Equivalence of Generalised Feistel Networks

Patrick Derbez[1] and Marie Euler[1,2]

[1] IRISA, Rennes, France, firstname.name@irisa.fr
[2] DGA MI, Bruz, France

Abstract. This paper focuses on equivalences between Generalised Feistel Networks (GFN) of type-II. We introduce a new definition of equivalence which captures the concept that two GFNs are identical up to re-labelling of the inputs/outputs, and give a procedure to test this equivalence relation. Such two GFNs are therefore cryptographically equivalent for several classes of attacks. It induces a reduction of the space of possible GFNs: the set of the $(k!)^2$ possible even-odd GFNs with $2k$ branches can be partitioned into $k!$ different classes.
This result can be very useful when looking for an optimal GFN regarding specific computationally intensive properties, such as the minimal number of active S-boxes in a differential trail. We also show that in several previous papers, many GFN candidates are redundant as they belong to only a few classes. Because of this reduction of candidates, we are also able to suggest better permutations than the one of `WARP`: they reach 64 active S-boxes in one round less and still have the same diffusion round that `WARP`. Finally, we also point out a new family of permutations with good diffusion properties.

Keywords: GFN · WARP · TWINE · LBlock

1 Introduction

A Feistel network is a widely spread structure for symmetric cryptography primitives. Invented by Feistel and Coppersmith in 1973 for IBM's `Lucifer` cipher, it was later standardised in the block cipher `DES` in 1976 [S+77]. In a Feistel network, the internal state is divided into two parts of the same size: the left branch x and the right branch y. The round function of the i-th round of the Feistel network is the involutive operation $F_i := (x, y) \mapsto (x, y \oplus f_i(x))$, where f_i is a keyed function, followed by the swap of x and y as depicted in Figure 1a. The advantage of such a construction is that one does not need the inverse of the function f_i to inverse the function F_i, and thus non-invertible (or complex to inverse) functions can be used in the cipher, which can be useful in some applications (e.g. constrained environments, FHE). Furthermore, Luby and Rackoff proved in [LR88] that if the f_i's are independent pseudorandom functions then a 3-round Feistel network is indistinguishable from a random permutation in the context of Chosen Plaintexts Attacks (CPA) while a 4-round Feistel network provides resistance against Chosen Ciphertexts Attacks (CCA). Hence, it is not surprising that many designers followed this strategy to build both efficient and secure ciphers.

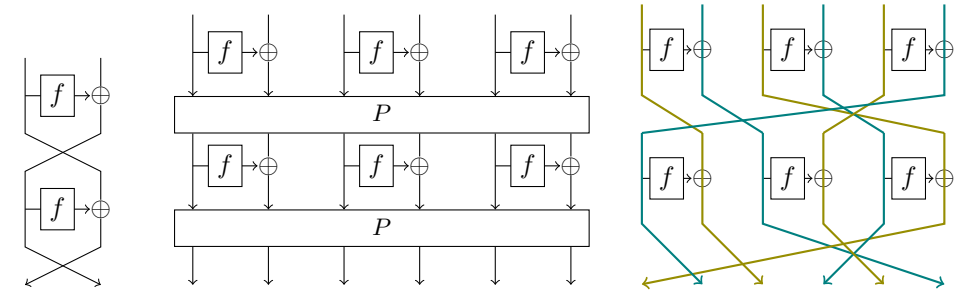
Later, Zheng et al. [ZMI90] generalised the original construction so that the state is now divided into $2k$ same-size parts $(x_0, x_1, \ldots, x_{2k-1})$. These parts are also called branches. Several generalisations – named type-I, type-II and type-III – were suggested, but in this paper we will focus on type-II, which seems to be the design favoured by the community. The round function of the i-th round of a type-II generalised Feistel network is

$$F_i^k := (x_0, x_1, \ldots, x_{2k-1}) \mapsto (x_0, x_1 \oplus f_i(x_0), \ldots, x_{2k-2}, x_{2k-1} \oplus f_i(x_{2k-2})),$$

followed by a circular shift of the $2k$ parts of the state, sending each branch to the next one:

