IACR Transactions on Symmetric Cryptology
ISSN 2519-173X, Vol. 2025, No. 1, pp. 515–541. DOI:10.46586/tosc.v2025.i1.515-541

Extending the Quasidifferential Framework: From Fixed-Key to Expected Differential Probability

Christina Boura[1], Patrick Derbez[2] and Baptiste Germon[2]

[1] IRIF, Université Paris Cité, Paris, France
christina.boura@irif.fr
[2] Univ Rennes, Inria, CNRS, IRISA, Rennes, France
patrick.derbez@inria.fr,baptiste.germon@inria.fr

Abstract. Beyne and Rijmen proposed in 2022 a systematic and generic framework to study the fixed-key probability of differential characteristics. One of the main challenges for implementing this framework is the ability to efficiently handle very large quasidifferential transition matrices (QDTMs) for big (e.g. 8-bit) S-boxes. Our first contribution is a new MILP model capable of efficiently representing such matrices, by exploiting the inherent block structure of these objects. We then propose two extensions to the original framework. First, we demonstrate how to adapt the framework to the related-key setting. Next, we present a novel approach to compute the average expected probability of a differential characteristic that takes the key schedule into account. This method, applicable to both linear and non-linear key schedules, works in both the single-key and related-key settings. Furthermore, it provides a faster way to verify the validity of characteristics compared to computing the fixed-key probability. Using these extensions and our MILP model, we analyze various (related-key) differential characteristics from the literature. First, we prove the validity of several optimal related-key differential characteristics of AES. Next, we show that this approach permits to obtain more precise results than methods relying on key constraints for SKINNY. Finally, we examine the validity of a differential distinguisher used in two differential meet-in-the-middle attacks on SKINNY-128, demonstrating that its probability is significantly higher than initially estimated.

Keywords: differential cryptanalysis · quasidifferential trails · MILP · related keys · SKINNY · AES

1 Introduction

Differential cryptanalysis is one of the most important and powerful cryptanalysis techniques against block ciphers. This technique, introduced by Biham and Shamir in 1990 [BS91] exploits input differences that propagate through the cipher to output differences with high probability. Computing the probability of a differential is a very hard problem. As the majority of the symmetric primitives are built by iterating a relatively simple function, called the round function, a classical approach for estimating the probability of a differential is to work with the so-called *differential characteristics*, that are sequences of one-round differentials. As the probability of a differential is the sum of the probabilities of all differential characteristics that it comprises, the probability of a differential can be lower bounded by the probability of any of its characteristics.

Computing the probability of a differential characteristic, though less hard than computing the probability of the whole differential, is still a very hard problem. For this reason, this probability is usually estimated by multiplying the probabilities of the one-round differentials composing the characteristic, by using the assumption that

Received: 2024-11-22 Accepted: 2025-01-23 Published: 2025-03-07

these one-round differentials are independent. In 1991, Lai, Massey and Murphy showed that this independence hypothesis holds for the so-called *Markov ciphers* and gives for these constructions the correct value for the *keyed-average probability* of the differential characteristic [LMM91].

When mounting an attack against a concrete instance of a cipher, the quantity that becomes important is the *fixed-key* probability of a differential or a differential characteristic. This is of course very hard to estimate and in practice, to overcome this problem, what has been used for years is the *stochastic equivalence hypothesis* [LMM91] that states that the probability of the differential characteristic for any specific key is close to the average probability of the characteristic across all keys.

The problem with both the independence and the stochastic equivalence hypothesis is that they are known to fail for the majority of ciphers. For instance, Knudsen demonstrated that the probability of the differential used to attack the DES in [BS91] varies significantly depending on the key value [Knu93]. Similarly, Ankele and Kölbl [AK19], Heys [Hey20] and Peyrin and Tan [PT22] provide several examples where the stochastic equivalence hypothesis does not hold. The differential characteristics of the AES are also known to behave differently depending on the key value. Notably, its designers showed that all 2-round characteristics of the AES, and most of its 4-round characteristics, are *plateau characteristics*. This means that their probability can only take two values: 0 for almost all keys and a fixed value p for the remaining ones [DR07].

The first complete and powerful framework for studying the fixed-key probability in differential cryptanalysis was provided in 2022 by Beyne and Rijmen [BR22]. In this work, the authors introduced the notions of *quasidifferential transition matrices* and *quasidifferential trails*. A quasidifferential trail can be seen as a sequence of mask-difference pairs that allows the propagation of probabilistic linear relations on the values satisfying a differential characteristic. One of the central results in [BR22] is a formula involving quasidifferential trails that can be used to exactly compute the probability of a differential characteristic, without relying on any underlying hypotheses.

Some researchers, on the other hand, took a different approach to study the fixed-key probability of differential characteristics. For example, Peyrin and Tan [PT22], whose article was published slightly after the quasidifferential framework [BR22], use a more *ad hoc* approach that consists in searching for specific key constraints for a given characteristic and then to estimate the impact of each detected constraint on the probability. They introduce several types of constraints in their work, namely (higher-order) linear and nonlinear constraints. By designing a tool to detect these different constraints, they show that most of the characteristics published in the literature for the lightweight ciphers SKINNY and GIFT are valid for only a small proportion of the key space, and several among them are not valid for any key.

At first glance, searching for key constraints may seem more intuitive, as it potentially allows for better understanding of these constraints. However, this method has several limitations. First, identifying such constraints can be a challenging and complex task. Moreover, there is no guarantee that all constraints can be identified in this manner, potentially leading to an erroneous partition of the key space for a given characteristic.

On the other hand, the quasidifferential framework is complete and allows for a precise partitioning of this key space. Its main downside is that it can be computationally intensive to calculate all quasidifferential trails for a given differential. This task is further complicated by the size of the quasidifferential transition matrix, which is significantly larger than a classical differential distribution table. These are the main reasons why approaches based on identifying key constraints are often preferred.

However, we believe that the right approach to tackling this difficult problem is not to bypass the quasidifferential framework but, on the contrary, to focus on efficiently implementing and extending it. This is what we propose in this article.

Our contributions In this work, we propose several extensions to the framework of Beyne and Rijmen [BR22]. First, while this framework included a basic Satisfiability Modulo Theories (SMT) tool for searching quasidifferential trails, one major challenge in its implementation is handling large S-boxes, such as 8-bit S-boxes. Specifically, an 8-bit S-box requires a quasidifferential transition matrix of size $2^{16} \times 2^{16}$, which is extremely difficult to model. Although several methods exist for modeling the Difference Distribution Table (DDT) and related tables in constraint-programming tools [SHW+14b, AST+17, BC20, SW23], they need a significant number of inequalities to represent 8-bit S-boxes effectively.

Our first contribution addresses this challenge by introducing a novel Mixed Integer Linear Programming (MILP) model that can efficiently handle 8-bit S-boxes while searching for all quasidifferential trails corresponding to a given differential characteristic. As demonstrated in Section 5, this model provides effective results for both SKINNY-128 and AES.

Additionally, we present a theoretical extension to the framework of [BR22] that incorporates the key schedule, particularly handling nonlinear key schedules. The original work considers the round keys as parameters, allowing quasidifferential trails to compute the probability of a differential characteristic as a function of the round key bits. While it is possible to apply the exact probability formula to the key schedule afterwards, we propose a novel approach that offers several advantages. In this approach, we treat the key as part of the *data* within the original framework. This enables the computation of the average probability of a differential characteristic over all pairs of messages, with round keys generated from the master key via the key schedule. Although this method does not yield an exact probability formula, it provides significant benefits. First, it is computationally efficient, whereas computing the exact formula is often intractable. Second, it is the first approach that permits to compute the expected differential probability of a differential characteristic by taking the key schedule into account.

Using our MILP model we provide several applications of both the original quasidifferential framework and its extensions developped in this work. First, we applied our extended framework on related-key characteristics on all three versions of AES and proved their validity. As a second application, we show that we are able to reproduce in a practical way most of experiments on SKINNY-64 and SKINNY-128 conducted in [PT22] in the fixed-key model. For some characteristics the results match while for some others we are able to detect more key constraints. This not only demonstrates that the framework in [PT22] is less complete than the one of [BR22] but also indicates that efficiently implementing the quasifferential framework is the best solution for computing the fixed-key probability of differential characteristics. Finally, we apply our extension of the quasidifferential framework to verify the validity of the 114 688 differential characteristics used as part of a differential distinguisher of two recent differential meet-in-middle attacks [BDD+23, AKM+24]. We show that a non-negligible portion of these characteristics are invalid, however, the average probability of the remaining characteristics is higher than expected and has as a consequence that the overall probability of the differential is much higher than estimated by the authors of these attacks. This permits to decrease the overall complexity of both attacks. Finally, we provide several examples on toy ciphers with different key schedules and explain how the key schedule influences the probability of a characteristic and why we believe that our model can be used as an efficient tool by designers to help them choose a key schedule that would strengthen the security of block ciphers against differential attacks.

The rest of the article is organized as follows. In Section 2, we recall the basics of differential cryptanalysis and briefly introduce the framework of Beyne and Rijmen. In Section 3 we introduce our extensions to this framework. Then, in Section 4 we describe our new MILP model for implementing the framework of [BR22]. Finally, in Section 5 we present our applications to SKINNY and AES.

Our complete source code is available at:

https://gitlab.inria.fr/capsule/extensions-quasidifferential-framework/

2 Differential cryptanalysis and quasidifferential framework

2.1 Differential cryptanalysis

Differential cryptanalysis is a powerful attack technique against block ciphers and other symmetric cryptographic primitives. Introduced in 1990 by Biham and Shamir [BS91], it exploits the existence of high-probability *differentials*, which are defined as follows.

Definition 1 (Differential)**.** Let F be a function over $\mathbb{F}_2^n$. A *differential* of F is a pair of input and output differences $(\Delta_{in}, \Delta_{out}) \in \mathbb{F}_2^n \times \mathbb{F}_2^n$. The *differential probability* of $(\Delta_{in}, \Delta_{out})$ is defined as:

$$\mathrm{DP}_F(\Delta_{in}, \Delta_{out}) := \mathrm{Pr}_x\left[F(x) \oplus F(x \oplus \Delta_{in}) = \Delta_{out}\right],$$

where P_x denotes the probability computed by averaging over all possible inputs $x \in \mathbb{F}_2^n$.

The attacker's goal is to identify a high-probability differential for a target block cipher E, which can be viewed as a family of permutations E_k, where each instance is parameterized by a secret key k. However, since the key used to instantiate the block cipher is unknown, cryptanalysts usually focus on estimating the so-called *expected differential probability (EDP)*, defined as the average probability of a differential $(\Delta_{in}, \Delta_{out})$ across all keys:

$$\mathrm{EDP}_E(\Delta_{in}, \Delta_{out}) := \mathrm{Pr}_{x,k}\left[E_k(x) \oplus E_k(x \oplus \Delta_{in}) = \Delta_{out}\right],$$

where $P_{x,k}$ represents the probability averaged over all possible inputs x and keys k.

After identifying a differential $(\Delta_{in}, \Delta_{out})$ with a high expected differential probability, the attacker uses it to distinguish a particular instance E_k from a random permutation, by explicitly relying on the *stochastic equivalence hypothesis* [LMM91] that says that the probability of the differential for a particular instance E_k is close to the average case:

$$\mathrm{EDP}_E(\Delta_{in}, \Delta_{out}) \approx \mathrm{DP}_{E_k}(\Delta_{in}, \Delta_{out}).$$

As discused in the introduction, this hypothesis is known to fail in many cases as it has been observed that the probability of the differential can significantly vary across keys [Knu93, DR07, AK19, Hey20, PT22].

