 CLEFIA の場合と同様に，有効でないと考えられる．従って，仕様通りの CLEFIA は関連鍵攻撃に対して強い耐性を有すると考える．

### 3.2.4 関連鍵 Boomerang 攻撃

関連鍵 Boomerang 攻撃のアイデアは，確率の低い，長い関連鍵差分パスではなく，確率の高い，短い 2 本の関連鍵差分パスを利用するというものである．

$n$ をブロックのビット長，$k$ を鍵のビット長とする．Boomerang 攻撃と同様に，CLEFIA の暗号化アルゴリズム $E : \{0,1\}^n \times \{0,1\}^k \to \{0,1\}^n$ を $E_0$ と $E_1$ のカスケード構造 $E = E_1 \circ E_0$ として表せるとし，$E_0$ に鍵差分 $\Delta K_0$ のもとで確率 $p$ の関連鍵差分パス $\alpha \to \beta$，$E_1$ に鍵差分 $\Delta K_1$ のもとで確率 $q$ の関連鍵差分パス $\gamma \to \delta$ が存在するとする．

関連鍵 Boomerang プロセスでは下記の 4 つの異なる未知の (しかし関連した) 鍵 $K_a, K_b, K_c, K_d$ を用いる．

$$
\begin{aligned}
& K_a \\
K_b &= K_a \oplus \Delta K_0 \\
K_c &= K_a \oplus \Delta K_1 \\
K_d &= K_a \oplus \Delta K_0 \oplus \Delta K_1
\end{aligned}
$$

攻撃は以下のプロセスで行う．

- 平文 $P_a$ をランダムに選び，$P_b = P_a \oplus \alpha$ を計算する．
- 鍵 $K_a$ のもとでの $P_a$ に対する暗号文 $C_a = E_{K_a}(P_a)$ と鍵 $K_b$ のもとでの $P_b$ に対する暗号文 $C_b = E_{K_b}(P_b)$ を求める．
- $C_c = C_a \oplus \delta$, $C_d = C_b \oplus \delta$ を計算する．
- 鍵 $K_c$ のもとでの $C_a$ に対する平文 $P_c = E^{-1}_{K_c}(C_c)$ と鍵 $K_d$ のもとでの $C_d$ に対する平文 $P_d = E^{-1}_{K_d}(C_d)$ を求める．
- $P_c \oplus P_d = \alpha$ を満たすかどうかをチェックする．

ランダム置換であれば，最後の条件が成り立つ確率は $2^{-n}$ である．しかしながら，$E$ では，この条件が成り立つ確率は，通常の Boomerang 攻撃と同様に $p^2q^2$ となる．従って，128 ビットブロック暗号 CLEFIA に関連鍵 Boomerang 攻撃を適用するには $(pq)^2 > 2^{-128}$，すなわち $pq > 2^{-64}$ が必要となる．

前節で述べたように，128 ビット鍵スケジュールにおいて $GFN_{4,12}$ を，192/256 ビット鍵スケジュールにおいて $GFN_{8,10}$ を採用しているため，$pq > 2^{-64}$ の条件が成立することは困難である．実際，$E_0$ および $E_1$ においてこの条件を満たす長い段数の経路は見つかっていない．従って，関連鍵 Boomerang 攻撃は CLEFIA の脅威にならないと考えられる．

### 3.2.5 関連鍵 Rectangle 攻撃

関連鍵 Boomerang 攻撃から関連鍵 Rectangle 攻撃への変換は，Boomerang 攻撃から Rectangle 攻撃への変換と同様である．

CLEFIA の暗号化アルゴリズム $E$ が前節と同様に分解できるとし，$\alpha$, $\delta$, $\hat{p}$, $\hat{q}$, $K_a$, $K_b$, $K_c$, $K_d$ も前節と同じ定義とする．このとき，関連鍵 Rectangle 識別攻撃は下記のようになる．

- $P_b = P_a \oplus \alpha$ なる $N$ 個の平文ペア $(P_a, P_b)$ をランダムに選び，鍵 $K_a$ のもとでの $P_a$ に対する暗号文 $C_a = E_{K_a}(P_a)$ と鍵 $K_b$ のもとでの $P_b$ に対する暗号文 $C_b = E_{K_b}(P_b)$ を求める．
- $P_d = P_c \oplus \alpha$ なる $N$ 個の平文ペア $(P_c, P_d)$ をランダムに選び，鍵 $K_c$ のもとでの $P_c$ に対する暗号文と鍵 $K_d$ のもとでの $P_d$ に対する暗号文を求める．
- 平文 $(P_a, P_b, P_c, P_d)$ と，対応する暗号文 $(C_a, C_b, C_c, C_d)$ に対し，$C_a \oplus C_c = C_b \oplus C_d = \delta$ を満たす四つ組 (quartet) を探す．

