Integral Cryptanalysis Using Algebraic Transition Matrices

Tim Beyne and Michiel Verbauwhede

COSIC, KU Leuven, Leuven, Belgium
firstname.lastname@esat.kuleuven.be

Abstract. In this work we introduce algebraic transition matrices as the basis for a new approach to integral cryptanalysis that unifies monomial trails (Hu et al., Asiacrypt 2020) and parity sets (Boura and Canteaut, Crypto 2016). Algebraic transition matrices allow for the computation of the algebraic normal form of a primitive based on the algebraic normal forms of its components by means of well-understood operations from linear algebra. The theory of algebraic transition matrices leads to better insight into the relation between integral properties of F and F^{-1}. In addition, we show that the link between invariants and eigenvectors of correlation matrices (Beyne, Asiacrypt 2018) carries over to algebraic transition matrices. Finally, algebraic transition matrices suggest a generalized definition of integral properties that subsumes previous notions such as extended division properties (Lambin, Derbez and Fouque, DCC 2020). On the practical side, a new algorithm is described to search for these generalized properties and applied to Present, resulting in new properties. The algorithm can be instantiated with any existing automated search method for integral cryptanalysis.

Keywords: Integral Cryptanalysis, Division Property, Nonlinear Invariants, ANF, Change-of-Basis, Algebraic Transition Matrices

1 Introduction

Integral cryptanalysis is an important technique for attacking symmetric-key primitives. It was originally described in 2002 by Knudsen and Wagner [KW02] as a generalization of a dedicated attack on the block cipher Square [DKR97] that exploits the byte-wise structure of Square to construct a zero-sum distinguisher. In 2015, Todo [Tod15] introduced the division property, which refines integral cryptanalysis by taking into account the algebraic degree of the components of the analyzed primitive. Except for the limited bit-pattern based integral attacks by Z'aba et al. [ZRHD08], integral cryptanalysis was mostly useful for word-oriented ciphers. This changed with the introduction of the bit-based division property by Todo and Morii [TM16], which made it possible to apply integral cryptanalysis to arbitrary ciphers. Also in 2016, Boura and Canteaut introduced parity sets as a different view of the division property [BC16], connecting the propagation of the division property in a bitwise manner with the algebraic normal form of the components of the primitive.

The term *integral cryptanalysis* refers to computing an 'integral' over a primitive, which can be interpreted as summing its outputs over a carefully chosen input set. Cube attacks [Vie07, DS09] and higher-order differentials [Knu95] could also be considered as integral cryptanalysis under this broad, albeit vague, description. This idea of integrals as generic sums is also reflected in further work on the division property, such as the three-subset division property without unknown subset by Hao et al. that can be used to reconstruct cubes [HLM+21] and the linearly equivalent S-boxes method of Lambin et al. that can describe higher-order differential [LDF20].

The division property is intimately linked with the algebraic normal form (ANF). Indeed, the ANF determines the propagation of parity sets, and ANF-coefficients can be computed using the three-subset division property without unknown subset. Hebborn et al. [HLLT20] have also shown that, in the absence of a key, parity set propagation can exactly compute arbitrary algebraic normal form coefficients from the parity of the number of trails. Conversely, any monomial that does not occur in the ANF of a primitive leads to an integral distinguisher. This is exploited by Hu et al. [HSWW20] in their work on monomial prediction, in which they reformulate division property algorithms as methods to detect the presence of specific monomials in the algebraic normal form. They also developed their own perfect propagation method based on backward propagation of monomials through primitives.

Contributions We introduce a new framework for integral cryptanalysis that aims to simplify the propagation of integral properties by relying on linear algebra. Inspired by the correlation-matrix description of linear cryptanalysis [DGV95] and the quasidifferential transition matrices that were proposed at Crypto 2022 [BR22], we introduce *algebraic transition matrices*. Algebraic transition matrices have similar properties to correlation matrices. In particular, the algebraic transition matrix of a composition of functions is the product of their corresponding algebraic transition matrices. Similar to linear trails, these products can be decomposed into a sum of *algebraic trails* to compute a single element of the algebraic transition matrix. Algebraic trails can also be interpreted as a generalization of division trails or monomial trails that can carry additional information on key-dependency.

Algebraic transition matrices and algebraic trails lead to several new theoretical insights. We investigate the relation between algebraic transition matrices and existing approaches to integral cryptanalysis such as the division property, parity sets and monomial trails. In particular, algebraic transition matrices unify the notions of parity sets and monomials trails and make the duality that exists between them precise. The link between the division properties of a permutation and those of its inverse that was first described by Udovenko at Asiacrypt 2021 [Udo21] is shown to have a natural description in terms of algebraic transition matrices and is generalized to arbitrary functions. In addition, we show that the link between invariants and eigenvectors of correlation matrices, introduced by Beyne at Asiacrypt 2018 [Bey18], carries over to algebraic transition matrices. As algebraic transition matrices are not defined over an algebraically closed field, they are not necessarily diagonalizable. However, we overcome this issue and are able to classify invariants precisely by combining the primary decomposition and generalized Jordan decomposition of algebraic transition matrices.

Our approach also leads to computational improvements. Based on the algebraic trail decomposition, and its specific form for key-alternating ciphers in particular, we obtain a more precise algorithm to compute division properties. Furthermore, algebraic transition matrices naturally lead to a broad definition of integral properties that further generalizes the extended integral properties of [LDF20]. A new algorithm is introduced to efficiently search for such integral properties. For the case of extended integral properties based on linear equivalences, it improves over the methods from [LDF20] as it is able to recover the same properties without enumerating linear equivalences. As a proof of concept, we apply the algorithm to 9-round Present, resulting in new properties that could not be found with previous methods.

Organisation The necessary notations and background for this work are discussed in Section 2. In Section 3, algebraic transition matrices are introduced and their properties are given. In Section 4, applications of algebraic transition matrices, such as their relation to parity sets and monomial trails and invariants, are developed. Section 5 describes the new algorithm to search for generalized integral properties.

2 Preliminaries

2.1 Naming and Notation

Vectors The vector space $\mathbb{F}_2^n$ has the following partial ordering: $x \leq y \Leftrightarrow \forall i : x_i \leq y_i$ with $x = (x_1, \ldots, x_n)$ and $y = (y_1, \ldots, y_n)$. The Hamming weight $\mathrm{wt}(x)$ of a vector $x \in \mathbb{F}_2^n$ is defined as the number of non-zero coordinates of that vector. The concatenation of two vectors $x \in \mathbb{F}_2^n$ and $y \in \mathbb{F}_2^m$ is denoted as $x\|y \in \mathbb{F}_2^{n+m}$. Notable vectors are $\underline{0} \in \mathbb{F}_2^n$ and $\underline{1} \in \mathbb{F}_2^n$, which respectively indicate the all-zero and all-one vector. The unit vectors $e_i \in \mathbb{F}_2^n$ are zero in all coordinates, except the i^{th}. Furthermore, $\overline{x}$ denotes the complement of x: $x + \underline{1}$. The precursor set of a vector $u \in \mathbb{F}_2^n$ consists of all vectors that are dominated by u, $\mathrm{Prec}(u) = \{x : x \leq u\}$. Similarly, the successor set of a vector u consists of all vectors that dominate u, $\mathrm{Succ}(u) = \{x : u \leq x\}$.

Boolean Functions This work considers functions operating on $\mathbb{F}_2^n$. For a field $\mathbb{K}$, we write these functions as $f : \mathbb{F}_2^n \to \mathbb{K}$ or equivalently $f \in \mathbb{K}^{\mathbb{F}_2^n}$. The second notation is used to emphasize that f is considered as a vector. When the field is $\mathbb{F}_2$, they are called Boolean functions. The vector space of Boolean functions on $\mathbb{F}_2^n$ can be identified with the polynomial ring $\mathbb{F}_2[x_1, \ldots, x_n]/(x_1^2 - x_1, \ldots, x_n^2 - x_n)$. Specifically, every Boolean function can be represented as a unique polynomial in this ring, which is called its algebraic normal form (ANF):

$$f(x) = \sum_{u \in \mathbb{F}_2^n} \lambda_u x^u \,, \tag{1}$$

where $x^u = \prod_{i=1}^n x_i^{u_i}$ for $x = (x_1, \ldots, x_n)$ and $u = (u_1, \ldots, u_n)$. The Kronecker delta function δ_x is zero everywhere except at x, where it equals one. Other notable functions are the indicator function $\mathbb{1}_X$ of a set $X \subseteq \mathbb{F}_2^n$ and the constant functions, 0 and 1. The tensor product, $\otimes$, of two functions, $f \in \mathbb{K}^{\mathbb{F}_2^n}$ and $g \in \mathbb{K}^{\mathbb{F}_2^m}$, is defined as $(f \otimes g)(x, y) = f(x)g(y)$. The tensor product of two matrices $A \in \mathbb{K}^{\mathbb{F}_2^n \times \mathbb{F}_2^m}$ and $B \in \mathbb{K}^{\mathbb{F}_2^s \times \mathbb{F}_2^t}$ is defined as $(A \otimes B)_{v\|v', u\|u'} = A_{v,u} B_{v',u'}$. Finally, vectorial Boolean functions are denoted with a capital letter: $F : \mathbb{F}_2^n \to \mathbb{F}_2^m$.

2.2 Linear Cryptanalysis

Even though this work is not about linear cryptanalysis, it is worth discussing it briefly as analogous concepts will be introduced. Linear cryptanalysis is based on the correlation between Boolean functions and linear functions, which can be exploited as a distinguishing feature of a cipher. We follow the interpretation by Beyne [Bey21].

2.2.1 Transition Matrices

Let $F : \mathbb{F}_2^n \to \mathbb{F}_2^m$ be a vectorial Boolean function. When F is a block cipher or a cryptographic permutation, it is a non-linear operation of high degree. However, a linear transformation from subsets of $\mathbb{F}_2^n$ to subsets of $\mathbb{F}_2^m$ can also be derived from it. That is, there exists a linear operator, called the transition matrix, that maps δ_x to $\delta_{F(x)}$. More generally, it maps any function $f : \mathbb{F}_2^n \to \mathbb{K}$ to a function $f' : \mathbb{F}_2^m \to \mathbb{K}$ in the following manner:

$$f'(y) = \sum_{\substack{x \in \mathbb{F}_2^n \\ F(x) = y}} f(x).$$

Definition 1 (Transition matrix [Bey21])**.** Let $\mathbb{K}$ be a field and let $F : \mathbb{F}_2^n \to \mathbb{F}_2^m$. Define $T^F : \mathbb{K}^{\mathbb{F}_2^n} \to \mathbb{K}^{\mathbb{F}_2^m}$ as the unique linear operator that maps δ_x to $\delta_{F(x)}$. The transition matrix of F is the coordinate representation of T^F with respect to the Kronecker delta bases of $\mathbb{K}^{\mathbb{F}_2^n}$ and $\mathbb{K}^{\mathbb{F}_2^m}$.

The transition matrix has useful properties that make it possible to describe the transition matrix of a cryptographic function based on the transition matrices of its components.

