Helsinki University of Technology Laboratory for Theoretical Computer Science

Research Reports 84

Teknillisen korkeakoulun tietojenkäsittelyteorian laboratorion tutkimusraportti 84

Espoo 2003 HUT-TCS-A84

ON THE DIFFERENTIAL AND LINEAR PROPERTIES OF ADDITION

Johan Wallén

TEKNILLINEN KORKEAKOULU
TEKNISKA HÖGSKOLAN
HELSINKI UNIVERSITY OF TECHNOLOGY
TECHNISCHE UNIVERSITÄT HELSINKI
UNIVERSITE DE TECHNOLOGIE D'HELSINKI

Helsinki University of Technology Laboratory for Theoretical Computer Science

Research Reports 84

Teknillisen korkeakoulun tietojenkäsittelyteorian laboratorion tutkimusraportti 84

Espoo 2003 HUT-TCS-A84

ON THE DIFFERENTIAL AND LINEAR PROPERTIES OF ADDITION

Johan Wallén

Helsinki University of Technology
Department of Computer Science and Engineering
Laboratory for Theoretical Computer Science

Teknillinen korkeakoulu
Tietotekniikan osasto
Tietojenkäsittelyteorian laboratorio

Distribution:
Helsinki University of Technology
Laboratory for Theoretical Computer Science
P.O.Box 5400
FIN-02015 HUT
Tel. +358-0-451 1
Fax. +358-0-451 3369
E-mail: lab@tcs.hut.fi

ISBN 951-22-6880-9
ISSN 1457-7615

Multiprint Oy
Helsinki 2003

ABSTRACT: We present a detailed analysis of some of the fundamental differential and linear properties of addition modulo 2^n: the differential probability xdp^+ of addition modulo 2^n when differences are expressed using exclusive-or, the dual differential probability $\mathrm{adp}^\oplus$ of exclusive-or when differences are expressed using addition modulo 2^n and the correlation lca of $\mathbf{F}_2$-linear approximations of addition modulo 2^n. We show that xdp^+, $\mathrm{adp}^\oplus$ and lca can be viewed as formal rational series with linear representations in base 8. For xdp^+ and lca, the linear representations give $\Theta(\log n)$-time algorithms for computing xdp^+ and lca, explicit descriptions of all differentials or linear approximations with a given probability or correlation, and allows us to determine the distributions of xdp^+ and lca. For $\mathrm{adp}^\oplus$, the linear representation immediately gives a linear-time algorithm for computing $\mathrm{adp}^\oplus$. We analyse the asymptotic average behaviour of $\mathrm{adp}^\oplus$. In particular, we derive a Fourier representation of a first-order summation function obtained by interpreting differentials as integers in a natural way.

KEYWORDS: Differential cryptanalysis, linear cryptanalysis, arithmetic operations, rational series

CONTENTS

LIST OF TABLES

NOTATION

$\alpha \xrightarrow{f} \beta$	Differential	2
$\Pr[\alpha \xrightarrow{f} \beta]$	Differential probability	2
$\mathrm{c}(f, g)$	Correlation of Boolean functions	5
l_w	Linear function $x \mapsto w \cdot x$	5
$u \xleftarrow{f} w$	Linear approximation	5
$\mathcal{C}(u \xleftarrow{f} w)$	Correlation of linear approximation	5
$\hat{f}$	Real-valued version of Boolean function	6
$\mathcal{B}_n$	Real algebra of Boolean functions	6
$\mathcal{F}$	Fourier transform	7
$\xi * \eta$	Convolution	7
δ	Delta function	7
xdp^{+}	XOR differential probability of addition	12
carry	Carry function	13
xdp^{-}	XOR differential probability of subtraction	17
borrow	Borrow function	17
$\mathrm{xdp}^{\mathrm{pht}}$	XOR differential probability of the PHT	18
lca	Linear correlation of addition	20
lcc	Linear correlation of carry	20
lcs	Linear correlation of addition	20
$\mathrm{adp}^{\oplus}$	Additive differential probability of XOR	30
$\mathrm{sbs}(n)$	The sequence obtained by putting $\mathrm{adp}^{\oplus}$ side-by-side	41

1 DIFFERENTIAL AND LINEAR CRYPTANALYSIS

Differential [8, 9] and linear [34, 35] cryptanalysis are two of the most powerful general cryptanalytic methods for symmetric ciphers proposed by date. Since their introduction, resistance against these attacks has been a standard design goal for ciphers. Although some design methodologies to achieve this goal have been proposed—for example [40, 39, 37, 17, 50]—many ciphers are still designed in a rather ad hoc manner, or dictated by other primary design goals. For these ciphers, it it important to have efficient methods for evaluating their resistance against differential and linear cryptanalysis.

At the heart of differential cryptanalysis lies the study of difference propagations through functions, whereas linear cryptanalysis is based on linear approximate relations between the input and output of functions. Good differentials and linear approximations for ciphers are usually found heuristically, or their strength is upper bounded [26, 42], by forming trails of differentials or linear approximations of the components of the cipher. To evaluate the security of a cipher against these cryptanalytic attacks, we thus need detailed analyses of the differential and linear properties of the components of the cipher. In order to search the space of trails, e.g. using a Branch-and-bound algorithm, we especially need efficient methods for generating the relevant differentials or linear approximations of the components. Towards this goal, we study a few basic functions often used as building blocks in symmetric ciphers.

Currently, symmetric primitives like block ciphers are usually build from local nonlinear mappings (usually called S-boxes), global linear mappings and arithmetic operations. The mixture of linear mappings and arithmetic operations seems fruitful, since they are suitable for software implementation, and their mixture is difficult to analyse mathematically. While the latter property intuitively should make standard cryptanalysis intractable, it also makes it difficult to say something concrete about the security of the cipher.

Perhaps the simplest arithmetic operation in wide use is addition modulo 2^n used e.g. in the block ciphers [32, 44, 47, 33], the stream ciphers [45, 19, 20] and the hash functions [41]. Surprisingly little is known about how this arithmetic operation interacts with $\mathbf{F}_2$-linear mappings in concrete cryptographic settings like differential and linear cryptanalysis. Even some of the most basic differential and linear properties of addition have been established only very recently [30, 29, 52]. When applying differential cryptanalysis to a cipher that uses both addition modulo 2^n and bitwise exclusive-or, both operations are natural choices for the difference operator (depending on how the round keys are added). Depending on this choice, we must study either the differential properties of addition when differences are expressed using exclusive-or, or the differential properties of exclusive-or when differences are expressed using addition modulo 2^n. Similarly, when applying linear cryptanalysis to these kinds of ciphers, one needs good tools for studying linear approximations of addition modulo 2^n.

In this thesis, we present a unifying framework for studying the differential and linear properties of addition. This framework gives straightforward proofs of the results in [30, 52] as well as generalisations to more complex functions. Within the same framework, we also study the differential properties of bitwise exclusive-or when differences are expressed using addition modulo 2^n. We hope that our results will facilitate advanced differential and linear cryptanalysis of ciphers using arithmetic operations.

In the rest of this chapter, we first give a short overview of differential and linear cryptanalysis. A more thorough discussion can be found e.g. in [18]. Readers already familiar with differential and linear cryptanalysis might want to skip directly to Section 1.3, where we define the differential and linear properties of addition studied in this thesis and give an overview of our results.

1.1 DIFFERENTIAL CRYPTANALYSIS

Most conventional cryptanalytic methods can be classified as either *algebraic* or *statistical cryptanalysis*. In algebraic attacks, the analyst tries to expresses the cipher as a relatively simple system of algebraic equations and then solve it. Algebraic attacks have achieved some attention recently [16, 15, 4, 14]. In statistical attacks, the analyst tries to find some form of exploitable statistical correlation between the input and output of a part of the cipher. This correlation can then e.g. be used to distinguish the cipher from an ideal one or to launch a key recovery attack. One of the best known examples of statistical cryptanalysis is *differential cryptanalysis*.

Differential cryptanalysis [9] is a chosen-plaintext attack that studies the propagation of input differences to output differences in iterated transformations. These difference propagations are formalised as follows. We will consider a generic iterated cipher $f_r \circ \cdots \circ f_1$, where each f_i may depend on an unknown key. Each function f_i is modelled as a function between arbitrary finite Abelian groups. Let G, H be Abelian groups, let $f\colon G \to H$ be a function, and let $a, a^* \in G$. The *difference* $a' = a - a^*$ is said to *propagate* to the difference $b' = f(a) - f(a^*)$ trough f. This is denoted by $a' \xrightarrow{f} b'$, or simply $a' \to b'$ when f is clear from context. A *differential* is a pair $(\alpha, \beta) \in G \times H$, usually denoted by $\alpha \xrightarrow{f} \beta$ or $\alpha \to \beta$

If the input difference of a pair is α, the differential $\alpha \to \beta$ can be used to predict the corresponding output difference. It is thus natural to measure the efficiency of a differential as the fraction of all input pairs with difference α that result in the output difference β. This is called the *differential probability* of $\alpha \xrightarrow{f} \beta$ and is defined by

$$\Pr[\alpha \xrightarrow{f} \beta] = \frac{1}{|G|} \left|\{x \in G \mid f(x+\alpha) - f(x) = \beta\}\right| \quad .$$

A generic differential attack against an r round iterated block cipher is the following. Suppose that $\alpha \xrightarrow{f_{r-1} \cdots f_1} \beta$ is an $r-1$ round differential for the cipher with high enough probability p. If a uniformly chosen pair of plaintexts (x, x^*) with difference α is encrypted, the difference between the intermediate values after $r-1$ rounds is β with probability p. If p is relatively large, the other intermediate differences after $r-1$ rounds are usually assumed to have significantly lower probability and be fairly evenly distributed. If the intermediate difference indeed is β, (x, x^*) is said to be a *right pair*. Otherwise, it is said to be a *wrong pair*. Assuming that (x, x^*) is a right pair, one can check which round keys at the last round are compatible with the *assumed* input difference β to the last round and the *observed* ciphertext pair (y, y^*). Typically, this only puts restrictions on some part of the round key. We call this part of the key the *round subkey*. The pair (x, x^*) is said to *suggest* all these compatible subkey values. Note that a right pair always suggests the correct subkey value (and some incorrect ones), while a wrong pair might not suggest the right value at all. Assuming that incorrect subkey values suggested both by right and wrong pairs are fairly evenly distributed, one would expect that the correct subkey value is suggested more often than the incorrect ones. If L denotes the number of possible round subkeys and γ denotes the average number of suggested subkeys per pair, the *signal-to-noise* S/N ratio is defined by

$$S/N = \frac{\Pr[\alpha \to \beta]}{\gamma} L \quad .$$

Experimental results for DES [9] indicate that only a few right pairs are needed if the S/N ratio is significantly above 1. A S/N ratio significantly below 1 makes the required number of right pairs too large for a practical attack.

This gives the following chosen-plaintext key-recovery attack using an $r-1$ round differential $\alpha \to \beta$ with sufficiently high probability:

1. Keep a counter for each possible round subkey k_r at round r. Initialise the counters to zero.

2. Pick a plaintext x uniformly at random and set $x^* = x + \alpha$. Obtain the corresponding ciphertexts y, y^* (two chosen plaintexts). For each possible round subkey k_r compatible with the assumed input difference β and the observed outputs y, y^* at round r, add one to the corresponding counter.

3. Repeat step 2 until some round subkeys are counted significantly more often than the others. Output these keys as the most likely candidates for the actual subkey at the last round.

This basic attack can be improved by filtering out obviously wrong pairs. There are several other improvements that reduce the number of plaintexts needed. The attack can also be carried out using a $r-2$ round differential and counting on the round subkeys of the last two rounds. See [9] for further details. There are also other types of differential attacks [28, 27, 7, 51].

Under a suitable set of assumptions, it can be shown that the correct round key can be distinguished from a randomly selected key with sufficient confidence, provided that the number of plaintexts available is inversely proportional to the probability of the used differential. It follows that a necessary condition for resistance against conventional differential attacks for block ciphers is that there does not exist any differential ranging over all but a few rounds with probability significantly larger than 2^{-n}, where n is the block size. A more detailed analysis of differential distinguishers can be found in [25].

1.1.1 Differential Trails

In practice, good differentials ranging over several rounds are found heuristically, or their probability is upper bounded, by forming trails of differentials of the components of the cipher. From the differentials $\alpha_i \xrightarrow{f_i} \alpha_{i+1}$, one can form the *differential trail*

$$\alpha_1 \xrightarrow{f_1} \alpha_2 \xrightarrow{f_2} \cdots \xrightarrow{f_r} \alpha_{r+1}$$

for the iterated mapping $f = f_r \circ \cdots \circ f_1$. The probability of a differential trail is defined to be the probability that a pair with input difference α_1 propagate to the intermediate difference α_{i+1} after the ith round for $i = 1, \ldots, r$. Since the events are disjoint,

$$\Pr[\alpha \xrightarrow{f} \beta] = \sum_{\substack{\alpha_1,\ldots,\alpha_{r+1} \\ \alpha_1=\alpha,\alpha_{r+1}=\beta}} \Pr[\alpha_1 \xrightarrow{f_1} \alpha_2 \xrightarrow{f_2} \cdots \xrightarrow{f_r} \alpha_{r+1}] \ . \tag{1.1}$$

If the difference propagations of each round are independent, or only weakly dependent, one can approximate

$$\Pr[\alpha_1 \xrightarrow{f_1} \alpha_2 \xrightarrow{f_2} \cdots \xrightarrow{f_r} \alpha_{r+1}] \approx \prod_{i=1}^{r} \Pr[\alpha_i \xrightarrow{f_i} \alpha_{i+1}] \ .$$

If the sum in (1.1) is dominated by a single differential trail $\alpha_1 \xrightarrow{f_1} \alpha_2 \xrightarrow{f_2} \cdots \xrightarrow{f_r} \alpha_{r+1}$, one can estimate $\Pr[\alpha_1 \xrightarrow{f} \alpha_{r+1}] \approx \prod_{i=1}^{r} \Pr[\alpha_i \xrightarrow{f_i} \alpha_{i+1}]$ for the differential probability, but one should be careful when interpreting this estimate, since many trails might contribute to the same differential, and the probability of trails is usually key dependent. In order to search the space of differential trails, e.g. using a Branch-and-bound algorithm (see e.g. [24, 36, 2]), we need efficient methods for computing the differential probability for the simplest components of the cipher, as well as methods for generating the relevant differentials for the components.

1.1.2 Some Special Functions

The differential probability of some special functions is straightforward to compute. If L is a linear function and f is the affine function $f(x) = L(x) + a$, where a is a constant, we see that $f(x + \alpha) - f(x) = L(\alpha)$ and hence $\Pr[\alpha \xrightarrow{f} \beta] = 1$ if $\beta = L(\alpha)$ and $\Pr[\alpha \xrightarrow{f} \beta] = 0$ otherwise. The effect of a linear or affine function before or after a nonlinear function is simple.

Let $f_i \colon G_i \to H_i$ and define $f \colon G_1 \times \cdots \times G_k \to H_1 \times \cdots \times H_k$ by

$$f(x_1, x_2, \ldots, x_k) = (f_1(x_1), f_2(x_2), \ldots, f_k(x_k)) \ .$$

Since the difference propagations through the components of f are independent, we see that

$$\Pr[(\alpha_1, \ldots, \alpha_k) \xrightarrow{f} (\beta_1, \ldots, \beta_k)] = \prod_{i=1}^{k} \Pr[\alpha_i \xrightarrow{f_i} \beta_i] \ .$$

If the groups G_i are small, $\Pr[\alpha_i \xrightarrow{f_i} \beta_i]$ can be computed directly from its definition. It is thus easy to compute the differential probability for the parallel application of small functions. For arithmetic operations, the situation is more complicated, since the operations are nonlinear and the input domain is large.

1.2 LINEAR CRYPTANALYSIS

Another prime example of statistical cryptanalysis is *linear cryptanalysis* [34]. In the basic version of linear cryptanalysis, we consider a generic iterated cipher $f = f_r \circ \cdots \circ f_1$, where each f_i is a (possibly key dependent) function between vector spaces over $\mathbf{F}_2$, $f_i \colon \mathbf{F}_2^{n_i} \to \mathbf{F}_2^{m_i}$. Linear cryptanalysis views (a part of) the cipher as a relation between the plaintext, the ciphertext and the key, and tries to approximate this relation using linear relations. The following standard terminology is convenient for discussing these linear approximations.

Let $f, g \colon \mathbf{F}_2^n \to \mathbf{F}_2$ be Boolean functions. The *correlation* between f and g is defined by

$$\mathrm{c}(f, g) = 2^{1-n}\bigl|\{x \in \mathbf{F}_2^n \mid f(x) = g(x)\}\bigr| - 1 \ .$$

This is simply the probability taken over x that $f(x) = g(x)$ scaled to a value in $[-1, 1]$. Let $u = (u_{m-1}, \dots, u_0) \in \mathbf{F}_2^m$ and $w = (w_{n-1}, \dots, w_0) \in \mathbf{F}_2^n$ be binary column vectors, and let $h \colon \mathbf{F}_2^n \to \mathbf{F}_2^m$ be a function. Let $w \cdot x = w_{n-1}x_{n-1} + \cdots + w_1x_1 + w_0x_0 \in \mathbf{F}_2$ denote the standard dot product. Define the linear function $l_w \colon \mathbf{F}_2^n \to \mathbf{F}_2$ by $l_w(x) = w \cdot x$ for all $w \in \mathbf{F}_2^n$. A *linear approximation* of h is an approximate relation of the form $u \cdot h(x) = w \cdot x$. Such a linear approximation will be denoted by $u \xleftarrow{h} w$, or simply $u \leftarrow w$ when h is clear from context. Here, u and w are the *output* and *input selection vectors*, respectively. The efficiency of a linear approximation is measured by its *correlation*

$$\mathcal{C}(u \xleftarrow{h} w) = \mathrm{c}(l_u \circ h, l_w) = 2^{1-n}\bigl|\{x \in \mathbf{F}_2^n \mid u \cdot h(x) = w \cdot x\}\bigr| - 1 \ .$$

This is just the probability that the approximation holds scaled to a value in $[-1, 1]$.

In the basic form of linear cryptanalysis of an r round iterated block cipher, the analyst tries to find a linear approximation over $r - 2$ rounds from the second round to the second to last round—that is, an approximation of the form

$$a \cdot X + b \cdot Y = c \cdot k \ , \tag{1.2}$$

where x is the plaintext, $X = f_1(x, k_1)$ is the intermediate value after the first round, Y is the intermediate value before the last round, $y = f_r(Y, k_r)$ is the ciphertext, and $k = (k_2, \dots, k_{r-1})$ is a vector of all the unknown round keys at rounds 2 to $r - 1$. For a fixed key, the right hand side of (1.2) is a constant. Given N known plaintext-ciphertext pairs (x, y), we can find parts of the round keys at the first and last rounds as follows. By guessing parts of the round keys k_1 and k_r, one can compute $a \cdot f_1(x, k_1) + b \cdot f_r^{-1}(y, k_r)$

and count the number of times N_0 that it is zero. We call the parts of the round keys the attacker tries to guess the *round subkeys*. If the key guess is correct, one would expect that $2N_0/N - 1$ is approximately plus or minus the correlation of the approximation (1.2), depending on the value of the constant $c \cdot k$. If the key guess is incorrect, the values $f_1(x, k_1)$ and $f_r^{-1}(y, k_r)$ are probably incorrect and one would expect that the correlation between $a \cdot f_1(x, k_1)$ and $b \cdot f_r^{-1}(y, k_r)$ is zero. Given N known plaintext-ciphertext pairs (x, y), we can thus try all possible round subkeys at rounds 1 and r, and count the number N_0 of plaintext for which

$$a \cdot f_1(x, k_1) + b \cdot f_r^{-1}(y, k_r) = 0$$

holds. The round subkeys that maximise $|N_0/N - 1/2|$ are chosen as the most likely candidates.

In [34], it was shown that the number of known plaintexts needed for the attack above is inversely proportional to the square of the correlation of the used linear approximation. A necessary condition for resistance against conventional linear cryptanalysis for block ciphers is thus that there does not exist any linear approximation ranging over all but a few rounds, such that the square of its correlation is significantly larger that 2^{-n}, where n is the block size. A more detailed analysis of linear distinguishers can be found in [25].

1.2.1 Fourier Analysis

There is a well-known Fourier-based framework for studying linear approximations [13]. Let $f\colon \mathbf{F}_2^n \to \mathbf{F}_2$ be a Boolean function. The corresponding real-valued function $\hat{f}\colon \mathbf{F}_2^n \to \mathbf{R}$ is defined by $\hat{f}(x) = (-1)^{f(x)}$. With this notation, $\mathrm{c}(f, g) = 2^{-n} \sum_{x \in \mathbf{F}_2^n} \hat{f}(x)\hat{g}(x)$. Note also that the real-valued function corresponding to $x \mapsto f(x) + g(x)$ is $x \mapsto \hat{f}(x)\hat{g}(x)$. Recall that an algebra $\mathcal{A}$ over a field $\mathbf{F}$ is a ring, such that $\mathcal{A}$ is a vector space over $\mathbf{F}$, and $a(xy) = (ax)y = x(ay)$ for all $a \in \mathbf{F}$ and $x, y \in \mathcal{A}$. We let $\mathcal{B}_n = \langle \hat{f} \mid f\colon \mathbf{F}_2^n \to \mathbf{F}_2 \rangle$ be the real algebra generated by the n-variable Boolean functions. As usual, the addition, multiplication, and multiplication by scalars are given by $(\xi + \eta)(x) = \xi(x) + \eta(x)$, $(\xi\eta)(x) = \xi(x)\eta(x)$ and $(a\xi)(x) = a(\xi(x))$ for all $\xi, \eta \in \mathcal{B}_n$ and $a \in \mathbf{R}$. The algebra $\mathcal{B}_n$ is of course unital and commutative.

The vector space $\mathcal{B}_n$ is turned into an inner-product space by adopting the standard inner-product for real-valued discrete functions. This inner-product is defined by

$$(\xi, \eta) = 2^{-n} \sum_{x \in \mathbf{F}_2^n} (\xi\eta)(x) \quad , \quad \forall \xi, \eta \in \mathcal{B}_n \quad .$$

For Boolean functions, $f, g\colon \mathbf{F}_2^n \to \mathbf{F}_2$, $(\hat{f}, \hat{g}) = \mathrm{c}(f, g)$. It is easy to see that the set of linear functions $\{\hat{l}_w \mid w \in \mathbf{F}_2^n\}$ forms an orthonormal basis for $\mathcal{B}_n$. Thus, every $\xi \in \mathcal{B}_n$ has a unique representation as

$$\xi = \sum_{w \in \mathbf{F}_2^n} \alpha_w \hat{l}_w \ , \text{ where } \quad \alpha_w = (\xi, \hat{l}_w) \in \mathbf{R} \quad .$$

The corresponding Fourier transform $\mathcal{F}\colon \mathcal{B}_n \to \mathcal{B}_n$ is given by

$$\mathcal{F}(\xi) = \Xi \ , \quad \text{where } \Xi \text{ is the mapping} \quad w \mapsto (\xi, \hat{l}_w) \ .$$

This is often called the *Walsh-Hadamard transform* of ξ. For a Boolean function $f\colon \mathbf{F}_2^n \to \mathbf{F}_2$, the Fourier transform $\hat{F} = \mathcal{F}(\hat{f})$ simply gives the correlation between f and the linear functions: $\hat{F}(w) = \mathrm{c}(f, l_w)$.

For $\xi, \eta \in \mathcal{B}_n$, their *convolution* $\xi * \eta \in \mathcal{B}_n$ is given by

$$(\xi * \eta)(x) = \sum_{t \in \mathbf{F}_2^n} \xi(x+t)\eta(t) \ .$$

Clearly, the vector space $\mathcal{B}_n$ is a commutative, unital real algebra also when convolution is the multiplicative operation. The unity with respect to convolution is the function δ such that $\delta(0) = 1$ and $\delta(x) = 0$ for $x \neq 0$. As usual, the Fourier transform is an algebra isomorphism between the commutative, unital real algebras $\langle \mathcal{B}_n, +, \cdot \rangle$ and $\langle \mathcal{B}_n, +, * \rangle$.

