Layout Graphs, Random Walks and the t-wise Independence of SPN Block Ciphers

Tianren Liu
Peking University
trl@pku.edu.cn

Angelos Pelecanos
UC Berkeley
apelecan@berkeley.edu

Stefano Tessaro
University of Washington
tessaro@cs.washington.edu

Vinod Vaikuntanathan
MIT CSAIL
vinodv@mit.edu

January 18, 2024

Abstract

We continue the study of t-wise independence of substitution-permutation networks (SPNs) initiated by the recent work of Liu, Tessaro, and Vaikuntanathan (CRYPTO 2021).

Our key technical result shows that when the S-boxes are *randomly and independently chosen* and kept secret, an r-round SPN with input length $n = b \cdot k$ is $2^{-\Theta(n)}$-close to t-wise independent within $r = O(\min\{k, \log t\})$ rounds for any t almost as large as $2^{b/2}$. Here, b is the input length of the S-box and we assume that the underlying mixing achieves maximum branch number. We also analyze the special case of AES parameters (with random S-boxes), and show it is 2^{-128}-close to pairwise independent in 7 rounds. Central to our result is the analysis of a random walk on what we call the *layout graph*, a combinatorial abstraction that captures equality and inequality constraints among multiple SPN evaluations.

We use our technical result to show concrete security bounds for SPNs with actual block cipher parameters and *small-input S-boxes.* (This is in contrast to the large body of results on ideal-model analyses of SPNs.) For example, for the censored-AES block cipher, namely AES with most of the mixing layers removed, we show that 192 rounds suffice to attain 2^{-128}-closeness to pairwise independence. The prior such result for AES (Liu, Tessaro and Vaikuntanathan, CRYPTO 2021) required more than 9000 rounds.

Contents

1 Introduction

The design of block ciphers like the Advanced Encryption Standard (AES) is one of the most central topics in practical cryptography. Our confidence in their security stems from decades of cryptanalysis, spanning a wide range of attacks including linear [MY92] and differential [BS91] cryptanalysis, higher-order [Lai94], truncated [Knu94] and impossible [Knu98] differential attacks, interpolation [JK97] and algebraic attacks [CP02], integral cryptanalysis [KW02], biclique attacks [BKR11], and so on. These attacks have so far failed to make a dent in the conjectured security of AES as a (fixed-parameter) pseudorandom permutation. Nonetheless, we remain very far from rigorously justifying that security actually holds. Crucially, the design methodology behind most block ciphers iterates a very weak round function (too weak to achieve any meaningful security notion). It is not clear whether it is even possible to formulate a meaningful non-tautological assumption that implies the security of a block cipher within the classical framework of provable security.

t-wise independent ciphers. Facing the above limitations, this paper continues a line of work justifying the security of block ciphers against *restricted* classes of attacks, with a focus on *substitution permutation networks* (SPNs), an important class of block ciphers that includes AES. In particular, we build on top of recent work by Liu, Tessaro, and Vaikuntanathan (LTV) [LTV21] that studies the t-wise independence of SPNs as a "catch-all" security property that prevents all t-input statistical attacks. (The notion was already studied earlier [HMMR05, BH08] for less standard block cipher constructions.)

We take a *quantitative* angle where, for a given t, we aim to know the smallest $\epsilon = \epsilon(r)$ for which an r-round SPN is $\epsilon(r)$-close to a t-wise independent permutation. The case $t = 2$ already implies, for a small enough ϵ, security against *linear* [MY92] and *differential* [BS91] attacks, which have (on their own) been the subject of hundreds of works. Similarly, security against degree-d higher-order differential attacks [Lai94] follows when $t = 2^d$.

The results from [LTV21] suffer however from two major limitations, which we aim to address here: First, they only prove *pairwise* independence of SPNs. Second, for AES-like parameters, their pairwise-independence bound effectively requires *thousands of rounds* to achieve meaningful security matching practical expectations. (Concretely, more than 9000.[1])

Our contributions, in a nutshell. In this work, we study the t-wise independence of SPNs when the S-boxes are *randomly chosen*, *independent*, and *secret*, and thus act as the actual secret keys. Unlike a number of recent works in the random S-box model (e.g., [MV12, CDK$^+$18, DKS$^+$17]), which assume the S-box inputs to be as large as the security parameter, here we target a scenario with *small-input* S-boxes (e.g., 8 bits, as in AES), which presents a unique challenge. Random S-box SPNs were for instance also studied by Baignères and Vaudenay [BV06], who quantified the linear and differential probabilities in the limit as the number of rounds goes to infinity. Here, instead, we prove *concrete* bounds for the *stronger* property of t-wise independence. A summary of our results is given in Table 1.

While it is interesting to study random S-boxes in their own right, as they have been used in actual ciphers (e.g., GOST [Nat89] and AES variants [TKKL15]), we really want to derive conclusions for block ciphers with fixed S-boxes (as [LTV21] did) from our results. An *optimistic* interpretation of our results is that random, secret, S-boxes yield a good heuristic approximation of the behavior of SPNs with a concrete S-box (e.g., the inversion map $x \mapsto x^{2^b-1}$ as in AES).

[1] LTV prove that $6r$-round AES is $2^{r-1}(0.472)^r$-close to pairwise independent, which becomes smaller than 2^{-128} for $r \geq 1528$.

	Rounds	t	**Closeness**	**Theorem**
SPN*	2	$O(1)$	$2^{-\Omega(kb)}$	Thm. 2
	2	$2^{(0.499-1/(4k))b}$	2^{-b}	Thm. 3
	$O(k)$	$2^{(0.499-1/(4k))b}$	$2^{-\Omega(kb)}$	Thm. 3 + [MPR07, KNR09a]
	$O(\log t)$	$2^{0.499b}$	$2^{-\Omega(kb)}$	Thm. 4
AES*	7	2	2^{-128}	Thm. 6
censored AES	192	2	2^{-128}	Thm. 7

Table 1: **Results for the t-wise independence of SPN* and AES*.** Here, b is the length of the input to the S-box (the word length or block size), and k is the width for SPN* (equivalently, the number of parallel S-box invocations). All of the SPN* results assume a linear mixing layer with maximum branch number. The AES* result uses the AES mixing layer, $k = 16$, $b = 8$.

But we also offer a more *pragmatic* interpretation, based on the fact that a random S-box can be approximated by the sequential composition of an actual S-box (where a key is XORed prior to each call). Our analyses in the random S-box model therefore carry over to a *concrete* block cipher which can be thought of as an SPN with a number of mixing layers removed (what we refer to as a "censored" SPN or SPN*).

We now go back to our contributions in a bit more in detail.

Substitution-permutation networks. To state our results more concretely, recall that a substitution permutation network (SPN) with *word length* b, *width* k, and r *rounds*, is defined by an invertible *substitution box* (or S-box) $S : \mathbb{F} \to \mathbb{F}$, where $\mathbb{F} = \mathbb{F}_{2^b}$, and an invertible *mixing layer* $M : \mathbb{F}^k \to \mathbb{F}^k$. (One usually focuses on *linear* mixing functions as we do in this paper.) Computation proceeds in r rounds, given input vector $\mathbf{x}^{(\mathrm{in})} = \mathbf{y}^{(0)} \in \mathbb{F}^k$ and round keys $\mathbf{k}^{(0)}, \ldots, \mathbf{k}^{(r)} \in \mathbb{F}^k$. For $i = 1, \ldots, r+1$ we compute

$$\mathbf{x}^{(i)} = \left[S\left(\mathbf{y}^{(i-1)}[1] + \mathbf{k}^{(i-1)}[1]\right), \ldots, S\left(\mathbf{y}^{(i-1)}[k] + \mathbf{k}^{(i-1)}[k]\right)\right] .$$

$$\mathbf{y}^{(i)} = M\mathbf{x}^{(i)}$$

The final output is $\mathbf{y}^{(\mathrm{out})} = \mathbf{x}^{(r+1)}$. See Figure 1 for an illustration. (Note that in this representation, the final operation is the application of S-boxes, with no further mixing. This differs from some of the literature; however, the difference is inconsequential to our results.) In an actual block cipher, one would compute the round keys from a short key via a suitable key-scheduling algorithm, but here we follow the convention from prior works of using independent keys for the analysis.

Typical choices for the above parameters are those from AES, where $k = 16$ and $b = 8$, and one should think of these when assessing whether a result is meaningful.

t-wise independence for random S-boxes. The bulk of our results will be concerned with the analysis of SPNs in a model where the S-boxes are ideal, i.e., randomly chosen and secret. In other words, we replace the step

$$\mathbf{x}^{(i)}[j] \leftarrow S(\mathbf{y}^{(i-1)}[j] \oplus \mathbf{k}^{(i-1)}[j])$$

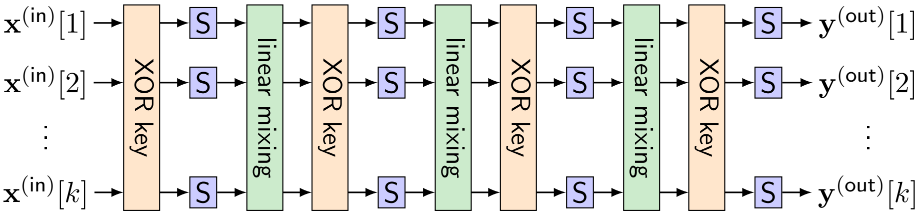
$\mathbf{x}^{(\mathsf{in})}[1]$
$\mathbf{x}^{(\mathsf{in})}[2]$
$\mathbf{x}^{(\mathsf{in})}[k]$
XOR key
S
linear mixing
$\mathbf{y}^{(\mathsf{out})}[1]$
$\mathbf{y}^{(\mathsf{out})}[2]$
$\mathbf{y}^{(\mathsf{out})}[k]$

Figure 1: Illustration of a 3-round SPN.

for $i = 1, \ldots, r+1$ and $j = 1, \ldots, k$ with

$$\mathbf{x}^{(i)}[j] \leftarrow S_j^{(i-1)}(\mathbf{y}^{(i-1)}[j])$$

where $S_j^{(i-1)}$ is a uniformly chosen random permutation on $\mathbb{F}$. Here, we can think of the S-box descriptions as part of a longer key, and following the notation from [BV06], we refer to this variant as SPN*.

Formally, we measure the proximity to t-wise independence by picking t arbitrary distinct input vectors and obtain the t output vectors processed by the r-round SPN* construction. We then give an upper bound on the statistical distance of these output vectors from t uniformly sampled, but distinct, vectors. As observed in [LTV21], such a distance bound also gives explicit concrete bounds for the linear and differential probabilities. (In particular, our result gives concrete bounds for such quantities, as opposed to [BV06] which only shows eventual convergence to a particular probability as the number of rounds goes to infinity.)

Layouts and random walks. At the core of our results is the formalization of the concept of a *layout*, which allows us to reduce the question of t-wise independence to the analysis of a random walk which is entirely defined by the mixing layer M. Concretely, if we are given a t-tuple of vectors $(\mathbf{y}_1, \ldots, \mathbf{y}_t)$, and map them to $(\mathbf{x}_1, \ldots, \mathbf{x}_t)$ by applying the same k random S-boxes to each of the vectors, we observe that the mapping respects equality and inequality constraints. For example, if $\mathbf{y}_i[j] = \mathbf{y}_{i'}[j]$ for $i \neq i'$, then $\mathbf{x}_i[j] = \mathbf{x}_{i'}[j]$. Inequalities are also similarly preserved. A t-wise *layout* I is, formally, a description of equality/inequality constraints among t k-dimensional vectors over $\mathbb{F}$. Crucially, applying random S-boxes to *any* t-tuple $(\mathbf{y}_1, \ldots, \mathbf{y}_t)$ satisfying the layout I results in a t-tuple picked *uniformly at random* from the set of *all* t-tuples that satisfy the same layout I. For the special case of $t = 2$, a layout is equivalent to an *activity pattern* formulated and studied in the AES literature [AES01].

This means in particular that the evaluation of an r-round SPN* on t inputs corresponds to taking r random steps on the *layout* graph. We start with an arbitrary layout I_0, and step $i = 1, \ldots, r$ consists of:

- Picking a random t-tuple $(\mathbf{x}_1^{(i)}, \ldots, \mathbf{x}_t^{(i)})$ that lies in layout I_{i-1};
- Compute $\mathbf{y}_j^{(i)} = M\mathbf{x}_j^{(i)}$ for all $j = 1, \ldots, t$; and
- Set I_i to be the (unique) layout satisfied by $(\mathbf{y}_1^{(i)}, \ldots, \mathbf{y}_t^{(i)})$.

The convergence of this walk to the distribution over layouts induced by a uniformly sampled t-tuple of distinct vectors directly yields t-wise independence of the r-round SPN*. For the case

$t = 2$, this random walk was also described in [BV06] without any explicit convergence guarantees, which we provide here.

We provide a careful analysis of this random walk by first characterizing the transition probability of going from a layout I to a layout J and then derive an upper bound on the distance from the stationary distribution after one single step, provided we start from a nice enough layout, i.e., one that does not induce too many collisions. Then, very roughly, one shows that a nice layout is reached in one round with very high probability. We use this analysis to derive a number of theorems, which all assume that the mixing layer achieves maximum branch number, i.e., for all $\mathbf{x} \in \mathbb{F}^k \setminus \{0\}$, we have $\mathsf{wt}(\mathbf{x}) + \mathsf{wt}(M\mathbf{x}) \geq k + 1$, where $\mathsf{wt}(\cdot)$ denotes Hamming weight, i.e., the number of non-zero components.

Our first two theorems give the smallest ϵ depending on whether t is small or large.

> **Theorem 2.** 2-round SPN* is ε-close to t-wise independent, for $\varepsilon = \frac{t^2 \cdot 2^{k+1}}{(2^b)^{k/(2t)}} + t \cdot \left(\frac{8 \cdot t^3}{2^b}\right)^{k/2}$.

> **Theorem 3.** For any $\alpha \in (0, 1]$, 2-round SPN* is ε-close to t-wise independent, where $\varepsilon = \frac{t^2}{\alpha \cdot 2^b} + t \cdot \left(\frac{(2t)^{2-\alpha}}{(2^b)^{1-\alpha}}\right)^k$.

A standard goal is to make ϵ equal $2^{-\Omega(k \cdot b)}$, as $n = k \cdot b$ is the input length of the SPN, and the first theorem implies that for small constant $t = O(1)$, we achieve distance $2^{-\Omega(n)}$ already after two rounds. In contrast, by picking the suitable α, the second theorem allows t to become almost as large as $2^{b/2}$ (concretely, we require $t < 2^{(0.499-1/(4k))b}$, which is as large as 14 for AES-like parameters), but only gives $\epsilon = 2^{-\Omega(b)}$. However, one can then amplify this using existing amplification results [MPR07, KNR09a] to achieve $\epsilon = 2^{-\Omega(bk)}$ after $2k$ rounds.

We also show an alternative theorem that also yields $\epsilon = 2^{-\Omega(bk)}$, but this time using $O(\log t)$ rounds, instead of $O(k)$. This follows from the following.

> **Theorem 4.** Let $t = 2^r$. Then, r-round SPN* is ε-close to 2^r-wise independent for $\varepsilon = \frac{t \cdot 2^{\frac{11}{4}}}{1 - 2^{-\frac{k}{4}}} \cdot \left(\frac{8 \cdot t^2}{2^b}\right)^{k/4}$ if $k > 4$.

The case of AES. The specific case of AES is interesting because its mixing layer does not achieve the maximal branch number. One could in fact extend some of our techniques above to a more relaxed branch number. However, we give a more precise analysis of a variant of AES with random S-boxes which, unlike the above SPN*, uses the *actual* AES mixing layer (alternating the ShiftRows and MixColumn operations). It also sets $k = 16$ and $b = 8$. We refer to this variant as AES*. We show that AES* is 2^{-128}-close to pairwise independent already for *seven* rounds. To achieve these results, we combine experimental computations with our random walk framework. We note that this result could have been obtained computationally also using results from [BV05], in particular their description of the random walk on layouts for the special case of AES* and $t = 2$. (Their description is however not sufficient to yield the results in the other sections of this paper, nor do they actually carry out the computation, or target a security property as strong as pairwise-independence.)

Concrete S-boxes and censored SPNs. For the special case of pairwise independence, one can easily transform our results for random S-boxes into results for concrete S-boxes if we are willing to replace the application of a *single* random S-box $S_j^{(i)}$ with the repeated application of the AES S-box (namely, the patched inversion function over $\mathbb{F}$) alternated with the addition of a key value prior to each S-box call. We refer to the resulting cipher as *censored SPN* (or *censored* AES), because it is equivalent to an SPN where a fraction of mixing layers have been removed (i.e., "censored"). We give a censored variant of AES which is 2^{-128}-close to pairwise independent after 192 rounds. We conjecture that 192-round of AES itself is also 2^{-128}-close to pairwise independent, i.e., the censoring mixing layers never increases security.

This should be contrasted with [LTV21], which shows that AES is 2^{-128}-close to pairwise independent after (more than) 9000 rounds.

1.1 Related Work: The "Large" S-box Model

A number of works [MV12, CDK+18, DKS+17] have considered SPNs with random S-boxes when the input length b is large (i.e., it can be thought of as the security parameter), and aims to prove an r-round SPN to be a (strong) pseudorandom permutation. Miles and Viola [MV12] deal with *secret S*-boxes (as we do here), whereas [CDK+18, DKS+17] consider a single public S-box (accessible as a random oracle) which is then keyed within the construction. (But clearly, this implies an analysis in a model where the S-box is secret.) These works fit within the bigger scope of a long line of works [EM97, BKL+12, LPS12, ABD+13, CLL+14, CS14, LS15, CS15, Tes15, GL15, FP15, DSSL16, CHK+16, DS16, HT16, DSST17]) analyzing block cipher constructions in ideal models. A recent paper by Dodis, Karthikeyan, and Wichs [DKW22] then suggests conjectures under which these large S-box analyses could imply security in the small S-box regime (for full pseudorandomness).

While the result is not explicitly stated, one can, in fact, apply the toolkit from [CDK+18], which in turn relies on the H-coefficient method [Pat08], to show that a 1-round SPN is ϵ-close to t-wise independent for $\epsilon = O(kt^2/2^b)$. For $b = 8$ and $k = 16$, one might hope to achieve $\epsilon = 1/2$ for $t = 2$ (and in turn, this can be boosted using [MPR07]), but the involved constants prevent that. In addition, we observe that this bound has the unnatural feature that it *degrades* as a function of the width parameter k, which is exactly what we show *not to be the case.* Our results adopt completely different techniques, that rely on the analysis of random walks on the layout graph, and indeed also indicate an improvement of the achievable ϵ as k grows, as intuition would suggest.

While (almost) t-wise independent permutations can be constructed in many other ways (see, e.g. [KNR09b]), that is not the point of this paper. Our goal is to analyze natural constructions, in this case following the substitution-permutation paradigm, which are *provably* almost t-wise independent and *plausibly* pseudorandom.

1.2 Technical Overview

In this overview, we briefly explain how our technique works in the special case of 2-wise (or pairwise) independence of SPN* (i.e., SPN with random S-boxes). A more detailed analysis of the pairwise setting can be found in Section 4. The more involved analysis of the general t-wise setting follows the same framework, and is presented in Section 5. Concrete bounds for censored AES are given in Section 6.

Differences and layouts. As we only consider two inputs, we can follow the standard differential cryptanalysis approach of working with *differences.* For any input difference $\mathbf{x}_\Delta^{(\mathsf{in})} = \mathbf{x}_1^{(\mathsf{in})} - \mathbf{x}_2^{(\mathsf{in})}$,

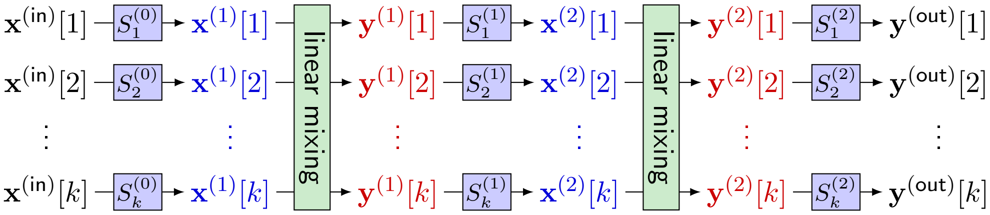

Figure 2: Illustration of a 2-round SPN* Network. Each S-box is a uniformly random permutation from $\mathbb{F}$ to $\mathbb{F}$. These S-boxes form the key of the SPN* network.

we need to show that the corresponding distribution of the output difference $\mathbf{y}_\Delta^{(\mathsf{out})} = \mathbf{y}_1^{(\mathsf{out})} - \mathbf{y}_2^{(\mathsf{out})}$ is close to uniform. We consider a two-round SPN*, so we can define analogously differences $\mathbf{x}_\Delta^{(1)}$, $\mathbf{y}_\Delta^{(1)}$, $\mathbf{x}_\Delta^{(2)}$, $\mathbf{y}_\Delta^{(2)}$, and $\mathbf{y}_\Delta^{(\mathsf{out})}$. See Figure 2 for an illustration.

Let $I^{(0)}$ denote the layout of $(\mathbf{x}_1^{(\mathsf{in})}, \mathbf{x}_2^{(\mathsf{in})})$. In the pairwise setting, the layout can be defined as a subset $I^{(0)} \subseteq [k]$ including the coordinates where $\mathbf{x}_1^{(\mathsf{in})}, \mathbf{x}_2^{(\mathsf{in})}$ collide, or, equivalently, $I^{(0)}$ consists of all the coordinates where $\mathbf{x}_\Delta^{(\mathsf{in})}$ is zero. In general, we say $I \subseteq [k]$ is the layout of $\mathbf{x} \in \mathbb{F}^k$, or $\mathbf{x}$ is in layout I, if I consists precisely of the zero coordinates of $\mathbf{x}$. That is,

$$\mathbf{x} \text{ in } I \quad \text{means} \quad \forall i \in [k],\ i \in I \iff \mathbf{x}[i] = 0.$$

Due to the randomness of the S-boxes, $\mathbf{x}_\Delta^{(1)}$ is distributed uniformly among all vectors in layout $I^{(0)}$. Similarly, if we let $I^{(1)}$ (resp. $I^{(2)}$) denote the layout of $\mathbf{y}_\Delta^{(1)}$ (resp. $\mathbf{y}_\Delta^{(2)}$), then $\mathbf{x}_\Delta^{(2)}$ (resp. $\mathbf{y}_\Delta^{(\mathsf{out})}$) is distributed uniformly among all vectors in layout $I^{(1)}$ (resp. $I^{(2)}$).