Theorem 1 (Properties of transition matrices (derived from [Bey21]))**.** *Let $F : \mathbb{F}_2^n \to \mathbb{F}_2^m$ be a vectorial Boolean function. The transition matrix of F, T^F, has the following properties:*

1. *If F is a bijection, then T^F is a permutation matrix.*
2. *If $F(x_2 \| x_1) = F_2(x_2) \| F_1(x_1)$, then $T^F = T^{F_2} \otimes T^{F_1}$.*
3. *If $F = F_2 \circ F_1$, then $T^F = T^{F_2} T^{F_1}$.*
4. *Let $g \in \mathbb{F}_2^{\mathbb{F}_2^m}$, then $g \circ F = \left(T^F\right)^\top g$, where $\left(T^F\right)^\top$ is the transpose of T^F.*

Proof. The first three properties are proven in [Bey21, Thm. 3.1]. The fourth property refers to the pullback operator [Bey21, Def. 3.1]. However, because the matrix representation of the pullback operator is the transpose of the transition matrix [Bey21, Sec. 3.2], we opt not to define the pullback operator and work with the transpose of T^F instead. □

2.2.2 Correlation Matrices

The transition matrix can be considered in different bases, either to simplify analysis or to emphasize specific properties of the functions. One such basis consists of the group characters of $\mathbb{F}_2^n$. These are the functions $\chi_u : \mathbb{F}_2^n \to \mathbb{R}$ defined by $\chi_u(x) = (-1)^{u^\top x}$, with $u \in \mathbb{F}_2^n$. The basis of group characters is of particular interest, because it diagonalizes the transition matrix of any constant addition, $x \mapsto x + t$, resulting in a simple representation of the key addition. The change-of-basis transformation from the Kronecker delta basis to the character basis is the Fourier transformation.

Definition 2 (Fourier Transformation [Bey21])**.** The Fourier transformation of a function $f \in \mathbb{R}^{\mathbb{F}_2^n}$ is the function $\mathscr{F}_n(f) = \widehat{f} \in \mathbb{R}^{\mathbb{F}_2^n}$ defined by $\widehat{f}(u) = \sum_{x \in \mathbb{F}_2^n} \chi_u(x) f(x)$. That is, $\widehat{f}(u)/2^n$ is equal to the coordinate of f corresponding to χ_u in the character basis.

Daemen et al. [DGV95] define the correlation matrix of a vectorial Boolean function F as the matrix C^F such that each coordinate is equal the correlation of a linear approximation. Beyne [Bey21] defines it as the Fourier transformation of the transition matrix of F. Both definitions are equivalent, but using the second definition the properties of the transition matrix can be translated to properties of the correlation matrix.

Definition 3 (Correlation Matrix [Bey21])**.** Let $F : \mathbb{F}_2^n \to \mathbb{F}_2^m$. Define $C^F : \mathbb{R}^{\mathbb{F}_2^n} \to \mathbb{R}^{\mathbb{F}_2^m}$ as the Fourier transformation of T^F. That is, $C^F = \mathscr{F}_m T^F \mathscr{F}_n^{-1}$. The correlation matrix of F is the coordinate representation of T^F with respect to the character bases of $\mathbb{R}^{\mathbb{F}_2^n}$ and $\mathbb{R}^{\mathbb{F}_2^m}$.

Theorem 2 (Properties of correlation matrices (derived from [Bey21]))**.** *Let $F : \mathbb{F}_2^n \to \mathbb{F}_2^m$ be a vectorial Boolean function. Its correlation matrix, C^F, has the following properties:*

1. *If F is a bijection, then C^F is an orthogonal matrix.*
2. *If $F = F_2 \| F_1$, then $C^F = C^{F_2} \otimes C^{F_1}$.*

3. If $F = F_2 \circ F_1$, then $C^F = C^{F_2} C^{F_1}$.

4. Let $g \in \mathbb{R}^{\mathbb{F}_2^m}$, then $\widehat{g \circ F} = 2^{n-m} \big(C^F\big)^\top \widehat{g}$, where $\big(C^F\big)^\top$ is the transpose of C^F.

Proof. The first three properties are proven in [Bey21, Thm. 3.2]. The fourth property follows from $\widehat{g \circ F} = \mathscr{F}_n (T^F)^\top g = \mathscr{F}_n (T^F)^\top \mathscr{F}_m^{-1} \widehat{g}$ and the fact that $\mathscr{F}_n^{-1} = \mathscr{F}_n^\top / 2^n$. □

2.2.3 Trails and Approximations

Let F be a vectorial Boolean function consisting of multiple rounds: $F = F_r \circ \cdots \circ F_1$. By the third property of Theorem 2, the correlation matrix of F can be decomposed as $C^F = C^{F_r} \cdots C^{F_1}$. Expanding this matrix product into coordinates results in

$$C^F_{u_r, u_0} = \sum_{u_1, \ldots, u_{r-1}} \prod_{i=1}^{r} C^{F_i}_{u_i, u_{i-1}}. \tag{2}$$

This equation expresses the correlation of a linear approximation as a sum of the correlations of trails, where a trail is a sequence $(u_0, \ldots, u_r)$ and its correlation is the product $\prod_{i=1}^{r} C^{F_i}_{u_i, u_{i-1}}$. By only considering a subset of the trails from u_0 to u_r with dominant absolute correlation, it is possible to approximate the actual correlation. Let Λ be this chosen set of dominant trails and let Λ^c be the set of all other trails. The correlation of a linear approximation can then be approximated as:

$$\begin{aligned} C^F_{u_r, u_0} &= \sum_{(u_1, \ldots, u_{r-1}) \in \Lambda} \prod_{i=1}^{r} C^{F_i}_{u_i, u_{i-1}} + \sum_{(u_1, \ldots, u_{r-1}) \in \Lambda^\mathsf{c}} \prod_{i=1}^{r} C^{F_i}_{u_i, u_{i-1}} \qquad (3) \\ &\approx \sum_{(u_1, \ldots, u_{r-1}) \in \Lambda} \prod_{i=1}^{r} C^{F_i}_{u_i, u_{i-1}}. \qquad (4) \end{aligned}$$

The absolute error of this approximation is $\big| \sum_{(u_1, \ldots, u_{r-1}) \in \Lambda^\mathsf{c}} \prod_{i=1}^{r} C^{F_i}_{u_i, u_{i-1}} \big|$ and the approximation only holds under the assumption that this error term is small.

2.3 Binary Möbius Transformation

The coefficients in the ANF of a Boolean function can be viewed as coordinates in the basis of monomials. The elements of this basis are denoted by $\mu_u(x) = x^u$. The change-of-basis transformation from the Kronecker delta basis to the monomial basis is known as the binary Möbius transformation.

Definition 4 (Binary Möbius Transformation)**.** The binary Möbius transformation of a Boolean function $f \in \mathbb{F}_2^{\mathbb{F}_2^n}$ is the Boolean function $\mathscr{M}_n(f) = f^\circ \in \mathbb{F}_2^{\mathbb{F}_2^n}$, where $f^\circ(u)$ is the coordinate of f corresponding to μ_u in the monomial basis.

By applying the Möbius inversion formula [Rot64, Prop. 2] to Equation (1), the binary Möbius transformation can be explicitly written as

$$f^\circ(u) = \sum_{x \preceq u} f(x). \tag{5}$$

The binary Möbius transformation is its own inverse. Furthermore, the matrix representation of the Möbius transformation derived from Equation (5) has a recursive structure that can be exploited to compute the binary Möbius transformation in $\mathcal{O}(n2^n)$ instead of $\mathcal{O}(2^{2n})$ operations. A description of this algorithm can be found in [Car20].

Theorem 3. *The binary Möbius transformation $\mathscr{M}_n$ has the following properties:*

1. *It is an involution:* $\mathscr{M}_n^{-1} = \mathscr{M}_n$*.*

2. *Its matrix representation has the following structure:* $\mathscr{M}_n = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}^{\otimes n}$.

Proof. These properties can, for example, be derived from [PWZ11, Thm. 2, Lem. 1]. □

Example 1. Let $f \in \mathbb{F}_2^{\mathbb{F}_2^2}$ be a 2-bit Boolean function with a truth table equal to $[0, 1, 0, 0]^\mathsf{T}$. The algebraic normal form coefficients of f can be computed by multiplication with $\mathscr{M}_2$:

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \end{bmatrix} .$$

This results in the truth table of f°, from which one can read that $x^{e_1} = x_1$ and $x^{\underline{1}} = x_1 x_2$ are the only two monomials in the ANF of f. That is, $f(x) = x_1 + x_1 x_2$ with $x = (x_1, x_2)$. From the second property of Theorem 3, it follows that the ANF coefficients of f° are given by the truth table of f, such that $f^\circ(u) = u_1$ with $u = (u_1, u_2)$. ▷

3 Algebraic Transition Matrices

Similar to correlation matrices, algebraic transition matrices are the result of applying a change-of-basis transformation to transition matrices. The choice of basis can be best understood from the perspective of integral cryptanalysis. Therefore, we first define integral and generalized integral properties in Section 3.1. Then, algebraic transition matrices are introduced in Section 3.3. Section 3.4 discusses algebraic trails and Section 3.5 discusses key addition.

3.1 Integral Cryptanalysis

In this work we consider a general definition of integral properties that encompasses the original integral properties from [KW02], division properties [Tod15, TM16] and, the properties from the linearly equivalent S-boxes method of [LDF20].

Definition 5 (Integral Property)**.** Let $F : \mathbb{F}_2^n \to \mathbb{F}_2^m$ be a vectorial Boolean function. An integral property for F is a pair (X, r) with $X \subseteq \mathbb{F}_2^n$ and $r : \mathbb{F}_2^m \to \mathbb{F}_2$, and its evaluation is equal to

$$\sum_{x \in X} r(F(x)) \,. \tag{6}$$

For example, a division property consists of an affine subspace $a+V$ that after evaluation leads to a zero sum for the i^{th} bit of the output. With Definition 5, this can be described by taking X equal to the affine subspace and by taking r equal to the degree-one monomial corresponding to that bit, i.e. $X = a + V$ and $r = x^{e_i}$. In [HLLT21], Hebborn et al. consider similar integral properties to Definition 5, but only with functions r of degree 1.

To the best of the authors' knowledge, no higher degree functions r have been considered in the integral cryptanalysis literature, although existing division property techniques can certainly find them. From the viewpoint of transition matrices, higher degree functions r are a natural generalization. Indeed, the sum in Equation (6) can be rewritten as a dot-product: $\sum_{x \in X} r(F(x)) = (r \circ F) \cdot \mathbb{1}_X$. Applying Property 4 from Theorem 1 gives

$$\sum_{x \in X} r(F(x)) = \left(\left(T^F\right)^\mathsf{T} r\right) \cdot \mathbb{1}_X = r \cdot \left(T^F \mathbb{1}_X\right) . \tag{7}$$

Equation (7) shows that Equation (6) is in fact an evaluation of the bilinear form defined by the transition matrix T^F.

In Section 5, a new algorithm is developed that searches for all constant-sum integral properties of the form described in Definition 5 within a given subset. However, as will be discussed in Section 5, it is actually simpler to first search for properties that are a further generalization of Definition 5 and then search for the integral properties of Definition 5. These generalized integral properties are defined in Definition 6. Any integral property can be turned into a generalized integral property by taking $r' = \mathbb{1}_X \otimes r$. Note that a generalized integral property can also be defined as an integral property for $x \mapsto (x, F(x))$.

Definition 6 (Generalized Integral Property)**.** Let $F : \mathbb{F}_2^n \to \mathbb{F}_2^m$ be a vectorial Boolean function. A generalized integral property for F is a function $r' : \mathbb{F}_2^n \times \mathbb{F}_2^m \to \mathbb{F}_2$, and its evaluation is equal to

$$\sum_{x \in \mathbb{F}_2^n} r'(x, F(x)). \tag{8}$$

3.2 Precursor Basis

Equation (7) can be modified further to simplify analysis by using the ANF of r. When replacing r by r°, an extra matrix multiplication $\mathscr{M}_m^{-1}$ has to be inserted in the equation:

$$\left(T^F\right)^\mathsf{T} r \cdot \mathbb{1}_X = \left(T^F\right)^\mathsf{T} \mathscr{M}_m^{-1} r^\circ \cdot \mathbb{1}_X .$$

However, to complete the change-of-basis transformation of the transition matrix, $\mathbb{1}_X$ has to be described in the dual of the monomial basis. We call this basis, with change-of-basis transformation $\mathscr{P}_n = \mathscr{M}_n^{-T}$, the *precursor basis.* The name is due to the observation that the precursor basis vectors π_u, are the indicator functions of precursor sets: $\pi_u = \mathbb{1}_{\mathrm{Prec}(u)}$.