Let $f\colon \mathbf{F}_2^n \to \mathbf{F}_2^m$ be a Boolean function. Since the correlation of a linear approximation of f is given by $\mathcal{C}(u \xleftarrow{f} w) = \mathcal{F}(\widehat{l_u f})(w)$, the correlation of linear approximations can conveniently be studied using the Fourier transform. Since $l_u f$ can be expressed as $\sum_{i : u_i = 1} f_i$, where f_i denotes the ith component of f, $\mathcal{C}(u \xleftarrow{f} w)$ is given by the convolution of the $\mathcal{F}(\hat{f}_i)$ for which $u_i = 1$. Especially when using this convolutional representation, it will be convenient to consider $\mathcal{C}(u \xleftarrow{f} w)$ as a function of w with u fixed.

1.2.2 Linear Trails

In practice, good linear approximations ranging over more than one round are found heuristically, or their correlation is upper bounded, by forming trails of linear approximations of the components of the cipher. From the approximations $\xi_{i+1} \xleftarrow{f_i} \xi_i$, one can form the *linear trail*

$$\xi_{r+1} \xleftarrow{f_r} \xi_r \xleftarrow{f_{r-1}} \cdots \xleftarrow{f_1} \xi_1$$

for the iterated mapping $f = f_r \circ \cdots \circ f_1$. The correlation of a linear trail is defined to be

$$\mathcal{C}(\xi_{r+1} \xleftarrow{f_r} \xi_r \xleftarrow{f_{r-1}} \cdots \xleftarrow{f_1} \xi_1) = \prod_{i=1}^{r} \mathcal{C}(\xi_{i+1} \xleftarrow{f_i} \xi_i) \ .$$

Let $g\colon \mathbf{F}_2^n \to \mathbf{F}_2^\ell$ and $h\colon \mathbf{F}_2^\ell \to \mathbf{F}_2^m$. Using the Fourier-based framework, it is easy to show that

$$\mathcal{C}(u \xleftarrow{h \circ g} w) = \sum_{v \in \mathbf{F}_2^\ell} \mathcal{C}(u \xleftarrow{h} v)\mathcal{C}(v \xleftarrow{g} w) \ .$$

If we let C^g be the real $2^\ell \times 2^n$ matrix whose element in row v and column w is $C^g_{vw} = \mathcal{C}(v \xleftarrow{g} w)$, we see that $C^{h \circ g} = C^h C^g$. These *correlation matrices* are discussed in [17]. In this way the Fourier transform induces a group monomorphism (injective homomorphism) from the group of permutations $\mathbf{F}_2^n \to \mathbf{F}_2^n$ to the general linear group $\mathrm{GL}_n(\mathbf{R})$. For iterated mappings, it follows that

$$\mathcal{C}(u \xleftarrow{f} w) = \sum_{\substack{\xi_1,\ldots,\xi_{r+1} \\ \xi_1=w,\xi_{r+1}=u}} \mathcal{C}(\xi_{r+1} \xleftarrow{f_r} \xi_r \xleftarrow{f_{r-1}} \cdots \xleftarrow{f_1} \xi_1) \ .$$

If the sum is dominated by a single linear trail $\xi_{r+1} \xleftarrow{f_r} \cdots \xleftarrow{f_2} \xi_2 \xleftarrow{f_1} \xi_1$, one can estimate

$$|\mathcal{C}(\xi_{r+1} \xleftarrow{f_r \cdots f_1} \xi_1)| \approx \prod_{i=1}^{r} |\mathcal{C}(\xi_{i+1} \xleftarrow{f_i} \xi_i)| \ ,$$

but one should be careful when interpreting this estimate, since many trails might contribute to the same approximation, some with negative correlation and some with positive, and the correlation is usually key dependent. Like in the differential case, we need efficient methods for computing the correlation of linear approximations of the simplest components of the cipher, as well as methods for generating the relevant approximations in order to search the space of linear trails.

1.2.3 Some Special Functions

The correlation of linear approximations of some special functions is straightforward to compute. Let L be a linear function and let f be the affine function $f(x) = L(x) + a$, where a is a constant. Since $u \cdot Lx = u^t Lx = (L^t u)^t x = (L^t u) \cdot x$, we see that $\mathcal{C}(u \xleftarrow{L} w) = \delta(L^t u + w)$. With this observation, it is easy to see that $\mathcal{C}(u \xleftarrow{f} w) = (-1)^{u \cdot a} \delta(L^t u + w)$. The effect of a linear or affine function before or after a nonlinear function is simple.

Let $f_i \colon \mathbf{F}_2^{n_i} \to \mathbf{F}_2^{m_i}$ and define $f \colon \mathbf{F}_2^{n_1+\cdots+n_k} \to \mathbf{F}_2^{m_1+\cdots+m_k}$ by

$$f(x_1, x_2, \ldots, x_k) = (f_1(x_1), f_2(x_2), \ldots, f_k(x_k)) \ .$$

Since the components of f are independent, we see that

$$\mathcal{C}((u_1, \ldots, u_k) \xleftarrow{f} (w_1, \ldots, w_k)) = \prod_{i=1}^{k} \mathcal{C}(u_i \xleftarrow{f_i} w_i) \ .$$

If n_i is small, $\mathcal{C}(u_i \xleftarrow{f_i} w_i)$ can be computed directly from its definition. It is thus easy to compute the correlation of linear approximations of the parallel application of small functions. For arithmetic operations, the situation is more complicated, since the operations are nonlinear and the input domain is large.

1.2.4 Fast Computation Using the FFT

Using the Fourier-based framework, the correlation of linear approximations of small functions can efficiently be computed using the Fast Fourier Transform (FFT). Let $f \colon \mathbf{F}_2^n \to \mathbf{F}_2^m$ be a Boolean function and let $\chi_f \colon \mathbf{F}_2^n \times \mathbf{F}_2^m \to$

$\mathbf{F}_2$ be the *characteristic function* of f, $\chi_f(x,y) = \delta(f(x)+y)$. Note that each Boolean function $g\colon \mathbf{F}_2^n \to \mathbf{F}_2 \in \mathcal{B}_n$, since $g(x) = \frac{1}{2}(1-\hat{g}(x))$ for all x. Thus, we can compute the Fourier transform of χ_f,

$$\begin{aligned}\mathcal{F}(\chi_f)(w,u) &= 2^{-m-n}\sum_{x,y}\chi_f(x,y)(-1)^{u\cdot y+w\cdot x}\\ &= 2^{-m-n}\sum_{x}(-1)^{u\cdot f(x)+w\cdot x} = 2^{-m}\mathcal{C}(u \overset{f}{\leftarrow} w)\ .\end{aligned}$$

Using the FFT (see e.g. [43]), the correlation of all linear approximations of f can be computed in time $O((n+m)2^{n+m})$, or $O(m+n)$ per approximation.

Similarly, the convolution $\chi_f * \chi_f$ is

$$\begin{aligned}(\chi_f * \chi_f)(\alpha,\beta) &= \sum_{x,y}\chi_f(x,y)\chi_f(x+\alpha,y+\beta)\\ &= \sum_{x}\chi_f(x+\alpha,f(x)+\beta)\\ &= \sum_{x}\delta(f(x+\alpha)+f(x)+\beta) = 2^n\Pr[\alpha \xrightarrow{f} \beta]\ .\end{aligned}$$

On the other hand, since the inverse Fourier transform is $\mathcal{F}^{-1} = 2^{m+n}\mathcal{F}$ and hence $\mathcal{F}(\chi_f * \chi_f) = 2^{m+n}\mathcal{F}(\chi_f)^2$, we have

$$\Pr[\alpha \xrightarrow{f} \beta] = 2^{2m+n}\mathcal{F}(\mathcal{F}(\chi_f)^2)(\alpha,\beta)\ . \tag{1.3}$$

If follows that the probability of all differentials for $f\colon \mathbf{F}_2^n \to \mathbf{F}_2^m$ can be computed using the FFT in time $O((m+n)2^{m+n})$, or $O(m+n)$ per differential. For large m, n, this FFT based approach to computing correlations of linear approximations and differential probabilities becomes infeasible. From (1.3), it is also easy to derive the well-known relationships

$$\begin{aligned}\Pr[\alpha \xrightarrow{f} \beta] &= 2^{-m}\sum_{u,w}(-1)^{w\cdot\alpha+u\cdot\beta}\mathcal{C}(u \overset{f}{\leftarrow} w)^2 \quad \text{and}\\ \mathcal{C}(u \overset{f}{\leftarrow} w)^2 &= 2^{-n}\sum_{\alpha,\beta}(-1)^{w\cdot\alpha+u\cdot\beta}\Pr[\alpha \xrightarrow{f} \beta]\end{aligned}$$

between the squares of the correlation coefficients and the differential probabilities.

1.3 DIFFERENTIAL AND LINEAR PROPERTIES OF ADDITION

In this thesis, we study the differential and linear properties of addition modulo 2^n. For the differential properties, we exclusively deal with the set of integers modulo 2^n, $\{0,1,\dots,2^n-1\}$, equipped with two group operations. On one hand, we have the usual addition modulo 2^n, which we denote by $+$. On the other hand, we identify $\{0,1,\dots,2^n-1\}$ and the vector space $\mathbf{F}_2^n$ using the natural correspondence

$$(x_{n-1},\dots,x_1,x_0) \in \mathbf{F}_2^n \leftrightarrow x_{n-1}2^{n-1}+\dots+x_n2+x_0 \in \mathbf{Z}_{2^n}\ .$$

In this way the addition in $\mathbf{F}_2^n$ (or bitwise exclusive-or) carries over to a group operation in $\{0, 1, \ldots, 2^n\}$ which we denote by $\oplus$. We can thus especially view $\oplus$ as a function $\oplus\colon \mathbf{Z}_{2^n} \times \mathbf{Z}_{2^n} \to \mathbf{Z}_{2^n}$. We call the differential probability of the resulting mapping the *additive differential probability* of exclusive-or and denote it by $\mathrm{adp}^{\oplus}\colon \mathbf{Z}_{2^n}^3 \to [0, 1]$,

$$\mathrm{adp}^{\oplus}(\alpha, \beta \to \gamma) = \Pr_{x,y}[((x + \alpha) \oplus (y + \beta)) - (x \oplus y) = \gamma] \enspace .$$

The dual mapping, the *exclusive-or differential probability* of addition, denoted $\mathrm{xdp}^{+}\colon \mathbf{Z}_{2^n}^3 \to [0, 1]$, is given by

$$\mathrm{xdp}^{+}(\alpha, \beta \to \gamma) = \Pr_{x,y}[((x \oplus \alpha) + (y \oplus \beta)) \oplus (x + y) = \gamma] \enspace .$$

This dual mapping was studied in detail by [30], who gave a closed formula for xdp^{+}. Their formula in particular lead to an $O(\log n)$-time algorithm for computing xdp^{+}.

We show that xdp^{+} can be expressed as rational series in the sense of formal language theory with linear representations in base 8. That is, if we write the differential $(\alpha, \beta \to \gamma)$ as an octal word $w = w_{n-1} \cdots w_1 w_0$ in a natural way by identifying $(\mathbf{F}_2^n)^3$ and $\{0, 1, \ldots, 7\}^n$, there are eight square matrices A_i, a column vector C and a row vector L such that

$$\mathrm{xdp}^{+}(\alpha, \beta \to \gamma) = \mathrm{xdp}^{+}(w) = L A_{w_{n-1}} \cdots A_{w_1} A_{w_0} C \enspace .$$

The simple structure of the linear representation allows us to give a straightforward proof of the closed formula from [30]. Using the linear representation, we also give an explicit description of all differentials with a given probability and determine the distribution of the differential probabilities. Using this approach, we obtain simpler and more intuitive proofs of the results in [30]. The approach can be generalised to more complex functions.

Similarly, we show that $\mathrm{adp}^{\oplus}$ can be seen as a rational series with a linear representation in base 8. This representation immediately gives a linear-time algorithm for computing $\mathrm{adp}^{\oplus}$. While some other properties (like the fraction of differentials with nonzero probability) are easily derived from it, the detailed analysis of the distribution of $\mathrm{adp}^{\oplus}$ requires more advanced methods. We study the asymptotic average behaviour of $\mathrm{adp}^{\oplus}$ using tools from analytic number theory. For this, we introduce a sequence $\mathrm{sbs}(n)$ (for *side-by-side*) by putting side-by-side the values of $\mathrm{adp}^{\oplus}(w)$ according to the length and rank in the lexicographic order of the octal word w. We derive an asymptotic expression for the sum of the first order,

$$\sum_{1 \le n < \nu} \mathrm{sbs}(n) \underset{\nu \to \infty}{=} \nu^{2/3} G_{2/3}(\log_8 \nu) + o(\nu^{2/3}) \enspace ,$$

where $G_{2/3}$ is a 1-periodic continuous function. The first terms of the Fourier series of $G_{2/3}$ can be numerically computed. These new results on $\mathrm{adp}^{\oplus}$ are joint work with Philippe Dumas and Helger Lipmaa.

For the linear properties, we exclusive deal with the vector space $\mathbf{F}_2^n$. Using the natural correspondence between $\mathbf{F}_2^n$ and $\mathbf{Z}_{2^n}$, the addition in $\mathbf{Z}_{2^n}$ carries over to a function $\mathbf{F}_2^n \times \mathbf{F}_2^n \to \mathbf{F}_2$ which we denote by $\boxplus$. We can thus study the correlation of $\mathbf{F}_2$-linear approximations of addition modulo 2^n, $\mathrm{lca}(u, v, w) = \mathcal{C}(u \overset{\boxplus}{\longleftarrow} v, w)$. We show that lca also can be seen as a formal rational series with a linear representation in base 8. The simple structure of the linear representation allows us to derive an $O(\log n)$-time algorithm for computing lca, a description of all linear approximations with a given correlation and determine the distribution of the correlation coefficients. All of these results are new and have originally been published in [52], but the presentation in this thesis uses a different approach giving simpler and more intuitive proofs. The approach can be generalised to more complex functions.

We will present our results in increasing difficulty. Thus, we start with the exclusive-or differential probability of addition in Chapter 2 and the linear approximations of addition in Chapter 3. Finally, we treat the additive differential probability of exclusive-or in Chapter 4. The chapters are written to be as independent as possible at the cost of repeating some definitions.

2 THE XOR DIFFERENTIAL PROBABILITY OF ADDITION

When studying the exclusive-or differential probability of addition modulo 2^n, we consider addition as a function $\mathbf{F}_2^n \times \mathbf{F}_2^n \to \mathbf{F}_2^n$ using the natural correspondence

$$(x_{n-1}, x_{n-2}, \ldots, x_0) \in \mathbf{F}_2^n \leftrightarrow x_{n-1}2^{n-1} + x_{n-2}2^{n-2} + \cdots + x_0 \in \mathbf{Z}_{2^n}$$

and express differences with respect to $\mathbf{F}_2^n$-addition. In this chapter, we will denote addition modulo 2^n by $+$ and addition in $\mathbf{F}_2^n$ by $\oplus$. With this notation, the exclusive-or differential probability of addition is defined by

$$\mathrm{xdp}^+(\alpha, \beta \to \gamma) = \Pr[\alpha, \beta \xrightarrow{+} \gamma] = \Pr_{x,y}[((x \oplus \alpha) + (y \oplus \beta)) \oplus (x + y) = \gamma] \enspace .$$

This function was studied by [30], who gave a closed formula for xdp^+. Their formula in particular lead to a $O(\log n)$-time algorithm for computing xdp^+. In this chapter, we reformulate their main results in our framework. Our approach gives simpler and more intuitive proofs of the results as well as generalisations to more complex mappings.

The analysis proceeds as follows. We first show that xdp^+ can be seen as a rational series and derive a linear representation for it. Using the linear representation, we then give an explicit description of all differentials with a given probability and determine the distribution of the differential probabilities. This also gives the closed formula for xdp^+ from [30] and an $O(\log n)$-time algorithm for computing xdp^+. Finally, we discuss generalisations to some other mappings.

2.1 THE RATIONAL SERIES xdp^+

We will consider xdp^+ as a function of octal words by writing the differential $(\alpha, \beta \to \gamma)$ as the octal word $w = w_{n-1} \cdots w_0$, where $w_i = \alpha_i 4 + \beta_i 2 + \gamma_i$. This defines xdp^+ as a function from the octal words of length n to the interval $[0, 1] \subseteq \mathbf{R}$. As n varies in the set of nonnegative integers, we obtain a function from the set of all octal words to $[0, 1]$.

In the terminology of formal language theory, the exclusive-or differential probability xdp^+ is a formal series over the monoid of octal words with coefficients in the field of real numbers. A remarkable subset of these series is the set of *rational series* [6]. One possible characterisation of such a rational series S is the following: there exists a square matrix A_x of size $d \times d$ for each letter x in the alphabet, a row matrix L of size $1 \times d$ and a column matrix C of size $d \times 1$ such that for each word $w = w_1 \ldots w_\ell$, the value of the series is

$$S(w) = L A_{w_1} \cdots A_{w_\ell} C \enspace .$$

The family L, $(A_x)_x$, C is called a *linear representation* of dimension d of the rational series. In our case, the alphabet is the octal alphabet $\{0, \ldots, 7\}$.

Theorem 2.1 (Linear representation of xdp^+**).** The formal series xdp^+ has the 2-dimensional linear representation L, $(A_k)_{k=0}^7$, C, where $L = \begin{pmatrix} 1 & 1 \end{pmatrix}$,

$C = \begin{pmatrix}1 & 0\end{pmatrix}^t$ and A_k is given by

$$(A_k)_{ij} = \begin{cases} 1 - T(k_2 + k_1 + j) & \text{if } i = 0 \text{ and } k_2 \oplus k_1 \oplus k_0 = j \ , \\ T(k_2 + k_1 + j) & \text{if } i = 1 \text{ and } k_2 \oplus k_1 \oplus k_0 = j \ , \\ 0 & \text{otherwise} \end{cases}$$

for $i, j \in \{0, 1\}$, where $k = k_2 4 + k_1 2 + k_0$ and $T\colon \{0, 1, 2, 3\} \to \mathbf{R}$ is the mapping $T(0) = 0$, $T(1) = T(2) = \frac{1}{2}$ and $T(3) = 1$. (For completeness, all the matrices A_k are given in Table 2.1.) Thus, xdp$^+$ is a rational series.

Table 2.1: All the eight matrices A_k in Theorem 2.1.

$A_0 = \frac{1}{2}\begin{pmatrix}2 & 0\\0 & 0\end{pmatrix}$	$A_1 = \frac{1}{2}\begin{pmatrix}0 & 1\\0 & 1\end{pmatrix}$	$A_2 = \frac{1}{2}\begin{pmatrix}0 & 1\\0 & 1\end{pmatrix}$	$A_3 = \frac{1}{2}\begin{pmatrix}1 & 0\\1 & 0\end{pmatrix}$
$A_4 = \frac{1}{2}\begin{pmatrix}0 & 1\\0 & 1\end{pmatrix}$	$A_5 = \frac{1}{2}\begin{pmatrix}1 & 0\\1 & 0\end{pmatrix}$	$A_6 = \frac{1}{2}\begin{pmatrix}1 & 0\\1 & 0\end{pmatrix}$	$A_7 = \frac{1}{2}\begin{pmatrix}0 & 0\\0 & 2\end{pmatrix}$

For example, if $(\alpha, \beta \to \gamma) = (11100, 00110 \to 10110)$, we have $w = 54730$ and $\mathrm{xdp}^+(\alpha, \beta \to \gamma) = \mathrm{xdp}^+(w) = LA_5A_4A_7A_3A_0C = \frac{1}{4}$. In order to prove this result, we introduce the following notation. Define the carry function carry: $\mathbf{F}_2^n \times \mathbf{F}_2^n \to \mathbf{F}_2^n$ of addition modulo 2^n by

$$\mathrm{carry}(x, y) = (x + y) \oplus x \oplus y \ .$$

It is easy to see that

$$\mathrm{xdp}^+(\alpha, \beta \to \gamma) = \Pr_{x,y}[\mathrm{carry}(x, y) \oplus \mathrm{carry}(x \oplus \alpha, y \oplus \beta) = \alpha \oplus \beta \oplus \gamma] \ .$$

Denote $c = \mathrm{carry}(x, y)$ and $c^* = \mathrm{carry}(x \oplus \alpha, y \oplus \beta)$, where x, y, α and β are understood from context. Note that c_i can be recursively defined as $c_0 = 0$ and $c_{i+1} = 1$ if and only if at least two of x_i, y_i and c_i are 1. To simplify some of the formulas, denote $\mathrm{xor}(x, y, z) = x \oplus y \oplus z$ and $\Delta c = c \oplus c^*$. Then $\mathrm{xdp}^+(\alpha, \beta \to \gamma) = \Pr_{x,y}[\Delta c = \mathrm{xor}(\alpha, \beta, \gamma)]$. Let furthermore xy denote the componentwise product of x and y, $(xy)_i = x_i y_i$.

The linear representation of xdp$^+$ follows easily from the following result [30, Lemma 2].

Lemma 2.1. Fix $\alpha, \beta \in \mathbf{F}_2^n$ and $i \geq 0$. Then

$$\Pr_{x,y}[\Delta c_{i+1} = 1 \mid \Delta c_i = r] = T(\alpha_i + \beta_i + r) \ .$$

Proof. Since $c_{i+1} = x_i y_i \oplus x_i c_i \oplus y_i c_i$, we have that $\Delta c_{i+1} = x_i y_i \oplus x_i c_i \oplus y_i c_i \oplus (x_i \oplus \alpha_i)(y_i \oplus \beta_i) \oplus (x_i \oplus \alpha_i) c_i^* \oplus (y_i \oplus \beta_i) c_i^*$. We consider the probability case-by-case.

$(\alpha_i, \beta_i, \Delta c_i)$	Δc_{i+1}	$\Pr[\Delta c_{i+1} = 1 \mid \Delta c_i]$
$(0,0,0)$	0	0
$(0,0,1)$	$x_i \oplus y_i$	$\frac{1}{2}$
$(0,1,0)$	$x_i \oplus c_i$	$\frac{1}{2}$
$(0,1,1)$	$y_i \oplus c_i \oplus 1$	$\frac{1}{2}$
$(1,0,0)$	$y_i \oplus c_i$	$\frac{1}{2}$
$(1,0,1)$	$x_i \oplus c_i \oplus 1$	$\frac{1}{2}$
$(1,1,0)$	$x_i \oplus y_i \oplus 1$	$\frac{1}{2}$
$(1,1,1)$	1	1

In all cases, $\Pr[\Delta c_{i+1} = 1 \mid \Delta c_i = r] = T(\alpha_i + \beta_i + r)$. □

Proof (of Theorem 2.1). Let $(\alpha, \beta \to \gamma)$ be the differential associated with the word w. Let x, y be uniformly distributed independent random variables over $\mathbf{F}_2^{|w|}$. For compactness, we denote $\mathrm{xor}(w) = \alpha \oplus \beta \oplus \gamma$. Let $P(w,k)$ be the 2×1 substochastic matrix given by

$$P_j(w,k) = \Pr_{x,y}[\Delta c \equiv \mathrm{xor}(w) \pmod{2^k}, \Delta c_k = j]$$

for $0 \leq k \leq |w|$ and let $M(w,k)$ be the 2×2 substochastic transition matrix

$$\begin{aligned} M_{ij}(w,k) = \Pr_{x,y}[\Delta c_k = \mathrm{xor}(w)_k, \Delta c_{k+1} = i \mid \\ \Delta c \equiv \mathrm{xor}(w) \pmod{2^k}, \Delta c_k = j] \end{aligned}$$

for $0 \leq k < |w|$. Since $P_i(w, k+1) = \sum_j M_{ij}(w,k) P_j(w,k)$, $P(w,k+1) = M(w,k)P(w,k)$. Note furthermore that $P(w,0) = C$ and that $\mathrm{xdp}^+(w) = \sum_j P_j(w,|w|) = LP(w,|w|)$. By Lemma 2.1, it is clear that

$$M_{ij}(w,k) = \begin{cases} 1 - T(\alpha_k + \beta_k + j) & \text{if } i = 0 \text{ and } \mathrm{xor}(w)_k = j \ , \\ T(\alpha_k + \beta_k + j) & \text{if } i = 1 \text{ and } \mathrm{xor}(w)_k = j \text{ and} \\ 0 & \text{otherwise} \ . \end{cases}$$

That is, $M(w,k) = A_{w_k}$ for all k. It follows by induction that $\mathrm{xdp}^+(w) = LA_{w_{|w|-1}} \cdots A_{w_0} C$. □

2.2 ENUMERATIVE ASPECTS

The simplicity of the linear representation of xdp^+ allows us to derive an explicit description of all words with a certain differential probability. We will use the notation of formal languages to describe words. For example, $(1+2+4+7)0^*$ denotes the set of all words with one of $\{1,2,4,7\}$ followed by any number of zeros.