It is easy to show that if $I^{(2)}$ is close to the distribution on layouts induced by a random (non-zero) vector, then the distribution of $\mathbf{y}_\Delta^{(\mathsf{out})}$ is close to uniform. Thus the heart of the analysis is to understand how the distribution of $I^{(r)}$ depends on that of $I^{(r-1)}$. Evidently, this depends on the characteristics of the linear mixing layer. In particular, we show the following lemma.

> **Lemma 3** (informal). If $I^{(r-1)}$ is *nice* in the sense that $|I^{(r-1)}| \le k/2$, then $I^{(r)}$ is $2^{-\Omega(kb)}$-close in variation distance to the layout of a random vector.

The blueprint. We now use the above lemma to prove that 2-round SPN* is close to 2-wise independent using the following blueprint. All the error terms in the analysis have magnitude $2^{-\Omega(kb)}$.

In the first round: If $I^{(0)}$ is nice, then $I^{(1)}$ is statistically close to the layout of a random vector by Lemma 3 above, so $I^{(1)}$ is nice with high probability. If $I^{(0)}$ is not nice, then we claim that $I^{(1)}$ must be nice due to the fact that the linear mixing matrix M has maximal branch number. Recall that this guarantees $\mathsf{wt}(\mathbf{x}) + \mathsf{wt}(M\mathbf{x}) \ge k+1$ for all $\mathbf{0} \ne \mathbf{x} \in \mathbb{F}^k$. Thus, if $I^{(0)}$ is not nice, $I^{(1)}$ must be nice. In either case, $I^{(1)}$ is very likely to be nice.

In the second round: Since $I^{(1)}$ is very likely to be nice, $I^{(2)}$ is close to the layout of a random vector again by Lemma 3, which implies that $\mathbf{y}_\Delta^{(\mathsf{out})}$ is close to uniform.

Our analysis of the t-wise setting in Section 5 follows the same high-level framework, which requires in particular generalizing the notion of a layout and its niceness.

Proof sketch of Lemma 3. The rest of this overview provides a proof sketch of the lemma. The transition probability from $I^{(r-1)}$ to $I^{(r)}$ can be written as

$$\Pr\big[I^{(r)} = J \;\big|\; I^{(r-1)} = I\big] = \Pr_{\mathbf{x}\,\mathsf{in}\,I}[M\mathbf{x} \ \mathsf{in}\ J] = \frac{\#\{\mathbf{x} \text{ s.t. } \mathbf{x} \ \mathsf{in}\ I \wedge M\mathbf{x} \ \mathsf{in}\ J\}}{\#\{\mathbf{x} \text{ s.t. } \mathbf{x} \ \mathsf{in}\ I\}} \ .$$

Define an indicator function $\mathbb{1}_M$ where $\mathbb{1}_M(\mathbf{x}, \mathbf{y}) = 1$ if and only if $M\mathbf{x} = \mathbf{y}$. Then

$$\Pr\big[I^{(r)} = J \;\big|\; I^{(r-1)} = I\big] = \frac{\sum_{\mathbf{x}\,\mathsf{in}\,I} \sum_{\mathbf{y}\,\mathsf{in}\,J} \mathbb{1}_M(\mathbf{x}, \mathbf{y})}{\sum_{\mathbf{x}\,\mathsf{in}\,I} 1} \ . \tag{1}$$

To compute the numerator, it turns out that it is convenient to relax the notion of being in a layout. In particular, we say that $\mathbf{x}$ *satisfies* layout I as follows:

$$\mathbf{x} \ \mathsf{SAT}\ I \qquad \text{means} \qquad \forall i \in [k],\ i \in I \implies \mathbf{x}[i] = 0.$$

In particular, if $\mathbf{x}$ is in layout I, it satisfies layout I, but not vice versa.

Note that if M has the maximal branch number, then one can show that

$$\sum_{\mathbf{x}\,\mathsf{SAT}\,I} \sum_{\mathbf{y}\,\mathsf{SAT}\,J} \mathbb{1}_M(\mathbf{x}, \mathbf{y}) = \begin{cases} (2^b)^{k-|I|-|J|} & \text{if } |I| + |J| \le k \ , \\ 1 & \text{if } |I| + |J| > k \ . \end{cases} \tag{2}$$

Also, note that

$$\sum_{\mathbf{x}\,\mathsf{SAT}\,I} \sum_{\mathbf{y}\,\mathsf{SAT}\,J} \frac{1}{(2^b)^k} = (2^b)^{k-|I|-|J|} \tag{3}$$

is very close to (2), off by at most 1 for any I and J. In order to express the numerator of (1) in closed form, we first note that (2) and (3) should remain close if the sum operator is replaced by $\sum_{\mathbf{x}\,\mathsf{in}\,I} \sum_{\mathbf{y}\,\mathsf{in}\,J}$. That is, by the inclusion-exclusion principle (details in Section 4.1)

$$\begin{aligned} &\sum_{\mathbf{x}\,\mathsf{in}\,I} \sum_{\mathbf{y}\,\mathsf{in}\,J} \left(\mathbb{1}_M(\mathbf{x}, \mathbf{y}) - \frac{1}{(2^b)^k}\right) \\ &\quad = \sum_{I' \supseteq I} \sum_{J' \supseteq J} (-1)^{|I' \setminus I| + |J' \setminus J|} \sum_{\mathbf{x}\,\mathsf{SAT}\,I'} \sum_{\mathbf{y}\,\mathsf{SAT}\,J'} \left(\mathbb{1}_M(\mathbf{x}, \mathbf{y}) - \frac{1}{(2^b)^k}\right) = \sum_{I' \supseteq I} \sum_{J' \supseteq J} O(1) = O(2^{2k}). \end{aligned}$$

Plugging it in (1) gives a good bound on the transition probability

$$\Pr\big[I^{(r)} = J \;\big|\; I^{(r-1)} = I\big] = \frac{\displaystyle\sum_{\mathbf{x}\,\mathsf{in}\,I} \sum_{\mathbf{y}\,\mathsf{in}\,J} \frac{1}{(2^b)^k} + O(2^k)}{\sum_{\mathbf{x}\,\mathsf{in}\,I} 1} = \overbrace{\sum_{\mathbf{y}\,\mathsf{in}\,J} \frac{1}{(2^b)^k}}^{=\Pr_{\mathbf{y}}[\mathbf{y}\,\mathsf{in}\,J]} + \overbrace{\frac{O(2^{2k})}{(2^b - 1)^{k-|I|}}}^{\text{err}} \ .$$

The error term is of the order of $2^{-\Omega(kb)}$ if I is nice (i.e., $|I| \le k/2$). The transition probability is close to $\sum_{\mathbf{y}\,\mathsf{in}\,J} \frac{1}{(2^b)^k}$, which is the probability that a random vector lies in J. This can then be turned into a bound on the statistical distance to conclude the proof of the lemma.

2 Preliminaries

For any positive integer n, let $[n]$ denote the set $\{1, 2, \ldots, n\}$. We will use bold-face letters such as $\mathbf{x}$ to denote vectors and will denote the i^{th} coordinate of such a vector by $\mathbf{x}[i]$. For an integer $b \geq 1$, we let $\mathbb{F}_{2^b}$ denote the finite field of size 2^b. We also denote a finite field by $\mathbb{F}$ when the field size is clear from the context.

2.1 Substitution-Permutation Networks (SPN)

A *Substitution-Permutation Network* (SPN) is parameterized by the number of rounds, denoted by r; the word length, denoted by b; the width parameter, denoted by k; the linear mixing permutation, a full rank matrix $M : (\mathbb{F}_{2^b})^k \to (\mathbb{F}_{2^b})^k$; and an S-box permutation $S : \mathbb{F}_{2^b} \to \mathbb{F}_{2^b}$. All these parameters are public. The network is a keyed permutation over $(\mathbb{F}_{2^b})^k$, so every input (output) vector is bk-bit long. The key is a tuple of $r + 1$ (meant to be uniformly random) vectors $\mathbf{k}_0, \mathbf{k}_1, \ldots, \mathbf{k}_r \in (\mathbb{F}_{2^b})^k$. The "independent round keys" assumption here is very common and rooted in the model of Markov Ciphers from the seminal works of Lai, Massey, and Murphy [LMM91], Nyberg [Nyb93] and follow-ups. We follow the convention that the number of rounds is the same as the number of mixing layers. In Figure 1, we give an illustration of a 3-round SPN.

SPN with Random Secret S-boxes (SPN*). Much of this work will deal with SPN networks where each S-box is chosen independently at random from the set of all permutations on $\mathbb{F} := \mathbb{F}_{2^b}$, and kept secret. In this case, the set of S-boxes acts as the key, and there is no reason to have a separate addition of round keys. Thus, the key of the network consists of $k(r + 1)$ permutations $S_j^{(i)} : \mathbb{F} \to \mathbb{F}$ (for $0 \leq i \leq r, 1 \leq j \leq k$).

Given input $\mathbf{x}^{(\mathsf{in})} = \mathbf{y}^{(0)} \in \mathbb{F}^k$ and the key, the output $\mathbf{y}^{(\mathsf{out})} = \mathbf{x}^{(r+1)} \in \mathbb{F}^k$ is determined by alternating the following two steps, as illustrated in Figure 2. For consistency, we let $\mathbf{y}^{(0)}$ be another name for $\mathbf{x}^{(\mathsf{in})}$ and let $\mathbf{x}^{(r+1)}$ be another name for $\mathbf{y}^{(\mathsf{out})}$.

Substitution Step-i $(0 \leq i \leq r)$ For $1 \leq j \leq k$, let $\mathbf{x}^{(i+1)}[j] = S_j^{(i)}(\mathbf{y}^{(i)}[j])$,

Permutation Step-i $(1 \leq i \leq r)$ Let $\mathbf{y}^{(i)} = M\mathbf{x}^{(i)}$.

We call $\mathbf{x}^{(i)}$ and $\mathbf{y}^{(i)}$ the intermediate values of the i-th round. Then the input $\mathbf{x}^{(\mathsf{in})}$, also called $\mathbf{y}^{(0)}$, is in "the 0-th round". This gets fed into the substitution step-0 which produces $\mathbf{x}^{(1)}$. Permutation step-i is inside the i-th round. Substitution step-i is the boundary between the i-th round and the $(i + 1)$-th round. The output $\mathbf{y}^{(\mathsf{out})}$, also called $\mathbf{x}^{(r+1)}$, is in "the $(r + 1)$-th round".

Branch number. We use the definition of the branch number of a matrix that quantifies how well the linear layer "mixes" its input.

Definition 1. *The branch number of a matrix $M \in \mathbb{F}^{k\times k}$ is defined to be*

$$\mathsf{br}(M) = \mathsf{min}_{0\neq\alpha\in(\mathbb{F}_{2^b})^k}(\mathsf{wt}(\alpha) + \mathsf{wt}(M\alpha))$$

where wt *denotes the Hamming weight.*

Having the maximal branch number (namely, $k+1$) is considered a desirable feature for mixing functions [Dae95, KHL$^+$02].

Summary of notations. The intermediate states in an SPN (or SPN*) network are denoted by boldface letters $\mathbf{x}$ or $\mathbf{y}$. The notation $\mathbf{x}^{(r)}$ (resp. $\mathbf{y}^{(r)}$) is used to denote the state at round r; and $\mathbf{x}^{(r)}[s]$ denotes the s^{th} coordinate of $\mathbf{x}^{(r)}$. When dealing with multiple inputs, we let the subscript denote which input we are referring to: i.e., $\mathbf{x}_i^{(r)}$ denotes round-r state of the i^{th} input. We let $\mathbf{x}_{1:t}^{(r)} = (\mathbf{x}_i^{(r)})_{i\in[t]} = (\mathbf{x}_1^{(r)}, \ldots, \mathbf{x}_t^{(r)})$ be a shorthand for a tuple of vectors.

3 Layouts

This section introduces *layout*, a key notion of this paper. In the pairwise setting, layout is similar to the notions of an activity pattern [DR02] or support [BV05] of an input that have been formulated in the literature in the context of differential and linear cryptanalysis. Our notion considers the generalized setting and deals with t-tuples of inputs for an arbitrary t.

Motivation. Given t inputs $\mathbf{x}_1^{(\mathsf{in})}, \ldots, \mathbf{x}_t^{(\mathsf{in})}$ to an SPN* network, we want to characterize the joint distribution of the outputs $\mathbf{y}_1^{(\mathsf{out})}, \ldots, \mathbf{y}_t^{(\mathsf{out})}$ when all the S-boxes are i.i.d. uniform. The evaluation of the SPN* on these t inputs is essentially a Markov chain. The dependency between the intermediate values can be illustrated by the following Bayesian network.

$$\begin{tikzcd}\mathbf{x}_{1:t}^{(\mathsf{in})} \arrow[r] & \mathbf{x}_{1:t}^{(1)} \arrow[r] & \mathbf{y}_{1:t}^{(1)} \arrow[r] & \mathbf{x}_{1:t}^{(2)} \arrow[r] & \mathbf{y}_{1:t}^{(2)} \arrow[r] & \cdots\end{tikzcd}$$

Here $\mathbf{x}_{1:t}^{(r)}$ denotes the tuple of t vectors $(\mathbf{x}_1^{(r)}, \ldots, \mathbf{x}_t^{(r)})$, and so does $\mathbf{y}_{1:t}^{(r)}$.

The tuple $\mathbf{y}_{1:t}^{(r)}$ depends deterministically on $\mathbf{x}_{1:t}^{(r)}$ via the permutation step. The substitution step is more interesting. The randomness of the substitution step-r consists of k S-boxes $S_1^{(r)}, \ldots, S_k^{(r)}$. Each S-box $S_s^{(r)}$ is applied to the corresponding coordinate for all inputs, namely, $\mathbf{y}_i^{(r)}[s]$ for all $i \in [t]$. The substitution step erases most information, but some are preserved. In particular,

- $\mathbf{y}_i^{(r)}[s] = \mathbf{y}_j^{(r)}[s]$ if and only if $\mathbf{x}_i^{(r+1)}[s] = \mathbf{x}_j^{(r+1)}[s]$.

And it is not hard to verify that this is the only information preserved. In particular, the distribution of $\mathbf{x}_{1:t}^{(r+1)}$ is uniform among all tuples that satisfy

$$\forall i, j \in [t],\ \forall s \in [k],\ \mathbf{x}_i^{(r+1)}[s] = \mathbf{x}_j^{(r+1)}[s] \iff \mathbf{y}_i^{(r)}[s] = \mathbf{y}_j^{(r)}[s].$$

To capture and formalize these constraints, we introduce the notion of a *layout*. The layout of t vectors $\mathbf{x}_{1:t}$ should specify whether $\mathbf{x}_i[s] = \mathbf{x}_j[s]$, for any $i, j \in [t], s \in [k]$.

Definition 2 (layouts)**.** *A t-wise layout I is defined as $I = (I_{i,j})_{1\le i<j\le t}$. Each $I_{i,j}$ is a subset of $[k]$. For a tuple of t vectors $\mathbf{x}_{1:t} = (\mathbf{x}_1, \ldots, \mathbf{x}_t) \in (\mathbb{F}^k)^t$, we say that the tuple is in a layout I, denoted by $\mathbf{x}_{1:t}$ in I, if*

$$\forall 1 \le i < j \le t,\ \forall s \in [k],\ s \in I_{i,j} \iff \mathbf{x}_i[s] = \mathbf{x}_j[s].$$

We say I is the layout of $\mathbf{x}_{1:t}$, denoted by $\mathrm{layout}(\mathbf{x}_{1:t}) = I$*, if $\mathbf{x}_{1:t}$ is in layout I.*

We also define a weaker notion: say $\mathbf{x}_{1:t}$ satisfies *a layout I, denoted by $\mathbf{x}_{1:t}$* SAT *I, if*

$$\forall 1 \le i < j \le t,\ \forall s \in [k],\ s \in I_{i,j} \implies \mathbf{x}_i[s] = \mathbf{x}_j[s].$$

Given another layout $J = (J_{i,j})_{1\le i<j\le t}$, we say J is stricter or equal to I, denoted by $J \supseteq I$ or $I \subseteq J$, if

$$\forall 1 \le i < j \le t,\ J_{i,j} \supseteq I_{i,j}.$$

Example 1. *Consider the 3-wise layout $I = (I_{1,2}, I_{1,3}, I_{2,3}) = (\{1\}, \{2\}, \{3\})$. Then, the tuple of vectors $\mathbf{x}_1 = [a, b, c']$, $\mathbf{x}_2 = [a, b', c]$, and $\mathbf{x}_3 = [a', b, c]$ lay in the layout I a long as $a \neq a'$, $b \neq b'$, $c \neq c'$.*

Note that not all layouts are "valid". For example,

$$I = (I_{1,2}, I_{1,3}, I_{2,3}) = (\{1\}, \varnothing, \{1\}).$$

is not the layout of any 3-tuple. Because $1 \in I_{1,2}$ means the first two vectors agree on coordinate 1, and $1 \in I_{2,3}$ means the last two vectors agree on coordinate 1, by transitivity, these imply $1 \in I_{1,3}$. We say a layout I is *valid* if for all $s \in [k]$ and for all $i < i' < i''$, if any two of $I_{i,i'}, I_{i,i''}, I_{i',i''}$ contain s, so does the third one.

Random walks on layouts. Using the notion of layouts, the distribution of $\mathbf{x}^{(r+1)}_{1:t}$ conditioned on $\mathbf{y}^{(r)}_{1:t}$ can be described more concisely: the substitution step simply samples a random $\mathbf{x}^{(r+1)}_{1:t}$ who is in the same layout as $\mathbf{y}^{(r)}_{1:t}$. In other words, the substitution step is equivalent to a two-step process: first extract the layout of $\mathbf{y}^{(r)}_{1:t}$, then sample a random tuple from the layout. If letting $I^{(r)}$ denote the layout of $\mathbf{y}^{(r)}_{1:t}$ (and also $\mathbf{x}^{(r+1)}_{1:t}$, since they are in the same layout), the Bayesian network of the SPN* evaluation can also be written in the following way:

$$\begin{tikzcd}
\mathbf{x}^{(\mathsf{in})}_{1:t} = \mathbf{y}^{(0)}_{1:t} \arrow[dr] & & \mathbf{x}^{(1)}_{1:t} \arrow[r] & \mathbf{y}^{(1)}_{1:t} \arrow[dr] & & \mathbf{x}^{(2)}_{1:t} \arrow[r] & \mathbf{y}^{(2)}_{1:t} \arrow[dr] & & \cdots \\
& I^{(0)} \arrow[ur] & & & I^{(1)} \arrow[ur] & & & I^{(2)} \arrow[ur] &
\end{tikzcd}$$

This Bayesian network view through the lens of layouts suggests that the right problem to study is the transition probability from $I^{(r)}$ to $I^{(r+1)}$ (induced by the linear mixing layer). This transition probability could be easier to characterize since the space of all layouts is much smaller than the space of all t-tuples.

All theorems in this paper follow this framework. They essentially prove the following statement: Starting from any layout $I^{(0)}$, after some r rounds, the distribution of $I^{(r)}$ is close to t-wise independent. To complete the framework, we need to answer two questions: 1) What is the definition of a layout being close to t-wise independent; and 2) How does a layout being close to t-wise independent imply that a random tuple in the layout is close to t-wise independent?

Definition 3 (closeness to t-wise independence)**.** *Let $\mathbf{z}_1, \ldots, \mathbf{z}_t$ be sampled uniformly at random from $\mathbb{F}^k$ with (resp. without) replacement. Then we say the tuple $(\mathbf{z}_1, \ldots, \mathbf{z}_t)$ is t-wise independent with (resp. without) replacement.*

Let $(\mathbf{x}_1, \ldots, \mathbf{x}_t)$ be sampled from a distribution. We say $(\mathbf{x}_1, \ldots, \mathbf{x}_t)$ is ε-close to t-wise independent with (resp. without) replacement if

$$\Delta_{TV}\Big((\mathbf{x}_1, \ldots, \mathbf{x}_t), (\mathbf{z}_1, \ldots, \mathbf{z}_t)\Big) \le \varepsilon.$$

Let layout I be sampled from a distribution. We say I is ε-close to t-wise independent with (resp. without) replacement if

$$\Delta_{TV}\Big(I, \mathrm{layout}(\mathbf{z}_1, \ldots, \mathbf{z}_t)\Big) \leq \varepsilon.$$

We say a keyed permutation (e.g., a SPN) is ε-close to t-wise independent with (resp. without) replacement if for any t distinct input $\mathbf{x}_{1:t}^{(in)}$, the joint distribution of the t corresponding output $\mathbf{y}_{1:t}^{(out)}$ is ε-close to t-wise independent with (resp. without) replacement, assuming the key is sampled properly.*

The following lemma and its corollary show how the distribution of t-tuples is related to the distribution of their layouts, and justify why this 'layout' analysis suffices for our purposes of proving t-wise independence. Their proofs are deferred to the full version of the paper.

Lemma 1. *Assume I and $\mathbf{x}_{1:t} = (\mathbf{x}_1, \ldots, \mathbf{x}_t)$ jointly come from a distribution where $\mathbf{x}_{1:t}$ is a random tuple in I when conditioning on I, and similarly for J and $\mathbf{z}_{1:t}$. Then*

(a) $\Delta_{TV}\big(I, J\big) = \Delta_{TV}\big(\mathbf{x}_{1:t}, \mathbf{z}_{1:t}\big)$.

(b) $\mathbf{x}_{1:t}$ *is ε-close to t-wise independent with (resp. without) replacement if and only if I is ε-close to t-wise independent with (resp. without) replacement.*

Proof. (a) follows from the data-processing inequality for total variation distance.

To prove (b), let the tuple $\mathbf{z}_{1:t} = (\mathbf{z}_1, \ldots, \mathbf{z}_t)$ be t-wise independent with (resp. without) replacement, and let $J = \mathrm{layout}(\mathbf{z}_{1:t})$. Note that conditioning on J, the tuple $\mathbf{z}_{1:t}$ is uniform in J. Thus

$$\Delta_{\mathrm{TV}}\big(\mathbf{x}_{1:t}, \mathbf{z}_{1:t}\big) = \Delta_{\mathrm{TV}}\big(I, \mathrm{layout}(\mathbf{z}_{1:t})\big). \qquad \square$$

4 Warm-up: 2-Wise Independence of 2-round SPN*

In this section, we present the core idea of our new technique and demonstrate its power by showing that a 2-round SPN* is $2^{-\Theta(kb)}$-close to 2-wise independent. That is, we show that for any two distinct inputs $(\mathbf{x}_1^{(\mathrm{in})}, \mathbf{x}_2^{(\mathrm{in})})$ (which is also named as $(\mathbf{y}_1^{(0)}, \mathbf{y}_2^{(0)})$) the joint distribution of their corresponding outputs $(\mathbf{y}_1^{(\mathrm{out})}, \mathbf{y}_2^{(\mathrm{out})})$ (which is also named as $(\mathbf{x}_1^{(3)}, \mathbf{x}_2^{(3)})$) is close to 2-wise independent.