$$(x_0, x_1, \ldots, x_{2k-1}) \mapsto (x_{2k-1}, x_0, x_1, \ldots, x_{2k-2}).$$

At Asiacrypt '96, Nyberg proposed to replace the circular shift by another specific permutation [Nyb96]. Then, in [SM10], Suzaki and Minematsu proposed the Generalised Feistel Network (GFN) by replacing the circular shift by a general permutation P (see Figure 1b), aiming to identify the permutation that would provide the best diffusion. Among others, two type-II GFN with 16 branches (`LBlock` [WZ11], `TWINE` [SMMK13]) and one with 32 branches (`WARP` [BBI+20]) were later proposed.



(a) A Feistel Network.

(b) A type-II Generalised Feistel Network with 6 branches.

(c) A GFN with an even-odd permutation.

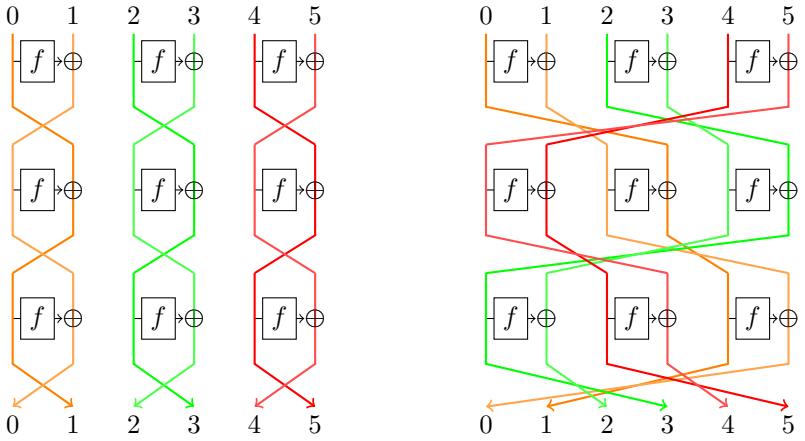
Figure 1: 2 rounds of some types of (Generalised) Feistel Networks.

The main problem one would face to find the best permutations for a certain property (diffusion, resistance against differential or linear attacks, etc.) comes from the huge search space: there are $(2k)!$ different permutations for a $2k$-branch GFN. In [CGT19], Cauchois *et al.* consider the so-called *natural* equivalence classes based on conjugacy: two Feistel networks relying on $2k$-permutations P and Q respectively are equivalent if and only if there exists a permutation of pairs A such that $P = AQA^{-1}$, where a permutation of pairs is a permutation A such that $A(2i+1) = A(2i) + 1$ for all i between 0 and $k-1$. In other words, up to re-labelling of the inputs/outputs, the two Feistel schemes are identical. This property allowed them to describe a more efficient generation of even-odd Feistel permutations (*i.e.* permutations that map branches with an even index to branches with an odd index and reciprocally), based on the cycle structures of the left-branches permutation. Moreover, the authors of [CGT19] also recall that, since F is involutive, the decryption through the Feistel scheme associated with P are, up to re-labelling of the inputs and outputs, identical to the encryption through the Feistel scheme associated with P^{-1}. Therefore, they coined the expression "extended equivalence" to describe the fact that P and Q or P and Q^{-1} are equivalent as we most often want the decryption function to be as resistant as the encryption function.

Note that the cryptographic properties covered by these notions of equivalence are the ones associated with propagation or trails: we may think for example of the diffusion round, truncated differential/linear trails (including both the minimal number of active S-boxes and impossible differentials) and integral characteristics. However, some cryptographic properties are not preserved as for instance the ones involving the key schedule such as related-key attacks require further analysis.

Our contribution In this paper, we present a wider definition of equivalence of the underlying permutations of GFNs: we say that two permutations P and Q are expanded-equivalent if and only if there exists a permutation of pairs A such that for all positive integer i, $Q^i A P^{-i}$ is a permutation of pairs. Compared to the conjugacy-based equivalence,

this definition takes into account several rounds of the GFN and therefore captures multiple-round equivalence even if it is not visible on one round. Our motivation comes from the observation that equivalence notions introduced in previous works do not cover all the cases. For example, both the Feistel networks depicted in Figure 2 share the exact same properties while the inner permutations are not isomorphic.