Definition 7 (Dual Binary Möbius Transformation)**.** The dual binary Möbius transformation of a Boolean function $f \in \mathbb{F}_2^{\mathbb{F}_2^n}$ is the Boolean function $\mathscr{P}_n(f) = \tilde{f} \in \mathbb{F}_2^{\mathbb{F}_2^n}$, where $\tilde{f}(u)$ is the coordinate of f corresponding to π_u in the precursor basis.

Similar to the properties in Theorem 3 for the ordinary binary Möbius transformation, the dual binary Möbius transformation is also its own inverse and its matrix representation has an analogous recursive structure.

Theorem 4. *The dual binary Möbius transformation $\mathscr{P}_n$ has the following properties:*

1. *It is an involution: $\mathscr{P}_n^{-1} = \mathscr{P}_n$.*
2. *Its matrix representation has the following structure: $\mathscr{P}_n = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}^{\otimes n}$.*

Using Definition 7, Equation (6) can be rewritten as

$$\sum_{x \in X} r(F(x)) = r^\circ \cdot \mathscr{P}_m T^F \mathscr{P}_n^{-1} \tilde{\mathbb{1}}_X .$$

This gives us the definition of the *algebraic transition matrix.*

3.3 Algebraic Transition Matrices

Algebraic transition matrices can now be defined by a change-of-basis of transition matrices from the Kronecker delta basis to the precursor basis.

Definition 8. (Algebraic Transition Matrix) Let $F : \mathbb{F}_2^n \to \mathbb{F}_2^m$. Define $A^F : \mathbb{F}_2^{\mathbb{F}_2^n} \to \mathbb{F}_2^{\mathbb{F}_2^m}$ as the dual binary Möbius transformation of T^F. That is, $A^F = \mathscr{P}_m T^F \mathscr{P}_n^{-1}$. The algebraic transition matrix of F is the coordinate representation of T^F with respect to the precursor bases of $\mathbb{F}_2^{\mathbb{F}_2^n}$ and $\mathbb{F}_2^{\mathbb{F}_2^m}$

The algebraic transition matrix has similar properties to the transition matrix and correlation matrix. However, note that Property 1 of Theorem 5 is weaker than Property 1 of Theorem 2, because $\mathscr{P}_n$ is not orthogonal.

Theorem 5 (Properties of Algebraic Transition Matrices)**.** *Let $F : \mathbb{F}_2^n \to \mathbb{F}_2^m$ be a vectorial Boolean function. The algebraic transition matrix of F, A^F, has the following properties:*

1. *If F is a bijection, then A^F is invertible and $\left(A^F\right)^{-1} = A^{F^{-1}}$.*
2. *If $F = F_1 \| F_2$, then $A^F = A^{F_2} \otimes A^{F_1}$.*
3. *If $F = F_2 \circ F_1$, then $A^F = A^{F_2} A^{F_1}$.*
4. *Let $g \in \mathbb{F}_2^{\mathbb{F}_2^m}$, then $(g \circ F)^\circ = \left(A^F\right)^\mathsf{T} g^\circ$.*

Proof. Similar to Theorem 2, all these properties follow from Theorem 1, the change-of-basis transformation $A^F = \mathscr{P}_m T^F \mathscr{P}_n^{-1}$ and the fact that $\mathscr{P}_n = \mathscr{P}_1^{\otimes n}$. □

Using Property 4 of the theorem above, an exact formula for each element of the algebraic transition matrix can also be derived.

Theorem 6. *Let $F : \mathbb{F}_2^n \to \mathbb{F}_2^m$. For all $u \in \mathbb{F}_2^n$ and $v \in \mathbb{F}_2^m$, one has $A^F_{v,u} = \left(F^v\right)^\circ(u)$.*

Proof. Consider Property 4 of Theorem 5. By choosing $g = \mu_v$, we get $\left(F^v\right)^\circ = \left(A^F\right)^\mathsf{T} \delta_v$. This means that each column of $\left(A^F\right)^\mathsf{T}$ is equal to $\left(F^v\right)^\circ$. Therefore, we can conclude that $A^F_{v,u} = \left(F^v\right)^\circ(u)$. □

Example 2. Let $F : \mathbb{F}_2^2 \to \mathbb{F}_2^2$ be the vectorial Boolean function defined by $F(x) = (F_1(x), F_2(x)) = (x_2 + 1, x_1 + x_1 x_2)$, with $x = (x_1, x_2)$. To compute the algebraic transition matrix, the transition matrix has to be determined first:

$$T^F = \begin{bmatrix} 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}.$$

The algebraic transition matrix can be calculated according to Definition 8. For larger matrices, this computation can be sped up with the algorithm discussed in Section 2.3.

$$A^F = \mathscr{P}_m T^F \mathscr{P}_n^{-1} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

This results in

$$\begin{array}{c|cccc|} & 1 & x_1 & x_2 & x_1x_2 \\ \hline 1 & 1 & 0 & 0 & 0 \\ F_1 & 1 & 0 & 1 & 0 \\ F_2 & 0 & 1 & 0 & 1 \\ F_1F_2 & 0 & 1 & 0 & 1 \end{array} = A^F.$$

From Theorem 6 it follows that each row of A^F contains the ANF coefficients of a product of component functions of F. Indeed, the first row of A^F is the ANF of $F^{\underline{0}} = 1$. The second and third rows are the ANFs of F_1 and F_2 respectively, and the last row is the ANF of the product of F_1 and F_2. The composition of F with a Boolean function can also easily be computed through Property 4 of Theorem 5. For example, let $g(x) = x_1 + x_2$, then the ANF of $g \circ F$ can be computed as

$$(g \circ F)^\circ = \left(A^F\right)^\mathsf{T} g^\circ = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}.$$

In Appendix A, the algebraic transition matrix of the Present S-box is given. ▷

3.4 Algebraic Trails

Using Theorem 6, the ANF coefficients of a function $F : \mathbb{F}_2^n \to \mathbb{F}_2^m$ can be read out from the algebraic transition matrix of F. For a function of the form $F = F_r \circ \cdots \circ F_1$, elements of A^F can be computed in the same way as in Equation (2):

$$A^F_{u_r,u_0} = \sum_{u_1,\ldots,u_{r-1}} \prod_{i=1}^{r} A^{F_i}_{u_i,u_{i-1}} . \tag{9}$$

Analogous to Equation (2), Equation (9) decomposes the matrix product into trails. These *algebraic trails* are defined by a tuple $(u_0, \ldots, u_r) \in \mathbb{F}_2^n \times \cdots \times \mathbb{F}_2^m$ and have an associated correlation equal to $\prod_{i=1}^{r} A^{F_i}_{u_i,u_{i-1}}$. Hence, any element of A^F is a sum of algebraic trails.

Equation (9) and Equation (2) use the same decomposition and therefore, algebraic trails can be treated in the same way as linear trails. However, their correlations are defined over different fields. Unlike in linear cryptanalysis, that is defined over $\mathbb{R}$, the field $\mathbb{F}_2$ only has the trivial norm, which is not useful for approximations. The computation can still be simplified by considering a set Λ containing – but not limited to – all algebraic trails with a nonzero-correlation, such that

$$A^F_{u_r,u_0} = \sum_{(u_1,\ldots,u_{r-1}) \in \Lambda} \prod_{i=1}^{r} A^{F_i}_{u_i,u_{i-1}} . \tag{10}$$

If Λ is shown to be empty, then it can be concluded that the sum is equal to zero. This is very similar to zero-correlation linear or impossible differential cryptanalysis, where the goal is also to show that there only exist trails or characteristics with respectively zero correlation or zero probability. In some cases, Λ is not empty but all trails cancel each other so that sum is still zero. In Section 4, the connection between algebraic trails and other notions of trails in integral cryptanalysis, such as division trails and monomial trails, is discussed as well as how these techniques can be applied to find nonzero-correlation algebraic trails.

3.5 Key Addition and Key-Alternating Ciphers

Dependency on a key k that parameterizes a vectorial Boolean function F_k, such as a block cipher, can be included in the algebraic normal form of F_k by parameterizing the ANF coefficients:

$$F_k(x) = \sum_u \lambda_u(k) x^u . \tag{11}$$

By Theorem 6, the coordinates of the algebraic transition matrix can similarly be parameterized by k. Furthermore, Equation (9) still holds for the composition of parameterized vectorial Boolean functions, but the correlations of the algebraic trails will be functions of the key.

In a block cipher, the key is often introduced through addition in $\mathbb{F}_2^n$. If we consider the ANF coefficients to be functions of the key, the algebraic transition matrix of the addition with a single bit k is equal to

$$\begin{bmatrix} 1 & 0 \\ k & 1 \end{bmatrix} . \tag{12}$$

Using Theorem 5, this can be extended to key addition on multiple bits. Theorem 7 gives an explicit formula for each coordinate of the corresponding algebraic transition matrix.

Theorem 7. *Let $\tau_k : \mathbb{F}_2^n \to \mathbb{F}_2^n$ be a vectorial Boolean function defined as $\tau_k(x) = x + k$. For each coordinate $(u, v) \in \mathbb{F}_2^n \times \mathbb{F}_2^m$ of the algebraic transition matrix of τ_k, it holds that:*

$$A^{\tau_k}_{v,u} = \begin{cases} k^{u+v} & \textit{if } u \leq v\,, \\ 0 & \textit{otherwise}\,. \end{cases}$$

Proof. This follows directly from Property 2 of Theorem 5 and Equation (12). □

Key-alternating ciphers are iterated ciphers that alternate the application of a round function with the addition of a round key. Let $k_1, \ldots, k_r$ be r round keys, let $F_1, \ldots, F_r$ be the round functions and let $G_i(x) = F_i(x) + k_i$, then $F = G_r \circ \cdots \circ G_1$ is a key-alternating cipher. By Theorem 7, the ANF coefficients of a key-alternating cipher can be written as

$$A^F_{v_r,v_0} = \sum_{\substack{v_0,u_1,v_1,\ldots,u_{r-1},v_{r-1},u_r \\ u_i \leq v_i}} \prod_{i=1}^{r} k_i^{u_i+v_i} A^{F_i}_{u_i,v_{i-1}}\,. \tag{13}$$

As a consequence of modelling the key addition as a separate layer, the correlations of all algebraic trails are either zero or a monomial function of the key.

4 Applications of Algebraic Transition Matrices

Algebraic transition matrices have implicitly been used in earlier work. In this section we explain how algebraic transition matrices have been used before and what new insights their definition brings. In Section 4.1, the connection between algebraic transition matrices and existing search methods for integral properties is discussed, starting from [BC16]. Furthermore, new search methods derived from algebraic transition matrices and trails are suggested. In Section 4.2, the connection between the integral properties of a permutation and its inverse is discussed from the perspective of the Möbius transformation and the dual Möbius transformation of the transition matrix. Section 4.3 establishes a connection between algebraic transition matrices and invariants. For correlation matrices it has already been shown that invariants are their eigenvectors [Bey18]. A similar result holds for algebraic transition matrices.

4.1 Parity Sets, Monomial Trails and Algebraic Transition Matrices

In this section the connection between parity sets [BC16], monomial trails [HSWW20] and algebraic transition matrices is discussed and Equation (10) is used to develop an improved method for computing elements of algebraic transition matrices.

