Bachelor's Thesis Computing Science

Trail Clipping, Clustering, and Spurious Trails in Farfalle

Jabril Abukar
s1123695

August 18, 2026

First supervisor/assessor:
prof. dr. J.J.C. Daemen (Joan)

Second assessor:
dr. S. Mella (Silvia)

Radboud University

Abstract

In this thesis, we investigate the extent to which the empirical collision probability of the Farfalle construction differs from the weight-based approximation predicted by the dominant trail, and how trail clipping, trail clustering, and spurious trails account for this divergence in the Subterranean 3.0-P round transformation. Using an empirical analysis of all differential trails in a 24-bit toy model, MiniSubterranean-24, we evaluate these three phenomena and track how collision probabilities evolve across one to five transformation rounds. In MiniSubterranean-24[1] and MiniSubterranean-24[2], empirical measurements match the dominant trail approximation closely, showing no trail clipping, no clustering, and a small sign of spurious trails. Starting from MiniSubterranean-24[3], the dominant trail model breaks down across all three metrics: trail weights exceed the 24-bit state space, causing trail clipping. Non-linear branching causes trail clustering where observed differential hits significantly exceed the single dominant trail approximation, causing millions of secondary trails to emerge, driving a sharp increase in the spurious ratio. In MiniSubterranean-24[4] and MiniSubterranean-24[5], this divergence reaches its extreme as the empirical collision probability flattens out at $2^{-22.19}$, while the dominant trail prefix approximation continues to drop exponentially toward zero, causing the spurious ratio to explode. These results demonstrate that relying solely on dominant trail approximations significantly overestimates Farfalle collision security by underestimating the collective contribution of weaker secondary trails. For the tested MiniSubterranean-24 inputs, the collision probability reaches an apparent plateau by four rounds, with no further reduction observed in the fifth round

Contents

Chapter 1

Introduction

1.1 Background and Motivation

In this thesis, we explore the Subterranean 3.0-P transformation, a lightweight cryptographic primitive designed for resource-constrained environments. Subterranean 3.0-P can achieve powerful cryptographic capabilities when combined with the Farfalle construction (2.2). The Farfalle construction uses this transformation to construct a deck (Double-Extendable Cryptographic Keyed) function. A deck function is highly versatile, as it accepts arbitrary-length input strings and produces arbitrary-length output strings.

The deck function becomes a secure deck function that is computationally indistinguishable to a random oracle if it is used with a secret key. A random oracle can be seen as a black box, given a specific string of input, it will produce a random sequence of bits. The oracle is deterministic, providing the same input will always produce the same sequence. Therefore, to an adversary who does not know the secret key, the oracle's output is unpredictable. Because of these properties, this construction can be flexibly deployed as a stream cipher, a Message Authentication Code (MAC), or applied to various other symmetric encryption tasks.

1.2 The Problem Statement

The security of a deck function built with the Farfalle construction cannot be formally proven. However, it can be evaluated by analyzing its resistance against known cryptographic attacks. One of the most effective attacks against Farfalle is the generation collision. A collision occurs when a pair of distinct inputs generates the same output. The most efficient method to achieve this is by processing two short messages that share a chosen input difference.

The probability of such a collision occurring (taken over all keys) is defined by an expression that depends on the differential properties of the transformation (2.2.1). However, evaluating this expression can be computationally infeasible for large state transformations. Therefore, in practice, the collision probability is often estimated using a simplified approximation that is much easier to compute.

The Subterranean 3.0-P transformation operates on a 384-bit state, arranged in a two-dimensional array of 3 rows and 128 columns, resulting in a state space of 2^{384} possible values. Because of

this vast state space, evaluating all 2^{384} input pairs for a given input difference using exhaustive search is computationally infeasible.

For this reason, cryptographers rely on Differential Cryptanalysis (4.1) to estimate the resistance against collision attacks. This field studies how an input difference propagates through a cryptographic primitive that results in an output difference. Specifically, analysts evaluate differential trails, which are the sequences of intermediate differences that form the trail through the successive rounds of the Subterranean 3.0-P transformation.

However, approximate differential analysis often assumes that the non-linear components of the transformation, specifically the active S-boxes (4.1.3), operate entirely independently in each round. When the internal geometry of the transformation causes this generally made assumption to fail, it results in trail clipping (4.2) where the actual probability of a trail diverges from its expected probability.

Furthermore, the approximation often assumes the existence of a single dominant trail and fails to account for trail clustering (4.3), in which multiple distinct trails converge on the same output difference. When estimating the collision resistance of the construction, analysts typically rely on a single dominant trail prefix. However, this approach completely ignores thousands or millions of secondary trails that, while individually much weaker, collectively dominate the actual collision resistance of the transformation, a phenomenon known as spurious trails (4.4.2).

The problem addressed in this thesis is the divergence between these two calculations. By making use of a scaled-down toy model, we evaluate both the collision probability and the theoretical approximation. This allows us to investigate and describe the specific phenomena that cause the approximation to deviate from the observed collision probability.

1.3 Methodology: The 24-bit Toy Model

In this thesis, we perform empirical research on a scaled-down toy version of the Subterranean 3.0-P transformation using an exhaustive approach. The toy version, MiniSubterranean-24, shares the same structural proportions as the original transformation. It maintains the architecture of 3 rows, but is scaled down to a total state size of 24 bits (3 rows of 8 columns).

Difference propagation is the process of tracking how an initial difference between a pair of inputs evolves as it passes through the linear and non-linear layers of a transformation. The vast state space of the original 384-bit Subterranean 3.0-P prevents us from evaluating all 2^{384} input pairs for a given input difference. However, because MiniSubterranean-24 has a state space of 2^{24}, we can iterate through all possible input pairs for a given input difference to evaluate this difference propagation. We do this by making use of an exhaustive search on the smaller toy model, where we can compute the collision probability, the probability that an input difference injected into two independent blocks results in identical output differences that cancel each other out in the accumulator of the Farfalle construction (5.8).

1.4 Research Questions

We aim to answer the following research question:

To what extent does the collision probability of the Farfalle construction differ from

the weight-based approximation predicted by the dominant trail prefix, and how do spurious trails account for this difference in the MiniSubterranean-24 transformation?

To answer this main question, we investigate the following sub-questions:

1. How can we evaluate the Differential Probability (DP) distributions for the MiniSubterranean-24 transformation through exhaustive evaluation?
2. To what extent do specific cryptographic phenomena, namely trail clipping, trail clustering, and spurious trails, contribute to the divergence between the weight-based approximation and empirical collision probabilities?
3. How does increasing the number of transformation rounds (from one to five) affect the progression of the clipping ratio, cluster ratio, and spurious trail ratio, and what does this reveal about the minimum rounds required for optimal Farfalle collision resistance?

Chapter 2

Preliminaries

2.1 Notations, Symbols and Definitions

We specify the mathematical notation used to express strings, operations, and blocks to maintain clarity throughout this thesis.

Strings and Bitwise Operations
We operate primarily on strings of bits. We denote arbitrary-length bit strings using uppercase italic letters (e.g., X, Y, Z) and their lengths as $|X|$. To address an individual bit at index j within a string X, we use the notation $X[j]$. Constant strings of identical bits are denoted with a superscript indicating their length, for example, 0^n represents a string of n zero bits where $n \in \mathbb{N}$.

When combining strings, we use the following standard operations:

- **Concatenation:**
  The operation $Z = X \parallel Y$ denotes the concatenation of string X and string Y, giving a string of length $|Z| = |X| + |Y|$. An individual bit at index j (where $0 \leq j < |X| + |Y|$) of the concatenated string is defined as:

$$(X \parallel Y)[j] = \begin{cases} X[j] & \text{if } 0 \leq j < |X|, \\ Y[j - |X|] & \text{if } |X| \leq j < |X| + |Y|. \end{cases}$$

- **XOR (Modulo 2 Addition):**
  The symbol $\oplus$ represents bitwise exclusive-OR addition, defined as $Z = X \oplus Y$, for strings of equal length ($|X| = |Y|$).

- **Truncation:**
  The truncation of a bit string X to its first t bits is denoted by $\lfloor X \rfloor_t$.

Blocks and Padding
When processing data, strings are often divided into smaller chunks of n bits. We use lowercase italic letters to denote these blocks, using a subscript i to indicate their position in a sequence (e.g., m_i represents the i-th block of the message M).

To convert an arbitrary-length input string into a sequence of n-bit blocks, padding must be applied. The notation 10^* appends a single 1-bit to the end of the message, followed by the minimum number of 0-bits. This is used to make the total length of the padded message a multiple of n. We denote this padded string as $P = M \parallel 1 \parallel 0^*$.

Hexadecimal representation
Bit strings are represented in hexadecimal notation. Because MiniSubterranean-24 has a state of 24 bits, we use a 6-digit hexadecimal representation, where each hexadecimal character represents 4 bits of the 24-bit state. The notation is formatted with a `0x` prefix to indicate that it is a hexadecimal value, followed by the hexadecimal digits. These digits can be $0, 1, 2, \ldots, 9, \mathrm{a}, \mathrm{b}, \ldots, \mathrm{f}$, where $\mathrm{a} = 10, \mathrm{b} = 11, \ldots, \mathrm{f} = 15$

In this 6-digit format, every two hexadecimal characters (1 byte, or 8 bits) correspond to one row of the state matrix. The right-most two digits correspond to Row 0, the middle two digits to Row 1, and the left-most two digits to Row 2. Within each 8-bit row, bits are indexed from right to left, such that bit 0 represents the least significant bit (on the far right, index 0) and bit 7 represents the most significant bit (on the far left, index 7).

For example, the notation `0xaabbcc` corresponds to the following bit structure:

```
Row 2 (aa): 10101010  (bits 7..0)
Row 1 (bb): 10111011  (bits 7..0)
Row 0 (cc): 11001100  (bits 7..0)
```

The total state space ranges from `0x000000` (where all bits are set to 0) to `0xffffff`, which is equal to the state space of $2^{24} - 1$ (where all bits are set to 1).

2.2 The Farfalle Construction

Farfalle is a cryptographic construction used for building deck functions (Double-Extendable Cryptographic Keyed functions) [1]. A deck function takes a secret key and a sequence of input strings, and can produce an arbitrary-length output sequence that should be computationally indistinguishable from a random oracle for an adversary that does not know the secret key.

Farfalle processes data blocks in parallel and relies on four cryptographic transformations (p_b, p_c, p_d, and p_e), alongside two lightweight rolling functions (roll_c and roll_c) that efficiently generate unique sequence masks.

For the scope of this thesis, we focus specifically on the compression layer of the Farfalle construction. In this phase, an input message is divided into blocks m_i and is padded with 10^* so its total length becomes a multiple of n, where n is the bit-width of the compression transformation p_c. A unique mask K_i is generated for each block using the rolling function. Each n-bit block m_i is XORed with its mask K_i and independently processed through p_c. Finally, the outputs of all parallel p_c operations are XORed together into the accumulator.

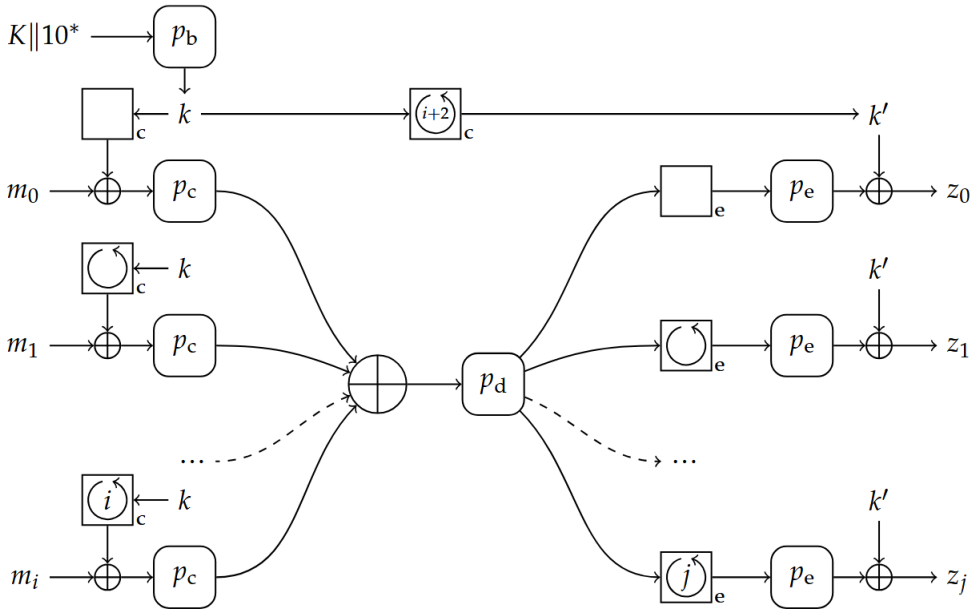


Figure 2.1: The Farfalle construction [1].

2.2.1 Cryptographic Collisions in Farfalle

Farfalle is designed as a *keyed deck function*, a primitive that uses a secret key K to generate arbitrary-length pseudo random output streams. While an adversary who does not know K can attempt various attack vectors against the construction, a collision attack is among the most dangerous. In a collision attack, the adversary aims to find two distinct input messages $M \neq M^*$ that produce identical output key streams $Z = Z^*$.

Because Farfalle processes input message blocks in parallel and sums the outputs into the accumulator, forcing an internal collision inside this accumulator would be devastating, as it guarantees matching output key streams regardless of the secret key.

Chapter 3

The Subterranean 3.0-P Transformation

In this chapter, we explain the structure of Subterranean 3.0-P and present the design specifications for our toy model, MiniSubterranean-24.

3.1 The Subterranean 3.0-P Transformation

Subterranean 3.0-P is an iterated cryptographic transformation that repeatedly applies a fixed non-linear round function $R : \mathbb{F}_2^n \to \mathbb{F}_2^n$ to an n-bit state. While Farfalle serves as the high-level deck function construction, a primitive like Subterranean 3.0-P acts as the underlying transformation engine, i.e., serving as the p_c transformation step.

Subterranean 3.0-P consists of six internal operations executed sequentially: five linear operations (ρ_{west}, θ, ι, ρ_{east}, π) and one non-linear operation (χ).

While section 2.1 introduces the state as a two-dimensional matrix of 3 rows and $n/3$ columns, the operations of the round function are defined over a one-dimensional bit array indexed by $k \in \{0, 1, \ldots, n-1\}$, with indices evaluated modulo n. The two representations are directly linked, for any bit position k, its row index $r \in \{0, 1, 2\}$ is given by $r = k \bmod 3$, and its column index c is given by $c = \lfloor k/3 \rfloor$. For the original Subterranean 3.0-P transformation, $n = 384$ bits (3×128 columns), whereas for our MiniSubterranean-24 model, $n = 24$ bits (3×8 columns).

The round function is the composition of the following six steps, executed in order:

- **Shift West (ρ_{west}):** A linear step that moves bits to new positions by adding an offset that depends on their row index (k mod 3).

$$\rho_{west} : x_k \leftarrow \begin{cases} x_k, & \text{for } k \bmod 3 = 0 \\ x_{k+3\cdot 94}, & \text{for } k \bmod 3 = 1 \\ x_{k+3\cdot 34}, & \text{for } k \bmod 3 = 2 \end{cases}$$

- **Mixing (θ):** A linear step that mixes bits across both rows and columns.

$$\theta : x_k \leftarrow x_k + x_{k+1} + x_{k+8} + x_{k+20}$$

- **Constant Addition (ι):** A linear addition of a round-dependent constant c to the first bit to break symmetry:

$$\iota : x_0 \leftarrow x_0 + c$$

- **Shift East (ρ_{east}):** Similar to the west shift, this linear step applies a different set of row-dependent offsets to move each bit to a new position:

$$\rho_{east} : x_k \leftarrow \begin{cases} x_k, & \text{for } k \bmod 3 = 0 \\ x_{k+3\cdot 13}, & \text{for } k \bmod 3 = 1 \\ x_{k+3\cdot 70}, & \text{for } k \bmod 3 = 2 \end{cases}$$

- **Transposition (π):** A linear mapping that moves each bit to a new position by multiplying its column index by the shuffling factor 19:

$$\pi : x_k \leftarrow x_{19\cdot k}$$

- **Non-linear Mapping (χ):** The only non-linear step in the transformation, it serves as the S-box layer. It updates each bit based on its neighbors in the same column.

$$\chi : x_k \leftarrow x_k + (x_{k+128} + 1) \cdot x_{k+256}$$

The notation MiniSubterranean-24[r] specifies the execution of r rounds of the compression transformation p_c, where $r \in \{1, 2, 3, 4, 5\}$. In Farfalle, the message block difference is injected directly before non-linear processing, meaning in the first round, $r = 1$, consists solely of the non-linear χ transformation. All subsequent rounds ($2 \leq i \leq r$) execute the full round transformation combining χ, π, ρ_{east}, ρ_{west}, θ, and ι.

Chapter 4

Differential propagation

This chapter presents the methods used in differential propagation, which are the approximation models that we compare our empirical findings with.

In Section 4.1, we formalize differential cryptanalysis, defining differential probability (DP), intermediate trails, active S-boxes, and weight metrics. Section 4.2 and 4.3 introduce trail clipping and trail clustering. These mechanisms cause observed empirical behavior to diverge from approximation models. Section 4.4 links these differential properties to Farfalle's accumulator collision and spurious trails. Finally, Section 4.5 explains the dynamic lookup methodology for exhaustive evaluation.