(a) With A being the identity map. (b) With A mapping i to $(i+1) \mod 3$.

Figure 2: Trivial example of isomorphic GFNs whose permutations are not conjugates. Both figure depict three rounds of a 6-branches GFN associated to $\Pi_{A,A}$: the i-th left-branch is mapped to the $A(i)$-th right-branch and similarly for right branches.

Our new equivalence relation comes with two different characterisations. The first one highlights the fact that this equivalence of GFNs can be seen as a cyclic behaviour on a finite number of rounds. It also brings a new way to test whether two GFNs are equivalent. The second one, only valid for even-odd permutations, captures the structure of the equivalence classes and leads to the computation of the size of the equivalence classes. It also brings out new class invariants. In particular, we show that the $k!^2$ GFNs associated with even-odd permutations on $2k$ branches can be grouped in exactly $k!$ equivalence classes, each of them containing $k!$ GFNs.

Moreover, we also describe some improvements on the previously known algorithm regarding the enumeration of conjugacy-based equivalence classes ([CGT19]): we show how to optimise the enumeration when dealing with extended equivalence and how to enumerate only one element per conjugacy-based equivalence class or per expanded equivalence class. This led us to a new family of permutations for which the associated GFNs have good diffusion.

Finally, we also exhaust interesting GFN permutations from the literature for various properties and regroup them into a few classes. In particular, regarding the case of 32-branch GFNs such as `WARP`, we study the cryptographic properties of more candidates than the designers did and thus find a permutation with the same diffusion round that in `WARP` but better differential and linear properties. Similarly, we study the diffusion round and differential properties of all even-odd permutations of 16 branches with finite diffusion and as a consequence prove the optimality of the permutations used in both `TWINE` and `LBlock`.

Our source code is available at `https://gitlab.inria.fr/meuler1/equivalence-of-generalized-feistel-network`.

Organisation of the paper Section 2 is dedicated to notations, definitions, and properties useful for the following parts. Then in Section 3, we introduce the new definition of

equivalence of GFNs, its characterisations, and equivalence testing. Section 4 focuses on the enumeration of the classes and provides a new family of GFNs offering good diffusion. Finally, Section 5, Section 6 and Section 7 apply this new notion of equivalence to some articles suggesting good permutations for GFNs.

2 Notations, definitions and previous works

Let us first introduce a few notations, definitions, and properties about permutations and how to use them in Generalised Feistel Networks.

2.1 Permutations

We will denote $\mathcal{S}_k$ the symmetric group acting on a set with k elements. Any permutation will be described by its value table: for instance, writing $P = [0, 1, 3, 4, 2]$ indicates that P is the permutation $P(0) = 0, P(1) = 1, P(2) = 3, P(3) = 4, P(4) = 2$. We denote by Id the identity permutation. The order of a permutation $P \in \mathcal{S}_k$, noted $order(P)$, is the smallest natural number i such that P^i is the identity permutation. Moreover, $\mathcal{S}_k$ may be partitioned in equivalence classes according to the conjugacy relation: two permutations P and Q in $\mathcal{S}_k$ are in the same equivalence class if and only if P and Q are conjugates *i.e.* there exists a permutation $A \in \mathcal{S}_k$ such that $Q = A \circ P \circ A^{-1}$. These equivalence classes are called conjugacy classes.

A disjoint cycle decomposition is associated with any permutation: for the previous example, a valid disjoint cycle decomposition is $P = (0)(1)(2, 3, 4)$. This decomposition is not unique, but the associated cycle type T_P is unique. $T_P[i]$ indicates the number of cycles of length i in any disjoint cycle decomposition of P. Here, we would have $T_P = \{1 : 2, 3 : 1\}$.