Theorem 2.2. For all nonempty words w, $\mathrm{xdp}^+(w) \in \{0\} \cup \{2^{-k} \mid k \in \{0, 1, \ldots, |w|-1\}\}$. The differential probability $\mathrm{xdp}^+(w) = 0$ if and only if w has the form $w = w'(1+2+4+7)$, $w = w'(1+2+4+7)0w''$ or $w = w'(0+3+5+6)7w''$ for arbitrary words w', w'', and $\mathrm{xdp}^+(w) = 2^{-k}$ if and only if $\mathrm{xdp}^+(w) \neq 0$ and $|\{0 \leq i < n-1 \mid w_i \neq 0, 7\}| = k$.

Proof. Let L, A_k and C be as in Theorem 2.1 and denote $e_0 = \begin{pmatrix}1 & 0\end{pmatrix}^t$ and $e_1 = \begin{pmatrix}0 & 1\end{pmatrix}^t$. Then the kernels of A_i are $\ker A_0 = \ker A_3 = \ker A_5 = \ker A_6 = \langle e_1 \rangle$ and $\ker A_1 = \ker A_2 = \ker A_4 = \ker A_7 = \langle e_0 \rangle$. By direct calculation, $A_0 e_0 = e_0$, $A_3 e_0 = A_5 e_0 = A_6 e_0 = \frac{1}{2}(e_0 + e_1)$, $A_1 e_1 = A_2 e_1 = A_4 e_1 = \frac{1}{2}(e_0 + e_1)$ and $A_7 e_1 = e_1$. Since $C = e_0$, we thus have $\mathrm{xdp}^+(w) = 0$ if and only if w has the form $w = w'(1{+}2{+}4{+}7)$, $w = w'(1{+}2{+}4{+}7)0w''$ or $w = w'(0{+}3{+}5{+}6)7w''$ for arbitrary words w', w''. Similarly, when w is such that $\mathrm{adp}^+(w) \neq 0$, we see that $A_{w_{n-1}} \cdots A_{w_0} C$ has the form $\begin{pmatrix}2^{-\ell} & 2^{-\ell}\end{pmatrix}^t$, $\begin{pmatrix}2^{-\ell} & 0\end{pmatrix}^t$ or $\begin{pmatrix}0 & 2^{-\ell}\end{pmatrix}^t$, where $\ell = |\{w_i \mid w_i \notin \{0,7\}, 0 \leq i < n\}|$ for all n. Thus, $\mathrm{xdp}^+(w) = 2^{-k}$, where $k = |\{0 \leq i < n-1 \mid w_i \neq 0, 7\}|$. □

For example, if w is the word $w = 54730$, we see that $\mathrm{xdp}^+(w) \neq 0$ and $|\{0 \leq i < 4\} \mid w_i \neq 0, 7\}| = 2$. Thus, $\mathrm{xdp}^+(w) = 2^{-2}$.

Based on this explicit description of all words with a certain differential probability, it is easy to determine the distribution of xdp^+. Let $\mathcal{A}(n,k)$, $\mathcal{B}(n,k)$ and $\mathcal{C}(n,k)$ denote the languages that consist of the words of length n with $\mathrm{xdp}^+(w) = 2^{-k}$, and $w_{n-1} = 0$, $w_{n-1} = 7$ and $w_{n-1} \neq 0, 7$, respectively. The languages are clearly given recursively by

$$\begin{aligned}
\mathcal{A}(n,k) &= 0\mathcal{A}(n-1,k) + 0\mathcal{C}(n-1,k-1) \ , \\
\mathcal{B}(n,k) &= 7\mathcal{B}(n-1,k) + 7\mathcal{C}(n-1,k-1) \ , \\
\mathcal{C}(n,k) &= \Sigma_0\mathcal{A}(n-1,k) + \Sigma_1\mathcal{B}(n-1,k) + (\Sigma_0 + \Sigma_1)\mathcal{C}(n-1,k-1) \ ,
\end{aligned}$$

where $\Sigma_0 = 3 + 5 + 6$ and $\Sigma_1 = 1 + 2 + 4$. The base cases are $\mathcal{A}(1,0) = 0$, $\mathcal{B}(1,0) = \emptyset$ and $\mathcal{C}(1,0) = 3 + 5 + 6$. Let $A(z,u) = \sum_{n,k} |\mathcal{A}(n,k)| u^k z^n$, $B(z,u) = \sum_{n,k} |\mathcal{B}(n,k)| u^k z^n$ and $C(z,u) = \sum_{n,k} |\mathcal{C}(n,k)| u^k z^n$ be the corresponding ordinary generating functions. By the recursive description of the languages, the generating functions are given by the linear system (see e.g. [22])

$$\begin{cases}
A(z,u) = zA(z,u) + uzC(z,u) + z \ , \\
B(z,u) = zB(z,u) + uzC(z,u) \ , \\
C(z,u) = 3zA(z,u) + 3zB(z,u) + 6uzC(z,u) + 3z \ .
\end{cases}$$

Denote $D(z,u) = A(z,u) + B(z,u) + C(z,u) + 1$. Then the coefficient of $u^k z^n$ in $D(z,u)$, $[u^k z^n] D(z,u)$, gives the number of words of length n with $\mathrm{xdp}^+(w) = 2^{-k}$ (the extra 1 comes from the case $n = 0$). By solving the linear system, we see that

$$D(z,u) = 1 + \frac{4z}{1 - (1 + 6u)z} \ .$$

Since the coefficient of z^n in $D(z,u)$ for $n > 0$ is

$$[z^n] D(z,u) = 4[z^n] z \sum_{m=0}^{\infty} (1 + 6u)^m z^m = 4(1 + 6u)^{n-1} \ ,$$

we see that

$$[u^k z^n] D(z,u) = 4 \cdot 6^k \binom{n-1}{k}$$

for all $0 \leq k < n$. The coefficient of z^n in $D(z,1)$ for $n > 0$, $[z^n] D(z,1) = 4[z^n] \frac{z}{1-7z} = 4 \cdot 7^{n-1}$ gives the number of words of length n with $\mathrm{xdp}^+(w) \neq 0$.

Theorem 2.3 ([30, Theorem 2])**.** There are $4 \cdot 7^{n-1}$ words of length $n > 0$ with $\mathrm{xdp}^+(w) \neq 0$. Of these, $4 \cdot 6^k \binom{n-1}{k}$ have probability 2^{-k} for all $0 \leq k < n$.

Note that

$$\Pr_{|w|=n}[\mathrm{xdp}^+(w) \neq 0] = \frac{1}{2}\left(\frac{7}{8}\right)^{n-1}$$

and

$$\Pr_{|w|=n}[-\log_2 \mathrm{xdp}^+(w) = k \mid \mathrm{xdp}^+(w) \neq 0] = \left(\frac{6}{7}\right)^k \left(\frac{1}{7}\right)^{n-1-k} \binom{n-1}{k}$$

for $0 \leq k < n$. Conditioned on $\mathrm{xdp}^+(w) \neq 0$, $-\log_2 \mathrm{xdp}^+(w)$ is thus binomially distributed with mean $\frac{6}{7}(n-1)$ and variance $\frac{6}{49}(n-1)$ for words of length $n > 0$.

2.3 EFFICIENT ALGORITHMS FOR xdp^+

The simple structure of the linear representation of xdp^+ can be used to derive very efficient algorithms for computing differential properties of addition. Let $\mathrm{eq}\colon (\mathbf{F}_2^n)^3 \to \mathbf{F}_2^n$ be the bitwise equality function, $\mathrm{eq}(x, y, z)_i = 1$ if and only if $x_i = y_i = z_i$. Let $\mathrm{mask}(\ell) \in \mathbf{F}_2^n$ be such that $\mathrm{mask}(\ell)_i = 1$ if and only if $0 \leq i < \ell$ and let $\mathrm{w_h}(x) = |\{i \mid x_i \neq 0\}|$ denote the Hamming weight of x. Let $x \wedge y$ denote the componentwise product (or bitwise and) of the binary vectors x and y, $(x \wedge y)_i = x_i y_i$ and let $\neg x$ denote the componentwise negation of x, $\neg x_i = x_i \oplus 1$. Let $x \lll k \in \mathbf{F}_2^n$ denote the vector x shifted left k positions, $(x \lll k)_i = x_{i-k}$ if $i \geq k$ and $(x \lll k)_i = 0$ otherwise. To simplify the formulas, denote $\mathrm{xor}(x, y, z) = x \oplus y \oplus z$. The main result of [30] is

Theorem 2.4 ([30, Theorem 1])**.** Let $\alpha, \beta, \gamma \in \mathbf{F}_2^n$. Then $\mathrm{xdp}^+(\alpha, \beta \to \gamma) = 0$ when

$$\mathrm{eq}(\alpha \lll 1, \beta \lll 1, \gamma \lll 1) \wedge (\mathrm{xor}(\alpha, \beta, \gamma) \oplus (\alpha \lll 1)) \neq 0$$

and

$$\mathrm{xdp}^+(\alpha, \beta \to \gamma) = 2^{-\mathrm{w_h}(\neg \mathrm{eq}(\alpha,\beta,\gamma) \wedge \mathrm{mask}(n-1))}$$

otherwise.

Proof. Follows directly from Theorem 2.2. □

Since the Hamming weight can be computed in time $O(\log n)$, this gives a $O(\log n)$-time algorithm for computing the exclusive-or differential probability of addition. Based on Theorem 2.2, it is trivial to derive an optimal (that is, linear-time in the size of the output) algorithm that generates all differentials with a given differential probability. Moreover, one or two of the input or output differences can optionally be fixed. Using similar ideas, it is also easy to come up with an efficient algorithm that given (α, β) finds all γ such that

$$\mathrm{xdp}^+(\alpha, \beta \to \gamma) = \max_{\gamma'} \mathrm{xdp}^+(\alpha, \beta \to \gamma') \enspace .$$

An explicit algorithm can be found in [30]. Although these kinds of optimisation problems falls out of scope of our problem statement, we mention that [30] also gives an $O(\log n)$-time algorithm that finds one γ achieving the maximum differential probability and an $O(\log n)$-time algorithm that given α computes

$$\max_{\beta,\gamma} \mathrm{xdp}^+(\alpha, \beta \to \gamma) \enspace .$$

Since xdp^+ is a symmetric function, the algorithms can also be used for maximisation with respect to the other inputs.

2.4 GENERALISATIONS TO SOME OTHER MAPPINGS

The results for xdp^+ are easy to generalise to mappings of the form $(x, y) \mapsto 2^k x \pm 2^\ell y \bmod 2^n$, $k, \ell \geq 0$. The differential probability of this type of mappings was derived using a linear-algebraic approach in [29]. We will, however, take a more direct approach.

Let xdp^- denote the exclusive-or differential probability of subtraction modulo 2^n,

$$\mathrm{xdp}^-(\alpha, \beta \to \gamma) = \Pr_{x,y}[((x \oplus \alpha) - (y \oplus \beta)) \oplus (x - y)] = \gamma] \enspace .$$

Let $\mathrm{borrow}(x, y) = (x - y) \oplus x \oplus y$ be the borrows in the subtraction $x - y$ modulo 2^n as an n-tuple of bits. It is easy to see that the exclusive-or differential probability of subtraction is given by

$$\mathrm{xdp}^-(\alpha, \beta \to \gamma) = \Pr_{x,y}[\mathrm{borrow}(x, y) \oplus \mathrm{borrow}(x \oplus \alpha, y \oplus \beta) = \alpha \oplus \beta \oplus \gamma] \enspace .$$

Denote $b = \mathrm{borrow}(x, y)$ and $b^* = \mathrm{borrow}(x \oplus \alpha, y \oplus \beta)$, where x, y, α and β are understood from context. Note that b_i can be recursively defined as $b_0 = 0$ and $b_{i+1} = 1$ if and only if $x_i < y_i + b_i$ as integers. Denote $\Delta b = b \oplus b^*$. Then $\mathrm{xdp}^-(\alpha, \beta \to \gamma) = \Pr_{x,y}[\Delta b = \alpha \oplus \beta \oplus \gamma]$.

Lemma 2.2. Fix $\alpha, \beta \in \mathbf{F}_2^n$ and let $i \geq 0$. Then

$$\Pr_{x,y}[\Delta b_{i+1} = 1 \mid \Delta b_i = r] = T(\alpha_i + \beta_i + r) \enspace ,$$

where T is as in Theorem 2.1.

Proof. By the recursive definition of $\mathrm{borrow}(x, y)$, we see that $b_{i+1} = 1$ if and only if either $y_i = b_i$ and at least two of x_i, y_i and b_i are 1, or $y_i \neq b_i$ and at least two of x_i, y_i and b_i are 0. Thus, $b_{i+1} = y_i \oplus b_i \oplus \mathrm{maj}(x_i, y_i, b_i)$, where $\mathrm{maj}(u, v, w)$ denotes the majority of the bits u, v, w. Since $\mathrm{maj}(u, v, w) = uv \oplus uw \oplus vw$, we have $b_{i+1} = y_i \oplus b_i \oplus x_i y_i \oplus x_i b_i \oplus y_i b_i$. This gives a recursive description of Δb_{i+1}. As in the proof of Lemma 2.1, we consider Δb_{i+1} case-by-case.

$(\alpha_i, \beta_i, \Delta b_i)$	Δb_{i+1}	$\Pr[\Delta b_{i+1} = 1 \mid \Delta b_i = r]$
$(0,0,0)$	0	0
$(0,0,1)$	$1 \oplus x_i \oplus y_i$	$\frac{1}{2}$
$(0,1,0)$	$1 \oplus x_i \oplus b_i$	$\frac{1}{2}$
$(0,1,1)$	$1 \oplus y_i \oplus b_i$	$\frac{1}{2}$
$(1,0,0)$	$y_i \oplus b_i$	$\frac{1}{2}$
$(1,0,1)$	$x_i \oplus b_i$	$\frac{1}{2}$
$(1,1,0)$	$x_i \oplus y_i$	$\frac{1}{2}$
$(1,1,1)$	1	1

In all cases, $\Pr[\Delta b_{i+1} = 1 \mid \Delta b_i = r] = T(\alpha_i + \beta_i + r)$. □

Using this lemma, it is easy to show that xdp^+ and xdp^- have the same linear representation.

Corollary 2.1. For all α, β, γ, $\mathrm{xdp}^-(\alpha, \beta \to \gamma) = \mathrm{xdp}^+(\alpha, \beta \to \gamma)$.

Proof. Since $\mathrm{xdp}^-(\alpha, \beta \to \gamma) = \Pr[\Delta b = \alpha \oplus \beta \oplus \gamma]$ and $\Pr[\Delta b_{i+1} = 1 \mid \Delta b_i = r] = T(\alpha_i + \beta_i + r)$, the proof of Theorem 2.1 with borrow taking the role of carry shows that xdp^- is a rational series with the same linear representation as xdp^+. □

Let $f \colon \mathbf{F}_2^n \times \mathbf{F}_2^n \to \mathbf{F}_2^n$ be a mapping of the form $f(x, y) = 2^k x \pm 2^\ell y \bmod 2^n$ where $k, \ell \geq 0$. Since the mapping $x \mapsto 2^k x \bmod 2^n$ is $\mathbf{F}_2$-linear, it follows that

$$\Pr[\alpha, \beta \xrightarrow{f} \gamma] = \mathrm{xdp}^+(2^k \alpha \bmod 2^n, 2^\ell \beta \bmod 2^n) \enspace .$$

The differential properties of mappings like the pseudo-Hadamard transform (PHT) $\mathrm{pht} \colon \mathbf{Z}_{2^n}^2 \to \mathbf{Z}_{2^n}^2$,

$$\mathrm{pht}(x, y) = \begin{pmatrix} 2 & 1 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = (2x + y, x + y) \pmod{2^n} \enspace ,$$

used e.g. in [32, 33, 47] can also be studied using the same framework. The exclusive-or differential probability of pht is given by

$$\begin{aligned} \mathrm{xdp}^{\mathrm{pht}}(\alpha, \beta \to \gamma, \delta) = \Pr_{x,y}[&(2(x \oplus \alpha) + (y \oplus \beta)) \oplus (2x + y) = \gamma, \\ &((x \oplus \alpha) + (y \oplus \beta)) \oplus (x + y) = \delta] \enspace . \end{aligned}$$

We can consider $\mathrm{xdp}^{\mathrm{pht}}$ as a function of hexadecimal words by writing the differential $(\alpha, \beta \to \gamma, \delta)$ as the hexadecimal word $w = w_{n-1} \cdots w_0$, where $w_i = \alpha_i 8 + \beta_i 4 + \gamma_i 2 + \delta_i$. As n varies, we obtain a function from all hexadecimal words to $[0, 1]$. A linear representation for the rational series $\mathrm{xdp}^{\mathrm{pht}}$ can easily be derived as follows.

Let $c = \mathrm{carry}(2x, y)$, $c^* = \mathrm{carry}(2(x \oplus \alpha), y \oplus \beta)$, $d = \mathrm{carry}(x, y)$ and $d^* = \mathrm{carry}(x \oplus \alpha, y \oplus \beta)$, where x, y, α and β are understood from context. Denote $\Delta c = c \oplus c^*$ and $\Delta d = d \oplus d^*$. Then

$$\mathrm{xdp}^{\mathrm{pht}}(\alpha, \beta \to \gamma, \delta) = \Pr_{x,y}[\Delta c = (2\alpha) \oplus \beta \oplus \gamma, \Delta d = \alpha \oplus \beta \oplus \delta] \enspace .$$

By the recursive description of the carry function, we see that $c_{i+1} = x_{i-1}y_i \oplus x_{i-1}c_i \oplus y_i c_i$ and $d_{i+1} = x_i y_i \oplus x_i d_i \oplus y_i d_i$. Thus,

$$(c_i, c_i^*, d_i, d_i^*, x_{i-1}, \alpha_{i-1}) \mapsto (c_{i+1}, c_{i+1}^*, d_{i+1}, d_{i+1}^*, x_i, \alpha_i)$$

forms a nonhomogenous Markov chain (see e.g. [11]) with easily computed transition probabilities depending on α_i and β_i. If we set the probability of all state transitions leading to states with $\Delta c_i \neq \alpha_{i-1} \oplus \beta_i \oplus \gamma_i$ or $\Delta d_i \neq \alpha_i \oplus \beta_i \oplus \delta_i$ to zero, we obtain a linear representation of dimension 64 of $\mathrm{xdp}^{\mathrm{pht}}$. In a similar manner, we obtain linear representations of the exclusive-or differential probability of all functions of the form

$$(x, y) \mapsto \begin{pmatrix} 2^{k_{11}} & \pm 2^{k_{12}} \\ 2^{k_{21}} & \pm 2^{k_{22}} \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \pmod{2^n} \ , \tag{2.1}$$

where $k_{ij} \geq 0$ and the signs are independent, although the dimensions of the straightforward linear representations will be large. While we will not discuss the differential properties of these mappings, this observation illustrates that it is trivial to obtain linear-time algorithms for computing the differential probability for large classes of mappings based on addition modulo 2^n. The exclusive-or differential probability of mappings of the form (2.1) was studied using a linear-algebraic approach in [29], who gave a $O(\log n)$-time algorithm for computing $\mathrm{xdp}^{\mathrm{pht}}$.

3 LINEAR APPROXIMATIONS OF ADDITION

When studying the $\mathbf{F}_2$-linear approximations of addition modulo 2^n, we will consider addition modulo 2^n as a Boolean function $\mathbf{F}_2^n \times \mathbf{F}_2^n \to \mathbf{F}_2^n$ using the natural correspondence

$$(x_{n-1}, \ldots, x_1, x_0) \in \mathbf{F}_2^n \leftrightarrow x_{n-1}2^{n-1} + \cdots + x_1 2^1 + x_0 2^0 \in \mathbf{Z}_{2^n} \ .$$

To avoid confusion, we sometimes use $\oplus$ and $\boxplus$ to denote addition in $\mathbf{F}_2^n$ and addition modulo 2^n, respectively. We furthermore let $\text{carry}\colon \mathbf{F}_2^n \times \mathbf{F}_2^n \to \mathbf{F}_2^n$ be the carry function for addition modulo 2^n defined by $\text{carry}(x, y) = x \oplus y \oplus (x \boxplus y)$. For notational convenience, we define the functions $\text{lca}, \text{lcc}, \text{lcs}\colon (\mathbf{F}_2^n)^3 \to [-1, 1]$ (for linear correlation of addition, the carry function and subtraction, respectively) by

$$\begin{aligned} \text{lca}(u, v, w) &= \mathcal{C}(u \xleftarrow{\boxplus} v, w) \ , \\ \text{lcc}(u, v, w) &= \mathcal{C}(u \xleftarrow{\text{carry}} v, w) \quad \text{and} \\ \text{lcs}(u, v, w) &= \mathcal{C}(u \xleftarrow{\boxminus} v, w) \ . \end{aligned}$$

We have previously studied these functions from an algorithmic point of view in [52]. The simpler case with one addend fixed has previously been considered by [38]. In this chapter, we reformulate the results from [52] in our framework. This approach gives a much simpler and more intuitive analysis of these linear approximations.

Since the only nonlinear part of addition modulo 2^n is the carry function, it should be no surprise that the linear properties of addition completely reduce to those of the carry function. Subtraction is also straightforward. When we are approximating the relation $x \boxminus y = z$ by $u \xleftarrow{\boxminus} v, w$, we are actually approximating the relation $z \boxplus y = x$ by $v \xleftarrow{\boxplus} u, w$. With this observation, it is trivial to prove that

$$\begin{aligned} \text{lca}(u, v, w) &= \text{lcc}(u, v + u, w + u) \quad \text{and} \\ \text{lcs}(u, v, w) &= \text{lcc}(v, u + v, w + v) \ . \end{aligned}$$

Moreover, the mappings $(u, v, w) \mapsto (u, v + u, w + u)$ and $(u, v, w) \mapsto (v, u + v, w + v)$ are permutations in $(\mathbf{F}_2^n)^3$. The linear properties of addition and subtraction modulo 2^n under consideration thus completely reduce to the corresponding properties of the carry function. In the sequel, we will focus on the function lcc knowing that the generalisations to lca and lcs are trivial.

Our analysis proceeds as follows. We first show that lcc can be seen as a formal rational series and derive a linear representation for it. Using this linear representation, we then derive an explicit description of all linear approximations with a given nonzero correlation, determine the distribution of the correlation coefficients and derive a $\Theta(\log n)$-time algorithm for computing lcc. Finally, we discuss generalisations to some other functions.

3.1 THE RATIONAL SERIES lcc

We will consider lcc as a function of octal words by writing the linear approximation $(u \xleftarrow{\text{carry}} v, w)$ as the octal word $x = x_{n-1} \cdots x_0$, where $x_i = u_i 4 + v_i 2 + w_i$. This defines lcc as a function form the octal words of length n to the interval $[-1, 1]$. As n varies in the set of nonnegative integers, we obtain a function from the set of all octal words to $[-1, 1]$. In the same way, we consider lca and lcs as functions from the set of all octal words to $[-1, 1]$.

In the terminology of formal language theory, lcc is a formal series over the monoid of octal words with coefficients in the field of real numbers. A remarkable subset of these series is the set of *rational series* [6]. One characterisation of such a rational series S is: there exists an row matrix L of dimension $1 \times d$, a square matrix A_x of dimension $d \times d$ for each letter in the alphabet, and a column matrix C of dimension $d \times 1$ such that for each word $w = w_1 \cdots w_\ell$, the value of the series is

$$S(w) = L A_{w_1} \cdots A_{w_\ell} C \ .$$

The family L, $(A_x)_x$, L is called a *linear representation* of dimension d of the rational series. For the series lcc, the alphabet is $\{0, \dots, 7\}$.