Theorem 1. *2-round SPN* is ε-close to 2-wise independent, where*

$$\varepsilon \leq \frac{3^k}{(2^{b-1} - \frac{1}{2})^{k/2}},$$

if its linear mixing function has maximal branch number (see Definition 1).

The theorem will be proved in Section 4.2. At a high level, the proof is the combination of the following two statements.

- **After the first round, the layout is nice w.h.p.** That is, starting from any pair of inputs, the intermediate layout is "nice" with overwhelmingly high probability. A layout is nice if the number of collisions (i.e., coordinates where the two vectors agree) is relatively small.

- **If the layout is nice before the second round, the output is close to 2-wise independent.** That is, conditioning on the intermediate layout being any nice layout, the pair of outputs will be close to 2-wise independent

Let $I^{(r)}$ denote the layout of $(\mathbf{y}_1^{(r)}, \mathbf{y}_2^{(r)})$ and $(\mathbf{x}_1^{(r+1)}, \mathbf{x}_2^{(r+1)})$. Since the section only discusses the 2-wise setting, the representation of a layout can be simplified. A layout is represented by a subset $I \subseteq [k]$, such that $i \in I$ means the two vectors agree on the i-th position.

As pointed out by the standard differential cryptanalysis, it would be helpful to consider the difference between each pair of vectors

$$\mathbf{x}_\Delta^{(r)} := \mathbf{x}_1^{(r)} - \mathbf{x}_2^{(r)}, \qquad \mathbf{y}_\Delta^{(r)} := \mathbf{y}_1^{(r)} - \mathbf{y}_2^{(r)}.$$

Note that for each $s \in [k]$

$$\mathbf{y}_\Delta^{(r)}[s] = 0 \iff (\mathbf{y}_1^{(r)}[s] = \mathbf{y}_2^{(r)}[s]) \iff s \in I^{(r)}.$$

This suggests that $\mathbf{y}_\Delta^{(r)}$ is also "in" $I^{(r)}$. This can be formalized by introducing the following simplified definition for the pairwise setting.

Definition 4. *A (pairwise) layout I is a subset of $[k]$. For any vector $\mathbf{x}_\Delta$ and layout I, define*

$$\begin{aligned} \mathbf{x}_\Delta \ \mathsf{SAT}\ I &\iff (\forall s \in [k],\ s \in I \implies \mathbf{x}[s] = 0), \\ \mathbf{x}_\Delta \ \mathsf{in}\ I &\iff (\forall s \in [k],\ s \in I \iff \mathbf{x}[s] = 0). \end{aligned}$$

And we say I is the layout of $\mathbf{x}_\Delta$, denoted by $I = \mathrm{layout}(\mathbf{x}_\Delta)$, if $\mathbf{x}_\Delta$ in I.

Then for any vector difference $\mathbf{x}_\Delta = \mathbf{x}_1 - \mathbf{x}_2$, we have

$$\mathbf{x}_\Delta \ \mathsf{SAT}\ I \iff (\mathbf{x}_1, \mathbf{x}_2) \ \mathsf{SAT}\ I, \qquad \mathbf{x}_\Delta \ \mathsf{in}\ I \iff (\mathbf{x}_1, \mathbf{x}_2) \ \mathsf{in}\ I,$$

and $\mathrm{layout}(\mathbf{x}_\Delta) = \mathrm{layout}(\mathbf{x}_1, \mathbf{x}_2)$.

As observed by differential cryptoanalysis, it suffices to *only* consider the difference vectors, since the whole analysis can ignore the original pair of vectors.

- Permutation step: $\mathbf{y}_\Delta^{(r)} = M\mathbf{x}_\Delta^{(r)}$.
- Substitution step: $\mathbf{x}_\Delta^{(r+1)}$ is a random tuple whose layout is the same as $\mathbf{y}_\Delta^{(r)}$.
- Output: The pair of output vectors is ε-close to 2-wise independent if and only if $I^{(2)} = \mathrm{layout}(\mathbf{y}_\Delta^{(2)})$ is ε-close to 2-wise independent (Lemma 1).

4.1 The Layout Transition Probability

This section computes the transition probability from layout $I^{(r)}$ to $I^{(r+1)}$. Their dependency can be captured by the following Bayesian network.

$$\begin{tikzcd} I^{(r-1)} \arrow[r] & \mathbf{x}_\Delta^{(r)} \arrow[r, "\mathsf{M}"] & \mathbf{y}_\Delta^{(r)} \arrow[r] & I^{(r)} \end{tikzcd}$$

Let trans-prob(I, J) denote the probability $I^{(r)} = J$ conditioning on $I^{(r-1)} = I$. Formally, trans-prob(I, J) is the probability layout$(M\mathbf{x}) = J$ when the (difference) vector $\mathbf{x}$ is sampled uniformly from layout I. By definition,

$$\text{trans-prob}(I, J) = \Pr_{\mathsf{x}\,\mathsf{in}\,I}\Big[M\mathbf{x}\ \mathsf{in}\ J\Big] = \frac{\#\big\{\mathbf{x} : \mathbf{x}\ \mathsf{in}\ I \text{ and } M\mathbf{x}\ \mathsf{in}\ J\big\}}{\#\big\{\mathbf{x} : \mathbf{x}\ \mathsf{in}\ I\big\}}.$$

To simplify this expression, we introduce some new notations.

For the denominator, we define free$(I) = k - |I|$, which stands for the number of "free" coordinates. Then $\#\big\{\mathbf{x} : \mathbf{x}\ \mathsf{in}\ I\big\} = (2^b - 1)^{\text{free}(I)}$.

Denote the numerator by trans-count(I, J). Define indicator function $\mathbb{1}_M$ as

$$\mathbb{1}_M(\mathbf{x}, \mathbf{y}) := \begin{cases} 1 & \text{if } M\mathbf{x} = \mathbf{y}, \\ 0 & \text{otherwise.} \end{cases}$$

Then the numerator can be written as

$$\text{trans-count}(I, J) = \#\big\{\mathbf{x} : \mathbf{x}\ \mathsf{in}\ I \text{ and } M\mathbf{x}\ \mathsf{in}\ J\big\} = \sum_{\mathsf{x}\,\mathsf{in}\,I} \sum_{\mathsf{y}\,\mathsf{in}\,J} \mathbb{1}_M(\mathbf{x}, \mathbf{y}).$$

The core idea is to also consider another sum operator $\sum_{\mathsf{x}\,\mathsf{SAT}\,I}$. For any function f, we have

$$\sum_{\mathsf{x}\,\mathsf{SAT}\,I} f(\mathbf{x}) = \sum_{I' \supseteq I} \sum_{\mathsf{x}\,\mathsf{in}\,I'} f(\mathbf{x}).$$

Then by the inclusion-exclusion principle,

$$\sum_{\mathsf{x}\,\mathsf{in}\,I} f(\mathbf{x}) = \sum_{I' \supseteq I} (-1)^{|I' \setminus I|} \sum_{\mathsf{x}\,\mathsf{SAT}\,I'} f(\mathbf{x}).$$

Consider the following sum

$$\sum_{\mathsf{x}\,\mathsf{SAT}\,I} \sum_{\mathsf{y}\,\mathsf{SAT}\,J} \mathbb{1}_M(\mathbf{x}, \mathbf{y}) = \#\big\{\mathbf{x} : \mathbf{x}\ \mathsf{SAT}\ I \text{ and } M\mathbf{x}\ \mathsf{SAT}\ J\big\} \tag{4}$$

that looks similar to trans-count(I, J). The only difference is whether to enumerate vectors *in* I, J or to enumerate vectors *satisfying* I, J. The value of (4) is easier to compute. It is the number of solutions of a linear system, which must be a power of $|\mathbb{F}| = 2^b$. In particular, if the matrix M has the maximal branch number, we have

$$\sum_{\mathsf{x}\,\mathsf{SAT}\,I} \sum_{\mathsf{y}\,\mathsf{SAT}\,J} \mathbb{1}_M(\mathbf{x}, \mathbf{y}) = \begin{cases} (2^b)^{\text{free}(I) + \text{free}(J) - k} & \text{if free}(I) + \text{free}(J) \geq k, \\ 1 & \text{if free}(I) + \text{free}(J) < k. \end{cases} \tag{5}$$

Then by the inclusion-exclusion principle,

$$\begin{aligned} \text{trans-count}(I, J) &= \sum_{\mathsf{x}\,\mathsf{in}\,I} \sum_{\mathsf{y}\,\mathsf{in}\,J} \mathbb{1}_M(\mathbf{x}, \mathbf{y}) \\ &= \sum_{I' \supseteq I} \sum_{J' \supseteq J} (-1)^{|I' \setminus I| + |J' \setminus J|} \sum_{\mathsf{x}\,\mathsf{SAT}\,I'} \sum_{\mathsf{y}\,\mathsf{SAT}\,J'} \mathbb{1}_M(\mathbf{x}, \mathbf{y}) \\ &= \sum_{I' \supseteq I} \sum_{J' \supseteq J} (-1)^{|I' \setminus I| + |J' \setminus J|} \big(2^b\big)^{\max(\text{free}(I') + \text{free}(J') - k, 0)}. \end{aligned} \tag{6}$$

Now we are ready to present our results about the layout transition probability. They are essentially polishing (6).

Lemma 2. *If M has the maximal branch number, the layout transition probability* $\text{trans-prob}(I,J) := \Pr_{\mathbf{x}\,\mathsf{in}\,I}\big[M\mathbf{x}\ \mathsf{in}\ J\big]$ *is bounded by*

$$\left|\text{trans-prob}(I,J) - \frac{(2^b-1)^{\text{free}(J)}}{(2^b)^k}\right| \le \frac{2^{\text{free}(I)+\text{free}(J)}}{(2^b-1)^{\text{free}(I)}}.$$

Proof. Consider function $u(\mathbf{x},\mathbf{y}) = \frac{1}{(2^b)^k}$. If we view $u(\mathbf{x},\mathbf{y})$ as the conditional probability of $\mathbf{y}$ given $\mathbf{x}$, then it captures the process that $\mathbf{y}$ is sampled uniformly at random and is independent of $\mathbf{x}$. Notice that

$$\sum_{\mathbf{x}\,\mathsf{SAT}\,I}\sum_{\mathbf{y}\,\mathsf{SAT}\,J} u(\mathbf{x},\mathbf{y}) = \sum_{\mathbf{x}\,\mathsf{SAT}\,I}\sum_{\mathbf{y}\,\mathsf{SAT}\,J}\frac{1}{(2^b)^k} = (2^b)^{\text{free}(I)+\text{free}(J)-k}$$

is very similar to (5). The difference is no more than 1 for any I, J. Therefore, in some sense, u is a very good approximation of $\mathbb{1}_M$. With this intuition in mind, we expect

$$\underbrace{\sum_{\mathbf{x}\,\mathsf{in}\,I}\sum_{\mathbf{y}\,\mathsf{in}\,J}\mathbb{1}_M(\mathbf{x},\mathbf{y})}_{=\text{trans-count}(I,J)} \quad - \quad \sum_{\mathbf{x}\,\mathsf{in}\,I}\sum_{\mathbf{y}\,\mathsf{in}\,J} u(\mathbf{x},\mathbf{y}) \tag{7}$$

to be very small as well. The difference between them is bounded by

$$\begin{aligned}\left|\sum_{\mathbf{x}\,\mathsf{in}\,I}\sum_{\mathbf{y}\,\mathsf{in}\,J}\left(\mathbb{1}_M(\mathbf{x},\mathbf{y}) - \frac{1}{(2^b)^k}\right)\right| &= \left|\sum_{I'\supseteq I}\sum_{J'\supseteq J}(-1)^{|I'\setminus I|+|J'\setminus J|}\sum_{\mathbf{x}\,\mathsf{SAT}\,I'}\sum_{\mathbf{y}\,\mathsf{SAT}\,J'}\left(\mathbb{1}_M(\mathbf{x},\mathbf{y}) - \frac{1}{(2^b)^k}\right)\right| \\ &\le \sum_{I'\supseteq I}\sum_{J'\supseteq J} 1 = 2^{\text{free}(I)+\text{free}(J)}.\end{aligned}$$

So we can approximate the transition probability by

$$\text{trans-prob}(I,J) = \frac{\displaystyle\sum_{\mathbf{x}\,\mathsf{in}\,I}\sum_{\mathbf{y}\,\mathsf{in}\,J}\frac{1}{(2^b)^k} + \text{term (7)}}{\displaystyle\sum_{\mathbf{x}\,\mathsf{in}\,I} 1} = \underbrace{\sum_{\mathbf{y}\,\mathsf{in}\,J}\frac{1}{(2^b)^k}}_{\text{approximation}} + \underbrace{\frac{\text{term (7)}}{\sum_{\mathbf{x}\,\mathsf{in}\,I} 1}}_{\text{error}}.$$

The approximation term is particularly nice, as it can be interpreted as the probability that a random vector lies in layout J. It equals to

$$\sum_{\mathbf{y}\,\mathsf{in}\,J}\frac{1}{(2^b)^k} = \Pr_{\mathbf{y}\in\mathbb{F}^k}\Big[\mathbf{y}\ \mathsf{in}\ J\Big] = \frac{(2^b-1)^{\text{free}(J)}}{(2^b)^k}.$$

The absolute value of the error term is at most $2^{\text{free}(I)+\text{free}(J)}/(2^b-1)^{\text{free}(I)}$. □

Lemma 3. *Let $M : \mathbb{F}^k \to \mathbb{F}^k$ be a matrix with maximal branch number. For any layout I. Let J denote the layout of I after one round of SPN. That is,* $\text{trans-prob}(I,J)$ *is the probability mass function of J. Then J is ε-close to 2-wise independent, where $\varepsilon \le 3^k/2(2^{b-1}-\frac{1}{2})^{\text{free}(I)}$.*

Proof. The statistical distance is bounded by

$$\begin{aligned}\varepsilon &= \frac{1}{2}\sum_J \left|\text{trans-prob}(I,J) - \Pr_{\mathbf{y}\in\mathbb{F}^k}\left[\mathbf{y}\ \mathsf{in}\ J\right]\right| \\ &\le \frac{1}{2}\sum_J \frac{2^{\text{free}(I)+\text{free}(J)}}{(2^b-1)^{\text{free}(I)}} = \frac{2^{\text{free}(I)}\cdot 3^k}{2\cdot(2^b-1)^{\text{free}(I)}} = \frac{3^k}{2\cdot(2^{b-1}-\frac{1}{2})^{\text{free}(I)}}.\end{aligned}$$

□

4.2 The Niceness of a Layout

As stated by Lemma 3, if the starting input difference is in a layout I with large free(I), then after one round it will be very close to 2-wise independent. However, consider the opposite case when free$(I) = 1$, that is, the input difference $\mathbf{x}_\Delta$ is zero on all but one coordinate. Then after one round of SPN with maximal branch number mixing, the difference must be non-zero on every coordinate, which is about $(k/2^b)$ away from 2-wise independence.

So, when aiming for 2-wise independence, a layout I with larger free(I) is "nicer". We formalize this by defining the *niceness* of a layout. We say a layout I is α-nice if $|I| = k - \text{free}(I) \le \alpha k$.

To prove Theorem 1, we show that after one round, the layout is likely to be nice, then after one more round, it will be close to 2-wise independent.

Lemma 4. *Assume the mixing function has maximal branch number. For any* 2*-wise layout* I*, let* J *be sampled according to* trans-prob(I,J)*. Then for any* $\alpha \in [0,1]$*,*

$$\Pr[J \text{ is } \alpha\text{-nice}] \ge 1 - \frac{e\cdot\binom{k}{>\alpha k}}{(2^b-1)^{\alpha k}}.$$

Here $\binom{k}{>h}$ *denotes* $\sum_{i>h}\binom{k}{i}$.

Proof. The proof starts with an upper bound on the transition probability trans-prob(I,J) that does not depend on I.

$$\text{trans-prob}(I,J) = \frac{\sum_{\mathbf{x}\,\mathsf{in}\,I}\sum_{\mathbf{y}\,\mathsf{in}\,J}\mathbb{1}_M(\mathbf{x},\mathbf{y})}{(2^b-1)^{\text{free}(I)}} \le \frac{\sum_{\mathbf{x}\,\mathsf{SAT}\,I}\sum_{\mathbf{y}\,\mathsf{SAT}\,J}\mathbb{1}_M(\mathbf{x},\mathbf{y})}{(2^b-1)^{\text{free}(I)}} = \frac{(2^b)^{\max(\text{free}(I)+\text{free}(J)-k,0)}}{(2^b-1)^{\text{free}(I)}}.$$

Focus on the case that free(I) + free$(J) > k$, since otherwise trans-prob$(I,J) = 0$.

$$\text{trans-prob}(I,J) \le \frac{(2^b)^{\text{free}(I)+\text{free}(J)-k}}{(2^b-1)^{\text{free}(I)}} \le \left(\frac{2^b}{2^b-1}\right)^k \cdot \frac{1}{(2^b-1)^{k-\text{free}(J)}} \le \frac{e}{(2^b-1)^{k-\text{free}(J)}}.$$

The last inequality holds because the mixing function has maximal branch number inherently implies $k \le 2^b$ (Lemma 12).

We finish the proof by applying the union bound over all layouts J that are not α-nice. The number of not-α-nice layouts is $\binom{k}{>\alpha k}$. □

Proof of Theorem 1. Let $I^{(0)}, I^{(1)}, I^{(2)}$ denote the layout of the inputs, the layout of the middle vectors, the layout of the outputs respectively.

By Lemma 4,

$$\Pr\big[I^{(1)} \text{ is } \alpha\text{-nice}\big] \geq 1 - \frac{e \cdot \binom{k}{>\alpha k}}{(2^b - 1)^{\alpha k}}.$$

Conditioning on $I^{(1)}$ being α-nice, $I^{(2)}$ is $(3^k/2(2^{b-1} - \frac{1}{2})^{(1-\alpha)k})$-close to 2-wise independent, as shown by Lemma 3. Adding up all the errors, $I^{(2)}$ is ε-close to 2-wise independent, where

$$\varepsilon \leq \frac{3^k}{2 \cdot (2^{b-1} - \frac{1}{2})^{(1-\alpha)k}} + \frac{e \cdot \binom{k}{>\alpha k}}{(2^b - 1)^{\alpha k}}.$$

Set $\alpha = 1/2$ to minimize the statistical distance bound. □

5 The General Case of t-Wise Independence

In this section, we generalize our analysis of 2-wise independence in Section 4 to the t-wise setting. The high-level framework is mostly the same:

- Introducing the proper definition of the *niceness* of a layout.
- Starting from any t distinct inputs $(\mathbf{x}_1^{(\mathsf{in})}, \ldots, \mathbf{x}_t^{(\mathsf{in})})$, after one round (or a few rounds), the tuple will fall into some nice layout with high probability.
- Core lemma: For any nice layout I, if t inputs $(\mathbf{x}_1, \ldots, \mathbf{x}_t)$ are uniformly sampled from layout I, then after the linear mixing, the layout of $(\mathbf{y}_1, \ldots, \mathbf{y}_1) := (M\mathbf{x}_1, \ldots, M\mathbf{x}_t)$ is close to t-wise independent.

We define *nice* layouts as follows: For any t-wise layout $I = \{I_{i,j}\}_{1 \leq i < j \leq t}$, we say I is α*-nice* if and only if for all $1 < j \leq t$,

$$\left| \bigcup_{i<j} I_{i,j} \right| < \alpha k.$$

Here $\alpha \in [0,1]$ is a parameter quantifying the niceness of the layout. An equivalent definition is as follows: For any t-tuple $\mathbf{x}_{1:t} = (\mathbf{x}_1, \ldots, \mathbf{x}_t)$, say $\mathbf{x}_j$ collides with $\mathbf{x}_{1:j-1} = (\mathbf{x}_1, \ldots, \mathbf{x}_{j-1})$ on coordinate s if and only if there exists $i < j$ such that $\mathbf{x}_i[s] = \mathbf{x}_j[s]$. Then $\mathbf{x}_{1:t}$ is in an α-nice layout if and only if for every $1 < j \leq t$, $\mathbf{x}_j$ collides with $\mathbf{x}_{1:j-1}$ on at most αk coordinates.

If a t-tuple is sampled from a nice layout, it will be close to t-wise independent after one more round, as shown by our core lemma (Lemma 5). At a high level, the proof inductively uses the technique of its pairwise analog in Section 4.

Thanks to this core lemma, in order to show a r-round SPN* is close to t-wise independent, it suffices to show that after the first $r-1$ rounds, the tuple falls into some nice layout with high probability. We present three different results of this flavor. They differ in the following three criteria

- How large t can be (the core lemma supports t up to $2^{0.499b}$);
- How small the statistical error is (we are aiming for $2^{-\Theta(bk)}$ error); and
- How many rounds are required (ideally 2 rounds).

Each of our results optimizes two of the criteria, and compromises on the third criterion. Section 5.2 can only handle small t. Section 5.3 supports t up to $2^{0.499b}$ but the statistical error is slightly larger. Section 5.4 supports large t and keeps the statistical error $2^{-\Theta(bk)}$, but it requires $O(\log t)$ rounds.

5.1 Core Lemma & Conditional Transition Probability

Lemma 5. *For $\alpha \in [0,1]$ and any α-nice t-wise layout I, if tuple $(\mathbf{x}_1, \dots, \mathbf{x}_t)$ is sampled uniformly from layout I and let $(\mathbf{y}_1, \dots, \mathbf{y}_t) = (M\mathbf{x}_1, \dots, M\mathbf{x}_t)$, then the layout of $(\mathbf{y}_1, \dots, \mathbf{y}_t)$ is ε-close to t-wise independence with replacement, where*

$$\varepsilon \le t \cdot \left(\frac{2t}{2^b}\right)^{(1-\alpha)k} (2t)^k = t \cdot \left(\frac{(2t)^{2-\alpha}}{(2^b)^{1-\alpha}}\right)^k .$$

The lemma assumes the mixing function M has maximal branch number.

This section proves Lemma 5, which is the core of our analysis. The lemma says, if the tuple is in a nice layout at the beginning of a round (must be uniform within this layout due to the S-boxes), then the tuple will become very close to t-wise independent after this round.