In Section 4, we will introduce efficient techniques to enumerate equivalence classes. They all use the property that the cycle type entirely determines the conjugacy classes of $\mathcal{S}_k$. In order to quantify the complexity of these algorithms, we will need to know the size of the conjugacy classes, *i.e.* the number $\gamma_{k,T}$ of permutations of $\mathcal{S}_k$ which have a cycle type T. For a cycle type T with n_ℓ cycles of length ℓ, we have

$$\gamma_{k,T} = \frac{k!}{\prod_\ell n_\ell! \ell^{n_\ell}}.$$

In particular, we have $\sum_T \gamma_{k,T} = k!$.

In Section 3, we will show that expanded equivalence is intrinsically linked with the set of permutations that commute with some derived permutations. That is why we also need to define the centraliser of a permutation $P \in \mathcal{S}_k$, which is the set of permutations that commute with P:

$$\texttt{Centr}(P) := \{Q \in \mathcal{S}_k, QP = PQ\}.$$

Its size depends only on the cycle type of P: $|\texttt{Centr}(P)| = \dfrac{k!}{\gamma_{k,T_P}} = \prod n_\ell! \ell^{n_\ell}$ if $T_p = \{\ell : n_\ell\}$.
More generally, we will use the centraliser of a set of permutations $E \subset \mathcal{S}_k$:

$$\texttt{Centr}(E) := \bigcap_{P \in E} \texttt{Centr}(P).$$

2.2 Permutations used in Feistel networks

A $2k$-branch GFN is defined by its permutation of branches $P \in \mathcal{S}_{2k}$. We write $\mathcal{F}_P$ to describe the GFN whose i-th round function is $P \circ F_i^k$ (for the Feistel step $F_i^k :=$

$(x_0, x_1, \ldots, x_{2k-1}) \mapsto (x_0, x_1 \oplus f_i(x_0), \ldots, x_{2k-2}, x_{2k-1} \oplus f_i(x_{2k-2}))$ and we will use the shorter product notation PF for this round function. This abstraction is due to the fact that we are only studying the formal structure of the GFN, a study which can be done in a preliminary phase, before instantiating this structure with specific functions f_i. Branches with an odd (resp. even) index are called left (resp. right) branches. Let us denote $\Upsilon \in \mathcal{S}_{2k}$ the permutation swapping left and right branches: for all $0 \leq i < k$, $\Upsilon(2i) = 2i + 1$ and $\Upsilon(2i + 1) = 2i$.

In the introduction, we have defined the even-odd permutations as the permutations of $\mathcal{S}_{2k}$ which map even numbers to odd numbers and reciprocally. We can describe such a permutation P by two smaller permutations L, R in $\mathcal{S}_k$ as follows:

$$\text{For all } i \text{ such that } 0 \leq i < k,\ L(i) := (P(2i) - 1)/2 \text{ and } R(i) := P(2i + 1)/2.$$

We call L the left-branches permutation and R the right-branches permutation. Conversely, for any L, R in $\mathcal{S}_k$, we denote as $\Pi_{L,R}$ the even-odd permutation constructed from L and R in the following way:

$$\Pi_{L,R}(2i) := 2L(i) + 1 \text{ and } \Pi_{L,R}(2i + 1) := 2R(i).$$

The GFN literature uses abundantly this type of permutation as they are usually simpler to implement and study. Moreover, for up to 32 branches, no non-even-odd permutation performs better than the optimal even-odd ones from a diffusion perspective [DDGP22].

Let us define as well the even permutations as the permutations which map even numbers to even numbers and odd numbers to odd numbers. Similarly, any even permutation P of $\mathcal{S}_{2k}$ can be described by two smaller permutations $L, R \in \mathcal{S}_k$: $L(i) := P(2i)/2$ and $R(i) := (P(2i + 1) - 1)/2$. In that case, we denote P as $\Phi_{L,R}$.

Finally, we consider the group of permutations of pairs:

$$\mathcal{S}_k^p = \{\Phi_{A,A}, A \in \mathcal{S}_k\} \subset \mathcal{S}_{2k}.$$

Any permutation of pairs commutes with the Feistel step F, a property which will be very useful for the GFN isomorphism.