4.1 Theory of Differential Cryptanalysis

Differential cryptanalysis evaluates the security of cryptographic primitives by analyzing how input differences propagate through internal transformation layers to produce output differences. In this section, we formalize the methods to trace and approximate difference propagation through p_c.

4.1.1 Differential Probability (DP)

The Differential Probability (DP) measures the likelihood that a given input difference Δ_a propagates to a specific output difference Δ_b through a transformation p_c.

Let $p_c : \mathbb{Z}_2^n \to \mathbb{Z}_2^n$ denote a n-bit transformation operating on state space $\mathbb{Z}_2^n$. Consider two distinct input states, X and X^*. Their relationship is defined by their bitwise XOR difference, denoted as Δ_a:

$$\Delta_a = X \oplus X^* .$$

Given a base state X and a specified input difference Δ_a, the paired state is determined as $X^* = X \oplus \Delta_a$.

Passing both states through the transformation p_c gives their respective output states:

$$Y = p_c(X) \quad \text{and} \quad Y^* = p_c(X \oplus \Delta_a) .$$

The resulting output difference Δ_b is the XOR difference of the outputs:

$$\Delta_b = Y \oplus Y^* = p_c(X) \oplus p_c(X \oplus \Delta_a)\,.$$

To measure how predictably an input difference Δ_a propagates to the target output difference Δ_b across the entire state space, we compute its *Differential Probability*, denoted as $\mathrm{DP}(\Delta_a, \Delta_b)$.

The Differential Probability $\mathrm{DP}(\Delta_a, \Delta_b)$ of a differential $(\Delta_a \to \Delta_b)$ through transformation p_c is the number of input states $X \in \mathbb{Z}_2^n$ for which the input difference Δ_a results in the output difference Δ_b:

$$\mathrm{DP}(\Delta_a, \Delta_b) = \frac{\left|\{X \in \mathbb{Z}_2^n \mid p_c(X) \oplus p_c(X \oplus \Delta_a) = \Delta_b\}\right|}{2^n}\,.$$

4.1.2 Differentials vs. Differential Trails

In order to analyze the propagation of input differences through an r-round iterated transformation, we must distinguish between an end-to-end *differential* and a step-by-step *differential trail*.

`Differential:`
A differential over an r-round transformation p_c is defined by an initial input difference b_0 and a final output difference a_r, written as (b_0, a_r). This describes the overall transition across the transformation, regardless of what the intermediate differences are.

`Differential Trail:`
A differential trail, denoted as Q, specifies the step-by-step sequence of intermediate state differences across both the non-linear and linear layers. For an r-round transformation, the trail propagates through the sequential chain:

$$b_0 \xrightarrow{\chi} a_1 \xrightarrow{\lambda} b_1 \xrightarrow{\chi} a_2 \xrightarrow{\lambda} b_2 \longrightarrow \cdots \xrightarrow{\chi} a_r$$

where:

- b_0 is the initial input difference entering the non-linear χ layer in the first round.
- a_i represents the state difference exiting the non-linear χ layer in round i (for $1 \le i \le r$).
- b_i represents the state difference exiting the linear layers $\lambda = \pi \circ \rho_{\text{east}} \circ \iota \circ \theta \circ \rho_{\text{west}}$ in round i, entering χ in round $i+1$ via $b_i = \lambda(a_i)$ (for $1 \le i < r$).
- a_r represents the final state difference exiting round r, forming the target output difference $\gamma = a_r$.

Because operations in λ commute with bitwise XOR addition, i.e $\lambda(x \oplus \Delta) \oplus \lambda(x) = \lambda(\Delta)$, the linear layers preserve differences independently of the base state x. As a result, an entering state difference a_i maps deterministically to $b_i = \lambda(a_i)$ with a DP of 1 (and a DP of 0 for any other target difference).

The output difference of the non-linear χ layer depends on the base state x. Because different base states produce different output differences, an incoming difference b_i entering χ splits across multiple potential output differences a_{i+1}, introducing transition probabilities $DP \le 1$. Measured by $\mathrm{DP}(b_i, a_{i+1})$ of the χ layer.

4.1.3 Active S-boxes

The non-linear χ layer acts as the S-box layer of the Subterranean 3.0-P transformation, operating independently on $n/3$ parallel columns. For a state width of $n = 24$ bits, this corresponds to 8 parallel S-boxes per round, indexed by column position $j \in \{0, 1, \ldots, 7\}$.

When a state difference enters the χ layer, each column contains a 3-bit input difference $\Delta_{\text{in}} \in \mathbb{Z}_2^3$, ranging from `000` to `111`. Based on this incoming difference, each S-box is classified into one of two categories:

- **Inactive S-box:**
  An S-box is inactive if its incoming column difference is `000`. When no difference exists in a column, the output difference Δ_{out} is guaranteed to be `000` with a probability of 1 (DP = 1).
- **Active S-box:** An S-box is active if its incoming column difference contains at least one non-zero bit ($\Delta_{\text{in}} \neq$ `000`). For an active S-box, the output difference Δ_{out} depends directly on the base input state x. As a result, the transition probability to any single output difference is strictly less than 1 (DP < 1).

To illustrate this classification, consider an example state difference entering the χ layer:

```
Column Index:  7 6 5 4 3 2 1 0
Row 2:         1 0 0 0 0 1 0 0
Row 1:         1 0 0 0 0 0 0 0
Row 0:         0 1 0 0 0 1 0 0
```

Analyzing the state column-wise reveals that columns 2, 6, and 7 contain non-zero differences (`101`, `001`, and `110`, respectively), making them active S-boxes. The remaining five columns (0, 1, 3, 4, and 5) read `000` and are inactive.

The number of active S-boxes in a state difference b_i is determined by computing the bitwise OR ($\vee$) across its three rows and taking the Hamming weight (HW) of the resulting vector:

$$\text{Active S-boxes}(b_i) = \text{HW}(\text{Row}_0 \vee \text{Row}_1 \vee \text{Row}_2)$$

4.1.4 Cryptographic Weights

When evaluating differential propagation across multiple rounds, multiplying small probability fractions can quickly become cumbersome. For convenience and readability, differential cryptanalysis translates transition probabilities into an additive metric known as *weight*, denoted as w.

The weight w of a differential transition with probability DP is defined as its negative base-2 logarithm:

$$w = -\log_2(DP) \quad \Longleftrightarrow \quad DP = 2^{-w}\,.$$

To see how DP is calculated and why the base-2 logarithm produces clean integer weights, we must look at how the χ S-box is defined. In Subterranean 3.0-P, the χ transformation updates

each bit $x_{j,i}$ using the following rule in $\mathbb{F}_2$ (the finite field with two elements, where addition represents XOR and multiplication represents the logical AND gate):

$$\chi : x_{j,i} \leftarrow x_{j,i} + (x_{j,i+1} + 1) \cdot x_{j,i+2} \,.$$

The two cases, as mentioned before, are

- **Inactive S-box:**
  For an incoming difference $\Delta_{\text{in}} = \texttt{000}$, the output difference is $\Delta_{\text{out}} = \texttt{000}$ with probability $DP = 1$. Its weight contribution is $w = -\log_2(1) = 0$.
- **Active S-box:**
  For any non-zero incoming difference $\Delta_{\text{in}} \neq \texttt{000}$, evaluating the 8 possible 3-bit input states reveals that the non-linear `AND` logic splits the outcomes into 4 distinct output differences, each with 2 inputs. The transition probability through an active S-box is therefore $DP = \frac{2}{8} = 2^{-2}$. Its weight contribution is $w = -\log_2(2^{-2}) = 2$.

Every active S-box in a round difference b_i contributes 2 to the round weight $w(b_i)$. To compute the total round weight $w(b_i)$, we combine the three state rows using the bitwise OR ($\vee$) operation and compute the Hamming weight (HW) of the resulting vector:

$$w(b_i) = 2 \times \text{HW}(\text{Row}_0 \vee \text{Row}_1 \vee \text{Row}_2) \,.$$

Because our 24-bit model consists of 8 parallel S-boxes per round, the weight $w(b_i)$ of a single round transition is bounded between 0 (when all S-boxes are inactive) and a maximum of 16 (8 active S-boxes $\times$ 2).

The total weight of a trail $Q = (b_0, a_1, b_1, \ldots, a_r)$ is the sum of the weights of its individual round transitions:

$$w_{\text{total}}(Q) = \sum_{i=0}^{r-1} w(b_i) \,.$$

Under the assumption that active S-boxes in consecutive rounds operate independently, the approximate DP of trail Q propagating from start to finish is called its `Expected Differential Probability` (EDP) given as:

$$EDP(Q) = 2^{-w_{\text{total}}(Q)} \,.$$

The `Differential Probability of a trail`, denoted as $\text{DP}(Q)$, is the fraction of input states $X \in \mathbb{Z}_2^n$ that follow trail Q:

$$\text{DP}(Q) = \frac{N(Q)}{2^n} \,.$$

where $N(Q) = \left|\{X \in \mathbb{Z}_2^n \mid X \text{ satisfies trail } Q\}\right|$ represents the observed number of hitting input pairs (or *hits*). Under the independent S-box assumption, the approximate expected number of hits, denoted as $EN(Q)$, is given by:

$$EN(Q) = 2^n \times \text{EDP}(Q) = 2^{n - w_{\text{total}}(Q)} \,.$$

Throughout our empirical evaluation of the MiniSubterranean-24 model, we use $N(Q)$ to denote the observed hits and $EN(Q)$ for the approximate expected hits to analyze deviations from the independent S-box assumption.

To analyze the empirical distribution of output differences across the state space, we denote the total pair hit count for an input difference Δ_a resulting in an output difference γ as $N(\Delta_a, \gamma)$:

$$N(\Delta_a, \gamma) = \left|\{X \in \mathbb{Z}_2^n \mid p_c(X) \oplus p_c(X \oplus \Delta_a) = \gamma\}\right|.$$

The hit frequency distribution, denoted as $f(N)$, measures the cardinality of output differences γ that receive exactly N hits:

$$f(N) = \left|\{\gamma \in \mathbb{Z}_2^n \mid N(\Delta_a, \gamma) = N\}\right|.$$

This distribution $f(N)$ is used throughout our empirical results to visualize how hit counts disperse and cluster across successive rounds.

4.1.5 Difference Distribution Tables (DDT)

In differential cryptanalysis, the differential properties of an n-bit non-linear transformation $F : \mathbb{Z}_2^n \to \mathbb{Z}_2^n$ are characterized using a *Difference Distribution Table* (DDT). The DDT is a $2^n \times 2^n$ matrix where rows represent input differences Δ_a and columns represent output differences Δ_b.

Each entry $\mathrm{DDT}(\Delta_a, \Delta_b)$ records the number of base states $X \in \mathbb{Z}_2^n$ for which an input difference Δ_a propagates to an output difference Δ_b:

$$\mathrm{DDT}(\Delta_a, \Delta_b) = \#\{X \in \mathbb{Z}_2^n \mid F(X \oplus \Delta_a) \oplus F(X) = \Delta_b\}.$$

Dividing each entry by the total state space size 2^n gives the Differential Probability (DP):

$$\mathrm{DP}(\Delta_a, \Delta_b) = \frac{\mathrm{DDT}(\Delta_a, \Delta_b)}{2^n}.$$

4.2 Trail Clipping

`Trail clipping` is a property of a single differential trail Q. It occurs when the empirical probability $\mathrm{DP}(Q)$ diverges from its approximate Expected Differential Probability ($\mathrm{EDP}(Q)$):

$$\mathrm{DP}(Q) \neq \mathrm{EDP}(Q).$$

While $\mathrm{EDP}(Q)$ assumes that S-box transitions in round i are independent of round $i+1$, the linear layer λ deterministically routes bits between consecutive rounds. These dependencies cause the actual state transition counts to diverge from independent probability calculations:

- If inter-round bit routing satisfies the S-box conditions of round $i+1$ more frequently than predicted by independent probabilities, $\mathrm{DP}(Q) > \mathrm{EDP}(Q)$.
- If inter-round bit routing creates conflicts between consecutive rounds, $\mathrm{DP}(Q) < \mathrm{EDP}(Q)$. In extreme cases, these conflicts make a trail impossible ($N(Q) = 0$), despite giving a calculated $\mathrm{EDP}(Q) > 0$.

A critical case of clipping occurs when a trail's total weight equals or exceeds the state size $w_{\mathrm{total}}(Q) \geq n$, i.e., $w_{\mathrm{total}}(Q) \geq 24$. For such heavy trails, the approximate expected hit count drops below one pair:

$$EN(Q) = 2^{n - w_{\mathrm{total}}(Q)} \leq 1.$$

Because base states evaluated with a non-zero input difference Δ_a always occur in symmetric pairs $(X, X \oplus \Delta a)$, any active empirical trail must record at least an integer pair of hits $N(Q) \geq 2$. Therefore, an observed trail with $w_{\text{total}}(Q) \geq 24$ exhibits severe clipping if its empirical probability $DP(Q) = \frac{N(Q)}{2^{24}} \geq 2^{-23}$ is constrained by this discrete lower bound, significantly exceeding its approximate expectation $EDP(Q) = 2^{-w_{\text{total}}(Q)}$.

4.3 Trail Clustering

A differential (b_0, a_r) specifies only the initial input difference b_0 and the final output difference a_r, abstracting away from the internal round transitions. Because different base input states x may follow different trails before arriving at the same final output difference a_r, a single differential (b_0, a_r) consists of a set (or cluster) of distinct trails, denoted as $\Omega(b_0, a_r)$.

The Differential Probability of the differential, $\mathrm{DP}(b_0, a_r)$, is the sum of the probabilities of all trails belonging to that cluster:

$$\mathrm{DP}(b_0, a_r) = \sum_{Q \in \Omega(b_0, a_r)} \mathrm{DP}(Q) = \frac{\sum_{Q \in \Omega(b_0, a_r)} N(Q)}{2^n} .$$

Similarly, the approximate Expected Differential Probability, $EDP(b_0, a_r)$, of the differential is the sum of the expected probabilities of its constituent trails:

$$\mathrm{EDP}(b_0, a_r) = \sum_{Q \in \Omega(b_0, a_r)} \mathrm{EDP}(Q) .$$

Trail clustering occurs when multiple distinct trails accumulate towards the same end-to-end differential (b_0, a_r), causing the total DP to exceed that of any individual trail.

When significant clustering occurs, evaluating the security of a differential based solely on a single dominant trail Q underestimates the differential probability $\mathrm{DP}(b_0, a_r)$, as the combined contribution of all clustered trails elevates the overall DP of reaching the output difference a_r.

4.4 Accumulator Collision attack

Farfalle splits an input message into sequence blocks $m_0, m_1, \ldots, m_{x-1}$. Each block m_i is XORed with a key-derived mask K_i, processed through the compression transformation p_c (Subterranean 3.0-P), and accumulated via bitwise XOR addition:

$$Y = \bigoplus_{i=0}^{x-1} p_c(m_i \oplus K_i) .$$

For an adversary to perform a collision attack on the Farfalle construction, they would need to construct a paired message M^* from an original message M by introducing bit-flip masks into specific blocks $(m_i^* = m_i \oplus \Delta m_i)$. For any block i, let $X_i = m_i \oplus K_i$ denote the input state entering p_c for Message 1, and $X_i^* = m_i^* \oplus K_i = (m_i \oplus \Delta m_i) \oplus K_i$ denote the input state entering p_c for Message 2.

Because bitwise key addition is linear, the secret key K_i and the original message content m_i cancel out completely when calculating the input difference ΔX_i entering transformation p_c:

$$\Delta X_i = X_i^* \oplus X_i = (m_i \oplus \Delta m_i \oplus K_i) \oplus (m_i \oplus K_i) = \Delta m_i \,.$$

Thus, the input difference entering p_c is equal to the chosen bit-flip mask Δm_i, giving the adversary control over ΔX_i regardless of the unknown secret key K.

Two-Block Accumulator Collision Strategy

In this thesis, we focus exclusively on two-block accumulator collisions, which exploit the parallel combination logic of Farfalle. We assume that an active input difference in a single block produces a non-zero output difference, meaning that accumulator collision requires matching output differences across at least two distinct message blocks.

To force an accumulator collision $\Delta Y = 0$, an adversary must inject a non-zero difference $n_0 \neq 0$ into at least two distinct message blocks, for example setting $\Delta m_0 = b_0$ ($m_0^* = m_0 \oplus b_0$) and $\Delta m_1 = b_0$ ($m_1^* = m_1 \oplus b_0$), while leaving all remaining blocks unmodified ($\Delta m_i = 0$ for $i \geq 2$).

The input differences Δm_0 and Δm_1 do not need to be identical, collision occurs whenever both blocks have matching output differences. Let Δm_0 and Δm_1 enter p_c, producing non-zero output differences $\gamma_1 \neq 0$ and $\gamma_2 \neq 0$:

$$\gamma_1 = p_c(m_0 \oplus \Delta m_0 \oplus K_0) \oplus p_c(m_0 \oplus K_0) \,.$$
$$\gamma_2 = p_c(m_1 \oplus \Delta m_1 \oplus K_1) \oplus p_c(m_1 \oplus K_1) \,.$$

Inside the accumulator, all block output differences sum together:

$$\Delta Y = \gamma_1 \oplus \gamma_2 \oplus \Delta y_2 \oplus \cdots = \gamma_1 \oplus \gamma_2 \oplus 0 \cdots = \gamma_1 \oplus \gamma_2 \,.$$

If both independent blocks happen to propagate to the exact same target output difference ($\gamma_1 = \gamma$ and $\gamma_2 = \gamma$), they cancel each other out in the accumulator:

$$\Delta Y = \gamma_1 \oplus \gamma_2 = \gamma \oplus \gamma = 0$$

When the accumulator difference cancels to zero ($\Delta Y = 0$), the expansion transformation p_e processes identical accumulator states for both messages ($Y^* = Y$), generating identical output key streams ($Z = Z^*$) and completing the key-independent collision.