We will use the following bracket notation: if ϕ is any statement, $[\phi] = 1$ when ϕ is true and $[\phi] = 0$ when ϕ is false.

Theorem 3.1 (Linear representation of lcc). The formal series lcc has the 2-dimensional linear representation L, $(A_k)_{k=0}^{7}$, C, where $L = \begin{pmatrix} 1 & 0 \end{pmatrix}$, $C = \begin{pmatrix} 1 & 1 \end{pmatrix}^t$, and

$$A_k = \frac{1}{2} \begin{pmatrix} 2[k_1 = k_0 = 0] & 0 \\ [k_1 \neq k_0] & (-1)^{k_0}[k_1 = k_0] \end{pmatrix} \qquad \text{if } k_2 = 0 \text{ and}$$
$$A_k = \frac{1}{2} \begin{pmatrix} 0 & 2[k_1 = k_0 = 0] \\ (-1)^{k_0}[k_1 = k_0] & [k_1 \neq k_0] \end{pmatrix} \qquad \text{if } k_2 = 1 \ ,$$

where $k = k_2 4 + k_1 2 + k_0$. Thus, lcc is a rational series. (All the matrices A_k are given in Table 3.1 on the next page.)

For example, if $(u \leftarrow v, w) = (10100 \leftarrow 01110, 01000)$, $x = 43620$ and $\mathrm{lcc}(u, v, w) = \mathrm{lcc}(x) = L A_4 A_3 A_6 A_2 A_0 C = -\frac{1}{8}$. In order to prove this result, we will first give a simple recursive expression for lcc. If $u \neq 0$, let k be maximal such that $u_k = 1$. Since $u \cdot \mathrm{carry}(x, y)$ is independent of x_ℓ and y_ℓ when $\ell \geq k$, we see that $\mathrm{lcc}(u, v, w) = 0$ whenever $v_\ell \neq 0$ or $w_\ell \neq 0$ for some $\ell \geq k$. This trivial observation is crucial in the proof. Let $e_i \in \mathbf{F}_2^n$ be a vector whose ith component is 1 and the others are 0. For $x, y \in \mathbf{F}_2^n$, $\overline{x}$ denotes the componentwise negation of x, $\overline{x}_i = x_i + 1$, and xy denotes the componentwise product of x and y, $(xy)_i = x_i y_i$.

Lemma 3.1. The function $\mathrm{lcc}(u, v, w)$ is given recursively as follows. First, $\mathrm{lcc}(0, v, w) = \mathrm{lcc}(e_0, v, w) = \delta(v, w)$. Second, if $u \notin \{0, e_0\}$, let k be maximal such that $u_k = 1$ and let $i \geq k$. Then

$$\mathrm{lcc}(u + e_{i+1}, v, w) = \frac{1}{2} \begin{cases} \mathrm{lcc}(u, v\overline{e_i}, w\overline{e_i}) & \text{if } v_i \neq w_i \text{ and} \\ (-1)^{v_i} \mathrm{lcc}(u + e_i, v\overline{e_i}, w\overline{e_i}) & \text{otherwise} \ . \end{cases}$$

Table 3.1: All the eight matrices A_k in Theorem 3.1.

$A_0 = \frac{1}{2}\begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix}$	$A_1 = \frac{1}{2}\begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix}$	$A_2 = \frac{1}{2}\begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix}$	$A_3 = \frac{1}{2}\begin{pmatrix} 0 & 0 \\ 0 & -1 \end{pmatrix}$
$A_4 = \frac{1}{2}\begin{pmatrix} 0 & 2 \\ 1 & 0 \end{pmatrix}$	$A_5 = \frac{1}{2}\begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}$	$A_6 = \frac{1}{2}\begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}$	$A_7 = \frac{1}{2}\begin{pmatrix} 0 & 0 \\ -1 & 0 \end{pmatrix}$

Proof. Let $c_i = \mathrm{carry}_i$ denote the ith component of the carry function. This function can be recursively computed as $c_0(x,y) = 0$ and $c_{i+1}(x,y) = 1$ if and only if at least two of x_i, y_i and $c_i(x,y)$ are 1. By considering the 8 possible values of x_i, y_i and $c_i(x,y)$, we see that $\hat{c}_0(x,y) = 1$ and $\hat{c}_{i+1}(x,y) = \frac{1}{2}\big((-1)^{x_i} + (-1)^{y_i} + \hat{c}_i(x,y) - (-1)^{x_i+y_i}\hat{c}_i(x,y)\big)$. The Fourier transform of $\hat{c}_i$ is thus given by the recurrence

$$\begin{aligned} \hat{C}_0(v,w) &= \delta(v,w) \quad \text{and} \\ 2\hat{C}_{i+1}(v,w) &= \delta(v+e_i,w) + \delta(v,w+e_i) + \hat{C}_i(v,w) - \hat{C}_i(v+e_i,w+e_i) \end{aligned}$$

for all i.

Denote $\hat{F}(v,w) = \mathrm{lcc}(u,v,w)$. Using the recurrence for $\hat{C}_{i+1}$, we have $\mathrm{lcc}(u+e_{i+1},v,w) = (\hat{C}_{i+1}*\hat{F})(v,w) = \frac{1}{2}\big(\hat{F}(v+e_i,w)+\hat{F}(v,w+e_i)+(\hat{C}_i*\hat{F})(v,w)-(\hat{C}_i*\hat{F})(v+e_i,w+e_i)\big)$. Note that at most one of the terms in this expression is nonzero. The four terms consider the cases $(v_i,w_i) = (1,0)$, $(0,1)$, $(0,0)$ and $(1,1)$, respectively. It follows that

$$\mathrm{lcc}(u+e_{i+1},v,w) = \begin{cases} \frac{1}{2}\,\mathrm{lcc}(u, v\overline{e_i}, w\overline{e_i}) & \text{if } v_i \neq w_i \text{ and} \\ \frac{1}{2}(-1)^{v_i}\,\mathrm{lcc}(u+e_i, v\overline{e_i}, w\overline{e_i}) & \text{if } v_i = w_i \ . \end{cases}$$

This completes the proof. □

Using this lemma, it is easy to derive a linear representation of the series lcc.

Proof (of Theorem 3.1). Fix a word x, $n = |x|$, and let $(u \xleftarrow{\text{carry}} v, w)$ be the corresponding linear approximation. For $z \in \mathbf{F}_2^n$, $b \in \{0,1\}$ and $0 \leq \ell < n$, denote $z|_\ell^b = (0,\ldots,0,b,z_{\ell-1},\ldots,z_0)$ and $\mathrm{lcc}_\ell^b(u,v,w) = \mathrm{lcc}(u|_\ell^b\ v|_\ell^0\,,\ w|_\ell^0)$. Let P^ℓ be the 2×1 matrix $P_b^\ell = \mathrm{lcc}_\ell^b(u,v,w)$. We show by induction on ℓ that $P^\ell = A_{x_{\ell-1}} \cdots A_{x_0} C$ for all ℓ. Since $P^0 = C$, the base case is clear, so consider $\ell > 0$. We consider the following four cases.

- If $b = u_{\ell-1} = 0$, $P_0^\ell = P_0^{\ell-1}$ when $v_{\ell-1} = w_{\ell-1} = 0$ and $P_0^\ell = 0$ otherwise.
- If $b = 1$ and $u_{\ell-1} = 0$, Lemma 3.1 (with $i > k$) shows that $P_1^\ell = \frac{1}{2}P_0^{\ell-1}$ when $v_{\ell-1} \neq w_{\ell-1}$ and $P_1^\ell = \frac{1}{2}(-1)^{v_{\ell-1}}P_1^{\ell-1}$ when $v_{\ell-1} = w_{\ell-1}$.
- If $b = 0$ and $u_{\ell-1} = 1$, $P_0^\ell = P_1^{\ell-1}$ when $v_{\ell-1} = w_{\ell-1} = 0$ and $P_0^\ell = 0$ otherwise.

- If $b = u_{\ell-1} = 1$, Lemma 3.1 (with $i = k$) shows that $P_1^\ell = \frac{1}{2}P_1^{\ell-1}$ when $v_{\ell-1} \neq w_{\ell-1}$ and $P_1^\ell = \frac{1}{2}(-1)^{v_{\ell-1}}P_0^{\ell-1}$ when $v_{\ell-1} = w_{\ell-1}$.

In all cases, $P^\ell = AP^{\ell-1}$, where A is the 2×2 matrix

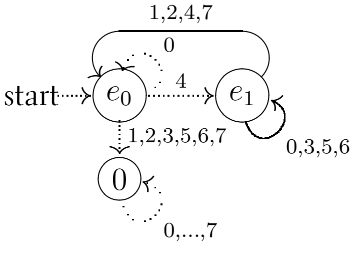
$$A = \frac{1}{2}\begin{pmatrix} 2[v_{\ell-1} = w_{\ell-1} = 0] & 0 \\ [v_{\ell-1} \neq w_{\ell-1}] & (-1)^{v_{\ell-1}}[v_{\ell-1} = w_{\ell-1}] \end{pmatrix} \quad \text{if } u_{\ell-1} = 0 \text{ and}$$
$$A = \frac{1}{2}\begin{pmatrix} 0 & 2[v_{\ell-1} = w_{\ell-1} = 0] \\ (-1)^{v_{\ell-1}}[v_{\ell-1} = w_{\ell-1}] & [v_{\ell-1} \neq w_{\ell-1}] \end{pmatrix} \quad \text{if } u_{\ell-1} = 1 \ .$$

In all cases, $A = A_{x_{\ell-1}}$. By induction, we have $P^\ell = A_{x_{\ell-1}} \cdots A_{x_0}C$ for all ℓ. Since $\operatorname{lcc}(u, v, w) = \operatorname{lcc}_n^0(u, v, w) = LP^n$, it follows that $\operatorname{lcc}(u, v, w) = \operatorname{lcc}(x) = LA_{x_{n-1}} \cdots A_{x_0}C$. □

An alternative proof of Theorem 3.1 was given by [12]. In this proof, we let $s_i = \sum_{j \leq i}(u_j \cdot \operatorname{carry}(x, y) + v_jx_j + w_jy_j)$ denote the partial sum of the chosen input and output bits up to (and including) position i. Let $c_i = \operatorname{carry}(x, y)$ denote the ith carry bit. Then (s_i, c_{i+1}) constitute a nonhomogenous Markov chain (see e.g. [11]) with different transition matrices depending on (u_i, v_i, w_i). Straightforward calculations give the eight 4×4 transition matrices. After a change of basis, the linear representation in Theorem 3.1 is obtained. We omit the details.

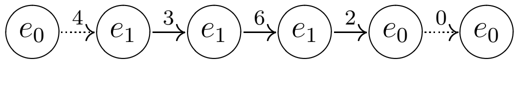
3.2 ENUMERATIVE ASPECTS

The simplicity of the linear representation of lcc allows us to derive an explicit description of all words with a given correlation. We will use the notation of formal languages to describe words. For example, $(3 + 5 + 6)1^*$ denotes the set of all words with one of $\{3, 5, 6\}$ followed by any number of ones. Let L, A_k and C be as in Theorem 3.1, and denote $e_0 = \begin{pmatrix} 1 & 0 \end{pmatrix}$ and $e_1 = \begin{pmatrix} 0 & 1 \end{pmatrix}$. Then $e_0A_0 = e_0$, $e_0A_4 = e_1$, $e_0A_i = 0$ for $i \neq 0, 4$, $e_1A_0 = e_1A_5 = e_1A_6 = \frac{1}{2}e_1$, $e_1A_1 = e_1A_2 = e_1A_4 = \frac{1}{2}e_0$, $e_1A_3 = -\frac{1}{2}e_1$ and $e_1A_7 = -\frac{1}{2}e_0$. It follows by induction that $e_0A_{w_{|w|-1}} \cdots A_{w_0}$ has the form 0, $\pm 2^{-k}e_0$ or $\pm 2^{-k}e_1$ for some $0 \leq k < |w|$. Since $L = e_0$ and $C = \begin{pmatrix} 1 & 1 \end{pmatrix}^t$, it follows that the computation of $\operatorname{lcc}(w) = LA_{w_{|w|-1}} \cdots A_{w_0}C$ can be described by the following automaton.



(3.1)

If the automaton ends up in state 0, $\operatorname{lcc}(w) = 0$. If the automaton ends up in state e_0 or e_1, $\operatorname{lcc}(w) = \pm 2^{-k}$, where k is the number of transitions marked by a solid arrow, and the sign is determined by the number of occurrences of $3 + 7$: $\operatorname{lcc}(w) > 0$ if and only if the number of occurrences is even. For example, when computing lcc(43620), we have the state transitions



and thus $\operatorname{lcc}(43620) = -2^{-3}$. Clearly, $\operatorname{lcc}(w) = 0$ if and only if w has the form

$$w = \bigl(0 + 4(0+3+5+6)^*(1+2+4+7)\bigr)^*(1+2+3+5+6+7)\Sigma^* \ ,$$

where $\Sigma = 0 + 1 + \cdots + 7$.

Let $\mathcal{S}^0(n,k)$ and $\mathcal{S}^1(n,k)$ denote the languages

$$\begin{aligned} \mathcal{S}^0(n,k) &= \{w \mid |w| = n, e_0 A_{w_{n-1}} \cdots A_{w_0} = \pm 2^{-k} e_0\} \quad \text{and} \\ \mathcal{S}^1(n,k) &= \{w \mid |w| = n, e_0 A_{w_{n-1}} \cdots A_{w_0} = \pm 2^{-k} e_1\} \end{aligned}$$

for $n > 0$. Then $\mathcal{S}^0(n,k) + \mathcal{S}^1(n,k)$ is the set of words of length $n > 0$ with $\operatorname{lcc}(w) = \pm 2^{-k}$. The languages are clearly given recursively by

$$\begin{aligned} \mathcal{S}^0(n,k) &= \mathcal{S}^0(n-1,k)0 + \mathcal{S}^1(n-1,k-1)(1+2+4+7) \quad \text{and} \\ \mathcal{S}^1(n,k) &= \mathcal{S}^0(n-1,k)4 + \mathcal{S}^1(n-1,k-1)(0+3+5+6) \end{aligned}$$

for all $0 \le k < n$. The base cases are $\mathcal{S}^0(1,0) = 0$ and $\mathcal{S}^1(1,0) = 4$. If $k < 0$ or $k \ge n$, $\mathcal{S}^0(n,k) = \mathcal{S}^1(n,k) = \emptyset$.

Theorem 3.2. For all nonempty words w, the correlation $\operatorname{lcc}(w) \in \{0\} \cup \{\pm 2^k \mid k \in \{0,1,\ldots,|w|-1\}\}$. The set of words of length $n > 0$ with correlation $\pm 2^{-k}$ is given by $\mathcal{S}^0(n,k) + \mathcal{S}^1(n,k)$, where $\mathcal{S}^0$ and $\mathcal{S}^1$ are determined recursively as follows. First, $\mathcal{S}^0(1,0) = 0$, $\mathcal{S}^1(1,0) = 4$ and $\mathcal{S}^0(n,k) = \mathcal{S}^1(n,k) = \emptyset$ when $k < 0$ or $k \ge n$. Second, when $0 \le k < n \ne 1$,

$$\begin{aligned} \mathcal{S}^0(n,k) &= \mathcal{S}^0(n-1,k)0 + \mathcal{S}^1(n-1,k-1)(1+2+4+7) \quad \text{and} \\ \mathcal{S}^1(n,k) &= \mathcal{S}^0(n-1,k)4 + \mathcal{S}^1(n-1,k-1)(0+3+5+6) \ . \end{aligned}$$

For all words $w \in \mathcal{S}^0(n,k) + \mathcal{S}^1(n,k)$, $\operatorname{lcc}(w) = 2^{-k}$ if and only if w contains an even number of occurrences of $(3+7)$ and $\operatorname{lcc}(w) = -2^{-k}$ otherwise.

From this result, it can be seen that there are $8(n-1)$ words of length $n > 0$ with $\operatorname{lcc}(w) = \pm\frac{1}{2}$. These are the words of the form

$$w = 0^*4(1+2+4+7)0^*(4+\lambda) \quad \text{and} \quad w = 0^*4(0+3+5+6) \ ,$$

where λ denotes the empty word. Among these words, $\operatorname{lcc}(w)$ is negative if and only if w contains a 3 or a 7.

The recursive description in Theorem 3.2 can easily be used to generate all linear approximations with a given nonzero correlation. The straightforward algorithm uses $O(n)$ space and is linear-time in the number of generated approximations. Clearly, this immediately generalises to the case when some of the selection vectors are fixed. The result is also trivial to generalise to addition and subtraction modulo 2^n.

Corollary 3.1. The set of linear approximations of the carry function, addition and subtraction modulo 2^n can be generated in optimal time (that is, linear in the size of the output) and $O(n)$ space in a standard RAM model of computation. Moreover, one or two of the selection vectors can optionally be fixed.

Let $S^0(z,u) = \sum_{n,k} |\mathcal{S}^0(n,k)| u^k z^n$ and $S^1(z,u) = \sum_{n,k} |\mathcal{S}^1(n,k)| u^k z^n$ be the ordinary generating functions associated to $\mathcal{S}^0$ and $\mathcal{S}^1$. By the recursive description of the languages, $S^0(z,u)$ and $S^1(z,u)$ are given by the linear system (see e.g. [22])

$$\begin{cases} S^0(z,u) = zS^0(z,u) + 4uzS^1(z,u) + z \quad \text{and} \\ S^1(z,u) = zS^0(z,u) + 4uzS^1(z,u) + z \ . \end{cases}$$

Let $S(z,u) = S^0(z,u) + S^1(z,u) + 1$. Then the coefficient of $u^k z^n$ in $S(z,u)$, $[u^k z^n]S(z,u)$, gives the number of words of length n with $|\mathrm{lcc}(w)| = 2^{-k}$ for all n (the extra 1 comes from the case $n = 0$). By solving the linear system, we get

$$S(z,u) = 1 + \frac{2z}{1 - (1+4u)z} \ .$$

Since $[z^n]S(z,u) = 2[z^{n-1}] \sum_{m \geq 0} (1+4u)^m z^m = 2(1+4u)^{n-1}$ for $n > 0$, we see that $[u^k z^n]S(z,u) = 2 \cdot 4^k \binom{n-1}{k}$ for all $0 \leq k < n$. The coefficient of z^n in $S(z,1)$ for $n > 0$, $[z^n]S(z,1) = 2[z^{n-1}]\frac{1}{1-5z} = 2 \cdot 5^{n-1}$ gives the number of words of length $n > 0$ with $\mathrm{lcc}(w) \neq 0$.

Theorem 3.3. There are $2 \cdot 5^{n-1}$ words of length $n > 0$ with $\mathrm{lcc}(w) \neq 0$. Of these, $2^{2k+1}\binom{n-1}{k}$ have correlation $\pm 2^{-k}$ for $0 \leq k < n$.

Note that

$$\Pr_{|w|=n}[\mathrm{lcc}(w) \neq 0] = \frac{1}{4}\left(\frac{5}{8}\right)^{n-1}$$

and that

$$\Pr_{|w|=n}[-\log_2 |\mathrm{lcc}(w)| = k \mid \mathrm{lcc}(w) \neq 0] = \left(\frac{4}{5}\right)^k \left(\frac{1}{5}\right)^{n-1-k} \binom{n-1}{k}$$

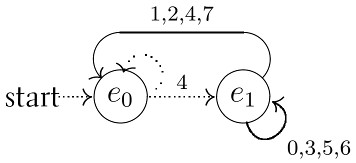
for $0 \leq k < n$. Conditioned on $\mathrm{lcc}(w) \neq 0$, $-\log_2 |\mathrm{lcc}(w)|$ is thus binomially distributed with mean $\frac{4}{5}(n-1)$ and variance $\frac{4}{25}(n-1)$ for words of length $n > 0$.

3.3 COMPUTING lcc

The linear representation L, A_k, C of lcc immediately implies that $\mathrm{lcc}(x)$ can be computed in time $O(|x|)$. Due to the simplicity of the linear representation, it can in fact be computed in time $O(\log|x|)$. For simplicity, we assume that $|x|$ is a power of 2 (if not, x can be padded with zeros, since $\mathrm{lcc}(0x) = \mathrm{lcc}(x)$).

We will use a standard n-bit RAM model of computation consisting of n-bit memory cells, and unit cost logical and arithmetic operations and conditional branches. Specifically, we will use bitwise and ($\wedge$), or ($\vee$), exclusive-or ($\oplus$) and negation ($\bar{\cdot}$), logical shifts ($\ll$ and $\gg$), and addition and subtraction modulo 2^n ($\boxplus$ and $\boxminus$). As a notational convenience, we allow our algorithms to return values of the form $s2^{-k}$, where $s \in \{0, 1, -1\}$. In our RAM model, this can be handled by returning s and k in two registers.

Let x be an octal word of length n, and let $u \leftarrow v, w$ be the corresponding linear approximation. We will modify the automaton (3.1) by merging states e_0 and 0. This give the following automaton.



Let $\mathrm{state}(b, x) \in \mathbf{F}_2^n$ be such that $\mathrm{state}(b, x)_i = 1$ if and only if the last transition of the modified automaton, when it has read the string $x_{n-1} \cdots x_i$ starting from state e_b, was *from* state e_1. Let $\mathrm{w_h}(z) = |\{i \mid z_i \neq 0\}|$ denote the Hamming weight of z. It is easy to see that $\mathrm{lcc}(x) = 0$ if and only if $\overline{\sigma} \wedge (v \vee w) \neq 0$, where $\sigma = \mathrm{state}(0, x)$. If $\mathrm{lcc}(x) \neq 0$, $\mathrm{lcc}(x) = (-1)^s 2^{-k}$, where $s = \mathrm{w_h}(v \wedge w) \bmod 2$ is the parity of $v \wedge w$ and $k = \mathrm{w_h}(\sigma)$ is the Hamming weight of $\mathrm{state}(0, x)$. That is,

$$\mathrm{lcc}(x) = \begin{cases} 0 & \text{if } \overline{\sigma} \wedge (v \vee w) \neq 0 \text{ and} \\ (-1)^{\mathrm{w_h}(v \wedge w)} 2^{-\mathrm{w_h}(\sigma)} & \text{otherwise} \end{cases} , \tag{3.2}$$

where $\sigma = \mathrm{state}(0, x)$. For example, when $x = 4362045$, and thus $u = 1010011$, $v = 0111000$ and $w = 0100001$, we have $\sigma = \mathrm{state}(0, x) = 0111001$, $\overline{\sigma} \wedge (v \vee w) = 0$, $\mathrm{w_h}(v \wedge w) = 1$ and $\mathrm{w_h}(\sigma) = 4$. It follows that $\mathrm{lcc}(4362045) = -2^{-4}$.

Note that $\mathrm{state}(b, x)$ can be recursively computed as follows. When $|x| = 1$, $\mathrm{state}(b, x) = b$. When $|x| > 1$, write $x = x^L x^R$ with $|x^L|, |x^R| > 0$. Then

$$\mathrm{state}(b, x) = (\mathrm{state}(b, x^L), \mathrm{state}(b', x^R)) \ , \tag{3.3}$$

where $b' = 1$ if and only if $\mathrm{state}(b, x^L)_0 = 0$ and $x^L_0 = 4$, or $\mathrm{state}(b, x^L)_0 = 1$ and $x^L_0 \in \{0, 3, 5, 6\}$. This observation leads to the following algorithm for computing $\mathrm{state}(b, x)$. We will maintain two registers σ_0 and σ_1 for $\mathrm{state}(0, \cdot)$ and $\mathrm{state}(1, \cdot)$, respectively. During the ith step of the algorithm, σ_b, x (and hence u, v, w) are considered to consist of blocks of length 2^{i+1}. Let σ'_b and x' be one of these blocks. At the end of the ith step, the algorithm ensures that $\sigma'_b = \mathrm{state}(b, x')$. This can be done inductively by combining the upper half of σ'_b with the lower half of either σ'_0 or σ'_1 according to (3.3). By applying this update rule to all blocks in parallel, we obtain the following $\Theta(\log n)$-time algorithm.