All along the paper, we will use the following properties.

Property 1. Let $A, B, C, D \in \mathcal{S}_k$. Then, we have:

- $\Pi_{A,B}\Pi_{C,D} = \Phi_{BC,AD}$.
- $\Pi_{A,B}\Phi_{C,D} = \Pi_{AC,BD}$ and $\Phi_{C,D}\Pi_{A,B} = \Pi_{DA,CB}$.
- $\Upsilon\Phi_{A,B} = \Phi_{B,A}\Upsilon$.
- $\Phi_{A,B} \in \mathcal{S}_k^p \Leftrightarrow A = B$.

Given L and R in $\mathcal{S}_k$, we call the permutation $\Pi_{R,L}$ the dual of the even-odd permutation $\Pi_{L,R}$. These permutations are not always equivalent, but an interesting property links them: due to the symmetric roles of branching and XOR in differential and linear characteristics of GFNs, any differential (resp. linear) characteristic of the GFN associated to $\Pi_{L,R}$ can be written as a linear (resp. differential) characteristic of the GFN associated to the dual of $\Pi_{L,R}$. Hence, the minimum number of active S-boxes in a differential characteristic of the GFN associated to $\Pi_{L,R}$ is the minimum number of active S-boxes in a linear characteristic of the GFN associated with $\Pi_{R,L}$. Such duality was already described in [Mat95] for DES and later on in [SM10] for CLEFIA and HIGHT.

Another important property of a GFN is its diffusion: how fast do the input values impact all the output values? One way to measure this is by computing the diffusion round which is the minimal number of rounds needed for all output branches to depend on all the

input branches. This can be computed by considering the power matrices of the truncated linear layer [BMT14]. More precisely, we consider the maximum of the diffusion round for encryption and decryption. This property highly depends on the permutation used in the GFNs. The historical candidates (such as the 4-branch GFN CLEFIA [SSA+07] and the 8-branch GFN HIGHT [HSH+06]), which used a circular shift as permutation, had a diffusion round of the same order as the number of branches while the generalisation of GFN to any permutation [SM10] led to much better diffusion properties. Moreover, there exists a lower bound of the diffusion round for even-odd GFNs based on the Fibonacci sequence (ϕ_i): if $\phi_i \geq k > \phi_{i-1}$, then the diffusion round of any even-odd GFN with $2k$ branches is at least $i + 1$.

2.3 A first approach to GFN equivalence

Let us begin with a natural definition of equivalence of generalised Feistel networks: two GFNs are equivalent if, for any number of rounds, one is equal to the other up to a re-labelling of the inputs and outputs. More formally, this can be defined as follows:

Definition 1. Let P and Q be two $2k$-permutations associated with two generalised Feistel networks $\mathcal{F}_P$ and $\mathcal{F}_Q$. $\mathcal{F}_P$ and $\mathcal{F}_Q$ are equivalent if and only if for all positive integer i, there exist two permutations A_i and B_i such that $(QF)^i = B_i(PF)^i A_i^{-1}$.

This definition is interesting for cryptographers because it implies that both Feistel networks share some cryptographic properties: not only linear and differential characteristics but also diffusion, impossible differentials, etc. More details will be given in Subsection 3.2.

However, it is more convenient to have a property that directly links the underlying permutations. Hence [CGT19] suggested the following *natural* equivalence relations:

Definition 2. Let P and Q be two $2k$-permutations.

- P and Q are pair-equivalent if and only if there exists $A \in \mathcal{S}_k^p$ such that $Q = APA^{-1}$.
- P and Q are extended pair-equivalent if and only if P and Q are pair-equivalent or P and Q^{-1} are pair-equivalent.

Indeed, in the first case, A commutes with F thus for all i, $(QF)^i = A(PF)^i A^{-1}$ and $\mathcal{F}_P$ and $\mathcal{F}_Q$ are equivalent. The second equivalence comes from the fact that $\mathcal{F}_{P^{-1}}$ corresponds to the decryption of $\mathcal{F}_P$[1] and thus both permutations are typically evaluated together. In the following, we will denote these equivalences as the (extended-)conjugacy-based equivalence.