4.4.1 Collision Probability Calculation (CP)

In the general setting where two message blocks receive distinct input differences Δm_0 and Δm_1, the probability of an accumulator collision across all output differences γ is:

$$CP(\Delta m_0, \Delta m_1) = \sum_{\gamma \in \mathbb{F}_2^n} \mathrm{DP}(\Delta m_0, \gamma) \cdot \mathrm{DP}(\Delta m_1, \gamma)$$

In this thesis, we focus on the case where the adversary injects an identical baseline input difference into both active blocks ($\Delta m_0 = \Delta m_1 = b_0$).

Because p_c is non-linear, the propagation of input difference b_0 into a specific output difference γ across all 2^n base states is calculated with Differential Probability $\mathrm{DP}(b_0, \gamma)$.

For a specific target output difference γ:

- Block 0 yields output difference $\gamma_1 = \gamma$ with probability $\mathrm{DP}(b_0, \gamma)$.
- Block 1 yields output difference $\gamma_2 = \gamma$ with probability $\mathrm{DP}(b_0, \gamma)$.

Because message blocks are processed independently by p_c, the joint probability that both blocks hit the same target difference γ (enabling collision $\gamma \oplus \gamma = 0$) is the product of their individual probabilities:

$$\mathrm{DP}(b_0, \gamma) \times \mathrm{DP}(b_0, \gamma) = \mathrm{DP}(b_0, \gamma)^2$$

An accumulator collision occurs if the two blocks match on any valid output difference $\gamma \in \mathbb{F}_2^n$. Therefore, the total Collision Probability (CP) for a specified input difference b_0 is calculated by summing these squared joint probabilities across all 2^n possible output differences:

$$CP(b_0) = \sum_{\gamma \in \mathbb{Z}_2^n} \mathrm{DP}(b_0, \gamma)^2$$

4.4.2 Spurious Trails

Weight-based approximations, such as those used in the security analyses of Farfalle [1] and Xoodoo [2], estimate collision resistance using only the single dominant prefix differential trail prefix Q_{dom}, which has a prefix-restricted collision probability:

$$CP_{\mathrm{prefix}}(b_0) = 2^{-\left(2\sum_{i=0}^{r-2} w(b_i) + w(b_{r-1})\right)}.$$

Continuous mixing and shuffling of bits create the potential for state transitions to fragment across thousands of alternative, non-dominant trails $Q \in \Omega(b_0, \gamma) \setminus \{Q_{\mathrm{dom}}\}$. Due to this dispersion, the single dominant trail prefix assumption is not reliable, and evaluating only the collision probability of Q_{dom} risks underestimating the total accumulator collision probability.

For this reason, we define the $\mathrm{Ratio}_{\mathrm{spurious}}$ to measure the total collision contribution of these alternative trails relative to the dominant prefix as:

$$\mathrm{Ratio}_{\mathrm{spurious}}(b_0) = \frac{CP_{\mathrm{exact}}(b_0)}{CP_{\mathrm{prefix}}(b_0)} = \frac{\sum_{\gamma \in \mathbb{Z}_2^n} \mathrm{DP}(b_0, \gamma)^2}{2^{-\left(2\sum_{i=0}^{r-2} w(b_i) + w(b_{r-1})\right)}}.$$

A ratio of $\mathrm{Ratio}_{\mathrm{spurious}}(b_0) \approx 1$ indicates that collision behavior is governed almost entirely by the dominant trail prefix. When $\mathrm{Ratio}_{\mathrm{spurious}}(b_0) > 1$, the vast majority of collisions originate from secondary trails, demonstrating that single-trail prefix approximations underestimate the true collision probability.

In our evaluation of the 24-bit MiniSubterranean-24 model, we compare collision resistance across two distinct methods:

1. **Exact Empirical Evaluation ($CP_{\mathbf{exact}}$):**
   We execute an exhaustive evaluation across all 2^{24} base states to determine the exact pair

hit counts $N(b_0, \gamma)$. This yields the empirical differential probabilities $\mathrm{DP}(b_0, \gamma) = \frac{N(b_0,\gamma)}{2^{24}}$, from which the exact collision probability is calculated as:

$$CP_{\text{exact}}(b_0) = \sum_{\gamma \in \mathbb{Z}_2^n} \mathrm{DP}(b_0, \gamma)^2 .$$

2. **Weight-Based Approximation (CP_{prefix}):**
To estimate collision probability without exhaustive measurement, approximate analyses rely on a dominant trail prefix, a fixed sequence of minimal-weight state transitions across early rounds ($b_0 \to b_1 \to \cdots \to b_{r-1}$) assumed to carry the overwhelming majority of differential probability:

$$CP_{\text{prefix}}(b_0) = 2^{-\left(2\sum_{i=0}^{r-2} w(b_i) + w(b_{r-1})\right)} .$$

This weight-based estimation relies on three key assumptions:

- It assumes S-box transitions across rounds operate independently, neglecting trail clipping.
- It assumes probability is concentrated entirely within a single trail, neglecting trail clustering.
- It ignores spurious trails, alternative, higher-weight trails starting at b_0 that diverge from the dominant trail prefix.

Comparing the empirical measurements of $CP(b_0)$ against this weight-based approximation highlights the combined impact of trail clipping, trail clustering, and spurious trails on Farfalle's collision resistance.

4.5 Empirical Evaluation

Generating and storing a complete DDT, even for our MiniSubterranean-24 model, is impractical under the available memory constraints. Because the state is 24 bits, a full 2D matrix would require $2^{24} \times 2^{24} = 2^{48}$ entries, which exceeds memory limits. Therefore, we do not compute the entire DDT.

Instead, we make use of a dynamic approach. We pre-compute a one-dimensional state transition lookup table T of size 2^{24} storing the output state for every possible input state $T[X] = p_c(X)$. With this table loaded into memory, we evaluate differential propagation on the fly. For a given input difference Δ_a, we iterate across all 2^{24} base states X to compute and dynamically tally the resulting output differences $\Delta Y = T[X] \oplus T[X \oplus \Delta_a]$. This procedure is formalized in Algorithm 1 . While this Algorithm outlines the core procedure for end-to-end output differences ΔY to compute CP_{exact}, our implementation also logs the intermediate round differences $(a_1, a_2, \ldots, a_r)$ on the fly to evaluate individual trail hits $N(Q)$, trail clipping, and differential clustering.

The first round of p_c is restricted solely to the χ transformation, after which full round iterations apply the complete composite transformation: $r = \chi \circ \pi \circ \rho_{\text{east}} \circ \iota \circ \theta \circ \rho_{\text{west}}$

Algorithm 1 Dynamic Output Difference Evaluation using State Lookup Table

Require: Δ_a ▷ Target input difference (e.g., active in a single S-box)
Require: T ▷ Pre-computed state transition lookup table of size 2^{24}
$N \leftarrow$ Array of size 2^{24} initialized to 0
for $X \leftarrow 0$ **to** $2^{24} - 1$ **do**
    $X^* \leftarrow X \oplus \Delta_a$ ▷ Derive the paired input state
    $Y \leftarrow T[X]$ ▷ Lookup output for the base state
    $Y^* \leftarrow T[X^*]$ ▷ Lookup output for the paired state
    $\Delta_b \leftarrow Y \oplus Y^*$ ▷ Calculate the resulting output difference
    $N[\Delta_b] \leftarrow N[\Delta_b] + 1$ ▷ Tally the occurrence $N(\Delta_a, \Delta_b)$
end for
return N ▷ Entries represent $N(\Delta_a, \Delta_b)$; dividing by 2^{24} yields $\mathrm{DP}(\Delta_a \to \Delta_b)$

4.5.1 Target Input Differences: Single Active S-Box Baselines

In our empirical evaluation, we focus on three baseline input differences: `0x000001`, `0x000100`, and `0x010000`. These starting values represent the cases where one bit is flipped in the least significant column (column 0) across Row 0, Row 1, and Row 2, respectively.

Due to the horizontal shift-invariance of the Subterranean 3.0-P round function, shifting an input difference horizontally across columns returns identical differential properties. Therefore, restricting the active bit to column 0 is done without loss of generality.

The baselines are represented as 3×8 bit matrices where rows correspond to Row 0, Row 1, and Row 2 from top to bottom, and columns are indexed 0 to 7 from right to left. These three starting input difference Δ_a are:

$$\Delta_{a0} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & \mathbf{1} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} = \texttt{0x000001}$$

$$\Delta_{a1} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \mathbf{1} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} = \texttt{0x000100}$$

$$\Delta_{a2} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \mathbf{1} \end{pmatrix} = \texttt{0x010000}$$

Chapter 5

Results & Analysis

In this chapter, we analyze the empirical differential propagation of MiniSubterranean-24[r] across the rounds $r \in \{1, 2, 3, 4, 5\}$. By executing the dynamic lookup evaluation presented in Algorithm 1 for minimal-weight input differences Δ_a, we record the output hit counts $N(\Delta_a, \Delta_b)$ across all 2^{24} paired states. Using these empirical measurements, we investigate how differences disperse across rounds, examine the impact of trail clipping, trail clustering, and spurious trails, and evaluate their effect on the collision probability within the Farfalle accumulator.

5.1 Experimental Setup and Evaluation Metrics

To compare weight-based theoretical approximations against our empirical measurements, we define the following evaluation metrics:

Clipping Ratio
The clipping ratio, $\text{Ratio}_{\text{clipping}}(Q)$, measures the divergence of empirical trail hits from the approximate expected hits, under the assumption of independent S-boxes:

$$\text{Ratio}_{\text{clipping}}(Q) = \frac{N(Q)}{EN(Q)} .$$

Clustering Ratio
The clustering ratio, $\text{Ratio}_{\text{clustering}}(a, \gamma)$, measures the amplification of a differential transition $(a \to \gamma)$ when multiple distinct trails merge into the same target output difference γ, relative to the EDP of dominant trail Q_{dom}:

$$\text{Ratio}_{\text{clustering}}(a, \gamma) = \frac{N(a, \gamma)}{EN(Q_{\text{dom}})} .$$

Spurious Trail Ratio
The spurious trail ratio, $\text{Ratio}_{\text{spurious}}(b_0)$, measures the proportion of collision-producing state pairs that diverge into alternative trails outside of the dominant trail prefix:

$$\text{Ratio}_{\text{spurious}}(b_0) = \frac{CP(b_0)}{CP_{\text{prefix}}(b_0)} .$$

5.2 Empirical Measurements vs. Approximations in Early Rounds

For the first two rounds $r \in \{1, 2\}$, the empirical measurements align closely with the weight-based approximations, showing no evidence of trail clipping or clustering and only a negligible contribution from spurious trails in round 2.

5.2.1 MiniSubterranean-24[1]

Single-round propagation of MiniSubterranean-24[1] consists solely of the non-linear layer χ. For a baseline input difference b_0, only a single S-box is active with $w = 2$, which splits the input difference evenly across four valid output differences γ. Each valid output state accumulates a cluster hit count of $N(b_0, \gamma) = 2^{24-2} = 4,194,304$, giving a frequency of $f(N) = 4$ for $N = 4,194,304$ and $f(N) = 0$ for all $N \neq 4,194,304$.

Figure 5.1 shows the single-round cluster hit count distribution $f(N)$. Table 5.1 details the four differential trails propagating from baseline input difference $\Delta_{a0} = \texttt{0x000001} \rightarrow \gamma$, alongside their respective active trail weights w, approximate expected hits $EN(Q)$, observed hits $N(Q)$, Expected Differential Probability $EDP(Q)$, Differential Probability $DP(Q)$, and clipping ratio $\text{Ratio}_{\text{clipping}}$.

Since $N(Q) = EN(Q)$, the clipping ratio $\text{Ratio}_{\text{clipping}} = 1.0$, meaning that empirical observations align perfectly with the weight-based approximation in the single-round propagation. The differential probability of each valid trail is extremely high at $DP(Q) = 0.25$. This high probability occurs because a single-bit input difference activates only a single S-box ($w = 2$), and the non-linear layer χ interacts with its adjacent bits. A 1-bit input difference cannot spread across the state space during χ alone.

Also, no differential clustering occurs in single-round propagation, as each valid target output difference γ is reached by one isolated trail $|\Omega| = 1$. The clustering ratio is $\text{Ratio}_{\text{clustering}} = 1.0$.

Following the same reasoning as the clipping and clustering ratios, the spurious trail ratio is $\text{Ratio}_{\text{spurious}}(b_0) = 1.0$. Applying the empirical collision probability $CP(\Delta_a)$ alongside the prefix-restricted probability $CP_{\text{prefix}}(\Delta_a)$:

$$\text{Ratio}_{\text{spurious}}(\Delta_a) = \frac{CP(\Delta_a)}{CP_{\text{prefix}}(\Delta_a)} = \frac{2^{-2}}{2^{-2}} = 1.0\,.$$

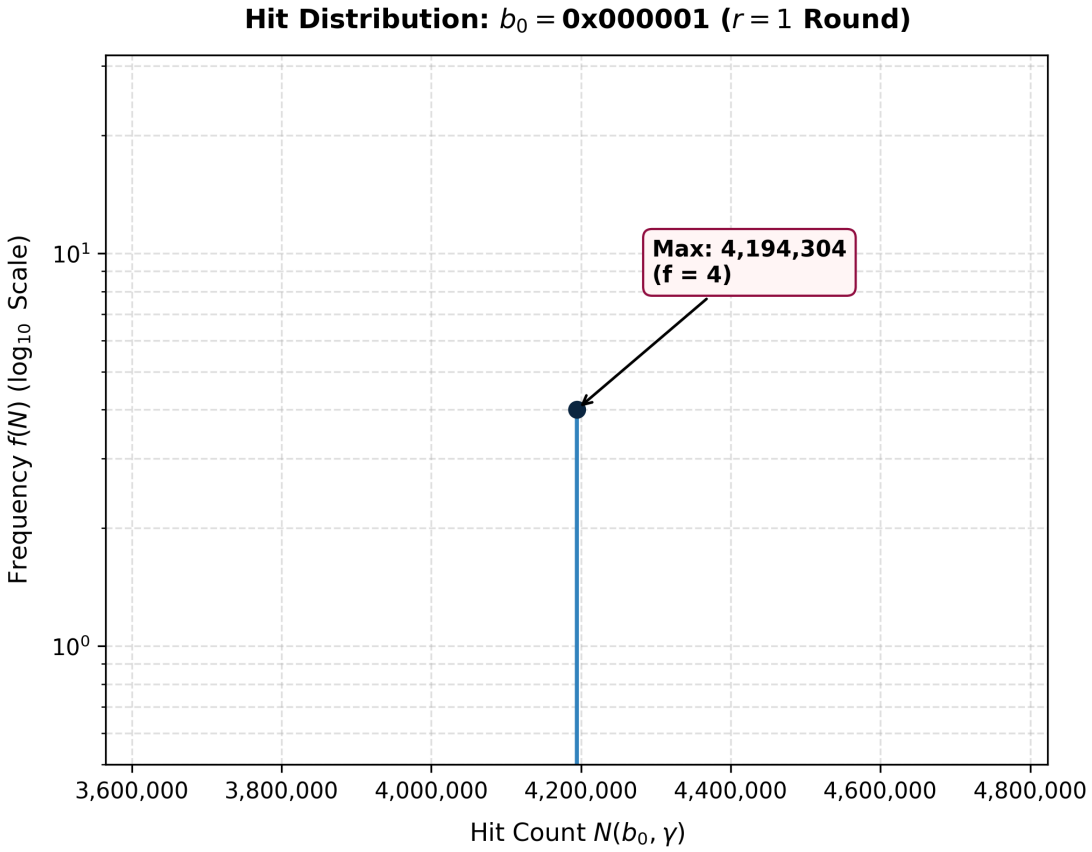


Figure 5.1: Hit count distribution $f(N)$ for baseline $\Delta_{a0} = 0x000001$ in MiniSubterranean-24[1].

γ	$N(Q)$	w	$EN(Q)$	$\mathbf{Ratio_{clipping}}$	$EDP(Q)$	$DP(Q)$
`010001`	4,194,304	2	4,194,304	1.0	0.25	0.25
`010101`	4,194,304	2	4,194,304	1.0	0.25	0.25
`000001`	4,194,304	2	4,194,304	1.0	0.25	0.25
`000101`	4,194,304	2	4,194,304	1.0	0.25	0.25

Table 5.1: Single-round differential trail propagation for baseline input $a =$ `0x000001`.

5.2.2 MiniSubterranean-24[2]

In MiniSubterranean-24[2], the execution sequence of the transformation consists of $\chi \to \lambda \to \chi$, where λ represents the linear layer $\pi \circ \rho_{\text{east}} \circ \iota \circ \theta \circ \rho_{\text{west}}$. Across this sequence, we find around 74,000 unique trails for the three baselines, with a total trail weight ranging from $W = 8$ up to a maximum of $W = 18$.

Figure 5.2 illustrates the resulting hit count distribution $f(N)$ for baseline $\Delta_{a2} =$ `0x010000`. Due to the linear mixing layer, more S-boxes are activated, which causes the maximum individual trail hit count to reduce from $4,194,304$ ($W = 2$) down to $N(Q) = 65,536$ ($W = 8$, $f = 64$).