入力差分 $\alpha$ をとる $N$ 個の平文ペアを用意し，鍵 $K_a$, $K_b$ で暗号化すると，$N^2 2^{-n} (\hat{p}\hat{q})^2$ 個の正しい四つ組が得られると期待できる．

この攻撃を CLEFIA に適用するには $(\hat{p}\hat{q})^2 > 2^{-128}$ となることが必要であるが，この条件を満たす 2，3 段以上の $E_0$, $E_1$ は見つかっていない．よって CLEFIA はこの攻撃に対して十分な強度をもつと考えられる．

# 第4章　実装評価

本章では CLEFIA のソフトウェアおよびハードウェアにおける実装評価とサイドチャネル攻撃に対する安全性について述べる．

## 4.1　ソフトウェア実装評価

本節では CLEFIA のソフトウェア実装評価について述べる．CLEFIA のソフトウェア実装性能を C 言語およびアセンブリ言語を用いて評価した結果を以下に示す．評価に用いたプラットフォームは表 4.1 の通りである．

表 4.2 は C 言語での評価結果を示している．暗復号及び鍵セットアップのソフトウェア処理速度を rdtsc 命令を用いて計測した．

ソフトウェア実装におけるメモリ使用量を表 4.3 に示す．スタック使用量は，最近の Linux カーネルソースに含まれる checkstack.pl スクリプトを用いて objdump の出力から最大スタック使用量を計算することで評価した．

表 4.4 はプラットフォーム 4 におけるアセンブリ言語での評価結果を示している．一般的な単一ブロック実装と 2 ブロック並列実装 [45] の 2 種類の実装法について，暗復号および鍵セットアップのソフトウェア処理速度を計測した．単一ブロック実装では 12.9 cycles/byte (1.48 Gbps) を実現している．また，CTR モードや CBC モードの復号といった並列実行可能なモードに有効な 2 ブロック並列実装では，11.1 cycles/byte を達成している．

## 4.2　ハードウェア実装評価

本節では CLEFIA のハードウェア実装評価について述べる．128 ビット鍵の CLEFIA に関しては 2 種類，192 ビット鍵，256 ビット鍵の CLEFIA に関しては 1 種類のアーキテクチャで実装を行ない，それらの回路規模と速度性能を ASIC ライブラリを用いて評価した．

128 ビット鍵の CLEFIA に対して，ループ実装，小型化実装の 2 種類のアーキテクチャで実装を行った．ループ実装は 1 サイクルに 1 ラウン

表 4.1: 評価プラットフォーム

| プラットフォーム | プロセッサ | 周波数 [GHz] | OS | コンパイラ |
|---|---|---|---|---|
| 1 | AMD Opteron | 2.6 | Red Hat Enterprise Linux 3 | gcc 3.2.3 |
| 2 | Intel Core2 Duo | 2.4 | Windows Vista (32-bit) | Intel C++ Compiler 11.0 |
| 3 | Intel Core2 Duo | 2.4 | Windows Vista (64-bit) | Intel C++ Compiler 11.0 |
| 4 | AMD Athlon 64 4000+ | 2.4 | Windows XP (64-bit) | Microsoft Visual Studio 2005 |

表 4.2: CLEFIA ソフトウエア実装結果 (C 言語)

| プラットフォーム | 鍵長 [bit] | 暗号化 [cycles/byte] | 復号 [cycles/byte] | 鍵セットアップ (暗号化) [cycles] | 鍵セットアップ (復号) [cycles] |
|---|---|---|---|---|---|
| 1 | 128 | 17.7 | 18.0 | 442 | 517 |
| | 192 | 21.5 | 21.8 | 683 | 789 |
| | 256 | 25.2 | 25.6 | 734 | 859 |
| 2 | 128 | 18.7 | 19.7 | 304 | 385 |
| | 192 | 22.6 | 23.7 | 545 | 653 |
| | 256 | 26.4 | 28.0 | 590 | 722 |
| 3 | 128 | 17.6 | 18.5 | 325 | 446 |
| | 192 | 21.4 | 22.1 | 460 | 616 |
| | 256 | 25.0 | 25.8 | 493 | 683 |
| 4 | 128 | 19.0 | 19.1 | 386 | 452 |
| | 192 | 23.0 | 23.0 | 583 | 681 |
| | 256 | 26.8 | 27.0 | 627 | 720 |