3 Expanded Feistel Equivalence

In this section, we present the core of our work: a larger equivalence relation between the permutations used in GFNs. We also highlight some useful properties regarding this equivalence.

3.1 New definition

We propose the following widened equivalence relation of permutations, which, as we will show, also implies the equivalence of associated GFNs.

[1] More precisely, $F^{-1} = F$ and thus for any number of rounds r, $(PF)^{-r} = (FP^{-1})^r = P(P^{-1}F)^r P^{-1}$ that is r rounds of decryption of $\mathcal{F}_P$ correspond up to relabelling the input and outputs to r rounds of encryption of $\mathcal{F}_P$.

Definition 3. Two $2k$-permutations P and Q are called expanded-equivalent if and only if there exists $A \in \mathcal{S}_k^p$ such that for all i, $Q^i A P^{-i} \in \mathcal{S}_k^p$.

The name has been chosen to highlight that the associated equivalence classes are larger than what was known before, without using once more the words *extended* or *generalised*. Since $\mathcal{S}_k^p$ is the set of permutations of pairs, this relation is trivially reflexive, symmetric, and transitive, and thus defines an equivalence relation. Moreover, for all $i \geq 0$, let us denote $Q^i A P^{-i}$ by A_i. Then we obtain that $A_{i+1}P = Q^{i+1}AP^{-i} = QA_i$ and thus

$$A_{i+1}P = A_1 P A^{-1} A_i.$$

This relation permits us to prove by induction that $(QF)^i = A_i(PF)^i A^{-1}$. First, the equality trivially holds for $i = 0$. Now, let us assume it holds for $i \geq 0$. Then $(QF)^{i+1} = QF(QF)^i = A_1(PF)A^{-1}A_i(PF)^i A^{-1}$ by induction hypothesis. Furthermore, F commutes with both A and A_i since they both are permutations of pairs. Therefore $(QF)^{i+1} = A_1 P A^{-1} A_i F(PF)^i A^{-1} = A_{i+1}(PF)^{i+1}A^{-1}$ and as a consequence, both $\mathcal{F}_P$ and $\mathcal{F}_Q$ are equivalent.

Note that P and Q are expanded-equivalent if and only if P^{-1} and Q^{-1} are expanded-equivalent. Indeed, for any pair of permutations P and Q, there exists an integer $n > 0$ such that $P^n = Q^n = \mathrm{Id}$, and then for any integer i, $(Q^{-1})^i A (P^{-1})^{-i} = Q^{-i}AP^i = Q^{ni}Q^{-i}AP^{-ni}P^i = Q^{ni-i}AP^{-(ni-i)}$ which is a permutation of pairs if P and Q are expanded-equivalent (since $ni - i \geq 0$). Furthermore, as for conjugacy-based equivalence, expanded equivalence can be extended to deal with the inverse permutations.

Definition 4. Let P and Q be two $2k$-permutations. P and Q are extended-expanded-equivalent if and only if P and Q are expanded-equivalent or P and Q^{-1} are expanded-equivalent.

We have the following trivial inclusions: for any permutation, its conjugacy-based equivalence class is a subset of its expanded equivalence class and its extended-conjugacy-based equivalence class is a subset of its extended-expanded equivalence class.