Theorem 3.4. Let n be a power of 2, let $\mathrm{mask}(i) \in \mathbf{F}_2^n$ consist of blocks of 2^i ones and zeros starting from the least significant end (for example, $\mathrm{mask}(1) = 0011 \cdots 0011$) and let $u, v, w \in \mathbf{F}_2^n$. The following algorithm computes the correlation $\mathrm{lcc}(u, v, w)$ using $\Theta(\log n)$ time and constant space in addition to the masks $\mathrm{mask}(i)$.

1. Initialise $m = 1010 \cdots 1010$, $s_0 = u \wedge \overline{v} \wedge \overline{w}$, $s_1 = \overline{u \oplus v \oplus w}$, $\sigma_0 = 0$ and $\sigma_1 = 11 \cdots 11$.

2. For $i = 0, \ldots, \log_2 n - 1$, do

   (a) Set $t_b \leftarrow ((\overline{\sigma_b} \wedge s_0) \vee (\sigma_b \wedge s_1)) \wedge m$ for $b = 0, 1$.
   (b) Set $t_b \leftarrow t_b \boxminus (t_b \gg 2^i)$ for $b = 0, 1$.
   (c) Set $r_b \leftarrow (\sigma_b \wedge \overline{\mathrm{mask}(i)}) \vee (\sigma_0 \wedge \overline{t_b} \wedge \mathrm{mask}(i)) \vee (\sigma_1 \wedge t_b)$ for $b = 0, 1$.
   (d) Set $\sigma_b \leftarrow r_b$ for $b = 0, 1$.
   (e) Set $m \leftarrow (m \gg 2^i) \wedge \overline{\mathrm{mask}(i+1)}$.

3. If $\overline{\sigma_0} \wedge (v \vee w) \neq 0$, return 0.

4. Otherwise, return $(-1)^{\mathrm{w_h}(v \wedge w)} 2^{-\mathrm{w_h}(\sigma_0)}$.

Note that the masks $\mathrm{mask}(i)$ and the values of m in the algorithm only depend on n.

Proof. Since the Hamming weight can be computed in time $O(\log n)$, the algorithm clearly terminates in time $\Theta(\log n)$ and uses constant space in addition to the masks $\mathrm{mask}(i)$. Let $m(i) \in \mathbf{F}_2^n$ be such that $m(i)_\ell = 1$ if and only if $\ell - 2^i$ is a nonnegative multiple of 2^{i+1} (e.g. $m(1) = 0100 \cdots 01000100$), and let

$$\sigma_b(i) = (\mathrm{state}(b, x_{n/2^i-1}), \ldots, \mathrm{state}(b, x_0)) \enspace ,$$

where $x = x_{n/2^i-1} \cdots x_0$ is the word associated to $(u \leftarrow v, w)$ and $|x_\ell| = 2^i$. We show by induction on i that $m = m(i)$ and $\sigma_b = \sigma_b(i)$ at the start of the ith iteration of the for-loop. For $i = 0$, this clearly holds, so let $i \geq 0$. Consider the words x and σ_b split into 2^{i+1}-bit blocks and let x' and σ'_b be one of these blocks. After step 2a, $t_{b,\ell} = 1$ if and only if $\ell - 2^i$ is a nonnegative multiple of 2^{i+1} and either $\sigma_{b,\ell} = 0$ and $u_\ell 4 + v_\ell 2 + w_\ell = 4$ or $\sigma_{b,\ell} = 1$ and $u_\ell 4 + v_\ell 2 + w_\ell \in \{0, 3, 5, 6\}$. After step 2b, a block of the form $\chi 00 \cdots 0$ has been transformed to a block of the form $0 \chi\chi \cdots \chi$ in t_b. In step 2c, the upper half of σ'_b is concatenated with the lower half of σ'_c, where $c = 1$ if and only if $\sigma_{b,\ell} = 0$ and $u_\ell 4 + v_\ell 2 + w_\ell = 4$ or $\sigma_{b,\ell} = 1$ and $u_\ell 4 + v_\ell 2 + w_\ell \in \{0, 3, 5, 6\}$. By induction and (3.3), we thus have $\sigma_b = \sigma_b(i+1)$ after step 2d. Finally, $m = m(i+1)$ after step 2e. It follows that $\sigma_0 = \mathrm{state}(0, x)$ after the for-loop. By (3.2), the algorithm returns $\mathrm{lcc}(u, v, w)$. □

3.4 GENERALISATIONS TO SOME OTHER MAPPINGS

The results on lcc can easily be generalised to more complex mappings. In this section, we will focus on mappings $h \colon (\mathbf{F}_2^n)^2 \to (\mathbf{F}_2^n)^2$ of the form $h(x, y) = (f(x, y), g(x, y))$. The correlation of linear approximations of h, $\mathcal{C}(t, u \overset{h}{\leftarrow} v, w)$ can be seen as a function of hexadecimal words by writing the linear approximation $(t, u \leftarrow v, w)$ as the hexadecimal word $x = x_{n-1} \cdots x_0$, where $x_i = t_i 8 + u_i 4 + v_i 2 + w_i$. When n varies, we obtain a function from all hexadecimal words to $[-1, 1]$. We will show that $\mathcal{C}(t, u \overset{h}{\leftarrow} v, w)$ is a rational series whenever $\mathcal{C}(t \overset{f}{\leftarrow} v, w)$ and $\mathcal{C}(u \overset{g}{\leftarrow} v, w)$ are.

Let S and T be rational series over the octal words with the linear representations L, $(A_k)_k$, C and L', $(A'_k)_k$, C' of dimensions d and d', respectively.

We define the *convolution* of the linear representations to be the linear representation L^*, $(A^*_k)_k$, C^* of dimension dd', where $L^*_{ij} = L_i L'_j$,

$$(A^*_k)_{ij,k\ell} = \sum_{p,q\in\{0,1\}} (A_{k_3 4+p2+q})_{i,k}(A'_{k_2 4+(k_1\oplus p)2+(k_0\oplus q)})_{j,\ell} \ ,$$

$k = k_3 8 + k_2 4 + k_1 2 + k_0$ and $C^*_{ij} = C_i C'_j$. Note that the convolution of the linear representations defines a rational series over the hexadecimal words.

Theorem 3.5. Let $f, g\colon \mathbf{F}_2^n \times \mathbf{F}_2^n \to \mathbf{F}_2^n$ be Boolean functions, such that $\mathcal{C}(t \xleftarrow{f} v, w)$ and $\mathcal{C}(u \xleftarrow{g} v, w)$ are rational series over the octal words with linear representations L, $(A_k)_k$, C and L', $(A'_k)_k$, C' of dimensions d and d', respectively. Let $h\colon (\mathbf{F}_2^n)^2 \to (\mathbf{F}_2^n)^2$ be the Boolean function $h(x,y) = (f(x,y), g(x,y))$. Then $\mathcal{C}(t, u \xleftarrow{h} v, w)$ is a rational series over the hexadecimal words. A linear representation of dimension dd' is given by the convolution of the linear representations L, $(A_k)_k$, C and L', $(A'_k)_k$, C'.

Proof. Let L^*, $(A^*_k)_k$, C^* be the convolution of the two linear representations. Fix t, u in $\mathbf{F}_2^n$, and denote

$$\begin{aligned} \hat{F}_i^m(v,w) &= (A_{x_{m-1}} \cdots A_{x_0} C)_i \quad \text{and} \\ \hat{G}_j^m(v,w) &= (A'_{y_{m-1}} \cdots A'_{y_0} C')_j \ , \end{aligned}$$

where x and y are the octal words associated to the linear approximations $(t \leftarrow v, w)$ and $(u \leftarrow v, w)$, respectively. Let $z = z_{n-1} \cdots z_0$ be the hexadecimal word $z_i = t_i 8 + u_i 4 + v_i 2 + w_i$, and let $\xi = \xi_{n-1} \cdots \xi_0$ be the word $\xi_i = v_i 2 + w_i$. We will express $\hat{F}_i^{m+1} * \hat{G}_j^{m+1}$ in terms of of $\hat{F}_k^m * \hat{G}_\ell^m$. By some abuse of notation,

$$\begin{aligned} (\hat{F}_i^{m+1} * G_j^{m+1})(v,w) &= \sum_{r,s\in\mathbf{F}_2^{m+1}} \hat{F}_i^{m+1}(r,s)\hat{G}_j^{m+1}(r+v, s+w) \\ &= \sum_{p=0}^{3} \sum_{r,s\in\mathbf{F}_2^m} \sum_{k,\ell} (A_{t_m 4+p})_{ik} \hat{F}_k^m(r,s) (A'_{u_m 4+(p\oplus\xi_m)})_{j\ell} \hat{G}_\ell^m(r+v, s+w) \\ &= \sum_{k,\ell} \sum_{p} (A_{t_m 4+p})_{ik} (A'_{u_m 4+(p\oplus\xi_m)})_{j\ell} \sum_{r,s} \hat{F}_k^m(r,s)\hat{G}_\ell^m(r+v, s+w) \\ &= \sum_{k,\ell} \left(\sum_{p} (A_{t_m 4+p})_{ik} (A'_{u_m 4+(p\oplus\xi_m)})_{j\ell} \right) (\hat{F}_k^m * \hat{G}_\ell^m)(v,w) \\ &= \sum_{k,\ell} (A^*_{z_m})_{ij,k\ell} (\hat{F}_k^m * \hat{G}_\ell^m)(v,w) \ . \end{aligned}$$

It follows by induction that $(\hat{F}_i^n * \hat{G}_j^n)(v,w) = A^*_{z_{n-1}} \cdots A^*_{z_0} C^*$. Since $\mathcal{C}(t, u \xleftarrow{h} v, w) = ((L\hat{F}^n) * (L'\hat{G}^n))(v,w) = L^*(F^n * G^n)(v,w)$, the result follows. □

This result and its proof can easily be generalised to functions of the form $f\colon (\mathbf{F}_2^n)^m \to (\mathbf{F}_2^n)^k$ and $g\colon (\mathbf{F}_2^n)^m \to (\mathbf{F}_2^n)^\ell$ for $k, m, \ell > 0$.

Since the mapping $(x,y) \mapsto (2^k x, 2^\ell y) \pmod{2^n}$, $k, \ell \geq 0$, is $\mathbf{F}_2$-linear, the results on lcc are easy to generalise to mappings of the form $(x,y) \mapsto$

$2^k x \pm 2^\ell y \bmod 2^n$. Using Theorem 3.5 we can then obtain linear representations of the correlation of linear approximations of all functions of the form

$$(x, y) \mapsto \begin{pmatrix} 2^{k_{11}} & \pm 2^{k_{12}} \\ 2^{k_{21}} & \pm 2^{k_{22}} \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \pmod{2^n} ,$$

where $k_{ij} \geq 0$ and the signs are independent, although the dimensions of the straightforward linear representations will be large. This class of functions in particular includes the Pseudo-Hadamard transform (PHT) $\mathrm{pht}\colon (\mathbf{F}_2^n)^2 \to (\mathbf{F}_2^n)^2$ defined by the expression

$$\mathrm{pht}(x, y) = \begin{pmatrix} 2 & 1 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = (2x + y, x + y) \pmod{2^n}$$

over $\mathbf{Z}_{2^n}$. The Pseudo-Hadamard transform is used e.g. in the block ciphers [32, 33, 47].

4 THE ADDITIVE DIFFERENTIAL PROBABILITY OF XOR

When studying the additive differential probability of exclusive-or, we exclusively deal with the set $\{0, 1, \ldots, 2^n - 1\}$ equipped with two group operations. On one hand, we use the usual addition modulo 2^n, which we denote by $+$. On the other hand, we identify $\{0, 1, \ldots, 2^n - 1\}$ and the set $\mathbf{F}_2^n$ of binary vectors using the natural correspondence that identifies $x_{n-1}2^{n-1} + \cdots + x_1 2 + x_0 \in \mathbf{Z}_{2^n}$ with $(x_{n-1}, \ldots, x_1, x_0) \in \mathbf{F}_2^n$. In this way the usual componentwise addition $\oplus$ in $\mathbf{F}_2^n$ (or bitwise exclusive-or) carries over to a group operation in $\{0, 1, \ldots, 2^n - 1\}$. We call the differential probability of the resulting mapping $\oplus\colon \mathbf{Z}_{2^n} \times \mathbf{Z}_{2^n} \to \mathbf{Z}_{2^n}$ the *additive differential probability* of exclusive-or and denote it by $\operatorname{adp}^{\oplus}$. In this chapter, we analyse this mapping $\operatorname{adp}^{\oplus}\colon \mathbf{Z}_{2^n}^3 \to [0, 1]$ defined by

$$\operatorname{adp}^{\oplus}(\alpha, \beta \to \gamma) = \Pr_{x,y}\left[((x + \alpha) \oplus (y + \beta)) - (x \oplus y) = \gamma\right] \ . \tag{4.1}$$

This concrete problem has been addressed (and in a rather ad hoc manner) in a few papers, including [5], but it has never been addressed completely—probably because of its “apparent complexity”.

We show that $\operatorname{adp}^{\oplus}$ can be expressed as a formal series in the sense of formal language theory with a linear representation in base 8. That is, if we write the differential $(\alpha, \beta \to \gamma)$ as an octal word $w = w_{n-1} \cdots w_1 w_0$ in a natural way, there are eight square matrices A_i, a column vector C and a row vector L, such that

$$\operatorname{adp}^{\oplus}(\alpha, \beta \to \gamma) = \operatorname{adp}^{\oplus}(w) = L A_{w_{n-1}} \cdots A_{w_1} A_{w_0} C \ .$$

This representation immediately gives us a linear-time algorithm for computing $\operatorname{adp}^{\oplus}$. A few additional properties (like the fraction $\frac{3}{7} + \frac{4}{7}8^{-n}$ of differentials with nonzero probability) can also be derived from it.

In order to study the average behaviour of $\operatorname{adp}^{\oplus}$, we introduce a sequence sbs (for *side-by-side*), by putting side-by-side the values of $\operatorname{adp}^{\oplus}(w)$ according to the length and rank in the lexicographic order of the octal word w. Using tools from analytic number theory, we derive an asymptotic expression for the sum of the first order,

$$\sum_{1 \le n < \nu} \operatorname{sbs}(n) = \nu^{2/3} G_{2/3}(\log_8 \nu) + o(\nu^{2/3}) \ ,$$

where $G_{2/3}$ is a 1-periodic continuous function. The first terms of the Fourier series of $G_{2/3}$ can be numerically computed.

Our analysis proceeds as follows. We first show that $\operatorname{adp}^{\oplus}$ is a rational series and derive a linear representation for it. This gives an efficient algorithm that computes $\operatorname{adp}^{\oplus}(w)$ in time $O(|w|)$. We then briefly discuss the distribution of $\operatorname{adp}^{\oplus}$. Section 4.3 provides an overview of the method we follow to put light on the asymptotic behaviour of $\operatorname{adp}^{\oplus}$. The rest of the chapter is a mere application of the method. The results in this chapter is joint work with Philippe Dumas and Helger Lipmaa.

4.1 THE RATIONAL SERIES $\mathrm{adp}^{\oplus}$

We will consider $\mathrm{adp}^{\oplus}$ as a function of octal words by writing the differential $(\alpha, \beta \to \gamma)$ as the octal word $w = w_{n-1} \cdots w_0$, where $w_i = 4\alpha_i + 2\beta_i + \gamma_i$. This defines $\mathrm{adp}^{\oplus}$ as a function from the octal words of length n to the interval $[0, 1]$. As n varies in the set of nonnegative integers, we obtain a function from the set of all octal words to $[0, 1]$.

In the terminology of formal language theory, the additive differential probability $\mathrm{adp}^{\oplus}$ is a formal series over the monoid of octal words with coefficients in the field of real numbers. A remarkable subset of these series is the set of *rational series* [6]. One possible characterisation of such a rational series S is the following: there exists a square matrix A_x of size $d \times d$ for each letter x in the alphabet, a row matrix L of size $1 \times d$ and a column matrix C of size $d \times 1$ such that for each word $w = w_1 \cdots w_\ell$, the value of the series is

$$S(w) = LA_{w_1} \cdots A_{w_\ell} C \ .$$

The family L, $(A_x)_x$, C is called a *linear representation* of dimension d of the rational series. In our case, the alphabet is the octal alphabet $\{0, 1, \ldots, 7\}$.

Theorem 4.1 (Linear representation of $\mathrm{adp}^{\oplus}$). The formal series $\mathrm{adp}^{\oplus}$ has the 8-dimensional linear representation L, $(A_k)_{k=0}^{7}$ and C, where the matrices are $L = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}$, $C = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}^t$,

$$A_0 = \frac{1}{4} \begin{pmatrix} 4 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

and A_k is obtained from A_0 by permuting row i with row $i \oplus k$ and column j with column $j \oplus k$: $(A_k)_{ij} = (A_0)_{i \oplus k, j \oplus k}$. (For completeness, the matrices $A_0, \ldots, A_7$ are given in Table 4.1 on page 33.) Thus, $\mathrm{adp}^{\oplus}$ is a rational series.

For example, if $(\alpha, \beta \to \gamma) = (00110, 10100 \to 01110)$, $w = 21750$ and $\mathrm{adp}^{\oplus}(\alpha, \beta \to \gamma) = \mathrm{adp}^{\oplus}(w) = LA_2A_1A_7A_5A_0C = \frac{5}{32}$. The linear representation immediately implies that $\mathrm{adp}^{\oplus}(w)$ can be computed using $O(|w|)$ arithmetic operations. Since the arithmetic operations can be carried out using $2|w|$-bit integer arithmetic, which can be implemented in constant time on a $|w|$-bit RAM model, we have

Proposition 4.1. The additive differential probability $\mathrm{adp}^{\oplus}(w)$ can be computed in time $O(|w|)$ on a standard unit cost $|w|$-bit RAM model of computation.

This can be compared with the $O(\log|w|)$-time algorithm for computing xdp^{+} from [30].

Note that the matrices $A_0, \dots A_7$ in the linear representation for $\mathrm{adp}^\oplus$ are substochastic . Thus, we could view the linear representation as a nonhomogenous Markov chain (see e.g. [11]) by adding a dummy state and dummy state transitions.

The rest of this section is devoted to the technical proof of Theorem 4.1. To prove this result, we will first give a different formulation of $\mathrm{adp}^\oplus$. For $x, y \in \{0, \dots, 2^n - 1\}$, let xy denote their componentwise product in $\mathbf{F}_2^n$ (or bitwise and). Let $\mathrm{borrow}(x, y) = x \oplus y \oplus (x - y)$ denote the borrows, as an n-tuple of bits, in the subtraction $x - y$. Alternatively, $\mathrm{borrow}(x, y)$ can be recursively defined by $\mathrm{borrow}(x, y)_0 = 0$ and $\mathrm{borrow}(x, y)_{i+1} = 1$ if and only if $x_i - \mathrm{borrow}(x, y)_i < y_i$ as integers. This can be used to *define* $\mathrm{borrow}(x, y)_n = 1$ if and only if $x_{n-1} - \mathrm{borrow}(x, y)_{n-1} < y_{n-1}$ as integers. The borrows can be used to give an alternative formulation of $\mathrm{adp}^\oplus$.

Lemma 4.1. For all $\alpha, \beta, \gamma \in \mathbf{Z}_{2^n}$,

$$\mathrm{adp}^\oplus(w) = \Pr_{x,y}[a \oplus b \oplus c = \alpha \oplus \beta \oplus \gamma] \ ,$$

where $a = \mathrm{borrow}(x, \alpha)$, $b = \mathrm{borrow}(y, \beta)$ and $c = \mathrm{borrow}(x \oplus y, (x - \alpha) \oplus (y - \beta))$.

Proof. By replacing x and y with $x - \alpha$ and $y - \beta$ in the definition (4.1) of $\mathrm{adp}^\oplus$ on page 30, we see that $\mathrm{adp}^\oplus(\alpha, \beta \to \gamma) = \Pr_{x,y}[(x \oplus y) - ((x - \alpha) \oplus (y - \beta)) = \gamma]$. Since $(x \oplus y) - ((x - \alpha) \oplus (y - \beta)) = \gamma$ if and only if $\gamma = c \oplus x \oplus y \oplus (x - \alpha) \oplus (y - \beta) = a \oplus b \oplus c \oplus \alpha \oplus \beta$ if and only if $a \oplus b \oplus c = \alpha \oplus \beta \oplus \gamma$, the result follows. □

We furthermore need the following technical lemma.

Lemma 4.2. For all x, y, α, β, γ,

$$\begin{aligned} a_{i+1} &= (aa' \oplus \alpha \oplus a'x)_i \ , \\ b_{i+1} &= (bb' \oplus \beta \oplus b'y)_i \quad \text{and} \\ c_{i+1} &= [c \oplus a' \oplus b' \oplus c(a' \oplus b') \oplus (a' \oplus b')(x \oplus y)]_i \ , \end{aligned}$$

where $a = \mathrm{borrow}(x, \alpha)$, $b = \mathrm{borrow}(y, \beta)$, $c = \mathrm{borrow}(x \oplus y, (x - \alpha) \oplus (y - \beta))$, $a' = a \oplus \alpha$ and $b' = b \oplus \beta$.

Proof. By the recursive definition of $\mathrm{borrow}(x, y)$, $\mathrm{borrow}(x, y)_{i+1} = 1$ if and only if $x_i < y_i + \mathrm{borrow}(x, y)_i$ as integers. The latter event occurs if and only if either $y_i = \mathrm{borrow}(x, y)_i$ and at least two of x_i, y_i and $\mathrm{borrow}(x, y)_i$ are equal to 1, or $y_i \neq \mathrm{borrow}(x, y)_i$ and at least two of x_i, y_i and $\mathrm{borrow}(x, y)_i$ are equal to 0. That is, $\mathrm{borrow}(x, y)_{i+1} = 1$ if and only if $y_i \oplus \mathrm{borrow}(x, y)_i \oplus \mathrm{maj}(x_i, y_i, \mathrm{borrow}(x, y)_i) = 1$, where $\mathrm{maj}(u, v, w)$ denotes the majority of the bits u, v, w. Since $\mathrm{maj}(u, v, w) = uv \oplus uw \oplus vw$, we have $\mathrm{borrow}(x, y)_{i+1} = [y \oplus \mathrm{borrow}(x, y) \oplus xy \oplus x\,\mathrm{borrow}(x, y) \oplus y\,\mathrm{borrow}(x, y)]_i$.