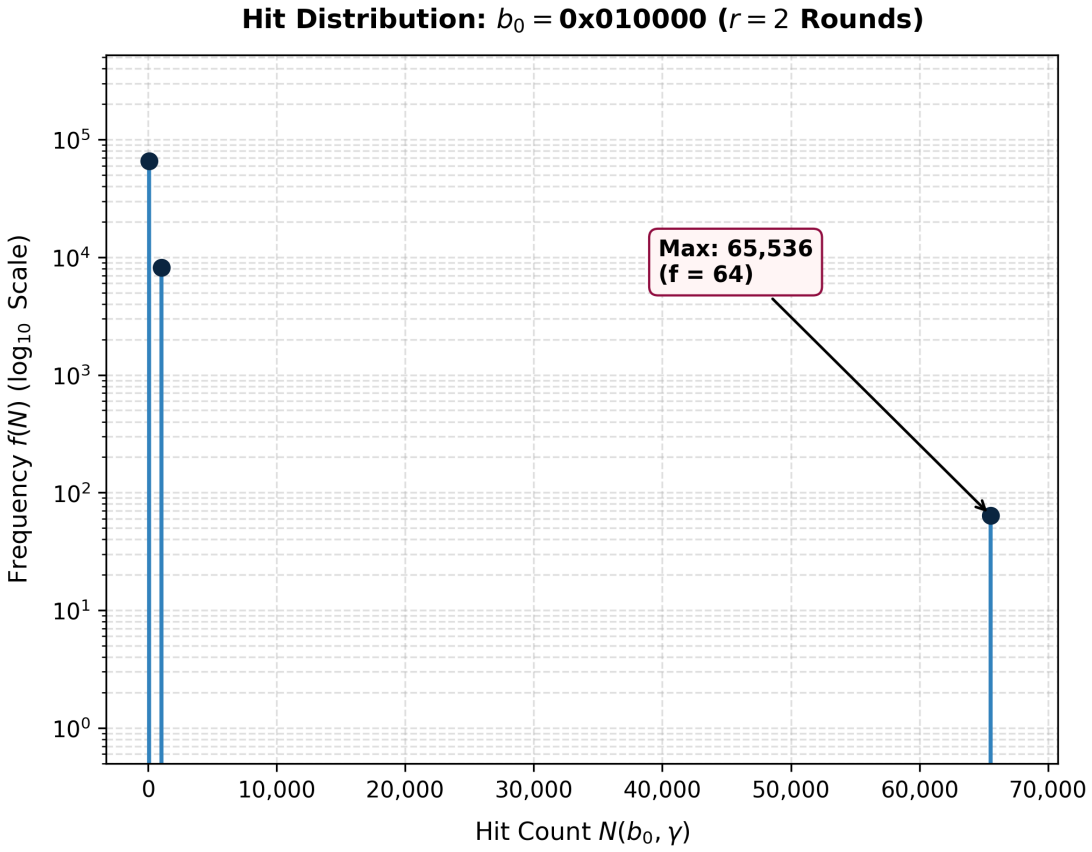


Figure 5.2: Hit count distribution $f(N)$ for baseline $\Delta_{a2} = 0x010000$ in MiniSubterranean-24[2].

For all the unique trails in MiniSubterranean-24[2], the empirical hit count aligns with the weight-based approximation $N(Q) = EN(Q)$, which spans from $N(Q) = 65,536$ ($W = 8$) down to $N(Q) = 64$ ($W = 18$), Table 5.2 shows this for Δ_{a2}. This results in a clipping ratio of $\text{Ratio}_{\text{clipping}} = 1.0$ across all trails. This ratio is also observed for the other two baselines, Δ_{a0} & Δ_{a1}. The only difference is a slightly higher minimum trail weight of $W = 10$, giving an individual hit count of $N(Q) = 16,384$ with a frequency of $f(16,384) = 256$.

$r = 1$	$r = 2$	$N(Q)$	W	$EN(Q)$	$\textbf{Ratio}_{\textbf{clipping}}$	$EDP(Q)$	$DP(Q)$
`010000`	`410044`	65,536	8	65,536	1.0	2^{-8}	2^{-8}
`010101`	`1eae75`	64	18	64	1.0	2^{-18}	2^{-18}

Table 5.2: Two-round differential propagation for baseline input $\Delta_{a2} =$ `0x010000`.

Differential trail clustering also does not occur in MiniSubterranean-24[2]. Every target output difference γ is reached by one isolated trail ($|\Omega(\Delta_a, \gamma)| = 1$). For example, for baseline Δ_{a2}, the cluster hit count for the dominant trail is $N(\Delta_{a2}, \gamma) = N(Q_{\text{dom}}) = 65,536$. Relative to the weight-based approximate hits $EN(Q_{\text{dom}}) = 2^{24-8} = 65,536$, this yields $\text{Ratio}_{\text{clustering}} = 1.0$. The same isolated single-trail behavior holds across all output differences for baselines Δ_{a0} and

Δ_{a1}.

The spurious trail ratio increases slightly. For example, for baseline Δ_{a2}, we observe $\text{Ratio}_{\text{spurious}}(\Delta_{a2}) \approx 1.0322$ when comparing the empirical collision probability $CP_{\text{exact}}(\Delta_{a2})$ against the prefix approximation $CP_{\text{prefix}}(\Delta_{a2})$ using the dominant trail weights $w_0 = 2$ and $w_1 = 6$.

This subtle 0.0322 increase occurs because the linear mixing layer λ disperses a small fraction of transitions into higher-weight secondary trails that activate additional S-boxes across the two rounds. Baselines Δ_{a0} and Δ_{a1} have the same minimal deviation, confirming that while spurious trails begin to emerge in round 2, their contribution to total collisions remains negligible.

5.3 Distribution of hits from MiniSubterranean-24[3] to MiniSubterranean-24[5]

Before analyzing individual cryptanalytic phenomena in isolation, this section presents the overall empirical hit count distributions for MiniSubterranean-24[3] through MiniSubterranean-24[5]. The figures below plot the hit frequency $f(N)$ against the observed pair hit counts $N(b_0, \gamma)$ across all 2^{24} possible output differences.

These distribution plots provide the baseline reference for the subsequent sections, where we examine the mechanics of trail clipping, differential clustering, and spurious trails separately.

5.3.1 MiniSubterranean-24[3]

In MiniSubterranean-24[3], the transformation sequence expands to five operations ($\chi \to \lambda \to \chi \to \lambda \to \chi$). The repeated linear mixing between non-linear layers causes massive trail dispersion, generating roughly 5.7 to 6.5 million unique active differential trails per baseline: $5,729,020$ trails for Δ_{a1}, $5,844,684$ trails for Δ_{a2}, and $6,493,992$ trails for Δ_{a0}. The resulting total trail weights span from $W = 14$ up to $W = 34$.

Figure 5.3 illustrates the hit count distribution $f(N)$ for the second baseline input $\Delta_{a1} =$ `0x000100`. The distribution plot is noticeably denser, showing a scattering of low-frequency trails. Within this baseline, individual trail weights range from $W = 16$ ($N(Q) = 512$) to $W = 34$ ($N(Q) = 8$), with the maximum observed hit count reaching $N(\Delta_{a1}, \gamma) = 976$ at a frequency of $f(976) = 1$.

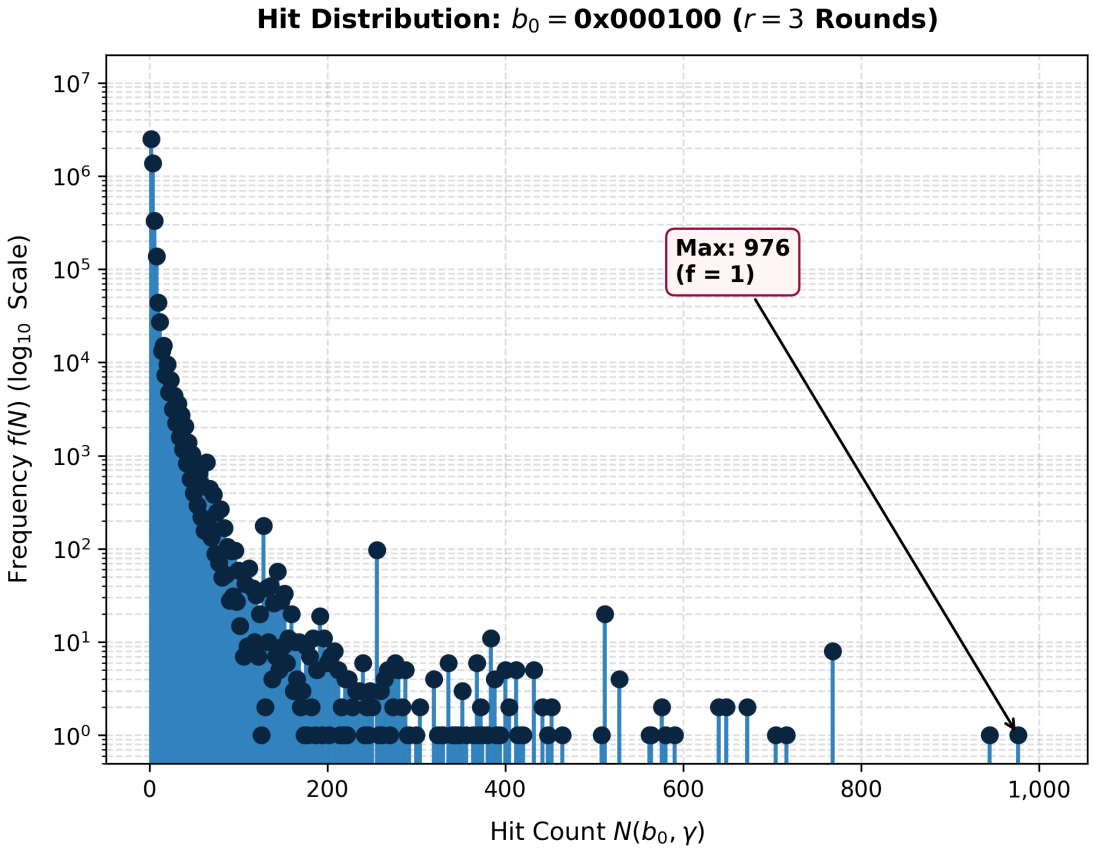


Figure 5.3: Hit count distribution $f(N)$ for baseline $\Delta_{a1} = 0x000100$ in MiniSubterranean-24[3].

Figure 5.4 displays the hit count distribution for the third baseline input $\Delta_{a2} = \texttt{0x010000}$. The trail weights range from $W = 14$ ($N(Q) = 1024$) to $W = 34$ ($N(Q) = 8$). The highest observed hit count is $N(\Delta_{a2}, \gamma) = 3,072$ with a frequency of $f(3,072) = 8$.

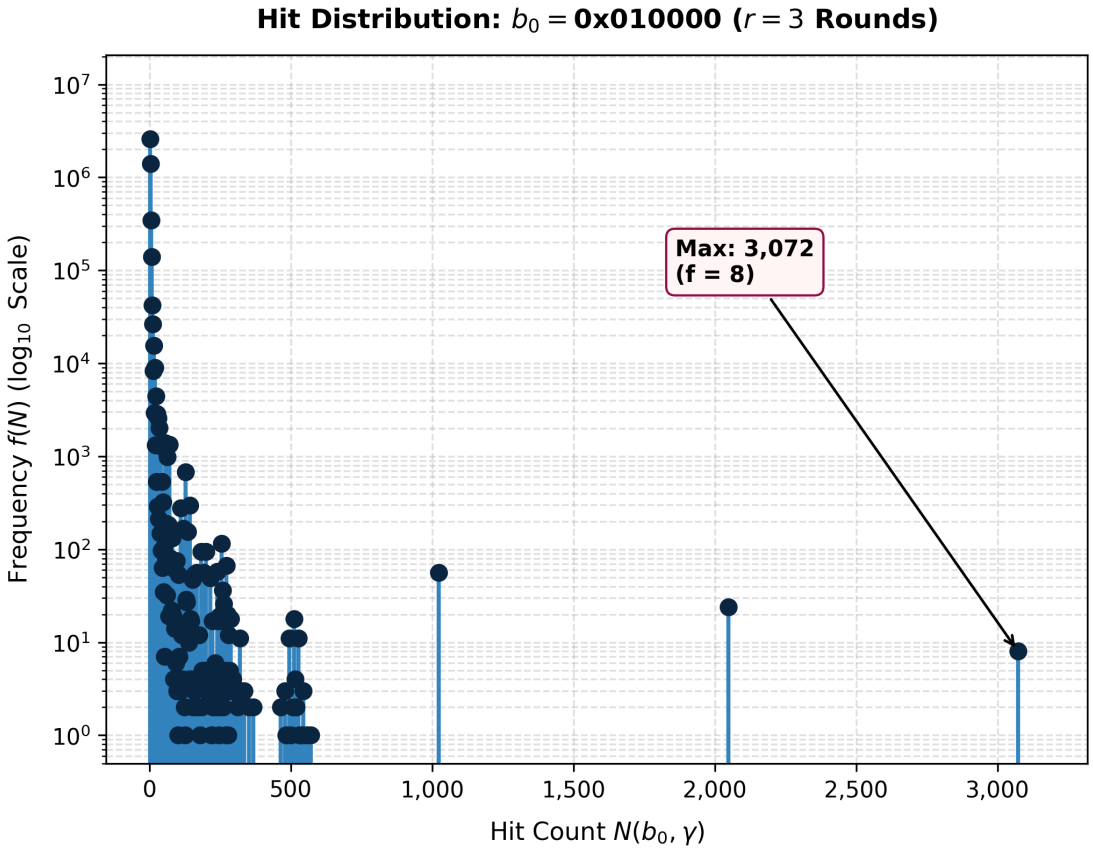


Figure 5.4: Hit count distribution $f(N)$ for baseline $\Delta_{a2} = 0x010000$ in MiniSubterranean-24[3].

5.3.2 MiniSubterranean-24[4] & MiniSubterranean-24[5]

Moving to MiniSubterranean-24[4] and MiniSubterranean-24[5], the hit count distributions flatten out significantly as differences disperse across the state space.

In MiniSubterranean-24[4], large clusters mostly disappear. For baseline $\Delta_{a2} = \texttt{0x010000}$, shown in figure 5.5, the peak hit count drops down to $N(\Delta_{a2}, \gamma) = 58$, occurring only once ($f(58) = 1$).

The same occurs in MiniSubterranean-24[5], as illustrated for baseline $\Delta_{a1} = \texttt{0x000100}$ in figure 5.6, the distribution decreases further, with the maximum observed hit count reaching only $N(\Delta_{a1}, \gamma) = 20$ with a frequency of $f(20) = 3$.

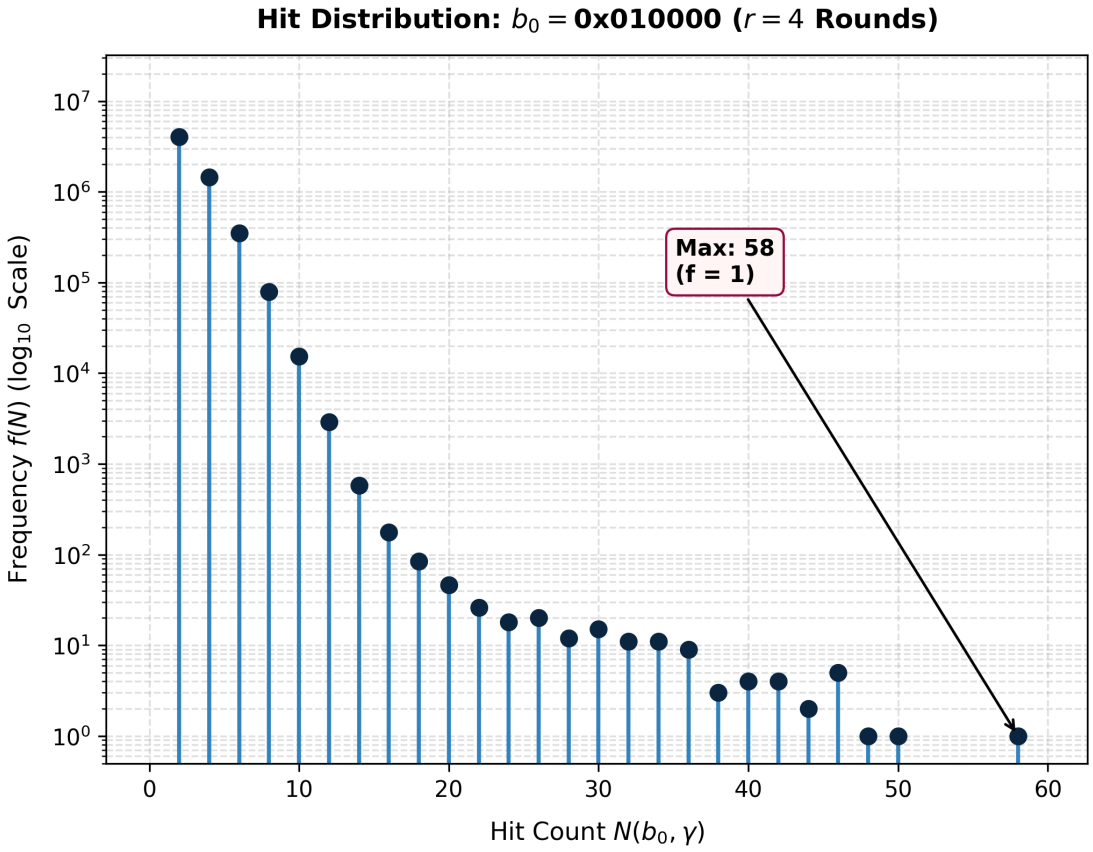


Figure 5.5: Hit count distribution $f(N)$ for baseline $\Delta_{a2} = \texttt{0x010000}$ in $MiniSubterranean-24$[4].

