混合整数計画法を用いた暗号の安全性評価

田中　秀磨

ブロック暗号の選択平文攻撃手法の一つに Integral 攻撃がある．この攻撃では，暗号化処理における Integral Property と呼ばれる特性の伝搬を解析する必要がある．しかしながら伝搬結果の場合分けが複雑であり，安全性評価の観点からは計算機実験に基づく予測しかできなかった．近年，藤堂によって再定義された Division Property を利用することで，Integral 攻撃に対する安全性評価が明確になり，さらに混合背整数計画法 (MILP) を適用することで高効率な評価が実行できるようになった．本稿では，このような Integral 攻撃に対する評価手法の変遷について解説し，MILP が実現した画期的な成果について紹介する．

キーワード：共通鍵暗号，選択平文攻撃，Integral 攻撃，Division Property，MILP

1. はじめに

現在，インターネットや身の回りの IoT デバイスで利用されている暗号技術は，鍵の性質から公開鍵暗号と共通鍵暗号の 2 種類に分類される．公開鍵暗号は安全性を数学的難問に帰着させることで実現しており，素因数分解問題や離散対数問題などが多く採用されている．近年では量子計算機および量子アルゴリズムの実利用が現実味を帯び，これらに対しても安全なように，格子問題のような量子計算機でも有効な解法が発見されていない問題に帰着できる耐量子暗号が注目されている．本稿で対象とするのは共通鍵暗号であり，これはさらにストリーム暗号とブロック暗号に分類される．ストリーム暗号は擬似乱数生成器から出力される乱数列と平文を EX-OR して暗号文を生成する．鍵は擬似乱数生成器の初期値である．したがって，乱数列から初期値を推定できるか否かが安全性評価の基本方針となる．通信上生じたエラーが伝搬しないことから，無線通信やストリーミングで好んで用いられる．

本稿で対象とするのはブロック暗号である．ブロック暗号は平文を 64 ビットや 128 ビットに区切り，これをひとまとめにして暗号化処理を行う．国内外問わず大量のブロック暗号が国際標準化もしくは商用利用されている．たとえば，2010 年頃までは 64 ビットブロック暗号で 4 種類，128 ビットブロック暗号で 5 種類がわが国の電子政府用途で推奨されていた．なお 2014 年に実利用実績なども加味したうえで 2 種類の 128 ビットブロック暗号のみに限定している．高速に大量のデータを扱うことに向いていることからデータの暗号化処理に用いられる．構成法は多岐にわたるが，基本的にはラウンド関数（もしくは F 関数）と呼ばれる，単純な非線形関数を繰り返す構造をもつ．図 1 にブロック暗号全体の概要を示す（$\mathbb{F}_2^n$ は標数 2 の拡大体）．繰り返し回数をラウンド数（もしくは段数）と呼び，この数で安全性と実装性を調整する．F 関数はさらに S-Box (Substitution-Box) と呼ばれる非線形関数と，ビットもしくはバイト単位での置き換えで構成される．S-Box は多くの場合，4 ビットもしくは 8 ビット入出力（サブブロック単位での処理）である．これに加えて EX-OR，AND，分岐 (Copy)，分割 (Split)，結合 (Concatenation) という単純な処理の組み合わせでブロック暗

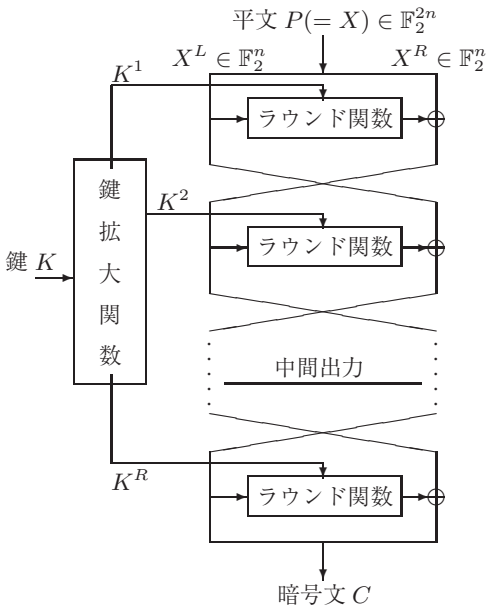

図 1　R ラウンド構成の Feistel 構造をもつブロック暗号の概要図 ($X = X^L||X^R$)

たなか　ひでま
防衛大学校情報工学科
〒 239–8686 神奈川県横須賀市走水 1–10–20

Copyright © by ORSJ. Unauthorized reproduction of this article is prohibited.

号全体が構成される．それ以外にもビット切り落とし (Truncation) や OR なども利用される場合がある．F 関数の繰り返し方にもいくつかの方法があり，大きく Feistel 構造と SPN (Substitution Permutation Network) 構造に分類される．鍵は鍵拡大関数に入力され，ラウンド鍵（段鍵）が分配される．

ブロック暗号の安全性評価では，攻撃者は暗号の仕様を知っており，さらに平文とそれに対応する暗号文を入手できると仮定したうえで秘密情報である鍵の推定が可能であるかを判断する．一見，すでに平文と暗号文を知っているので，これ以上何をするのかわけのわからない条件に思えるが，スパイ小説的な暗号解読のように暗号文のみの攻撃になると，たとえば $X+Y=1213$ という式が与えられたうえで整数 X と Y を特定せよ，という問題のように解を特定できない．また，たとえば交通系 IC カードでは，日時，乗車区間，料金，残金などの情報が平文と暗号文の両方で記録されているので，実利用的な観点からも「平文とそれに対応する暗号文を入手できる」という仮定は非現実的ではない．鍵の推定に関しても，ユーザが直接利用する鍵の推定を意味しているのではなく，ラウンド鍵の，それもその 1 ビットだけでも 1/2 よりも有意な確率で推定できれば攻撃者の勝ち（この暗号はブレイクされた）と判断される．これは，鍵のサイズが保証するエントロピーをその処理で保証できなかったとみなされるからである．逆に言えば，このような厳しい条件であっても攻撃できないのであれば，将来にわたって安全に利用できる見込みがある．