The lemma is proved by induction. Assume the lemma holds for smaller t. Say $I = \{I_{a,b}\}_{1 \le a < b \le t}$ is a nice layout, $\mathbf{x}_{1:t}$ is sampled uniformly from layout I and $\mathbf{y}_{1:t} = M\mathbf{x}_{1:t}$, as in the lemma statement. By the definition of niceness, $\mathbf{x}_{1:t-1}$ is sampled uniformly from a nice $(t-1)$-wise layout $I' = \{I_{a,b}\}_{1 \le a < b \le t-1}$. By the induction hypothesis, layout$(\mathbf{y}_{1:t-1})$ is close to $(t-1)$-wise independent. To complete the induction, we need to show that the "conditional layout" of $\mathbf{y}_t$ is close to uniform. First, we need to formalize "conditional layout".

We want to analyze the distribution of $(\mathbf{x}_t, \mathbf{y}_t)$ conditioning on the value of $\mathbf{x}_{1:t-1}, \mathbf{y}_{1:t-1}$. Let's start with a simpler question: What is the conditional distribution of $\mathbf{x}_t$? Since the tuple is sampled from layout I, any constraint in I saying $\mathbf{x}_a[i] = \mathbf{x}_t[i]$ (i.e., if $i \in I_{a,t}$) affects the conditional distribution of $\mathbf{x}_t$. In more detail, the constraints on $\mathbf{x}_t$ can be formalized as[2]

$$I_c(i) = \begin{cases} \mathbf{x}_a[i] & \text{if } i \in I_{a,t} \text{ for some } a < t, \\ \perp & \text{otherwise.} \end{cases} \tag{8}$$

For each $i \in [k]$, if $I_c(i) \neq \perp$ then $\mathbf{x}_t[i]$ must equal to $I_c(i)$, otherwise $\mathbf{x}_t[i]$ is uniform in $\mathbb{F} \setminus \{\mathbf{x}_1[i], \dots, \mathbf{x}_{t-1}[i]\}$.

Inspired by the above discussion, we formally define *conditional layouts.* When conditioning on $\mathbf{x}_{1:t-1}$ and $\mathbf{y}_{1:t-1} = M\mathbf{x}_{1:t-1}$, define

$$S_i := \{\mathbf{x}_a[i] \mid a < t\}, \qquad T_j := \{\mathbf{y}_a[j] \mid a < t\} \qquad \text{for every } i, j \in [k].$$

A *conditional layout* for $\mathbf{x}_t$ is specified by a function $I_c : [k] \to \mathbb{F} \cup \{\perp\}$ such that $I_c(i) \in S_i \cup \{\perp\}$ for every $i \in [k]$. Define $\mathbf{x}_t$ is in I_c (denoted by $\mathbf{x}_t$ in I_c) and $\mathbf{x}_t$ satisfies I_c (denoted by $\mathbf{x}_t$ SAT I_c) as

$$\mathbf{x}_t \ \mathsf{in}\ I_c \iff \forall i \in [k], \begin{pmatrix} I_c(i) \neq \perp \implies \mathbf{x}_t[i] = I_c(i), \\ I_c(i) = \perp \implies \mathbf{x}_t[i] \notin S_i \end{pmatrix},$$

$$\mathbf{x}_t \ \mathsf{SAT}\ I_c \iff \forall i \in [k], \Big(I_c(i) \neq \perp \implies \mathbf{x}_t[i] = I_c(i)\Big).$$

We say I_c is the layout of $\mathbf{x}_t$, denoted by $\text{layout}_c(\mathbf{x}^{(t)}) = I_c$, if $\mathbf{x}_t \in I_c$. Define

$$\text{free}(I_c) := |I_c^{-1}(\perp)| = \#\{i \in [k] \text{ s.t. } I_c(i) = \perp\}$$

[2] Even if there exists distinct a, a' such that $i \in I_{a,t} \cap I_{a',t}$, I_c is still well-defined. Because in such case, we must have $i \in I_{a,a'}$ (otherwise I is not a valid layout), then $\mathbf{x}_a[i] = \mathbf{x}_{a'}[i]$.

as the number of coordinates that I_{c} outputs $\perp$. Note that, if I_{c} is derived from an α-nice layout I as in (8), then

$$\mathrm{free}(I_{\mathrm{c}}) = k - \Big|\bigcup_{a<t} I_{a,t}\Big| \geq (1-\alpha)k.$$

Define I'_{c} is stricter or equal to I_{c}, denoted by $I'_{\mathrm{c}} \supseteq I_{\mathrm{c}}$, as

$$I'_{\mathrm{c}} \supseteq I_{\mathrm{c}} \iff \forall i \in [k], \Big(I_{\mathrm{c}}(i) \neq \perp \implies I'_{\mathrm{c}}(i) = I_{\mathrm{c}}(i)\Big).$$

Symmetrically, a conditional layout for $\mathbf{y}_t$ is specified by a function $J_{\mathrm{c}} : [k] \to \mathbb{F} \cup \{\perp\}$ such that $J_{\mathrm{c}}(j) \in T_j \cup \{\perp\}$ for every $j \in [k]$. We adopt the same notations and terminology from the conditional layout of $\mathbf{x}_t$.

Let $\mathbf{y}^*$ be sampled uniformly at random from $\mathbb{F}^k$. Then

$$\Pr\Big[\mathrm{layout}_{\mathrm{c}}(\mathbf{y}^*) = J_{\mathrm{c}}\Big] = \sum_{\mathbf{y} \text{ in} J_{\mathrm{c}}} \frac{1}{2^{bk}} = \frac{\prod_{j\in[k] \text{ s.t. } J_{\mathrm{c}}(j)=\perp}(2^b - |T_j|)}{2^{bk}}. \tag{9}$$

We hope $\mathrm{layout}_{\mathrm{c}}(\mathbf{y}_t)$ is close to $\mathrm{layout}_{\mathrm{c}}(\mathbf{y}^*)$ by distribution. So we analyze the transition probability from I_{c} to J_{c}. That is, if $\mathbf{x}$ is sampled from layout I_{c}, what is the distribution of the layout of $\mathbf{y} = M\mathbf{x}$. We found that, if $\mathrm{free}(I_{\mathrm{c}})$ is large enough, the layout of $\mathbf{y}$ is close to the layout of random $\mathbf{y}^*$ by distribution.

Lemma 6. *Assume the linear mixing M has maximal branch number. Conditioning on any sets $S_1, \ldots, S_k, T_1, \ldots, T_k$, each of size at most $t-1$. For any conditional layout I_{c}, if $\mathbf{x}$ is sampled uniformly at random from layout I_{c} and let $\mathbf{y} := M\mathbf{x}$, then the statistical distance between* $\mathrm{layout}_{\mathrm{c}}(\mathbf{y})$ *and the conditional layout of a random vector is no greater than*

$$\Big(\frac{2t-1}{2^b}\Big)^{\mathrm{free}(I_{\mathrm{c}})}(2t-1)^k.$$

We start by bounding the transition probability. For any conditional layouts $I_{\mathrm{c}}, J_{\mathrm{c}}$, the transition probability from I_{c} to J_{c}, denoted by $\mathrm{trans\text{-}prob}(I_{\mathrm{c}}, J_{\mathrm{c}})$, is the probability $M\mathbf{x}$ in J_{c} when $\mathbf{x}$ is sampled from layout I_{c}. By definition,

$$\mathrm{trans\text{-}prob}(I_{\mathrm{c}}, J_{\mathrm{c}}) = \frac{\mathrm{trans\text{-}count}(I_{\mathrm{c}}, J_{\mathrm{c}})}{\text{size of layout } I_{\mathrm{c}}} = \frac{\displaystyle\sum_{\mathbf{x} \text{ in} I_{\mathrm{c}}} \sum_{\mathbf{y} \text{ in} J_{\mathrm{c}}} \mathbb{1}_M(\mathbf{x}, \mathbf{y})}{\displaystyle\sum_{\mathbf{x} \text{ in} I_{\mathrm{c}}} 1} \tag{10}$$

where $\mathbb{1}_M$ is defined as

$$\mathbb{1}_M(\mathbf{x}, \mathbf{y}) = \begin{cases} 1 & \text{if } M\mathbf{x} = \mathbf{y}, \\ 0 & \text{otherwise.} \end{cases}$$

We show in the following lemma that if $\mathrm{free}(I_{\mathrm{c}})$ is sufficiently large, then the transition probability $\mathrm{trans\text{-}prob}(I_{\mathrm{c}}, J_{\mathrm{c}})$ is close to the probability that random $\mathbf{y}^*$ lies in layout J_{c}.

Lemma 7. *Assume the linear mixing has maximal branch number. Conditioning on any sets $S_1, \ldots, S_k, T_1, \ldots, T_k$, each of size at most $t-1$. For any (conditional) layouts $I_{\mathrm{c}}, J_{\mathrm{c}}$, the transition probability from I_{c} to J_{c} is bounded by*

$$\Bigg|\mathrm{trans\text{-}prob}(I_{\mathrm{c}}, J_{\mathrm{c}}) - \sum_{\mathbf{y} \text{ in} J_{\mathrm{c}}} \frac{1}{2^{bk}}\Bigg| \leq \Big(\frac{2t-1}{2^b}\Big)^{\mathrm{free}(I_{\mathrm{c}})} t^{\mathrm{free}(J_{\mathrm{c}})}.$$

Proof. In the definition of transition probability (Equation (10)), the sum is over $\mathbf{x}$ in I_c, which is hard to analyze. But we know how $\sum_{\mathbf{x}\,\mathsf{SAT}\,I_\mathrm{c}}$ and $\sum_{\mathbf{x}\,\mathsf{in}\,I_\mathrm{c}}$ are closely connected. On the easy direction, we have

$$\sum_{\mathbf{x}\,\mathsf{SAT}\,I_\mathrm{c}} \equiv \sum_{I'_\mathrm{c}\supseteq I_\mathrm{c}} \sum_{\mathbf{x}\,\mathsf{in}\,I'_\mathrm{c}}, \qquad \sum_{\mathbf{y}\,\mathsf{SAT}\,J_\mathrm{c}} \equiv \sum_{J'_\mathrm{c}\supseteq J_\mathrm{c}} \sum_{\mathbf{y}\,\mathsf{in}\,J'_\mathrm{c}}.$$

Then by the inclusion-exclusion principle

$$\sum_{\mathbf{x}\,\mathsf{in}\,I_\mathrm{c}} \equiv \sum_{I'_\mathrm{c}\supseteq I_\mathrm{c}} (-1)^{\Delta(I'_\mathrm{c},I_\mathrm{c})} \sum_{\mathbf{x}\,\mathsf{SAT}\,I'_\mathrm{c}}, \qquad \sum_{\mathbf{y}\,\mathsf{in}\,J_\mathrm{c}} \equiv \sum_{J'_\mathrm{c}\supseteq J_\mathrm{c}} (-1)^{\Delta(J'_\mathrm{c},J_\mathrm{c})} \sum_{\mathbf{y}\,\mathsf{SAT}\,J'_\mathrm{c}},$$

where Δ denotes the Hamming distance. Since $I'_\mathrm{c} \supseteq I_\mathrm{c}$, the Hamming distance can also be written as $\Delta(I'_\mathrm{c}, I_\mathrm{c}) = \mathrm{free}(I_\mathrm{c}) - \mathrm{free}(I'_\mathrm{c})$.

We can apply the inclusion-exclusion principle to the numerator of (10),

$$\text{trans-count}(I_\mathrm{c}, J_\mathrm{c}) = \sum_{\mathbf{x}\,\mathsf{in}\,I_\mathrm{c}} \sum_{\mathbf{y}\,\mathsf{in}\,J_\mathrm{c}} \mathbb{1}_M(\mathbf{x},\mathbf{y}) = \sum_{I'_\mathrm{c}\supseteq I_\mathrm{c}} \sum_{J'_\mathrm{c}\supseteq J_\mathrm{c}} (-1)^{\Delta(I'_\mathrm{c},I_\mathrm{c})+\Delta(J'_\mathrm{c},J_\mathrm{c})} \sum_{\mathbf{x}\,\mathsf{SAT}\,I'_\mathrm{c}} \sum_{\mathbf{y}\,\mathsf{SAT}\,J'_\mathrm{c}} \mathbb{1}_M(\mathbf{x},\mathbf{y}).$$

As we have observed in previous sections, $\sum_{\mathbf{x}\,\mathsf{SAT}\,I'_\mathrm{c}} \sum_{\mathbf{y}\,\mathsf{SAT}\,J'_\mathrm{c}} \mathbb{1}_M(\mathbf{x},\mathbf{y})$ is easy to bound. Since the linear mixing has maximal branch number,

$$\sum_{\mathbf{x}\,\mathsf{SAT}\,I'_\mathrm{c}} \sum_{\mathbf{y}\,\mathsf{SAT}\,J'_\mathrm{c}} \mathbb{1}_M(\mathbf{x},\mathbf{y}) = \begin{cases} (2^b)^{\mathrm{free}(I'_\mathrm{c})+\mathrm{free}(J'_\mathrm{c})-k} & \text{if } \mathrm{free}(I'_\mathrm{c}) + \mathrm{free}(J'_\mathrm{c}) \geq k \\ 0 \text{ or } 1 & \text{otherwise.} \end{cases}$$

It can be approximated by

$$\sum_{\mathbf{x}\,\mathsf{SAT}\,I'_\mathrm{c}} \sum_{\mathbf{y}\,\mathsf{SAT}\,J'_\mathrm{c}} \frac{1}{2^{bk}} = (2^b)^{\mathrm{free}(I'_\mathrm{c})+\mathrm{free}(J'_\mathrm{c})-k},$$

such that the absolute value of the error is no more than 1 for any $I'_\mathrm{c}, J'_\mathrm{c}$.

As $\sum_{\mathbf{x}\,\mathsf{SAT}\,I'_\mathrm{c}} \sum_{\mathbf{y}\,\mathsf{SAT}\,J'_\mathrm{c}} \frac{1}{2^{bk}}$ is a good approximation of $\sum_{\mathbf{x}\,\mathsf{SAT}\,I'_\mathrm{c}} \sum_{\mathbf{y}\,\mathsf{SAT}\,J'_\mathrm{c}} \mathbb{1}_M(\mathbf{x},\mathbf{y})$ and the inclusion-exclusion principle has small coefficients, $\sum_{\mathbf{x}\,\mathsf{in}\,I'_\mathrm{c}} \sum_{\mathbf{y}\,\mathsf{in}\,J'_\mathrm{c}} \frac{1}{2^{bk}}$ should also be a fairly good approximation of trans-count$(I_\mathrm{c}, J_\mathrm{c})$.

$$\begin{aligned} &\left|\text{trans-count}(I_\mathrm{c}, J_\mathrm{c}) - \sum_{\mathbf{x}\,\mathsf{in}\,I_\mathrm{c}} \sum_{\mathbf{y}\,\mathsf{in}\,J_\mathrm{c}} \frac{1}{2^{bk}}\right| = \left|\sum_{\mathbf{x}\,\mathsf{in}\,I_\mathrm{c}} \sum_{\mathbf{y}\,\mathsf{in}\,J_\mathrm{c}} \left(\mathbb{1}_M(\mathbf{x},\mathbf{y}) - \frac{1}{2^{bk}}\right)\right| \\ &= \left|\sum_{I'_\mathrm{c}\supseteq I_\mathrm{c}} \sum_{J'_\mathrm{c}\supseteq J_\mathrm{c}} (-1)^{\Delta(I'_\mathrm{c},I_\mathrm{c})+\Delta(J'_\mathrm{c},J_\mathrm{c})} \sum_{\mathbf{x}\,\mathsf{SAT}\,I'_\mathrm{c}} \sum_{\mathbf{y}\,\mathsf{SAT}\,J'_\mathrm{c}} \left(\mathbb{1}_M(\mathbf{x},\mathbf{y}) - \frac{1}{2^{bk}}\right)\right| \\ &\leq \sum_{I'_\mathrm{c}\supseteq I_\mathrm{c}} \sum_{J'_\mathrm{c}\supseteq J_\mathrm{c}} 1 \leq t^{\mathrm{free}(I_\mathrm{c})+\mathrm{free}(J_\mathrm{c})}. \end{aligned} \tag{11}$$

This can be translated into a bound on the transition probability,

$$\left|\text{trans-prob}(I_\mathrm{c}, J_\mathrm{c}) - \frac{\displaystyle\sum_{\mathbf{x}\,\mathsf{in}\,I_\mathrm{c}} \sum_{\mathbf{y}\,\mathsf{in}\,J_\mathrm{c}} \frac{1}{2^{bk}}}{\sum_{\mathbf{x}\,\mathsf{in}\,I_\mathrm{c}} 1}\right| \leq \frac{t^{\mathrm{free}(I_\mathrm{c})+\mathrm{free}(J_\mathrm{c})}}{\sum_{\mathbf{x}\,\mathsf{in}\,I_\mathrm{c}} 1}.$$

In the fraction on the left-hand side, the $\sum_{\mathsf{x}\,\mathsf{in} I_\mathrm{c}} 1$ in the numerator and in the denominator can cancel out. So

$$\left|\text{trans-prob}(I_\mathrm{c}, J_\mathrm{c}) - \sum_{\mathsf{y}\,\mathsf{in} J_\mathrm{c}} \frac{1}{2^{bk}}\right| \le \frac{t^{\mathrm{free}(I_\mathrm{c})+\mathrm{free}(J_\mathrm{c})}}{\sum_{\mathsf{x}\,\mathsf{in} I_\mathrm{c}} 1} \le \frac{t^{\mathrm{free}(I_\mathrm{c})+\mathrm{free}(J_\mathrm{c})}}{(2^b-(t-1))^{\mathrm{free}(I_\mathrm{c})}} \le \left(\frac{2t-1}{2^b}\right)^{\mathrm{free}(I_\mathrm{c})} t^{\mathrm{free}(J_\mathrm{c})}.$$

The last inequality assumes $t \le 2^{b-1}$, we can assume this without loss of generality, because the lemma is trivialized otherwise. □

Now we can prove Lemma 6, by adding up the error term over all layouts J_c.

Proof of Lemma 6. The statistical distance between the conditional layout of $\mathbf{y}$ and the conditional layout of a random $\mathbf{y}^* \in \mathbb{F}^k$ is bounded by

$$\sum_{J_\mathrm{c}} \left(\frac{2t-1}{2^b}\right)^{\mathrm{free}(I_\mathrm{c})} t^{\mathrm{free}(J_\mathrm{c})} \le \left(\frac{2t-1}{2^b}\right)^{\mathrm{free}(I_\mathrm{c})} (2t-1)^k.$$

The inequality holds because

$$\sum_{J_\mathrm{c}} t^{\mathrm{free}(J_\mathrm{c})} = \sum_i \sum_{\substack{J_\mathrm{c}\text{ s.t.}\\ \mathrm{free}(J_\mathrm{c})=i}} t^i \le \sum_i \binom{k}{i}(t-1)^{k-i}t^i = (2t-1)^k.$$ □

We are now ready to complete our inductive proof of the core lemma (Lemma 5).

Proof of Lemma 5. Let $\mathbf{x}_{1:t} = (\mathbf{x}_1, \ldots, \mathbf{x}_t)$ be sampled uniformly from an α-nice layout I. We need to show that the layout of $\mathbf{y}_{1:t} := (M\mathbf{x}_1, \ldots, M\mathbf{x}_t)$ is statistically close to the layout of t random vectors.

Consider $\mathbf{x}_{1:t}^{(\mathsf{next})} = (\mathbf{x}_1^{(\mathrm{next})}, \ldots, \mathbf{x}_t^{(\mathrm{next})})$, which is obtained by applying k independent random S-boxes on $\mathbf{y}_{1:t}$. By Lemma 1, it is equivalent to study the statistical distance between $\mathbf{x}_{1:t}^{(\mathsf{next})}$ and t random vectors. Denote this statistical distance by $\varepsilon(t)$. Clearly $\varepsilon(1) = 0$.

For $t > 1$, assume the lemma holds for smaller t. By our definition of niceness, $\mathbf{x}_{1:t-1}$ is sampled from an α-nice layout I'. By the induction hypothesis, $\mathbf{x}_{1:t-1}^{(\mathrm{next})}$ is $\varepsilon(t-1)$-close to uniform by distribution. Implied by Lemma 6, the distribution of $\mathbf{x}_t^{(\mathrm{next})}$ conditioning on the values of $\mathbf{x}_{1:t-1}, \mathbf{y}_{1:t-1}, \mathbf{x}_{1:t-1}^{(\mathrm{next})}$ is very close to uniform. The (conditional) statistical distance is at most $(\frac{2t-1}{2^b})^{\mathrm{free}(I_\mathrm{c})}(2t-1)^k$ where I_c is determined by (8). Since I is α-nice, $\mathrm{free}(I_\mathrm{c}) \ge (1-\alpha)k$. Therefore, the statistical distance between $\mathbf{x}_{1:t}^{(\mathrm{next})}$ and t random vectors is bounded by

$$\varepsilon(t) \le \varepsilon(t-1) + \left(\frac{2t-1}{2^b}\right)^{(1-\alpha)k}(2t-1)^k.$$

By induction on t,

$$\varepsilon(t) \le \sum_{t'=2}^{t} \left(\frac{2t'-1}{2^b}\right)^{(1-\alpha)k}(2t'-1)^k \le t \cdot \left(\frac{2t}{2^b}\right)^{(1-\alpha)k}(2t)^k.$$ □

5.2 2-Round SPN* is $2^{-\Theta(bk)}$-Close to $O(1)$-Wise Independence

In this section, we use the core lemma (Lemma 5) to prove that a 2-round SPN* is $2^{-\Theta(bk)}$-close to t-wise independent, for constant t.

Theorem 2. *The 2-round SPN* is ε-close to t-wise independent, where*

$$\varepsilon = \frac{t^2 \cdot 2^{k+1}}{(2^b)^{k/(2t)}} + t \cdot \left(\frac{8 \cdot t^3}{2^b}\right)^{k/2},$$

if the linear mixing has maximal branch number. When t is a constant, the distance $\varepsilon = 2^{-\Theta(bk)}$.

The proof follows the same high-level framework introduced at the beginning of Section 5. We will show in Lemma 8 that for constant t, despite the staring tuple, the first-round tuple $\mathbf{y}^{(1)}_{1:t}$ will be in an α-nice layout with high probability. Thus, the core lemma (Lemma 5) implies that the layout of the second-round tuple $\mathbf{y}^{(2)}_{1:t}$ is exponentially close to t-wise independent.

Lemma 8. *For any $\alpha \in [0, 1]$ and any t-wise layout I, if tuple $\mathbf{x}_{1:t}$ is sampled uniformly from layout I, and let $\mathbf{y}_{1:t} = M\mathbf{x}_{1:t}$, then $J = \mathrm{layout}(\mathbf{y}_{1:t})$ is α-nice with probability*

$$\Pr[J \text{ is } \alpha\text{-nice}] \geq 1 - \frac{2^{k+1} \cdot t^2}{(2^b)^{\alpha k/t}}.$$

Proof. We will upper bound the probability that J is α-nice by requiring that each pair of vectors collide in at most $\alpha k/(t-1)$ coordinates. Then every vector collides with other vectors on at most αk coordinates, which implies that the layout of the tuple is α-nice.