2.1.1 Differential characteristics

Estimating the exact probability of a differential $(\Delta_{in}, \Delta_{out})$ after r rounds is a hard problem. For this reason, it is common to exploit the iterative structure of block ciphers, i.e., the fact that a block cipher instance can be written as $E_k = F_r \circ \cdots \circ F_1$, and compute instead the probability of a *differential characteristic*:

Definition 2 (Differential Characteristic)**.** A differential characteristic over $F_r \circ \cdots \circ F_1$ is an $(r+1)$-tuple $Q = (a_1, \ldots, a_{r+1}) \in (\mathbb{F}_2^n)^{r+1}$ with differential probability:

$$\mathrm{DP}(Q) := \mathrm{Pr}\left[\bigwedge_{i=1}^{r} F_i(\boldsymbol{x_i} \oplus a_i) = F_i(\boldsymbol{x_i}) \oplus a_{i+1}\right],$$

where $\boldsymbol{x_1}$ is uniformly random over $\mathbb{F}_2^n$ and $\boldsymbol{x}_i = F_{i-1}(\boldsymbol{x}_{i-1})$ for $i = 2, \ldots, r$.

As for the case of differentials, cryptanalysts usually focus on estimating the *expected differential probability (EDP)* of a differential characteristic that is defined as the probability of the characteristic averaged on all keys.

The probability of a differential characteristic is usually computed by assuming the one-round differentials composing a differential characteristic being independent, reducing the overall computation to smaller computations of probabilities over a single round:

$$\Pr\left[\bigwedge_{i=1}^{r} F_i(\boldsymbol{x}_i \oplus a_i) = F_i(\boldsymbol{x}_i) \oplus a_{i+1}\right] = \prod_{i=1}^{r} \Pr\big[F_i(\boldsymbol{x}_i \oplus a_i) = F_i(\boldsymbol{x}_i) \oplus a_{i+1}\big]. \qquad (1)$$

In key-alternating ciphers, if the round keys are independent and uniformly distributed, this round independence results from the addition to the state of a new round key at the beginning of each round. However, this independence assumption is rarely satisfied for block ciphers used in practice, for the simple reason that the round keys are usually derived from the master key by using an algorithm called the key schedule. For this reason, the round keys cannot be considered as independent.

Difference Distribution Table (DDT) To compute the probability of a differential over one round in S-box-based designs, one typically relies on the so-called Difference Distribution Table (DDT) of the cipher's S-box. A DDT for an m-bit S-box S is a $2^m \times 2^m$ table. For an input difference $a \in \mathbb{F}_2^m$ and an output difference $b \in \mathbb{F}_2^m$, the entry $\mathrm{DDT}[a][b]$ records the number of solutions to the equation $S(x) \oplus S(x \oplus a) = b$.

Often, in differential attacks, the probability of the differential is approximated by the probability of its dominant (i.e., highest-probability) differential characteristic. Indeed, for a fixed key k, a pair of input values $(x, x \oplus \Delta_{in})$ to the function $F_r \circ \cdots \circ F_1$ follows exactly one differential characteristic. Consequently, if we denote by $\mathrm{DT}(\Delta_{in}, \Delta_{out})$ the set of differential characteristics of the form $(\Delta_{in} = a_1, a_2, \ldots, \Delta_{r+1} = \Delta_{out})$, it holds that:

$$\mathrm{DP}_F(\Delta_{in}, \Delta_{out}) = \sum_{Q \in \mathrm{DT}(\Delta_{in}, \Delta_{out})} \mathrm{DP}_F(Q),$$

which shows that the probability of a differential can be lower bounded by the probability of any of its characteristics.

2.2 The framework of Beyne and Rijmen [BR22]

Beyne and Rijmen were the first to study in a systematic and generic way the probability of differential characteristics in the fixed-key setting. We briefly introduce in this section the central notions of their framework. We follow for this the notations of [BR22] as close as possible. More details can be found in the original article.

To evaluate the fixed-key probability of a characteristic, one aims to track the probability distribution of pairs through each step of the cipher while keeping track of key-related conditions. More formally, this involves operations on distributions, i.e., functions from $\mathbb{F}_2^n$ to $\mathbb{R}$. The vector space of all such functions is denoted by $\mathbb{R}[\mathbb{F}_2^n]$. A basis for this space is given by $(\delta_a)_{a \in \mathbb{F}_2^n}$, where $\delta_a(x)$ equals 1 if $a = x$ and 0 otherwise. This basis is referred to as the *standard basis*. Using the standard basis, Beyne and Rijmen introduce the concept of *transition matrix*, defined as follows:

Definition 3 (Transition Matrix, [BR22])**.** Let $F : \mathbb{F}_2^n \to \mathbb{F}_2^m$ be a function. The transition matrix of F is the coordinate representation of the unique linear operator $T^F : \mathbb{R}[\mathbb{F}_2^n] \to \mathbb{R}[\mathbb{F}_2^m]$, defined by:

$$\delta_x \mapsto \delta_{F(x)} \quad \text{for all } x \in \mathbb{F}_2^n.$$

The matrix is expressed with respect to the standard bases of $\mathbb{R}[\mathbb{F}_2^n]$ and $\mathbb{R}[\mathbb{F}_2^m]$.

In the study of differential cryptanalysis, it is necessary to work with pairs, i.e., to consider the space $\mathbb{R}[\mathbb{F}_2^n \times \mathbb{F}_2^n]$. The standard basis for this space is given by $\left(\delta_{(a,b)}\right)_{a,b\in\mathbb{F}_2^n}$, where $\delta_{(a,b)}(x,y)$ equals 1 if $x = a$ and $y = b$, and 0 otherwise, for any $(x,y) \in \mathbb{F}_2^n \times \mathbb{F}_2^n$.

The analogue of the transition matrix for pairs, for a function $F : \mathbb{F}_2^n \to \mathbb{F}_2^m$, is represented by $T^F \otimes T^F$ and is defined as:

$$(T^F \otimes T^F)_{(y_1,y_2),(x_1,x_2)} = T^F_{y_1,x_1} T^F_{y_2,x_2} = \delta_{y_1}(F(x_1))\delta_{y_2}(F(x_2)).$$

This corresponds to the representation of the linear operator $\delta_{(x,y)} \mapsto \delta_{(F(x),F(y))}$ with respect to the standard bases of $\mathbb{R}[\mathbb{F}_2^n \times \mathbb{F}_2^n]$ and $\mathbb{R}[\mathbb{F}_2^m \times \mathbb{F}_2^m]$.

However, the choice of basis must also be well-suited to facilitate tracking of key-related operations, such as key addition in most cases. Consequently, [BR22] introduced the concept of the *quasidifferential basis.*

Definition 4 (Quasidifferential Basis, [BR22])**.** Let n be a positive integer. For any $u, a \in \mathbb{F}_2^n$, the function $\beta_{u,a} : \mathbb{F}_2^n \times \mathbb{F}_2^n \to \mathbb{R}$ is defined as:

$$\beta_{u,a}(x,y) = \chi_u(x)\delta_a(x+y),$$

where $\chi_u(x) = (-1)^{u^T x}$. The set of all functions $\{\beta_{u,a} \mid u, a \in \mathbb{F}_2^n\}$ is called the *quasidifferential basis* for $\mathbb{R}[\mathbb{F}_2^n \times \mathbb{F}_2^n]$. This basis is orthogonal.

The quasidifferential basis is translation invariant, i.e., for all $u, a, t \in \mathbb{F}_2^n$, we have:

$$\beta_{u,a}(x+t, y+t) = \chi_u(t)\beta_{u,a}(x,y).$$

A change-of-basis operator between the *standard basis* and the *quasidifferential basis* is defined as $\mathcal{Q}_n : \mathbb{R}[\mathbb{F}_2^n \times \mathbb{F}_2^n] \to \mathbb{R}[\mathbb{F}_2^n \times \mathbb{F}_2^n]$, where:

$$(\mathcal{Q}_n f)(u,a) = \langle \beta_{u,a}, f\rangle.$$

Using the properties of the quasidifferential basis, the value $(\mathcal{Q}_n f)(u,a)/2^n$ corresponds to the coordinate of $\beta_{u,a}$ in the quasidifferential basis.

This change-of-basis operator is applied to $T^F \otimes T^F$, resulting in the definition of the *quasidifferential transition matrix* (QDTM). Its associated properties are listed in Theorem 1.

Definition 5 (Quasidifferential Transition Matrix, [BR22])**.** Let n and m be positive integers, and let $F : \mathbb{F}_2^n \to \mathbb{F}_2^m$ be a function. The *quasidifferential transition matrix* D^F is defined as the matrix representation of $T^F \otimes T^F$ with respect to the quasidifferential basis. Specifically:

$$D^F = \mathcal{Q}_m(T^F \otimes T^F)\mathcal{Q}_n^{-1}.$$

Developing the expression of D^F yields the following formula:

$$D^F_{(v,b),(u,a)} = \frac{1}{2^n} \sum_{\substack{x\in\mathbb{F}_2^n \\ F(x+a)=F(x)+b}} (-1)^{u^\intercal x + v^\intercal F(x)} \tag{2}$$

With Equation (2) one can prove the following quasidifferential transition matrix properties:

Theorem 1 ([BR22])**.** *Let n and m be positive integers and $F : \mathbb{F}_2^n \to \mathbb{F}_2^m$ a function. The matrix D^F has the following properties:*

1. *If F is a bijection, then D^F is an orthogonal matrix.*

2. *If* $F = (F_1, \ldots, F_m)$*, then* $D^F = \bigotimes_{i=1}^m D^{F_i}$.

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

4. *If* $F(x) = x + t$ *for some* $t \in \mathbb{F}_2^n$*, then* $D^F_{(v,b),(u,a)} = \chi_v(t)\delta_v(u)\delta_b(a)$.

5. *If* F *is linear, then* $D^F_{(v,b),(u,a)} = \delta_u(F^\intercal(v))\delta_b(F(a))$.

2.2.1 Quasidifferential trails and associated theorems

To analyze the propagation of probability distributions of pairs through an iterated function, Beyne and Rijmen introduced the notion of *quasidifferential trail*, which is defined using the quasidifferential transition matrix and its associated properties.

Definition 6 (Quasidifferential Trail, [BR22])**.** A quasidifferential trail for a function $F = F_r \circ \cdots \circ F_1$ is a sequence $(u_1, a_1), \ldots, (u_{r+1}, a_{r+1})$ of so-called *mask-difference pairs*. The correlation of this quasidifferential trail is defined as

$$\prod_{i=1}^{r} D^{F_i}_{(u_{i+1},a_{i+1}),(u_i,a_i)}$$

A first observation is that a quasidifferential trail $(0, a_1), \ldots, (0, a_{r+1})$ with zero masks corresponds directly to a differential characteristic with intermediate differences $a_1, \ldots, a_{r+1}$. By Equation (2), the correlation of such a quasidifferential trail is given by:

$$\prod_{i=1}^{r} D^{F_i}_{(0,a_{i+1}),(0,a_i)} = \prod_{i=1}^{r} \Pr[F_i(\boldsymbol{x} + a_i) = F_i(\boldsymbol{x}) + a_{i+1}],$$

where $\boldsymbol{x}$ is uniformly random over $\mathbb{F}_2^n$. This expression matches then the formula given in Equation (1) for computing the probability of a differential characteristic under the independence hypothesis.

In the following, we present two key results related to quasidifferential trails that are most relevant to our work.

Theorem 2 ([BR22])**.** *Let* $F : \mathbb{F}_2^n \to \mathbb{F}_2^m$ *be a function such that* $F = F_r \circ \cdots \circ F_1$*. The probability of a characteristic with differences* $a_1, \ldots, a_{r+1}$ *is equal to the sum of the correlations of all quasidifferential trails with the same intermediate differences:*

$$\Pr[\bigwedge_{i=1}^{r} F_i(\boldsymbol{x}_i + a_i) + F_i(\boldsymbol{x}_i) = a_{i+1}] = \sum_{u_2,\ldots,u_r} \prod_{i=1}^{r} D^{F_i}_{(u_{i+1},a_{i+1}),(u_i,a_i)}$$

with $u_1 = u_{r+1} = 0$*,* $\boldsymbol{x}_i = F_{i-1}(\boldsymbol{x}_{i-1})$ *for* $i = 2, \ldots, r$ *and* $\boldsymbol{x}_1$ *uniform random on* $\mathbb{F}_2^n$*.*

This important theorem implies that the fixed-key probability of a differential characteristic can be computed exactly if all quasidifferential trails associated with it are identified. Consequently, determining the fixed-key probability of a characteristic reduces to the problem of developing efficient tools to find quasidifferential trails, or at least those with high correlation, to obtain a good approximation of the fixed-key probability.