2. ブロック暗号に対する安全性評価の概要

ブロック暗号は前節で示したようにラウンド関数を複数回繰り返し構成している．ブロック暗号の安全性評価では，この途中ラウンドにおける出力（中間出力）に注目する．この中間出力がそれまでに用いられたラウンド鍵の値によらず成立する特性には，確率的に成立するものと確定的に成立するものがある．確率的に成立する特性を利用した攻撃手法の代表として差分解読法や不能差分攻撃法がある．本稿で扱うのは確定的に成立する特性を利用する攻撃手法であり，代数的解読法として分類され，SQUARE 攻撃，高階差分攻撃 [1] や Integral 攻撃 [2] がある．

以下，説明を簡単にするために 64 ビットブロック暗号を例として Integral 攻撃の概要を述べる．r ラウンド目のラウンド関数に対する入力を $X^r=(x_0^r, x_1^r, \ldots, x_{63}^r)$，出力を $Y^r=(y_0^r, y_1^r, \ldots, y_{63}^r)$，ラウンド鍵を $K^r=(k_0^r, k_1^r, \ldots, k_b^r)$ とする．一般的な Feistel 構造ではラウンド関数が 32 ビット構成のため $b=31$，SPN 構造では 64 ビット構成のため $b=63$ となるが，同一ラウンド関数で複数種類の鍵を用いる暗号もある．なお，1 段目の入力 X^1 は平文 P を意味し，同様に最終段出力は暗号文 C である．ここでは表記を簡潔にするため，以下では 1 段目入力および平文を単に X と表記する．また，繰り返し構造であるので $Y^r=X^{r+1}$ となる．安全性評価においては，暗号化処理中に鍵が変更されないことを前提とするので K^r はランダムに選ばれた定数として扱う．したがって任意の出力ビット y_m^r は 64 ビット変数をもつ関数として表現できる．

$$y_m^r = f_m^r(X^r; K^1, \ldots, K^r) \tag{1}$$

これを Algebraic Normal Form (ANF) で表現すると，y_m^r は $K^1, \ldots, K^r$ を係数とする X の関数として扱うことができる．広義には暗号化関数は random function として定義されているが，このように鍵によってランダムに関数を選択しているとみなすことができる．このとき，AND 項に含まれる X の変数の数を代数次数と呼ぶ．たとえば $k_0k_2k_{18}x_2x_5x_{53}$ の X に関する代数次数は 3 次である．ここで X において変数として扱うビット（変数ビット）と定数とするビット（定数ビット）を意図的に選べば，y_m^r の代数次数をコントロールすることができる．たとえば $k_0k_2k_{18}x_2x_5x_{53}$ において，x_2 のみ変数ビットとすれば事実上線形項として扱うことができる．このように X において意図的に変数ビットと定数ビットを決定して中間出力をコントロールする手法を選択平文攻撃と呼び，変数ビットによって取りうるすべての平文の集合を選択平文集合と呼ぶ．

鍵と変数ビットを決定し，変数ビットが取りうるすべての入力値に対する出力を算出したとする．たとえば 64 ビット入力のうち，下位 2 ビット（x_0^r と x_1^r）のみを変数ビットとすれば，$(0,0,\ldots,0,0)$, $(0,0,\ldots,0,1)$, $(0,0,\ldots,1,0)$, $(0,0,\ldots,1,1)$ の 4 通りの入力（選択平文集合の要素）が考えられ，それらに対する出力を算出する．Y^r において y_m^r を任意に M 個選んで構成した部分集合を $\mathbb{Y}_M^r$ とすれば，X に対する M ビット値の集合として表現できる．たとえば 64 ビット出力のうち，r 段目出力の上位 2 ビット（y_{63}^r と y_{62}^r）に注目して集合 $\mathbb{Y}_2^r$ を構成すれば $(0,0)$, $(0,1)$, $(1,0)$, $(1,1)$ の 4 通りの要素による集合として考えられる．この集合の特性は以下の四つに分類される．

$\mathcal{A}$ (all) M ビットの値が重複なくすべて含まれる

$\mathcal{B}$ (balance) 集合に含まれる要素すべての EX-

 Copyright © by ORSJ. Unauthorized reproduction of this article is prohibited.

OR がゼロになる

$\mathcal{C}$ **(constant)**　すべて同じ値のみの集合になっている

$\mathcal{U}$ **(unknown)**　上記以外

前述の集合 $\mathbb{Y}_2^r$ を例として挙げると，たとえば以下のような状態である．

$\mathcal{A}$**:**　$(0,0)$, $(0,1)$, $(1,0)$, $(1,1)$

$\mathcal{B}$**:**　$(0,1)$, $(0,1)$, $(1,1)$, $(1,1)$

$\mathcal{C}$**:**　$(1,0)$, $(1,0)$, $(1,0)$, $(1,0)$

$\mathcal{U}$**:**　$(0,0)$, $(1,0)$, $(1,0)$, $(1,0)$

これが Wagner と Knudsen によって定義された Integral Property である [2].

Integral 値の算出は以下のように定義される．

$$ {}^{i}y_m^r = \bigoplus_{x\in\mathbb{X}} f_m^r(x; K^1, \ldots, K^r) \tag{2} $$

ただし $K^1, \ldots, K^r$ はランダムに選ばれた定数としてのラウンド鍵，$\mathbb{X}$ は選択平文集合を表す．${}^{i}y_m^r$ は 0 か 1 の値をとり，部分集合 $\mathbb{Y}_M^r$ に対する Integral 値は M ビットとなる．このとき，部分集合の Integral Property が $\mathcal{A}$, $\mathcal{B}$, $\mathcal{C}$ のいずれかであれば $\mathbb{Y}_M^r$ の Integral 値はゼロであり，$\mathcal{U}$ であれば非ゼロの値となることは明らかである．部分集合 $\mathbb{Y}_M^r$ の Integral 値は暗号文側から算出しても同じ値となるので，ラウンド鍵の値を予測しながら Integral 値を算出し，それがゼロになったとき，暗号文側で使用したラウンド鍵の推定に成功したことになる．このような攻撃手法が Integral 攻撃である．選択平文集合 $\mathbb{X}$ と途中出力の部分集合 $\mathbb{Y}_M^r$ の選び方で Integral Property が決定されるので，なるべく多段のラウンド関数を経たうえで Integral Property が成立するような，これらの選び方を探索することが重要な課題となる．