The number of collisions between each pair of vectors can be bounded by Lemma 4, which does not depend on the starting layout. The probability $|J_{i,j}| > \alpha k/t$ is no more than $e \cdot 2^k/(2^b - 1)^{\alpha k/t}$.

$$\Pr\Big[J \text{ is not } \alpha\text{-nice}\Big] \leq \Pr\Bigg[\bigwedge_{1 \leq i < j \leq t} |J_{i,j}| > \frac{\alpha k}{t}\Bigg] \leq \frac{t^2 \cdot 2^{k+1}}{(2^b - 1)^{\alpha k/t}}$$

The last inequality is obtained by applying the union bound inequality over all $\binom{t}{2} \leq \frac{t^2}{2}$ pairs of vectors. □

We are now ready to present the proof of the main theorem of this section.

Theorem 2. Lemma 8 shows that

$$\varepsilon_1 := \Pr[J \text{ is not } \alpha\text{-nice}] \leq \frac{t^2 \cdot 2^{k+1}}{(2^b - 1)^{\alpha k/t}}.$$

Conditioning on J being α-nice, consider the (conditional) distribution of $\mathbf{y}^{(2)}_{1:t}$. The core lemma (Lemma 5) shows that the conditional distribution is ε_2-close to t-wise independent, for

$$\varepsilon_2 \leq t \cdot \left(\frac{(2t)^{2-\alpha}}{(2^b)^{1-\alpha}}\right)^k.$$

In conclusion, the output tuple $\mathbf{x}^{(3)}_{1:t}$, alias $\mathbf{y}^{(\mathsf{out})}_{1:t}$, is $(\varepsilon_1 + \varepsilon_2)$-close to t-wise independent. If we set $\alpha = \frac{1}{2}$, the statistical distance is bounded by

$$\varepsilon_1 + \varepsilon_2 \leq \frac{t^2 \cdot 2^{k+1}}{(2^b)^{k/(2t)}} + t \cdot \left(\frac{8 \cdot t^3}{2^b}\right)^{k/2}.$$

□

5.3 2-Round SPN* is $2^{-\Theta(b)}$-Close to t-Wise independent

This section shows a similar result for larger t. In particular, we prove that 2-round SPN* with a maximal-branch-number mixing is $2^{-\Theta(b)}$-close to t-wise independent, for t almost up to $2^{0.499b}$.

By applying the amplification result of Maurer, Pietrzak, and Renner [MPR07], we can reduce the error to $2^{-\Theta(bk)}$ by having $O(k)$ rounds.

Theorem 3. *For any $\alpha \in (0,1]$, the 2-round SPN* is ε-close to t-wise independent, where*

$$\varepsilon = \frac{t^2}{\alpha \cdot 2^b} + t \cdot \left(\frac{(2t)^{2-\alpha}}{(2^b)^{1-\alpha}}\right)^k,$$

if the mixing function has the maximal branch number.

If $t < 2^{(0.499-1/(4k))b}$, the distance is $\varepsilon = 2^{-\Theta(b)}$ by choosing the optimal α.

Corollary 1. *Assuming $t < 2^{(0.499-1/(4k))b}$, $\Theta(k)$-round SPN* with maximal-branch-number linear mixing is $2^{-\Theta(bk)}$-close to t-wise independent.*

The proof of this theorem is in the full version of the paper.

5.4 $(\log t)$-Rounds SPN* is $2^{-\Theta(bk)}$-Close to t-Wise Independent

In this section, we discuss how to achieve $2^{-\Theta(bk)}$-closeness to t-wise independent, for t up to $2^{0.499b}$, at the cost of a slightly larger number of rounds.

This result is proved by induction. The base case is closeness to 2-wise independent in 2 rounds. Assume that we have already shown ε-closeness to t-wise independent in r rounds. As the inductive step, we will prove the closeness to $(2t-1)$-wise independent in $r+1$ rounds.

As for notations, let $\mathbf{x}^{(\mathsf{in})}_{1:2t-1}$ denote $2t-1$ distinct inputs, let $\mathbf{y}^{(\mathsf{out})}_{1:2t-1}$ denote their corresponding outputs, and let $\mathbf{x}^{(\mathsf{last})}_{1:2t-1}, \mathbf{y}^{(\mathsf{last})}_{1:2t-1}$ denote the intermediate values in the last round (as illustrated in Fig. 3).

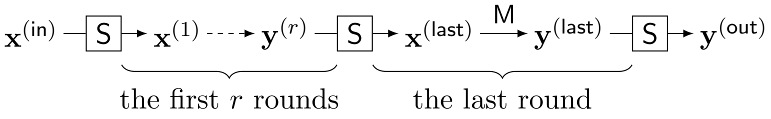
$\mathbf{x}^{(\mathsf{in})}$
S
$\mathbf{x}^{(1)}$
$\mathbf{y}^{(r)}$
S
$\mathbf{x}^{(\mathsf{last})}$
M
$\mathbf{y}^{(\mathsf{last})}$
S
$\mathbf{y}^{(\mathsf{out})}$
the first r rounds
the last round

Figure 3: Illustration of a $(r+1)$-round SPN

Due to the core lemma (Lemma 5), it suffices to show that: With overwhelming probability, $(\mathbf{x}^{(\mathsf{last})}_{1:2t-1})$ lies in a α-nice layout for some $\alpha \in (0,1)$ of our choice.

By the induction hypothesis, we know that the distribution of $\mathbf{x}^{(\mathsf{last})}_{1:t}$ is $\varepsilon(t)$-close to t-wise independent. If they are actually t-wise independent, then the probability $\mathbf{x}^{(\mathsf{last})}_{t}$ collides with $\mathbf{x}^{(\mathsf{last})}_{1:t-1}$ in more than $\alpha k/2$ coordinates is exponentially small due to Chernoff bound. The same argument also bounds the probability that $\mathbf{x}^{(\mathsf{last})}_{t}$ collides with $\mathbf{x}^{(\mathsf{last})}_{t+1:2t-1}$ in more than $\alpha k/2$ coordinates. Then the probability $\mathbf{x}^{(\mathsf{last})}_{t}$ collides with the other $2t-2$ vectors in at most αk coordinates is bounded by the union bound. Due to the symmetry and the union bound, $\mathbf{x}^{(\mathsf{last})}_{1:2t-1}$ is α-nice with good probability. Then we can finish the induction step by Lemma 5.

Such analysis can show $\varepsilon(t)$-closeness to t-wise independent in $O(\log t)$ rounds, where $\varepsilon(t)$ is inductively bounded by

$$\varepsilon(2t-1) \leq O(t) \cdot \Big(\varepsilon(t) + \underset{\text{from Chernoff bound}}{\text{a small term}}\Big) + \underset{\text{from Lemma 5}}{\text{another small term}}.$$

The $O(t)$ multiplicative factor before $\varepsilon(t)$ turns out to be problematic. It results in a multiplicative blow-up of order $t^{O(\log t)}$. When $t = 2^{\Theta(b)}$, this blow-up is about $2^{O(b^2)}$, which is unacceptable especially if $b = \Omega(k)$. In the actual proof of our result (Theorem 4), we conduct a more sophisticated analysis, though the high-level inductive idea is the same.

Theorem 4. *If $k > 4$, r-round SPN* is ε-close to 2^r-wise independent for*

$$\varepsilon = \frac{2^{r+\frac{3}{4}}}{1-2^{-\frac{k}{4}}} \cdot \left(\frac{2^{2r+3}}{2^b}\right)^{k/4} = \frac{t \cdot 2^{\frac{11}{4}}}{1-2^{-\frac{k}{4}}} \cdot \left(\frac{8 \cdot t^2}{2^b}\right)^{k/4}.$$

As usual, let $\mathbf{x}^{(r)}_{1:t}, \mathbf{y}^{(r)}_{1:t}$ denote the intermediate values in the r-th round. We also introduce a new notation $\mathbf{x}^{(r)}_{i,\times 2^\rho}$

$$\mathbf{x}^{(r)}_{i,\times 2^\rho} := \mathbf{x}^{(r)}_{i2^\rho+1:i2^\rho+2^\rho} = (\mathbf{x}^{(r)}_{i2^\rho+1}, \ldots, \mathbf{x}^{(r)}_{i2^\rho+2^\rho})$$

to denote 2^ρ consecutive vectors. Similarly we define $\mathbf{y}^{(r)}_{i,\times 2^\rho}$.

In the ρ-th round, for $0 \leq i < j < 2^{r-\rho}$, define $A^{(\rho)}_{i,j}$ as the event that

$$\Big(\underbrace{\mathbf{x}^{(\rho)}_{i2^{\rho-1}+1}, \ldots, \mathbf{x}^{(\rho)}_{i2^{\rho-1}+2^{\rho-1}}}_{\mathbf{x}^{(\rho)}_{i,\times 2^{\rho-1}}}, \underbrace{\mathbf{x}^{(\rho)}_{j2^{\rho-1}+1}, \ldots, \mathbf{x}^{(\rho)}_{j2^{\rho-1}+2^{\rho-1}}}_{\mathbf{x}^{(\rho)}_{j,\times 2^{\rho-1}}}\Big) \tag{12}$$

is in an α_ρ-nice layout. For $0 \leq i < j < 2^{r-\rho+1}$, define $B^{(\rho)}_{i,j}$ as the event that

$$\Big(\underbrace{\mathbf{x}^{(\rho)}_{i2^{\rho-2}+1}, \ldots, \mathbf{x}^{(\rho)}_{i2^{\rho-2}+2^{\rho-2}}}_{\mathbf{x}^{(\rho)}_{i,\times 2^{\rho-2}}}, \underbrace{\mathbf{x}^{(\rho)}_{j2^{\rho-2}+1}, \ldots, \mathbf{x}^{(\rho)}_{j2^{\rho-2}+2^{\rho-2}}}_{\mathbf{x}^{(\rho)}_{j,\times 2^{\rho-2}}}\Big) \tag{13}$$

is in a $\frac{1}{3}\alpha_\rho$-nice layout. The value of α_ρ will be fixed later.

The proof of Theorem 4 is inductive. The induction hypothesis is that with overwhelming probability $\bigwedge_{0\leq i<j<2^{r-\rho}} A^{(\rho)}_{i,j}$ holds. Then by Lemma 5, the joint distribution of $\mathbf{x}^{(\rho)}_{i,\times 2^{\rho-1}}, \mathbf{x}^{(\rho)}_{j,\times 2^{\rho-1}}$ is close to 2^ρ-wise uniform, for each $0 \leq i < j < 2^{r-\rho}$. Then by the following Lemma 9, they are very likely to be $\frac{1}{3}\alpha_{\rho+1}$-nice, that is, $B^{(\rho+1)}_{i,j}$ is likely to hold. To complete the induction step, we bridge the remaining gap by proving the following statement for $\rho > 2$,

$$\bigwedge_{0\leq i<j<2^{r-\rho+1}} B^{(\rho)}_{i,j} \implies \bigwedge_{0\leq i<j<2^{r-\rho}} A^{(\rho)}_{i,j}. \tag{14}$$

Lemma 9. *Assume $\mathbf{x}_{1:t}$ are uniformly sampled from $(\mathbb{F}^k)^t$, for any $\alpha > \frac{t-1}{2^b}$,*

$$\Pr\big[\text{layout}(\mathbf{x}_1, \ldots, \mathbf{x}_t) \textit{ is } \alpha\textit{-nice}\big] \geq 1 - \frac{t \cdot 2^k}{1+\alpha k} \cdot \left(\frac{t}{2^b}\right)^{\alpha k}.$$

The proofs of statement (14) and of Lemma 9 are deferred to the full version of the paper.

Now we are nearly ready to prove Theorem 4. We introduce a few additional notations. For $\rho \geq 1$, define

$$A_\rho := \bigwedge_{0 \leq i < j < 2^{r-\rho}} A_{i,j}^{(\rho)}, \qquad\qquad \delta_\rho := 1 - \Pr[A_\rho].$$

Define $\varepsilon_{\rho,i,j}$ as the statistical distance between the uniform distribution and the distribution of $\mathbf{x}_{i,\times 2^{\rho-1}}^{(\rho+1)}, \mathbf{x}_{j,\times 2^{\rho-1}}^{(\rho+1)}$ (the vectors in the definition of $B_{i,j}^{(\rho+1)}$) *conditioning on event* A_ρ. Lemma 5 shows that

$$\varepsilon_{\rho,i,j} \leq 2^\rho \cdot \left(\frac{2^{\rho+1}}{2^b}\right)^{(1-\alpha_\rho)k} (2^{\rho+1})^k.$$

for all $\rho \geq 2$. Define $\varepsilon_\rho = \sum_{0 \leq i < j < 2^{r-\rho}} \varepsilon_{\rho,i,j}$.

Note that $\varepsilon_r = \varepsilon_{r,1,2}$ is the statistical distance between the 2^r output vectors and uniform, conditioning on A_r. So r-round SPN* is $(\delta_r + \varepsilon_r)$-close to 2^r-wise independent.

Theorem 4. For each $2 < \rho \leq r$, conditional on $A_{\rho-1}$, the (conditional) distribution of $\mathbf{x}_{i,\times 2^{\rho-2}}^{(\rho)}, \mathbf{x}_{j,\times 2^{\rho-2}}^{(\rho)}$ is $\varepsilon_{\rho-1,i,j}$-close to uniform. Then by Lemma 9

$$\Pr\left[\neg B_{i,j}^{(\rho)} \;\middle|\; A_{\rho-1}\right] \leq \varepsilon_{\rho-1,i,j} + 2^{\rho-1} \cdot 2^k \cdot \left(\frac{2^{\rho-1}}{2^b}\right)^{\frac{1}{3}\alpha_\rho k}.$$

By the union bound,

$$\Pr\left[\neg \bigwedge_{0 \leq i < j < 2^{r-\rho+1}} B_{i,j}^{(\rho)} \;\middle|\; A_{\rho-1}\right] \leq \varepsilon_{\rho-1} + \frac{(2^{r-\rho+1})^2}{2} \cdot 2^{\rho-1} \cdot 2^k \cdot \left(\frac{2^{\rho-1}}{2^b}\right)^{\frac{1}{3}\alpha_\rho k}.$$

By (14), the left-hand side is lower bounded by $\Pr\left[\neg A_\rho \;\middle|\; A_{\rho-1}\right]$. And we know

$$\Pr\left[\neg A_\rho \;\middle|\; A_{\rho-1}\right] \geq \Pr\left[\neg A_\rho \wedge A_{\rho-1}\right] \geq \delta_\rho - \delta_{\rho-1}.$$

So

$$\delta_\rho \leq \delta_{\rho-1} + \varepsilon_{\rho-1} + \frac{(2^{r-\rho+1})^2}{2} 2^{\rho-1} \cdot 2^k \cdot \left(\frac{2^{\rho-1}}{2^b}\right)^{\frac{1}{3}\alpha_\rho k}.$$

For the base case $\rho = 2$, Lemma 4 directly bounds the probability of $B_{i,j}^{(2)}$ by $\frac{e \cdot 2^k}{(2^b-1)^{\frac{1}{3}\alpha_2 k}}$. Then by the union bound

$$\delta_2 \leq \Pr\left[\neg \bigwedge_{i,j} B_{i,j}^{(2)}\right] \leq \frac{(2^r)^2}{2} \frac{e \cdot 2^k}{(2^b-1)^{\frac{1}{3}\alpha_2 k}}.$$

As the final goal is to bound $\delta_r + \varepsilon_r$, we are interested in how $\delta_\rho + \varepsilon_\rho$ depends on $\delta_{\rho-1} + \varepsilon_{\rho-1}$,

$$\begin{aligned} &(\delta_\rho + \varepsilon_\rho) - (\delta_{\rho-1} + \varepsilon_{\rho-1}) \\ &\leq \frac{(2^{r-\rho})^2}{2} 2^\rho \cdot \left(\frac{2^{\rho+1}}{2^b}\right)^{(1-\alpha_\rho)k} (2^{\rho+1})^k + \frac{(2^{r-\rho+1})^2}{2} 2^{\rho-1} \cdot 2^k \cdot \left(\frac{2^{\rho-1}}{2^b}\right)^{\frac{1}{3}\alpha_\rho k} \\ &= 2^{2r-\rho-1}\left(\left(\frac{2^{\rho+1}}{2^b}\right)^{(1-\alpha_\rho)k} (2^{\rho+1})^k + 2^{k+1} \cdot \left(\frac{2^{\rho-1}}{2^b}\right)^{\frac{1}{3}\alpha_\rho k}\right). \end{aligned} \tag{15}$$

Number of rounds r	$\log_2$(TV distance from 2-wise ind.)
3	-23.4275
4	-48.9916
5	-117.1745
6	-126.3073
7	-141.2575

Table 2: Statistical (TV) distance from pairwise independence of the r-round AES* given two inputs that differ in exactly one coordinate. This corresponds to starting from a layout I with Hamming weight 1, e.g. $I = \{1, \dots, k-1\}$.

The value of α_ρ should be chosen so that (15) is minimized. Note that

$$\text{Right-hand side of (15)} \approx \left(\frac{2^\rho}{2^b}\right)^{-\alpha_\rho k} \left(\frac{2^{2\rho}}{2^b}\right)^{k} + \left(\frac{2^\rho}{2^b}\right)^{\frac{1}{3}\alpha_\rho k}$$

so (15) is minimized when $\alpha \approx \frac{3}{4}\frac{b-2\rho}{b-\rho}$, and the minimum value is about $\left(\frac{2^{2\rho}}{2^b}\right)^{k/4}$. If we tune the value of α_ρ, we get

$$(\delta_\rho + \varepsilon_\rho) - (\delta_{\rho-1} + \varepsilon_{\rho-1}) \le 2^{2r-\rho} \cdot 2^{\frac{1}{2}k\rho - \frac{1}{4}kb + \frac{3}{4}k + \frac{3}{4}} = 2^{2r-\rho+\frac{3}{4}} \cdot \left(\frac{2^{2\rho+3}}{2^b}\right)^{k/4}.$$

We defer the analysis of the base case to the full version.

$$\delta_r + \varepsilon_r \le \delta_2 + \varepsilon_2 + \sum_{\rho=3}^{r} 2^{2r-\rho+\frac{3}{4}} \cdot \left(\frac{2^{2\rho+3}}{2^b}\right)^{k/4} \le \frac{2^{r+\frac{3}{4}}}{1 - 2^{-\frac{k}{4}}} \cdot \left(\frac{2^{2r+3}}{2^b}\right)^{k/4}. \qquad \square$$

6 Pairwise Independence of AES* and Censored AES

In this section, we obtain concrete bounds on the pairwise independence of (1) an SPN cipher with random, independent S-boxes and the *actual* AES mixing (we refer to this as AES*) as well as (2) a "censored" version of the actual AES block cipher (with the *actual* AES S-box, but some mixing layers removed). We will use partially computational methods for our theorems. The source code for our computations is available at `https://github.com/AnPelec/t-wise-ind-SPN`.

6.1 Pairwise independence of AES*

We can represent the evaluation of AES* as a Markov chain over $2^{16} - 1$ layouts. Our goal is to describe this random walk exactly, and then use numerical calculations to infer an upper bound on the statistical distance of an output pair after a certain number of rounds. To compute the transition probabilities, we start with an exact version of Lemma 2. A similar lemma was already proved in [BV06], by relating the number of transitions to the number of codewords of specific weight in an MDS code.

Lemma 10. *If M has the maximal branch number, the layout transition probability* trans-prob$(I, J) :=$ $\Pr_{\mathbf{x}\,\mathsf{in}\,I}\big[M\mathbf{x} \text{ in } J\big]$ *equals*

$$\text{trans-prob}(I, J) = \sum_{i=0}^{\text{free}(I)+\text{free}(J)-k-1} (-1)^i \, \frac{\binom{k-1+i}{k-1}}{(2^b-1)^{k-\text{free}(J)+i}}. \tag{16}$$

Lemma 10 assumes however a full-branch mixing layer, which is not the case for AES mixing. Another issue is that the number of layouts is still quite high and poses a non-trivial computational challenge. Thankfully, we can overcome this obstacle by representing the AES mixing layer in terms of permutations and full-branch mixings, an observation first made by [BV06]. More details can be found in Appendix B.

As our starting point, we numerically compute the total variation distance from uniform after r rounds starting with a pair of inputs that differ in exactly one 8-bit word. The results are summarized in Table 2, and are obtained by computing the corresponding r-th power of the transition matrix of the random walk. (This requires leveraging a number of symmetries to be computationally feasible.)

We then derive conjectures on the maximum distance over all possible input layouts and verify that our conjectures hold by computing the statistical distance for all input layouts. As a result of this, we obtain the following theorems.

Theorem 5. *The 3-round AES* is $2^{-23.42}$-close to pairwise independent.*

Theorem 6. *The 7-round AES* is 2^{-128}-close to pairwise independent.*

6.2 Censored AES

To translate our results from the random S-box setting to the AES S-box, we replace a random S-box by consecutive applications of the AES one, namely the patched inverse function over $\mathbb{F}_{2^8}$ where the input is XOR with a fresh key byte. Note that the resulting SPN which we refer to as "censored" AES is simply AES with several mixing layers removed.

We numerically compute the closeness to pairwise independence of the sequential composition of AES S-boxes over $\mathbb{F}_{2^8}$, where a fresh key byte is XORed into the input prior to each call. These distances can be found in Table 3. Note that analytical bounds were obtained in [LTV21], however here we obtain tighter numerical bounds for our parameter settings. We defer the implementation details to Appendix C.

Overall, we prove the following theorem. It considers what we (informally) refer to as "192-round censored AES." One should think of this as a 191-round SPN (thus with 192 layers of S-boxes), with independent keys, using the true AES S-box (patched inverse) and the AES mixing layer, but with a subset of mixing layers removed. Which mixing layers remain can be inferred from the proof below.

Theorem 7. *192-round censored AES is 2^{-128}-close to pairwise independent.*

Proof. First off, Theorem 5 implies that 3-round AES* (that is, 4 layers of random S-boxes) is $\varepsilon_{ideal} = 2^{-23.42}$-close to pairwise independent. We then replace each random S-box with the sequential composition of c consecutive AES S-boxes (and xoring an independent uniform key byte to each call) and show that the resulting construction (which consists of $4c$ layers of S-boxes) is ϵ-close to pairwise independent, for some suitable ϵ. This value of ϵ will be then amplified, via further sequential composition. By the amplification theorem of [KNR09a, MPR07], the resulting $4cr$-round censored AES is in particular $(2^{r-1}\epsilon^r)$-close to pairwise independent. The exact constants c and r are chosen to optimize the final number of rounds required to reach 2^{-128}-closeness.