ドの処理を行う実装手法であり，2 つの F 関数 $F_0$, $F_1$ は回路の共有化を行っていない．小型化実装においては，回路面積を削減するために F 関数 $F_0$, $F_1$ を共有化した $F_0/F_1$ 回路を用いている．$F_0/F_1$ 回路では 1 ラウンドの処理に 2 サイクルかかり，初めのサイクルで $F_0$ の処理，次のサイクルで $F_1$ の処理を行う．暗復号におけるループ実装，小型化実装の所要サイクル数は 18 および 36 となる．192 ビット，256 ビット鍵の CLEFIA についてはループ実装でのみ実装を行なった．暗復号における所要サイクル数は，192 ビット鍵の場合 22，256 ビット鍵の場合 26 となる．

我々が用いた実装環境は以下の通りである．

表 4.3: メモリ使用量

| コードサイズ [byte] | スタック使用量 [byte] |
|---|---|
| 17955 | 224 |

表 4.4: CLEFIA ソフトウエア実装結果 (アセンブリ言語)

| 実装法 | 鍵長 [bit] | 暗号化 [cycles/byte] | 復号 [cycles/byte] | 鍵セットアップ (暗号化) [cycles] | 鍵セットアップ (復号) [cycles] |
|---|---|---|---|---|---|
| 単一ブロック | 128 | 12.9 | 13.3 | 217 | 229 |
| | 192 | 15.8 | 16.2 | 272 | 293 |
| | 256 | 18.3 | 18.4 | 328 | 357 |
| 2 ブロック並列実装 | 128 | 11.1 | 11.1 | 217 | 229 |
| | 192 | 13.3 | 13.3 | 272 | 293 |
| | 256 | 15.6 | 15.6 | 328 | 357 |

| | |
|---|---|
| 記述言語 | Verilog-HDL |
| 設計ライブラリ | 0.09 $\mu$m CMOS ASIC library |
| シミュレータ | VCS version 2005.06 |
| 論理合成ツール | Design Compiler version 2006.06 |

1 ゲートは 2 入力 NAND ゲートに相当し，速度は最悪条件での評価を用いている．

表 4.5 に実装結果を示す．それぞれの実装に対して，規模優先または速度優先を指定することにより，2 種類の回路を合成した．上段が規模優先の結果，下段が速度優先の結果である．また，比較として AES および Camellia の既知の結果としては最良となる実装データを掲載した [61]．128 ビット鍵の CLEFIA は，ループ実装において 5,979 gate で 1.60 Gbps を，小型化実装において 4,950 gate で 0.71 Gbps を実現している．このとき，ゲート規模当りの速度は，それぞれ 268.63 Kbps/gate, 144.59 Kbps/gate となっており，AES の高速版および小型版実装と比較して 1.98 倍および 2.51 倍，Camellia の高速版および小型版実装と比較して 3.04 倍および 2.89 倍の実装効率を示している．これらの数値は，ASIC ライブラリの違いによる性能差を考慮しても十分なアドバンテージを持っており，CLEFIA が高いハードウェア実装性能を持ったブロック暗号であることを示している．

表 4.5: CLEFIA のハードウェア実装結果

| | 鍵長 [bits] | 暗復号 [cycles] | 鍵セットアップ [cycles] | ゲート規模 [gates] | 周波数 [MHz] | 速度 [Mbps] | 速度/ゲート規模 [Kbps/gate] |
|---|---|---|---|---|---|---|---|
| CLEFIA ($0.09\mu$m) | 128 | 18 | 12 | 5,979 | 225.83 | 1,605.94 | 268.63 |
| | | | | 12,009 | 422.29 | 3,003.00 | 250.06 |
| | | 36 | 24 | 4,950 | 201.28 | 715.69 | 144.59 |
| | | | | 9,377 | 389.55 | 1,385.10 | 147.71 |
| | 192 | 22 | 20 | 8,536 | 206.56 | 1,201.85 | 140.81 |
| | | | | 15,718 | 391.08 | 2,275.39 | 144.76 |
| | 256 | 26 | 20 | 8,482 | 206.56 | 1,016.95 | 119.89 |
| | | | | 15,542 | 391.08 | 1,925.33 | 123.88 |
| AES [61] ($0.13\mu$m) | 128 | 11 | N/A | 12,454 | 145.35 | 1,691.35 | 135.81 |
| | | | | 21,337 | 224.22 | 2,609.11 | 122.28 |
| | | 54 | N/A | 5,398 | 131.24 | 311.09 | 57.63 |
| | | | | 9,227 | 220.75 | 523.26 | 56.71 |
| Camellia [61] ($0.13\mu$m) | 128 | 22 | N/A | 10,993 | 166.94 | 971.29 | 88.36 |
| | | | | 16,905 | 256.41 | 1,491.84 | 88.25 |
| | | 44 | N/A | 6,511 | 111.98 | 325.76 | 50.03 |
| | | | | 12,231 | 238.10 | 692.65 | 56.63 |

各条件において, 上段が規模優先, 下段が速度優先の結果を示す.