一方で，ANF の代数次数に注目すれば同様の特性になるか否かを解析することができる．前述の例に挙げた $k_0k_2k_{18}x_2x_5x_{53}$ が最大次数項である ANF の場合，(x_2, x_5, x_{53}) を変数ビットに選び Integral 値を計算すれば $k_0k_2k_{18}$ になる（3 次の Integral.2^3 で 8 通りの入力に対して 8 通りの出力を算出し EX-OR をとる）．したがって適当な四つ目の変数（たとえば x_0）を選んで Integral 値を計算すればゼロになることは明らかである．（4 次の Integral.2^4 通りの入力に対して算出することになるが x_0 に対応する項が存在しないので 3 次の Integral が 2 セット得られることになる．結果的に同じ 3 次の Integral 値を EX-OR することになり，値はゼロになる．）これは偏微分を計算していることと同じ意味合いであり，ANF における最大次数よりも一つ大きい数の変数ビットを選べば Integral 値がゼロになることが保証される．このような考え方に基づく攻撃手法は高階差分攻撃と呼ばれる．なお高階差分攻撃では式 (2) は Integral 値ではなく，高階差分値として定義されている．Integral 攻撃と高階差分攻撃は厳密な定義では異なるものであるが，実際の計算と注目している特性は同じである．ただし特性の探索において前者が Integral Property に注目し，後者が ANF の代数次数に注目している点で方針が異なる．

3. Integral Property と Division Property

ある選択平文集合を仮定し，r ラウンド目の出力において Integral Property が成立するか否かを探索するには，出力の Integral Property が次のラウンドでどのように変化するか（Integral Property の伝搬）を解析する手法と，ラウンド処理ごとに ANF の変数と代数次数がどのように増加していくかを解析する手法とがある．前者は伝搬探索，後者は代数次数見積もりと呼ばれる．

ブロック暗号全体を見れば平文と暗号文は 1 対 1 に関連していなければ復号できないので，全単射となっている．したがって 64 ビットブロック暗号の場合，64 次の Integral Property は必ず $\mathcal{A}$ である．しかしながら，このことは 2^{64} のすべての平文入力を意味するので安全性評価の観点からは意味をなさない（全平文暗号文組を算出するのでそもそも攻撃の意味がない）．そこで 63 次以下での特性を考えなければならない．ブロック暗号は，前述のようにサブブロック単位の非線形関数（ANF で表現すると AND が含まれる非線形部分）とビット単位の置き換え（permutation，線形部分）を繰り返す構造的な特徴をもつ．このとき，非線形関数だけが Integral Property を変化させる要因となる（代数次数の見積もり視点からも明らかである）．たとえば AES で用いられている S-Box は 8 ビット入出力の全単射な非線形関数であり，すべての出力ビットの ANF の代数次数は 7 である．この場合，8 ビット変数の $\mathcal{A}$ の入力を与えれば出力も $\mathcal{A}$ となり，$\mathcal{B}$ の入力を与えれば $\mathcal{U}$ となることが知られている．このような伝搬ルールを利用することから，入力 X を S-Box の入出力サイズと同じサイズに区切り（サブブロック分割），それぞれのサブブロックに対し $\mathcal{A}$ もしくは $\mathcal{C}$ の特性を与えた選択平文を考える．このとき $\mathcal{A}$ が与えられたサブブロックが変数ビットの位置，$\mathcal{C}$ が与えられたサブブロックが定数ビットの位置にそれぞれ対応する．ラウンドごとの処理において，それぞれのサブブロック

Copyright © by ORSJ. Unauthorized reproduction of this article is prohibited.

の Integral Property の変化を追跡し，すべてのサブブロックが$\mathcal{U}$となった時点で探索を終了する．このような考え方に基づく手法が伝搬解析である．非常に単純な考え方であるが，特性の取り方が複数の場合が存在する．たとえば EX-OR において，$\mathcal{A}$ と $\mathcal{C}$ が入力されれば出力は必ず $\mathcal{A}$ となるが，$\mathcal{A}$ と $\mathcal{A}$ や $\mathcal{A}$ と $\mathcal{B}$ の場合は $\mathcal{A}$, $\mathcal{B}$, $\mathcal{C}$ のいずれかの特性を取りうる．攻撃に都合よく解釈すれば $\mathcal{A}$ か $\mathcal{C}$ である．このように特性を選べば，次のラウンドで処理される非線形関数において伝搬が維持され，出力は再び $\mathcal{A}$ か $\mathcal{C}$ となる．結果的に Integral Property が成立するラウンド数を大きくできる．一方で $\mathcal{B}$ とした場合は出力の特性は $\mathcal{U}$ となり伝搬はそこで終わる．このような特性の伝搬をビタビアルゴリズムを用いるなどして効率的に探索する方法や，$\mathcal{A}$, $\mathcal{B}$, $\mathcal{C}$ 以外にも特性を定義した拡大解析手法などが考案された [3, 4]．しかしながら実際の計算機による確認なしでは評価結果はあくまでも予測であり，現実的な計算機環境を考えると 48 次を超える Integral property については不確かである．

代数次数の見積もりは，出力の ANF を次のラウンド入力として代入することを繰り返して導出する．この手法はサブブロック単位で Integral property を扱う必要がないので，自由に変数ビットの位置を決定できる自由さをもつ．したがって Integral property の探索範囲が伝搬解析よりも広いという長所がある．しかしながら，たとえば N 次の Integral Property を解析する場合，64 ビットブロック暗号であれば ${}_{64}\mathrm{C}_N$ 通りの変数ビットの組み合わせが存在するので，実際に実行すると計算機能力に対する要求が伝搬解析よりも大きくなる欠点がある．そのため，次数が高い項のみに着目して ANF を限定的に処理する方法や，EX-OR は 1 階差分と解釈できる（出力の代数次数を一つ減じる）ことも考慮した形式的な代数次数見積もり手法などが考案されたが，これも限定的な 48 次程度の解析が限界である [5, 6]．なお，代数次数の見積もりは元々は高階差分攻撃に対する特性探索手法として発展してきた経緯がある．