For a, we thus have $a_{i+1} = (\alpha \oplus a \oplus x\alpha \oplus xa \oplus \alpha a)_i = (a' \oplus a\alpha \oplus a'x)_i = [a' \oplus a(a' \oplus a) \oplus a'x]_i = (aa' \oplus \alpha \oplus a'x)_i$. The formula for b_{i+1} is completely analogous. For c, we have $c_{i+1} = [(x - \alpha) \oplus (y - \beta) \oplus c \oplus (x \oplus y)((x - \alpha) \oplus (y - \beta)) \oplus (x \oplus y)c \oplus ((x - \alpha) \oplus (y - \beta))c]_i = [x \oplus a' \oplus y \oplus b' \oplus c \oplus (x \oplus y)(x \oplus a' \oplus y \oplus b') \oplus (x \oplus y)c \oplus (x \oplus a' \oplus y \oplus b')c]_i = [c \oplus a' \oplus b' \oplus c(a' \oplus b') \oplus (a' \oplus b')(x \oplus y)]_i$. □

Table 4.1: All the eight matrices A_k in Theorem 4.1.

$$A_0 = \frac{1}{4}\begin{pmatrix} 4 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \qquad A_1 = \frac{1}{4}\begin{pmatrix} 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 4 & 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

$$A_2 = \frac{1}{4}\begin{pmatrix} 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 4 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \qquad A_3 = \frac{1}{4}\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 4 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \end{pmatrix}$$

$$A_4 = \frac{1}{4}\begin{pmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 4 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \qquad A_5 = \frac{1}{4}\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 4 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \end{pmatrix}$$

$$A_6 = \frac{1}{4}\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 & 4 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \end{pmatrix} \qquad A_7 = \frac{1}{4}\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 & 4 \end{pmatrix}$$

Proof (of Theorem 4.1). Let $(\alpha, \beta \to \gamma)$ be the differential associated with the word w. Denote $n = |w|$ and let x, y be uniformly distributed random variables in $\mathbf{Z}_{2^n}$. Denote $a = \text{borrow}(x, \alpha)$, $b = \text{borrow}(y, \beta)$ and $c = \text{borrow}(x \oplus y, (x - \alpha) \oplus (y - \beta))$. Let ξ be the octal word of borrow triples, $\xi_i = 4a_i + 2b_i + c_i$. We define ξ_n in the natural way using $\text{borrow}(u, v)_n = 1$ if and only if $u_{n-1} - \text{borrow}(u, v)_{n-1} < v_{n-1}$ as integers. For compactness, denote $\text{xor}(w) = \alpha \oplus \beta \oplus \gamma$ and $\text{xor}(\xi) = a \oplus b \oplus c$. Let $P(w, k)$ be the 8×1 substochastic matrix

$$P_j(w, k) = \Pr_{x,y}[\text{xor}(\xi) \equiv \text{xor}(w) \pmod{2^k}, \xi_k = j]$$

for $0 \leq k \leq n$. Let $M(w, k)$ be the 8×8 substochastic transition matrix

$$M_{ij}(w, k) = \Pr_{x,y}[\text{xor}(\xi)_k = \text{xor}(w)_k, \xi_{k+1} = i \mid \\ \text{xor}(\xi) \equiv \text{xor}(w) \pmod{2^k}, \xi_k = j]$$

for $0 \leq k < n$. Then $P_i(w, k + 1) = \sum_j M_{ij}(w, k) P_j(w, k)$ and thus $P(w, k + 1) = M(w, k) P(w, k)$. Note furthermore that $P(w, 0) = C$, since $a_0 = b_0 = c_0 = 0$, and that $LP(w, n) = \sum_j \Pr_{x,y}[\text{xor}(\xi) \equiv \text{xor}(w) \pmod{2^n}, \xi_n = j] = \Pr_{x,y}[\text{xor}(\xi) \equiv \text{xor}(w) \pmod{2^n}] = \text{adp}^{\oplus}(w)$, where the last equality is due to Lemma 4.1. We will show that $M(w, k) = A_{w_k}$ for all k. By induction, it follows that $\text{adp}^{\oplus}(w) = LP(w, n) = LM(w, n - 1) \cdots M(w, 0)C = LA_{w_{n-1}} \cdots A_{w_0} C$.

It remains to show that $M(w, k) = A_{w_k}$ for all k. Towards this end, let x, y be such that $\text{xor}(\xi) \equiv \text{xor}(w) \pmod{2^k}$ and $\xi_k = j$. We will count the number of ways we can choose (x_k, y_k) such that $\text{xor}(\xi)_k = \text{xor}(w)_k$ and $\xi_{k+1} = i$.

Denote $a' = a \oplus \alpha$, $b' = b \oplus \beta$ and $c' = c \oplus \gamma$. Note that $\text{xor}(\xi)_k = \text{xor}(w)_k$ if and only if $c'_k = (a' \oplus b')_k$. Under the assumption that $\text{xor}(\xi)_k = \text{xor}(w)_k$ we have $(cc' \oplus \gamma)_k = [c(a' \oplus b') \oplus c \oplus a' \oplus b']_k$. By Lemma 4.2, (x_k, y_k) must thus be a solution to $V \begin{pmatrix} x_k & y_k \end{pmatrix}^t = U$ in $\mathbf{Z}_2$, where U and V are the matrices

$$U = \begin{pmatrix} (aa' \oplus \alpha)_k \oplus a_{k+1} \\ (bb' \oplus \beta)_k \oplus b_{k+1} \\ (cc' \oplus \gamma)_k \oplus c_{k+1} \end{pmatrix} \quad \text{and} \quad V = \begin{pmatrix} a'_k & 0 \\ 0 & b'_k \\ (a' \oplus b')_k & (a' \oplus b')_k \end{pmatrix}$$

over $\mathbf{Z}_2$. If this equation has a solution, the total number of solutions is $2^{2-\text{rank}(V)}$. But $\text{rank}(V) = 0$ if and only if $a'_k = b'_k = 0$ (then there are 4 solutions) and $\text{rank}(V) = 2$ otherwise (then there is 1 solution).

The equation has a solution precisely when $\text{rank}(V) = \text{rank}(V\ U)$. From this and from the requirement that $c'_k = (a' \oplus b')_k$, we see that there is a solution exactly in the following cases.

- If $a'_k = b'_k = 0$, then $c'_k = 0$ and $\text{rank}(V) = 0$. There are solutions (4 solutions) if and only if $a_{k+1} = \alpha_k$, $b_{k+1} = \beta_k$ and $c_{k+1} = \gamma_k$.

- If $a'_k = 0$ and $b'_k = 1$ then $c'_k = 1$ and $\text{rank}(V) = 2$. There is a single solution if and only if $a_{k+1} = \alpha_k$.

- If $a'_k = 1$ and $b'_k = 0$, then $c'_k = 1$ and $\mathrm{rank}(V) = 2$. There is a single solution if and only if $b_{k+1} = \beta_k$.
- If $a'_k = 1$ and $b'_k = 1$ then $c'_k = 0$ and $\mathrm{rank}(V) = 2$. There is a single solution if and only if $c_{k+1} = \gamma_k$.

Since $j = \xi_k = 4a_k + 2b_k + c_k$ and $i = \xi_{k+1} = 4a_{k+1} + 2b_{k+1} + c_{k+1}$, the derivation so far can be summarised as

$$M_{ij}(w,k) = \begin{cases} 1 \ , & j = (\alpha_k, \beta_k, \gamma_k) \ , \ i = (\alpha_k, \beta_k, \gamma_k) \ , \\ 1/4 \ , & j = (\alpha_k, \beta_k \oplus 1, \gamma_k \oplus 1) \ , \ i = (\alpha_k, *, *) \ , \\ 1/4 \ , & j = (\alpha_k \oplus 1, \beta_k, \gamma_k \oplus 1) \ , \ i = (*, \beta_k, *) \ , \\ 1/4 \ , & j = (\alpha_k \oplus 1, \beta_k \oplus 1, \gamma_k) \ , \ i = (*, *, \gamma_k) \ , \\ 0 \ , & \text{otherwise} \ , \end{cases}$$

where we have identified the integer $4r_2 + 2r_1 + r_0$ with the binary tuple (r_2, r_1, r_0) and $*$ represents an arbitrary element of $\{0, 1\}$. From this, we see that $M(w,k) = A_0$ if $w_k = 0$ and $M_{i,j}(w,k) = M_{i \oplus w_k, j \oplus w_k}(0,k)$. That is, $M(w,k) = A_{w_k}$ for all w, k. This completes the proof. □

4.2 DISTRIBUTION OF $\mathrm{adp}^{\oplus}$

We will use notation from formal languages to describe octal words. For example, $(3+5+6)0^*$ denotes the set of words with one of $\{3,5,6\}$ followed by any number of zeros. We first consider words of the form $w0^*$ and 0^*w.

Corollary 4.1. For all octal words w, $\mathrm{adp}^{\oplus}(w0^*) = \mathrm{adp}^{\oplus}(w)$.

Proof. Follows from $A_0 C = C$. □

Corollary 4.2. Let w be a word and $a = \begin{pmatrix} a_0 & \cdots & a_7 \end{pmatrix}^t = A_{|w|-1} \cdots A_{w_0} C$. Let $\alpha = a_0$ and $\beta = a_3 + a_5 + a_6$. Let w' be a word of the form $w' = 0^* w$. Then $\mathrm{adp}^{\oplus}(w') = \alpha + \frac{\beta}{3} + \frac{8}{3} \cdot \beta \cdot 4^{-(|w'|-|w|)}$.

Proof. Using a Jordan form $J = P^{-1} A_0 P$ of A_0, it is easy to see that

$$A_0^k = 4^{-k} \begin{pmatrix} 4^k & 0 & 0 & \frac{4^k-1}{3} & 0 & \frac{4^k-1}{3} & \frac{4^k-1}{3} & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} .$$

If we let $j = |w'| - |w|$, we see that $LA_0^j a = a_0 + \frac{4^j-1}{3 \cdot 4^j}(a_3 + a_5 + a_6) + \frac{3}{4^j}(a_3 + a_5 + a_6) = \alpha + \frac{\beta}{3} + \frac{8}{3} \cdot \beta \cdot 4^{-(|w'|-|(|w|))}$. □

This means that $\mathrm{adp}^{\oplus}(0^n w)$ decreases with n and that $\mathrm{adp}^{\oplus}(0^n w) \to \alpha + \beta/3$ when $n \to \infty$. This can be compared to [30], where it was shown that $\mathrm{xdp}^{+}(00w) = \mathrm{xdp}^{+}(0w)$ for all w.

Proposition 4.2. The additive differential probability $\mathrm{adp}^{\oplus}(w)$ is nonzero if and only if w has the form $w = 0^*$ or $w = w'(3+5+6)0^*$ for any octal word w'.

Proof. Since $A_1C = A_2C = A_4C = A_7C = 0$, $\mathrm{adp}^{\oplus}(w'(1+2+4+7)0^*) = 0$. Conversely, let w be a word of the form $w = w'(3+5+6)0^*$. Let e_i be the canonical (column) basis vector with a 1 in the ith component and 0 in the others. By direct computation, $\ker A_0 = \ker A_3 = \ker A_5 = \ker A_6 = \langle e_1, e_2, e_4, e_7\rangle$ and $\ker A_1 = \ker A_2 = \ker A_4 = \ker A_7 = \langle e_0, e_3, e_5, e_6\rangle$. For all i and $j \neq i$, $e_j \notin \ker A_i$, it can be seen that $A_ie_i = e_i$ and that A_ie_j has the form $A_ie_j = (e_k + e_\ell + e_m + e_n)/4$, where $k \neq \ell$, $m \neq n$, $e_k, e_\ell \in \ker A_0$ and $e_m, e_n \in \ker A_1$. Since $C = e_0$, we see by induction that $A_{w_i} \cdots A_{w_0}C \notin \ker A_{w_{i+1}}$ for all i. Thus, $\mathrm{adp}^{\oplus}(w) \neq 0$. □

A complete determination of the distribution of $\mathrm{adp}^{\oplus}$ falls out of scope of this chapter. We will restrict ourselves to some of the most important results. First, we turn to the fraction of *possible* differentials—that is, differentials with $\mathrm{adp}^{\oplus}(w) \neq 0$.

Proposition 4.3. For all $n \geq 0$, $\Pr_{|w|=n}[\mathrm{adp}^{\oplus}(w) \neq 0] = \frac{3}{7} + \frac{4}{7} \cdot \frac{1}{8^n}$.

Proof. According to Proposition 4.2, $\mathrm{adp}^{\oplus}(w) \neq 0$ if and only if it is the zero word or has form $w = w'\xi 0^k$, where w' is an arbitrary word of length $n-k-1$ and $\xi \in \{3, 5, 6\}$. For a fixed value of k, we can choose w' and ξ in $3 \cdot 8^{n-k-1}$ ways. Thus, there are $1 + \sum_{k=0}^{n-1} 3 \cdot 8^{n-k-1} = \frac{4}{7} + \frac{3}{7} \cdot 8^n$ words with $\mathrm{adp}^{\oplus}(w) \neq 0$ in total. □

This result can be compared with [30, Theorem 2], which states that the corresponding probability for xdp^+ is $\Pr_{|w|=n}[\mathrm{xdp}^+(w) \neq 0] = \frac{4}{7}\left(\frac{7}{8}\right)^n$. This means, in particular, that

$$\Pr_{|w|=n}[\mathrm{adp}^{\oplus}(w) \neq 0] \to \frac{3}{7} \quad \text{while} \quad \Pr_{|w|=n}[\mathrm{xdp}^{+}(w) \neq 0] \to 0$$

as $n \to \infty$. Since the number of possible differentials is larger for $\mathrm{adp}^{\oplus}$ than for xdp^+, then the average possible differential will obtain a smaller value.

Finally, note that if $w = (0+3+5+6)0^*$ then clearly $\mathrm{adp}^{\oplus}(w) = 1$. On the other hand, for any $\xi \in \{0, \ldots, 7\}$, $\mathrm{adp}^{\oplus}(\xi w) \leq 1/2$. Therefore, $\Pr_{|w|=n}[\mathrm{adp}^{\oplus}(w) = 1] = 4 \cdot 8^{-n}$, and $\Pr_{|w|=n}[\mathrm{adp}^{\oplus}(w) = k] = 0$ if $k \in]1/2, 1[$. One can further establish easily that $\mathrm{adp}^{\oplus}(w) = 1/2$ if and only if $w = \Sigma(0+3+5+6)0^*$, where $\Sigma = 0 + 1 + \cdots + 7$.

4.3 THE AVERAGE BEHAVIOUR OF $\mathrm{adp}^{\oplus}$: OVERVIEW

We will give a detailed analysis of the asymptotic average behaviour of the formal series $\mathrm{adp}^{\oplus}(w)$. Towards this end, we study the rational sequence $\mathrm{sbs}(n)$ (for *side-by-side*) obtained by putting the values of $\mathrm{adp}^{\oplus}(w)$ side-by-side. Next, we derive an asymptotic expression for the sum $\sum_{n<\nu} \mathrm{sbs}(n)$. This will give detailed information about the average behaviour of $\mathrm{sbs}(n)$ and thus of $\mathrm{adp}^{\oplus}(w)$. The analysis proceeds as follows.

We first view the family $(\mathrm{adp}^{\oplus}(w)/4^n)_{|w|=n}$ as a probability distribution on the real segment $[0,1]$ by interpreting the word w as the real number whose octal expansion is $(0.w)_8$. For each n, we have an associated distribution function F_n. Using the linear representation for $\mathrm{adp}^{\oplus}(w)$, we prove a limit theorem stating that the sequence of distribution functions converges to a continuous distribution function F. This limit theorem translates to a formula

$$\sum_{n<\nu} \mathrm{sbs}(n) = \nu^{2/3} G_{2/3}(\log_8 \nu) + o(\nu^{2/3}) \ , \qquad (4.2)$$

where $G_{2/3}$ is a 1-periodic continuous function. The important thing here is the existence of $G_{2/3}$. (Section 4.4.)

Second, the linear representation of $\mathrm{adp}^{\oplus}(w)$ gives a 17-dimensional linear representation of the 8-rational sequence $\mathrm{sbs}(n)$. We take the 17 sequences associated with the linear representation for $\mathrm{sbs}(n)$ by taking each canonical basis vector of $\mathbf{Q}^{17}$ as the column vector. Let U_n be the row vector of these sequences and let $U(s)$ be its Dirichlet series. Each sequence is bounded and the Dirichlet series have abscissa of convergence not greater than 1. The function $U(s)$ is analytic for $\sigma > 1$ and satisfies the functional equation

$$U(s)(I_{17} - 8^{-s}Q) = \nabla U(s) \ ,$$

where Q is the sum of the square matrices in the linear representation for $\mathrm{sbs}(n)$ and the function $\nabla U(s)$ is analytic for $\sigma > 0$. This formula provides a meromorphic extension of $U(s)$ to $\sigma > 0$. The rightmost (possible) singularities have $\sigma = 2/3$. A change of coordinates using a Jordan form $J = P^{-1}QP$ transforms U_n and $U(s)$ to the sequence V_n and its Dirichlet series $V(s)$. This allows us to show that $2/3$ indeed is a singularity for the first component $v^1(s)$ of $V(s)$. The singularities of the other components have $\sigma \le 1/3$. Finally, the order of growth for $U(s)$ is at most $1-\sigma$ for $0 < \sigma < 1$ and 0 for $\sigma > 1$. (Section 4.5.)

Third, we apply a Mellin-Perron formula to get an integral expression for the sums of the second order of $\mathrm{sbs}(n)$. The integral is evaluated using the residue theorem by pushing the vertical line of integration to the left. This gives the asymptotic expansion

$$\sum_{1\le n<\nu} \sum_{k=1}^{n-1} \mathrm{sbs}(k) \underset{\nu\to\infty}{=} \nu^{5/3} H_{5/3}(\log_8 \nu) + \nu^{4/3} H_{4/3}(\log_8 \nu) + O(\nu^{1+\epsilon}) \ ,$$

where $H_{5/3}$ and $H_{4/3}$ are 1-periodic continuous functions and $\epsilon \in]0,1/3[$. A pseudo-Tauberian theorem combined with (4.2) gives the Fourier series

$$G_{2/3}(\lambda) = \frac{1}{\ln 8} \sum_{k\in\mathbf{Z}} \frac{\nabla v^1(2/3 + k\chi)}{2/3 + k\chi} e^{2\pi i k \lambda} \ .$$

The first terms of the series can be numerically computed. The Fourier series converges in the sense of Cesàro. (Section 4.6.)

4.4 LIMIT THEOREM AND PERIODICITY

Let L, C, and $A_0, \ldots, A_7$ be the linear representation of $\operatorname{adp}^{\oplus}(w)$, and let $B_r = A_r/4$ for $r = 0, \ldots, 7$. For compactness, we will denote $B_w = B_{w_{|w|-1}} \cdots B_{w_0}$. For each integer $n > 0$, we define a probability distribution on the real segment $[0, 1]$ by its distribution function

$$F_n(x) = 4^{-n} \sum_{\substack{|w|=n \\ (w)_8 < 8^n x}} \operatorname{adp}^{\oplus}(w) = \sum_{\substack{|w|=n \\ (w)_8 < 8^n x}} L B_w C \enspace ,$$

where $(w)_8 = w_{|w|-1} 8^{|w|-1} + w_{|w|-2} 8^{|w|-2} + \cdots + w_0$ is the octal integer represented by w. This is indeed a probability distribution, since

$$\sum_{|w|=n} \operatorname{adp}^{\oplus}(w) = \sum_{\alpha,\beta} \sum_{\gamma} \operatorname{adp}^{\oplus}(\alpha, \beta \to \gamma) = \sum_{\alpha,\beta} 1 = 4^n \enspace .$$

Recall that the characteristic function $\phi\colon \mathbf{R} \to \mathbf{C}$ of a (probability) distribution function F is its Fourier transform defined by $\phi(t) = \int e^{itx} dF(x)$, where the integral is an Lebesgue-Stieltjes integral. For a discrete probability distribution, this reduces to $\phi(t) = \sum_x e^{itx} \Pr[X = x]$. The characteristic function ϕ_n of F_n is given by

$$\phi_n(t) = \sum_{|w|=n} \exp\left(\frac{it(w)_8}{8^n}\right) L B_w C = L \Phi_n(t) C \enspace ,$$

where the 8×8 matrix $\Phi_n(t)$ is defined by $\Phi_n(t) = \sum_{|w|=n} \exp\left(\frac{it(w)_8}{8^n}\right) B_w$. If we write the word w of length $n+1$ as $w = w'r$ with $r \in \{0, \ldots, 7\}$, we have

$$\begin{aligned} \Phi_{n+1}(t) &= \sum_{r=0}^{7} \sum_{|w'|=n} \exp\left(\frac{it(8(w')_8 + r)}{8^{n+1}}\right) B_{w'} B_r \\ &= \Phi_n(t) \sum_{r=0}^{7} \exp\left(\frac{itr}{8^{n+1}}\right) B_r \enspace . \end{aligned}$$

Let $Q(t)$ to be the 8×8 matrix $Q(t) = \sum_{r=0}^{7} e^{itr} B_r$. Then $\Phi_1(t) = Q\left(\frac{t}{8}\right)$ and $\Phi_{n+1}(t) = \Phi_n(t) Q\left(\frac{t}{8^{n+1}}\right)$. Thus,

$$\Phi_n(t) = Q\left(\frac{t}{8}\right) Q\left(\frac{t}{8^2}\right) \cdots Q\left(\frac{t}{8^n}\right) \enspace . \tag{4.3}$$

Proposition 4.4. The sequence of matrices (Φ_n) defined by (4.3) converges uniformly for $t \in [-\tau, \tau]$ for all $\tau > 0$.

Proof. Note that $Q(0)$ is similar to the diagonal matrix

$$D = \operatorname{diag}(1, 1/2, 1/2, 1/2, 1/4, 1/4, 1/4, 1/4) \enspace .$$

Hence it is sufficient to consider the following situation. We have a matrix $D(t) = \sum_{r=0}^{7} e^{irt} C_r$ for some fixed 8×8 matrices $C_0, \ldots, C_7$ such that $D(0) = D$. We want to prove that the sequence of matrices

$$\Psi_n(t) = D\left(\frac{t}{8}\right) D\left(\frac{t}{8^2}\right) \cdots D\left(\frac{t}{8^n}\right)$$

is uniformly convergent for $|t| \leq \tau$.

Let $\|\cdot\|$ denote the matrix norm $\|A\| = \|A\|_1 = \max_j \sum_i |a_{ij}|$. We divide $D(t)$ and $\Psi_n(t)$ into blocks as

$$D(t) = \begin{pmatrix} p(t) & q(t) \\ r(t) & s(t) \end{pmatrix} \quad \text{and} \quad \Psi_n(t) = \begin{pmatrix} P_n(t) & Q_n(t) \\ R_n(t) & S_n(t) \end{pmatrix} \ ,$$

where $p(t)$ and $P_n(t)$ are complex numbers, $q(t)$ and $Q_n(t)$ are row vectors of size 1×7, $r(t)$ and $R_n(t)$ are column vectors of size 7×1, and $s(t)$ and $S_n(t)$ are square matrices of size 7×7. Since $|e^{iu} - 1| \leq |u|$ for all real u, we obtain the bound

$$\|D(t)\| \leq \|D(0)\| + \|D(t) - D(0)\| \leq 1 + \gamma|t|$$

for some constant γ. From this inequality, we see that

$$\|\Psi_n(t)\| \leq \prod_{k=1}^{n} \left(1 + \gamma \frac{|t|}{8^k}\right) \ .$$

Since the last product is a partial product of a convergent infinite product, it follows that all the matrices $\Psi_n(t)$ are uniformly bounded for $|t| \leq \tau$. As a consequence, there is a constant Γ such that

$$\|P_n(t)\|, \|Q_n(t)\|, \|R_n(t)\|, \|S_n(t)\| \leq \Gamma$$

for all n and $|t| \leq \tau$.

Using the formula $\Psi_{n+1}(t) = \Psi_n(t) D(t/8^{n+1})$, we obtain the recurrence

$$\begin{aligned} P_{n+1}(t) &= P_n(t)p(t/8^{n+1}) + Q_n(t)r(t/8^{n+1}) \ , \\ R_{n+1}(t) &= R_n(t)p(t/8^{n+1}) + S_n(t)r(t/8^{n+1}) \ , \\ Q_{n+1}(t) &= P_n(t)q(t/8^{n+1}) + Q_n(t)s(t/8^{n+1}) \ , \\ S_{n+1}(t) &= R_n(t)q(t/8^{n+1}) + S_n(t)s(t/8^{n+1}) \ . \end{aligned}$$

Since $\|q(t)\|$ is uniformly bounded, we may assume that γ is large enough so that $\|q(t)\| \leq \gamma|t|$. Since $s(u)$ is continuous at $u = 0$, there is a constant $1/2 \leq \rho < 1$ such that $\|s(t/8^{n+1})\| \leq \rho$ for all sufficiently large n. From the recurrence for $S_{n+1}(t)$, we thus have

$$\|S_{n+1}(t)\| \leq \Gamma\gamma \frac{|t|}{8^{n+1}} + \|S_n(t)\| \rho \leq \|S_n(t)\| \frac{1+\rho}{2}$$

for all sufficiently large n. Since $\frac{1+\rho}{2} < 1$, we conclude that the sequence $S_n(t)$ converges towards the 7×7 0-matrix. Moreover, the convergence is uniform for $|t| \leq \tau$. By an analogous argument, we see that the sequence $Q_n(t)$ converges uniformly to the 1×7 0-matrix for $|t| \leq \tau$.