First of all, we pick $c = 8$. Indeed, according to Table 3,the 8-fold sequential composition of the S-box (with independent key bytes XORed to each S-box input) is $\varepsilon_{sim} \leq 2^{-29.39}$-close to pairwise independent, and hence to the behavior of a random S-box. Recall that the random S-box in AES* is applied to $k = 16$ blocks in parallel, hence by the triangle inequality we deduce that we can simulate 4 random S-box layers with an error of at most $16 \cdot 4 \cdot \varepsilon_{sim} \leq 2^{-23.39}$.

Therefore, we conclude that this partial 32-round censored AES is ϵ-close to pairwise independent for

$$\epsilon \leq \varepsilon_{ideal} + 16 \cdot 4 \cdot \varepsilon_{sim} \leq 2^{-23.42} + 2^{-23.39} < 2^{-22.39} \; .$$

Then, amplification for $r = 6$ repetitions gives that the 192-round censored AES is

$$2^5 \cdot (2^{-22.39})^6 = 2^{5-22.39 \cdot 6} < 2^{-128}$$

close to pairwise independent. □ □

If one believes that the mixing layers are useful for AES to achieve pseudorandomness, then it is natural to expect that removing a large fraction of them should only hurt the convergence to pairwise independence. This leads us to conjecture that 192-round AES is 2^{-128}-close to pairwise independent. We view proving this conjecture formally to be an outstanding open problem.

Acknowledgements.

Pelecanos was supported by DARPA under Agreement No. HR00112020023. Tessaro was supported in part by NSF grants CNS-2026774, CNS-2154174, a JP Morgan Faculty Award, a CISCO Faculty Award, and a gift from Microsoft. Vaikuntanathan was supported by DARPA under Agreement No. HR00112020023, NSF CNS-2154149, and a Thornton Family Faculty Research Innovation Fellowship.

References

[ABD+13] Elena Andreeva, Andrey Bogdanov, Yevgeniy Dodis, Bart Mennink, and John P. Steinberger. On the indifferentiability of key-alternating ciphers. In Ran Canetti and Juan A. Garay, editors, *CRYPTO 2013, Part I*, volume 8042 of *LNCS*, pages 531–550. Springer, Heidelberg, August 2013. pages 7

[AES01] Advanced Encryption Standard (AES). National Institute of Standards and Technology, NIST FIPS PUB 197, U.S. Department of Commerce, November 2001. pages 5

[BH08] Alex Brodsky and Shlomo Hoory. Simple permutations mix even better. *Random Struct. Algorithms*, 32(3):274–289, 2008. pages 3

[BKL$^+$12] Andrey Bogdanov, Lars R. Knudsen, Gregor Leander, François-Xavier Standaert, John P. Steinberger, and Elmar Tischhauser. Key-alternating ciphers in a provable setting: Encryption using a small number of public permutations - (extended abstract). In David Pointcheval and Thomas Johansson, editors, *EUROCRYPT 2012*, volume 7237 of *LNCS*, pages 45–62. Springer, Heidelberg, April 2012. pages 7

[BKR11] Andrey Bogdanov, Dmitry Khovratovich, and Christian Rechberger. Biclique cryptanalysis of the full AES. In Dong Hoon Lee and Xiaoyun Wang, editors, *ASIACRYPT 2011*, volume 7073 of *LNCS*, pages 344–371. Springer, Heidelberg, December 2011. pages 3

[BS91] Eli Biham and Adi Shamir. Differential cryptanalysis of des-like cryptosystems. *J. Cryptol.*, 4(1):3–72, 1991. pages 3

[BV05] Thomas Baignères and Serge Vaudenay. Proving the security of AES substitution-permutation network. In Bart Preneel and Stafford E. Tavares, editors, *Selected Areas in Cryptography, 12th International Workshop, SAC 2005, Kingston, ON, Canada, August 11-12, 2005, Revised Selected Papers*, volume 3897 of *Lecture Notes in Computer Science*, pages 65–81. Springer, 2005. pages 6, 11

[BV06] Thomas Baignères and Serge Vaudenay. Proving the security of AES substitution-permutation network. In Bart Preneel and Stafford Tavares, editors, *SAC 2005*, volume 3897 of *LNCS*, pages 65–81. Springer, Heidelberg, August 2006. pages 3, 5, 6, 27, 28, 35

[Car53] L. Carlitz. Permutations in a finite field. *Proceedings of American Mathematical Society*, page 538, 1953. pages 38

[CDK$^+$18] Benoît Cogliati, Yevgeniy Dodis, Jonathan Katz, Jooyoung Lee, John P. Steinberger, Aishwarya Thiruvengadam, and Zhe Zhang. Provable security of (tweakable) block ciphers based on substitution-permutation networks. In Hovav Shacham and Alexandra Boldyreva, editors, *CRYPTO 2018, Part I*, volume 10991 of *LNCS*, pages 722–753. Springer, Heidelberg, August 2018. pages 3, 7

[CHK$^+$16] Jean-Sébastien Coron, Thomas Holenstein, Robin Künzler, Jacques Patarin, Yannick Seurin, and Stefano Tessaro. How to build an ideal cipher: The indifferentiability of the Feistel construction. *Journal of Cryptology*, 29(1):61–114, January 2016. pages 7

[CLL$^+$14] Shan Chen, Rodolphe Lampe, Jooyoung Lee, Yannick Seurin, and John P. Steinberger. Minimizing the two-round Even-Mansour cipher. In Juan A. Garay and Rosario Gennaro, editors, *CRYPTO 2014, Part I*, volume 8616 of *LNCS*, pages 39–56. Springer, Heidelberg, August 2014. pages 7

[CP02] Nicolas Courtois and Josef Pieprzyk. Cryptanalysis of block ciphers with overdefined systems of equations. In Yuliang Zheng, editor, *ASIACRYPT 2002*, volume 2501 of *LNCS*, pages 267–287. Springer, Heidelberg, December 2002. pages 3

[CS14] Shan Chen and John P. Steinberger. Tight security bounds for key-alternating ciphers. In Phong Q. Nguyen and Elisabeth Oswald, editors, *EUROCRYPT 2014*, volume 8441 of *LNCS*, pages 327–350. Springer, Heidelberg, May 2014. pages 7

[CS15] Benoit Cogliati and Yannick Seurin. On the provable security of the iterated Even-Mansour cipher against related-key and chosen-key attacks. In Elisabeth Oswald and Marc Fischlin, editors, *EUROCRYPT 2015, Part I*, volume 9056 of *LNCS*, pages 584–613. Springer, Heidelberg, April 2015. pages 7

[Dae95] Joan Daemen. Cipher and hash function design strategies based on linear and differential cryptanalysis. *Ph.D. Thesis, KU Leuven*, 1995. pages 10, 34

[DGJM06] Martin Dyer, Leslie Ann Goldberg, Mark Jerrum, and Russell Martin. Markov chain comparison. *Probability Surveys*, 3(none), jan 2006. pages 46, 47

[DKS$^+$17] Yevgeniy Dodis, Jonathan Katz, John Steinberger, Aishwarya Thiruvengadam, and Zhe Zhang. Provable security of substitution-permutation networks. Cryptology ePrint Archive, Report 2017/016, 2017. https://eprint.iacr.org/2017/016. pages 3, 7

[DKW22] Yevgeniy Dodis, Harish Karthikeyan, and Daniel Wichs. Small-box cryptography. In Mark Braverman, editor, *13th Innovations in Theoretical Computer Science Conference, ITCS 2022, January 31 - February 3, 2022, Berkeley, CA, USA*, volume 215 of *LIPIcs*, pages 56:1–56:25. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2022. pages 7

[DR02] Joan Daemen and Vincent Rijmen. *The Design of Rijndael: AES - The Advanced Encryption Standard*. Information Security and Cryptography. Springer, 2002. pages 11

[DS16] Yuanxi Dai and John P. Steinberger. Indifferentiability of 8-round Feistel networks. In Matthew Robshaw and Jonathan Katz, editors, *CRYPTO 2016, Part I*, volume 9814 of *LNCS*, pages 95–120. Springer, Heidelberg, August 2016. pages 7

[DSC93] Persi Diaconis and Laurent Saloff-Coste. Comparison Theorems for Reversible Markov Chains. *The Annals of Applied Probability*, 3(3):696 – 730, 1993. pages 37

[DSSL16] Yevgeniy Dodis, Martijn Stam, John P. Steinberger, and Tianren Liu. Indifferentiability of confusion-diffusion networks. In Marc Fischlin and Jean-Sébastien Coron, editors, *EUROCRYPT 2016, Part II*, volume 9666 of *LNCS*, pages 679–704. Springer, Heidelberg, May 2016. pages 7

[DSST17] Yuanxi Dai, Yannick Seurin, John P. Steinberger, and Aishwarya Thiruvengadam. Indifferentiability of iterated Even-Mansour ciphers with non-idealized key-schedules: Five rounds are necessary and sufficient. In Jonathan Katz and Hovav Shacham, editors, *CRYPTO 2017, Part III*, volume 10403 of *LNCS*, pages 524–555. Springer, Heidelberg, August 2017. pages 7

[EM97] Shimon Even and Yishay Mansour. A construction of a cipher from a single pseudorandom permutation. *Journal of Cryptology*, 10(3):151–162, June 1997. pages 7

[FP15] Pooya Farshim and Gordon Procter. The related-key security of iterated Even-Mansour ciphers. In Gregor Leander, editor, *FSE 2015*, volume 9054 of *LNCS*, pages 342–363. Springer, Heidelberg, March 2015. pages 7

[Fri00] Joel Friedman. On cayley graphs on the symmetric group generated by tranpositions. *Combinatorica*, 20(4):505–519, 2000. pages 48

[GL15] Chun Guo and Dongdai Lin. On the indifferentiability of key-alternating Feistel ciphers with no key derivation. In Yevgeniy Dodis and Jesper Buus Nielsen, editors, *TCC 2015, Part I*, volume 9014 of *LNCS*, pages 110–133. Springer, Heidelberg, March 2015. pages 7

[HMMR05] Shlomo Hoory, Avner Magen, Steven A. Myers, and Charles Rackoff. Simple permutations mix well. *Theor. Comput. Sci.*, 348(2-3):251–261, 2005. pages 3

[HT16] Viet Tung Hoang and Stefano Tessaro. Key-alternating ciphers and key-length extension: Exact bounds and multi-user security. In Matthew Robshaw and Jonathan Katz, editors, *CRYPTO 2016, Part I*, volume 9814 of *LNCS*, pages 3–32. Springer, Heidelberg, August 2016. pages 7

[JK97] Thomas Jakobsen and Lars R. Knudsen. The interpolation attack on block ciphers. In *FSE*, volume 1267 of *Lecture Notes in Computer Science*, pages 28–40. Springer, 1997. pages 3

[KHL$^+$02] Ju-Sung Kang, Seokhie Hong, Sangjin Lee, Okyeon Yi, Choonsik Park, and Jongin Lim. Practical and provable security against differential and linear cryptanalysis for substitution-permutation networks. *Etri Journal*, 23, 02 2002. pages 10, 34

[KNR09a] Eyal Kaplan, Moni Naor, and Omer Reingold. Derandomized constructions of k-wise (almost) independent permutations. *Algorithmica*, 55(1):113–133, 2009. pages 4, 6, 29

[KNR09b] Eyal Kaplan, Moni Naor, and Omer Reingold. Derandomized constructions of k-wise (almost) independent permutations. *Algorithmica*, 55(1):113–133, 2009. pages 7

[Knu94] Lars R. Knudsen. Truncated and higher order differentials. In Bart Preneel, editor, *Fast Software Encryption: Second International Workshop. Leuven, Belgium, 14-16 December 1994, Proceedings*, volume 1008 of *Lecture Notes in Computer Science*, pages 196–211. Springer, 1994. pages 3

[Knu98] Lars Knudsen. Deal - a 128-bit block cipher. In *NIST AES Proposal*, 1998. pages 3

[KW02] Lars R. Knudsen and David A. Wagner. Integral cryptanalysis. In *FSE*, volume 2365 of *Lecture Notes in Computer Science*, pages 112–127. Springer, 2002. pages 3

[Lai94] Xuejia Lai. *Higher Order Derivatives and Differential Cryptanalysis*, pages 227–233. Springer US, Boston, MA, 1994. pages 3

[LMM91] Xuejia Lai, James L. Massey, and Sean Murphy. Markov ciphers and differential cryptanalysis. In *EUROCRYPT*, volume 547 of *Lecture Notes in Computer Science*, pages 17–38. Springer, 1991. pages 10

[LPS12] Rodolphe Lampe, Jacques Patarin, and Yannick Seurin. An asymptotically tight security analysis of the iterated Even-Mansour cipher. In Xiaoyun Wang and Kazue Sako, editors, *ASIACRYPT 2012*, volume 7658 of *LNCS*, pages 278–295. Springer, Heidelberg, December 2012. pages 7

[LS15] Rodolphe Lampe and Yannick Seurin. Security analysis of key-alternating Feistel ciphers. In Carlos Cid and Christian Rechberger, editors, *FSE 2014*, volume 8540 of *LNCS*, pages 243–264. Springer, Heidelberg, March 2015. pages 7

[LTV21] Tianren Liu, Stefano Tessaro, and Vinod Vaikuntanathan. The t-wise independence of substitution-permutation networks. *CRYPTO*, 2021. pages 3, 5, 7, 28

[MPR07] Ueli M. Maurer, Krzysztof Pietrzak, and Renato Renner. Indistinguishability amplification. In Alfred Menezes, editor, *Advances in Cryptology - CRYPTO 2007, 27th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2007, Proceedings*, volume 4622 of *Lecture Notes in Computer Science*, pages 130–149. Springer, 2007. pages 4, 6, 7, 24, 29

[MV12] Eric Miles and Emanuele Viola. Substitution-permutation networks, pseudorandom functions, and natural proofs. In Reihaneh Safavi-Naini and Ran Canetti, editors, *CRYPTO 2012*, volume 7417 of *LNCS*, pages 68–85. Springer, Heidelberg, August 2012. pages 3, 7

[MY92] Mitsuru Matsui and Atsuhiro Yamagishi. A new method for known plaintext attack of FEAL cipher. In *EUROCRYPT*, volume 658 of *Lecture Notes in Computer Science*, pages 81–91. Springer, 1992. pages 3

[Nat89] National Soviet Bureau of Standards. Information processing system – cryptographic protection – cryptographic algorithm gost 28147-89, 1989. pages 3

[Nyb93] Kaisa Nyberg. Differentially uniform mappings for cryptography. In Tor Helleseth, editor, *Advances in Cryptology - EUROCRYPT '93, Workshop on the Theory and Application of of Cryptographic Techniques, Lofthus, Norway, May 23-27, 1993, Proceedings*, volume 765 of *Lecture Notes in Computer Science*, pages 55–64. Springer, 1993. pages 10

[Pat08] Jacques Patarin. A proof of security in $O(2^n)$ for the Benes scheme. In Serge Vaudenay, editor, *AFRICACRYPT 08*, volume 5023 of *LNCS*, pages 209–220. Springer, Heidelberg, June 2008. pages 7

[Tes15] Stefano Tessaro. Optimally secure block ciphers from ideal primitives. In Tetsu Iwata and Jung Hee Cheon, editors, *ASIACRYPT 2015, Part II*, volume 9453 of *LNCS*, pages 437–462. Springer, Heidelberg, November / December 2015. pages 7

[TKKL15] Tyge Tiessen, Lars R. Knudsen, Stefan Kölbl, and Martin M. Lauridsen. Security of the AES with a secret S-box. In Gregor Leander, editor, *FSE 2015*, volume 9054 of *LNCS*, pages 175–189. Springer, Heidelberg, March 2015. pages 3

[Zie13] Michael E. Zieve. On a theorem of carlitz. *Journal of Group Theory*, 17(4):667–669, nov 2013. pages 38

A Maximum Branch Number

As the diffusion step of a SPN, having the *maximal branch number* is considered a desirable feature for mixing functions [Dae95, KHL+02]. Such linear function is also known as MDS (maximum distance separable) matrix. This section recaps a few useful consequences of a mixing function with the maximal branch number.

Lemma 11. *If the linear mapping defined by matrix $M \in \mathbb{F}^{k\times k}$ has the maximal branch number, then all entries of M are non-zero.*

Proof. Proof by contradiction. Assume M has zero entry. W.l.o.g., assume $M_{1,1} = 0$. Consider the vector $\mathbf{x} = (1, 0, \ldots, 0)$ that is zero everywhere except the first coordinate. Then $M\mathbf{x}$ is the first column of M, so $\mathsf{wt}(M\mathbf{x}) \le k - 1$. The branch number of M is no more than $\mathsf{wt}(\mathbf{x}) + \mathsf{wt}(M\mathbf{x}) = k$. □

Lemma 12. *If the linear mapping defined by matrix $M \in \mathbb{F}^{k\times k}$ has the maximal branch number, then $k < |\mathbb{F}|$.*

Proof. Proof by contradiction, assume $k \ge |\mathbb{F}|$. Consider $M_{i,1}/M_{i,2}$ for all $i \in [k]$. Their values lie in $\mathbb{F} \setminus \{0\}$ (Lemma 11). By the pigeonhole principle, there exist distinct i, j such that $M_{i,1}/M_{i,2} = M_{j,1}/M_{j,2}$. Denote the ratio by c. Define vector $\mathbf{x}$ that is zero everywhere except $\mathbf{x}[1] = 1$ and $\mathbf{x}[2] = -c$. Then the i-th and j-th entry of $M\mathbf{x}$ equal 0. The branch number of M is no more than $\mathsf{wt}(\mathbf{x}) + \mathsf{wt}(M\mathbf{x}) = k$. □

For any $I, J \subseteq [k]$, the set

$$\mathcal{L}^M_{I,J} := \left\{\mathbf{x} \;\middle|\; \forall i \in I, \mathbf{x}[i] = 0 \wedge \forall j \in J, (M\mathbf{x})[j] = 0\right\} \tag{17}$$

is a linear subspace of $\mathbb{F}^k$. Then for any $I' \supseteq I$ and $J' \supseteq J$,

$$\begin{aligned} \dim \mathcal{L}^M_{I',J'} &\le \dim \mathcal{L}^M_{I,J}, \\ \dim \mathcal{L}^M_{I',J'} &\ge \dim \mathcal{L}^M_{I,J} - |I' \setminus I| - |J' \setminus J|. \end{aligned} \tag{18}$$

The first says adding constraints cannot increase the solution space dimension. The second says adding one constraint can decrease the solution space by at most 1.

When the matrix M has the maximal branch number, the dimension of the solution space has a clean expression.

Lemma 13. *If the linear mapping defined by matrix $M \in \mathbb{F}^{k\times k}$ has the maximal branch number, then for any $I, J \subseteq [k]$,*

$$\dim \mathcal{L}^M_{I,J} = \max(k - |I| - |J|, 0),$$

where $\mathcal{L}^M_{I,J}$ is defined as in (17).

Proof. First consider I, J such that $|I| + |J| \ge k$. For the sake of contradiction, assume that $\dim \mathcal{L}^M_{I,J} > 0$, so there is a non-zero vector $\mathbf{x} \in \mathcal{L}^M_{I,J}$. The branch number of M is no more than

$$\mathsf{wt}(\mathbf{x}) + \mathsf{wt}(M\mathbf{x}) \le (k - |I|) + (k - |J|) \le k,$$

which contradicts M having the maximal branch number.

Then consider I, J such that $|I| + |J| < k$. We can always find $I' \supseteq I$ and $J' \supseteq J$ such that $|I'| + |J'| = k$. Then by (18),

$$\dim \mathcal{L}^M_{I,J} \leq \dim \mathcal{L}^M_{I',J'} + |I' \setminus I| + |J' \setminus J| = k - |I| - |J|,$$
$$\dim \mathcal{L}^M_{I,J} \geq \dim \mathcal{L}^M_{\varnothing,\varnothing} - |I| - |J| = k - |I| - |J|. \qquad \square$$

B Implementation details for AES* layout graph

Our current analysis is not directly applicable to the AES case because of the special structure. We show that we can still use our SPN* results on the AES cipher, using a number of simple observations and implementation details, first proposed by [BV06]. In particular, we give a detailed description of how to compute the mixing time of the AES* layout graph efficiently, to obtain Theorem 6, Lemma 14, and Table 2.

The first obstacle is the fact that Lemma 10 assumes a maximal-branch mixing layer, which is not the case for AES, whose mixing step is comprised of two separate parts, ShiftRows and MixColumns. Formally, write the transition probability between two layouts I and J in terms of the transition probabilities of the ShiftRows (SR) and MixColumns (MC) operations.

$$\text{trans-prob}(I, J) = \sum_{I'} \mathcal{SR}(I, I') \cdot \mathcal{MC}(I', J)$$

The ShiftRows operation of the AES mixing is a permutation map on the layouts, which makes $\mathcal{SR}(I, I')$ a simple-to-calculate binary function. The AES MixColumns operation has maximal branch number if we think of it as a map from $\mathbb{F}^4_{2^8}$ to $\mathbb{F}^4_{2^8}$. Equivalently, the MixColumns operation acts as four maximal-branch-number mixing maps being applied to each column of blocks separately. If we write the columns of layouts I' and J as $\{I'_i\}_{i\in[4]}$ and $\{J_i\}_{i\in[4]}$ respectively, then $\mathcal{MC}(I', J)$ is the product of the transition probabilities between the 4 columns under a maximal-branch-number mixing layer.

$$\mathcal{MC}(I', J) = \prod_{i=1}^{4} \text{trans-prob}(I'_i, J_i)$$

Now we have a way to compute the transition probabilities for AES mixing and random S-boxes, which means we can compute the distance of AES* from pairwise independence using the powers of its transition matrix. The size of the transition matrix is $(2^{16} - 1) \times (2^{16} - 1)$, since it stores the probabilities between any pair of layouts. Raising this matrix to a power is not a trivial task even for today's computing capabilities. A final optimization comes from the observation that the transition matrix has rank at most 5^4.