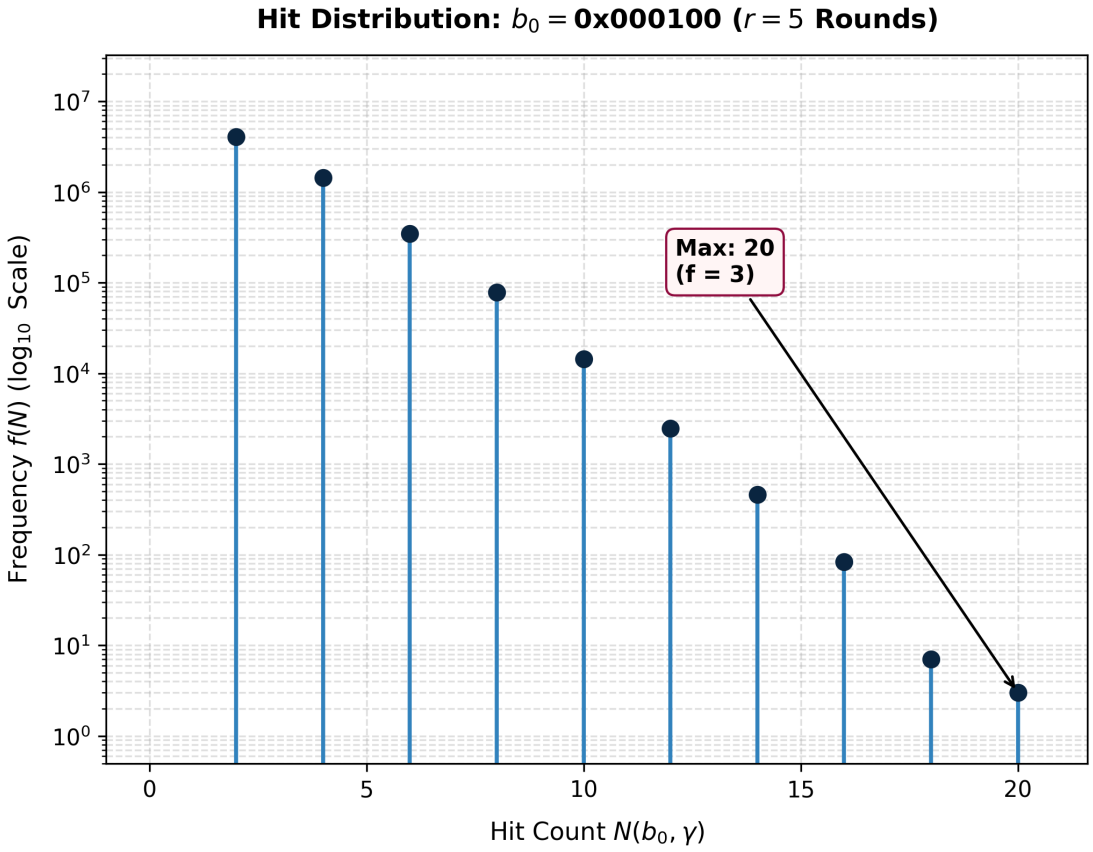


Figure 5.6: Hit count distribution $f(N)$ for baseline Δ_{a1} = 0x000100 in $MiniSubterranean-24[5]$.

5.4 Trail Clipping Analysis

In this section, we look at trail clipping that starts to occur from MiniSubterranean-24[3] up to MiniSubterranean-24[5].

5.4.1 The first sign of Clipping in MiniSubterranean-24[3]

Figures 5.7, 5.8 & 5.9 plot the observed trail hits $N(Q)$ against approximate expected hits $EN(Q)$ for $\Delta_{a0}, \Delta_{a1}, \Delta_{a2}$ over the total weights of MiniSubterranean-24[3]. From the graphs, we can conclude the following:

1. $W < 24$**:**
   For trail weights below 24, the approximate weight-based hits remain at or above one hit $EN(Q) \geq 1$. In this region, the measured trail hits (blue dots) closely track the approximate expected line (red dashed line). However, we can still see several points landing above the line. This upward scatter occurs because differential clustering is also

active in MiniSubterranean-24[3], causing multiple combining trails to merge into the same output difference and accumulate their hit counts. In section 5.5 we explore clustering in more detail.

2. $W \geq 24$**:**
   For heavy trails at or above $W = 24$, the approximate weight-based expectation drops below one hit $EN(Q) < 1$. However, fractional hits cannot occur, so any active trail that appears must register at least an integer pair of hits $N(Q) \geq 2$. The measured hits $N(Q)$ completely detach from the approximate $EN(Q)$ red line and flatten along the lower bound $N(Q) \geq 2$, providing clear visual evidence of trail clipping.

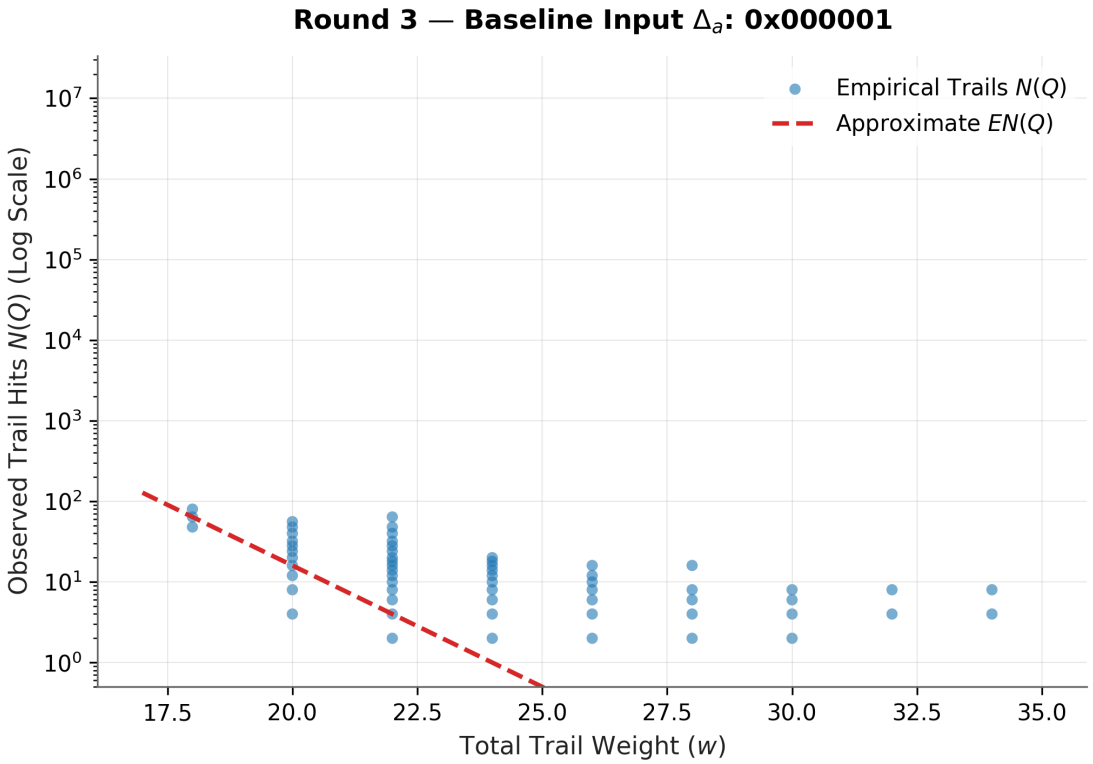


Figure 5.7: Observed $N(Q)$ and the approximate $EN(Q)$ in MiniSubterranean-24[3] over the total weights w for Δ_{a0}

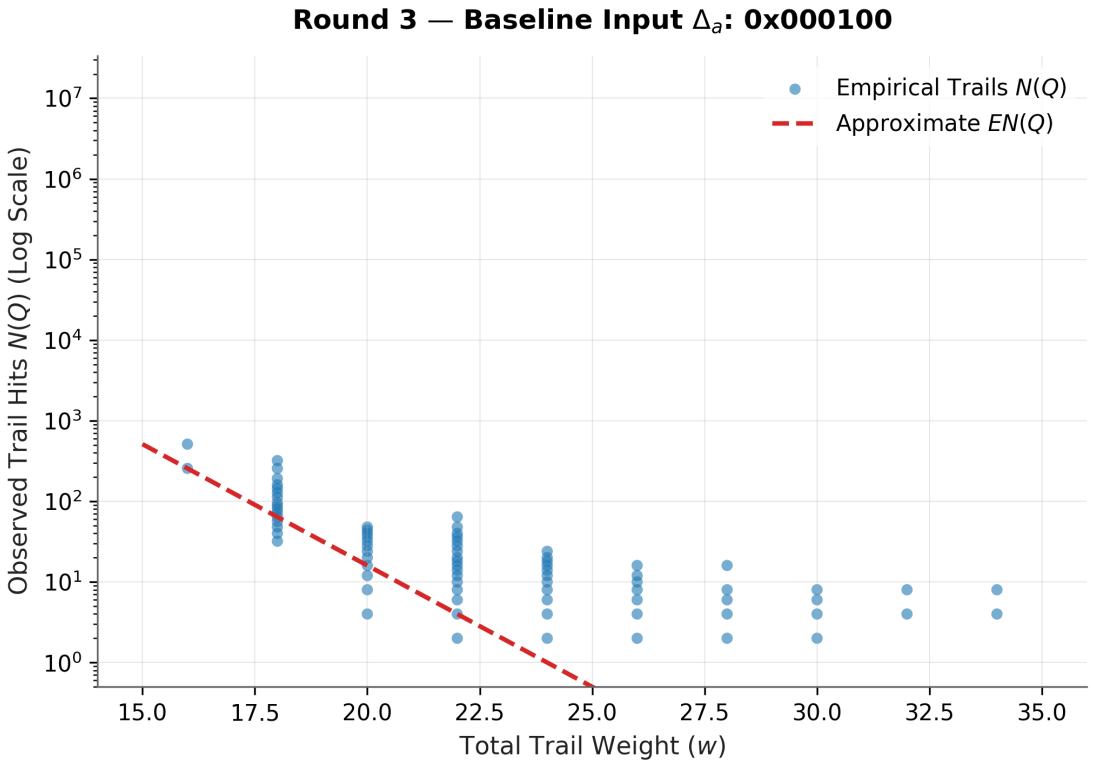


Figure 5.8: Observed $N(Q)$ and the approximate $EN(Q)$ in MiniSubterranean-24[3] over the total weights w for Δ_{a1}

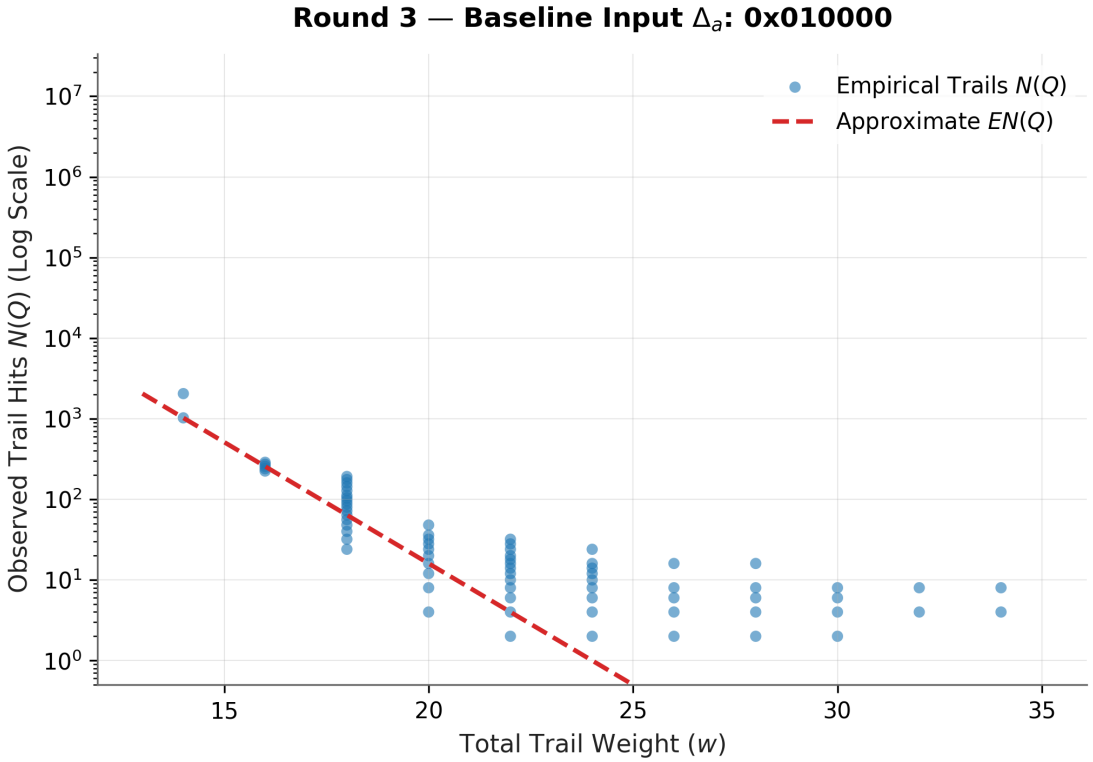


Figure 5.9: Observed $N(Q)$ and the approximate $EN(Q)$ in MiniSubterranean-24[3] over the total weights w for Δ_{a2}

Table 5.3 shows the different weight boundaries and how $\text{Ratio}_{\text{clipping}}$ reacts as the total trail weights increase. Smaller total trail weights have fewer active S-boxes, allowing a larger volume of state pairs to propagate along the trail. The weight-based approximation hits $EN(Q)$ scale with weight. This explains why trails with the lowest total weight have the highest overall hit counts.

The amplification of $\text{Ratio}_{\text{clipping}} = 2$, observed for the total weight $W = 14$ trail, is caused by adjacent S-box dependencies across the $\chi \to \lambda \to \chi \to \lambda \to \chi$ sequence. Because the non-linear layer χ operates on overlapping adjacent bits, the independent round assumption breaks down when intermediate differences align more favorably. That is why the number of state pairs $N(Q) = 2048$, exceeds the independent weight-based approximation:

$$EN(Q) = 2^{24-14} = 2^{10} = 1024 \implies \text{Ratio}_{\text{clipping}} = \frac{N(Q)}{EN(Q)} = \frac{2^{11}}{2^{10}} = 2.0\,.$$

This example shows the first minor breakdown of the independent S-box assumption. As weight increases, the independence assumption breaks down completely. For the state boundary $W = 24$, the assumption predicts $EN(Q) = 1$ hit, but the measured hits are $N(Q) = 2$. By $W = 34$, the assumption fails entirely. It predicts fractional hits of $EN(Q) = 2^{-10} \approx 0.000977$, yet we observe a trail with $N(Q) = 8$ hits, driving $\text{Ratio}_{\text{clipping}}$ to 8192.0.

boundary	γ	$N(Q)$	W	$EN(Q)$	$EDP(Q)$	$DP(Q)$	**Ratio$_{\text{clipping}}$**
$(W < 24)$	`088101`	2^{11}	14	2^{10}	2^{-14}	2^{-13}	2.0
$(W = 24)$	`b98ae7`	2^{1}	24	2^{0}	2^{-24}	2^{-23}	2.0
$(W = 28)$	`a23cf2`	2^{0}	28	2^{-4}	2^{-28}	2^{-23}	32.0
$(W = 30)$	`8d2bf9`	2^{0}	30	2^{-6}	2^{-30}	2^{-23}	128.0
$(W = 34)$	`55fe55`	2^{3}	34	2^{-10}	2^{-34}	2^{-21}	8192.0

Table 5.3: Representative differential trails in MiniSubterranean-24[3] with different total weight boundaries.

5.4.2 Clipping in MiniSubterranean-24[4] & MiniSubterranean-24[5]

In MiniSubterranean-24[4] and MiniSubterranean-24[5], total trail weights far exceed the 24-bit state, reaching maximum observed values of $W = 50$ in MiniSubterranean-24[4] and $W = 66$ in MiniSubterranean-24[5], shown in Table 5.4 and Table 5.5.

Boundary	γ	$N(Q)$	W	$EN(Q)$	$EDP(Q)$	$DP(Q)$	**Ratio$_{\text{clipping}}$**
$W = 20$	`444545`	50	20	16	2^{-20}	$2^{-18.36}$	3.125
$W = 24$	`004566`	14	24	1	2^{-24}	$2^{-20.19}$	14
$W = 38$	`996d1c`	2	38	2^{-14}	2^{-38}	2^{-23}	2^{15}
$W = 44$	`9846f4`	2	44	2^{-20}	2^{-44}	2^{-23}	2^{21}
$W = 50$	`3ce5bb`	4	50	2^{-26}	2^{-50}	2^{-22}	2^{28}

Table 5.4: Differential trails in MiniSubterranean-24[4] for baseline Δ_{a2} = `0x010000` across different weight boundaries.