このような既存の探索手法を根本から変えたのが藤堂によって提案された Division Property である [7]．本人による解説が CRYPTREC（電子政府推奨暗号技術評価委員会）の技術報告書として公開されており，第 3.1 節に着想に至る動機が記されている [8]．ここでは簡単に述べると，

・伝搬解析では特性の変化が非線形関数にしか発生しないため，事実上，線形部分しか活用できていない．
・代数次数の見積もりでは線形部分が効果的に活用されず，非線形部分しか注目していない．

という問題点を同時に解決することを目指したものである．簡単にその特徴をまとめると，$\mathcal{A}$ と $\mathcal{B}$ の性質を分けて考えないで有用な性質を見逃さないような処理方法を導入したことにある．厳密な定義は文献 [7] に譲るとして，本稿に必要な定義だけを以下に文献 [7] より抜粋する．

定義 1. Bit Product Function π_u. Let $\pi_u : \mathbb{F}_2^n \to \mathbb{F}_2$ be a function for any $u \in \mathbb{F}_2^n$. Let $x \in \mathbb{F}_2^n$ be an input of π_u, and $\pi_u(x)$ is the AND of x_i satisfying $u_i = 1$, i.e., it is defined as

$$\pi_u(x) := \prod_{i=1}^{n} x_i^{u_i} \tag{3}$$

これを用いると ANF は以下のように表現できることが示されている．

定義 2. Algebraic Normal Form (ANF). Any $f : \mathbb{F}_2^n \to \mathbb{F}_2$ can be represented as

$$f(x) = \bigoplus_{u \in \mathbb{F}_2^n} a_u^f (\prod_{i=1}^{n} x_i^{u_i}) = \bigoplus_{u \in \mathbb{F}_2^n} a_u^f \pi_u(x), \tag{4}$$

where $a_u^f \in \mathbb{F}_2$ is a constant value depending on f and u.

これらより，Division Property は以下のように定義されている．

定義 3. Division Property. Let $\mathbb{X}$ be a multiset whose elements take a value of $\mathbb{F}_2^n$, and k takes a value between 0 and n. When the multiset $\mathbb{X}$ has the division property $\mathcal{D}_k^n$, it fulfills the following conditions:

$$\bigoplus_{x \in \mathbb{X}} \pi_u(x) = \begin{cases} \text{Unknown} & \text{if } w(u) \geq k, \\ 0 & \text{otherwise,} \end{cases} \tag{5}$$

where $w(\cdot)$ denotes Hamming weight.

$\mathcal{D}_k^N$ の場合，u は all-zero から all-one まで 2^N 通りの値がある．すべての $\mathbb{X}$ の要素に対して u が含まれるときは 1，含まれないときは 0 とする．ちなみに $u =$

 Copyright © by ORSJ. Unauthorized reproduction of this article is prohibited.

all-zero はすべての $\mathbb{X}$ に対して必ず含まれるとする．ある u に対し，各要素から返される 0/1 の EX-OR の総和が 0 となるとき，「その u は $w(u) < k$ となる $\mathcal{D}_k^N$ が成立する」と判定する．このようにして，$\mathcal{D}_k^N$ が成立する u とそうでない場合で u の集合を分割していく．なお $\mathcal{D}_1^N$ は $\mathcal{U}$ の，$\mathcal{D}_2^N$ は $\mathcal{B}$ の Integral property と同様の性質となる．一方で $\mathcal{A}$ を満足する集合は $\mathcal{D}_N^N$ を満足するが，その逆は必ずしも成立しない．例として挙げられているのが，各要素が奇数個で構成されている場合であり，$\mathcal{A}$ という特性にはならないが $\mathcal{D}_N^N$ として扱うことができる．

このような $\mathcal{D}_k^N$ の表記に基づき，ラウンドを経た Division Property の伝搬解析を行う．探索の開始条件は前述の伝搬解析と同様に $\mathcal{A}$ と $\mathcal{C}$ をサブブロック単位で与えた選択平文集合を用いる．線形部分と非線形部分に対しそれぞれ伝搬特性のルールが示されている．例として非線形部分の伝搬ルールを抜粋する [7].

非線形部分の Division Property の伝搬特性. Let s be a function (S-Box) from n bits to m bits, and the degree is d. Assuming that an input multiset $\mathbb{X}$ has the division property $\mathcal{D}_k^n$, the output multiset $\mathbb{Y}$ has $\mathcal{D}^m_{\lceil \frac{k}{d} \rceil}$. In addition, assuming that $n = m$ and the S-Box is a permutation, the output multiset $\mathbb{Y}$ has $\mathcal{D}_n^n$ when the input multiset has $\mathcal{D}_n^n$.

実際には Vectorial Division Property や Collective Division Property など，さらなる応用発展手法を駆使することになるが，本稿では省略する．線形部分の伝搬特性は以下の 4 種類が定義されている．ルールの記述については紙面の都合上，省略する．

- Copy $[y_1, y_2] = [x, x]$
- Split $y_1 || y_2 = x$
- Concatenation $y = x_1 || x_2$
- EX-OR $y = x_1 \oplus x_2$

このように $\mathcal{A}$ と $\mathcal{B}$ を分けず，しかも $\mathcal{D}_k^N$ の成立のみに着目するので従来手法では探索できない特性の発見を可能にしている．Division Property が不定になるような u の選び方は最初は限定的であるが，ラウンドを経るごとに増えていき最終的には $u =$ all-zero 以外がすべて不定になる．その時点で探索は終了となる．

Division Property を用いた安全性評価で特筆すべき点は 64 ビットブロック暗号 MISTY1 に対する安全性評価である [9]．MISTY1 は 1995 年に三菱電機によって開発された 64 ビットブロック暗号であり，線形解読法および差分解読法に対して証明可能な安全性を有する特徴をもつ．従来手法では 4 ラウンドの高階差分特性しか発見できなかったが，6 ラウンドの Divion Property の成立を明らかにし 8 ラウンド全体に対して攻撃を成功させた．MISTY1 はわが国の電子政府推奨暗号の一つであり，3G 携帯電話で標準採用されているブロック暗号 KASUMI の元になったものである．このように，ある種の標準化されたものでブレイクされた暗号としては，米国政府標準暗号であった DES に次いで 2 例目と考えられる．また，現在の米国政府標準暗号である 128 ビットブロック暗号 AES に対しては，4 ラウンドにおいて従来手法では発見できない特性を示すことに成功している [8].