Note that $p(u) = 1 + O(u)$ and thus

$$p(t/8^n) \underset{n \to \infty}{=} 1 + O(1/8^n)$$

uniformly for $|t| \leq \tau$. It follows that the infinite product $\prod_{k=1}^{\infty} p(t/8^k)$ is uniformly convergent for $|t| \leq \tau$. Denote

$$\pi_n(t) = \prod_{k=n_0}^{n} p(t/8^k) \ ,$$

where n_0 is such that all $\pi_n(t) \neq 0$ for $|t| \leq \tau$. Then

$$\frac{1}{\beta} \leq \|\pi_n(t)\| \leq \beta$$

for some positive constant β. Write $P_n(t) = \pi_n \delta_n(t)$. Then the recurrence for $P_{n+1}(t)$ becomes

$$\pi_{n+1}(t)\delta_{n+1}(t) = \pi_{n+1}(t)\delta_n(t) + Q_n(t)r(t/8^{n+1}) \ .$$

This gives

$$\delta_{n+1}(t) - \delta_n(t) = \frac{Q_n(t)r(t/8^{n+1})}{\pi_{n+1}(t)}$$

and hence

$$\|\delta_{n+1}(t) - \delta_n(t)\| \leq \Gamma\gamma\frac{|t|}{8^{n+1}}\beta \ .$$

It follows that the series of general term $\delta_{n+1}(t) - \delta_n(t)$ converges uniformly and hence the sequence $P_n(t)$ converges uniformly for $|t| \leq \tau$. An analogous argument shows that the sequence $R_n(t)$ converges uniformly for $|t| \leq \tau$.

We conclude that the sequence $\Psi_n(t)$ converges uniformly for $|t| \leq \tau$ for all $\tau > 0$. It follows that the same result holds for the sequence $\Phi_n(t)$. □

Since $\phi_n(t) = L\Phi_n(t)C$, Proposition 4.4 implies that the sequence of characteristic functions $\phi_n(t)$ converges uniformly for $t \in [-\tau, \tau]$ for all τ. As a consequence, the limit function is continuous at $t = 0$.

We will next show that (F_n) converges to a continuous distribution function F. For this, we need the following result on characteristic functions [31, Theorem 3.6.1].

Fact 4.1. Let (F_n) be a sequence of distribution functions and let (ϕ_n) be the sequence of corresponding characteristic functions. Then the sequence (F_n) converges weakly to a distribution function $F(x)$ if and only if the sequence (ϕ_n) converges to a function $\phi(t)$ that is continuous at $t = 0$. The limiting function ϕ is then the characteristic function of $F(x)$. (Weak convergence means that $\lim_{n\to\infty} F_n(x) = F(x)$ for all continuity points x of $F(x)$.)

Recall that the maximum jump or saltus of a function f is the maximum (if it exists) of $\lim_{x\to x_0^+} f(x) - \lim_{x\to x_0^-} f(x)$ taken over all discontinuity points x_0 of f. The following result follows from [31, Theorem 3.7.6].

Fact 4.2. Let (F_n) be a sequence that converges weakly to a distribution function F. Let p_n denote the maximum jump (saltus) of F_n. If the infinite product $\prod_{n=1}^{\infty} p_n$ diverges to zero, then F is continuous.

By Fact 4.1, (F_n) converges weakly to a distribution function F. Let p_n denote the maximum jump of F_n. Note that $\mathrm{adp}^{\oplus}(w)$ is 0 for the word $w_1 = 27^{n-1}$ and is 1 for the word $w_2 = 30^{n-1}$. Since w_1 and w_2 are consequent, they give a maximum jump of $p_n = 4^{-n}$ for F_n. Thus, $\prod_{n=1}^{\infty} p_n$ diverges to 0 and Fact 4.2 implies that the limit function F is continuous. We have obtained

Proposition 4.5. There exists a continuous distribution function F such that the summation function of $\mathrm{adp}^{\oplus}$ for word lengths n satisfies

$$\sum_{\substack{|w|=n\\(w)_8<8^n x}} \mathrm{adp}^{\oplus}(w) \underset{n\to\infty}{=} 4^n \cdot (F(x) + o(1))$$

for all x.

Let $\mathrm{sbs}(n)$ (for *side-by-side*) be the sequence obtained by putting side-by-side the values of $\mathrm{adp}^{\oplus}(w)$ according to their length and rank in the lexicographic order of the octal word w. Note that the words of length n corresponds to the integer interval from $(8^n - 1)/7$ to $(8^{n+1} - 1)/7 - 1$, since $\sum_{n=0}^{n-1} 8^n = (8^n - 1)/7$. For all real $x \in [0, 1[$, we thus have

$$\begin{aligned}
\sum_{k<\frac{8^n-1}{7}+8^n x} \mathrm{sbs}(k) &= \sum_{k=0}^{n-1} \sum_{|w|=k} \mathrm{adp}^{\oplus}(w) + \sum_{\substack{|w|=n\\(w)_8<8^n x}} \mathrm{adp}^{\oplus}(w)\\
&= \sum_{k=0}^{n-1} 4^k + \sum_{\substack{|w|=n\\(w)_8<8^n x}} \mathrm{adp}^{\oplus}(w)\\
&\underset{n\to\infty}{=} \frac{4^n - 1}{3} + 4^n(F(x) + o(1))\\
&= 4^n \left(\frac{1}{3} + F(x) + o(1)\right) \quad .
\end{aligned}$$

Let $\nu = \frac{8^n-1}{7} + 8^n x$. Since

$$\nu^{2/3} = \frac{4^n}{7^{2/3}} \left(1 + 7x - \frac{1}{8^n}\right)^{2/3} \underset{n\to\infty}{=} \frac{4^n}{7^{2/3}} (1 + 7x)^{2/3} + O\left(\frac{1}{2^n}\right) \quad ,$$

we obtain

$$\frac{1}{\nu^{2/3}} \sum_{k<\nu} \mathrm{sbs}(k) \underset{n\to\infty}{=} 7^{2/3} \frac{\frac{1}{3} + F(x)}{(1 + 7x)^{2/3}} + o(1) \quad .$$

Note that this also can be written as

$$\sum_{k<\nu} \mathrm{sbs}(k) \underset{\nu\to\infty}{=} (7\nu)^{2/3} \frac{\frac{1}{3} + F(x)}{(1 + 7x)^{2/3}} + o(\nu^{2/3})$$

where $\log_8 7\nu = n + \log_8(1 + 7x) + O\left(\frac{1}{8^n}\right)$.

Let $\{z\} = z - \lfloor z \rfloor$ denote the fractional part of z. Since $0 \le x < 1$, we have $1 \le 1 + 7x < 8$ and asymptotically $\{\log_8 7\nu\} =_{\nu\to\infty} \log_8(1 + 7x)$. We conclude that

$$\sum_{n<\nu} \mathrm{sbs}(n) \underset{\nu\to\infty}{=} \nu^{2/3} G_{2/3}(\log_8 \nu) + o(v^{2/3}) \quad ,$$

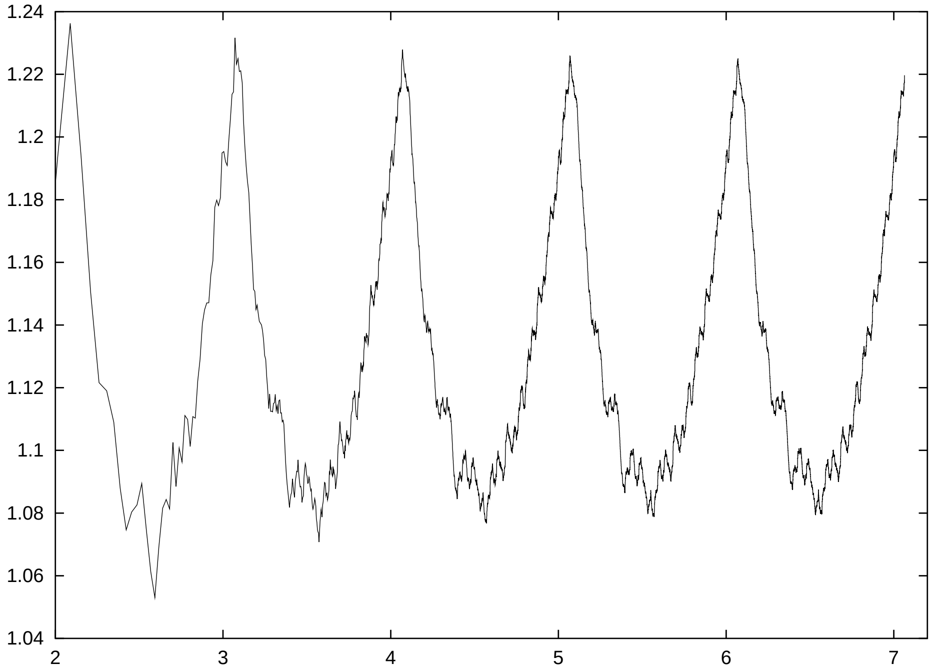


Figure 4.1: Rough plot of $\frac{1}{\nu^{2/3}}\sum_{n<\nu}\mathrm{sbs}(n)$ (y-axis) against $\log_8 \nu$ (x-axis). Note the clear periodic behaviour already for small values of ν.

where $G_{2/3}$ is a 1-periodic continuous function. Moreover, $G_{2/3}(0) = \frac{7^{2/3}}{3}$, since $F(0) = 0$, and $7^{2/3}/12 \le G_{2/3}(z) \le 4 \cdot 7^{2/3}/3$ for all z, since F takes values in $[0, 1]$.

Theorem 4.2. There exists a strictly positive 1-periodic continuous function $G_{2/3}$ such that

$$\sum_{n<\nu} \mathrm{sbs}(n) \underset{\nu\to\infty}{=} \nu^{2/3} G_{2/3}(\log_8 \nu) + o(\nu^{2/3}) \ .$$

The result is illustrated in Figure 4.1. In the following, we will give a quantitative version of this qualitative result.

4.5 DIRICHLET SERIES

The precise study of the summation function of $\mathrm{sbs}(n)$ relies on the use of its Dirichlet series. Recall that the Dirichlet series $f(s)$ of a sequence (a_n) is defined by

$$f(s) = \sum_{n=1}^{\infty} \frac{a_n}{n^s}$$

for $s \in \mathbf{C}$. Following tradition, we will write $s = \sigma + it$ where $\sigma, t \in \mathbf{R}$. Each Dirichlet series has an abscissa of convergence σ_c such that $f(s)$ converges for $\sigma > \sigma_c$ and diverges for $\sigma < \sigma_c$. If $f(s)$ diverges or converges for all s, $\sigma_c = \pm\infty$. The following result [3, Theorem 8.2] can be used for computing the abscissa of convergence.

Fact 4.3. If the abscissa of convergence of the Dirichlet series for a sequence (a_n) is positive, it is given by

$$\sigma_c = \limsup_{n\to\infty} \frac{\log |\sum_{k=1}^{n} a_n|}{\log n} \quad .$$

Combined with Theorem 4.2, this shows that the abscissa of convergence σ_c of the Dirichlet series for $\mathrm{sbs}(n)$ is $\sigma_c = \frac{2}{3}$.

The sequence $\mathrm{sbs}(n)$ does not exist alone, but is part of a family of sequences linked by a linear representation, like in the case of rational series [1]. A sequence (S_n) is called *k-rational* if and only if there exists k $d \times d$ square matrices A_i, $i = 0, \ldots, k-1$, a $1 \times d$ row matrix L and a $d \times 1$ column matrix C such that if we write n in base k as $n = n_\ell \cdots n_0$ with $n_\ell \neq 0$, the value of S_n is given by

$$S_n = L A_{n_\ell} \cdots A_{n_0} C \quad .$$

By convention, $S_0 = LC$. The family L, (A_i), C is called a *linear representation* of dimension d of the sequence.

Fact 4.4. Let T be a rational series over the alphabet $\{0, \ldots, k-1\}$. The sequence S obtained by putting side-by-side the values of T according to the length and rank in the lexicographic order of words is a k-rational sequence. Moreover, if L, $(A_i)_{i=0}^{k-1}$, C is a linear representation of T of dimension d, the following is a linear representation of S with dimension $2d+1$: $L' = \begin{pmatrix} 1 & L & 0 & \cdots & 0 \end{pmatrix}, C' = \begin{pmatrix} 0 & C & 0 & \cdots & 0 \end{pmatrix}^t$,

$$A'_0 = \begin{pmatrix} 1 & L & 0 \\ 0 & 0 & 0 \\ 0 & A_{k-1} & A_{k-2} \end{pmatrix} \quad , \quad A'_1 = \begin{pmatrix} 0 & 0 & L \\ 0 & A_0 & 0 \\ 0 & 0 & A_{k-1} \end{pmatrix} \quad \text{and}$$

$$A'_r = \begin{pmatrix} 0 & 0 & 0 \\ 0 & A_{r-1} & A_{r-2} \\ 0 & 0 & 0 \end{pmatrix}$$

for $1 < r < k$.

Proof. Let w be a word in base k and let n be the rank of w in the side-by-side ordering of the words. Then $n = \sum_{i=0}^{|w|-1} +(w)_k$, where $(w)_k = w_{|w|-1} k^{|w|-1} + \cdots + w_1 k + w_0$ is the integer represented by w in base k, and $S_n = T(w)$. The base k representation of n has the form $n = n_{|w|} k^{|w|} + \cdots + n_1 k + n_0$, where $n_{|w|} \in \{0, 1\}$. Define the carries c_i by $c_0 = 0$ and $c_{i+1} = 1$ if and only if $w_i + c_i + 1 \geq k$ (so we have $n_i = w_i + 1 + c_i \bmod k$). Denote $P_i = A_{w_{i-1}} \cdots A_{w_0} C$ and $P'_i = A'_{n_{i+1}} \cdots A'_{n_0} C'$. It is easy to see by induction that rows 2 to $d+1$ of P'_i contains P_i when $c_i = 0$ and all zeros otherwise, and that rows $d+2$ to $2d+1$ of P'_i contains P_i when $c_i = 1$ and all zeros otherwise. In the case $n_{|w|} = 1$, row 1 of $P'_{|w|+1}$ will contain $LP_{|w|}$ and $c_{|w|} = 1$. In the case $n_{|w|} = 0$, rows 2 to $d+1$ of $P'_{|w|}$ will contain $P_{|w|}$ and $c_{|w|} = 0$. It follows that $L' A'_{n_\ell} \cdots A'_{n_0} C' = S_n$. □

Applying Fact 4.4 to the linear representation of $\mathrm{adp}^{\oplus}(w)$ gives a linear representation L, $A_0, \ldots, A_7$, C of dimension 17 of the 8-rational sequence sbs(n). We consider the 17 rational sequences $u^1, u^2, \ldots, u^{17}$ that arise from the linear representation L, $A_0, \ldots, A_7$ and each vector from the canonical basis of $\mathbf{Q}^{17}$. Note that $u^2 =$ sbs. Let $u^i(s)$ be the Dirichlet series of u^i, and let $U(s)$ be the row vector of the series $u^i(s)$. Since the norm of the linear representation of sbs is 1, each sequence under consideration is bounded, and each $u^i(s)$ has an abscissa of convergence ≤ 1. Thus, $U(s)$ has abscissa of convergence $\sigma_c \leq 1$.

4.5.1 Meromorphicity

Our next goal is to show that the analytic function $U(s)$, defined in the half-plane $\sigma > \sigma_c$ admits an meromorphic extension. Let $U_n = (u_n^1, \ldots, u_n^{17})$. Then

$$\begin{aligned}
U(s) &= \sum_{n=1}^{\infty} \frac{U_n}{n^s} = \sum_{n=1}^{\infty} \frac{U_{8n}}{(8n)^s} + \sum_{r=1}^{7} \left(\frac{U_r}{r^s} + \sum_{n=1}^{\infty} \frac{U_{8n+r}}{(8n+r)^s} \right) \\
&= \sum_{n=1}^{\infty} \frac{U_n A_0}{(8n)^s} + \sum_{r=1}^{7} \left(\frac{U_r}{r^s} + \sum_{n=1}^{\infty} \frac{U_n A_r}{(8n+r)^s} \right) \\
&= \sum_{r=0}^{7} \sum_{n=1}^{\infty} \frac{U_n A_r}{(8n+r)^s} + \sum_{r=1}^{7} \frac{U_r}{r^s} \quad .
\end{aligned}$$

Denote $Q = \sum_{r=0}^{7} A_r$ and

$$\nabla U(s) = \sum_{r=1}^{7} \frac{U_r}{r^s} + \sum_{r=1}^{7} \sum_{n=1}^{\infty} U_n A_r \left(\frac{1}{(8n+r)^s} - \frac{1}{(8n)^s} \right) \quad . \tag{4.4}$$

Then we have $U(s)(\mathrm{I}_{17} - 8^{-s}Q) = \nabla U(s)$. Let Δ_h be the difference operator $\Delta_h u(n) = u(n+h) - u(n)$. Then we can write $\nabla U(s)$ as

$$\nabla U(s) = \sum_{r=1}^{7} \frac{U_r}{r^s} + \frac{1}{8^s} \sum_{r=1}^{7} \sum_{n=1}^{\infty} U_n A_r \Delta_{r/8} \frac{1}{n^s} \quad . \tag{4.5}$$

Since $\Delta_{r/8} \frac{1}{n^s} = \frac{1}{(n+r/8)^s} - \frac{1}{n^s} = -s \int_n^{n+r/8} \frac{du}{u^{s+1}}$,

$$\left| \Delta_{r/8} \frac{1}{n^s} \right| \leq |s| \int_n^{n+r/8} \frac{du}{u^{\sigma+1}} = \frac{|s|}{\sigma} \Delta_{r/8} \frac{1}{n^\sigma} \quad . \tag{4.6}$$

For fixed s with $\sigma > 1/3$, we have $\Delta_{r/8} \frac{1}{n^\sigma} \sim_{n \to \infty} \frac{\sigma}{n^{\sigma+1}}$. Since the components of U_n are bounded sequences, it follows that

$$\frac{1}{8^s} U_n A_r \Delta_{r/8} \frac{1}{n^s} \underset{n \to \infty}{=} O\left(\frac{1}{n^{\sigma+1}} \right) \quad .$$

We conclude that the series (4.4) converges absolutely in the half-plane $\sigma > 0$. Thus, the function $\nabla U(s)$ is analytic in this half-plane.

Table 4.2: The Jordan form $J = P^{-1}QP$ of Q.

$$J = \begin{pmatrix}
4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/4 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/4 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/4 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{pmatrix} .$$

4.5.2 Poles

The poles of $U(s)$ come from the term $I_{17} - 8^{-s}Q$. The eigenvalues of Q are 4, 2, 1, 1/4 and 0, with multiplicities 1, 3, 6, 3 and 4, respectively. Let $J = P^{-1}QP$ be the Jordan form of Q, where J is the quasi-diagonal matrix given in Table 4.2. We make a change of coordinates to get a new sequence V_n with $U_n = V_n P^{-1}$ and $U(s) = V(s)P^{-1}$. More precisely, if $L, C, A_0, \ldots, A_7$ is the linear representation of U_n, we get the linear representation $L' = LP$, $C' = P^{-1}C$, $A_0' = P^{-1}A_0P$, ..., $A_7' = P^{-1}A_7P$. Applying this change of coordinates to (4.4) gives $\nabla V(s) = V(s)(I_{17} - 8^{-s}J)$, where

$$\nabla V(s) = \sum_{r=1}^{7} \frac{V_r}{r^s} + \sum_{r=1}^{7} \sum_{n=1}^{\infty} V_n A_r' \left(\frac{1}{(8n+r)^s} - \frac{1}{(8n)^s} \right) .$$

The function $\nabla V(s)$ is analytic for $\sigma > 0$.

The equation $\nabla V(s) = V(s)(I_{17} - 8^{-s}J)$ gives a system of equations of the form $\nabla v^j(s) = v^j(s)(1 - J_{jj}8^{-s})$, $j = 1, \ldots, 17$. From these equations we see, with $\chi = 2\pi i / \ln 8$, that

- The function $v^1(s)$ is meromorphic in the half plane $\sigma > 0$, with possible poles as $2/3 + k\chi$, $k \in \mathbf{Z}$.
- The functions $v^2(s), \ldots, v^4(s)$ are meromorphic in the half plane $\sigma > 0$, with possible poles as $1/3 + k\chi$, $k \in \mathbf{Z}$.
- The functions $v^5, \ldots, v^{17}(s)$ are analytic in the half plane $\sigma > 0$.

Recall that the Dirichlet series $u^2(s)$ of $\mathrm{sbs}(n)$ has abscissa of convergence 2/3, and $U(s) = V(s)P^{-1}$. Hence, $u^2(s)$ extends to a meromorphic function in $\sigma > 0$. Since $\mathrm{sbs}(n)$ is nonnegative, 2/3 is a singularity of $u^2(s)$.

If $2/3$ would not be a pole of $v^1(s)$, the argument above would show that $u^2(s)$ is analytic in $\sigma > 1/3$—a contradiction. Thus, $2/3$ indeed is a pole for $v^1(s)$. For the other Dirichlet series $u^j(s)$, we do not know exactly their abscissa of convergence, but since the Dirichlet series have nonnegative coefficients, and the rightmost possible singularity are at $2/3$, we know that all the Dirichlet series have abscissa of convergence no greater that $2/3$.

4.5.3 Order of Growth

Recall that the order of growth $\mu_g(\sigma)$ [49, 48] of a function $g(s)$ along the line $\sigma = c$ is

$$\mu_g(\sigma) = \inf\left\{\lambda \mid g(\sigma + it) \underset{|t|\to\infty}{=} O(|t|^\lambda)\right\} \ .$$

Since the Dirichlet series defining $u^j(s)$ have nonnegative coefficients and abscissae of convergence $\leq 2/3$, their order of growth is $\mu_{u^j}(s) = 0$ for $\sigma > 2/3$.

From (4.5) and (4.6) on page 44, we get the inequality $|\nabla U(s)| \leq A + B\frac{|s|}{\sigma}\frac{\zeta(\sigma+1)}{8^\sigma}$ for some constants A and B, where $\zeta(s)$ is the Riemann zeta function. It follows that $\mu_{\nabla u^j}(\sigma) \leq 1$ for $0 < \sigma < 1$. Since the functions $s \mapsto 1 - J_{jj}8^{-s}$ are periodic with respect to t, this is valid also for the functions u^j. According to Lindelöf's theorem [23, Theorem 14], $\mu(\sigma)$ is a convex function, and thus

$$\mu_{u^j}(\sigma) \leq 1 - \sigma \quad \text{for } 0 < \sigma < 1 \quad \text{and} \quad \mu_{u^j}(\sigma) = 0 \quad \text{for } \sigma > 1 \ .$$

Since the functions $v^j(s)$ are linear combinations of the $u^j(s)$, the same result holds for all the $v^j(s)$.

4.6 FOURIER SERIES

4.6.1 Mellin-Perron Formula

We now apply the following Mellin-Perron formula [48].

Fact 4.5 (Mellin-Perron formula). Let $f(s) = \sum_{k=1}^{\infty} f_k k^{-s}$ be the Dirichlet series of the sequence (f_k). Let the line $\sigma = c > 0$ lie inside the half-plane of absolute convergence of $f(s)$. Then for $m > 0$,

$$\frac{1}{m!}\sum_{1\leq k<\nu} f_k\left(1 - \frac{k}{\nu}\right)^m = \frac{1}{2\pi i}\int_{c-i\infty}^{c+i\infty} f(s)\nu^s \frac{ds}{s(s+1)\cdots(s+m)} \ . \quad (4.7)$$

Note that when $m = 1$, the left hand side of (4.7) can be written as

$$\sum_{1\leq k<\nu} f_k\left(1 - \frac{k}{\nu}\right) = \frac{1}{\nu}\sum_{1\leq k<\nu}\sum_{1\leq \ell\leq k} f_\ell = \frac{1}{\nu}\sum_{1\leq k\leq\nu}\sum_{1\leq \ell<k} f_\ell \ .$$

We apply the formula with $m = 1$ to the row vector $V(s)$, and push the line of integration to the left, taking the residues of the function into account.