## 4.3 サイドチャネル攻撃に対する安全性

本節では CLEFIA のサイドチャネル攻撃に対する安全性について述べる.

CLEFIA は AES と同じく 8 ビット S-box を持った SP 型の F 関数で構成されているため，AES [6,54] と同様のキャッシュ攻撃が CLEFIA に適用可能であることが報告されている [57, 60]．一方，ビットスライス実装等の AES におけるキャッシュ攻撃に対する対策法も CLEFIA に適用可能であると考えられる．また，AES [55] と同様のフォルト攻撃が CLEFIA に適用可能であることも報告されている [16, 75, 76].

電力解析 [39] や電磁波解析 [56] に関しては，2 線式ロジック [77] やマスキング・ロジック [74] 等の論理レベルの対策法が CLEFIA の実装に適用可能であると考える.

# 第5章　第三者評価

CLEFIA は 2007 年に外部の研究者によって評価を受けている．評価者は Alex Biryukov 教授，Vincent Rijmen 教授，Serge Vaudenay 教授，ABT の Lars R. Knudsen 教授と Bart Preneel 教授である．我々は 2007 年の 7 月と 9 月に評価レポートを受け取った．その評価レポートは CLEFIA のウェブサイト (http://www.sony.net/clefia) から取得できる [13,38,58,81]. レポートには，事前に評価者へ提供された CLEFIA の設計資料に基づき，CLEFIA のセキュリティの側面からの評価結果，及び追加コメントが記載されている．評価にあたり提供された設計資料もウェブサイトにて公開されており，仕様書 [68]，設計方針 [69]，及び自己評価書 [70] である．

また評価の際，評価者に対して，「ソニーの暗号アルゴリズムは現在の暗号解析技術に対して安全か?」をはじめとしたいくつかの質問をしている．この質問に対して，評価者からは肯定的な回答をもらっている．詳細については評価レポートを参照されたい．

# 参考文献


[1] R. Anderson, E. Biham, and L. R. Knudsen, "Serpent: A proposal for the advanced encryption standard." Primitive submitted to AES, 1998. Available at http://www.cs.technion.ac.il/~biham/Reports/Serpent/.

[2] K. Aoki, T. Ichikawa, M. Kanda, M. Matsui, S. Moriai, J. Nakajima, and T. Tokita, "Camellia: A 128-bit block cipher suitable for multiple platforms." 2000. Available at http://info.isl.ntt.co.jp/crypt/camellia/dl/support.pdf.

[3] K. Aoki, T. Ichikawa, M. Kanda, M. Matsui, S. Moriai, J. Nakajima, and T. Tokita, "Camellia: A 128-bit block cipher suitable for multiple platforms." in *Proceedings of Selected Areas in Cryptography – SAC '00* (D. R. Stinson and S. E. Tavares, eds.), no. 2012 in LNCS, pp. 41–54, Springer-Verlag, 2001.

[4] P. S. L. M. Barreto and V. Rijmen, "The Anubis block cipher." Primitive submitted to NESSIE, Sept. 2000. Available at http://www.cryptonessie.org/.

[5] P. S. L. M. Barreto and V. Rijmen, "The Whirlpool hashing function." Primitive submitted to NESSIE, Sept. 2000. Available at http://www.cryptonessie.org/.

[6] D. J. Bernstein, "Cache-timing Attacks on AES." 2005. Available at `http://cr.yp.to/antiforgery/cachetiming-20050414.pdf`.

[7] E. Biham, "New Types of Cryptanalytic Attacks Using Related Keys." *Journal of Cryptology*, vol. 7, no. 4, pp. 229–246, 1994.

[8] E. Biham, A. Biryukov, and A. Shamir, "Cryptanalysis of Skipjack reduced to 31 rounds using impossible differentials." in *Proceedings of Eurocrypt'99* (J. Stern, ed.), no. 1592 in LNCS, pp. 12–23, Springer-Verlag, 1999.

[9] E. Biham, O. Dunkelman, and N. Keller, "Related-Key Impossible Differential Attacks on 8-Round AES-192." in *Topics in Cryptology – CT-RSA 2006, The Cryptographers' Track* (D. Pointcheval, ed.), no. 3860 in LNCS, pp. 21–33, Springer-Verlag, 2006.

[10] E. Biham and A. Shamir, "Differential cryptanalysis of DES-like cryptosystems." *Journal of Cryptology*, vol. 4, pp. 3–72, 1991.

[11] E. Biham and A. Shamir, *Differential Cryptanalysis of the Data Encryption Standard*. Springer-Verlag, 1993.