Remark: An important observation is that this theorem does not take directly into account relations between the round keys. In order, to obtain a formula on the master key one must replace the involved round key bits by an expression of the master key bits. This can lead to non-negligible overhead in the fixed-key analysis, especially for non-linear key schedules. We discuss this point in more details in Section 3.

Theorem 3 ([BR22])**.** *For a function $F = F_r \circ \cdots \circ F_1$ and a characteristic $a_1, \ldots, a_{r+1}$ with correlation p (as a quasidifferential trail, i.e., with all masks set to 0), it holds that:*

1. *Let $(u_1, a_1), \ldots, (u_{r+1}, a_{r+1})$ be a quasidifferential trail with correlation $(-1)^b p$ where $b \in \{0, 1\}$. Let $(v_1, a_1), \ldots, (v_{r+1}, a_{r+1})$ be a quasidifferential trail of correlation c. Note that the intermediate differences are the same, only the masks change. Then the correlation of the quasidifferential trail $(u_1 + v_1, a_1), \ldots, (u_{r+1} + v_{r+1}, a_{r+1})$ equals $(-1)^b c$. Formally we have,*

$$\prod_{i=1}^{r} D^{F_i}_{(u_{i+1}+v_{i+1}, a_{i+1}), (u_i+v_i, a_i)} = (-1)^b \prod_{i=1}^{r} D^{F_i}_{(v_{i+1}, a_{i+1}), (v_i, a_i)}$$

2. *Let T be the set of all quasidifferential trails having differences $a_1, \ldots, a_{r+1}$ and correlation $\pm p$. If there exists $D \subseteq T$ such that*

$$\sum_{d \in D} \mathrm{corr}(d) = 0$$

then the characteristic has a probability equal to zero. Here, $\mathrm{corr}(d)$ *denotes the correlation of a trail d.*

The second property highlights the interest of quasidifferential trails that share the same absolute correlation as the corresponding differential characteristic. This property can lead to impossibility results, as illustrated for example in [BN24], where Beyne and Neyt analyzed the characteristic used in [BDBN23] to attack SPEEDY-7-192. They identified a quasidifferential trail with a correlation equal to the negation of the differential characteristic's correlation for all keys. As a result, using Theorem 3.2, they demonstrated that the characteristic has a probability of zero for every possible key, invalidating thus the proposed attack against this instance of SPEEDY.

We provide next a simple example that demonstrates how the quasidifferential framework is used in practice to compute the exact probability of a differential characteristic and the way that key constraints appear.

Example 1. For this example we consider a tiny toy cipher composed of two applications of a 4-bit S-box with a key addition between them. The S-box used is the one of SKINNY-64. We propose a simple characteristic illustrated in Figure 1.

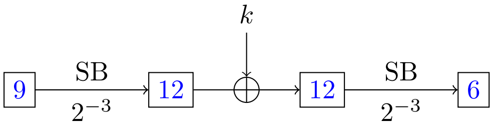


Figure 1: A differential characteristic on the toy cipher which has *a priori* a probability of 2^{-6} for all keys.

By looking at the quasidifferential transition matrix D^S of this S-box and in particular the two blocks of this matrix corresponding to the differential transitions 9 → 12 and 12 → 6 (the blocks are respectively given in Figure 9 and Figure 10 in Appendix A) one can compute all the quasidifferential trails associated to this characteristic. An example of such quasidifferential trail is given by this sequence of mask-difference pairs:

$$(0, 9) \xrightarrow{\quad\text{SB}\quad} (1, 12) \xrightarrow{\quad\text{AK}\quad} (1, 12) \xrightarrow{\quad\text{SB}\quad} (0, 6)$$

where the differences are colored in blue and the masks in red. The correlation of this trail equals to:

$$D^S_{(1,12),(0,9)} \times \chi_1(k) \times D^S_{(0,6),(1,12)} = (-2^{-3}) \times (-1)^{k^T(0,0,0,1)} \times (-2^{-3}) = 2^{-6}(-1)^{k_0}$$

where $k = (k_3k_2k_1k_0)_2$ is the binary decomposition of k with the most significant bit being k_3.

Only 8 quasidifferential trails are associated to this characteristic and their correlations sum up to this expression:

$$\begin{aligned} p_k &= 2^{-6}\big(1 + (-1)^{k_0} + (-1)^{k_1} + (-1)^{k_2+k_3} \\ &\quad + (-1)^{k_0+k_1} + (-1)^{k_0+k_2+k_3} + (-1)^{k_1+k_2+k_3} + (-1)^{k_0+k_1+k_2+k_3}\big) \\ &= 2^{-6}((-1)^{k_1}+1)((-1)^{k_0}+1)((-1)^{k_2+k_3}+1) \\ &= \begin{cases} 2^{-3} \text{ if } k_1 = k_0 = 0 \text{ and } k_2 = k_3 \\ 0 \text{ otherwise.} \end{cases} \end{aligned}$$

3 Extension: related-key setting and average probability

In this section, we present two extensions of the quasidifferential framework. First, we demonstrate how the framework can be easily adapted to compute the exact probability of a related-key differential characteristic. Next, recognizing that the exact probability formula can be highly complex, we propose an alternative approach to utilize the quasidifferential framework. This approach enables the computation of the expected average probability of a differential characteristic for round keys generated by the key schedule.

3.1 Quasidifferentials in the related-key setting

The applicability of the quasidifferential framework in the context of related-key characteristics is not straightforward. Indeed, the original framework proposed in [BR22] was designed to analyze differential characteristics where both elements of a pair are encrypted under the same key, meaning that the exactly same operations are applied to both elements. However, in the case of related-key characteristics, the key addition is asymmetrical for these two elements as the first undergoes an addition with some value t, while the other one with some value $t + c$. To address this, we propose an extension of the framework to account for operations that treat the elements of a pair in an asymmetrical manner.

Let F and G be two functions from $\mathbb{F}_2^n$ to $\mathbb{F}_2^m$. Instead of considering $T^F \otimes T^F$, we analyze now the transition matrix $T^F \otimes T^G$. Following the approach of Beyne and Rijmen, we apply the change-of-basis operator $\mathcal{Q}_n$ and derive a formula analogous to Equation (2):

$$D^{F/G}_{(v,b),(u,a)} = \frac{1}{2^n} \sum_{\substack{x\in\mathbb{F}_2^n \\ G(x+a)=F(x)+b}} (-1)^{u^Tx+v^TF(x)} \tag{3}$$

Then, Theorem 2 can be reformulated as:

Theorem 4. *Let $F, G : \mathbb{F}_2^n \to \mathbb{F}_2^m$ be two functions such that $F = F_r \circ \cdots \circ F_1$ and $G = G_r \circ \cdots \circ G_1$. Then,*

$$\Pr[\bigwedge_{i=1}^{r} G_i(\boldsymbol{x}_i + a_i) + F_i(\boldsymbol{x}_i) = a_{i+1}] = \sum_{u_2,\ldots,u_r} \prod_{i=1}^{r} D^{F_i/G_i}_{(u_{i+1},a_{i+1}),(u_i,a_i)}$$

with $u_1 = u_{r+1} = 0$, $\boldsymbol{x}_i = F_{i-1}(\boldsymbol{x}_{i-1})$ for $i = 2, \ldots, r$ and $\boldsymbol{x}_1$ uniform random on $\mathbb{F}_2^n$.

This yields an exact formula for the probability of a related-key characteristic with independent round keys.

Regarding the behavior of mask-difference pairs under key addition, we can apply Equation (3) using the functions $F : x \mapsto x + t$ and $G : x \mapsto x + t + c$, where $t, c \in \mathbb{F}_2^n$. This yields the following expression for the correlation coefficient:

$$D^{F/G}_{(v,b),(u,a)} = \chi_u(t)\delta_u(v)\delta_b(a + c)$$

Example 2. Consider the same toy cipher as in Example 1. Again, we propose a simple related-key characteristic (see Figure 2) to illustrate the application of this extension. Here, by applying our extended framework we obtain 4 quasidifferential trails, leading to the following formula for the fixed-key probability:

$$\begin{aligned} p_k &= 2^{-6}\left(1 + (-1)^{k_1} + (-1)^{k_2+k_3} + (-1)^{k_1+k_2+k_3}\right) \\ &= 2^{-6}\left((-1)^{k_1} + 1)(1 - (-1)^{k_2+k_3}\right) \\ &= \begin{cases} 2^{-4} \text{ if } k_1 = 0 \text{ and } k_3 \neq k_2 \\ 0 \text{ otherwise} \end{cases} \end{aligned}$$

which can be easily verified to be the exact distribution for this characteristic.

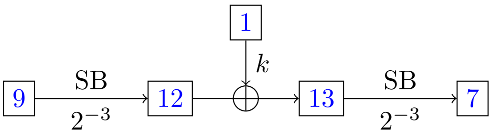


Figure 2: Related-key characteristic on our toy cipher.

An important remark about this extension is that it provides an exact formula for a related-key characteristic when making abstraction of the key-schedule. In the case of a linear key schedule, while the round keys are uniform, they are not independent. However, it is straightforward to reverse the key schedule to derive constraints on the master key. For a non-linear key schedule, the round keys are neither independent *nor* uniform. While the dependency issue can be addressed by reversing the key schedule, by using for example the algebraic normal form of the S-box, the non-uniformity cannot be easily accounted for. This limitation motivated us to propose a new extension of the quasidifferential framework. This extension aims to compute the EDP of a characteristic by directly incorporating the constraints of the key schedule. The details of this new approach are presented in Section 3.2.

3.2 Quantifying the EDP of single-key and related-key characteristics

In this section, we propose an approach to compute the EDP of a differential characteristic in both the single-key and related-key models by treating the key as *data* within the quasidifferential framework. While determining the exact probability of a differential characteristic as a function of the master key is theoretically appealing and should be done whenever possible, it is often infeasible due to the inherent complexity of the problem. Computing the exact probability requires evaluating the entire cluster of associated quasidifferential trails, which can be intractable due to the very high number of trails and the difficulty of enumerating them.

Furthermore, restricting the computation to quasidifferential trails with correlations above a fixed threshold can lead to inconsistencies, such as formulas yielding negative probabilities for certain keys. Additionally, the fixed-key probability of a differential characteristic typically results in a complex formula involving many key bits, making practical estimation unmanageable. Finally, cryptanalysts are generally more concerned with assessing the validity of a differential rather than obtaining a detailed breakdown of its probability. One practical approach is to verify the validity of a large number of characteristics and justify their diversity to ensure coverage of the entire key space.

To address these challenges, we propose a method to efficiently evaluate both the validity and average probability of a characteristic without relying on exact formulas. Notably, this method is particularly well-suited for related-key characteristics, even when the key schedule is non-linear, and it is simpler to apply than the original framework based on independent round keys. Our method builds on an extension of the framework introduced in [BR22], treating the key as data. In the following, we first provide a theoretical explanation of this extension and then validate it using a toy cipher. Applications of this extension are presented in Section 5.