First example of a class Let us denote $\mathtt{Cl}(\Pi_{L,R})$ the class of expanded equivalence of $\Pi_{L,R}$. The easiest class to compute is $\mathtt{Cl}(\Pi_{\mathrm{Id},\mathrm{Id}}) = \{\Pi_{P,P}, P \in \mathcal{S}_k\}$. Indeed, if $Q \in \mathtt{Cl}(\Pi_{\mathrm{Id},\mathrm{Id}})$, then there exist $A = \Phi_{a,a}, B = \Phi_{b,b} \in \mathcal{S}_k^p$ such that $QA\Pi_{\mathrm{Id},\mathrm{Id}}^{-1} = B$ *i.e.* $Q = \Pi_{ba^{-1},ba^{-1}} \in \{\Pi_{P,P}, P \in \mathcal{S}_k\}$. Conversely, if $Q = \Pi_{P,P}$ then $Q^i \Pi_{\mathrm{Id},\mathrm{Id}}^{-i} = \Upsilon^i \Phi_{P^i,P^i} \Upsilon^i \Phi_{\mathrm{Id},\mathrm{Id}} = \Phi_{P^i,P^i} \in \mathcal{S}_k^p$. As $\Pi_{\mathrm{Id},\mathrm{Id}}^{-1} = \Pi_{\mathrm{Id},\mathrm{Id}}$, the extended-expanded equivalence class of $\Pi_{\mathrm{Id},\mathrm{Id}}$ and the expanded equivalence class are the same. Note that $Cl(\Pi_{\mathrm{Id},\mathrm{Id}})$ is significantly larger than the conjugacy-based equivalence class of $\Pi_{\mathrm{Id},\mathrm{Id}}$: the former has $k!$ elements while the latter is reduced to one element.

3.2 Invariant cryptographic properties

We formerly stated that some cryptographic properties are invariant in the equivalence classes. We formalise here in which cases this is applicable.

We first start with a few remarks which hold for both conjugacy-based equivalence and expanded equivalence. The main property needed to convert the equivalence of permutations to an equivalence of GFNs is that any permutation of pairs commutes with the function F. This implies that in each round, all the underlying Feistel functions (f_i at round i) are identical or can be considered as such. Many cryptanalysis techniques do not rely on the exact specification of either the S-boxes or the key schedule. This is the case for truncated cryptanalysis such as, for example, the minimal number of active S-boxes in a differential or a linear trail, the diffusion round and word-oriented Meet-in-the-Middle

(MITM) distinguishers. However, if the Feistel functions are not all identical (e.g. LBlock), instantiated differential/linear trails can no longer be transposed from one GFN to another equivalent one. Similarly, related-key attacks are not invariant as the role of the round keys changes from one branch to another. More generally, key-recovery attacks are not invariant, as the behaviour of the key in the key-recovery rounds changes from one GFN to another. Hence, we believe that, when designing a new primitive based on a GFN, the underlying permutation should be selected before defining the key schedule and the S-boxes in order to provide the best structural resistance and to make the other choices less critical regarding security.

Let us now introduce the main difference between conjugacy-based equivalence and expanded equivalence: invariant subspaces. An invariant subspace is a set S invariant by the round operation of the cipher. For a GFN $\mathcal{F}_P$, it means $PF(S) = S$. This property is preserved by conjugacy-based equivalence. Indeed, let $A \in \mathcal{S}_k^p$ and $Q = APA^{-1}$ equivalent to P. Then $A(S)$ is invariant by the round operation of $\mathcal{F}_Q$ (omitting that round constants may take different values for different branches in the same round). However, for expanded equivalence, some invariant spaces are no longer preserved. Indeed, let us consider the example of Figure 2 depicting two expanded-equivalent GFNs. In Figure 2a, one can see that the subspace where only the first pair of branches is active is mapped to itself for one round of the first GFN and thus is an invariant of the GFN. In Figure 2b, let us observe that this space is mapped by one round of the second GFN to the subspace where only the second pair of branches is active. It is therefore not an invariant subspace for this GFN. Of course, it will be vulnerable to a subspace trail attack but it is easier to search for invariant subspaces than subspace trails. We thus see two interesting properties offered by the expanded equivalence. First, instead of searching for subspace trails on a given GFN, one could search for invariant subspaces for all equivalent GFNs. It would be interesting to understand to which extent this approach would overcome the first one and how exhaustive it is. This also links with an idea from [LMR15]: invariant subspace trails can be found by looking for linear applications which commute with the linear layer. Second, we wonder whether it would be possible to hide a subspace trail (of low dimension) by first designing a GFN with a small invariant subspace and then releasing an equivalent GFN. It is indeed well-known that finding subspace trails involving small subspaces is quite hard and if the number of branches is high enough it might be impossible to exhaust the equivalent GFN to retrieve the invariant subspace.