Parity Sets In [BC16], Boura and Canteaut introduce parity sets as a new approach to understanding division properties of sets, and as an improved technique to search for integral properties. For every set $X \subseteq \mathbb{F}_2^n$, they define a corresponding parity set $\mathcal{U}(X) = \left\{u : \sum_{x \in X} x^u = 1\right\}$ [BC16, Def. 2]. The parity set of a set can also be described as an involutive linear transformation of the indicator vector of X [BC16, Thm. 1]. In fact, this transformation is equal to the change-of-basis transformation $\mathscr{P}_n$. Therefore $\tilde{\mathbb{1}}_X$, is equal to $\mathbb{1}_{\mathcal{U}(X)}$.

To find integral properties, parity sets can be propagated through a function $F : \mathbb{F}_2^n \to \mathbb{F}_2^m$ using the inclusion $\mathcal{U}(F(X)) \subseteq \bigcup_{u \in \mathcal{U}(X)} V_F(u)$, where $V_F(u) = \left\{v : F^v(x) \text{ contains } x^u\right\}$ [BC16, Prop. 7]. Since x^u is contained in $F^v(x)$ if and only if $A^F_{v,u}$ is nonzero, $V_F(u)$ is also equal to the support of the column of A^F at index u. This is also reflected in [BC16, Table 2], which, up to the order of rows and columns, is equal to the transpose of the algebraic transition matrix of the Present S-box given in Appendix A. This shows that

[BC16, Prop. 7] can be improved upon by considering it as a matrix-vector product with the transition matrix:

$$\mathcal{U}(F(X)) = \mathop{\triangle}_{u \in \mathcal{U}(X)} V_F(u)\,, \tag{14}$$

where $\triangle$ denotes the symmetric difference of sets. Furthermore, parity sets can also be propagated through the addition with a secret key. By [BC16, Prop. 6], it holds that $\mathcal{U}(k + X) \subseteq \bigcup_{u \in \mathcal{U}(X)} \mathrm{Succ}(u)$. This can directly be deduced from the algebraic transition matrix of the key addition operation, which is given in Theorem 7.

To find integral properties for a key alternating cipher F, [BC16, Prop. 6 and Prop. 7] can repeatedly be applied to a growing superset of the parity set, starting from an initial set X. Any element, v that is not present in the final superset then leads to an integral property of the form $\sum_{x \in X} F^v(x) = 0$. However, [BC16, Prop. 7] also gives rise to a trail based approach. Consider $F = F_r \circ \cdots \circ F_1$, then there is a parity-set derived trail $(u_0, \ldots, u_r)$ if and only if for all u_i it holds that $u_i \in V_{F_i}(u_{i-1})$. Since u_i is an element of $V_{F_i}(u_{i-1})$ if and only if $A^F_{u_i,u_{i-1}}$ is nonzero, the trails derived from parity sets coincide with the algebraic trails with a nonzero correlation. Therefore, parity set propagation can be used to model the set Λ of all nonzero-correlation algebraic trails in Equation (10).

Monomial Trails The work on monomial trails by Hu et al. [HSWW20] can be seen as dual to the parity set approach. Whereas parity sets are propagated in the forward direction through a function, monomial trails result from the backward propagation of monomials. That is, a monomial trail through $F(x) = y$ is defined by the relation that x^u propagates to y^v if and only if the ANF of $F^v(x)$ contains the monomial x^u. Again, this holds if and only if $A^F_{u_i,u_{i-1}}$ is nonzero. However, since the propagation of monomial trails happens in the backwards direction, the monomial trails correspond to the nonzero-correlation trails resulting from the decomposition of $(A^F)^\mathsf{T}$ like in Equation (9). By undoing the transpose, which corresponds to reversing the monomial trails, the corresponding algebraic trail can be found. In this way, every monomial trail corresponds to a nonzero-correlation algebraic trail and every nonzero-correlation algebraic trail corresponds to a monomial trail.

New Method Due to the relationship between parity sets, monomial trails and the division property, techniques based on the division property [Tod15, TM16] and their efficient trail-based implementations as MILP or SAT problems [XZBL16, EKKT19, Udo21, DL22] can also be used to model Λ or show that it is empty. However, algebraic transition matrices allow for greater flexibility in the design of new methods to find integral properties or to evaluate coordinates of A^F. We propose a new method that improves upon [HSWW20] by using Equation (10). This method will also be used in Section 5.

The goal of this method is to evaluate the value of $A^F_{v,u}$ or, if this is not feasible, to determine whether or not $A^F_{v,u}$ is key-dependent. The method starts by enumerating up to N_1 different algebraic trails with a key-dependent correlation. If N_1 is reached, the algorithm cannot conclude anything. If there are less than N_1 key dependent trails, the sum of the correlations of the enumerated trails can be computed and a conclusion about the key dependence of $A^F_{v,u}$ can be made. This process can also be sped up by enumerating the key-dependent trails per monomial in the key. Since Equation (13) indicates that the correlation all trails is either zero or a monomial in the key, an odd number of trails for a specific monomial of the key implies that $A^F_{v,u}$ is key dependent.

Then, as a second step, up to N_2 algebraic trails with correlation one are enumerated. If N_2 trails are enumerated, then only the conclusion of the previous step can be used. If less than N_2 trails are enumerated, then their sum can be computed, which results in the value of $A^F_{v,u}$. This method is easy in use, as it requires only up to $N_1 + N_2$ calls to a SAT or MILP solver with similar models as used in [XZBL16, EKKT19, HSWW20, DL22], with some slight tweaks to keep track of the key dependence and to remove enumerated trails. It is also more flexible, as it can compute $A^F_{v,u}$ exactly as long as there less than N_1 key-dependent algebraic trails and less than N_2 algebraic trails with correlation one. This is unlike most previous methods, which can only differentiate between zero and nonzero or constant and non-constant.

4.2 Duality

Due to the non-orthogonality of $\mathscr{P}_n$, two different but related change-of-basis transformations of the transition matrix exist. Definition 8 is specifically chosen to coincide with the algebraic normal form as well as the parity set approach. However, the dual change-of-basis transformation is also meaningful. This raises the question whether or not more key-independent integral properties can be found by analyzing a cipher in this other basis. In this section, the connection between the two bases is discussed and this question is answered.

Let $E^F = \mathscr{M}_m T^F \mathscr{M}_n^{-1}$ with $F : \mathbb{F}_2^n \to \mathbb{F}_2^m$. In the setting of integral properties, E^F swaps the bases of $\mathbb{1}_X$ and r compared to A^F. That is, r is expressed in the basis of indicators of precursor sets and X is expressed as a symmetric difference of the support of monomial functions:

$$\sum_{x \in X} r(F(x)) = \left(\left(E^F\right)^\intercal \tilde{r}\right) \cdot \mathbb{1}_X^\circ = \tilde{r} \cdot \left(E^F \mathbb{1}_X^\circ\right). \tag{15}$$

Since both the precursor sets and the monomials are a basis for the space of Boolean functions, properties of E^F can be translated to properties of the algebraic transition matrix A^F and vice-versa. Therefore, with complete knowledge of A^F or E^F, the choice between them is arbitrary, as it will not result in any new information.

$$\begin{tikzcd}
A^F \arrow[r, "\intercal"] \arrow[d, "\mathscr{M}\mathscr{P}^{-1}"'] & E^{F^*} \arrow[d, "\mathscr{P}\mathscr{M}^{-1}"] \\
E^F \arrow[r, "\intercal"] & A^{F^*}
\end{tikzcd}$$

Figure 1: Relations between the algebraic transition matrix and E^F.

Figure 1 shows the relations between E^F and A^F. In the diagram, we also refer to

$$E^{F^*} = \mathscr{M}_n\left(T^F\right)^\intercal \mathscr{M}_m^{-1} \text{ and } A^{F^*} = \mathscr{P}_n\left(T^F\right)^\intercal \mathscr{P}_m^{-1}.$$

Note by Property 1 of Theorem 1, if F is a bijection then $(T^F)^\intercal$ is equal to $T^{F^{-1}}$. Hence, if F is a bijection, then E^{F^*} is equal to $E^{F^{-1}}$ and A^{F^*} is equal to $A^{F^{-1}}$. The horizontal arrows indicate that the transpose of A^F and E^F are equal to the change-of-basis of $(T^F)^\intercal$ with the dual basis. That is, $(A^F)^\intercal = E^{F^*}$ and $(E^F)^\intercal = A^{F^*}$. Hence, any integral property on a bijection can be converted to an equivalent integral property on its inverse.

Theorem 8. *Let $F : \mathbb{F}_2^n \to \mathbb{F}_2^n$ be a bijection. Every integral property on F corresponds to an integral property on F^{-1} as follows:*

$$\sum_{x \in X} r(F(x)) = \sum_{x \in \operatorname{supp}(r)} \mathbb{1}_X\left(F^{-1}(x)\right).$$

Conversely, any integral property on F^{-1} corresponds to an integral property on F.

Proof. This follows by substituting $(T^F)^\mathsf{T} = T^{F^{-1}}$ in Equation (7):

$$\sum_{x \in X} r(F(x)) = \left(\left(T^F\right)^\mathsf{T} r\right) \cdot \mathbb{1}_X = \mathbb{1}_X \cdot \left(T^{F^{-1}} r\right) = \sum_{x \in \mathrm{supp}(r)} \mathbb{1}_X\left(F^{-1}(x)\right). \qquad \square$$

The vertical arrows in Figure 1 indicate the change-of-basis from the precursor set basis to the monomial basis or vice-versa. That is, $\mathscr{M}_n \mathscr{P}_n^{-1} A^F \mathscr{P}_n \mathscr{M}_n^{-1} = E^F$. By exploiting the structure of $\mathscr{M}_n \mathscr{P}_n^{-1}$, Theorem 9 shows that some elements of A^F and E^F are equal. Note that Equation (13) implies that independent pre- and post-whitening keys guarantee that the conditions in Theorem 9 are met.

Theorem 9. *Let $F : \mathbb{F}_2^n \to \mathbb{F}_2^m$ be a vectorial Boolean function and let $u \in \mathbb{F}_2^n$ and $v \in \mathbb{F}_2^m$. It holds that $E^F_{\overline{v},\overline{u}} = \sum_{v' \leq v} \sum_{u' \geq u} A^F_{v',u'}$ and $A^F_{\overline{v},\overline{u}} = \sum_{v \leq v'} \sum_{u \geq u'} E^F_{v',u'}$. In particular, if for all $u' > u$ and for all $v' < v$ it holds that $A^F_{v',u'} = 0$ then $A^F_{v,u} = E^F_{\overline{v},\overline{u}}$. Conversely, if for all $u' < u$ and for all $v' > v$ it holds that $E^F_{v',u'} = 0$, then $E^F_{v,u} = A^F_{\overline{v},\overline{u}}$.*

Proof. By Property 2 of Theorems 3 and 4 and multiplication of 2 × 2 matrices,

$$\mathscr{M}_n \mathscr{P}_n^{-1} = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}^{\otimes n} \text{ and } \mathscr{P}_n \mathscr{M}_n^{-1} = \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}^{\otimes n}.$$

Hence, it holds that $E^F_{\overline{v},\overline{u}} = \sum_{v' \leq v} \sum_{u' \geq u} A^F_{v',u'}$ and $A^F_{\overline{v},\overline{u}} = \sum_{v \leq v'} \sum_{u \geq u'} E^F_{v',u'}$. The second part follows immediately from the first part. □

Due to Theorem 8, it suffices to analyze either A^F and E^F, A^F and $A^{F^{-1}}$, E^F and $E^{F^{-1}}$ or $A^{F^{-1}}$ and $E^{F^{-1}}$ when searching for key-independent integral properties. When pre- and post-whitening keys are present, it further suffices to analyze any of A^F, E^F, $A^{F^{-1}}$ and $E^{F^{-1}}$, due to Theorem 9. However, this only holds if every element of these matrices is computed exactly. In the case of trail-based analysis without computing the exact sum in Equation (9), such as the methods discussed in Section 4.1, different trails and thus potentially different results, are expected in each case. However, when every modelled round has a pre- and post-whitening key, Theorem 9 applies on a per-trail basis and the analysis of one of A^F, E^F, $A^{F^{-1}}$ and $E^{F^{-1}}$ is sufficient.