3.2.1 Extension of [BR22]'s framework

When studying a related-key characteristic, one needs to take into account the constraints on the key that arise from the key schedule in addition with the constraints on the data path. Then, it is required to check that those constraints are compatible. Here, we chose to view this by considering the related-key characteristic as a single characteristic on both the state-update function and the key schedule. Accordingly, we now consider quasidifferential trails on both the plaintext and the key. More formally, an operation F on the data path is now seen as:

$$\begin{aligned}\overline{F}\colon \mathbb{F}_2^n \times \mathbb{F}_2^m &\to \mathbb{F}_2^n \times \mathbb{F}_2^m\\ (x,k) &\mapsto (F(x),k)\end{aligned}$$

Using Theorem 1, for a function $\overline{F} = (F, id_{\mathbb{F}_2^m})$, it holds that:

$$\begin{aligned}D^{\overline{F}}_{(v_1||v_2,b_1||b_2),(u_1||u_2,a_1||a_2)} &= (D^F \otimes D^{id_{\mathbb{F}_2^m}})_{(v_1||v_2,b_1||b_2),(u_1||u_2,a_1||a_2)}\\ &= D^F_{(v_1,b_1),(u_1,a_1)}\delta_{v_2}(u_2)\delta_{b_2}(a_2)\end{aligned}$$

where $\|$ denotes the concatenation. Other operations involving only the key (i.e., key schedule's operations) are treated the same way. Consequently, the quasidifferential trail on the data path and the one on the key path do not interact with each other for operations on only one of the inputs. Let now see how masks are affected during the key addition, given by the following operation:

$$\begin{aligned}G : \mathbb{F}_2^n \times \mathbb{F}_2^n &\to \mathbb{F}_2^n \times \mathbb{F}_2^n\\ (x,k) &\mapsto (x+k,k)\end{aligned}$$

The correlation coefficient associated to this operation can be computed using Proposition 1.

Proposition 1. *Let n be a positive integer.*
For any $v_x, v_k, b_x, b_k, u_x, u_k, a_x, a_k \in \mathbb{F}_2^n$ it holds that:

$$D^G_{(v_x\|v_k,b_x\|b_k),(u_x\|u_k,a_x\|a_k)} = \delta_{b_x}(a_x+a_k)\delta_{b_k}(a_k)\delta_{v_x}(u_x)\delta_{v_k}(u_x+u_k)$$

Proof. G is a linear mapping thus we can apply Theorem 1.5 with $G^T(x,k) :- (x, x+k)$:

$$\begin{aligned}&D^G_{(v_x\|v_k,b_x\|b_k),(u_x\|u_k,a_x\|a_k)}\\ &= \delta_{(b_x,b_k)}(G(a_x,a_k))\delta_{(u_x,u_k)}(G^T(v_x,v_k))\\ &= \delta_{b_x}(a_x+a_k)\delta_{b_k}(a_k)\delta_{u_x}(v_x)\delta_{u_k}(v_x+v_k)\end{aligned}$$

□

It is important to note that no information about the key is present in this formula, meaning the fixed-key information is entirely lost. However, we can still apply Theorem 2 and obtain the exact formula for the EDP of a characteristic:

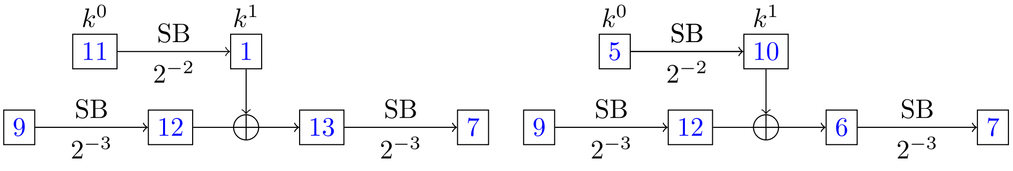
Theorem 5. *Let $E = F_r \circ \cdots \circ F_1$ and let $Q = \big((a_x^1, a_k^1), \dots, (a_x^{r+1}, a_k^{r+1})\big)$ represent a related-key differential characteristic over E. Here, $(a_x^1, \dots a_x^{r+1})$ and $(a_k^1, \dots, a_k^{r+1})$ correspond to the differential characteristics on the plaintext and key, respectively. If $a_k^1 = \cdots = a_k^{r+1} = 0$, then Q is a characteristic in the single-key model. Then,*

$$EDP(Q) :- \Pr\left[\bigwedge_{i=1}^{r} F_i\left((\boldsymbol{x_i}, \boldsymbol{k_i}) + (a_x^i, a_k^i)\right) + F_i(\boldsymbol{x_i}, \boldsymbol{k_i}) = (a_x^{i+1}, a_k^{i+1})\right]$$
$$= \sum_{\substack{u_x^2, \dots, u_x^r \\ u_k^2, \dots, u_k^r}} \prod_{i=1}^{r} D^{F_i}_{\left((u_x^{i+1}, u_k^{i+1}),(a_x^{i+1}, a_k^{i+1})\right),\left((u_x^i, u_k^i),(a_x^i, a_k^i)\right)}$$

where $(u_x^1, u_k^1) = (u_x^{r+1}, u_k^{r+1}) = (0, 0)$, $(\boldsymbol{x_i}, \boldsymbol{k_i}) = F_{i-1}(\boldsymbol{x_{i-1}}, \boldsymbol{k_{i-1}})$ for $i = 2, \dots, r$ and $(\boldsymbol{x_1}, \boldsymbol{k_1})$ uniform random on $\mathbb{F}_2^n \times \mathbb{F}_2^n$.

This theorem is a corollary of Theorem 2 and simply states that the EDP of a characteristic can be computed by identifying the associated quasidifferential trails when considering both the plaintext and the key as data. This probability will be zero if the characteristic is impossible and non-zero otherwise, with potential variations in the average probability. We illustrate our new method in Example 3.

Example 3. Again, we consider our toy cipher and add an S-box in the key schedule. We use two characteristics shown in Figure 3. The left one is possible for some keys and the right one is impossible for all keys.



(a) Possible characteristic

(b) Impossible characteristic

Figure 3: Two related-key characteristics on our toy cipher

The possible characteristic corresponds to the one used in Example 2 except that a differential transition was added before the key addition. In the fixed-key setting, the extended framework outputs that the possible values for k^1 are $\{4, 5, 8, 9\}$ as already discussed in Example 2. In addition, the possible values for k^1 by looking only at the differential transition 11 → 1 are $\{8, 9, 12, 13\}$. Thus the probability to draw a valid key is 2^{-3} and the probability that a plaintext passes through the characteristic knowing that a valid key was drawn is 2^{-4} leading to an average probability of 2^{-7}. Here, we can find this by hand but on a real cipher with multiple S-boxes and linear operations it represents a real challenge. Using our method, we find two quasidifferential trails, one has all its masks equal to zero and the other is represented in Figure 4. Each of them has a correlation equal to 2^{-8}, and summing them results in the correct average probability.

Now, if we look at the other characteristic the possible values of k^1 in a fixed-key setting would be $\{2, 4, 8, 14\}$ but the possible values that pass through the differential transition 5 → 10 are $\{5, 7, 13, 15\}$ which explains the impossibility. Again, our model outputs two quasidifferential trails one having the opposite correlation of the other yielding an average probability equal to zero.

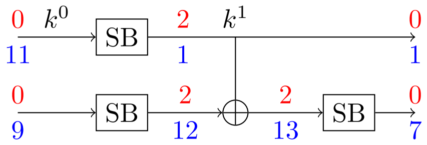


Figure 4: Quasidifferential trail associated to the valid characteristic. Differences are in blue and masks in red

What is particularly powerful about this use of the quasidifferential framework is that, for the first time, we can compute the average probability of a differential characteristic for round keys generated by a key schedule. This approach eliminates the need to assume that the round keys are independent, significantly broadening the applicability of the framework.

An important question is whether it is easier to compute the average probability or the fixed-key probability of a differential characteristic. In all our experiments, computing the average probability consistently proved to be faster. This is because the number of trails involved was always smaller, and the corresponding model completed more quickly. We attribute this to the formula in Proposition 1, specifically the terms $\delta_{b_x}(a_x + a_k)$ and $\delta_{v_k}(u_x + u_k)$ which impose strong constraints between the masks on the data path and those on the key path.

Moreover, key schedules are often relatively simple functions with few or no non-linear components, resulting in a limited number of possible masks on the key path. This, in turn, reduces the potential masks on the data path. Intuitively, each quasidifferential trail captures a variation in the probability of the characteristic. When the key is included in the trail, a quasidifferential trail explains a global variation in the probability, applicable across all keys. In contrast, in the fixed-key setting, the variation is local and applies only to a subset of keys. Consequently, it is unsurprising that fewer quasidifferential trails are required to compute the average probability than the fixed-key probability.

To illustrate our claims we detail an example of a single-key characteristic on which we apply three different key schedules. Each choice of key schedule has a different impact on the probability of the characteristic. We compare the findings of our extended extension with a fixed-key approach to illustrate how this extension could be used in order to design a strong key schedule.

Example 4. In this example, we study a cipher composed of 3 successive applications of the `AES` S-box with a key addition between every S-box. This single-key characteristic is given in Figure 5. Let k_0 and k_1 denote the two round keys used in the characteristic. We apply our model on three different key schedules: the first one is described by $k_1 = k_0$, the second one by $k_1 = SB(k_0)$ and the last one by $k_1 = SB'(k_0)$ where SB denotes the `AES` S-box and SB' corresponds to the S-box of `SKINNY-128`.

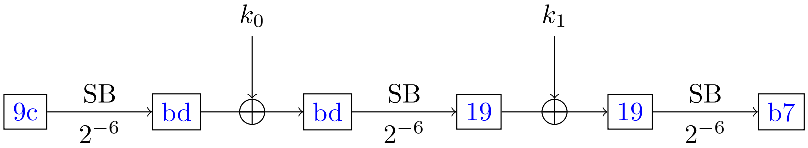


Figure 5: Single-key characteristic on our toy cipher having an estimated probability of 2^{-18}. The differences are given in hexadecimal notation

First, we conducted a fixed-key analysis of this characteristic. We exhausted all the associated quasidifferential trails resulting in 2048 trails. We then derived a fixed-key expression for each key schedule by reversing it using the algebraic normal form of the two different S-boxes. Finally, we evaluated the three expressions for all possible values of the key bits involved in the expression. For the first key schedule, two valid keys are found with an associated probability of 2^{-15} each. The second key schedule created an incompatibility in the characteristic thus no valid key was found. Lastly, when replacing in the key schedule the S-box of the AES by the one of SKINNY-128 the characteristic was possible for one key with a probability of 2^{-15}. We then applied our extension on this characteristic with the three different key schedules and it yielded the correct average probability i.e., respectively $2^{-14}, 0$ and 2^{-15}. It is important to remark that our model identified fewer trails compared to the fixed-key model, with 16, 1875, and 785 trails for the three respective key schedules. We also stress the fact that here the fixed-key expression was only involving a few key bits. On a real cipher with more rounds and linear operations this expression can quickly become unusable with too many bits involved. On the other hand, we believe that the extension that we proposed can scale for more complex study cases as we show in Section 5.1.

4 Implementation of the quasidifferential framework

To effectively apply the quasidifferential framework in practice, an efficient implementation is essential. In this work, we introduce a novel Mixed Integer Linear Programming (MILP) model to achieve this. As we will discuss shortly, the key challenge for an efficient implementation lies in finding a compact modelization of the QDTM, minimizing the number of inequalities required. The focus of this section is to demonstrate how to achieve such a compact representation of the QDTM. Additional details on the choices we made to accelerate the global model can be found in Section 5 and specifically in Section 5.2 where we discuss several applications of the framework to the SKINNY block cipher.

4.1 Modeling quasidifferential trails

Theorem 2 states that, to evaluate the fixed-key probability of a given differential characteristic, one can search for all quasidifferential trails with the same intermediate differences, and sum their correlations. This problem is very similar to that of finding high-probability differential characteristics and can thus be approached, at least from a theoretical point of view, in a similar manner.

The propagation rules that a quasidifferential trail must satisfy are detailed in Theorem 1. For a linear layer L, incorporating the constraints into the MILP model is straightforward: in a transition from (u, a) to (v, b) through L, the associated correlation coefficient is non-zero if and only if $b = L(a)$ and $L^T(v) = u$. However, for a nonlinear operation S, there is no direct expression for D^S. The challenge lies in identifying a set of inequalities that are satisfied exclusively by valid transitions over the QDTM.

This problem is analogous to modeling a Difference Distribution Table (DDT) or any other cryptographic table and has been extensively studied, with various methods proposed [SHW$^+$14b, SHW$^+$14a, AST$^+$17, BC20, Sun21]. When applied to DDTs, the most effective of these methods enable modeling differential transitions for S-boxes of up to 8 bits, which is sufficient for most applications.