4. Division Property 探索の弱点と混合整数計画法

Division Property $\mathcal{D}_k^N$ において $k(= w(u))$ は $1 \le k \le N$ の間を取る．$\mathcal{D}_k^N$ は k 未満の u に対し成立か不成立を判定していくので，直感的には探索の for ループ回数の上限が k で決められていると考えてよい．もし r ラウンド目の探索において，すべての k の u に対して Division Property が不成立の場合，その次のラウンドにおける探索においても不成立になる見込みが高い．for ループの回数は計算コストに直接関わるので無駄な探索は削除するのが望ましい．そのため，r ラウンド目で成立する $k'(< k)$ が決定できた場合，$(r+1)$ ラウンド目では for ループの上限を $k = k'$ と置き換える．厳密な定義は文献 [7] に譲るとして，このように探索範囲を狭め効率性を向上させる SizeReduce と呼ばれる技術が追加されている．SizeReduce の考え方は直感的には合理的に思えるものの，Division Property の着想方針とは矛盾する印象を与える．というのは，r ラウンド目で成立しないからといって，それ以降でも成立しないとは必ずしも言えないからである．実際に久保は，後で紹介する軽量ブロック暗号 Piccolo の Division Property 探索において，高階差分特性探索によって得られた結果よりも下回る場合があることを示している [10]．久保はこの事実を「SizeReduce による特性の劣化」と名づけている．また，Division Property 自体は従来の Integral Property 探索よりもはるかに計算機能力に対する要求が低いものの，一般に利用可能な計算機環境としては高性能なものが必要である．その結果，実行できる探索に対する制限という形で欠点が現れ，サブブロック単位での Division Property 探索は有効であっても，ビット単位で実行しようとす

Copyright © by ORSJ. Unauthorized reproduction of this article is prohibited.

るとブロック暗号の入出力サイズが 32 ビットよりも大きい場合は不可能である．これは，前述した代数次数の見積もりにおける欠点と同様の状況と言える．

このような弱点を解決する手段として，Xiang によって Division Property 探索を混合整数計画法 (MILP: Mixed Integer Linear Programming) の問題として扱い解決する手法が提案された [11]．具体的には，ブロック暗号を構成する基本演算である

・Copy

・AND

・EX-OR

における Division Property の伝搬特性について制約式を導出し，

・非線形関数 (S-Box)

・MDS (Maximum Distance Separable) 行列

は，それぞれ ANF 表現などにしたうえで上述の制約式を適用し導出するものである．注意すべき点は，通常の演算の制約式ではなく，あくまでも Division Property の伝搬特性に関するものという点である．たとえば EX-OR の場合，$(a_0, a_1) \xrightarrow{\text{EX-OR}} b$（ただし $a_0, a_1, b \in \{0,1\}$）という関係において，$(0,0) \to 0$, $(0,1) \to 1$, $(1,0) \to 1$ という入出力関係は通常の EX-OR と同じであるが，$(1,1)$ の場合の b は Division Property が成立しないので探索終了を意味する．結果的に EX-OR の制約式は

$$a_0 + a_1 - b = 0 \tag{6}$$

という極めて単純な式となる．

S-Box の制約式導出には ANF を用い，SageMath における inequality_generator() などを応用することで制約式を簡単に得ることができる．ただしそのまま適用すると式の数が大きくなる傾向にあることや，Division Property 探索では不要な制約式まで導出されることから，Sun によって効率的に導出する Greedy Algorithm と名づけられた手法が提案されている [12]．

MDS 行列は拡大体上で定義された行列であり，サブブロック単位で拡散させる非線形部分である．最初に binary のプリミティブ表現に変換したのち，非ゼロの要素に独立した変数を与えたうえで Copy と EX-OR に対応する制約式を導出する．以下に佐藤が導出した軽量ブロック暗号 Piccolo の MDS 行列の制約式導出を例として示す [13]．Piccolo の MDS 行列の処理は

$$\begin{aligned} &{}^t\big(x_{0(4)}, x_{1(4)}, x_{2(4)}, x_{3(4)}\big) \\ &\quad \leftarrow M\,{}^t\big(x_{0(4)}, x_{1(4)}, x_{2(4)}, x_{3(4)}\big) \end{aligned} \tag{7}$$

と表記され，$\mathbb{F}_2^4$ 上で定義された特性多項式 $x^4 + x + 1$ に基づいて計算される．

$$M = \begin{pmatrix} 2 & 3 & 1 & 1 \\ 1 & 2 & 3 & 1 \\ 1 & 1 & 2 & 3 \\ 3 & 1 & 1 & 2 \end{pmatrix} \tag{8}$$