[12] A. Biryukov and D. Wagner, "Slide attack." in *Proceedings of Fast Software Encryption – FSE'99* (L. R. Knudsen, ed.), no. 1636 in LNCS, pp. 245–259, Springer-Verlag, 1999.

[13] A. Biryukov, "Review of CLEFIA." 2007. Available at `http://www.sony.net/clefia`.

[14] A. Biryukov and D. Khovratovich, "Related-Key Cryptanalysis of the Full AES-192 and AES-256." in *Advances in Cryptology – ASIACRYPT 2009* (M. Matsui, ed.), no. 5912 in LNCS, pp. 1–18, Springer-Verlag, 2009.

[15] A. Biryukov, D. Khovratovich, and I. Nikolić, "Distinguisher and Related-Key Attack on the Full AES-256." in *Advances in Cryptology – CRYPTO 2009* (S. Halevi, ed.), no. 5677 in LNCS, pp. 231–249, Springer-Verlag, 2009.

[16] H. Chen, W. Wu, and D. Feng, "Differential Fault Analysis on CLEFIA." in *Proceedings of ICICS 2007*, no. 4861 in LNCS, pp. 284–295, Springer-Verlag, 2007.

[17] D. Coppersmith and S. Winograd, "Matrix multiplication via arithmetic progressions." *Journal of Symbolic Computation*, vol. 9, no. 3, pp. 251–280, 1990.

[18] N. Courtois and J. Pieprzyk, "Cryptanalysis of block ciphers with overdefined systems of equations." in *Proceedings of ASIACRYPT'02* (Y. Zheng, ed.), no. 2501 in LNCS, pp. 267–287, Springer-Verlag, 2002.

[19] J. Daemen, L. R. Knudsen, and V. Rijmen, "The block cipher SQUARE." in *Proceedings of Fast Software Encryption – FSE'97*

(E. Biham, ed.), no. 1267 in LNCS, pp. 149–165, Springer-Verlag, 1997.

[20] J. Daemen and V. Rijmen, "Statistics of Correlation and Differentials in Block Ciphers." in *IACR ePrint archive 2005/212*, 2005.

[21] J. Daemen and V. Rijmen, "Two-Round AES Differentials." in *IACR ePrint archive 2006/039*, 2006.

[22] J. Daemen and V. Rijmen, *The Design of Rijndael: AES – The Advanced Encryption Standard (Information Security and Cryptography)*. Springer, 2002.

[23] N. Ferguson, J. Kelsey, S. Lucks, B. Schneier, M. Stay, D. Wagner, and D. Whiting, "Improved Cryptanalysis of Rijndael." in *Proceedings of Fast Software Encryption – FSE 2000* (B. Schneier, ed.), vol. 1978 of *Lecture Notes in Computer Science*, pp. 213–230, Springer, 2001.

[24] H. Gilbert, H. Handshuh, A. Joux, and S. Vaudenay, "A stastical attack on RC6." in *Proceedings of Fast Software Encryption – FSE'00* (B. Schneier, ed.), no. 1978 in LNCS, pp. 64–74, Springer-Verlag, 2001.

[25] H. Gilbert and M. Minier, "A collision attack on 7 rounds of Rijndael." in *Proceedings of 3rd AES candidate conference*, pp. 230–241, 2001.

[26] S. Hong, S. Lee, J. Lim, J. Sung, D. H. Cheon, and I. Cho, "Provable security against differential and linear cryptanalysis for the SPN structure." in *Proceedings of Fast Software Encryption – FSE'00* (B. Schneier, ed.), no. 1978 in LNCS, pp. 273–283, Springer-Verlag, 2001.

[27] IPA and TAO, "CRYPTREC report 2002." 2003. Available at `http://www.ipa.go.jp/security/enc/CRYPTREC/fy15/doc/c02_2.pdf` (in Japanese) and `http://www2.nict.go.jp/y/y213/cryptrec_publicity/c02_report_english.pdf`(in English).

[28] T. Jakobsen and L. R. Knudsen, "The interpolation attack on block ciphers." in *Proceedings of Fast Software Encryption – FSE'97* (E. Biham, ed.), no. 1267 in LNCS, pp. 28–40, Springer-Verlag, 1997.

[29] P. Junod and S. Vaudenay, "FOX : A new family of block ciphers." in *Proceedings of Selected Areas in Cryptography – SAC'04* (H. Handschuh and M. A. Hasan, eds.), no. 3357 in LNCS, pp. 114–129, Springer-Verlag, 2004.

[30] M. Kanda, "Practical security evaluation against differential and linear cryptanalyses for Feistel ciphers with SPN round function." in *Proceedings of Selected Areas in Cryptography – SAC'00* (D. R. Stinson and S. E. Tavares, eds.), no. 2012 in LNCS, pp. 324–338, Springer-Verlag, 2001.