Boundary	γ	$N(Q)$	W	$EN(Q)$	$EDP(Q)$	$DP(Q)$	**Ratio$_{\text{clipping}}$**
$W = 24$	`000000`	4	24	1	2^{-24}	2^{-22}	4
$W = 30$	`018801`	8	30	2^{-6}	2^{-30}	2^{-21}	2^{9}
$W = 48$	`32f802`	2	48	2^{-24}	2^{-48}	2^{-23}	2^{25}
$W = 56$	`005ba8`	2	56	2^{-32}	2^{-56}	2^{-23}	2^{33}
$W = 66$	`ee99e9`	4	66	2^{-42}	2^{-66}	2^{-22}	2^{44}

Table 5.5: Differential trails in MiniSubterranean-24[5] for baseline Δ_{a2} = `0x010000` across different weight boundaries.

The weight-based approximation decays exponentially to $EN(Q) = 2^{-26}$ hits at $W = 50$ in Table 5.4 and $EN(Q) = 2^{-42}$ hits at $W = 66$ in Table 5.5. Because base states always hit in symmetric pairs $(X, X \oplus \Delta a)$, any active empirical trail that survives must register at least an integer pair of hits $N(Q) \geq 2$. This establishes a discrete empirical floor at $DP(Q) = 2/2^{24} = 2^{-23}$. For heavy trails with $W \geq 24$, where approximate expectations drop below one pair, i.e,. $EDP(Q) \leq 2^{-24}$, this resolution limit prevents empirical probabilities from decaying any further. Whenever a high-weight trail records even a single pair of hits $N(Q) = 2$, its empirical probability is locked at 2^{-23}, resulting in severe clipping where Ratio$_{\text{clipping}}$ explodes to 2^{28} in MiniSubterranean-24[4] and 2^{44} in MiniSubterranean-24[5].

Figures 5.10 and 5.11 demonstrate this breakdown for baseline input Δ_{a1}. In both figures, the weight-based approximation $EN(Q)$ is barely visible in the frame because of the high total weights in MiniSubterranean-24[4] and is completely out of the frame in MiniSubterranean-24[5].

At the same time, we clearly see the flattening of empirical hits $N(Q)$, where all surviving trails stay below 10^2 in MiniSubterranean-24[4] and drop entirely below 10^1 in MiniSubterranean-24[5].

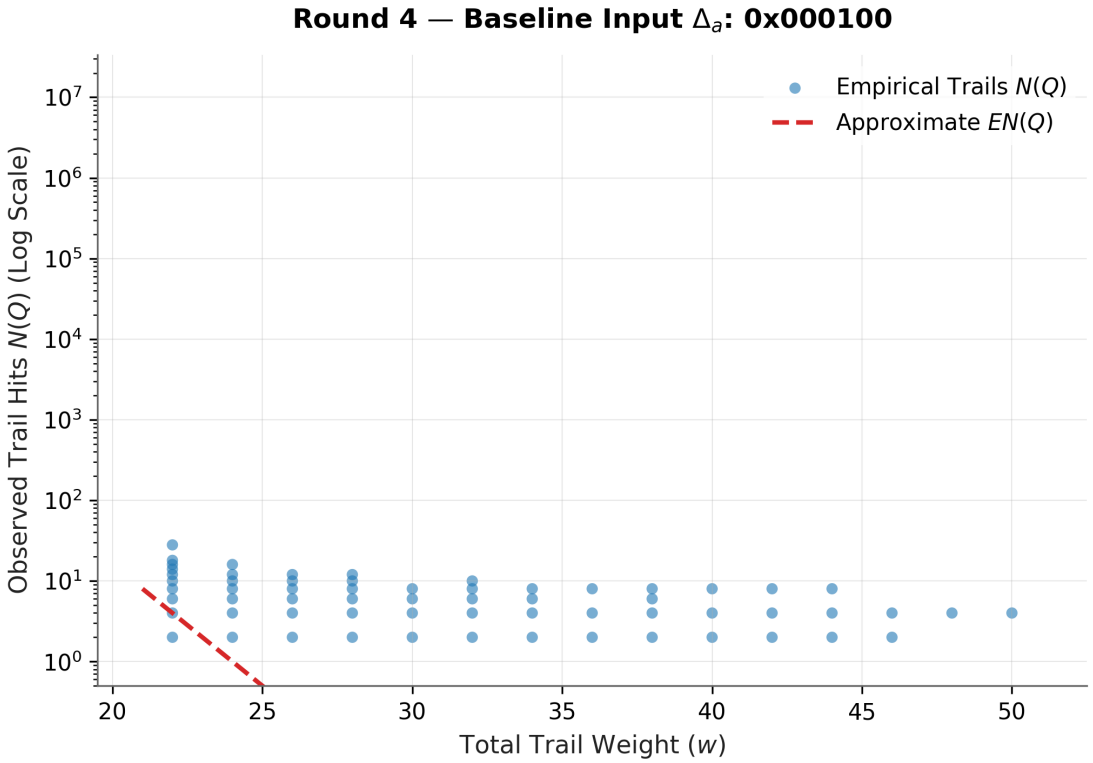


Figure 5.10: Observed $N(Q)$ and the approximate $EN(Q)$ in MiniSubterranean-24[4] over the total weights w for Δ_{a1}

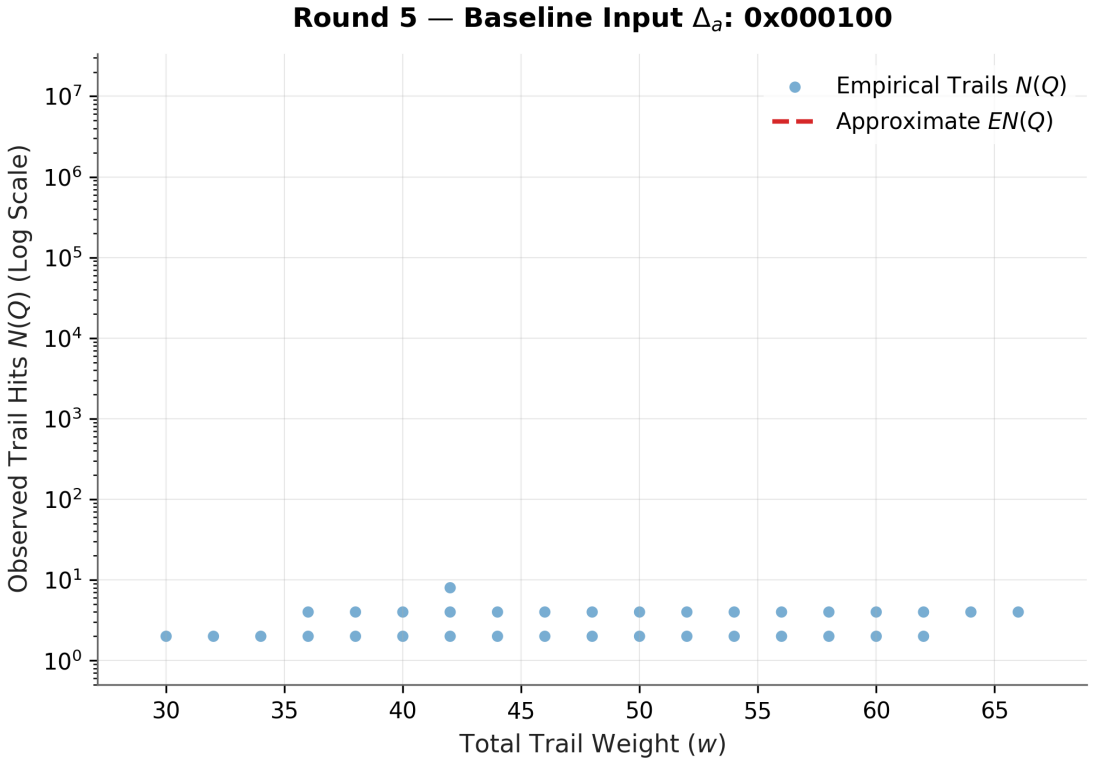


Figure 5.11: Observed $N(Q)$ and the approximate $EN(Q)$ in MiniSubterranean-24[5] over the total weights w for Δ_{a1}

5.5 Trail Clustering Analysis

The effect of differential trail clustering becomes visible starting from MiniSubterranean-24[3]. As the number of rounds increases, the differential probability $DP(b_0, \gamma)$ can no longer be accurately approximated by the probability of a single dominant trail $DP(Q_{\text{dom}})$.

5.5.1 Clustering in MiniSubterranean-24[3]

Table 5.6 summarizes the most prominent differential clusters observed across all three baseline inputs $\Delta_{a0}, \Delta_{a1}, \Delta_{a2}$ in MiniSubterranean-24[3]. Showing how multiple intermediate trails $|\Omega| > 1$ aggregate into a shared target difference γ, making the overall differential probability $\text{DP}(a, \gamma)$ higher than the weight-based approximation expectation $N(a, \gamma) > EN(Q_{\text{dom}})$.

Δ_a	γ	$\lvert\Omega\rvert$	$N(a,\gamma)$	$N(Q_{\mathbf{dom}})$	W	$EN(Q_{\mathbf{dom}})$	$\mathbf{DP}(a,\gamma)$	$\mathbf{Ratio}_{\mathbf{clustering}}$
0x000001	0x020082	2	128	64	18	64	$2^{-17.00}$	2.00
	0xa0a803	6	98	32	20	16	$2^{-17.39}$	6.13
0x000100	0x2a0888	8	976	256	18	64	$2^{-14.07}$	15.25
	0x000089	2	768	512	16	256	$2^{-14.41}$	3.00
0x010000	0x888988	2	3072	2048	14	1024	$2^{-12.41}$	3.00

Table 5.6: Differential clusters in MiniSubterranean-24[3] across all three baseline inputs, comparing empirical hits against weight-based approximation.

The cluster with the highest overall hit count occurs in the third baseline Δ_{a2} where $N(a,\gamma) = 3072$ across $|\Omega| = 2$ trails. The cluster showing the highest trail diversity, $|\Omega| = 8$, occurs in the second baseline input Δ_{a1}, accumulating a combined total of $N(a,\gamma) = 976$ hits. The intermediate trail transitions contributing to this cluster are visualized in Figure 5.12. Each of the eight trails shares the same sequence of round weights $w_1 = 2, w_2 = 8, w_3 = 8$, resulting in a total trail weight of $W = 18$.

The figure shows the origin of these colliding pairs starting from Δ_{a1} going through MiniSubterranean-24[1] to MiniSubterranean-24[3], where in the second round the trails split into eight intermediate difference states that collide into the final target of `0x2a0888`. These intermediate trails are:

- Intermediate state `0x333333` ($N = 256$ hits, $w = 8$)
- Intermediate state `0x332233` ($N = 160$ hits, $w = 8$)
- Intermediate state `0x313333` ($N = 112$ hits, $w = 8$)
- Intermediate state `0x133333` ($N = 112$ hits, $w = 8$)
- Intermediate state `0x113333` ($N = 96$ hits, $w = 8$)
- Intermediate state `0x112233` ($N = 96$ hits, $w = 8$)
- Intermediate state `0x312233` ($N = 80$ hits, $w = 8$)
- Intermediate state `0x132233` ($N = 64$ hits, $w = 8$)

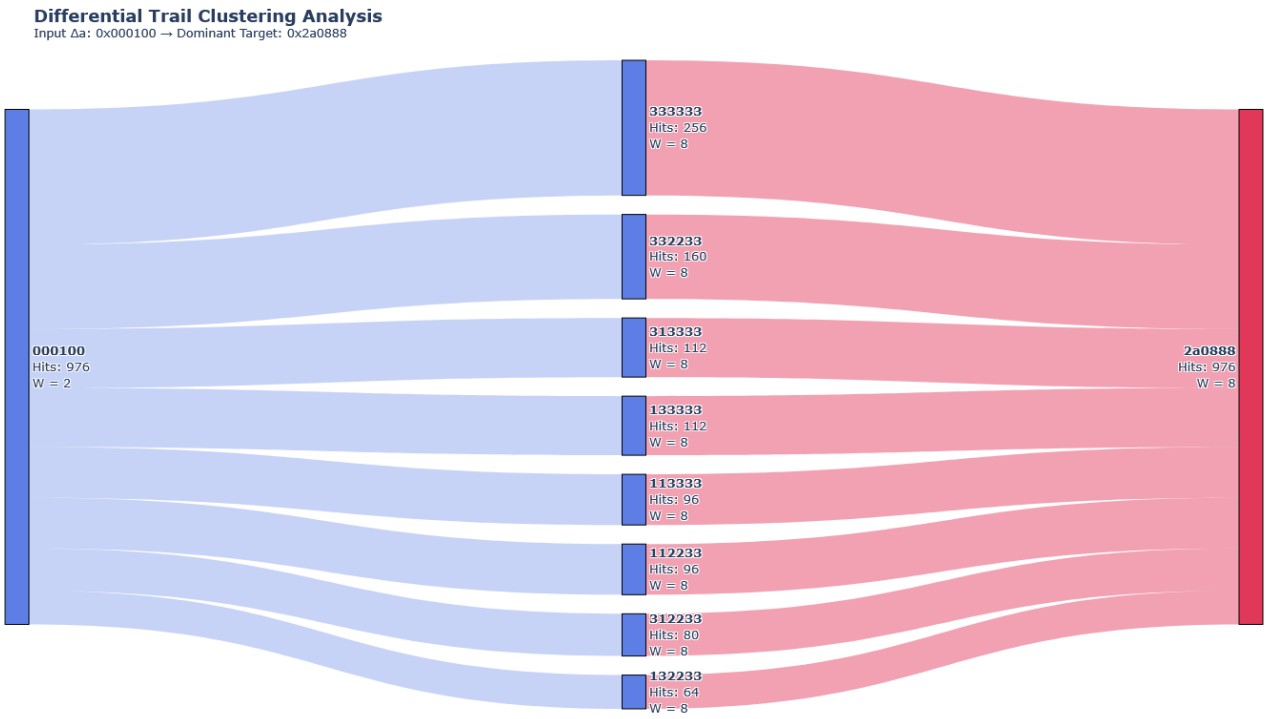


Figure 5.12: Sankey diagram showing trail clustering for baseline input Δ_{a1} = `0x000100` → `0x2a0888` in MiniSubterranean-24[3].

The clustering is caused by the quadratic non-linear layer χ, which allows active S-boxes in intermediate rounds to branch into multiple valid output differences. In the example above, the branching from baseline Δ_{a1} = `0x000100` is not random. The eight intermediate states combining in γ = `0x2a0888` (`0x333333`, `0x332233`, ..., `0x132233`) differ only in specific bit positions toggling between `1`, `2`, and `3` ($0001_2, 0010_2, 0011_2$).

5.5.2 Cluster Ratio Across Rounds

Figure 5.13 shows the evolution of the clustering ratio $\text{Ratio}_{\text{clustering}}$ across MiniSubterranean-24[1] up to MiniSubterranean-24[5], showing three distinct phases:

1. **MiniSubterranean-24[1] & MiniSubterranean-24[2]:**
   In the first two rounds, the ratio remains at $\text{Ratio}_{\text{clustering}} = 1.0$. The transitions follow a single dominant path without branching ($|\Omega| = 1$), matching the approximate weight model.

2. **MiniSubterranean-24[3]**
   As shown in Table 5.6, the non-linear layer χ begins branching into multiple trails, raising the clustering ratio up to 15.25 for Δ_{a1}.

3. **MiniSubterranean-24[4] & MiniSubterranean-24[5]:**

In MiniSubterranean-24[4] and MiniSubterranean-24[5], due to all the mixing and shuffling of the bits, more and more S-boxes are activated, forcing the total trail weights beyond the 24-bit state limit $W \geq 24$. This causes the approximation $EN(Q_{\text{dom}})$ to drop to fractional values, e.g., $EN(Q_{\text{dom}}) = 2^{-26}$ in MiniSubterranean-24[4] and $EN(Q_{\text{dom}}) = 2^{-42}$ in MiniSubterranean-24[5].

Because the empirical clusters maintain discrete counts of $N(a, \gamma) \geq 2$ while aggregating branching trails become fractions, the $\text{Ratio}_{\text{clustering}}$ escalates exponentially, reaching orders of 10^7 in MiniSubterranean-24[4] up to 10^{13} in MiniSubterranean-24[5].

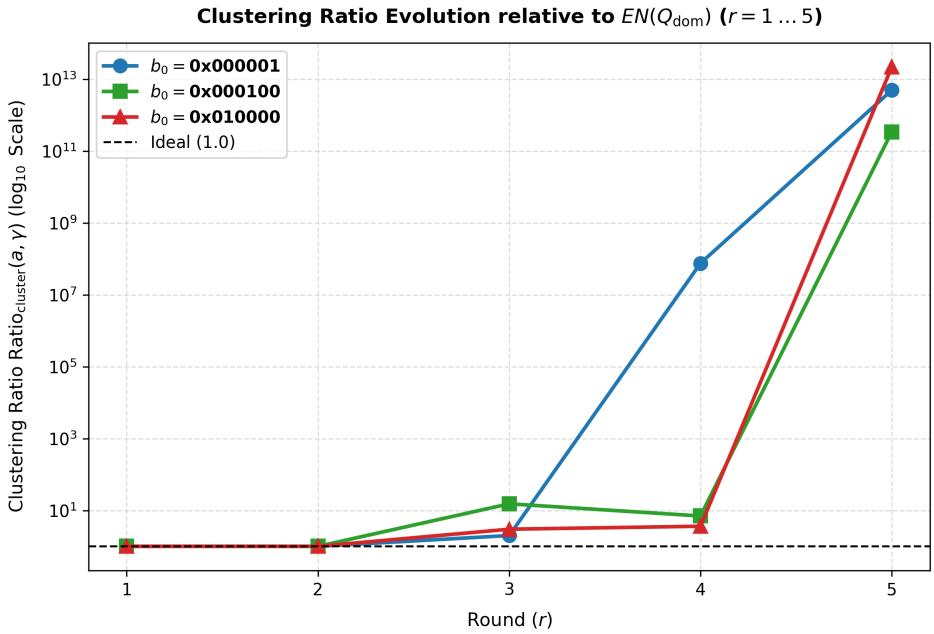


Figure 5.13: Clustering ratio relative to weight-based approximation $EN(Q_{\text{dom}})$ across rounds $r = 1 \ldots 5$ for all three baseline inputs.