これのプリミティブ表現は以下となる．

$$\left(\begin{array}{cccc|cccc|cccc|cccc} 0&1&0&0&1&1&0&0&1&0&0&0&1&0&0&0 \\ 0&0&1&0&0&1&1&0&0&1&0&0&0&1&0&0 \\ 1&0&0&1&1&0&1&1&0&0&1&0&0&0&1&0 \\ 1&0&0&0&1&0&0&1&0&0&0&1&0&0&0&1 \\ \hline 1&0&0&0&0&1&0&0&1&1&0&0&1&0&0&0 \\ 0&1&0&0&0&0&1&0&0&1&1&0&0&1&0&0 \\ 0&0&1&0&1&0&0&1&1&0&1&1&0&0&1&0 \\ 0&0&0&1&1&0&0&0&1&0&0&1&0&0&0&1 \\ \hline 1&0&0&0&1&0&0&0&0&1&0&0&1&1&0&0 \\ 0&1&0&0&0&1&0&0&0&0&1&0&0&1&1&0 \\ 0&0&1&0&0&0&1&0&1&0&0&1&1&0&1&1 \\ 0&0&0&1&0&0&0&1&1&0&0&0&1&0&0&1 \\ \hline 1&1&0&0&1&0&0&0&1&0&0&0&0&1&0&0 \\ 0&1&1&0&0&1&0&0&0&1&0&0&0&0&1&0 \\ 1&0&1&1&0&0&1&0&0&0&1&0&1&0&0&1 \\ 1&0&0&1&0&0&0&1&0&0&0&1&1&0&0&0 \end{array}\right) \tag{9}$$

このうち非ゼロの要素は 88 個あるので，変数 $(t_0 \sim t_{87})$ を用いて置き換える．

$$\left(\begin{array}{cccc|cccc|cccc|cccc} 0&t_7&0&0&t_{22}&t_{29}&0&0&t_{44}&0&0&0&t_{66}&0&0&0 \\ 0&0&t_{12}&0&0&t_{30}&t_{34}&0&0&t_{51}&0&0&0&t_{73}&0&0 \\ t_0&0&0&t_{17}&t_{23}&0&t_{35}&t_{39}&0&0&t_{56}&0&0&0&t_{78}&0 \\ t_1&0&0&0&t_{24}&0&0&t_{40}&0&0&0&t_{61}&0&0&0&t_{83} \\ \hline t_2&0&0&0&0&t_{31}&0&0&t_{45}&t_{52}&0&0&t_{67}&0&0&0 \\ 0&t_8&0&0&0&0&t_{36}&0&0&t_{53}&t_{57}&0&0&t_{74}&0&0 \\ 0&0&t_{13}&0&t_{25}&0&0&t_{41}&t_{46}&0&t_{58}&t_{62}&0&0&t_{79}&0 \\ 0&0&0&t_{18}&t_{26}&0&0&0&t_{47}&0&0&t_{63}&0&0&0&t_{84} \\ \hline t_3&0&0&0&t_{27}&0&0&0&0&t_{54}&0&0&t_{68}&t_{75}&0&0 \\ 0&t_9&0&0&0&t_{32}&0&0&0&0&t_{59}&0&0&t_{76}&t_{80}&0 \\ 0&0&t_{14}&0&0&0&t_{37}&0&t_{48}&0&0&t_{64}&t_{69}&0&t_{81}&t_{85} \\ 0&0&0&t_{19}&0&0&0&t_{42}&t_{49}&0&0&0&t_{70}&0&0&t_{86} \\ \hline t_4&t_{10}&0&0&t_{28}&0&0&0&t_{50}&0&0&0&0&t_{77}&0&0 \\ 0&t_{11}&t_{15}&0&0&t_{33}&0&0&0&t_{55}&0&0&0&0&t_{82}&0 \\ t_5&0&t_{16}&t_{20}&0&0&t_{38}&0&0&0&t_{60}&0&t_{71}&0&0&t_{87} \\ t_6&0&0&t_{21}&0&0&0&t_{43}&0&0&0&t_{65}&t_{72}&0&0&0 \end{array}\right) \tag{10}$$

この結果，Copy に関する制約式として以下が導出される．

$$\begin{cases} x_0 - t_0 - t_1 - t_2 - t_3 - t_4 - t_5 - t_6 = 0 \\ x_1 - t_7 - t_8 - t_9 - t_{10} - t_{11} = 0 \\ x_2 - t_{12} - t_{13} - t_{14} - t_{15} - t_{16} = 0 \\ x_3 - t_{17} - t_{18} - t_{19} - t_{20} - t_{21} = 0 \\ x_4 - t_{22} - t_{23} - t_{24} - t_{25} - t_{26} - t_{27} - t_{28} = 0 \\ x_5 - t_{29} - t_{30} - t_{31} - t_{32} - t_{33} = 0 \\ x_6 - t_{34} - t_{35} - t_{36} - t_{37} - t_{38} = 0 \\ x_7 - t_{39} - t_{40} - t_{41} - t_{42} - t_{43} = 0 \\ x_8 - t_{44} - t_{45} - t_{46} - t_{47} - t_{48} - t_{49} - t_{50} = 0 \\ x_9 - t_{51} - t_{52} - t_{53} - t_{54} - t_{55} = 0 \\ x_{10} - t_{56} - t_{57} - t_{58} - t_{59} - t_{60} = 0 \\ x_{11} - t_{61} - t_{62} - t_{63} - t_{64} - t_{65} = 0 \\ x_{12} - t_{66} - t_{67} - t_{68} - t_{69} - t_{70} - t_{71} - t_{72} = 0 \\ x_{13} - t_{73} - t_{74} - t_{75} - t_{76} - t_{77} = 0 \\ x_{14} - t_{78} - t_{79} - t_{80} - t_{81} - t_{82} = 0 \\ x_{15} - t_{83} - t_{84} - t_{85} - t_{86} - t_{87} = 0 \\ x_0,\ x_1,\ \ldots,\ x_{15},\ t_0,\ t_1, \ldots, t_{87} \in \mathbb{F}_2 \end{cases} \tag{11}$$

 Copyright © by ORSJ. Unauthorized reproduction of this article is prohibited.