[31] M. Kanda, S. Moriai, K. Aoki, H. Ueda, Y. Takashima, K. Ohta, and T. Matsumoto, "E2 — A New 128-bit Block Cipher." *IEICE. Trans. Fundamentals, E83A*, no. 1, pp. 48–59, 2000.

[32] J. Kelsey, T. Kohno, and B. Schneier, "Amplified Boomerang Attacks Against Reduced-Round MARS and Serpent." in *Proceedings of Fast Software Encryption – FSE'00* (B. Schneier, ed.), no. 1978 in LNCS, pp. 75–93, Springer-Verlag, 2001.

[33] J. Kelsey, B. Schneier, and D. Wagner, "Related-key cryptanalysis of 3-Way, Biham-DES, CAST, DES-X, NewDES, RC2, and TEA." in *Proceedings of Information and Communication Security '97*, no. 1334 in LNCS, pp. 233–246, Springer-Verlag, 1997.

[34] J. Kim, S. Hong, J. Sung, C. Lee, and S. Lee, "Impossible differential cryptanalysis for block cipher structure." in *Proceedings of Indocrypt 2003* (T. Johansson and S. Maitra, eds.), no. 2904 in LNCS, pp. 82–96, Springer-Verlag, 2003.

[35] L. R. Knudsen and W. Meier, "Correlations in RC6 with a reduced number of rounds." in *Proceedings of Fast Software Encryption – FSE'00* (B. Schneier, ed.), no. 1978 in LNCS, pp. 94–108, Springer-Verlag, 2001.

[36] L. R. Knudsen and D. Wagner, "Integral cryptanalysis." in *Proceedings of Fast Software Encryption – FSE'02* (J. Daemen and V. Rijmen, eds.), no. 2365 in LNCS, pp. 112–127, Springer-Verlag, 2002.

[37] L. R. Knudsen, "Truncated and higher order differentials." in *Fast Software Encryption: Second International Workshop* (B. Preneel, ed.), no. 1008 in LNCS, pp. 196–211, Springer-Verlag, 1994.

[38] L. R. Knudsen and B. Preneel, "Evaluation of CLEFIA." 2007. Available at `http://www.sony.net/clefia`.

[39] P. Kocher, J. Jaffe, and B. Jun, "Differential Power Analysis." in *Proceedings of CRYPTO 99* (M. J. Wiener, ed.), no. 1666 in LNCS, pp. 388–397, Springer-Verlag, 1999.

[40] X. Lai, "Higher order derivatives and differential cryptanalysis." in *Proceedings of symposium on communication, coding and cryptography, in honor of J. L. Massey on the occation of his 60th birthday*, 1994.

[41] S. K. Langford and M. E. Hellman, "Differential-linear cryptanalysis." in *Proceedings of CRYPTO 94* (Y. Desmedt, ed.), no. 839 in LNCS, pp. 17–25, Springer-Verlag, 1994.

[42] C. Lim and K. Khoo, "An analysis of XSL applied to BES." in *Preproceedings of Fast Software Encryption – FSE'07* (A. Biryukov, ed.), pp. 253–265, 2007.

[43] M. Matsui, "Linear cryptanalysis of the data encryption standard." in *Proceedings of Eurocrypt'93* (T. Helleseth, ed.), no. 765 in LNCS, pp. 386–397, Springer-Verlag, 1994.

[44] M. Matsui, "On correlation between the order of s-boxes and the strength of DES." in *Proceedings of Eurocrypt'94* (A. D. Santis, ed.), no. 950 in LNCS, pp. 366–375, Springer-Verlag, 1995.

[45] M. Matsui, "How far can we go on the x64 processors?." in *Proceedings of Fast Software Encryption – FSE'06* (M. Robshaw, ed.), no. 4047 in LNCS, pp. 341–358, Springer-Verlag, 2006.

[46] M. Matsui and T. Tokita, "Cryptanalysis of reduced version of the block cipher E2." in *Proceedings of Fast Software Encryption – FSE'99* (L. R. Knudsen, ed.), no. 1636 in LNCS, pp. 71–80, Springer-Verlag, 1999.

[47] M. Matusi, "New block encryption algorithm MISTY." in *Proceedings of Fast Software Encryption – FSE'97* (E. Biham, ed.), no. 1267 in LNCS, pp. 54–68, Springer-Verlag, 1997.

[48] S. Moriai, M. Sugita, K. Aoki, and M. Kanda, "Security of E2 against truncated differential cryptanalysis." in *Proceedings of Selected Areas in Cryptography – SAC'99* (H. M. Heys and C. M.

Adams, eds.), no. 1758 in LNCS, pp. 106–117, Springer-Verlag, 2000.