In [Udo21, Prop. 6], Udovenko remarks that if u propagates to v through F then $\overline{v}$ propagates to $\overline{u}$ through F^{-1} given bit-based division propagation. Since bit-based division propagation is equivalent to parity sets propagation with the assumption of a full-state pre- and post-whitening key in every round, Theorems 8 and 9 can be applied consecutively to construct an alternative explanation for this phenomenon. Furthermore, this explanation also shows that bit-based division propagation in the dual basis does not result in new properties either.

4.3 Invariants

A nonlinear invariant [TLS16] of a function $F : \mathbb{F}_2^n \to \mathbb{F}_2^n$ is a Boolean function $g : \mathbb{F}_2^n \to \mathbb{F}_2$ such that there exists a constant c where for all $x \in \mathbb{F}_2^n$,

$$g(F(x)) = g(x) + c. \tag{16}$$

In [TLS16], nonlinear invariants are computed by solving a system of linear equations derived from Equation (16). Indeed, the algebraic normal form of g satisfies

$$\sum_{u \in \mathbb{F}_2^n} g^\circ(u)\left(x^u + F^u(x)\right) = c.$$

Solving this system in g° for both values of c comes down to finding the kernel of $(A^F+I)^\mathsf{T}$ and a single solution to the system $(A^F+I)^\mathsf{T}f=1$. Note that the kernel of $(A^F+I)^\mathsf{T}$ is equal to the eigenspace of $(A^F)^\mathsf{T}$ corresponding to eigenvalue one.

From Beyne's work [Bey18], it is known that nonlinear invariants correspond to eigenvectors of the correlation matrix of F with corresponding eigenvalues ±1. This can be explained by rewriting Equation (16) in terms of correlation matrices and $f=(-1)^g$, which results in the eigenvalue problem

$$\left(C^F\right)^\mathsf{T}\widehat{f}=(-1)^c\,\widehat{f}\,. \tag{17}$$

If F is a permutation, then the eigenvectors of $(C^F)^\mathsf{T}$ and C^F coincide. Although the eigenvectors can be different for general functions F, the matrices $(C^F)^\mathsf{T}$ and C^F both fully characterize the invariants of F. This is clear from the following more general definition of invariants given by Beyne in later work [Bey21].

Definition 9 (Invariant [Bey21])**.** Let $F:\mathbb{F}_2^n\to\mathbb{F}_2^n$. A subspace V of $\mathbb{R}^{\mathbb{F}_2^n}$ such that $C^FV\subseteq V$, is called an invariant of F.

If V is an invariant of C^F, then the orthogonal complement $V^\perp$ of V satisfies $(C^F)^\mathsf{T}V^\perp\subseteq V^\perp$. Beyne further notes that if F is a permutation, then any invariant V splits into one-dimensional eigenspaces of C^F. Since the correlation matrix and algebraic transition matrix are both a change-of-basis of the transition matrix, it is no surprise that the eigenvectors of the correlation matrix and the eigenvectors of the algebraic transition matrix correspond to the same invariants.

Definition 10 (Invariant (alternative definition))**.** Let $F:\mathbb{F}_2^n\to\mathbb{F}_2^n$. A subspace V of $\mathbb{F}_2^{\mathbb{F}_2^n}$ such that $A^FV\subseteq V$, is called an invariant of F.

However, since algebraic transition matrices are not defined over an algebraically closed field, a full eigendecomposition is in general not possible, even for permutations. Hence, it is not enough to look at eigenvectors alone. To circumvent this problem, we propose a new method that directly computes the invariants from the primary decomposition and generalized Jordan form of the algebraic transition matrix.

Theorem 10 (Primary Decomposition [Dum04, p. 465-466])**.** *Let L be a linear operator on $\mathbb{F}_2^n$. The module $\mathbb{F}_2^{\mathbb{F}_2^n}$ over the principal ideal domain $\mathbb{F}_2[L]$ is isomorphic to the direct sum of the quotients of $\mathbb{F}_2[L]$ with the module's primary ideals, (f_i): $\mathbb{F}_2^{\mathbb{F}_2^n}\cong\bigoplus_i\mathbb{F}_2[L]/(f_i)$. These primary ideals are called the elementary divisors of the $\mathbb{F}_2[L]$-module $\mathbb{F}_2^{\mathbb{F}_2^n}$.*

In practice, Theorem 10 implies that the space $\mathbb{F}_2^n$ is a direct sum of invariant subspaces of the linear operator L. Hence, up to choosing arbitrary bases for these subspaces, the matrix representation of L is of the form

$$\begin{bmatrix} M(f_1) & & & \\ & M(f_2) & & \\ & & \ddots & \\ & & & M(f_l) \end{bmatrix},$$

where $M(f_i)$ is a matrix with minimal polynomial equal to f_i.

Since $\mathbb{F}_2[L]$ is a principal ideal domain, $f_i=g^r$ with g an irreducible polynomial and r a positive integer. If $r>1$, then the corresponding invariant subspace V_{f_i} contains smaller invariants subspaces corresponding to the kernels of the restriction of $g(L)^j$ to V_{f_i}, for j between 1 and r. This gives rise to chains of invariant subspaces that can be seen as a generalization of Jordan chains. To express these additional subspaces in the matrix representation of L, each matrix $M(f_i)$ is chosen to equal the hypercompanion matrix or generalized Jordan block of f_i. Up to arbitrary reordering of the invariant subspaces, this fixes the decomposition of L completely.

Theorem 11 (Hypercompanion Matrix or Generalized Jordan Block [Rob70])**.** *Let $g = x^d + \sum_{i=0}^{d-1} \alpha_i x^i$ be an irreducible polynomial of degree d and let $f = g^r$. The hypercompanion matrix or generalized Jordan block of f is the following block matrix of size $rd \times rd$:*

$$\begin{bmatrix} C_g & N & & \\ & \ddots & \ddots & \\ & & C_g & N \\ & & & C_g \end{bmatrix},$$

where C_g is the companion matrix of g, which is a matrix of the form

$$\begin{bmatrix} 0 & 1 & & \\ & \ddots & \ddots & \\ & & 0 & 1 \\ -\alpha_0 & -\alpha_1 & \cdots & -\alpha_{d-1} \end{bmatrix},$$

and where all entries of N are zero except the bottom left, which is equal to 1. The basis in which a linear transformation L with minimal polynomial f is represented by the above hypercompanion matrix is of the form $\bigcup_{i=0}^{d-1} g^i(L)B$, where B is the basis of the smallest non-trivial invariant subspace of L relative to which the restriction of L is represented by the companion matrix C_g.

Theorem 10 and Theorem 11 can be applied to find all invariants of a vectorial Boolean function $F : \mathbb{F}_2^n \to \mathbb{F}_2^n$ in the following manner. First compute the elementary divisors of T^F and use them to construct the matrix in Theorem 10 with the hypercompanion matrices of Theorem 11. Let us denote this matrix by D. One can compute a change-of-basis matrix P such that $T^F = PDP^{-1}$. By Theorem 10, for each diagonal block in D, the corresponding columns in P span an invariant subspace. Furthermore, for each such invariant subspace, Theorem 11 yields a chain of invariant subspaces that grows from left to right with each companion matrix on the diagonal of the hypercompanion matrix. Any invariant subspace of T^F and therefore, any invariant of F is a direct sum of the invariants resulting from Theorems 10 and 11. Since A^F is a change-of-basis of T^F it suffices to find the invariant subspaces of T^F; their representation in the monomial or precursor basis can be computed by multiplication with $\mathscr{M}_n$ or $\mathscr{P}_n$.

Example 3. To show how this method works in practice, all invariants of the LowMC S-box, that are not a direct sum of other invariants, will be enumerated. The S-box of LowMC is defined by the following permutation [ARS⁺15]:

$$F = \begin{pmatrix} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ 0 & 1 & 3 & 6 & 7 & 4 & 5 & 2 \end{pmatrix}.$$

This can also be written in cycle notation as $F = (0)(1)(2\ 3\ 6\ 5\ 4\ 7)$. Note that the LowMC S-box has two fixed points and a cycle of length six.

The elementary divisors of T^F are $x + 1$, $x + 1$, $(x + 1)^2$ and $(x^2 + x + 1)^2$. The two irreducible factors present in the elementary divisors are $x + 1$ and $x^2 + x + 1$, which have companion matrices respectively equal to

$$\begin{bmatrix} 1 \end{bmatrix} \text{ and } \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}.$$

These companion matrices can be used to construct the hypercompanion matrices of the three distinct elementary divisors $x + 1$, $(x + 1)^2$ and $(x^2 + x + 1)^2$. Respectively, these are

$$\begin{bmatrix} 1 \end{bmatrix}, \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \text{ and } \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}.$$

Finally, their direct sum leads to the matrix D and change-of-basis matrix P:

$$D = \left[\begin{array}{c|c|c:c|cc:cc} 1 & & & & & & & \\ \hdashline & 1 & & & & & & \\ \hdashline & & 1 & 1 & & & & \\ \cdashline{1-8}[1pt/2pt] & & 0 & 1 & & & & \\ \hdashline & & & & 0 & 1 & 0 & 0 \\ & & & & 1 & 1 & 1 & 0 \\ \cdashline{1-8}[1pt/2pt] & & & & 0 & 0 & 0 & 1 \\ & & & & 0 & 0 & 1 & 1 \end{array}\right] \text{ and } P = \left[\begin{array}{c|c|c:c|cc:cc} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \end{array}\right].$$

The invariants can now be read from left to right. The first two blocks, both derived from an elementary divisor $x + 1$, give rise to two one-dimensional invariants corresponding to the two fixed points. The block derived from $(x + 1)^2$ is a hypercompanion matrix derived from the second power of a linear function and therefore it provides both a one dimensional and a two dimensional invariant, which correspond to saturating the cycle of length 6 and taking every second element of that cycle respectively. Finally, the last block is derived from $(x^2 + x + 1)^2$, resulting in an invariant of dimension 2 and an invariant of dimension 4, corresponding to different cycles of sets that can be obtained from the 6-cycle. The nonlinear invariants of the LowMC S-box are equal to its three one-dimensional invariants. In general, all two-dimensional invariants containing a balanced Boolean function and its complement would also correspond to nonlinear invariants. In Appendix B, a more complex example with the Present S-box is given. ▷

5 Finding Integral Properties

In this section an algorithm is constructed that can efficiently find integral properties (X, r) such that $\sum_{x \in X} r(F_k(x)) = 0$, with $F_k : \mathbb{F}_2^n \to \mathbb{F}_2^m$ a vectorial Boolean function parameterized by k. When X or r is fixed this problem can conceptually be solved by finding the largest subspace of the kernel of $r \mapsto r^\circ \cdot A^{F_k} \tilde{\mathbb{1}}_X$, respectively $\mathbb{1}_X \mapsto (A^{F_k})^\mathsf{T} r^\circ \cdot \tilde{\mathbb{1}}_X$, that is independent of k. This approach cannot be directly applied to find $\mathbb{1}_X$ and r simultaneously, as the function $(\mathbb{1}_X, r) \mapsto r^\circ \cdot A^{F_k} \tilde{\mathbb{1}}_X$ is quadratic. This can be overcome by considering the vectorization of A^{F_k}, since for any pair $(\mathbb{1}_X, r)$ it holds that $r^\circ \cdot (A^{F_k} \tilde{\mathbb{1}}_X) = \operatorname{vec}(A^{F_k}) \cdot (\tilde{\mathbb{1}}_X \otimes r^\circ)$, where $\operatorname{vec}(A^{F_k}) \cdot (\delta_u \otimes \delta_v) = A^{F_k}_{v,u}$ for all $u \in \mathbb{F}_2^n$ and $v \in \mathbb{F}_2^m$. Therefore, the largest key-independent subspace of the kernel of $\mathbb{1}_X \otimes r \mapsto \operatorname{vec}(A^{F_k}) \cdot (\tilde{\mathbb{1}}_X \otimes r^\circ)$ contains all integral properties such that $\sum_{x \in X} r(F_k(x)) = 0$ as well as all generalized integral properties of the form $\sum_{x \in \mathbb{F}_2^n} r(F_k(x), x) = 0$.