However, modeling the non-zero coefficients of the S-box's QDTM is a significantly more challenging problem, primarily due to the size of QDTM tables. Indeed, for an n-bit S-box, the corresponding QDTM is comparable in size to the DDT of a $2n$-bit S-box. This is because each non-zero coefficient $\mathrm{QDTM}_{(b\|v),(a\|u)}$ represents a possible transition $(u, a) \to (v, b)$ for $u, a, v, b \in \mathbb{F}_2^n$. If we represent each possible transition by $(b\|v\|a\|u)$, the set of these transitions can be viewed as a subset of $\mathbb{F}_2^{4n}$. In comparison, the set of possible transitions in the DDT can be viewed as a subset of only $\mathbb{F}_2^{2n}$, as differential transitions to be modeled are of the form $a \to b$, where $a, b \in \mathbb{F}_2^n$.

Therefore, for QDTMs, we need a method that takes an arbitrary subset of $\mathbb{F}_2^{4n}$ as input and returns a set of linear inequalities that describe this subset. Existing methods proposed in the literature for modeling transitions in the DDT are efficient only for subsets up to $\mathbb{F}_2^{16}$, which corresponds to 8-bit S-boxes. However, modeling the QDTM of these same S-boxes would require working with subsets of $\mathbb{F}_2^{32}$, which makes a direct application of these methods impractical.

Our main contribution in this part is to propose a straightforward and practical approach for computing a set of inequalities corresponding to the non-zero coefficients of a QDTM for large, notably 8-bit, S-boxes.

4.2 QDTM modeling

4.2.1 Exploiting QDTM's structure

The QDTM matrix is highly structured: it consists of $2^n \times 2^n$ blocks of size $2^n \times 2^n$. Each block corresponds to a specific differential transition, so when a differential transition is impossible, the entire block is composed of zeros; otherwise, there exist some non-zero coefficients in the block. The main idea behind this approach is that, since we already know the differential trail, we only need mask variables in our model, and we can apply the correct constraints only to these variables. Note that this approach was already integrated in the original SMT model of [BR22]. More formally, this means we do not need to find inequalities in $\mathbb{F}_2^{4n}$ that describe $\mathcal{P}_{b,a} = \{(b\|v\|a\|u) \mid u, v \in \mathbb{F}_2^n, \mathrm{QDTM}_{(b\|v),(a\|u)} \neq 0\}$; but only need inequalities in $\mathbb{F}_2^{2n}$ that describe $\overline{\mathcal{P}_{b,a}} = \{(v\|u) \mid u, v \in \mathbb{F}_2^n, \mathrm{QDTM}_{(b\|v),(a\|u)} \neq 0\}$ for any given $a, b \in \mathbb{F}_2^n$.

In the following, we represent a transition $(u, a) \to (v, b)$, where $u, a, v, b \in \mathbb{F}_2^n$, by $(b\|v\|a\|u) = (b_0, \ldots, b_{n-1}, v_0, \ldots, v_{n-1}, a_0, \ldots, a_{n-1}, u_0, \ldots, u_{n-1})$, following the convention that x_0 is the MSB of x. Our approach consists of two main steps: first, we map a block within $\mathbb{F}_2^{4n}$ into $\mathbb{F}_2^{2n}$; second, we use state-of-the-art methods to find a minimal set of inequalities describing the possible transitions in $\mathbb{F}_2^{2n}$. We will first describe these steps in Sections 4.2.2 and 4.2.3 and then illustrate the method with a detailed example (see Example 5).

4.2.2 Embedding in $\mathbb{F}_2^{2n}$

This step is straightforward. We begin with an initial set

$$\mathcal{P}_{b,a} = \{(b\|v\|a\|u) \mid u, v \in \mathbb{F}_2^n, \mathrm{QDTM}_{(b\|v),(a\|u)} \neq 0\} \subseteq \mathbb{F}_2^{4n}$$

and trivially transform it into

$$\overline{\mathcal{P}_{b,a}} = \{(v\|u) \mid u, v \in \mathbb{F}_2^n, \mathrm{QDTM}_{(b\|v),(a\|u)} \neq 0\} \subseteq \mathbb{F}_2^{2n}.$$

This new set retains the structure of the original set but can be efficiently modeled with a relatively small number of inequalities, using state-of-the-art methods as described in the following section.

4.2.3 Modeling techniques for an arbitrary subset in $\mathbb{F}_2^{2n}$

For efficiency reasons, one wants to minimize the number of constraints in a MILP model. Indeed, when the number of constraints is reduced by one or more orders of magnitude, the MILP solver can find an optimal solution much faster. This is why several techniques have been developed to produce a small subset of inequalities that describe a given subset of $\mathbb{F}_2^{2n}$. Initially, [SHW$^+$14b] and [SHW$^+$14a] proposed two different approaches, but these could not be applied to 8-bit S-boxes. The first approach that could be effectively applied to 8-bit S-boxes was introduced in [AST+17]. Later, several different methods applicable to large S-boxes were proposed in [BC20]. Given their success in significantly reducing the number of inequalities for modeling the S-boxes of ciphers such as AES and SKINNY-128, two ciphers central to our study, we opted to integrate these methods into our model.

Using these method, we can efficiently find a compact set of inequalities that describe $\overline{\mathcal{P}_{b,a}}$ for $a, b \in \mathbb{F}_2^n$ where $n \leq 8$. For instance, in the case of the 4-bit S-box of SKINNY-64, there are 97 possible transitions, and we computed a set of inequalities for each block in the QDTM corresponding to a possible transition. The average number of inequalities across these sets is 17.18, with a maximum of 128 inequalities, and 90% of the sets contain fewer than 32 inequalities. For SKINNY-128 that uses a 8-bit S-box, during all our applications we only needed to model 136 blocks among the 11 469 non-zero blocks of the QDTM, which by itself justifies our approach. The average number of inequalities is 350.51, with a maximum of 4578 inequalities, a minimum of 7 inequalities and 85% of the sets contains less than 115 inequalities. This clearly shows that it is possible to obtain a practical modeling of the QDTM, despite of what one could have though at first glance. To evaluate the generality of our approach, we conducted an experiment to model all nonzero blocks of the QDTM. The process took approximately 10 hours on our server, utilizing up to 128 threads. Every transition with a probability greater than 2^{-5} can be modeled with at most 1088 inequalities, though most require significantly fewer. Overall, 90% of all transitions can be modeled using at most 1024 inequalities. Still, it is important to reinforce the fact that the efficiency of this modeling depends on the underlying technique used to generate the inequalities for a single block. Recent works, such as [LS22], have improved the modelizations of [BC20], suggesting the possibility of a more efficient implementation of the quasidifferential framework. In this work, since we focus solely on fixed differential transitions, we deal exclusively with inequalities over $2n$ bits involving mask variables. Consequently, there is no need to convert these inequalities into inequalities over $4n$ bits that would describe $\mathcal{P}_{a,b}$. Such a conversion would be relevant in a more comprehensive model where the goal is to search for high-probability differential transitions, treating both differences and masks as variables. However, as this extension lies beyond the scope of this work, we only provide the method to translate inequalities describing $\overline{\mathcal{P}_{b,a}}$ into inequalities describing $\mathcal{P}_{a,b}$ in Appendix B, for completeness.

We now provide an example to illustrate the above described approach using a small block-by-block matrix.

Example 5. Consider the following matrix representing a set of possible transitions:

$$M = \left(\begin{array}{cc|cc} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ \hline \mathbf{1} & \mathbf{1} & 0 & 1 \\ \mathbf{1} & \mathbf{0} & 0 & 0 \end{array}\right)$$

Here, possible transitions are indicated by a '1' in the matrix. The set of such transitions is therefore $\{1, 4, 8, 9, 11, 12\}$, where each transition is indexed by $4i + j$, with i as the row index and j as the column index, starting from zero. Suppose we want to model the bottom-left 2×2 block, displayed in bold for clarity.

Step 1: The possible transitions within the selected block are $\{8, 9, 12\}$, i.e.,

$$\mathcal{P}_{1,0} = \{(1, 0, 0, 0), (1, 0, 0, 1), (1, 1, 0, 0)\} \subseteq \mathbb{F}_2^4$$

and $\overline{\mathcal{P}_{1,0}} = \{(0, 0), (0, 1), (1, 0)\} \subseteq \mathbb{F}_2^2$.

Step 2: The inequality $-x_0^v - x_0^u + 1 \geq 0$ is false only for $(x_0^v \| x_0^u) = (1, 1)$.

Step 3 (Optional, see Appendix B): Given $b = 1$ and $a = 0$, we use Proposition B.2 to convert the inequality, resulting in

$$-2(1 - x_0^b) - x_0^v - 2x_0^a - x_0^u + 1 \geq 0 \quad \iff \quad 2x_0^b - x_0^v - 2x_0^a - x_0^u - 1 \geq 0.$$

The reader can verify that the only elements in $\mathbb{F}_2^4$ satisfying this inequality are those in the set $\{8, 9, 12\} = \mathcal{P}_{b,a}$.

With this approach, the problem of modeling an n-bit QDTM, i.e., modeling an arbitrary subspace of $\mathbb{F}_2^{4n}$ is reduced to modeling an arbitrary subspace in $\mathbb{F}_2^{2n}$ for all non-zero blocks (or only for those blocks of interest in a specific problem).

Using this method, we can construct a model for any block cipher with an S-box of up to 8 bits. We applied our model to both versions of SKINNY, and we discuss the results obtained, along with a comparison to other works, in Section 5.2.

5 Applications

In this section, we detail the different applications of our extended framework. First we applied it to related-key characteristics on the AES and confirmed the average probability computed previously only under the Markov assumption. Next, in Section 5.2, we focus on SKINNY for which we conducted a fixed-key analysis of several characteristics (in both single-key and related-key model) and compared it to Peyrin and Tan's framework. Notably, we found a new constraint on a characteristic that was not detected in [PT22] therefore proving that the quasidifferential framework is more complete. We also applied the extension presented in Section 3.2 and compare it to the fixed-key approach. Finally, we applied the same extension to a differential presented in [BDD+23] and analysed the average probability of this differential, improving the complexity of two differential meet-in-the-middle attacks associated to it.

5.1 Related-key differential characteristics on AES

At first sight, it is not clear what kind of patterns in a related-key characteristic lead to an impossibility on the AES especially if the impossibility is due to the key schedule. Thus, we begin this section by giving an example of such impossible related-key characteristic on AES-128 which was successfully detected by our model. In order to explain this key constraint we use the two sets $\mathcal{X}_{DDT}$ and $\mathcal{Y}_{DDT}$, that respectively correspond to the set of input values and output values of a given differential transition through an S-box S:

$$\mathcal{X}_{DDT}(\Delta_{in}, \Delta_{out}) :- \{a \mid a \in \mathbb{F}_2^n, S(a) \oplus S(a \oplus \Delta_{in}) = \Delta_{out}\}$$
$$\mathcal{Y}_{DDT}(\Delta_{in}, \Delta_{out}) :- \{S(a) \mid a \in \mathbb{F}_2^n, S(a) \oplus S(a \oplus \Delta_{in}) = \Delta_{out}\}$$

Example 6. In this example, we detail one related-key characteristic on two rounds of the AES that is impossible due to the key schedule. This characteristic is illustrated in Figure 6.