[49] S. Moriai and S. Vaudenay, "On the pseudorandomness of top-level schemes of block ciphers." in *Proceedings of ASIACRYPT'00* (T. Okamoto, ed.), no. 1976 in LNCS, pp. 289–302, Springer-Verlag, 2000.

[50] S. Murphy and M. Robshaw, "Essential algebraic structure within the AES." in *Proceedings of CRYPTO 2002* (M. Yung, ed.), no. 2442 in LNCS, pp. 1–16, Springer-Verlag, 2002.

[51] S. Murphy and M. Robshaw, "Comments on the security of the AES and the XSL technique." *Electronic Letters*, vol. 39, no. 1, pp. 36–38, 2003.

[52] K. Nyberg, "Generalized Feistel network." in *Proceedings of ASIACRYPT'96* (K. Kim and T. Matsumoto, eds.), no. 1163 in LNCS, pp. 91–104, Springer-Verlag, 1996.

[53] K. Ohkuma, H. Muratani, F. Sano, and S. Kawamura, "The block cipher Hierocrypt." in *Proceedings of Selected Areas in Cryptography – SAC'00* (D. R. Stinson and S. E. Tavares, eds.), no. 2012 in LNCS, pp. 72–88, Springer-Verlag, 2001.

[54] D. A. Osvik, A. Shamir, and E. Tromer, "Cache attacks and countermeasures: The case of aes." in *Proceedings of CT-RSA 2006* (D. Pointcheval, ed.), no. 2860 in LNCS, pp. 1–20, Springer-Verlag, 2006.

[55] G. Piret and J.-J. Quisquater, "A Differential Fault Attack Technique against SPN Structure, with Application to the AES and KHAZAD." in *Proceedings of CHES 2003*, no. 2779 in LNCS, pp. 77–88, Springer-Verlag, 2003.

[56] J. J. Quisquater and D. Samyde, "ElectroMagnetic Analysis (EMA):Measures and Countermeasures for Smart Cards." in *Proceedings of E-smart 2001*, 2001.

[57] C. Rebeiro, D. Mukhopadhyay, J. Takahashi, and T. Fukunaga, "Cache Timing Attacks on Clefia." in *Proceedings of Indocrypt 2009*, no. 5922 in LNCS, pp. 104–118, Springer-Verlag, 2009.

[58] V. Rijmen, "Evaluation of CLEFIA." 2007. Available at `http://www.sony.net/clefia`.

[59] R. L. Rivest, M. J. B. Robshaw, R. Sidney, and Y. L. Yin, "The RC6 block cipher." Primitive submitted to AES, 1998. Available at http://www.rsasecurity.com/.

[60] C. Robeiro and D. Mukhopadhyay, "Differnce Cache Trace Attack against CLEFIA." in *IACR ePrint archive 2010/012*, 2010.

[61] A. Satoh and S. Morioka, "Hardware-focused performance comparison for the standard block ciphers AES,Camellia, and Triple-DES." in *Proceedings of ISC 2003* (C. Boyd and W. Mao, eds.), no. 2851 in LNCS, pp. 252–266, Springer-Verlag, 2003.

[62] B. Schneier, J. Kelsey, D. Whiting, D. Wagner, C. Hall, and N. Ferguson, "Twofish: A 128-bit block cipher." Primitive submitted to AES, 1998. Available at http://www.schneier.com/.

[63] T. Shirai and B. Preneel, "On Feistel ciphers using optimal diffusion mappings across multiple rounds." in *Proceedings of ASIACRYPT'04* (P. J. Lee, ed.), no. 3329 in LNCS, pp. 1–15, Springer-Verlag, 2004.

[64] T. Shirai and K. Shibutani, "Improving immunity of Feistel ciphers against differential cryptanalysis by using multiple MDS matrices." in *Proceedings of Fast Software Encryption – FSE'04* (B. Roy and W. Meier, eds.), no. 3017 in LNCS, pp. 260–278, Springer-Verlag, 2004.

[65] T. Shirai and K. Shibutani, "On Feistel structures using a diffusion switching mechanism." in *Proceedings of Fast Software Encryption – FSE'06* (M. Robshaw, ed.), no. 4047 in LNCS, pp. 41–56, Springer-Verlag, 2006.

[66] T. Shirai, K. Shibutani, T. Akishita, S. Moriai, and T. Iwata, "The 128-bit blockcipher CLEFIA." in *Proceedings of Fast Software Encryption – FSE'07* (A. Biryukov, ed.), no. 4593 in LNCS, pp. 181–195, Springer-Verlag, 2007.

[67] T. Shirai and K. Araki, "On generalized Feistel structures using the diffusion switching mechanism." *IEICE. Trans. Fundamentals*, vol. E91A, no. 8, pp. 2120–2129, 2008.