Recall that $\text{trans-prob}(I_i, J_i)$ only depends on $\text{free}(I_i) + \text{free}(J_i)$. This means that knowing the Hamming weights $(\text{free}(I'_i))_{i\in[4]}$ and $(\text{free}(J_i))_{i\in[4]}$ is sufficient if one wishes to compute $\mathcal{MC}(I', J)$. Let us denote the tuple $(\text{free}(I'_i))_{i\in[4]}$ as the *compressed layout representation* of layout I'. Since a compressed layout is defined by 4 weights in the range $\{0, \ldots, 4\}$, there are at most 5^4 compressed layouts. For emphasis, we will refer to our original non-compressed definition of layouts as "full layouts". The AES mixing transition matrix can now be expressed in terms of these compressed layouts, by first projecting a full layout to its compressed representation, computing applying the MixColumns operation to the compressed layout, and then expanding it to the full layout representation. Formally, the transition matrix T can be written as

$$T = \mathcal{SR} \cdot \mathcal{MC} = \mathcal{SR} \cdot \mathcal{P} \cdot \mathcal{MC}_{compr} \cdot \mathcal{P}_{inv}.$$

The descriptions of the 3 new matrices are below:

1. **Matrix $\mathcal{P}$.** Projects a layout to its compressed layout. In other words, a compressed layout will have probability mass equal to the total probability mass of the layouts it contains. Matrix $\mathcal{P}$ has size $(2^{16} - 1) \times 5^4$.
2. **Matrix $\mathcal{MC}_{compr}$.** Applies MixColumns to the compressed layout. We have seen above that the compressed layout description is sufficient to determine the transition probability. $\mathcal{MC}_{compr}$ is a $5^4 \times 5^4$ stochastic matrix.
3. **Matrix $\mathcal{P}_{inv}$.** Expands the compressed layouts to the original layout space. In particular, the total probability mass of a compressed layout is evenly distributed to the layouts it contains. Matrix $\mathcal{P}_{inv}$ has size $5^4 \times (2^{16} - 1)$.

Now we can define the $(2^{16} - 1) \times 5^4$ matrix $\mathcal{FC} := \mathcal{SR} \cdot \mathcal{P} \cdot \mathcal{MC}_{compr}$ (which stands for Full-to-Compressed) and the $5^4 \times (2^{16} - 1)$ matrix $\mathcal{CF} := \mathcal{P}_{inv}$ (Compressed-to-Full). Our transition matrix T is then the product of these lower-rank matrices, which allows for more efficient computation of its powers.

$$\begin{aligned}
T^r &= (\mathcal{FC} \cdot \mathcal{CF})^r \\
&= \mathcal{FC} \cdot (\mathcal{CF} \cdot \mathcal{FC})^{r-1} \cdot \mathcal{CF} \\
&= \mathcal{FC} \cdot \mathcal{CC}^{r-1} \cdot \mathcal{CF}
\end{aligned}$$

where $\mathcal{CC} := \mathcal{FC} \cdot \mathcal{CF}$ is a $5^4 \times 5^4$ stochastic matrix (Compressed-to-Compressed).

Dealing with precision errors. Our goal is to compute the statistical distance from the stationary distribution up to a precision of at least 2^{-128}. To avoid floating-point arithmetic errors, we only operate on multiples of the matrices with integer entries. Thus our calculations are exact and the final result is obtained by normalizing at the very end. This allows us to make sure that no precision errors accumulate throughout the intermediate steps of our computation. With the above set of optimizations, we are able to experimentally prove Theorem 6, Lemma 14, and complete Table 2.

Lemma 14. *The 3-round AES* is $2^{-23.42}$-close to pairwise independent.*

C Implementation details for AES S-box composition

To obtain the tighter total variation (TV) distance bounds of Table 3, we model the AES S-box (XOR-key followed by patched inversion over $\mathbb{F}_{2^b}$) as a Markov chain over $\mathbb{F}_{2^8} \setminus \{0\}$.

The transition probabilities can be computed exactly by iterating over all 2^8 possible choices of the random key. Then we compute the powers of the $(2^8 - 1) \times (2^8 - 1)$ transition matrix and compute the largest TV distance from the uniform distribution over all $2^8 - 1 = 255$ possible starting states, which is at most the TV distance of distribution from uniform over $\mathbb{F}_{2^8} \setminus \{0\}$, due to the convexity of the metric.

INV Repetitions r	$\log_2$(TV distance to random S-box)
1	-0.99
2	-7.11
8	-29.39
11	-40.24

Table 3: Statistical distance upper bound of (AES S-box)$^{\otimes r}$ from pairwise random S-box over $\mathbb{F}_{2^8} \setminus \{0\}$.

Similar to our AES* implementation, we only store and operate on a scaled version of the transition matrix such that the entries are integers to avoid accumulating precision errors from floating-point operations. The final TV distance is normalized at the very end.

A summary of our experimental results is in Table 3.

D Approximating a random S-box via INV S-boxes

In this section, we formally prove our claim that a random S-box over $\mathbb{F}_{2^b}$ can be approximated via the sequential composition of alternating AddRoundKey and INV S-box operations.

Theorem 8. *The sequential composition of $O\left(b \cdot 2^{2b} \cdot \log(1/\epsilon)\right)$ alternating* AddRoundKey *and INV S-box operations generates a permutation that is ϵ-close to t-wise independent over $\mathbb{F}_{2^b}$ (random S-box) for $t < 2^b - 2$.*

We do this by representing the above procedure as a random walk and bounding its mixing time using the comparison method of Diaconis and Saloff-Coste [DSC93]. The Markov chain $\mathcal{M}$ we consider is the graph over the alternating group A_{2^b} of even permutations of length 2^b. Note that the number of states in this Markov chain is $\Theta((2^b)!) = \Theta(2^{b \cdot 2^b})$. The generating set of this graph will be the set of permutations we can obtain via the composition of an AddRoundKey, an INV S-box, and another AddRoundKey operation (with possibly a different key). Namely, the generating set is the following:

$$T = \{\pi : x \to \texttt{INV}(x + r_1) + r_2 \mid r_1, r_2 \in \mathbb{F}_{2^b}\}.$$

Obtaining mixing time bounds on $\mathcal{M}$ implies bounds on the number of INV S-boxes required to approximate a truly random S-box. This is because k steps of the random walk on $\mathcal{M}$ by following permutations $\sigma_1, \ldots, \sigma_k$ are the same as composing k INV S-boxes:

$$\pi = \sigma_k \circ \cdots \circ \sigma_1$$

$$\begin{aligned} \implies \pi : x \to \texttt{INV}\left(\ldots \texttt{INV}\left(\texttt{INV}(x + r_1^{(1)}) + r_2^{(1)} + r_1^{(2)}\right) + r_2^{(2)} + \ldots\right) + r_2^{(k)} \\ = \texttt{INV}\left(\ldots \texttt{INV}\left(\texttt{INV}(x + \rho_1) + \rho_2\right) + \rho_3 + \ldots\right) + \rho_k. \end{aligned}$$

Where we used $r_1^{(i)}, r_2^{(i)}$ to denote the random keys of permutation σ_i.

Representing permutations. For ease of notation, we will denote the permutation σ generated by adding key r_1, applying the `INV` S-box, and adding key r_2 using the pair r_1, r_2 in square brackets $[r_1, r_2]$. This is to avoid confusion with the transposition between elements r_1 and r_2. As a concrete example, the permutations σ_i defined in the previous paragraph would be denoted by $[r_1^{(i)}, r_2^{(i)}]$.

To denote the permutation one obtains after composing $k+1$ AddRoundKey operations with k `INV` S-boxes, we write the keys of each AddRoundKey operation in the square brackets. In the example above, we would use the following notation to represent $\pi = \sigma_k \circ \cdots \circ \sigma_1$:

$$\pi = [r_1^{(1)}, r_2^{(1)} + r_1^{(2)}, r_2^{(2)} + r_1^{(3)}, \ldots, r_2^{(k)}] = [\rho_1, \rho_2, \ldots, \rho_k].$$

Prior work. The comparison method is used to relate the mixing time of two Markov chains $\mathcal{M}$ and $\mathcal{M}'$, by constructing a flow between each edge of $\mathcal{M}'$, using paths of $\mathcal{M}$. In our case, this means constructing a transposition between 0 and $i \in \mathbb{F}_{2^b}$ using AddRoundKey $\circ$ `INV` $\circ$ AddRoundKey. Our result can be seen as a generalization of a result of Carlitz [Car53, Zie13], which shows that we can construct a transposition $(0, i)$ by composing degree-one polynomials and `INV`. Our result differs from this prior work in two ways: First, we only allow AddRoundKey operations, which is a subset of the degree-one polynomials (polynomials whose linear coefficient is equal to 1). Secondly, we give a bound on the number of AddRoundKey operations that one needs to compose to obtain a random transposition of the form $(0, i)$.

The main technical ingredient of this section is Lemma 15, which shows how to generate a transposition by alternating AddRoundKey and `INV` S-boxes.

Lemma 15. *For any $\alpha, \beta, \gamma \in \mathbb{F}_{2^b}$ such that $\alpha\beta \neq 1$, $\alpha(\beta+1) \neq 1$, and $\beta \notin \{0, 1\}$, we can generate the transposition $\left(\alpha + \frac{\beta}{\alpha\beta+1}, \alpha + \frac{\beta+1}{\alpha\beta+\alpha+1}\right)$ using the following sequence of* AddRoundKey *and `INV` S-box operations*

$$[\alpha, \alpha, \beta, \mathit{INV}(\gamma) + 1, \gamma, \mathit{INV}(\gamma), \gamma, \mathit{INV}(\gamma), \gamma + 1, \beta + 1, \alpha, \alpha]\,.$$

This construction can be easily adapted to give transpositions with a fixed element, i.e. $(0, i)$ for non-zero $i \in \mathbb{F}_{2^b}$.

Corollary 2. *For any $\alpha, \beta, \gamma \in \mathbb{F}_{2^b}$ such that $\alpha\beta \neq 1$, $\alpha(\beta+1) \neq 1$, and $\beta \notin \{0, 1\}$, we can generate the transposition $\left(0, \frac{\beta}{\alpha\beta+1} + \frac{\beta+1}{\alpha\beta+\alpha+1}\right)$ using the following sequence of* AddRoundKey *and `INV` S-box operations*

$$\left[\frac{\beta}{\alpha\beta+1}, \alpha, \beta, \mathit{INV}(\gamma) + 1, \gamma, \mathit{INV}(\gamma), \gamma, \mathit{INV}(\gamma), \gamma + 1, \beta + 1, \alpha, \frac{\beta}{\alpha\beta+1}\right].$$

Proof. For any two $x, y \in \mathbb{F}_{2^b}$, $x, y \neq 0$, we can obtain the transposition $(0, x+y)$ by composing (x, y) with two AddRoundKey permutations $\mathrm{ARK}_\delta : z \to z + \delta$ as follows:

$$(0, x+y) = \mathrm{ARK}_x \circ (x, y) \circ \mathrm{ARK}_x\,.$$

Indeed

$$\begin{aligned}
(\mathrm{ARK}_x \circ (x,y) \circ \mathrm{ARK}_x)(0) &= (\mathrm{ARK}_x \circ (x,y))(x) &&= \mathrm{ARK}_x(y) &&= x + y \\
(\mathrm{ARK}_x \circ (x,y) \circ \mathrm{ARK}_x)(x+y) &= (\mathrm{ARK}_x \circ (x,y))(y) &&= \mathrm{ARK}_x(x) &&= 0 \\
(\mathrm{ARK}_x \circ (x,y) \circ \mathrm{ARK}_x)(z) &= (\mathrm{ARK}_x \circ (x,y))(x+z) &&= \mathrm{ARK}_x(x+z) &&= z
\end{aligned}$$

where z is not equal to 0 or $x+y$, and thus $x+z$ is not equal to x or y.

Now note that composing two AddRoundKey operations gives another AddRoundKey operation with key equal to the sum of the two original round keys. Hence we add $\alpha + \frac{\beta}{\alpha\beta+1}$ to the first and last keys of the tuple of Lemma 15 and obtain the desired result. □

Before we proceed with the proof of Lemma 15, let us consider what transpositions we can create using the above construction by studying the distribution of $\frac{\beta}{\alpha\beta+1} + \frac{\beta+1}{\alpha\beta+\alpha+1}$ over $\mathbb{F}_{2^b}$.

$$\begin{aligned}\mu &= \frac{\beta}{\alpha\beta+1} + \frac{\beta+1}{\alpha\beta+\alpha+1}\\ &= \frac{\beta(\alpha\beta+\alpha+1)+(\beta+1)(\alpha\beta+1)}{(\alpha\beta+1)(\alpha\beta+\alpha+1)}\\ &= \frac{\alpha\beta^2+\alpha\beta+\beta+\alpha\beta^2+\beta+\alpha\beta+1}{(\alpha\beta+1)(\alpha\beta+\alpha+1)}\\ &= \frac{1}{(\alpha\beta+1)(\alpha\beta+\alpha+1)}.\end{aligned}$$

We can see that μ is non-zero, hence we consider the distribution of its inverse for random valid α, β. Claim 1 below shows that any non-zero value of μ (except 1) appears $2^b - 4$ times, whereas $\mu = 1$ will appear roughly twice as often. We conclude that if we choose α, β at random, we will get $(0, \mu)$ with roughly uniform probability (up to a constant factor) for all non-zero $\mu \in \mathbb{F}_{2^b}$.

Claim 1. *The number $T(\kappa)$ of solutions α, β to the equation*

$$(\alpha\beta+1)(\alpha\beta+\alpha+1) = \kappa$$

that satisfy $\beta \notin \{0,1\}$, and $\alpha\beta \neq 1$, $\alpha(\beta+1) \neq 1$ is

$$T(\kappa) = \begin{cases} 0 & \kappa = 0\\ 2\cdot 2^b - 4 & \kappa = 1\\ 2^b - 4 & \kappa \notin \{0,1\}\end{cases}.$$

Proof. We first consider $\kappa = 0$. For the RHS to equal 0, either $\alpha\beta = 1$, or $\alpha(\beta+1) = 1$, which is not a valid parameter setting by our assumption. Thus, $T(0) = 0$.

Set $\alpha = 0$. Then for all $2^b - 2$ valid values of $\beta \in \mathbb{F}_{2^b} \setminus \{0,1\}$, we have that $\kappa = 1$.

We now consider the case when $\alpha \neq 0$. If we expand the expression and divide by α^2, we get that

$$\beta(\beta+1) = \frac{\kappa+\alpha+1}{\alpha^2}.$$

It is well known that the quadratic equation above has 2 solutions if the trace of the RHS is equal to 0, and no solutions otherwise. The trace of the RHS equals

$$\begin{aligned}\mathrm{Tr}\left(\frac{\kappa+\alpha+1}{\alpha^2}\right) &= \mathrm{Tr}\left(\frac{\kappa}{\alpha^2}\right) + \mathrm{Tr}\left(\frac{1}{\alpha}\right) + \mathrm{Tr}\left(\frac{1}{\alpha^2}\right)\\ &= \mathrm{Tr}\left(\frac{\kappa}{\alpha^2}\right).\end{aligned}$$

Where the first equality follows from the linearity of trace and the second equality from the fact that $\mathrm{Tr}(x^2) = \mathrm{Tr}(x)$. The square is an injective map over $\mathbb{F}_{2^b}$, and thus $\frac{\kappa}{\alpha^2}$ obtains every value over $\mathbb{F}_{2^b} \setminus \{0\}$. Since $\mathrm{Tr}(\cdot) = 0$ defines a subspace, the number of α's the make the RHS have 0 trace is exactly $2^{b-1} - 1$ (we exclude zero, since $\frac{\kappa}{\alpha^2}$ is never zero.

All of these values are valid, except $\alpha = \kappa + 1$, for $\kappa \neq 1$. This is because even though

$$\mathrm{Tr}\left(\frac{\kappa}{(\kappa+1)^2}\right) = \mathrm{Tr}\left(\frac{1}{\kappa+1} + \frac{1}{(\kappa+1)^2}\right) = 0,$$

the RHS is equal to 0, and thus the two solutions to this equation are $\beta = 0, 1$, which are not valid. The remaining $2^{b-1} - 2$ values of α give 2 valid solutions for β, which means that all $\kappa \notin \{0, 1\}$ have $2^b - 4$ solutions for non-zero α.

The case of $\kappa = 1$ has the $2^b - 2$ solutions with $\alpha = 0$ and 2 solutions for the $2^{b-1} - 1$ non-zero valid values of α. Thus $T(1) = 2 \cdot 2^b - 4$. □

Our proof of Lemma 15 will follow from Claims 2 and 3. Note that Claim 2 is already enough to give us a bound on the number of operations required to simulate a random S-box. We use Claim 3 to get an improved comparison constant and get a better quantitative bound.

A note on notation. In the proofs of Claims 2 and 3, we will make frequent use of the following notation when computing the image of $x \in \mathbb{F}_{2^b}$ under the permutation $[k_1, \ldots, k_n]$:

$$\begin{aligned}
& x \\
\xrightarrow{k_1}\ & x + k_1 \\
\xrightarrow{k_2}\ & \mathtt{INV}(x + k_1) + k_2 \\
\xrightarrow{k_3}\ & \mathtt{INV}(\mathtt{INV}(x + k_1) + k_2) + k_3 \\
& \cdots \\
\xrightarrow{k_n}\ & \mathtt{INV}(\ldots \mathtt{INV}(\mathtt{INV}(x + k_1) + k_2) \ldots) + k_n.
\end{aligned}$$

Observe that the first arrow only applies AddRoundKey with the key being k_1 to the input, whereas the i^{th} arrow (for $i > 1$) applies the `INV` operation and then AddRoundKey with k_i as key.

Claim 2. *For any $\alpha, \beta \in \mathbb{F}_{2^b}$ such that $\alpha\beta \neq 1$, $\alpha(\beta + 1) \neq 1$, and $\beta \notin \{0, 1\}$, we can generate the transposition $\left(\alpha + \frac{\beta}{\alpha\beta+1}, \alpha + \frac{\beta+1}{\alpha\beta+\alpha+1}\right)$ using the following sequence of* AddRoundKey *and `INV` S-box operations*

$$[\alpha, \alpha, \beta, 1, 1, \beta + 1, \alpha, \alpha]\,.$$

Proof. Denote the permutation defined by the sequence of the statement by π. Our proof will proceed as follows:

1. Prove the statement for $\alpha = 0$.
2. Prove the general statement
   (a) Assume that no input to $\mathtt{INV}(\cdot)$ is equal to zero. Thus all intermediate values we obtain during our calculations will be denoted by rational expressions.
   (b) Consider the cases when we compute the inverse of a zero value separately.

Case 1. $\alpha = 0$**:** We will show that the sequence $[0, 0, \beta, 1, 1, \beta{+}1, 0, 0]$ generates the transposition $(\beta, \beta + 1)$.

We consider first the application of π on some x that is not equal to β or $\beta + 1$.

$$\begin{aligned} x \xrightarrow{0} x \xrightarrow{0} \texttt{INV}(x) \xrightarrow{\beta} x + \beta \xrightarrow{1} \frac{1}{x+\beta} + 1 = \frac{x+\beta+1}{x+\beta} \\ \xrightarrow{1} \frac{x+\beta}{x+\beta+1} + 1 = \frac{1}{x+\beta+1} \xrightarrow{\beta+1} x \xrightarrow{0} \texttt{INV}(x) \xrightarrow{0} x. \end{aligned}$$

Now consider what happens when $x = \beta$:

$$\beta \xrightarrow{0} \beta \xrightarrow{0} \texttt{INV}(\beta) \xrightarrow{\beta} 0 \xrightarrow{1} 1 \xrightarrow{1} 0 \xrightarrow{\beta+1} \beta + 1 \xrightarrow{0} \texttt{INV}(\beta+1) \xrightarrow{0} \beta + 1.$$

And when $x = \beta + 1$:

$$\beta + 1 \xrightarrow{0} \beta + 1 \xrightarrow{0} \texttt{INV}(\beta+1) \xrightarrow{\beta} 1 \xrightarrow{1} 0 \xrightarrow{1} 1 \xrightarrow{\beta+1} \beta \xrightarrow{0} \texttt{INV}(\beta) \xrightarrow{0} \beta.$$

Case 2: We will now prove the result for all valid parameters $\alpha \neq 0, \beta$ and inputs x that do not make an input to $\texttt{INV}(\cdot)$ to vanish. Thus, for these calculations, we will use the fact that $y \cdot \texttt{INV}(y) = 1$ for all non-zero y. Also for brevity, we will use $\texttt{INV}(y)$ and $\frac{1}{y}$ interchangeably.

$$\begin{aligned}
& x \\
\xrightarrow{\alpha}\ & x + \alpha \\
\xrightarrow{\alpha}\ & \frac{1}{x+\alpha} + \alpha = \frac{\alpha x + \alpha^2 + 1}{x + \alpha} \\
\xrightarrow{\beta}\ & \frac{x+\alpha}{\alpha x + \alpha^2 + 1} + \beta = \frac{(\alpha\beta+1)x + \alpha^2\beta + \beta + \alpha}{\alpha x + \alpha^2 + 1} \\
\xrightarrow{1}\ & \frac{\alpha x + \alpha^2 + 1}{(\alpha\beta+1)x + \alpha^2\beta + \beta + \alpha} + 1 = \frac{(\alpha\beta+\alpha+1)x + \alpha^2\beta + \beta + \alpha + \alpha^2 + 1}{(\alpha\beta+1)x + \alpha^2\beta + \beta + \alpha} \\
\xrightarrow{1}\ & \frac{(\alpha\beta+1)x + \alpha^2\beta + \beta + \alpha}{(\alpha\beta+\alpha+1)x + \alpha^2\beta + \beta + \alpha + \alpha^2 + 1} + 1 = \frac{\alpha x + \alpha^2 + 1}{(\alpha\beta+\alpha+1)x + \alpha^2\beta + \beta + \alpha + \alpha^2 + 1} \\
\xrightarrow{\beta+1}\ & \frac{(\alpha\beta+\alpha+1)x + \alpha^2\beta + \beta + \alpha + \alpha^2 + 1}{\alpha x + \alpha^2 + 1} + \beta + 1 = \frac{x + \alpha}{\alpha x + \alpha^2 + 1} \\
\xrightarrow{\alpha}\ & \frac{\alpha x + \alpha^2 + 1}{x + \alpha} + \alpha = \frac{1}{x + \alpha} \\
\xrightarrow{\alpha}\ & x + \alpha + \alpha = x
\end{aligned}$$

So we have seen that for α, β, x such that no input to $\texttt{INV}(\cdot)$ is zero, π acts like the identity and maps x to itself. To complete our proof, we now consider what happens if some input to $\texttt{INV}(\cdot)$ equals 0. This happens when one of the following equalities hold:

(a) $x + \alpha = 0 \implies x = \alpha$.

(b) $\alpha x + \alpha^2 + 1 = 0 \implies \alpha x = \alpha^2 + 1 \implies x = \alpha + \frac{1}{\alpha}$, since the equality doesn't hold if $\alpha = 0$.

(c) $(\alpha\beta+1)x+\alpha^2\beta+\beta+\alpha=0 \implies x=\alpha+\frac{\beta}{\alpha\beta+1}$, since we have imposed that $\alpha\beta\neq 1$.