In practice, the computation of the null space is infeasible because of the size of A^{F_k} and the difficulty of computing elements of A^{F_k} exactly. In Section 5.1, an algorithm is described that searches for a subspace of generalized integral properties within a user-specified vector space. However, the size of subspace that the method can find is ultimately limited by the available memory and computing time. The method relies on a modelling technique such as (three-subset) division property, monomial trails, or the new approach from Section 4.1 as a subroutine. In Section 5.2 the algorithm is applied to Present. In the rest of this section, all vectorial Boolean functions are assumed to be parameterized by a key k, even when this parameter is not written as a subscript.

Remark 1. Interestingly, a concept similar to the vectorization of the algebraic transition matrix was already described by Udovenko in [Udo21]. He considers the ANF of the graph indicator of vectorial Boolean functions. The difference between the two is that $\mathbb{1}^\circ_{\Gamma_F} = \mathscr{M}_{m+n} \mathbb{1}_{\Gamma_F}$ whereas $\operatorname{vec}(A^F) = (\mathscr{M}_n^\mathsf{T} \otimes \mathscr{M}_m) \mathbb{1}_{\Gamma_F}$, where $\Gamma_F = \{(x, F(x)) : x \in \mathbb{F}_2^n\}$ is the graph of the function F. They contain similar information in the sense that $\operatorname{vec}(A^F)(u, v) = 0$ if and only if $\mathbb{1}^\circ_{\Gamma_F}(u, \overline{v}) = 0$.

Remark 2. Generalized integral properties r such that $\sum_{x \in \mathbb{F}_2^n} r(x, F(x)) = 1$ are also of interest. Since these functions r are found by solving a linear system of equations, it suffices to find a single solution and add it to the kernel to find all solutions. The sum $\sum_{x \in \mathbb{F}_2^n} F^{\underline{0}}(x)\underline{0}^x$ is trivially equal to one and therefore yields a property r such that $\sum_{x \in \mathbb{F}_2^n} r(x, F(x)) = 1$. Hence, all such properties can be derived with no additional effort.

5.1 Finding Generalized Integral Properties

In this section we construct an algorithm to compute a subspace of the kernel of $\mathbb{1}_X \otimes r \mapsto \mathrm{vec}(A^F) \cdot (\tilde{\mathbb{1}}_X \otimes r^\circ)$ from a decomposition of $F = F_3 \circ F_2 \circ F_1$, a user-supplied vector space of candidate generalized integral properties, $V \subseteq \mathbb{F}_2^{\mathbb{F}_2^n \times \mathbb{F}_2^m}$, and an oracle $\mathcal{A}$. The oracle $\mathcal{A} : \mathbb{F}_2^{n^{\mathbb{F}_2^m}} \times \mathbb{F}_2^n \times \mathbb{F}_2^m \to \mathbb{F}_2$ gives partial information on the elements of the algebraic transition matrix of a parameterized function. In particular, if $\mathcal{A}(F, u, v)$ returns zero then $A^F_{v,u}$ is zero. Note that the implication only goes in one direction, $\mathcal{A}$ can indicate that $A^F_{v,u}$ is non-constant (by returning one) even though it is actually zero. This emulates the information that can realistically be recovered with imperfect SAT/MILP models to find algebraic trails, such as those described in Section 4.1.

The crux of this algorithm is the following decomposition:

$$\mathrm{vec}(A^F) \cdot (\tilde{\mathbb{1}}_X \otimes r^\circ) = \mathrm{vec}(A^{F_2}) \cdot \left(A^{F_1} \otimes (A^{F_3})^\top\right) \left(\tilde{\mathbb{1}}_X \otimes r^\circ\right) . \tag{18}$$

Equation (18) can be used to compute a subspace of the kernel of $\mathbb{1}_X \otimes r \mapsto \mathrm{vec}(A^F) \cdot (\tilde{\mathbb{1}}_X \otimes r^\circ)$ by computing the nullspace of $S_{F_2}\left(A^{F_1} \otimes (A^{F_3})^\top\right)$, where S_{F_2} selects all the rows (u, v) of $A^{F_1} \otimes (A^{F_3})^\top$ where $\mathcal{A}(F_2, u, v)$ is equal to one. Given an embedding P_V of V into $\mathbb{F}_2^{\mathbb{F}_2^n \times \mathbb{F}_2^m}$, the original kernel computation is reduced to computing the kernel of the smaller matrix $S_{F_2}\left(A^{F_1} \otimes (A^{F_3})^\top\right)P_V$.

Ideally, to simplify the nullspace computation, all elements in $S_{F_2}\left(A^{F_1} \otimes (A^{F_3})^\top\right)P_V$ should be independent of the key. This can be achieved by choosing V in such a way that $S_{F_2}\left(A^{F_1} \otimes (A^{F_3})^\top\right)P_V$ is independent of the key. Such a space V with its basis a subset of the standard basis can be efficiently found for functions F with pre- and post-whitening key additions. Since the standard basis elements define integral properties of the form $\sum_{x \leq u} F^v(x) = A^F_{v,u}$, Lemma 1 below can be applied to dynamically discard large subsets of basis elements. That is, for u and v such that $A^F_{v,u}$ is nonzero, every $A^F_{u',v'}$ with $u' \leq u$, $v \leq v'$ and $(u', v') \neq (u, v)$ will be key-dependent and every corresponding standard basis function $\delta_{u'} \otimes \delta_{v'}$ can be discarded.

Lemma 1. *Let $F_k : \mathbb{F}_2^n \to \mathbb{F}_2^m$ be a vectorial Boolean function parameterized by $k = (k_i, k', k_o)$, with $F_k(x) = F'_{k'}(x + k_i) + k_o$. For all pairs $(u, v) \in \mathbb{F}_2^n \times \mathbb{F}_2^m$ where $A^F_{v,u} \neq 0$ and for all pairs (u', v') such that $u' \| \overline{v'} < u \| \overline{v}$, it holds that $A^F_{v',u'}$ is dependent on the key.*

Proof. Using Equation (13), A^F can be expressed in terms of $A^{F'}$:

$$A^F_{v',u'} = \sum_{(a,b) : a \geq u', b \leq v'} k_i^{a+u'} k_o^{b+v'} A^{F'}_{b,a} . \tag{19}$$

Since we know that $A^F_{v,u} \neq 0$ and therefore that $A^{F'}_{v,u} \neq 0$, Equation (19) contains the summand $k_i^{v+u'} k_o^{v+v'} A^{F'}_{v,u}$. Furthermore, because $k_i^{a+u'} k_o^{b+v'}$ is unique for every summand in Equation (19), $k_i^{v+u'} k_o^{v+v'} A^{F'}_{v,u}$ cannot be cancelled by another term. Hence, $A^F_{v',u'}$ is not constant in the key. □

Algorithm 1 summarizes the complete algorithm. It takes as inputs a function $F = F_3 \circ F_2 \circ F_1$ and a set $C \subseteq \mathbb{F}_2^n \times \mathbb{F}_2^m$, which represents the user-chosen subset of the standard basis from which V will be derived. First, C is filtered, by going through it from high to low weight of $u\|\overline{v}$ (line 3) and removing any key-dependent basis elements based on Lemma 1 (lines 4 to 9). As part of evaluating the criteria for Lemma 1, a sparse representation of $S_{F_2}\big(A^{F_1} \otimes (A^{F_3})^\intercal\big)(\delta_u \otimes \delta_v)$ is computed (line 4). This is reused to determine whether $\delta_u \otimes \delta_v$ is part of the basis that results in a key-independent nullspace computation (lines 10-12). It is also reused in the kernel computation (lines 14-16).

In practice, a sparse data structure is used to keep track of the elements in C. Furthermore, the computation in line 4 is performed by using a SAT model to enumerate all a and b such that there exists a nonzero-correlation algebraic trail $u \xrightarrow{F_1} a \xrightarrow{F_2} b \xrightarrow{F_3} v$. The values $A^{F_1}_{a,u}$ and $A^{F_3}_{v,b}$ are computed exactly and stored for later use by line 10. The computation of $\mathcal{A}(F_2, a, b)$ is also cached for reuse with other pairs (u, v). Line 16 is necessary to transform the representation of the kernel in the basis B back to the complete precursor and monomial bases.

Algorithm 1 Find generalized integral properties that evaluate to zero.

```
Input: $F : \mathbb{F}_2^n \to \mathbb{F}_2^m$, $F = F_3 \circ F_2 \circ F_1$ and $C \subseteq \mathbb{F}_2^n \times \mathbb{F}_2^m$
Output: A subspace of generalized integral properties that sum to zero.
 1: $B \leftarrow \emptyset$
 2: while $|C| > 0$ do                                          ▷ Step 1: filter candidates
 3:     $(u, v) \leftarrow \arg\max_{(x,y)\in C} \mathrm{wt}(x\|\overline{y})$
 4:     $W_{u,v} \leftarrow \mathrm{supp}\big((a, b) \mapsto A^{F_1}_{a,u} A^{F_3}_{v,b} \mathcal{A}(F_2, a, b)\big)$
 5:     if $|W_{u,v}| > 0$ then
 6:         $C \leftarrow C \setminus (\mathrm{Prec}(u) \times \mathrm{Succ}(v))$
 7:     else
 8:         $C \leftarrow C \setminus \{(u, v)\}$
 9:     end if
10:     if $\forall (a, b) \in W_{u,v} : A^{F_1}_{a,u}$ and $A^{F_3}_{v,b}$ are key-independent then
11:         $B \leftarrow B \cup \{(u, v)\}$
12:     end if
13: end while
14: Index the elements $(u_i, v_i)$ of $B$ from 1 to $|B|$                ▷ Step 2: Compute basis
15: $N \leftarrow \ker \begin{bmatrix} \mathbb{1}_{W_{u_1,v_1}} & \cdots & \mathbb{1}_{W_{u_{|B|},v_{|B|}}} \end{bmatrix}$
16: return $P_B N$ where $P_B$ is a matrix with the vectors $\delta_{u_i} \otimes \delta_{v_i}$ as columns.
```

There are two main factors that influence the results of Algorithm 1. The first factor is the choice of F_1 and F_3. With increasing number of rounds, each bit of the output of F_1 and F_3 will be dependent on a larger part of their input. Therefore, the algorithm will find generalized integral properties that are dependent on a larger part of the input and output state. For example, in Present, the found generalized integral properties are localized to a single S-box at the input and output, when F_1 and F_3 are one round. With more rounds, generalized integral properties are found that use the input and output of multiple S-boxes. However, increasing the number of rounds in F_1 and F_3 also increases the key-dependency of $A^{F_1} \otimes (A^{F_3})^\intercal$ and hence reduces the size of V and the number of properties found. In general, it is interesting to run the algorithm multiple times for different partitions of F and take the union of the results.