5.6 Spurious Trails Analysis

Spurious trails play a crucial role in the security analysis of the Farfalle accumulator under the single dominant trail assumption. Since this assumption breaks down in MiniSubterranean-24[3] due to both trail clipping and trail clustering, this section investigates the resulting consequences for collision security.

5.6.1 Spurious trail ratio

To evaluate how the dominant trail prefix weight-based approximation deviates across the three baselines, Figures 5.14, 5.15, and 5.16 show the evolution of the empirical collision probability $CP(b_0)_{\text{exact}}$ against the approximate dominant prefix probability $CP(b_0)_{\text{prefix}}$ across MiniSubterranean-24[1] through MiniSubterranean-24[5] for the three baseline inputs $\Delta_{a0}, \Delta_{a1}, \Delta_{a2}$.

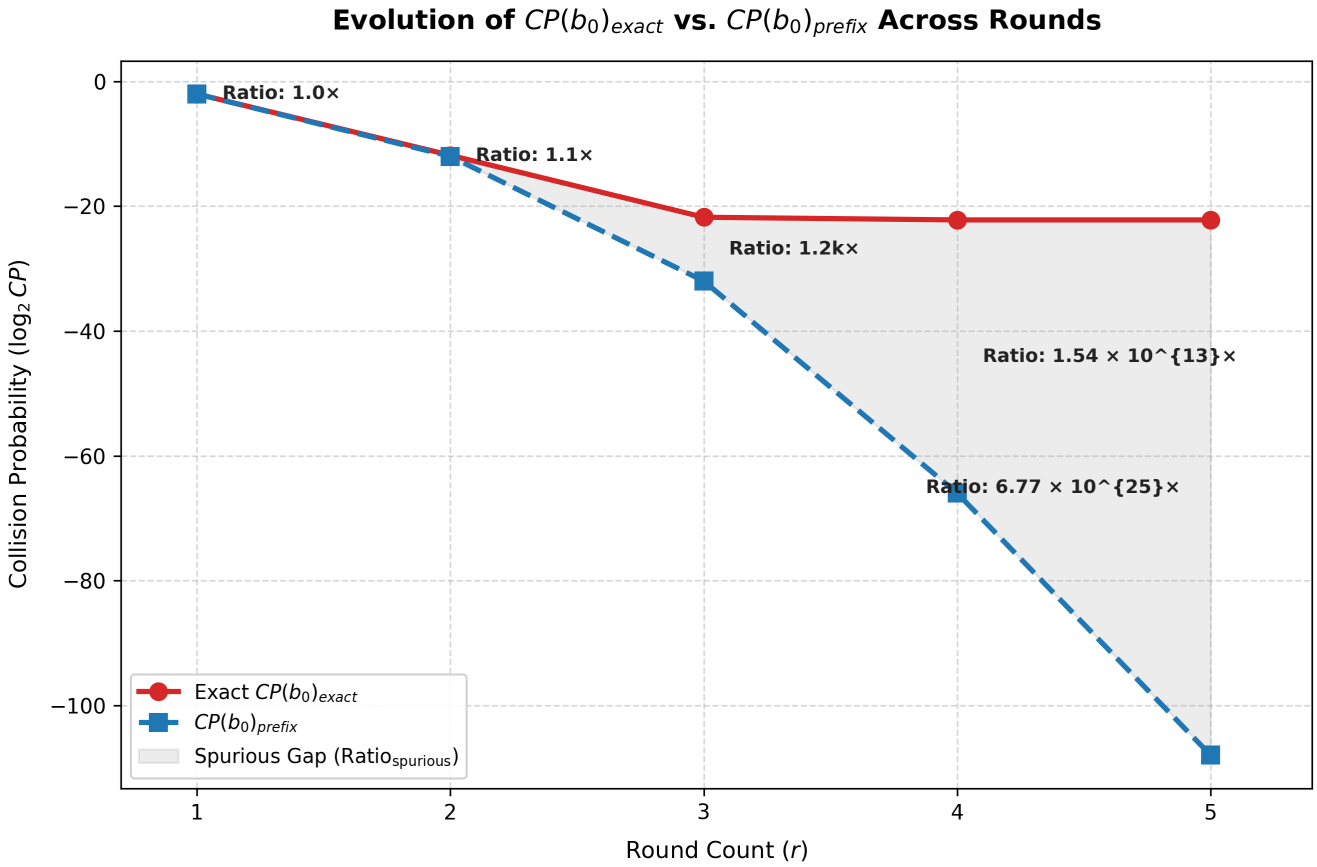


Figure 5.14: $\text{Ratio}_{\text{spurious}}(\Delta_{a0})$ across rounds.

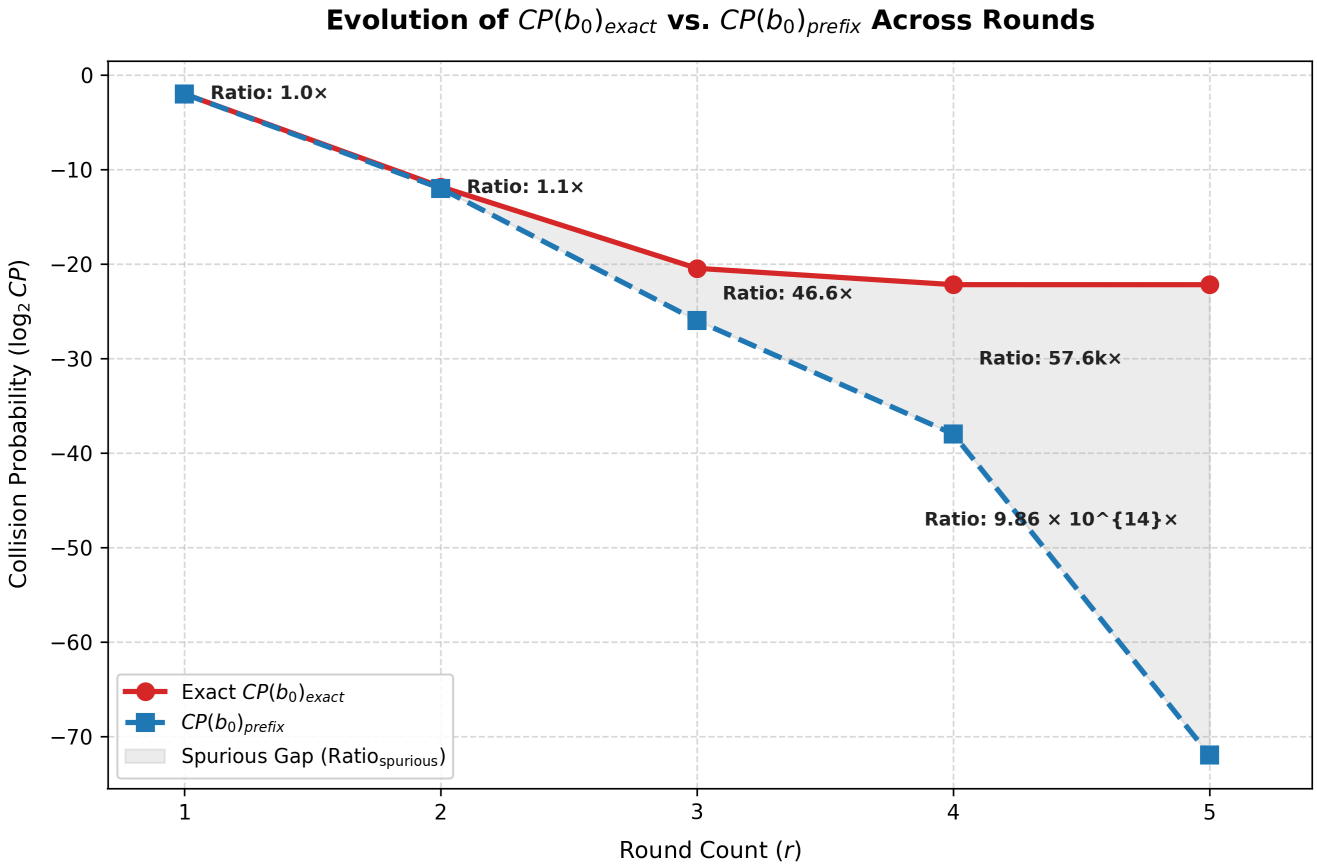


Figure 5.15: $\text{Ratio}_{\text{spurious}}(\Delta_{\text{a1}})$ across rounds.

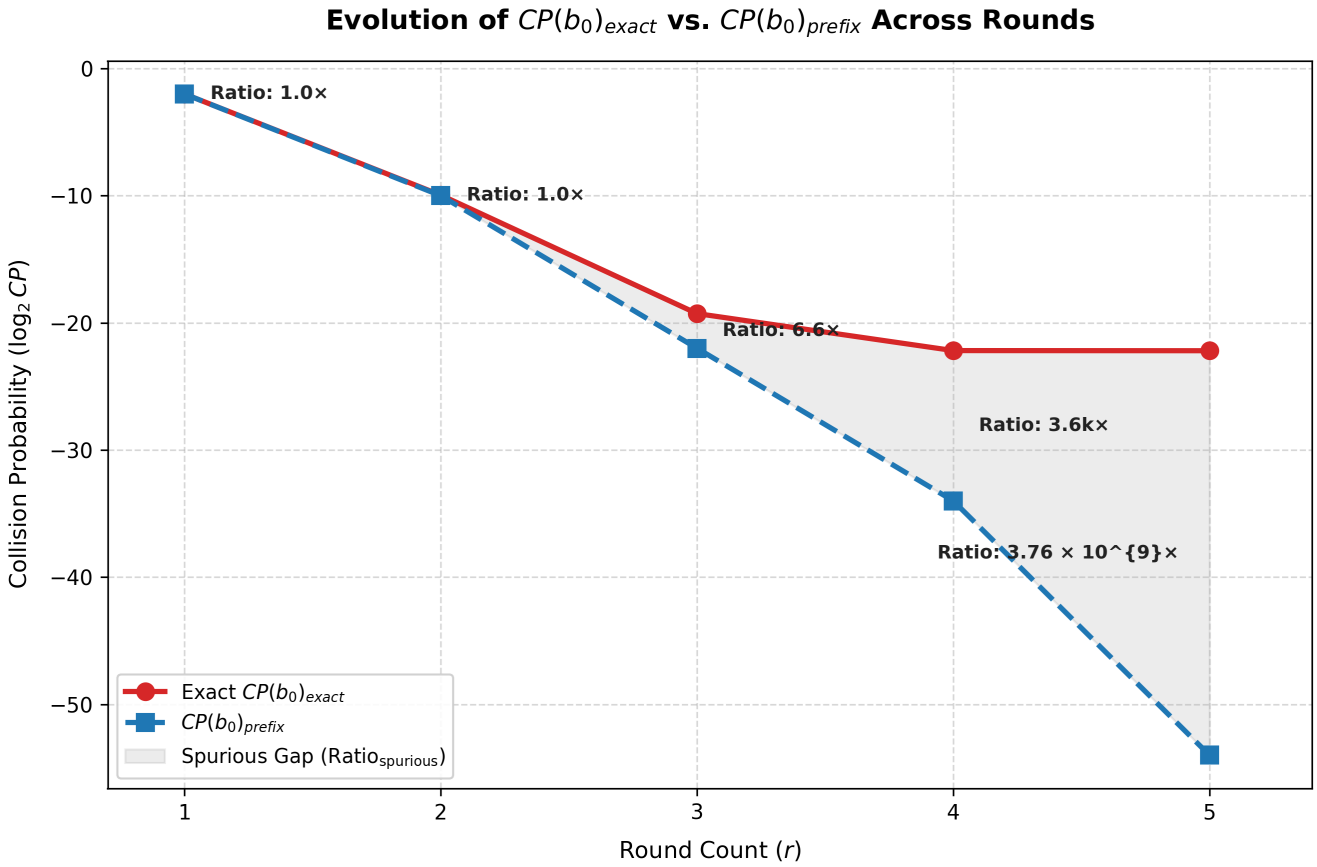


Figure 5.16: $\text{Ratio}_{\text{spurious}}(\Delta_{a2})$ across rounds.

The graphs all seem to show the same pattern for all three baseline inputs. There is a close match between weight-based approximation and empirical values for $r \in \{1, 2\}$, begins to diverge noticeably at $r = 3$, and diverges drastically in rounds $r = 4$ and $r = 5$. Observing each round separately:

1. **MiniSubterranean-24[1]:**
   In the first round, the ratio equals $\text{Ratio}_{\text{spurious}} = 1.0\times$ across all baseline inputs. Because only the non-linear layer χ is applied with a single active S-box ($w_1 = 2$), the 2^{24} state pairs distribute evenly across 4 trails with $N(Q) = 2^{22}$ hits each. Both the exact collision probability and the approximate dominant trail prefix model evaluate to:

   $$CP_{\text{exact}} = \sum_{\gamma=1}^{4} \left(\frac{2^{22}}{2^{24}}\right)^2 = 4 \times 2^{-4} = 0.25, \quad CP_{\text{prefix}} = 2^{-w_1} = 2^{-2} = 0.25\,.$$

   The dominant trail prefix perfectly predicts the empirical collision probability of the primitive.

2. **MiniSubterranean-24[2]:**
   In $r = 2$, the ratio remains minimal at $\text{Ratio}_{\text{spurious}} \approx 1.1\times$ across all three baselines.

Although the linear layer λ activates additional S-boxes, the effect is still insufficient to cause significant trail splitting. Conforming that state pairs remain concentrated along the dominant trail prefix, allowing the dominant prefix approximation to match the empirical collision probability.

3. **MiniSubterranean-24[3]:**
In $r = 3$, a large jump occurs where the empirical and the approximate collisions no longer match. The ratio jumps to 1.2k× (1,206.5×) for Δ_{a0}, 46.6× for Δ_{a1}, and 6.6× for Δ_{a2}. This difference happens because branching in χ produces over 6.5 million unique trails. The dominant trail prefix alone can no longer predict the Farfalle accumulator collisions, as the collision probability becomes driven by the aggregate sum of these millions of weaker, secondary (spurious) trails.

4. **MiniSubterranean-24[4] and MiniSubterranean-24[5]:**
In $r = 4$ and $r = 5$, the spurious ratio explodes exponentially, reaching orders of $1.54{\times}10^{13}\times$ in MiniSubterranean-24[4] and $6.77 \times 10^{25}\times$ in MiniSubterranean-24[5] for baseline Δa_0, as illustrated in Figure 5.14 .

   This extreme amplification is caused by two factors happening at the same time:

   - Starting from MiniSubterranean-24[4], CP_{exact} flattens out, reaching an empirical plateau of $CP_{\text{exact}} \approx 2^{-22.19}$ due to the pseudorandom dispersion of state pairs across small even hit counts $N \in \{2, 4, 6, \dots\}$.
   - Meanwhile, the approximation CP_{prefix} continues to decay exponentially toward zero. Because the prefix collision formula doubles all intermediate round weights in the exponent, $CP_{\text{prefix}} = 2^{-(2\sum_{i=0}^{r-2} w_i + w_{r-1})}$, evaluating the dominant trail prefix specifically for the baseline Δa_0 example yields:
     - In $r = 4$ with individual weights $(w_0 = 2, w_1 = 16, w_2 = 10, w_3 = 10)$ and total weight $W = 38$:

       $$2(2 + 16 + 10) + 10 = 2(28) + 10 = 66 \implies CP_{\text{prefix}} = 2^{-66}.$$

       Dividing the empirical plateau $CP_{\text{exact}} \approx 2^{-22.19}$ by $CP_{\text{prefix}} = 2^{-66}$ directly accounts for the spurious ratio:

       $$\text{Ratio}_{\text{spurious}}(\Delta a_0) = \frac{2^{-22.1917}}{2^{-66}} = 2^{43.8083} \approx 1.5403 \times 10^{13}\,.$$

     - In $r = 5$ with individual weights $(w_0 = 2, w_1 = 16, w_2 = 14, w_3 = 14, w_4 = 16)$ and total weight $W = 62$:

       $$2(2 + 16 + 14 + 14) + 16 = 2(46) + 16 = 108 \implies CP_{\text{prefix}} = 2^{-108}.$$

       Dividing the empirical plateau $CP_{\text{exact}} \approx 2^{-22.19}$ by $CP_{\text{prefix}} = 2^{-108}$ yields:

       $$\text{Ratio}_{\text{spurious}}(\Delta a_0) = \frac{2^{-22.1919}}{2^{-108}} = 2^{85.8081} \approx 6.7737 \times 10^{25}\,.$$

   Dividing the stable empirical plateau $CP_{\text{exact}} \approx 2^{-22.19}$ by these small probabilities 2^{-66} and 2^{-108} causes $\text{Ratio}_{\text{spurious}}$ to surge.