First, the possible values for the sole active cell after the MixColumns operation (denoted x) correspond precisely to the outputs of the MixColumns transformation applied to all columns consisting of valid outputs from the first S-box layer. In other words:

$$x \in MC\Big(\big\{(x_0, x_1, x_2, x_3) | x_i \in \mathcal{Y}_{DDT}(a_i, b_i)\big\}\Big)$$

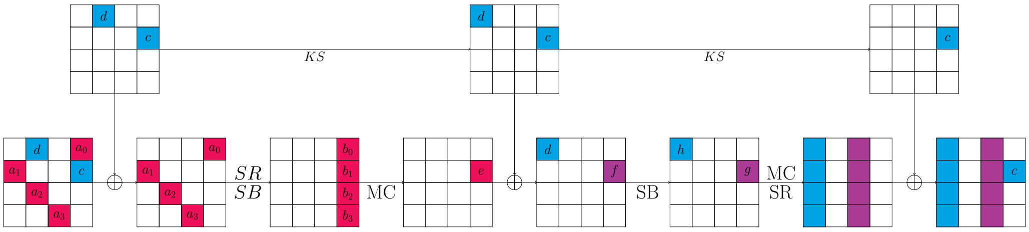


Figure 6: Impossible related-key characteristic on `AES-128` with: $(a_0, a_1, a_2, a_3, b_0, b_1, b_2, b_3)$ = `(0x94, 0x1c, 0x45, 0xd6, 0x41, 0xff, 0xd, 0x95)` and (c, d, e, f, g, h) = `(0x6, 0xc, 0x26, 0x20, 0xd4, 0x9d)` in hexadecimal notation.

Secondly, $x \oplus k$ must be a valid input for the transition through the second S-box layer. Meaning that:

$$x \oplus k = y, y \in \mathcal{X}_{DDT}(f, g)$$

Through enumeration, the possible values of k are reduced to just 128. However, k must also be a valid input for the transition $c \rightarrow d$ through the S-box in the key schedule. This requirement is incompatible with the constraints arising from the plaintext, thereby creating an impossibility. Our model successfully identified this incompatibility using only two trails, whose correlations summed to zero.

In practice, cryptanalysts search for high-probability characteristics. Patterns such as the one described in Example 6 are unlikely to result in such characteristics. Typically, high-probability related-key characteristics would be such that the differences on the key "desactivate" most of the active S-boxes in the data path. This condition is not met in our example, where f is non-zero. In other words, optimal related-key characteristics on the `AES` are really sparse. This is, we believe, the reason why we did not detect any impossibilities on any of the optimal related-key characteristics provided in the literature that we analysed. Indeed, in order to evaluate our new technique for computing the EDP of a differential characteristic while accounting for the key schedule, as well as to assess the efficiency of our model for 8-bit S-boxes we accurately calculated the average expected probability of several related-key differential characteristics of all three variants of the `AES`. Specifically, we analyzed the differential characteristics presented in [BKN09], [FJP13], and [GLMS18]. For each of these characteristics, it took only a few minutes to compute their average expected probability with subkeys generated by the key schedule. As a result, we conclude that the differential probabilities computed under the Markov assumption match the corresponding expected differential probabilities. This result is significant because, to the best of our knowledge, it is the first time that the validity of these characteristics has been verified when taking the key schedule into account.

5.2 SKINNY

We then applied our model to several differential characteristics previously analyzed in [PT22]. Specifically, for `SKINNY-64`, we selected Tables 5, 6, 7, and 8 from [DDH+21] and Table 4 from [PT22]. For `SKINNY-128`, we used Tables 9 and 10 from [DDH+21] and Table 11 from [AST+17]. Most of those characteristics are in the related-key model, the application of the quasidifferential framework in this model is discussed in Section 3.

Recall that we refer to the correlation of the quasidifferential trails with all masks being zero by the term optimal correlation and we point out that the optimal correlation is the greatest correlation (in absolute value) possible for any quasidifferential trail for the same associated differential characteristic. Indeed, from Equation (2) it follows that for any $u, a, v, b \in \mathbb{F}_2^n, D^F_{(v,b),(u,a)} \leq D^F_{(0,b),(0,a)}$. We detail the different modelings that we implemented and explain the improvements that each of them brought.

5.2.1 Finding optimal quasidifferential trails

Initially, we focused on quasidifferential trails with optimal correlation to determine what could be achieved using only these trails and Theorem 3.2. Another motivation for this approach was the difficulty of enumerating and analyzing all quasidifferential trails for a given differential characteristic on SKINNY in most of our case studies. First, when there is a large number of quasidifferential trails, the model becomes slow and may fail to terminate within a reasonable time. Second, after the model terminates, we sum the correlations of all the quasidifferential trails identified, resulting in a multivariate expression dependent on specific bits of the key. Evaluating this expression for every possible key bit combination, to check when the probability is zero, requires exponential time in the number of key bits involved. This motivated us to restrict the model to search only for optimal quasidifferential trails.

To efficiently find these trails, we introduced the following constraint into the model: if an S-box is inactive in the differential trail, it should also be inactive for the masks. This is based on the observation that $D^{SB}_{(v,0),(u,0)} < D^{SB}_{(0,0),(0,0)}$ for any $(u, v) \in \mathbb{F}_2^n$ where $(u, v) \neq (0, 0)$. Thus, if the masks are non-zero when the difference is zero, the correlation of the quasidifferential trail will be strictly lower than the optimal correlation.

Another optimization for finding optimal trails involves modifying the Quasidifferential Transition Matrix (QDTM) model for each block (corresponding to a differential transition $a \to b$) to retain only masks satisfying $D^{SB}_{(v,b),(u,a)} = D^{SB}_{(0,b),(0,a)}$. Specifically, instead of modeling the set

$$\mathcal{P}_{b,a} = \{(b\|v\|a\|u) \mid u, v \in \mathbb{F}_2^n, D^{SB}_{(v,b),(u,a)} \neq 0\},$$

we model the subset

$$\mathcal{P}'_{b,a} = \{(b\|v\|a\|u) \mid u, v \in \mathbb{F}_2^n, D^{SB}_{(v,b),(u,a)} = D^{SB}_{(0,b),(0,a)}\}.$$

Using this approach, we were able to identify all quasidifferential trails with optimal correlation for each differential trail analyzed.

After finding these quasidifferential trails, we sum their correlations and evaluate the resulting expression. If the expression equals zero for specific round key bit values, Theorem 3.2 guarantees that the fixed-key probability of the characteristic is zero. We then reverse the key schedule to detect additional incompatibilities.

At this stage, we obtained results for both SKINNY-64 and SKINNY-128. For some characteristics we did not find same results as in [PT22]. In fact, for all those characteristics our first model was always detecting fewer constraints than [PT22]. This means that some key constraints were not solely explained by optimal quasidifferential trails and that it is required to consider trails with a lower correlation to explain those constraints. An example of such a constraint along with its explanation is given in Section 5.2.3.

5.2.2 Considering sub-optimal quasidifferential trails

In order to grasp more constraints we modeled all the coefficients of the QDTM instead of only the ones that granted optimal correlation. Besides, we tried different approaches to further restrict the model. First we added an upper bound on the number of reactivated S-boxes, where a reactivated S-box is an S-box that is active for masks while the corresponding S-box is inactive for differences. That way, we found more trails, some of them having sub-optimal correlation, and by using Theorem 2 we obtain a better approximation of the fixed-key probability and the associated key space. Notably, on the 5-round differential presented in Table 4 of [PT22] we were able to exhaust all quasidifferential trails and we obtained the same key distribution as Peyrin and Tan, thus showing that anything that is detected by Peyrin and Tan's framework is also detected by Beyne and Rijmen's framework. We could also find the exact same results on the remaining characteristics of SKINNY-64. However, when applying this model on SKINNY-128 for some characteristics controlling the number of reactivated S-boxes was insufficient.

The limitation of our first approach was mainly due to the fact that setting an upper bound on reactivated S-boxes is not accurate enough and the model has no control over the quality (in terms of correlations) of the solutions that it finds. Thus, we decided to incorporate the computation of the absolute value of the correlation directly into the model. Before, discussing how we modified the model we describe an approach that was used for characteristics on which our approach is based. In 2017, Abdelkhalek *et al.* introduced a method to add the computation of the probability of a differential directly into a MILP model that searched for high-probability characteristics. The method consisted in dividing the DDT into several sub-DDTs depending on the value of the coefficients.

Definition 7 (pb-DDT, [AST+17])**.** For a given S-box and its DDT, if the probability of entries in the DDT is pb, the corresponding entry of the pb-DDT is 1. Otherwise, entries of the pb-DDT are 0.

The DDT can then be expressed as a weighted sum of pb-DDTs, with a set of inequalities computed for each pb-DDT. The approach for a single S-box is as follows: a binary variable Q_{pb} is introduced for each possible probability in the DDT. This variable determines whether the set of inequalities corresponding to pb is active. Additionally, a constraint is imposed to ensure that only one set of inequalities can be active at a time. The base-2 logarithm of the probability for a single S-box is then calculated as:

$$\sum_{pb} \log_2(pb) \times Q_{pb}.$$

This technique can clearly be extended to compute correlations for quasidifferential trails, which is precisely what we implemented. Specifically, we computed several sets of inequalities, each corresponding to different correlation values of coefficients, for every block of the QDTM that needed to be modeled. This allowed us to set a lower bound on the correlation for the quasidifferential trails we aimed to identify. However, even with this feature, we were unable to obtain convincing results for the characteristics listed in Tables 9 and 10 of [DDH+21]. While our model completes its computations within a few hours, the evaluated probability expression yields negative values for certain combinations of key bits, indicating that our lower bound on correlation was overly restrictive. Relaxing this bound would allow the discovery of additional trails to compensate for these negative values. Unfortunately, when we attempted to relax the bound, the model required significantly more time and produced an overwhelming number of trails, making the results impractical to process.

The problem of finding non-zero correlation quasidifferential trails for a given differential characteristic intuitively appears similar to exhausting a cluster of characteristics for a given differential. However, there is a fundamental difference between these two problems. While stopping the exploration of a cluster of differential characteristics below a specified probability threshold has no significant drawbacks, stopping the search for quasidifferential trails at a certain correlation bound can result in an incomplete or inaccurate formula. This issue is intrinsic to quasidifferential trails and not specific to our model.

In Section 6.2 of [BR22], Beyne and Rijmen argue that lower-correlation trails are unlikely to significantly affect the overall probability and, if they do, the effect is limited to a small subset of keys. Our experiments clearly show that this assumption does not hold for some characteristics. Identifying a criterion for quasidifferential trails that ensures a reliable formula remains an open problem.

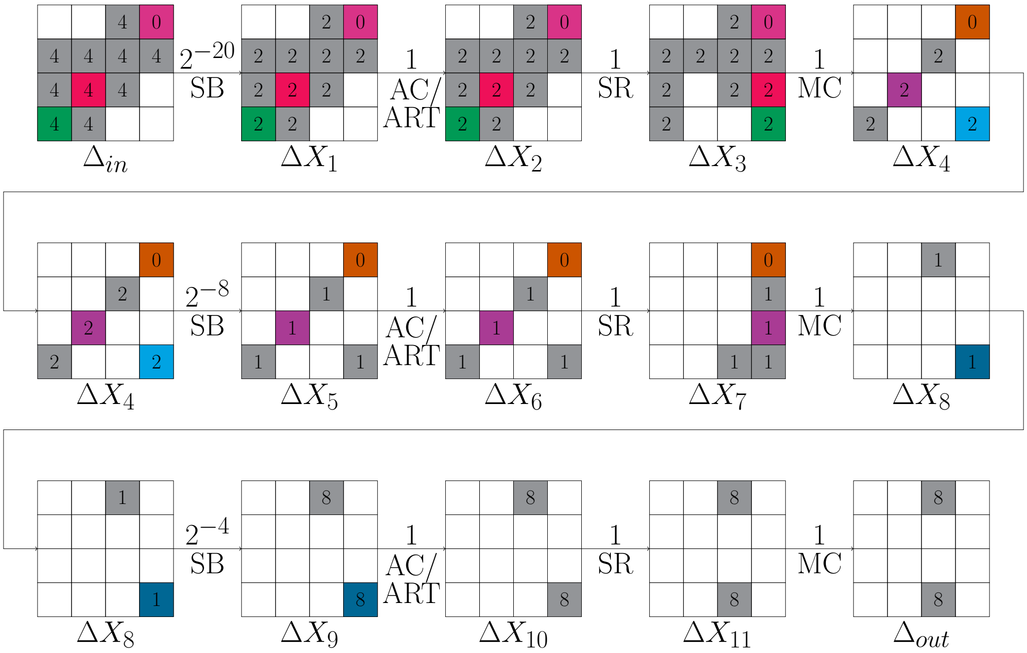


Figure 7: 3-round differential characteristic on SKINNY-64

This challenge underscores the need for the analysis we conducted using our new extension presented in Section 5.2.4. Through this approach, we achieved convincing results for both SKINNY-64 and SKINNY-128.