(d) $(\alpha\beta+\alpha+1)x+\alpha^2\beta+\beta+\alpha+\alpha^2+1=0 \implies x=\alpha+\frac{\beta+1}{\alpha\beta+\alpha+1}$, since we have imposed that $\alpha(\beta+1)\neq 1$.

Note that the third and fourth cases are the claimed non-fixed points of π. Looking forward, we will verify that π transposes these two inputs.

Case 2(a). $x=\alpha\neq 0$**:** The permutation π maps α to itself as we show below:

$$\begin{aligned}
&\alpha\\
\xrightarrow{\alpha}\ &\alpha+\alpha=0\\
\xrightarrow{\alpha}\ &0+\alpha=\alpha\\
\xrightarrow{\beta}\ &\frac{1}{\alpha}+\beta=\frac{\alpha\beta+1}{\alpha}\\
\xrightarrow{1}\ &\frac{\alpha}{\alpha\beta+1}+1=\frac{\alpha\beta+\alpha+1}{\alpha\beta+1}\\
\xrightarrow{1}\ &\frac{\alpha\beta+1}{\alpha\beta+\alpha+1}+1=\frac{\alpha}{\alpha\beta+\alpha+1}\\
\xrightarrow{\beta+1}\ &\frac{\alpha\beta+\alpha+1}{\alpha}+\beta+1=\frac{1}{\alpha}\\
\xrightarrow{\alpha}\ &\alpha+\alpha=0\\
\xrightarrow{\alpha}\ &0+\alpha=\alpha
\end{aligned}$$

Note that in all the above computations, we have only evaluated INV$(\cdot)$ at the values $\alpha, \alpha\beta+1$, and $\alpha\beta+\alpha+1$, which are non-zero.

Case 2(b). $x=\alpha+\frac{1}{\alpha}$**:** The permutation π maps $\alpha+\frac{1}{\alpha}$ to itself as we show below:

$$\begin{aligned}
&\alpha+\frac{1}{\alpha}\\
\xrightarrow{\alpha}\ &\alpha+\frac{1}{\alpha}+\alpha=\frac{1}{\alpha}\\
\xrightarrow{\alpha}\ &\alpha+\alpha=0\\
\xrightarrow{\beta}\ &0+\beta=\beta\\
\xrightarrow{1}\ &\frac{1}{\beta}+1=\frac{\beta+1}{\beta}\\
\xrightarrow{1}\ &\frac{\beta}{\beta+1}+1=\frac{1}{\beta+1}\\
\xrightarrow{\beta+1}\ &0\\
\xrightarrow{\alpha}\ &\alpha\\
\xrightarrow{\alpha}\ &\frac{1}{\alpha}+\alpha
\end{aligned}$$

Note that in all the above computations, we have only evaluated $\texttt{INV}(\cdot)$ at the values α, β, and $\beta + 1$, which are non-zero.

Case 2(c). $x = \alpha + \frac{\beta}{\alpha\beta+1}$**:** The permutation π maps $\alpha + \frac{\beta}{\alpha\beta+1}$ to $\alpha + \frac{\beta+1}{\alpha\beta+\alpha+1}$ as we show below:

$$\begin{aligned}
&\alpha + \frac{\beta}{\alpha\beta+1} \\
&\xrightarrow{\alpha} \alpha + \frac{\beta}{\alpha\beta+1} + \alpha = \frac{\beta}{\alpha\beta+1} \\
&\xrightarrow{\alpha} \frac{\alpha\beta+1}{\beta} + \alpha = \frac{1}{\beta} \\
&\xrightarrow{\beta} \beta + \beta = 0 \\
&\xrightarrow{1} 0 + 1 = 1 \\
&\xrightarrow{1} 1 + 1 = 0 \\
&\xrightarrow{\beta+1} 0 + \beta + 1 = \beta + 1 \\
&\xrightarrow{\alpha} \frac{1}{\beta+1} + \alpha = \frac{\alpha\beta+\alpha+1}{\beta+1} \\
&\xrightarrow{\alpha} \frac{\beta+1}{\alpha\beta+\alpha+1} + \alpha
\end{aligned}$$

Note that in all the above computations, we have only evaluated $\texttt{INV}(\cdot)$ at the values $\alpha\beta + 1, \alpha\beta + \alpha + 1, \beta$, and $\beta + 1$, which are non-zero.

Case 2(d). $x = \alpha + \frac{\beta+1}{\alpha\beta+\alpha+1}$**:** The permutation π maps $\alpha + \frac{\beta+1}{\alpha\beta+\alpha+1}$ to $\alpha + \frac{\beta}{\alpha\beta+1}$ as we show below:

$$\begin{aligned}
&\alpha + \frac{\beta+1}{\alpha\beta+\alpha+1} \\
&\xrightarrow{\alpha} \alpha + \frac{\beta+1}{\alpha\beta+\alpha+1} + \alpha = \frac{\beta+1}{\alpha\beta+\alpha+1} \\
&\xrightarrow{\alpha} \frac{\alpha\beta+\alpha+1}{\beta+1} + \alpha = \frac{1}{\beta+1} \\
&\xrightarrow{\beta} \beta + 1 + \beta = 1 \\
&\xrightarrow{1} 1 + 1 = 0 \\
&\xrightarrow{1} 0 + 1 = 1 \\
&\xrightarrow{\beta+1} 1 + \beta + 1 = \beta \\
&\xrightarrow{\alpha} \frac{1}{\beta} + \alpha = \frac{\alpha\beta+1}{\beta} \\
&\xrightarrow{\alpha} \frac{\beta}{\alpha\beta+1} + \alpha
\end{aligned}$$

Note that in all the above computations, we have only evaluated $\texttt{INV}(\cdot)$ at the values $\alpha\beta+1, \alpha\beta+\alpha+1, \beta$, and $\beta+1$, which are non-zero. □

Claim 3. *The following two sequences of* AddRoundKey *and INV S-box operations implement the same permutations*

$$[1, 1] \equiv [\mathit{INV}(\gamma) + 1, \gamma, \mathit{INV}(\gamma), \gamma, \mathit{INV}(\gamma), \gamma + 1]$$

Proof. Denote by π_{LHS}, π_{RHS} as the permutations of the LHS and RHS respectively. Then π_{LHS} maps $x \to \texttt{INV}(x+1) + 1$. We will show that this is the case of π_{RHS}. For simplicity, we will first compute the image of x under π_{RHS}, assuming that no input to $\texttt{INV}(\cdot)$ is equal to zero. Thus, we will use the fact that $y \cdot \texttt{INV}(y) = 1$ for all non-zero y. Also for brevity, we will use $\texttt{INV}(y)$ and $\frac{1}{y}$ interchangeably.

$$\begin{aligned}
&x \\
&\xrightarrow{\texttt{INV}(\gamma)+1} x + \frac{1}{\gamma} + 1 = \frac{x\gamma+\gamma+1}{\gamma} \\
&\xrightarrow{\gamma} \frac{\gamma}{x\gamma+\gamma+1} + \gamma = \frac{x\gamma^2+\gamma^2}{x\gamma+\gamma+1} \\
&\xrightarrow{\texttt{INV}(\gamma)} \frac{x\gamma+\gamma+1}{x\gamma^2+\gamma^2} + \frac{1}{\gamma} = \frac{1}{x\gamma^2+\gamma^2} \\
&\xrightarrow{\gamma} x\gamma^2 + \gamma^2 + \gamma \\
&\xrightarrow{\texttt{INV}(\gamma)} \frac{1}{x\gamma^2+\gamma^2+\gamma} + \frac{1}{\gamma} = \frac{x\gamma+\gamma}{x\gamma^2+\gamma^2+\gamma} = \frac{x+1}{x\gamma+\gamma+1} \\
&\xrightarrow{\gamma+1} \frac{x\gamma+\gamma+1}{x+1} + \gamma + 1 = \frac{x\gamma+\gamma+1+(x\gamma+x)+(\gamma+1)}{x+1} = \frac{x}{x+1} = \texttt{INV}(x+1) + 1
\end{aligned}$$

To complete our proof, we now consider what happens if some input to $\texttt{INV}(\cdot)$ equals 0. Thus, we will consider the following cases separately:

1. $\gamma = 0$,
2. $x\gamma + \gamma + 1 = 0 \implies x = \frac{\gamma+1}{\gamma}$
3. $x\gamma^2 + \gamma^2 = 0 \implies x = 1$
4. $x + 1 = 0 \implies x = 1$.

Case 1. $\gamma = 0$**:** For $\gamma = 0$, π_{RHS} becomes the permutation denoted by $[1, 0, 0, 0, 0, 1]$. This permutation maps

$$x \xrightarrow{1} x + 1 \xrightarrow{0} \texttt{INV}(x+1) \xrightarrow{0} x + 1 \xrightarrow{0} \texttt{INV}(x+1) \xrightarrow{0} x + 1 \xrightarrow{1} \texttt{INV}(x+1) + 1.$$

Note that in the above expression, we only used the fact that $\texttt{INV}(\texttt{INV}(y)) = y$, which holds for all y. Hence the above mapping holds for all x.

Case 2. $x = \frac{\gamma+1}{\gamma}$**:** For simplicity we will assume that $\gamma \neq 0$, as this case was already covered above. This allows us to replace $\texttt{INV}(\gamma) + 1$ with $\frac{1}{\gamma} + 1 = \frac{\gamma+1}{\gamma}$. The permutation π_{RHS} maps $\frac{\gamma+1}{\gamma}$ to

$$\begin{aligned}
&\frac{\gamma+1}{\gamma}\\
\xrightarrow{\texttt{INV}(\gamma)+1}\ &\frac{\gamma+1}{\gamma} + \frac{\gamma+1}{\gamma} = 0\\
\xrightarrow{\gamma}\ &0 + \gamma = \gamma\\
\xrightarrow{\texttt{INV}(\gamma)}\ &\texttt{INV}(\gamma) + \texttt{INV}(\gamma) = 0\\
\xrightarrow{\gamma}\ &0 + \gamma = \gamma\\
\xrightarrow{\texttt{INV}(\gamma)}\ &\texttt{INV}(\gamma) + \texttt{INV}(\gamma) = 0\\
\xrightarrow{\gamma+1}\ &0 + \gamma + 1 = \gamma + 1.
\end{aligned}$$

Note that

$$\texttt{INV}(x+1) + 1 = \texttt{INV}\left(\frac{\gamma+1}{\gamma} + 1\right) + 1 = \texttt{INV}\left(\frac{1}{\gamma}\right) + 1 = \gamma + 1.$$

Thus π_{RHS} maps this value of x to the same image as the $[1, 1]$ permutation.

Case 3. $x = 1$**:** For simplicity we will assume that $\gamma \neq 0$, as this case was already covered above. This allows us to replace $\texttt{INV}(\gamma) + 1$ with $\frac{1}{\gamma} + 1 = \frac{\gamma+1}{\gamma}$. The permutation π_{RHS} maps $\frac{\gamma+1}{\gamma}$

to

$$\begin{aligned}
& 1 \\
\xrightarrow{\texttt{INV}(\gamma)+1}\ & 1+\texttt{INV}(\gamma)+1=\texttt{INV}(\gamma) \\
\xrightarrow{\gamma}\ & \gamma+\gamma=0 \\
\xrightarrow{\texttt{INV}(\gamma)}\ & 0+\texttt{INV}(\gamma)=\texttt{INV}(\gamma) \\
\xrightarrow{\gamma}\ & \gamma+\gamma=0 \\
\xrightarrow{\texttt{INV}(\gamma)}\ & 0+\texttt{INV}(\gamma)=\texttt{INV}(\gamma) \\
\xrightarrow{\gamma+1}\ & \gamma+\gamma+1=1.
\end{aligned}$$

Again, $\texttt{INV}(1+1)+1=1$, thus π_{RHS} maps $x=1$ to the same image as the $[1,1]$ permutation.

□

Comparison method. Now we will employ the comparison method to obtain mixing time bounds on our Markov chain $\mathcal{M}$ with respect to the mixing time of the random walk of the Markov chain $\mathcal{M}'$ on the Cayley graph generated by the transpositions of the form $(0,y)$ for non-zero y. We start with a short overview of the comparison method, partially taken verbatim from [DGJM06].

Suppose that $\mathcal{M}$ is an ergodic Markov chain on state space Ω with transition matrix P and stationary distribution π, and that $\mathcal{M}'$ is another ergodic Markov chain on the same state space with transition matrix P' and stationary distribution π'.

For every edge (x,y) of $\mathcal{M}'$, let $\mathcal{P}_{x,y}$ be the set of paths from x to y using transitions of $\mathcal{M}$. More formally, let $\mathcal{P}_{x,y}$ be the set of paths $\gamma=(x=x_0,x_1,\dots,x_k=y)$ such that each (x_i,x_{i+1}) is in $\mathcal{M}$. We write $|\gamma|$ to denote the length of path γ. So, for example, if $\gamma=(x_0,\dots,x_k)$ we have $|\gamma|=k$. Let $\mathcal{P}=\cup_{(x,y)\text{ in }\mathcal{M}'}\mathcal{P}_{x,y}$.

An $(\mathcal{M},\mathcal{M}')$-flow is a function f from $\mathcal{P}$ to the interval $[0,1]$ such that for every (x,y) in $\mathcal{M}'$,

$$\sum_{\gamma\in\mathcal{P}_{x,y}} f(\gamma)=\pi'(x)P'(x,y).$$

The flow is said to be an *odd* $(\mathcal{M},\mathcal{M}')$-flow if it is supported by odd-length paths. That is, for every $\gamma\in\mathcal{P}$, either $f(\gamma)=0$ or $|\gamma|$ is odd.

Let $r((z,w),\gamma)$ be the number of times that the edge (z,w) appears on path γ. For every (z,w) in $\mathcal{M}$, the *congestion* of edge (z,w) in the flow f is the quantity

$$A_{z,w}(f)=\frac{1}{\pi(z)P(z,w)}\sum_{\gamma\in\mathcal{P}:(z,w)\in\gamma} r((z,w),\gamma)\cdot|\gamma|\cdot f(\gamma).$$

The *congestion* of the flow is the quantity

$$A(f)=\max_{(z,w)\text{ in }\mathcal{M}} A_{z,w}(f).$$

Having a flow between two Markov chains allows one to compare their mixing times. In particular,

Theorem 9 (Theorem 8 of [DGJM06])**.** *Suppose that $\mathcal{M}$ is a reversible ergodic Markov chain with stationary distribution π and that $\mathcal{M}'$ is another reversible ergodic Markov chain with the same stationary distribution. Suppose that f is an odd $(\mathcal{M}, \mathcal{M}')$-flow. Then, for any $0 < \delta < \frac{1}{2}$,*

$$\tau_x(\mathcal{M}, \epsilon) \leq A(f) \left[\frac{\tau(\mathcal{M}', \delta)}{\ln(1/2\delta)} + 1 \right] \ln \frac{1}{\epsilon \pi(x)}.$$

Proof of Theorem 8. We will construct an odd $(\mathcal{M}, \mathcal{M}')$-flow with low congestion. In particular, we will use Corollary 2 to construct for every edge $(\sigma, (0, y) \circ \sigma)$ of $\mathcal{M}'$ a set of paths $\mathcal{P}_{\sigma,(0,y)\circ\sigma}$ using transitions of $\mathcal{M}$. We want the total flow through these paths to be equal to $\pi'(\sigma) \cdot P'(\sigma, (0, y) \circ \sigma)$. Since the stationary distribution of $\mathcal{M}'$ is the uniform distribution, and each edge is chosen with the same probability, the total flow through each edge has to be the same. Thus for simplicity, we will denote by C the value of $\pi'(x)P'(x, y)$ for all edges (x, y) of $\mathcal{M}'$.

Our paths will be constructed as in Corollary 2. In particular, for every $\alpha, \beta, \gamma \in \mathbb{F}_{2^b}$ such that $\alpha\beta \neq 1$, $\alpha(\beta + 1) \neq 1$, and $\beta \notin \{0, 1\}$, we will define $\mathcal{P}_{\sigma,\left(0,\frac{\beta}{\alpha\beta+1}+\frac{\beta+1}{\alpha\beta+\alpha+1}\right)\circ\sigma}$ to include all paths of the form:

$$\left[\frac{\beta}{\alpha\beta + 1}, \alpha, \beta, \texttt{INV}(\gamma) + 1, \gamma, \texttt{INV}(\gamma), \gamma, \texttt{INV}(\gamma), \gamma + 1, \beta + 1, \alpha, \frac{\beta}{\alpha\beta + 1} \right].$$

Recall that to construct these paths, we will use the edges of $\mathcal{M}$ as follows:

$$\left(\left[\frac{\beta}{\alpha\beta + 1}, r_1 \right], \left[r_1', r_2 \right], [r_2', r_3], \ldots, \left[r_{10}', \frac{\beta}{\alpha\beta + 1} \right] \right)$$

that satisfy $r_1 + r_1' = \alpha$, $r_2 + r_2' = \beta$, $r_3 + r_3' = \texttt{INV}(\gamma) + 1$, and so on.

Thus, our paths are parametrized by 3 variables α, β, γ, and 10 'auxiliary' variables $r_1, \ldots, r_{10}$. Thus $|\mathcal{P}| = \Theta\left(2^{13b}\right)$, and Claim 1 shows that each edge (x, y) of $\mathcal{M}'$ will be satisfied roughly the same number of times as the other edges of $\mathcal{M}'$ (up to a small constant factor). Since $\mathcal{M}'$ has $\Theta(2^b)$ edges $|\mathcal{P}_{x,y}| = \Theta\left(2^{12b}\right)$. Additionally, all paths have lengths of exactly 11 edges.

If we push the same amount of flow through each such path, Claim 1 implies that all edges will get the same total amount of flow, except the edges of the form $(\sigma, (0, 1) \circ \sigma)$, which will receive roughly twice as much flow. Thus we will reduce the flow through these paths accordingly to get the same total flow as the rest of the edges. Note that $\frac{2 \cdot 2^b - 4}{2^b - 4} \leq 3$ for $b \geq 3$, and thus there exists some flow value f^* such that $f^* \leq f(\gamma) \leq 3f^*$ for all $\gamma \in \mathcal{P}$. The total flow equality then gives

$$\begin{aligned}
\sum_{\gamma \in \mathcal{P}_{x,y}} f(\gamma) &= \pi'(x)P'(x, y) \\
\implies |\mathcal{P}_{x,y}| \cdot \Theta\left(f^*\right) &= C \\
\implies \Theta\left(2^{12b} f^*\right) &= C \\
\implies f^* &= \Theta\left(C \cdot 2^{-12b}\right).
\end{aligned}$$

We will now bound the number of paths $\gamma \in \mathcal{P}$ that use some edge (z, w) of $\mathcal{M}$. We will show that due to the way we constructed our paths, (z, w) is only used by $\Theta(2^{11b})$ paths. First, we show in Claim 4 that $\mathcal{M}$ has no parallel edges, thus the edge (z, w) specifies a specific transition $[s, t]$ in $\mathcal{M}$.

As an example, let's consider the number of paths γ that have (z, w) as their second edge. From the structure of our paths, we know that $r_1' = s$ and $r_2 = t$. This specifies two equations that the parameters of a candidate path γ must satisfy to include edge (z, w). Since each path has 13 parameters, there are 11 remaining degrees of freedom, and each edge can be in a constant number (at most 11) of locations in a path, the total number of paths passing through any edge is at most $\Theta(2^{11b})$.

Now lets compute the congestion of edge (z, w) of $\mathcal{M}$:

$$\begin{aligned}
A_{z,w}(f) &= \frac{1}{\pi(z)P(z,w)} \sum_{\gamma\in\mathcal{P}:(z,w)\in\gamma} r((z,w),\gamma)\cdot|\gamma|\cdot f(\gamma) \\
&\le \frac{11}{\pi(z)P(z,w)} \sum_{\gamma\in\mathcal{P}:(z,w)\in\gamma} r((z,w),\gamma)\cdot f(\gamma) \\
&\le \frac{11^2}{\pi(z)P(z,w)} \sum_{\gamma\in\mathcal{P}:(z,w)\in\gamma} f(\gamma) \\
&\le \frac{3\cdot 11^2}{\pi(z)P(z,w)} \cdot f^* \cdot (\text{\# of paths } (z,w) \text{ appears in}) \\
&\le \frac{3\cdot 11^2}{\pi(z)P(z,w)} \cdot \Theta\left(\frac{C}{2^{12b}}\cdot 2^{11b}\right) \\
&= \frac{3\cdot 11^2}{\pi(z)P(z,w)} \cdot \Theta\left(\frac{\pi'(x)P'(x,y)}{2^b}\right) \\
&= \Theta\left(\frac{P'(x,y)}{P(z,w)\cdot 2^b}\right) \\
&= \Theta\left(1\right).
\end{aligned}$$

Where the last equality follows because the degree of $\mathcal{M}$ is $\Theta(2^{2b})$, whereas the degree of $\mathcal{M}'$ is $\Theta(2^b)$.

It follows from a result of Friedman [Fri00], that the spectral gap of $\mathcal{M}'$ is $\Theta\left(\frac{1}{2^b}\right)$, and thus its mixing time is bounded by

$$\tau(\mathcal{M}', \delta) = O\left(b\cdot 2^{2b} + 2^b\cdot\log(1/\delta)\right).$$

We conclude that Theorem 9 for a small enough δ implies

$$\tau(\mathcal{M}, \epsilon) = O\left(b\cdot 2^{2b}\cdot\log(1/\epsilon)\right).$$

□

Claim 4. *The Markov chain $\mathcal{M}$ does not have any parallel edges. Formally, if there exists $\sigma \in A_{2^b}$ such that*

$$\mathrm{ARK}_i \circ \mathtt{INV} \circ \mathrm{ARK}_j \circ \sigma = \mathrm{ARK}_k \circ \mathtt{INV} \circ \mathrm{ARK}_\ell \circ \sigma,$$

then $(i, j) = (k, \ell)$.

Proof. First, observe that if $i = k$, then the statement is true. Indeed, we can apply the permutation $\texttt{INV} \circ \mathrm{ARK}_i$ to both sides and obtain

$$\mathrm{ARK}_j \circ \sigma = \mathrm{ARK}_\ell \circ \sigma \implies j = \ell.$$

Similarly, if $j = \ell$, then the statement also holds. Hence we proceed by considering the case when both $i \neq k$, and $j \neq \ell$.

Now choose an input x such that $\sigma(x) \notin \{j, \ell\}$. Thus we can write the value of x on the two sides as a fraction:

$$\frac{iy + ij + 1}{y + j} = \frac{ky + k\ell + 1}{y + \ell}$$
$$\implies iy^2 + (ij + 1 + i\ell)y + ij\ell + \ell = ky^2 + (k\ell + 1 + kj)y + jk\ell + j.$$

For the above equality to be true for more than 2 values of y, it must hold that $i = k$. This concludes the proof. □