The second factor that influences the result is how the oracle $\mathcal{A}$ is implemented. More precise methods can give better results, but might be prohibitively expensive. In this work, the models for nonzero-correlation trails have been expressed as SAT problems with the pysat library [IMM18]. The CNF models of the components were computed by finding the prime implicants with [Udo21, Alg. 5] and solving the resulting minimal set cover problem with Google's or-tools [PF23]. All code used in this work can be found at https://github.com/michielverbauwhede/AlgebraicTransitionMatrices.

Remark 3. Algorithm 1 can also be adapted to find constant generalized integral properties, by replacing the oracle $\mathcal{A}$ in line 6 of Algorithm 1 with another oracle that determines whether $A^F_{v,u}$ is constant. The nullspace computation then guarantees that any solution is independent of the key, and therefore constant. However, the algorithm does not give any information on what this constant is. This will result in non-trivial new results, because unlike in Remark 2, the complete kernel is not computed.

Comparison to [LDF20] and [DF20] The only method comparable to Algorithm 1 is the linearly equivalent S-boxes method of Lambin et al. [LDF20], that was further developed by Derbez et al [DF20]. This method searches for integral properties (X, r) on functions $F : \mathbb{F}_2^n \to \mathbb{F}_2^m$, where X is a subspace of dimension $n-1$ and r is a linear function. Apart from the fact that Algorithm 1 can find integral properties in a larger space, it also provides algorithmic improvements over [LDF20, DF20] when restricted to the same subspace and instantiated with the same oracle $\mathcal{A}$.

Similar to Algorithm 1, the linearly equivalent S-boxes method also requires that F is split in three parts $F = F_3 \circ F_2 \circ F_1$. For every pair of invertible linear functions L_i, L_o on $\mathbb{F}_2^n$ and $\mathbb{F}_2^m$ the method tries to find division properties for every $L_o \circ F \circ L_i$. To not have L_i and L_o introduce new trails, the propagation through $F_1 \circ L_i$ and $L_o \circ F_3$ is modelled similar to computing $A^{F_1} A^{L_1} \tilde{\pi}_u$ or $(A^{L_o} A^{F_3})^\top \mu^\circ_v$. For every pair $(\overline{e_i}, e_j)$, an integral property exists if there is no pair (a, b) with $a \in \operatorname{supp} A^{F_1} A^{L_1} \tilde{\pi}_{\overline{e_i}}$ and $b \in \operatorname{supp}(A^{L_o} A^{F_3})^\top \mu^\circ_{e_j}$ such that $\mathcal{A}(F_2, a, b) = 1$.

With perfect caching of the calls to division trail model of F_2, the computational cost is mainly determined by the matrix-vector multiplications that have to be performed for every L_i and L_o. To make this computation feasible, some optimizations are applied. In particular, if the input set $X \in \{\pi_{\overline{e}_0}, \dots, \pi_{\overline{e}_{n-1}}\}$ and $r \in \{\mu_{\overline{e}_0}, \dots, \mu_{\overline{e}_{n-1}}\}$ then only 2^n functions L_i and 2^m functions L_o have to be checked. If F_1 and F_3, are of the form $G_1(x)\| \cdots \|G_b(x)$, respectively $H_1(x)\| \cdots \|H_c(x)$, then the linear functions can be restricted to each of the functions G_i. Assuming all G_i have the same number of input bits, n/b, and all H_i have m/c input bits, then a total number of $n2^{n/b} + m2^{m/c}$ matrix-vector multiplications are performed. Since Algorithm 1 only requires $n + m$ matrix-vector multiplications, the structure of F_1 and F_3 is not a bottleneck. Algorithm 1 is therefore more flexible in the choice of F_1 and F_3, and the input and output exponents (u, v). This makes it possible to choose F_1 and F_3 that consist of more than one round and to look for properties that require less data.

5.2 Application to Present

Present [BKL+07] is a substitution-permutation network with a 64 bit state size. Its round function consists of a full-state key-addition followed by a substitution layer consisting of 16 parallel 4-bit S-boxes of degree 3 and a bit permutation. We analyze Present with independent round keys reduced to R rounds but including the key-addition of round $R+1$. Integral properties with a constant sum are known for up to 9 rounds of Present. The lowest data constant sum property was found using bit-based division property and requires 2^{60} data [XZBL16]. Furthermore, Lambin et al. [LDF20] note that even with the linearly equivalent S-boxes method, they could not find any constant sum integral properties on 10 rounds of Present.

Algorithm 1 was applied to both 9 and 10 rounds of Present. In both cases, the algorithm was run 9 times, once for each combination of F_1 and F_3, consisting of 1 to 3 rounds. The algorithm was instantiated with the oracle from Remark 3, which is itself based on the method described at the end of Section 4.1 with $N_1 = N_2 = 1024$. V was chosen to remove trivial properties for permutations, $V = \mathbb{F}_2^{64} \setminus \{\underline{1}\} \times \mathbb{F}_2^{64} \setminus \{\underline{0}\}$. For 9 rounds of Present, a subspace of constant generalized integral properties of dimension 470 was found. This includes the 2^{60} data property of [XZBL16], but no lower data properties.

Table 1: Statistics on experiments on 9 rounds of Present. Each column of the table corresponds to a choice of F_1, F_2 and F_3. Runtime is given in minutes and the number of oracle calls in millions (M) or billions (B).

	$1,7,1$	$2,6,1$	$1,6,2$	$2,5,2$	$3,5,1$	$3,4,2$	$1,5,3$	$2,4,3$
runtime	138 min	190 min	229 min	378 min	52 min	396 min	45 min	296 min
$\#\mathcal{A}$ calls	2.67 M	22.9 M	19.1 M	705 M	159 M	220 B	119 M	141 B
dimension	455	425	420	401	338	331	338	331

Dividing out the space of properties that could be found with existing methods, such as parity sets [BC16], monomial trails [HSWW20] and linearly-equivalent S-boxes [LDF20], or that are a linear combination thereof, results in a quotient space of dimension 22. This space contains all newly discovered properties. A subspace of dimension 14 contains all properties that could be described, but not found, by [LDF20]. For example, Equation (20) gives a second property with 2^{60} data and with the same input set as the property from [XZBL16]. Combining these properties results in a new distinguisher for 9 rounds of Present with 2^{60} data and an improved advantage of $1-1/4$ instead of $1-1/2$. Dividing out this subspace of dimension 14 gives a final quotient space of dimension 8 containing generalized integral properties. An example of such a property is given in Equation (21), and a basis of the dimension 22 quotient space is given in Appendix C. No properties were found for 10 rounds of Present.

$$\sum_{x \preceq (0,0,0,0,1,\ldots,1)} F^{e_5}(x) + F^{e_{13}}(x) = c, \text{ where } c \text{ is independent of the key.} \tag{20}$$

$$\sum_{x \preceq \overline{e}_5} F^{e_5}(x) + \sum_{x \preceq \overline{e}_9} F^{e_{13}}(x) = c, \text{ where } c \text{ is independent of the key.} \tag{21}$$

In Table 1, statistics on the experiments for 9 rounds of Present are given. All experiments were run on a 40-core Intel(R) Xeon(R) Gold 6230 CPU at 2.10GHz. The runtime of the algorithm increases when the number of rounds in F_1 or F_3 increases, except for the cases of $(3,5,1)$ and $(1,5,3)$. This trend in the runtime can be explained by two phenomena. First, as the number of rounds in F_1 and F_3 increases, $A^{F_1} \otimes (A^{F_3})^\intercal$ becomes denser and the number of calls to the oracle increases. This can also be seen in the second row of Table 1. Second, each oracle call becomes easier to evaluate because, with fewer rounds in F_2, the corresponding SAT models are simpler. As predicted, the dimension of the resulting space decreases with increasing number of rounds, but new properties are still found. Note that with our oracle implementation, up to two rounds for F_1 and F_3 suffices, as the other experiments did not result in new properties.

Acknowledgements. This work is based on Michiel Verbauwhede's master thesis "Tools for Integral Cryptanalysis" [Ver22]. He wants to thank his daily supervisors Tim Beyne and Chaoyun Li and his supervisors, Bart Preneel and Vincent Rijmen. We are also grateful to Vincent Rijmen for his comments on an early draft of this paper. Tim Beyne is supported by a junior postdoctoral fellowship from the Research Foundation – Flanders (FWO) with reference number 1274724N. This work was partially supported by the Research Council KU Leuven, C16/18/004 through the C1 on New Block Cipher Structures. In addition, this work was supported by CyberSecurity Research Flanders with reference number VR20192203.

References

[ARS⁺15] Martin R. Albrecht, Christian Rechberger, Thomas Schneider, Tyge Tiessen, and Michael Zohner. Ciphers for MPC and FHE. In Elisabeth Oswald and

Marc Fischlin, editors, *EUROCRYPT 2015, Part I*, volume 9056 of *LNCS*, pages 430–454. Springer, Heidelberg, April 2015.

[BC16] Christina Boura and Anne Canteaut. Another view of the division property. In Matthew Robshaw and Jonathan Katz, editors, *CRYPTO 2016, Part I*, volume 9814 of *LNCS*, pages 654–682. Springer, Heidelberg, August 2016.

[Bey18] Tim Beyne. Block cipher invariants as eigenvectors of correlation matrices. In Thomas Peyrin and Steven Galbraith, editors, *ASIACRYPT 2018, Part I*, volume 11272 of *LNCS*, pages 3–31. Springer, Heidelberg, December 2018.

[Bey21] Tim Beyne. A geometric approach to linear cryptanalysis. In Mehdi Tibouchi and Huaxiong Wang, editors, *ASIACRYPT 2021, Part I*, volume 13090 of *LNCS*, pages 36–66. Springer, Heidelberg, December 2021.

[BKL+07] Andrey Bogdanov, Lars R. Knudsen, Gregor Leander, Christof Paar, Axel Poschmann, Matthew J. B. Robshaw, Yannick Seurin, and C. Vikkelsoe. PRESENT: An ultra-lightweight block cipher. In Pascal Paillier and Ingrid Verbauwhede, editors, *CHES 2007*, volume 4727 of *LNCS*, pages 450–466. Springer, Heidelberg, September 2007.

[BR22] Tim Beyne and Vincent Rijmen. Differential cryptanalysis in the fixed-key model. In Yevgeniy Dodis and Thomas Shrimpton, editors, *CRYPTO 2022, Part III*, volume 13509 of *LNCS*, pages 687–716. Springer, Heidelberg, August 2022.

[Car20] Claude Carlet, editor. *Boolean Functions for Cryptography and Coding Theory.* Cambridge University Press, 2020.

[DF20] Patrick Derbez and Pierre-Alain Fouque. Increasing precision of division property. *IACR Trans. Symm. Cryptol.*, 2020(4):173–194, 2020.

[DGV95] Joan Daemen, René Govaerts, and Joos Vandewalle. Correlation matrices. In Bart Preneel, editor, *FSE'94*, volume 1008 of *LNCS*, pages 275–285. Springer, Heidelberg, December 1995.

[DKR97] Joan Daemen, Lars R. Knudsen, and Vincent Rijmen. The block cipher Square. In Eli Biham, editor, *FSE'97*, volume 1267 of *LNCS*, pages 149–165. Springer, Heidelberg, January 1997.

[DL22] Patrick Derbez and Baptiste Lambin. Fast MILP models for division property. *IACR Trans. Symm. Cryptol.*, 2022(2):289–321, 2022.

[DS09] Itai Dinur and Adi Shamir. Cube attacks on tweakable black box polynomials. In Antoine Joux, editor, *EUROCRYPT 2009*, volume 5479 of *LNCS*, pages 278–299. Springer, Heidelberg, April 2009.