Let $0 < \epsilon < 1/3$ and $\chi = 2\pi i/\ln 8$. Let (ϵ) denote the line of integration $\sigma = \epsilon \pm i\infty$. For $v^1(s)$, we get

$$\frac{1}{\nu}\sum_{1\le n<\nu}\sum_{k=1}^{n-1} v_k^1 = \sum_{k\in\mathbf{Z}} \operatorname*{Res}_{s=\frac{2}{3}+k\chi} \frac{\nabla v^1(s)\nu^s}{(1-4\cdot 8^{-s})s(s+1)}\frac{1}{2\pi i} \\ + \int_{(\epsilon)} \frac{\nabla v^1(s)\nu^s}{1-4\cdot 8^{-s}}\frac{ds}{s(s+1)} \quad .$$

For $j = 2, 3, 4$, we get

$$\frac{1}{\nu}\sum_{1\le n<\nu}\sum_{k=1}^{n-1} v_k^j = \sum_{k\in\mathbf{Z}} \operatorname*{Res}_{s=\frac{1}{3}+k\chi} \frac{\nabla v^j(s)\nu^s}{(1-2\cdot 8^{-s})s(s+1)} \\ + \frac{1}{2\pi i}\int_{(\epsilon)} \frac{\nabla v^j(s)\nu^s}{1-2\cdot 8^{-s}}\frac{ds}{s(s+1)} \quad .$$

For $j = 5, \ldots, 17$, we get

$$\frac{1}{\nu}\sum_{1\le n<\nu}\sum_{k=1}^{n-1} v_k^j = \frac{1}{2\pi i}\int_{(\epsilon)} \frac{\nabla v^j(s)\nu^s}{1-J_{jj}\cdot 8^{-s}}\frac{ds}{s(s+1)} \quad .$$

All the integrals above can be bounded as

$$\nu^\epsilon \left| \frac{1}{2\pi i}\int_{(\epsilon)} \frac{\nabla v^j(s)\nu^{it}}{1-J_{jj}\cdot 8^{-s}}\frac{ds}{s(s+1)} \right| \underset{\nu\to\infty}{=} O(\nu^\epsilon) \quad .$$

By computing the residues, we obtain

$$\frac{1}{\nu}\sum_{1\le n<\nu}\sum_{k=1}^{n-1} v_k^1 \underset{\nu\to\infty}{=} \frac{\nu^{2/3}}{\ln 8}\sum_{k\in\mathbf{Z}} \frac{\nabla v^1(2/3+k\chi)}{(2/3+k\chi)(5/3+k\chi)}\exp(2\pi i k\log_8\nu) \\ + O(\nu^\epsilon) \quad .$$

For $j = 2, 3, 4$, we obtain

$$\frac{1}{\nu}\sum_{1\le n<\nu}\sum_{k=1}^{n-1} v_k^j \underset{\nu\to\infty}{=} \frac{\nu^{1/3}}{\ln 8}\sum_{k\in\mathbf{Z}} \frac{\nabla v^j(1/3+k\chi)}{(1/3+k\chi)(4/3+k\chi)}\exp(2\pi i k\log_8\nu) \\ + O(\nu^\epsilon) \quad .$$

Finally, for $j = 5, \ldots, 17$, we have

$$\frac{1}{\nu}\sum_{1\le n<\nu}\sum_{k=1}^{n-1} v_k^j \underset{\nu\to\infty}{=} O(\nu^\epsilon) \quad .$$

Since $\nabla v^1(2/3+it) =_{|t|\to\infty} O(|t|^{1/3})$ and $\nabla v^j(1/3+it) =_{|t|\to\infty} O(|t|^{2/3})$, the series above converge absolutely. It follows that the trigonometric series define 1-periodic continuous functions. Since the sequence $\mathrm{sbs}(n)$ is a linear combination of the sequences $v^1, \ldots, v^{17}$, we obtain the following result.

Theorem 4.3. For all $0 < \epsilon < 1/3$,

$$\sum_{1\le n<\nu}\sum_{k=1}^{n-1} \mathrm{sbs}(k) \underset{\nu\to\infty}{=} \nu^{5/3}H_{5/3}(\log_8\nu) + \nu^{4/3}H_{4/3}(\log_8\nu) + O(\nu^{1+\epsilon}) \quad ,$$

where $H_{5/3}$ and $H_{4/3}$ are 1-periodic continuous functions.

4.6.2 From Double to Simple Sums

By Theorem 4.2 on page 42,

$$\sum_{1\le n<\nu} \operatorname{sbs}(n) \underset{\nu\to\infty}{=} \nu^{2/3} G_{2/3}(\log_8 \nu) + o(v^{2/3}) \ ,$$

where $G_{2/3}$ is a 1-periodic continuous function. We will use the following pseudo-Tauberian result to derive a Fourier series expansion for $G_{2/3}$.

Fact 4.6 ([21, Proposition 6.4]). Let f be a 1-periodic continuous function and let τ be a complex number with positive real part. Then there exists a 1-periodic continuously differentiable function g such that

$$\frac{1}{\nu^{\tau+1}} \sum_{1\le n<\nu} n^{\tau} f(\log_8 n) = g(\log_8 \nu) + o(1) \ .$$

Moreover, the function $g(u)$, which depends on $f(t)$ and τ, satisfies

$$\int_0^1 g(u)\ du = \frac{1}{\tau+1}\int_0^1 f(u)\ du$$

and

$$g\left(f(t)e^{-2\pi i t}, \tau + \frac{2\pi i}{\ln 8}\ ;\ u\right) = g(f(t),\tau\ ;\ u)e^{-2\pi i u} \ .$$

Fact 4.6 (with $\tau = 2/3$) implies that there exists a 1-periodic and continously differentiable function $G_{5/3}$ such that

$$\sum_{1\le n<\nu}\sum_{k=1}^{n-1} \operatorname{sbs}(k) \underset{\nu\to\infty}{=} \nu^{5/3} G_{5/3}(\log_8 \nu) + o(\nu^{5/3}) \ .$$

The uniqueness of asymptotic expansion with variable coefficients [10, Chapter V] shows that $G_{5/3} = H_{5/3}$.

Let $c_k(F)$ denote the Fourier coefficients of the periodic function F. By Fact 4.6, we get with $\chi = 2\pi i/\ln 8$

$$\begin{aligned} c_k(H_{5/3}) &= \int_0^1 H_{5/3}(u)e^{-2\pi i k u}du = \int_0^1 g(G_{2/3}(t)e^{-2\pi i k t}, 2/3 + \chi k)(u)\ du \\ &= \frac{1}{2/3+\chi k + 1}\int_0^1 G_{2/3}(t)e^{-2\pi i k t}dt = \frac{1}{5/3+\chi k} c_k(G_{2/3}) \ . \end{aligned}$$

This shows that the Fourier coefficients of $G_{2/3}$ are given by

$$c_k(G_{2/3}) = \left(\frac{5}{3} + k\chi\right) c_k(H_{5/3}) = \frac{1}{\ln 8}\frac{\nabla v^1(2/3+k\chi)}{2/3+k\chi} \ .$$

Theorem 4.4. The summation function of the first order for sbs satisfies

$$\sum_{1\le n<\nu} \operatorname{sbs}(n) \underset{\nu\to\infty}{=} \nu^{2/3} G_{2/3}(\log_8 \nu) + o(\nu^{2/3}) \ ,$$

where $G_{2/3}$ is a 1-periodic continuous function. The function $G_{2/3}$ has the Fourier series

$$G_{2/3}(\xi) = \frac{1}{\ln 8} \sum_{k \in \mathbf{Z}} \frac{\nabla v^1(2/3 + k\chi)}{2/3 + k\chi} e^{2\pi i k \xi} \quad ,$$

where $\chi = 2\pi i / \ln 8$. The series is not absolutely convergent, but it is a Fourier series of a continuous function. According to Fejér's theorem, the series thus converges uniformly towards the function in the sense of Cesàro.

The Fourier coefficients can be numerically computed. In particular, the mean is approximately $c_0(G_{2/3}) \approx 1.131362078$.

5 CONCLUSIONS

In this thesis, we have brought together results on the differential and linear properties of addition modulo 2^n from several sources into one coherent framework. This framework based on rational series gives straightforward and intuitive derivations of complete characterisations of the differential and linear properties of addition modulo 2^n. Within the framework, we can also conveniently study the differential properties of bitwise exclusive-or when differences are expressed using addition modulo 2^n, although a complete characterisation in this case seems difficult. As we have illustrated, the approach can be generalised to more complex functions built from addition.

We would like to point out three natural lines of further research: the classification of the differential and linear properties of other (more complex) functions, cryptanalytic applications to existing ciphers and the design of ciphers resistant against differential and linear cryptanalysis based on our results.

A crucial property of addition modulo 2^n and the other functions considered in this thesis is that the ith output only depends on the inputs in position lower than i. In particular, the functions under consideration are of the form $F\colon A^n \to B^n$, where F can be expressed using two functions $f\colon S \times A \to B$ and $g\colon S \times A \to S$ such that $F(x_{n-1}, \dots, x_0)_i = f(s_i, x_i)$, where $s_{i+1} = g(s_i, x_i)$. These are exactly the functions computed by Mealy machines [46]. For addition, s_i is simply the ith carry bit. One would expect that the differential and linear properties of these types of functions could be studied using rational series. A complete characterisation of the differential and linear properties of other arithmetic operations such as multiplication modulo 2^n would also be highly desirable, but this probably requires different methods.

We have completely left open the application of our results to existing ciphers. Given the generation algorithms, it would be natural to attempt to find the best differentials and linear approximations of existing ciphers, thus either obtaining improved attacks or proofs that the ciphers are resistant against basic differential and linear cryptanalysis. Due to the abundance of differentials or approximations of addition with nontrivial probability or correlation, this will probably require the development of improved search algorithms for finding optimal trails, or for upper bounding the differential probability or correlation.

Finally, we propose the challenge to design a simple and efficient cipher that uses only addition modulo 2^n and $\mathbf{F}_2$-affine functions, and that is provably resistance against basic differential and linear cryptanalysis. If the octal word x corresponds to a differential or linear approximation of addition modulo 2^n, $\mathrm{xdp}^+(x)$ is upper bounded by 2^{-k}, where $k = |\{i < n-1 \mid x_i \neq 0, 7\}|$ (see Theorem 2.2 on page 14), whereas $|\mathrm{lca}(x)|$ is upper bounded by $2^{-\ell}$, where $\ell = |\{i \mid x_i \neq 0, 7\}|$ (see the automaton (3.1) on page 23). That is, both the differential probability and correlation drops exponentially with the number of bit positions where the input differences or input selection vectors differ. The affine parts of the cipher could thus be designed to maximise the sum of the number of bit positions where the input differences or selection vectors differ for the addition operations in consequent rounds.

ACKNOWLEDGEMENTS

This report is a reprint of my Master's thesis. Apart from some typographical changes and fixed typos no changes have been made in this reprint. The results in Chapter 4 are joint work with Philippe Dumas (Algorithms Project, INRIA, France) and Helger Lipmaa (Laboratory for Theoretical Computer Science, Helsinki University of Technology). I would like to thank Kaisa Nyberg (Nokia Research Center, Finland) for giving a formal statement of the thesis as well as valuable comments on the presentation. This work has been done at the Laboratory for Theoretical Computer Science, Helsinki University of Technology, and has been supported by the Finnish Defence Forces Research Institute of Technology.

BIBLIOGRAPHY

[1] Jean-Paul Allouche and Jeffrey Shallit. The ring of k-regular sequences. *Theoretical Computer Science*, 98(2):163–197, 1992. [43]

[2] Kazumaro Aoki, Kunio Kobayashi, and Shiho Moriai. Best differential characteristic search for FEAL. In *Fast Software Encryption 1997*, volume 1267 of *LNCS*, pages 41–53. Springer-Verlag, 1997. [4]

[3] Tom Apostol. *Modular Functions and Dirichlet Series in Number Theory*. Springer-Verlag, second edition, 1990. [42]

[4] Frederik Armknecht and Matthias Krause. Algebraic attacks on combiners with memory. In *Advances in Cryptology—Crypto 2003*, volume 2729 of *LNCS*, pages 162–175. Springer-Verlag, 2003. [2]

[5] Thomas Berson. Differential cryptanalysis mod 2^{32} with applications to MD5. In *Advances in Cryptology—Eurocrypt 1992*, volume 658 of *LNCS*, pages 71–80. Springer-Verlag, 1992. [30]

[6] Jean Berstel and Christophe Reutenauer. *Rational Series and Their Languages*. EATCS Monographs on Theoretical Computer Science. Springer-Verlag, 1988. [12, 21, 31]

[7] Eli Biham, Alex Biryukov, and Adi Shamir. Cryptanalysis of Skipjack reduced to 31 rounds using impossible differentials. In *Advances in Cryptology—Eurocrypt 1999*, volume 1592 of *LNCS*, pages 12–23. Springer-Verlag, 1999. [3]

[8] Eli Biham and Adi Shamir. Differential cryptanalysis of DES-like cryptosystems. *Journal of Cryptology*, 4(1):3–72, 1991. [1]

[9] Eli Biham and Adi Shamir. *Differential Cryptanalysis of the Data Encryption Standard*. Springer-Verlag, 1993. [1, 2, 3]

[10] Nicolas Bourbaki. *Éléments de mathématique Fonctions d'une variable réelle Théorie élémentaire*. Hermann, 1976. [48]

[11] Pierre Brémaud. *Markov Chains: Gibbs Fields, Monte Carlo Simulation, and Queues*. Springer-Verlag, 1999. [19, 23, 32]

[12] Lennart Brynielsson. Unbalance of bits from addition with carry. Personal communication, March 2003. [23]

[13] Florent Chabaud and Serge Vaudenay. Links between differential and linear cryptanalysis. In *Advances in Cryptology—Eurocrypt 1994*, volume 950 of *LNCS*, pages 356–365. Springer-Verlag, 1995. [6]

[14] Nicolas Courtois. Fast algebraic attacks on stream ciphers with linear feedback. In *Advances in Cryptology—Crypto 2003*, volume 2729 of *LNCS*, pages 176–194. Springer-Verlag, 2003. [2]

[15] Nicolas Courtois and Willi Meier. Algebraic attacks on stream ciphers with linear feedback. In *Advances in Cryptology—Eurocrypt 2003*, volume 2656 of *LNCS*, pages 345–359. Springer-Verlag, 2003. [2]

[16] Nicolas Courtois and Josef Pieprzyk. Cryptanalysis of block ciphers with overdefined systems of equations. In *Advances in Cryptology—Asiacrypt 2002*, volume 2501 of *LNCS*, pages 267–287. Springer-Verlag, 2002. [2]

[17] Joan Daemen. *Cipher and Hash Function Design: Methods Based on Linear and Differential Cryptanalysis.* PhD thesis, Katholieke Universiteit Leuven, March 1995. [1, 7]

[18] Joan Daemen and Vincent Rijmen. *The Design of Rijndael.* Springer-Verlag, 2002. [2]

[19] Patrik Ekdahl and Thomas Johansson. A new version of the stream cipher SNOW. In *Selected Areas in Cryptography 2002*, volume 2595 of *LNCS*, pages 47–61. Springer-Verlag, 2003. [1]

[20] Niels Ferguson, Doug Whiting, Bruce Schneier, John Kelsey, Stefan Lucks, and Tadayoshi Kohno. Helix: fast encryption and authentication in a single cryptographic primitive. In *Fast Software Encryption 2003*, volume 2887 of *LNCS*, pages 330–346. Springer-Verlag, 2003. [1]

[21] Philippe Flajolet, Peter Grabner, Peter Kirschenhofer, Helmut Prodinger, and Robert Tichy. Mellin transforms and asymptotics: Digital sums. *Theoretical Computer Science*, 123(2):291–314, 1994. [48]

[22] Philippe Flajolet and Robert Sedgewick. Analytic combinatorics, 2002. Book in preparation. Individual chapters are available from `http://algo.inria.fr/flajolet/Publications/books.html`. [15, 25]

[23] G. H. Hardy and M. Riesz. *The general theory of Dirichlet's series.* Stechert-Hafner, Inc., New York, 1964. Cambridge Tracts in Mathematics and Mathematical Physics, No. 18. [46]

[24] Juraj Hromkovič. *Algorithmics for Hard Problems.* Springer-Verlag, second edition, 2003. [4]

[25] Pascal Junod. On the optimality of linear, differential, and sequential distinguishers. In *Advances in Cryptology—Eurocrypt 2003*, volume 2656 of *LNCS*, pages 17–32. Springer-Verlag, 2003. [3, 6]

[26] Liam Keliher, Henk Meijer, and Stafford Travares. New method for upper bounding the maximum average linear hull probability for SPNs. In *Advances in Cryptology—Eurocrypt 2001*, volume 2045 of *LNCS*, pages 420–436. Springer-Verlag, 2001. [1]

[27] Lars Knudsen. Truncated and higher order differentials. In *Fast Software Encryption 1994*, volume 1008 of *LNCS*, pages 196–210. Springer-Verlag, 1995. [3]

[28] Xuejia Lai. Higer order derivatives and differential cryptanalysis. In *Symposium on Communication, Coding and Cryptography*, pages 227–233. Kluwer Academic Publishers, 1994. [3]

[29] Helger Lipmaa. On differential properties of Pseudo-Hadamard transform and related mappings. In *Progress in Cryptology—Indocrypt 2002*, volume 2551 of *LNCS*, pages 48–61. Springer-Verlag, 2002. [1, 17, 19]

[30] Helger Lipmaa and Shiho Moriai. Efficient algorithms for computing differential properties of addition. In *Fast Software Encryption 2001*, volume 2355 of *LNCS*, pages 336–350. Springer-Verlag, 2002. [1, 10, 12, 13, 16, 17, 31, 35, 36]

[31] Eugene Lukacs. *Characteristic Functions*. Griffin, second edition, 1970. [40]

[32] James Massey. SAFER K-64: A byte-oriented block-ciphering algorithm. In *Fast Software Encryption 1993*, volume 809 of *LNCS*, pages 1–17. Springer-Verlag, 1994. [1, 18, 29]

[33] James Massey, Gurgen Khachatrian, and Melsik Kuregian. Nomination of safer++ as candidate algorithm for the New European Schemes for Signatures, Integrity, and Encryption (NESSIE), 2000. `http://www.cryptonessie.org`. [1, 18, 29]

[34] Mitsuru Matsui. Linear cryptanalysis method for DES cipher. In *Advances in Cryptology—Eurocrypt 1993*, volume 765 of *LNCS*, pages 386–397. Springer-Verlag, 1993. [1, 5, 6]

[35] Mitsuru Matsui. The first experimental cryptanalysis of the Data Encryption Standard. In *Advances in Cryptology—Crypto 1994*, volume 839 of *LNCS*, pages 1–11. Springer-Verlag, 1994. [1]

[36] Mitsuru Matsui. On correlation between the order of S-boxes and the strength of DES. In *Advances in Cryptology—Eurocrypt 1994*, volume 950 of *LNCS*, pages 366–375. Springer-Verlag, 1995. [4]

[37] Mitsuru Matsui. New structure of block ciphers with provable security against differential and linear cryptanalysis. In *Fast Software Encryption 1996*, volume 1039 of *LNCS*, pages 205–218. Springer-Verlag, 1996. [1]

[38] Hiroshi Miyano. Addend dependency of differential/linear probability of addition. *IEICE Transactions*, E81-A(1):106–109, 1998. [20]

[39] Kaisa Nyberg. Linear approximation of block ciphers. In *Advances in Cryptology—Eurocrypt 1994*, volume 950 of *LNCS*, pages 439–444. Springer-Verlag, 1995. [1]

[40] Kaisa Nyberg and Lars Knudsen. Provable security against a differential attack. *Journal of Cryptology*, 8(1):27–37, 1995. [1]

[41] National Institute of Standards and Technology (NIST). Secure hash standard. *Federal Information Processing Standards Publication (FIPS PUB)*, 180-2, 2002. [1]

[42] Sangwoo Park, Soo Hak Sung amd Sangjin Lee, and Jongin Lim. Improving the upper bound on the maximum differential and the maximum linear hull probability for SPN structures and the AES. In *Fast Software Encryption 2003*, volume 2887 of *LNCS*, pages 247–260. Springer-Verlag, 2003. [1]

[43] William Press, Brian Flannery, Saul Teukolsky, and William Vetterling. *Numerical Recipes in Fortran, C and C++*. Cambridge University Press, second edition, 1993. [9]

[44] Ron Rivest, Matt Robshaw, Ray Sidney, and Yiqun Lisa Yin. The RC6 block cipher. In *First AES Candidate Conference*, 1998. See also `http://www.rsasecurity.com/rsalabs/rc6/`. [1]

[45] Phillip Rogaway and Don Coppersmith. A software-optimized encryption algorithm. *Journal of Cryptology*, 11(4):273–287, 1998. [1]

[46] Arto Salomaa. *Computation and Automata*, volume 25 of *Encyclopedia of Mathematics and Its Applications*. Cambridge University Press, 1985. [50]

[47] Bruce Schneier, John Kelsey, Doug Whiting, David Wagner, Chris Hall, and Niels Ferguson. Twofish: a 128-bit block cipher. In *First AES Candidate Conference*, 1998. See also `http://www.counterpane.com/twofish.html`. [1, 18, 29]

[48] Gérald Tenenbaum. *Introduction to Analytic and Probabilistic Number Theory*, volume 46 of *Cambridge Studies in Advanced Mathematics*. Cambridge University Press, 1995. [46]

[49] E. C. Titchmarsh. *The Theory of Functions*. Oxford University Press, second edition, 1932. [46]

[50] Serge Vaudenay. Provable security for block ciphers by decorrelation. In *Symposium on Theoretical Aspects of Computer Science 1998*, volume 1373 of *LNCS*, pages 249–275. Springer-Verlag, 1998. [1]

[51] David Wagner. The boomerang attack. In *Fast Software Encryption 1999*, volume 1636 of *LNCS*, pages 156–170. Springer-Verlag, 1999. [3]

[52] Johan Wallén. Linear approximations of addition modulo 2^n. In *Fast Software Encryption 2003*, volume 2887 of *LNCS*, pages 261–273. Springer-Verlag, 2003. [1, 11, 20]

INDEX

HELSINKI UNIVERSITY OF TECHNOLOGY LABORATORY FOR THEORETICAL COMPUTER SCIENCE RESEARCH REPORTS

HUT-TCS-A71 Keijo Heljanko
Combining Symbolic and Partial Order Methods for Model Checking 1-Safe Petri Nets.
February 2002.

HUT-TCS-A72 Tommi Junttila
Symmetry Reduction Algorithms for Data Symmetries. May 2002.

HUT-TCS-A73 Toni Jussila
Bounded Model Checking for Verifying Concurrent Programs. August 2002.

HUT-TCS-A74 Sam Sandqvist
Aspects of Modelling and Simulation of Genetic Algorithms: A Formal Approach.
September 2002.

HUT-TCS-A75 Tommi Junttila
New Canonical Representative Marking Algorithms for Place/Transition-Nets. October 2002.

HUT-TCS-A76 Timo Latvala
On Model Checking Safety Properties. December 2002.

HUT-TCS-A77 Satu Virtanen
Properties of Nonuniform Random Graph Models. May 2003.

HUT-TCS-A78 Petteri Kaski
A Census of Steiner Triple Systems and Some Related Combinatorial Objects. June 2003.

HUT-TCS-A79 Heikki Tauriainen
Nested Emptiness Search for Generalized Büchi Automata. July 2003.

HUT-TCS-A80 Tommi Junttila
On the Symmetry Reduction Method for Petri Nets and Similar Formalisms.
September 2003.

HUT-TCS-A81 Marko Mäkelä
Efficient Computer-Aided Verification of Parallel and Distributed Software Systems.
November 2003.

HUT-TCS-A82 Tomi Janhunen
Translatability and Intranslatability Results for Certain Classes of Logic Programs.
November 2003.

HUT-TCS-A83 Heikki Tauriainen
On Translating Linear Temporal Logic into Alternating and Nondeterministic Automata.
December 2003.

HUT-TCS-A84 Johan Wallén
On the Differential and Linear Properties of Addition. December 2003.

ISBN 951-22-6880-9
ISSN 1457-7615