5.2.3 Finding more key dependencies

Furthermore, when running our model on a reduced version of the 7-round characteristic presented in Table 5 of [DDH+21], we found new constraints that were not detected by Peyrin and Tan's framework. Indeed, we ran Peyrin and Tan's tool on this characteristic restricted to 3 rounds and it detected only one linear constraint which reduced the key space to 25% and the probability was said to be 2^{-30} for all valid keys. However, when using our tool we found that the key space was only 18.75% and the probability was 2^{-29} for one third of valid keys and 2^{-30} for the rest. In order to verify the validity of our results we drew 1000 keys among keys having a claimed probability of 2^{-29} and tried on 2^{32} pairs to see how many among them were valid. We found a mean of 7.998 pairs which corroborates the claimed probability and therefore shows that the approach of [PT22] is less complete than the framework in [BR22].

Next, we detail this constraint and explain why it wasn't detected in [PT22]. The characteristic is illustrated in Figure 7. Cells that are inactive and not interesting in the differential trail are left empty, active but not interesting ones are in grey and in different colors the ones that lead to the new constraint. Looking at the cells (0,3) and (3,3) after the first MixColumns, it follows that:

$$\begin{cases} t = x \oplus y \oplus (z \oplus tk^0_{0,3}), \\ w = y \oplus (z \oplus tk^0_{0,3}). \end{cases}$$

where $tk^0_{0,3}$ is the first round tweakey cell at position (0,3), $x, y \in \mathcal{Y}_{DDT}(4,2), z \in \mathcal{Y}_{DDT}(0,0)$, $t \in \mathcal{X}_{DDT}(0,0)$ and $w \in \mathcal{X}_{DDT}(2,1)$. Thus, $t = x \oplus w$ and by enumerating, $t \in \{0,2,8,10\}$. Then, if we look at the cell (3,3) after the second MixColumns we have:

$$v = (\mathrm{SB}(t) \oplus tk^1_{0,3}) \oplus u.$$

where $tk^1_{0,3}$ is the second round tweakey cell at position (0,3), $u \in \mathcal{Y}_{DDT}(2,1)$ and $v \in \mathcal{X}_{DDT}(1,8)$. From, the first round we know that SB(t) is constrained so the possible values for $tk^1_{0,3}$ are $\{0,1,2,3,4,5,6,7,8,9,12,13\}$ with twice more solutions for the values $\{8,9,12,13\}$. This explains why one third of the valid keys has a probability of 2^{-29} instead of 2^{-30} and why one sixteenth of the keys has a zero probability. The reason why Peyrin and Tan's framework didn't detect this constraint is due to the fact that z is not constrained so their algorithm considered that the value of t is also not constrained, which doesn't take into account the possibility that this variable may vanish when combined with some other cell.

5.2.4 Computing the expected average probability

Finally, to evaluate the validity of the differential characteristics on SKINNY-128 that we could not satisfactorily analyze using a fixed-key approach, we applied the extension described in Section 3.2 to compute the expected average probability of these characteristics. For these cases, our model identified significantly fewer trails than expected given the complexity of the problem. However, we did not find any additional impossible characteristics compared to [PT22]. In Table 1 we provide a summary of the obtained results and compare them to the results of [PT22]. Our model efficiently detected the same impossible characteristics and for the first time, we are able to obtain an accurate estimation of the average probability, even for SKINNY-128 characteristics. Also, note that for some characteristics the average probability is higher than expected. For example, in the characteristic for SKINNY-128-256, the probability of transitioning through an S-box in round 6 is twice as high when it is known that the pair successfully transitioned through the previous S-box layer.

Table 1: Analysis of SKINNY Characteristics. *Stated prob.* refers to the probability provided by the respective authors. *Average prob.* represents the probability averaged over all plaintexts and keys, computed using our extension, with the number of trails found by our model shown in parentheses. *Key space* refers to the proportion of valid keys estimated by [PT22]. *Prob. range* indicates the estimated probability (or range of probabilities) for valid keys, as per [PT22]. Values marked with (E) were estimated experimentally.

SKINNY	Rounds	Stated prob.	Average prob.	[PT22]		Source
				Key Space	Prob. Range	
64-64	7	2^{-52}	2^{-52} (1)	2^{-6}	2^{-46}	Table 5 [DDH+21]
	10	2^{-46}	0 (8)	0	—	Table 6 [DDH+21]
64-128	13	2^{-55}	2^{-55} (1)	2^{-4}	2^{-51}	Table 7 [DDH+21]
	5	2^{-44}	2^{-44} (4)	Not given	$2^{-39} - 2^{-35.415}$	Table 4 [PT22]
64-192	15	2^{-54}	2^{-54} (1)	$2^{-6.19}$	$2^{-48} - 2^{-47}$	Table 8 [DDH+21]
128-128	13	2^{-123}	0 (16)	0	—	Table 11 [AST+17]
	14	2^{-120}	$\mathbf{2^{-119.05}}$ **(44)**	$2^{-7.66}$	$2^{-122.39} - 2^{-106.88}$ (E)	Table 9 [DDH+21]
128-256	16	$2^{-127.66}$	$\mathbf{2^{-126.41}}$ **(26)**	$2^{-6.11}$	$2^{-133.80} - 2^{-112.15}$(E)	Table 10 [DDH+21]

5.3 Probability of the differential used in [BDD+23] and [AKM+24]

In [BDD+23], Boura *et al.* introduced a novel cryptanalysis technique called the *Differential Meet-in-the-Middle* attack. This method combines a differential distinguisher with a meet-in-the-middle procedure to construct pairs that satisfy both the input and output conditions of a differential characteristic. Using this technique, they presented the first attack on SKINNY-128-384 reduced to 25 rounds in the single-key model.

Their attack relies on the truncated differential characteristic illustrated in Figure 8. Specifically, they demonstrated that by fixing the differences of the active bytes to 0x32 at the input and 0x64 at the output, the probability of the differential exceeds $2^{-116.5}$. An improvement of this attack, relying on the same differential, was later proposed by Ahmadian *et al.* in [AKM+24].

The complexity of both attacks directly depends on the probability of this differential. Therefore, an accurate evaluation of this probability is essential for reliably estimating the complexity of these attacks. In [BDD+23], the authors identified a significant number of differential characteristics with the same input and output differences that exhibit sufficiently high probabilities. Specifically, they found 2048 characteristics with probability 2^{-131}, 10 240 with 2^{-132}, 28 672 with 2^{-133}, and 73 728 with 2^{-134}.

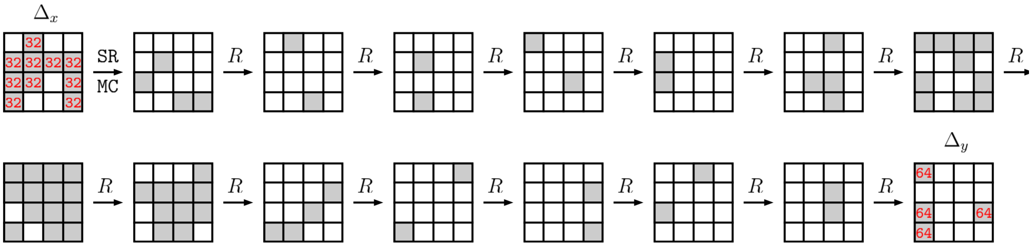


Figure 8: Truncated differential trail of the attack on 25-round SKINNY-128-384 [BDD+23]

The probabilities of all these characteristics were computed using the classical Markov (independence) assumption. However, given recent works, notably [PT22] suggesting that many differential characteristics for SKINNY may be invalid, we considered it important to verify the validity of the characteristics underlying the attacks of [BDD+23] and [AKM+24]. To this end, we calculated the average probability of the 114 688 differential characteristics using our extended quasidifferential framework (described in Section 3.2). This computation was performed on a 128-core server (using only 32 of them) and required 18 hours to complete.

Our analysis revealed that a non-negligible portion of the characteristics are indeed invalid— 63 488 characteristics to be precise. Surprisingly, however, the probabilities of the remaining valid characteristics turned out to be higher than previously estimated. As a result, we demonstrate that the overall probability of the differential is higher than $2^{-113.6}$. Hence, the actual time complexities of both attacks given in [BDD+23] and [AKM+24] are improved by a factor $2^{2.9}$, while both their data and memory complexities are decreased by a factor $2^{1.4}$.

6 Conclusion

In this work, we proposed several extensions to the quasidifferential framework, along with an efficient MILP model capable of effectively handling ciphers with 8-bit S-boxes. Among our contributions, we consider the most significant to be the adaptation of the framework to compute the expected differential probability of a characteristic while accounting for the key schedule. This allowed us to verify, in a very short time, the validity of thousands of characteristics on SKINNY-128. Using this method, we also demonstrated that the expected differential probability of all known optimal characteristics for all variants of the AES remains the same, whether the round keys are considered independent or derived from the key schedule. The ability of our approach to efficiently incorporate the key schedule in these computations provide block cipher designers with a practical tool for selecting a key or tweakey schedule that maximizes the cipher's resistance to (related-key) differential attacks.

Acknowledgments

This research is partially supported through the France 2030 program under grant agreement No. ANR-22-PECY-0010, by the French Agence Nationale de la Recherche through the OREO project under Contract ANR-22-CE39-0015 and the SWAP project under Contract ANR-21-CE39-0012, and by the European Research Council (ERC) project 101125450 (SoBaSyC).

References

[AK19] Ralph Ankele and Stefan Kölbl. Mind the gap - A closer look at the security of block ciphers against differential cryptanalysis. In Carlos Cid and Michael J.: Jacobson, Jr., editors, *SAC 2018*, volume 11349 of *LNCS*, pages 163–190. Springer, Cham, August 2019.

[AKM+24] Zahra Ahmadian, Akram Khalesi, Dounia M'foukh, Hossein Moghimi, and María Naya-Plasencia. Improved differential meet-in-the-middle cryptanalysis. In Marc Joye and Gregor Leander, editors, *EUROCRYPT 2024, Part I*, volume 14651 of *LNCS*, pages 280–309. Springer, Cham, May 2024.

[AST+17] Ahmed Abdelkhalek, Yu Sasaki, Yosuke Todo, Mohamed Tolba, and Amr M. Youssef. MILP modeling for (large) s-boxes to optimize probability of differential characteristics. *IACR Trans. Symm. Cryptol.*, 2017(4):99–129, 2017.

[BC20] Christina Boura and Daniel Coggia. Efficient MILP modelings for sboxes and linear layers of SPN ciphers. *IACR Trans. Symm. Cryptol.*, 2020(3):327–361, 2020.

[BDBN23] Christina Boura, Nicolas David, Rachelle Heim Boissier, and María Naya-Plasencia. Better steady than speedy: Full break of SPEEDY-7-192. In Carmit Hazay and Martijn Stam, editors, *EUROCRYPT 2023, Part IV*, volume 14007 of *LNCS*, pages 36–66. Springer, Cham, April 2023.

[BDD+23] Christina Boura, Nicolas David, Patrick Derbez, Gregor Leander, and María Naya-Plasencia. Differential meet-in-the-middle cryptanalysis. In Helena Handschuh and Anna Lysyanskaya, editors, *CRYPTO 2023, Part III*, volume 14083 of *LNCS*, pages 240–272. Springer, Cham, August 2023.

[BKN09] Alex Biryukov, Dmitry Khovratovich, and Ivica Nikolic. Distinguisher and related-key attack on the full AES-256. In Shai Halevi, editor, *CRYPTO 2009*, volume 5677 of *LNCS*, pages 231–249. Springer, Berlin, Heidelberg, August 2009.