[Dum04] David Steven Dummit. *Abstract algebra.* Wiley, Hoboken, NJ, 3rd ed. edition, 2004.

[EKKT19] Zahra Eskandari, Andreas Brasen Kidmose, Stefan Kölbl, and Tyge Tiessen. Finding integral distinguishers with ease. In Carlos Cid and Michael J. Jacobson Jr:, editors, *SAC 2018*, volume 11349 of *LNCS*, pages 115–138. Springer, Heidelberg, August 2019.

[HLLT20] Phil Hebborn, Baptiste Lambin, Gregor Leander, and Yosuke Todo. Lower bounds on the degree of block ciphers. In Shiho Moriai and Huaxiong Wang, editors, *ASIACRYPT 2020, Part I*, volume 12491 of *LNCS*, pages 537–566. Springer, Heidelberg, December 2020.

[HLLT21] Phil Hebborn, Baptiste Lambin, Gregor Leander, and Yosuke Todo. Strong and tight security guarantees against integral distinguishers. In Mehdi Tibouchi and Huaxiong Wang, editors, *ASIACRYPT 2021, Part I*, volume 13090 of *LNCS*, pages 362–391. Springer, Heidelberg, December 2021.

[HLM$^+$21] Yonglin Hao, Gregor Leander, Willi Meier, Yosuke Todo, and Qingju Wang. Modeling for three-subset division property without unknown subset. *Journal of Cryptology*, 34(3):22, July 2021.

[HSWW20] Kai Hu, Siwei Sun, Meiqin Wang, and Qingju Wang. An algebraic formulation of the division property: Revisiting degree evaluations, cube attacks, and key-independent sums. In Shiho Moriai and Huaxiong Wang, editors, *ASIACRYPT 2020, Part I*, volume 12491 of *LNCS*, pages 446–476. Springer, Heidelberg, December 2020.

[IMM18] Alexey Ignatiev, Antonio Morgado, and Joao Marques-Silva. PySAT: A Python toolkit for prototyping with SAT oracles. In *SAT*, pages 428–437, 2018.

[Knu95] Lars R. Knudsen. Truncated and higher order differentials. In Bart Preneel, editor, *FSE'94*, volume 1008 of *LNCS*, pages 196–211. Springer, Heidelberg, December 1995.

[KW02] Lars R. Knudsen and David Wagner. Integral cryptanalysis. In Joan Daemen and Vincent Rijmen, editors, *FSE 2002*, volume 2365 of *LNCS*, pages 112–127. Springer, Heidelberg, February 2002.

[LDF20] Baptiste Lambin, Patrick Derbez, and Pierre-Alain Fouque. Linearly equivalent s-boxes and the division property. *Des. Codes Cryptogr.*, 88(10):2207–2231, 2020.

[PF23] Laurent Perron and Vincent Furnon. OR-Tools, 2023.

[PWZ11] Josef Pieprzyk, Huaxiong Wang, and Xian-Mo Zhang. Möbius transforms, coincident boolean functions and non-coincidence property of boolean functions. *Int. J. Comput. Math.*, 88(7):1398–1416, 2011.

[Rob70] DW Robinson. Generalized jordan canonical form. *The American mathematical monthly*, 77(4):392–395, 1970.

[Rot64] Gian Carlo Rota. On the foundations of combinatorial theory i. theory of möbius functions. *Zeitschrift für Wahrscheinlichkeitstheorie und Verwandte Gebiete*, 2(4):340–368, Jan 1964.

[TLS16] Yosuke Todo, Gregor Leander, and Yu Sasaki. Nonlinear invariant attack - practical attack on full SCREAM, iSCREAM, and Midori64. In Jung Hee Cheon and Tsuyoshi Takagi, editors, *ASIACRYPT 2016, Part II*, volume 10032 of *LNCS*, pages 3–33. Springer, Heidelberg, December 2016.

[TM16] Yosuke Todo and Masakatu Morii. Bit-based division property and application to simon family. In Thomas Peyrin, editor, *FSE 2016*, volume 9783 of *LNCS*, pages 357–377. Springer, Heidelberg, March 2016.

[Tod15] Yosuke Todo. Structural evaluation by generalized integral property. In Elisabeth Oswald and Marc Fischlin, editors, *EUROCRYPT 2015, Part I*, volume 9056 of *LNCS*, pages 287–314. Springer, Heidelberg, April 2015.

[Udo21] Aleksei Udovenko. Convexity of division property transitions: Theory, algorithms and compact models. In Mehdi Tibouchi and Huaxiong Wang, editors, *ASIACRYPT 2021, Part I*, volume 13090 of *LNCS*, pages 332–361. Springer, Heidelberg, December 2021.

[Ver22] Michiel Verbauwhede. Tools for integral cryptanalysis. Master's thesis, KU Leuven, 2022.

[Vie07] Michael Vielhaber. Breaking ONE.FIVIUM by AIDA an algebraic IV differential attack. Cryptology ePrint Archive, Report 2007/413, 2007. https://eprint.iacr.org/2007/413.

[XZBL16] Zejun Xiang, Wentao Zhang, Zhenzhen Bao, and Dongdai Lin. Applying MILP method to searching integral distinguishers based on division property for 6 lightweight block ciphers. In Jung Hee Cheon and Tsuyoshi Takagi, editors, *ASIACRYPT 2016, Part I*, volume 10031 of *LNCS*, pages 648–678. Springer, Heidelberg, December 2016.

[ZRHD08] Muhammad Reza Z'aba, Håvard Raddum, Matthew Henricksen, and Ed Dawson. Bit-pattern based integral attack. In Kaisa Nyberg, editor, *FSE 2008*, volume 5086 of *LNCS*, pages 363–381. Springer, Heidelberg, February 2008.

A Algebraic Transition Matrix of PRESENT S-box

The PRESENT S-box is defined by the following permutation:

$$\begin{pmatrix} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & a & b & c & d & e & f \\ c & 5 & 6 & b & 9 & 0 & a & d & 3 & e & f & 8 & 4 & 7 & 1 & 2 \end{pmatrix},$$

and its algebraic transition matrix is equal to:

$$\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\
1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 0 & 0 \\
0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 \\
1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1
\end{bmatrix}.$$

B Invariants of the PRESENT S-box

The PRESENT S-box is equal to $F = (0, c, 4, 9, e, 1, 5) \circ (2, 6, a, f) \circ (3, b, 8) \circ (7, d)$ in cycle notation. The elementary divisors of T^F are $x + 1$, $x + 1$, $(x + 1)^2$, $(x + 1)^4$, $x^2 + x + 1$,

$x^3 + x + 1$ and $x^3 + x^2 + 1$. The normal form after the primary and Jordan decomposition is

$$D = \left[\begin{array}{c|c|cc|cccc|cc|ccc|ccc}
1 & & & & & & & & & & & & & & & \\ \hline
 & 1 & & & & & & & & & & & & & & \\ \hline
 & & 1 & 1 & & & & & & & & & & & & \\
 & & 0 & 1 & & & & & & & & & & & & \\ \hline
 & & & & 1 & 1 & 0 & 0 & & & & & & & & \\
 & & & & 0 & 1 & 1 & 0 & & & & & & & & \\
 & & & & 0 & 0 & 1 & 1 & & & & & & & & \\
 & & & & 0 & 0 & 0 & 1 & & & & & & & & \\ \hline
 & & & & & & & & 0 & 1 & & & & & & \\
 & & & & & & & & 1 & 1 & & & & & & \\ \hline
 & & & & & & & & & & 0 & 1 & 0 & & & \\
 & & & & & & & & & & 0 & 0 & 1 & & & \\
 & & & & & & & & & & 1 & 1 & 0 & & & \\ \hline
 & & & & & & & & & & & & & 0 & 1 & 0 \\
 & & & & & & & & & & & & & 0 & 0 & 1 \\
 & & & & & & & & & & & & & 1 & 0 & 1
\end{array}\right]$$

and the transformation matrix is

$$P = \left[\begin{array}{c|c|c|c|c|c|c|c|cc|ccc|ccc}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{array}\right].$$

Similar to Example 3, the invariants are related to the cycles of the S-box. The first, second, third and fifth column of P are the invariants resulting from saturating the 7-cycle, 3-cycle, 2-cycle and 4-cycle respectively. The third and fourth column combined are the invariant resulting from selecting a single element of the 2-cycle. The fifth, sixth, seventh and eighth column are a Jordan chain. Growing from left to right, these invariants result from the saturation of the 4-cycle, selecting every second element of the 4-cycle, selecting two adjacent elements of the 4-cycle and selecting a single element of the 4-cycle. The ninth and tenth column are an invariant related to selecting two elements of the 3-cycle. The eleventh through 13th and 14th through 16th column are two invariants related to selecting four elements from the 7-cycle.

C Subspace of New Properties on 9-round PRESENT

Simple Properties on Linearly Equivalent Cipher

Let $x = (x_1, \dots, x_{64}) \in \mathbb{F}_2^{64}$.

$$\sum_{x_5=0} F^{e_5}(x) + F^{e_{13}}(x) = c.$$

$$\sum_{x_9=0} F^{e_5}(x) + F^{e_{13}}(x) = c.$$

$$\sum_{x_{13}=0} F^{e_5}(x) + F^{e_{13}}(x) = c.$$

$$\sum_{x_6+x_7=0} F^{e_5}(x) + F^{e_{13}}(x) = c.$$

$$\sum_{x_{10}+x_{11}=0} F^{e_5}(x) + F^{e_{13}}(x) = c.$$

$$\sum_{x_{14}+x_{15}=0} F^{e_5}(x) + F^{e_{13}}(x) = c.$$

$$\sum_{x_1x_2x_3x_4=0} F^{e_5}(x) + F^{e_{13}}(x) = c.$$

$$\sum_{x_5+x_9=0} F^{e_5}(x) = c.$$

$$\sum_{x_5+x_9=0} F^{e_9}(x) = c.$$

$$\sum_{x_5+x_9=0} F^{e_{13}}(x) = c.$$

$$\sum_{x_5+x_9=0} F^{e_{21}}(x) + F^{e_{53}}(x) = c.$$

$$\sum_{x_5+x_9=0} F^{e_{25}}(x) + F^{e_{57}}(x) = c.$$

$$\sum_{x_5+x_9=0} F^{e_{29}}(x) + F^{e_{61}}(x) = c.$$

$$\sum_{x_5+x_9=0} F^{e_1}(x)F^{e_{17}}(x)F^{e_{33}}(x)F^{e_{49}}(x) = c.$$

Remaining Generalized Integral Properties

$$\sum_{x_{14}=0} F^{e_{17}}(x) + \sum_{x_{15}=0} F^{e_{49}}(x) = c.$$

$$\sum_{x_{10}=0} F^{e_{17}}(x) + \sum_{x_{11}=0} F^{e_{49}}(x) = c.$$

$$\sum_{x_{6}=0} F^{e_{17}}(x) + \sum_{x_{7}=0} F^{e_{49}}(x) = c.$$

$$\sum_{x_{2}=0} F^{e_{17}}(x) + \sum_{x_{3}=0} F^{e_{49}}(x) = c.$$

$$\sum_{x_{2}=0} F^{e_{21}}(x) + \sum_{x_{3}=0} F^{e_{53}}(x) = c.$$

$$\sum_{x_{2}=0} F^{e_{25}}(x) + \sum_{x_{3}=0} F^{e_{57}}(x) = c.$$

$$\sum_{x_{2}=0} F^{e_{29}}(x) + \sum_{x_{3}=0} F^{e_{61}}(x) = c.$$

$$\sum_{x_{5}=0} F^{e_{5}}(x) + \sum_{x_{9}=0} F^{e_{13}}(x) = c.$$