The graphs also reveal a notable difference in $\text{Ratio}_{\text{spurious}}$ across the three baseline inputs. Because the prefix approximation is dictated by the round weights of the dominant trail, baseline differences directly affect the denominator CP_{prefix}. In MiniSubterranean-24[4], while the maximum observed trail weight across all three baselines reaches 50, the minimum trail weight for Δ_{a0} is $W_{\text{min}} = 24$, compared to 22 for Δa_1 and 20 for Δa_2. This higher minimum weight significantly suppresses $CP_{\text{prefix}}(\Delta_{\text{a0}})$, driving the denominator down and causing the pool of higher-weight spurious trails to dominate the exact collision probability CP_{exact}, which inflates $\text{Ratio}_{\text{spurious}}$. This pattern also holds in MiniSubterranean-24[5], where the lower minimum weights of Δ_{a1} and Δ_{a2} preserve a relatively larger share for their dominant prefixes, resulting in smaller spurious gaps across both rounds.

5.7 Weight distribution

To evaluate how the linear and non-linear operations spread active bits across the state space, we trace the average per-round differential weight $\overline{w}_r$ across MiniSubterranean-24[r].

Figure 5.17 illustrates the progression of average per-round weight $\overline{w}_r$ and the standard deviation across rounds for all three baseline inputs.

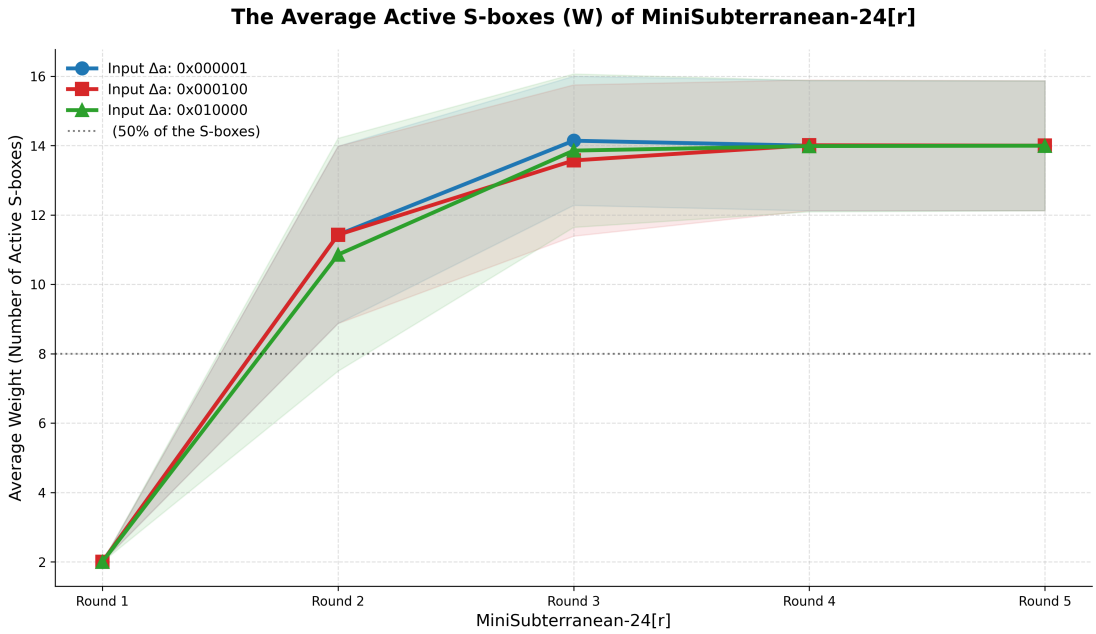


Figure 5.17: The average active S-boxes $\overline{w}_r$ with standard deviation.

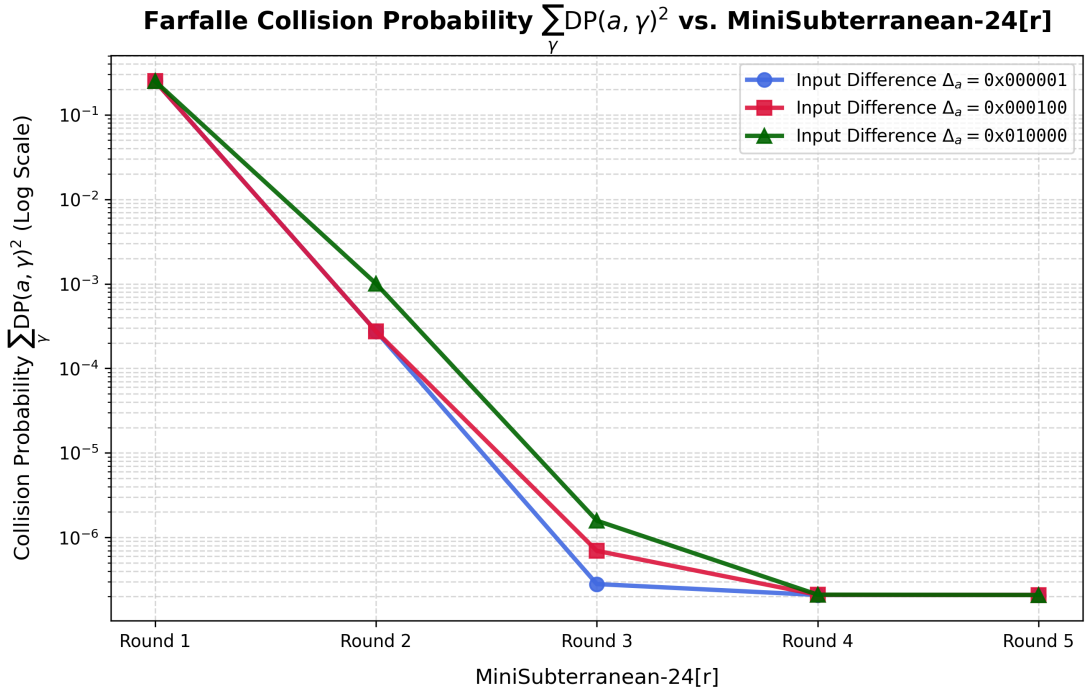
All baseline inputs start with a single active S-box ($w_1 = 2$) in MiniSubterranean-24[1]. The linear layer mixes and shuffles the bits in MiniSubterranean-24[2], causing $\overline{w}_2 \approx 11$. By MiniSubterranean-24[3], the per-round weight flattens out at $\overline{w}_r \approx 14$. This indicates that from MiniSubterranean-24[3] onward, an average of 7 out of 8 S-box columns (87.5%) are active in every round, causing total trail weights W to increase linearly by approximately 14 per additional round. This explains

why the maximum total trail weights reached $W = 50$ in MiniSubterranean-24[4] and $W = 66$ in MiniSubterranean-24[5].

5.8 Farfalle Accumulator Collision Probability

In the Farfalle construction, resistance against two-block collision attacks depends on the accumulator collision probability $CP_{\text{exact}}(\Delta_a) = \sum_\gamma \text{DP}(\Delta_a, \gamma)^2$. To maximize security against collision attacks, $CP_{\text{exact}}(\Delta_a)$ must be minimized. As the rounds r increase, the transformation mixes and shuffles active bits across the entire state space, driving $CP_{\text{exact}}(\Delta_a)$ down towards a minimum.

Figure 5.18 plots the progression of $CP_{\text{exact}}(\Delta_a)$ across MiniSubterranean-24[1] through MiniSubterranean-24[5] for all three baseline inputs $\Delta_{a0}, \Delta_{a1}, \Delta_{a2}$.



	Round 1	Round 2	Round 3	Round 4	Round 5
0x000001	$2^{-2.00}$	$2^{-11.83}$	$2^{-21.76}$	$2^{-22.19}$	$2^{-22.19}$
0x000100	$2^{-2.00}$	$2^{-11.83}$	$2^{-20.46}$	$2^{-22.19}$	$2^{-22.19}$
0x010000	$2^{-2.00}$	$2^{-9.95}$	$2^{-19.27}$	$2^{-22.18}$	$2^{-22.19}$

Figure 5.18: $CP_{\text{exact}}(\Delta_a)$ collision probability across MiniSubterranean-24[r]

In MiniSubterranean-24[1], the collision probability sits at its maximum value of $CP_{\text{exact}}(\Delta_a) = 0.25 = 2^{-2.00}$. This is because only a single S-box is active with weight $w_1 = 2$, distributing the state pairs evenly across 4 valid output transitions.

As linear operations λ are executed from MiniSubterranean-24[2] onward, bit mixing and shuffling

activate multiple S-boxes, reducing collision probability down to the range of $2^{-9.95}$ to $2^{-11.83}$.

In MiniSubterranean-24[3], where trail clustering, trail clipping, and spurious trails first emerge, $CP_{\text{exact}}(\Delta_a)$ drops sharply to $2^{-21.76}$ for Δ_{a0}, $2^{-20.46}$ for Δ_{a1}, and $2^{-19.27}$ for Δ_{a2}. This steep decline is driven by the expansion of active differential trails, jumping from roughly 74,000 unique trails in MiniSubterranean-24[2] to over 6.5 million trails in MiniSubterranean-24[3]. The total state pairs scatter across millions of distinct trails, making individual transition probabilities fragment, pulling $CP_{\text{exact}}(\Delta_a)$ down significantly.

By MiniSubterranean-24[4] and MiniSubterranean-24[5], $CP_{\text{exact}}(\Delta_a)$ flattens out across all three baseline inputs, reaching an empirical plateau of $CP_{\text{exact}}(\Delta_a) \approx 2^{-22.19}$. The absolute minimum collision probability across the 24-bit state space is 2^{-23}, which would occur only if every single active output difference received exactly $N(a, \gamma) = 2$ hits. However, in rounds 4 and 5, the transformation behaves like a pseudorandom permutation, dispersing differences across discrete hit counts such as $\{2, 4, 6, 8\}$. Because calculating the collision probability squares each hit count $\sum N(a, \gamma)^2$, the presence of these higher count pulls the sum up slightly, stabilizing $CP_{\text{exact}}(\Delta_a)$ at this empirical plateau of $2^{-22.19}$.

Security against two-block collision attacks
From a security perspective, these results show that for the three baselines tested with a single-active S-box in MiniSubterranean-24, the two-block accumulator collision probability reaches its empirical plateau of $CP_{\text{exact}}(\Delta a) \approx 2^{-22.19}$ by round 4. For these evaluated inputs, increasing the round count to MiniSubterranean-24[5] yields no further reduction in collision probability. These observations are specific to minimal-weight single-bit inputs within our 24-bit toy model and do not constitute a formal proof for all possible multi-bit difference pairs or the full-scale 384-bit primitive. It only demonstrates that four rounds are sufficient for our tested baselines to reach the empirical collision floor of a pseudorandom permutation.

Chapter 6

Related Work

This thesis builds directly upon the parallel permutation framework of Farfalle [1] and the differential trail analysis methodology established in Xoodoo [2]. Using these two methods to analysis the round transformation of Subterranean 3.0-P.

6.1 The Farfalle Construction and Collision Security

The Farfalle construction [1] is a parallel permutation-based construction for building pseudorandom functions (PRFs) with arbitrary-length inputs and extendable outputs. In Farfalle, variable-length input sequences are compressed by applying the permutation p_c to rolling-masked input blocks and XORing their outputs into a central accumulator. Resistance against two-block differential collision attacks was formulated under the dominant trail prefix hypothesis, yielding the approximation:

$$CP(\Delta_a) \approx CP_{\text{prefix}}(\Delta_a) = 2^{-\left(2\sum_{i=1}^{r-1} w_i + w_r\right)}.$$

where intermediate round weights w_i for $i < r$ are doubled because two independent branches must simultaneously follow the trail, while the final round weight w_r is not double because it accounts for the output collision. This hypothesis assumes that as the rounds increase differential transitions are overwhelmingly dominated by a single optimal trail Q_{dom}, ignoring the collective contribution of secondary (spurious) trails.

6.2 Xoodoo and the Dominant Trail Assumption

To instantiate the Farfalle construction efficiently on modern processors, the paper *The Design of Xoodoo and Xoofff* [2] introduced the 384-bit permutation Xoodoo and its associated deck function Xoofff. Xoodoo introduced two key structural features:

1. **3-Bit Non-Linear Layer (χ_3):**
   By applying the non-linear χ transformation to independent 3-bit columns, compatible input and output differences form compact affine spaces of dimension 2. This property significantly simplifies tree-search algorithms used to prove lower bounds on trail weights.

2. **Weak Alignment:**
   The permutation makes use of two plane-shifting steps, ρ_{west} and ρ_{east}, before and after the mixing layer θ to continuously disrupt column alignment across rounds.

In the security analysis of Xoofff, resistance against two-block Farfalle accumulator collisions was modeled under the hypothesis that as the rounds increase differentials are dominated by a single optimal differential trail prefix Q_{dom}, arguing that weak alignment renders trail clustering negligible.

6.3 Positioning of This Thesis

The Subterranean 3.0-P round transformation inherits the 3-bit χ non-linear layer and dual-shift dispersion $(\rho_{\text{west}}, \rho_{\text{east}})$ introduced in Xoodoo. In this thesis, we investigate whether the dominant trail prefix hypothesis—relied upon by Farfalle and Xoofff to estimate two-block accumulator collision resistance also holds for the Subterranean 3.0-P transformation.

To evaluate this, we constructed a 24-bit toy model MiniSubterranean-24 to perform an empirical census of all differential trails across the state space, computing both the exact accumulator collision probability and the approximate collision probability of the dominant trail prefix.

Our analysis evaluates the boundaries of the dominant trail model, demonstrating where and why empirical collision probabilities deviate from single-prefix predictions due to the combined effects of trail clipping, differential clustering, and spurious trail accumulation.

Chapter 7

Conclusions

In this thesis, we evaluated trail clipping, differential clustering, and the role of spurious trails in the compression transformation of the Farfalle construction based on Subterranean 3.0-P. To assess these phenomena, we performed an exhaustive empirical cryptanalysis on MiniSubterranean-24, a 24-bit toy model preserving the structure of the cipher, across five transformation rounds ($r \in \{1, \ldots, 5\}$).

To answer our primary research question: our findings demonstrate that while the dominant-trail weight approximation is highly accurate at low round counts ($r \in \{1, 2\}$), it breaks down and severely underestimates the empirical collision probability as the number of rounds increases. This divergence occurs because the single dominant trail accounts for an increasingly negligible share of the total differential probability mass, while secondary and alternative trails become collectively dominant.

Our investigation of the three core metrics explains the mechanics behind this divergence:

- **Trail Clipping:**
  In the first two rounds, empirical measurements matched the weight-based approximations closely with no evidence of clipping. From round 3 onwards, total trail weights exceeded the state width $W \geq 24$, where weight-based models approximate fractional hit counts of $EN(Q) < 1$. However, because states hit in symmetric pairs $X, X \oplus \Delta a$, any surviving empirical trail must record at least an integer pair of hits $N(Q) \geq 2$, establishing a discrete empirical floor at $DP(Q) = 2^{-23}$. In rounds 4 and 5, as total weights escalated to $W = 50$ and $W = 66$, making the approximations decayed to negligible fractions of $EN(Q) = 2^{-26}$ and 2^{-42}, causing the clipping ratio to explode.

- **Differential Clustering:**
  Clustering became prominently visible from round 3 onwards. Driven by non-linear branching in the χ layer, the number of observed unique differential trails surges from approximately 74 thousand in the second round to over 6.5 million in third round. These millions of distinct trails converge toward identical output differences γ, making the aggregated probability substantially larger than the probability of the dominant trail alone. As a result, the dominant trail's relative share shrinks drastically, demonstrating that single-trail heuristics, relying exclusively on the dominant trail's differential probability to assess security of the cipher, fails to capture majority of the differential probability of the transformation as the rounds increase.

- **Spurious Trails:**
  The dominant-prefix approximation increasingly fails to account for secondary trails as the number of rounds grows. Although individual spurious trails are individually weaker and possess higher weights, their vast collective majority dominates the exact collision probability CP_{exact}. This divergence is most evident in rounds 4 and 5, where the prefix probability continues to decay exponentially toward zero ($CP_{\text{prefix}} = 2^{-66}$ and 2^{-108}), while the empirical collision probability stabilizes.

The evaluation of the three single-active S-box baseline differences $\Delta_{\text{a0}}, \Delta_{\text{a1}}$ and Δ_{a2} (exploiting horizontal shift-invariance), showed that the two-block accumulator collision probability reaches an empirical plateau of $CP_{\text{exact}} \approx 2^{-22.19}$ by round 4, with no further reduction observed at round 5. This plateau demonstrates that MiniSubterranean-24 behaves as a pseudorandom permutation, where output difference pairs disperse across a distribution of small even hit counts ($N \in \{2, 4, 6, 8, \dots\}$). If every active output had $N = 2$ hits, the collision probability would reach the theoretical floor of 2^{-23}. However, because the cipher behaves pseudorandomly with varying hit counts, $N \in \{2, 4, 6, \dots\}$, the sum of squared probabilities settles slightly higher at $2^{-22.19}$. Thus, for the tested baselines in our 24-bit toy model, four rounds are sufficient to reach the empirical collision floor of a pseudorandom permutation.

Bibliography

[1] Guido Bertoni, Joan Daemen, Seth Hoffert, Michaël Peeters, Gilles Van Assche, and Ronny Van Keer. Farfalle: parallel permutation-based cryptography. *Cryptology ePrint Archive*, 2016.

[2] Joan Daemen, Seth Hoffert, Gilles Van Assche, and Ronny Van Keer. The design of xoodoo and xoofff. *IACR Transactions on Symmetric Cryptology*, pages 1–38, 2018.