[68] Sony Corporation, “The 128-bit Blockcipher CLEFIA : Algorithm Specification, Rev 1.0.” 2007. Available at http://www.sony.net/clefia.

[69] Sony Corporation, “The 128-bit Blockcipher CLEFIA : Design Rationale, Rev 1.0.” 2007. Available at http://www.sony.net/clefia.

[70] Sony Corporation, “The 128-bit Blockcipher CLEFIA : Security and Performance Evaluations.” 2007. Available at http://www.sony.net/clefia.

[71] Sony Corporation, “The 128-bit Blockcipher CLEFIA: Specification, Version 1.0 (Japanese/English).” 2010. Submission to CRYPTREC (Application for Cryptographic Techniques towards the Revision of the e-Government Recommended Ciphers List).

[72] M. Sugita, K. Kobara, and H. Imai, “Security of reduced version of the block cipher Camellia against truncated and impossible differential cryptanalysis.” in *Proceedings of ASIACRYPT'01* (C. Boyd, ed.), no. 2248 in LNCS, pp. 193–207, Springer-Verlag, 2001.

[73] B. Sun, R. Li, M. Wang, P. Li, and C. Li, “Impossible Differential Cryptanalysis of CLEFIA.” in *IACR ePrint archive 2008/151*, 2008.

[74] D. Suzuki, M. Saeki, and T. Ichikawa, “Random Switching Logic: A Countermeasure against DPA based on Transition Probability.” in *IACR ePrint archive 2004/236*, 2004.

[75] J. Takahashi and T. Fukunaga, “Differential Fault Analysis on CLEFIA.” in *Proceedings of Symposium on Cryptography and Information Security 2009 –SCIS 2009 2A3-4, (in Japanese)*, 2009.

[76] J. Takahashi and T. Fukunaga, “Improved Differential Fault Analysis on CLEFIA.” in *Proceedings of FDTC 2008*, LNCS, pp. 25–34, IEEE Computer Society, 2008.

[77] K. Tiri and I. Verbauwhede, “A Logic Level Design Methodology for A Secure DPA Resistant ASIC or FPGA Implementation.” in *Proceedings of DATE 2004*, pp. 246–251, 2004.

[78] E. Tsujihara, M. Shigeri, T. Suzaki, T. Kawabata, and Y. Tsunoo, “New Impossible Differentials of CLEFIA.” in *IEICE Technical Report – ISEC2008-3 (in Japanese)*, 2008.

[79] Y. Tsunoo, E. Tsujihara, M. Shigeri, T. Saito, T. Suzaki, and H. Kubo, "Impossible Differential Cryptanalysis of CLEFIA." in *Proceedings of Fast Software Encryption – FSE 2008* (K. Nyberg, ed.), no. 5086 in LNCS, pp. 398–411, Springer-Verlag, 2008.

[80] S. Vaudenay, "An experimental on DES statistical cryptanalysis." in *3rd ACM conference on computer and communications security*, pp. 139–147, ACM Press, 1996.

[81] S. Vaudenay, "Evaluation report on CLEFIA." 2007. Available at `http://www.sony.net/clefia`.

[82] D. Wagner, "The boomerang attack." in *Proceedings of Fast Software Encryption – FSE'99* (L. R. Knudsen, ed.), no. 1636 in LNCS, pp. 156–170, Springer-Verlag, 1999.

[83] W. Wang and X. Wang, "Improved Impossible Differential Cryptanalysis of CLEFIA." in *IACR ePrint archive 2007/466*, 2007.

[84] H. Wu, "Related-Cipher Attacks." in *Proceedings of Information and Communications Security – ICICS 2002* (R. H. Deng, S. Qing, F. Bao, and J. Zhou, eds.), no. 2513 in LNCS, pp. 447–455, Springer-Verlag, 2002.

[85] W. Zhang. Private Communication, 2008.

[86] W. Zhang and J. Han, "Impossible Differential Analysis of Reduced Round CLEFIA." in *Pre-Proceedings of Inscrypt 2008* (M. Yung, P. Liu, and D. Lin, eds.), pp. 143–154, 2008.

[87] W. Zhang and J. Han, "Impossible Differential Analysis of Reduced Round CLEFIA." in *Proceedings of Inscrypt 2008* (M. Yung, P. Liu, and D. Lin, eds.), no. 5487 in LNCS, pp. 181–191, Springer-Verlag, 2009.

[88] Y. Zheng, T. Matsumoto, and H. Imai, "On the Construction of Block Ciphers Provably Secure and Not Relying on Any Unproved Hypotheses." in *Proceedings of CRYPTO 89* (G. Brassard, ed.), no. 435 in LNCS, pp. 461–480, Springer-Verlag, 1989.

## 著作権について