[BN24] Tim Beyne and Addie Neyt. Note on the cryptanalysis of speedy. Cryptology ePrint Archive, Report 2024/262, 2024.

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

[BS91] Eli Biham and Adi Shamir. Differential cryptanalysis of DES-like cryptosystems. In Alfred J. Menezes and Scott A. Vanstone, editors, *CRYPTO'90*, volume 537 of *LNCS*, pages 2–21. Springer, Berlin, Heidelberg, August 1991.

[DDH$^+$21] Stéphanie Delaune, Patrick Derbez, Paul Huynh, Marine Minier, Victor Mollimard, and Charles Prud'homme. Efficient methods to search for best differential characteristics on SKINNY. In Kazue Sako and Nils Ole Tippenhauer, editors, *ACNS 21International Conference on Applied Cryptography and Network Security, Part II*, volume 12727 of *LNCS*, pages 184–207. Springer, Cham, June 2021.

[DR07] Joan Daemen and Vincent Rijmen. Plateau characteristics. *IET Inf. Secur.*, 1(1):11–17, 2007.

[FJP13] Pierre-Alain Fouque, Jérémy Jean, and Thomas Peyrin. Structural evaluation of AES and chosen-key distinguisher of 9-round AES-128. In Ran Canetti and Juan A. Garay, editors, *CRYPTO 2013, Part I*, volume 8042 of *LNCS*, pages 183–203. Springer, Berlin, Heidelberg, August 2013.

[GLMS18] David Gérault, Pascal Lafourcade, Marine Minier, and Christine Solnon. Revisiting AES related-key differential attacks with constraint programming. *Inf. Process. Lett.*, 139:24–29, 2018.

[Hey20] Howard M. Heys. Key dependency of differentials: Experiments in the differential cryptanalysis of block ciphers using small S-boxes. Cryptology ePrint Archive, Report 2020/1349, 2020.

[Knu93] Lars R. Knudsen. Iterative characteristics of DES and s^2-DES. In Ernest F. Brickell, editor, *CRYPTO'92*, volume 740 of *LNCS*, pages 497–511. Springer, Berlin, Heidelberg, August 1993.

[LMM91] Xuejia Lai, James L. Massey, and Sean Murphy. Markov ciphers and differential cryptanalysis. In Donald W. Davies, editor, *EUROCRYPT'91*, volume 547 of *LNCS*, pages 17–38. Springer, Berlin, Heidelberg, April 1991.

[LS22] Ting Li and Yao Sun. SuperBall: A new approach for MILP modelings of boolean functions. *IACR Trans. Symm. Cryptol.*, 2022(3):341–367, 2022.

[PT22] Thomas Peyrin and Quan Quan Tan. Mind your path: On (key) dependencies in differential characteristics. *IACR Trans. Symm. Cryptol.*, 2022(4):179–207, 2022.

[SHW$^+$14a] Siwei Sun, Lei Hu, Meiqin Wang, Peng Wang, Kexin Qiao, Xiaoshuang Ma, Danping Shi, Ling Song, and Kai Fu. Towards finding the best characteristics of some bit-oriented block ciphers and automatic enumeration of (related-key) differential and linear characteristics with predefined properties. Cryptology ePrint Archive, Report 2014/747, 2014.

[SHW$^+$14b] Siwei Sun, Lei Hu, Peng Wang, Kexin Qiao, Xiaoshuang Ma, and Ling Song. Automatic security evaluation and (related-key) differential characteristic search: Application to SIMON, PRESENT, LBlock, DES(L) and other bit-oriented block ciphers. In Palash Sarkar and Tetsu Iwata, editors, *ASIACRYPT 2014, Part I*, volume 8873 of *LNCS*, pages 158–178. Springer, Berlin, Heidelberg, December 2014.

[Sun21] Yao Sun. Towards the least inequalities for describing a subset in ${Z_2}^n$. Cryptology ePrint Archive, Report 2021/1084, 2021.

[SW23] Ling Sun and Meiqin Wang. SoK: Modeling for large S-boxes oriented to differential probabilities and linear correlations. *IACR Trans. Symm. Cryptol.*, 2023(1):111–151, 2023.

A Quasidifferential transition matrix blocks

We provide here the two blocks of the Quasidifferential Transition Matrix (QDTM) for the S-box of SKINNY, which are necessary to follow the computations in the toy cipher examples. The two coefficients involved in the computation are highlighted in bold and marked in red for clarity.

$$2^{-3}\times \overset{u}{\begin{pmatrix}
1 & 0 & -1 & 0 & -1 & 0 & 1 & 0 & 0 & -1 & 0 & 1 & 0 & 1 & 0 & -1\\
\color{red}{\mathbf{-1}} & 0 & 1 & 0 & 1 & 0 & -1 & 0 & 0 & 1 & 0 & -1 & 0 & -1 & 0 & 1\\
-1 & 0 & 1 & 0 & 1 & 0 & -1 & 0 & 0 & 1 & 0 & -1 & 0 & -1 & 0 & 1\\
1 & 0 & -1 & 0 & -1 & 0 & 1 & 0 & 0 & -1 & 0 & 1 & 0 & 1 & 0 & -1\\
0 & -1 & 0 & 1 & 0 & 1 & 0 & -1 & 1 & 0 & -1 & 0 & -1 & 0 & 1 & 0\\
0 & 1 & 0 & -1 & 0 & -1 & 0 & 1 & -1 & 0 & 1 & 0 & 1 & 0 & -1 & 0\\
0 & 1 & 0 & -1 & 0 & -1 & 0 & 1 & -1 & 0 & 1 & 0 & 1 & 0 & -1 & 0\\
0 & -1 & 0 & 1 & 0 & 1 & 0 & -1 & 1 & 0 & -1 & 0 & -1 & 0 & 1 & 0\\
0 & 1 & 0 & -1 & 0 & -1 & 0 & 1 & -1 & 0 & 1 & 0 & 1 & 0 & -1 & 0\\
0 & -1 & 0 & 1 & 0 & 1 & 0 & -1 & 1 & 0 & -1 & 0 & -1 & 0 & 1 & 0\\
0 & -1 & 0 & 1 & 0 & 1 & 0 & -1 & 1 & 0 & -1 & 0 & -1 & 0 & 1 & 0\\
0 & 1 & 0 & -1 & 0 & -1 & 0 & 1 & -1 & 0 & 1 & 0 & 1 & 0 & -1 & 0\\
-1 & 0 & 1 & 0 & 1 & 0 & -1 & 0 & 0 & 1 & 0 & -1 & 0 & -1 & 0 & 1\\
1 & 0 & -1 & 0 & -1 & 0 & 1 & 0 & 0 & -1 & 0 & 1 & 0 & 1 & 0 & -1\\
1 & 0 & -1 & 0 & -1 & 0 & 1 & 0 & 0 & -1 & 0 & 1 & 0 & 1 & 0 & -1\\
-1 & 0 & 1 & 0 & 1 & 0 & -1 & 0 & 0 & 1 & 0 & -1 & 0 & -1 & 0 & 1
\end{pmatrix}} v$$

.

Figure 9: $D^{S}_{(v,12),(u,9)}$ for $u, v \in \mathbb{F}_2^4$

B Converting inequalities to describe $\mathcal{P}_{a,b}$

This appendix is related to Section 4.2.3. More precisely, we describe here the method to transform inequalities describing the set $\overline{\mathcal{P}_{b,a}}$ to inequalities describing the set $\mathcal{P}_{b,a}$.

We suppose that we have obtained a set $\overline{\mathcal{I}_{a,b}}$ of inequalities in $2n$ variables, which are only satisfied by elements of $\overline{\mathcal{P}_{b,a}}$ within $\mathbb{F}_2^{2n}$. Our goal is to convert these inequalities into ones defined in $4n$ variables, such that they are satisfied only by elements of $\mathcal{P}_{b,a}$. This is precisely the purpose of Proposition 2.

Proposition 2. *Let $c_0^v x_0^v + \cdots + c_{n-1}^v x_{n-1}^v + c_0^u x_0^u + \cdots + c_{n-1}^u x_{n-1}^u + d \geq 0$ be an inequality over $\mathbb{F}_2^{2n}$ satisfied by a set $\overline{\mathcal{P}} \subseteq \mathbb{F}_2^{2n}$. Let $a, b \in \mathbb{F}_2^n$ and $M = 1 + d + \sum_{i|c_i^v>0} c_i^v + \sum_{i|c_i^u>0} c_i^u$. It holds that the inequality:*

$$-M\sum_{i=0}^{n-1}\overline{x_i^b} + c_0^v x_0^v + \cdots + c_{n-1}^v x_{n-1}^v - M\sum_{i=0}^{n-1}\overline{x_i^a} + c_0^u x_0^u + \cdots + c_{n-1}^u x_{n-1}^u + d \geq 0$$

is only satisfied by the set $\{(b\|v\|a\|u) \mid (v\|u) \in \overline{\mathcal{P}}\}$ where

$$\overline{x_i^b} = \begin{cases} x_i^b \text{ if } b_i = 0 \\ (1 - x_i^b) \text{ if } b_i = 1 \end{cases} \quad \text{and } \overline{x_i^a} = \begin{cases} x_i^a \text{ if } a_i = 0 \\ (1 - x_i^a) \text{ if } a_i = 1 \end{cases}$$

$$2^{-3}\times \begin{array}{c} u \\ \left(\begin{array}{cccccccccccccccc}
1 & -1 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 1 & -1 \\
-1 & 1 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & -1 & 1 \\
0 & 0 & 0 & 0 & 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & -1 & 1 & 1 & -1 & 1 & -1 & -1 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & -1 & 1 & 1 & -1 & 1 & -1 & -1 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 & 0 & 0 & 0 & 0 \\
-1 & 1 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & -1 & 1 \\
1 & -1 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 1 & -1 \\
-1 & 1 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & -1 & 1 \\
1 & -1 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 1 & -1 \\
0 & 0 & 0 & 0 & -1 & 1 & 1 & -1 & 1 & -1 & -1 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & -1 & 1 & 1 & -1 & 1 & -1 & -1 & 1 & 0 & 0 & 0 & 0 \\
1 & -1 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 1 & -1 \\
-1 & 1 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & -1 & 1
\end{array}\right) v \end{array}$$

Figure 10: $D^S_{(v,6),(u,12)}$ for $u, v \in \mathbb{F}_2^4$

Proof. Let $x = (x^b_0, \ldots, x^b_{n-1}, x^v_0, \ldots, x^v_{n-1}, x^a_0, \ldots, x^a_{n-1}, x^u_0, \ldots, x^u_{n-1}) \in \mathbb{F}_2^{4n}$. Then $c^v_0 x^v_0 + \cdots + c^v_{n-1} x^v_{n-1} + c^u_0 x^u_0 + \cdots + c^u_{n-1} x^u_{n-1} + d < M$ by definition of M. So if there exists $i \in \{0, \ldots, n-1\}$ such that $\overline{x^a_i} = 1$ or $\overline{x^b_i} = 1$, then the inequality cannot be satisfied. However,

$$\overline{x^b_i} = \begin{cases} 0 \text{ if } b_i = x^b_i \\ 1 \text{ if } b_i \neq x^b_i \end{cases} \text{ and } \overline{x^a_i} = \begin{cases} 0 \text{ if } a_i = x^a_i \\ 1 \text{ if } a_i \neq x^a_i \end{cases}$$

Thus, the inequality can only be satisfied if $x^a_i = a_i$ and $x^b_i = b_i$ for all $i \in \{0, \ldots, n-1\}$ i.e., the set of solutions lies in $\{(b \| v \| a \| u) \mid u, v \in \mathbb{F}_2^n\}$. And when the condition is met, the inequality turns into $c^v_0 x^v_0 + \cdots + c^v_{n-1} x^v_{n-1} + c^u_0 x^u_0 + \cdots + c^u_{n-1} x^u_{n-1} + d \geq 0$ which is satisfied only by $(u, v) \in \overline{\mathcal{P}}$ which concludes the proof. □