表 1　軽量ブロック暗号 Piccolo の Division Property 評価

次数	既存結果		MILP [13]	
	代数次数見積もり [15]	藤堂の探索手法 [10]	ラウンド	探索時間 [s]
8	4	4	4	34.73
12	4	4	4	110.70
24	5	4	5	10.68
32	6	5	6	26.40
48	7	6	6	86894.30
63	—	6	7	1186.53

同様に EX-OR に関する制約式として以下が導出される．

$$\begin{cases} t_7+t_{22}+t_{29}+t_{44}+t_{66}-y_0=0 \\ t_{12}+t_{30}+t_{34}+t_{51}+t_{73}-y_1=0 \\ t_0+t_{17}+t_{23}+t_{35}+t_{39}+t_{56}+t_{78}-y_2=0 \\ t_1+t_{24}+t_{40}+t_{61}+t_{83}-y_3=0 \\ t_2+t_{31}+t_{45}+t_{52}+t_{67}-y_4=0 \\ t_8+t_{36}+t_{53}+t_{57}+t_{74}-y_5=0 \\ t_{13}+t_{25}+t_{41}+t_{46}+t_{58}+t_{62}+t_{79}-y_6=0 \\ t_{18}+t_{26}+t_{47}+t_{63}+t_{84}-y_7=0 \\ t_3+t_{27}+t_{54}+t_{68}+t_{75}-y_8=0 \\ t_9+t_{32}+t_{59}+t_{76}+t_{80}-y_9=0 \\ t_{14}+t_{37}+t_{48}+t_{64}+t_{69}+t_{81}+t_{85}-y_{10}=0 \\ t_{19}+t_{42}+t_{49}+t_{70}+t_{86}-y_{11}=0 \\ t_4-t_{10}-t_{28}-t_{50}-t_{77}-y_{12}=0 \\ t_{11}+t_{15}+t_{33}+t_{55}+t_{82}-y_{13}=0 \\ t_5+t_{16}+t_{20}+t_{38}+t_{60}+t_{71}+t_{87}-y_{14}=0 \\ t_6+t_{21}+t_{43}+t_{65}+t_{72}-y_{15}=0 \\ y_0,\ y_1,\ \ldots,\ y_{15},\ t_0,\ t_1,\ldots,t_{87}\in\mathbb{F}_2 \end{cases} \tag{12}$$

導出した制約式に基づき Division Property 探索を開始するが，その過程で Gurobi など optimizer が適用可能であり，さらに効率的な探索が実行できる．これらのプログラムはオープン化され GitHub から入手できる [14]．現時点では MDS 行列の制約式導出以外はツール化されており，Integral 攻撃や Division Property に関する数学的知識がなくても ANF さえ記述できれば安全性評価が実行できる状況にある．

5. MILP による Division Property 探索の効果

2017 年頃までに文献 [11, 12] に示されるように，主要なブロック暗号の Division Property の特性評価は MILP によって更新された．その多くは Integral Property や藤堂の探索手法よりも効率的に高次の特性を明らかにしている．言うまでもなく，代数次数見積もりとそれに基づく評価よりも効率的である．ここでは主要なブロック暗号の評価の更新を述べるのではなく，これらの既存の評価手法と MILP の差が明確になった，佐藤による軽量ブロック暗号 Piccolo の評価を例に挙げて効果について述べる [13]．Piccolo は 2011 年に SONY の渋谷によって提案された 64 ビットブロック暗号であり，小回路規模で高速実行可能であることが特徴である．鍵長が 80 ビットと 128 ビットをサポートし，それぞれの総ラウンド数は 25 ラウンド，31 ラウンドである．

表 1 に Piccolo に対する芝山による高階差分特性評価（代数次数見積もり）[15]，久保による藤堂の探索手法を適用した Division Property 評価 [10]，佐藤による MILP を適用した Division Property 評価を示す．表 1 では次数に対する Division Property が成立する最大ラウンド数を示している．高階差分特性評価と藤堂の探索手法における計算機環境や探索時間は不明であるが，経験的に数秒から 4, 5 日程度ではないかと予測される．MILP による計算機環境は Intel Corei7-6700 (3.4 GHz)，8 GB メモリの一般的な PC であり，GitHub から入手した評価ツールをそのまま使用し，Gurobi による最適化はオンラインサービスを利用している．MILP が正確な評価を行える詳細な解析結果については文献 [13] に譲り，ここでは三つの大きなポイントのみ述べる．

一つ目は，24 次および 32 次の評価である．これは久保が指摘した SizeReduce による特性劣化であり，代数次数見積もりよりも藤堂の探索手法の結果が劣る．これらに関しては代数次数見積もりのほうが正しいことが計算機実験により確認されている．翻って MILP の場合は，SizeReduce を実行しないのでそのような特性劣化が生じず，藤堂の探索手法よりも正確に評価が実行できていることがわかる．

二つ目は，48 次の評価である．代数次数見積もりは 32 次の結果からの予測であり，藤堂の探索手法の結果より疑問が生じた．しかしながら前述のように SizeReduce による特性劣化があることから，明確にならなかった問題である．MILP の結果では 7 ラウンドで Division Property は成立しない．佐藤の評価では，念のために実行された計算機実験による検証が示されている．この検証には Intel Xeon CPU E5-2620 (24core,

Copyright © by ORSJ. Unauthorized reproduction of this article is prohibited.

2 GHz)，64 GB メモリの計算機を 1 台，Intel Xeon CPU E5-2430v2 (24core, 2.5 GHz)，128 GB メモリの計算機を 3 台の計 4 台による並列的な処理で，約 1 週間の時間を要した．その結果，48 次は 6 ラウンドで成立することを明らかにした．なお，同じ 6 ラウンドでの Division Property は 32 次でも成立する．この場合，次数の低い Division Property のほうが攻撃にも有利であるので，6 ラウンド 48 次の評価結果は安全性評価においては特に意味を生じない．高階差分特性は偏微分と同じ意味合いであると上述したが，この意味で考えると 6 ラウンド目出力の ANF の代数次数は最大 31 次であるということがわかる（32 次の高階差分値はゼロになる）．したがって 33 次以上での高階差分値もゼロになるのは明らかなので冗長な評価である．翻って暗号アルゴリズム的には，代数次数を上昇させる技術として 5, 6 ラウンド目に欠点があることを示唆している．

最後に 63 次の評価である．代数次数見積もりは現実的な計算機能力では実行不能であり，評価には言及がない．一方で Division Property による評価は可能であり，代数的特性評価における優位性は確固たるものがある．しかしながら前述のように SizeReduce による特性劣化があり，MILP を適用することのほうがより正確に評価できることが明確になった．

上記のように，MILP を適用することで従来よりも正確な代数特性の評価が実行できるようになった．また，これまでの評価からわからなかった代数次数を上昇させるための線形部分の役割が明確になり，今後のブロック暗号設計においても重要な役割を果たすことが期待できる．一方で，MILP による効率的な探索とはいえビット単位での Division Property 特性探索における欠点がないわけではない．MILP では制約式の成立／不成立を判定するだけなので不成立の場合の探索時間は非常に早いが，成立する制約式が多い場合は，予想外に長期の探索時間を要する．これは 12 次と 48 次の探索において顕著である．また，試行している 54〜56 次の探索でも 2 カ月以上かかるという報告もあり，さらなる効率性追求の余地があると考えられる．

6. おわりに

本稿では数多あるブロック暗号の安全性評価手法の中で，特に OR 的な視点が顕著であり，かつ最新の話題である Integral 攻撃に対する評価手法の変遷について紹介し，効果的な手法として MILP が実現した画期的な成果について述べた．ブロック暗号の評価では，ラウンド関数や S-Box といった構成要素単位で特性を評価し，全体に波及させる手法をとる．最終的には入力としての平文の条件と途中出力の特性から安全性評価を見積もるが，これは有利な特性をうまく結合させる経路の探索とみなせる．実際に暗号研究者はこの探索のことを「Path 探索」とか「Trail 探索」と呼んでおり，極論を言ってしまえば「どのような特性がありうるか？」と「どのような Path がありえるか？」という二つの視点で研究を行っている．本稿で紹介した Division Property は，Integral Property の Path 探索を有利にするため，特性の再定義と効率的な Division Trail 探索の両方をひとまとめに実現した手法とも言える．また，MILP はこの Division Trail 探索を線形計画問題に帰着させ，さらに探索効率を高めたという解釈も成り立つ．途中出力の特性の発見には線形近似をはじめ差分特性や代数特性など数学的傾向が強く，新しいブロック暗号はすぐに対策されてしまうため新しい特性の発見が難しい．一方で後者は OR 的な視点の応用により，既存攻撃手法に対してもさらなる発展と攻撃効果の向上の可能性があると考えられる．

参考文献

[1] L. Knudsen, “Truncated and higher order differentials,” *Fast Software Encryption, 1994*, Lecture Notes in Computer Science, **1008**, pp. 196–211, 2005.

[2] L. Knudsen and D. Wagner, “Integral cryptanalysis,” *Fast Software Encryption, 2002*, Lecture Notes in Computer Science, **2365**, pp. 112–127, 2002.

[3] H. Kosuge and H. Tanaka, “Algebraic degree estimation for integral attack by randomized algorithm,” *Information Security Applications, 2016*, Lecture Notes in Computer Science, **10144**, pp. 292–304, 2016.

[4] H. Kosuge, K. Iwai, H. Tanaka and T. Kurokawa, “Search algorithm of precise integral distinguisher of byte-based block cipher,” *Information Systems Security, 2015*, Lecture Notes in Computer Science, **9478**, pp. 303–323, 2015.

[5] Y. Hatano, H. Tanaka and T. Kaneko, “An optimized algebraic method for higher order differential attack,” *Applied Algebra, Algebraic Algorithms and Error-Correcting Codes*, Lecture Notes in Computer Science, **2643**, pp. 61–70, 2003.

[6] H. Tanaka, K. Hisamatsu and T. Kaneko, “Strength of MISTY1 without FL function for higher order differential attack,” *Applied Algebra, Algebraic Algorithms and Error-Correcting Codes, 1999*, Lecture Notes in Computer Science, **1719**, pp. 221–230, 1999.

[7] Y. Todo, “Structural evaluation by generalized integral property,” *Advances in Cryptology–EUROCRYPT, 2015*, Lecture Notes in Computer Science, **9056**, pp. 287–314, 2015.

[8] 藤堂洋介，“Integral 攻撃の最新動向と MISTY1 等への適用，” CRYPTREC 外部評価報告書，https://www.cryptrec.go.jp/exreport/cryptrec-ex-2501-2015.pdf

 Copyright © by ORSJ. Unauthorized reproduction of this article is prohibited.

（2019 年 7 月 25 日閲覧）
[9] Y. Todo,“Integral cryptanalysis on full MISTY1,” *CRYPTO, 2015*, Lecture Notes in Computer Science, **9215**, pp. 413–432, 2015.
[10] 久保卓也，五十嵐保隆，金子敏信，“軽量ブロック暗号 Piccolo の Division Property 及び高階差分特性の比較・検証，”信学技報，**115**, pp. 153–158, 2013.
[11] Z. Xiang, W. Zhang, Z. Bao and D. Lin, “Applying MILP method to searching integral distinguishers based on division property for 6 lightweight block ciphers,” *Advances in Cryptology, 2016*, Lecture Notes in Computer Science, **10031**, pp. 648–678, 2016.
[12] L. Sun, W. Wang and M. Wang, “MILP-aided bit-based division property for primitives with non-bit-permutation linear layers,” IACR Cryptology ePrint Archive, No. 811, 2016. https://eprint.iacr.org/2016/811.pdf
[13] H. Sato, M. Mimura and H. Tanaka, “Analysis of division property using MILP method for lightweight blockcipher Piccolo,” In *Proceedings of The 14th Asia Joint Conference on Information Security*, 2019, to be appeared.
[14] Z. Xiang, “MILPtool,” Open Public MILP Tool, https://github.com/xiangzejun/MILP_Division_Property （2019 年 8 月 21 日閲覧）
[15] 芝山直喜，金子敏信，“Piccolo の新しい高階差分特性，”信学技報，**114**, pp. 247–252, 2014.

Copyright © by ORSJ. Unauthorized reproduction of this article is prohibited.