3.3 Characterisation on a finite number of rounds

In practice, the former definition seems difficult to apply, as it relies on a property for all positive integers i. Fortunately, as we only study permutations of a finite set, we can reduce the definition to a property verifiable on a finite number of i. This comes from the following smaller equivalence relation.[2]

Definition 5 (r-cyclic equivalence)**.** Let P and Q be two $2k$-permutations. P and Q are called r-cyclic equivalent if and only if for all $i \leq r$, there exists a permutation of pairs A_i such that $Q^i = A_i P^i A_0^{-1}$ and $A_r = A_0$.

This definition naturally leads to the following characterisation of expanded equivalence.

Property 2 (First characterisation)**.** Let P and Q be two $2k$-permutations. P and Q are expanded-equivalent if and only if there exists a positive integer r such that they are r-cyclic equivalent.

[2]This definition is somehow close to the *permutational equivalence* from [BS13] which was applied to a wider definition of 4-branch GFNs. Yet, in their article, the equivalence is defined at the GFN level and not at the permutation level.

Proof. The direct implication is trivial with $r = \text{lcm}(order(Q), order(P))$. The converse comes from the fact that we can apply the Euclidean division on any exponent $i = rk + j$:

$$Q^i = Q^j Q^{rk} = Q^j (Q^r)^k = A_j P^j A_0^{-1} A_0 P^r A_0^{-1} = A_j P^{rk+j} A_0^{-1}. \qquad \square$$

Let us detail a few simple cases of r-cyclic equivalence (summarised in Table 1). In all cases, we consider two permutations P and Q in $\mathcal{S}_k$ and a permutation A in $\mathcal{S}_k^p$. Let us first notice that if $Q = A^{-1}PA$, then Q and P are 1-cyclic-equivalent, *i.e.* conjugacy-based equivalence boils down to 1-cyclic equivalence. Moreover, if $Q = AP = PA$, then $Q^r = A^r P^r$ *i.e.* Q and P are $order(A)$-cyclic-equivalent. This is for instance the case of the example in the introduction (Figure 2), where the rotation of pairs of branches commutes with the identity. Furthermore, if $Q = AP = PA^{-1}$, then $Q^2 = APAP = PA^{-1}AP = P^2$ *i.e.* Q and P are 2-cyclic-equivalent. More generally, if $Q = A^\alpha P = PA$, then $Q^r = (AP)^r = A^{1+\alpha+\cdots+\alpha^{r-1}} P^r$. Let r be such that $order(A)$ divides $1 + \alpha + \cdots + \alpha^{r-1}$. Then Q and P are r-cyclic-equivalent.

Table 1: Few examples of r-cyclic equivalence (for $A \in \mathcal{S}_k^p$).

r-cyclic equivalence	Example of P and Q which produces such equivalence
1-cyclic equivalence	$Q = A^{-1}PA$
2-cyclic equivalence	$Q = AP = PA^{-1}$
$Order(A)$-cyclic equivalence	$Q = AP = PA$
r-cyclic equivalence	$Q = AP = PA^\alpha$ (such that $A^{1+\alpha+\cdots+\alpha^{r-1}} = \text{Id}$)

We now suggest a procedure to test the r-cyclic equivalence between two permutations. We present here only the version for even-odd permutations. Nonetheless, it can also be adapted to deal with more generic permutations by imposing a different colouring on edges from E_P (corresponding to the shuffling of branches) and from E_F (corresponding to the Feistel horizontal wiring) and considering an isomorphism preserving the colouring. A version based on linear algebra over integers should also work but seems more laborious to work with.

Definition 6 (Cyclic Feistel graph of length r)**.** Let P be an even-odd permutation of $2k$ elements. We call cyclic Feistel graph of length r associated to P the directed graph $G_F^c(P, r)$ such that its set of vertices is

$$V = \{0, \ldots, 2k-1\} \times \{0, \ldots, r-1\}$$

and its edges are $E = E_P \bigcup E_F$ with $E_P = \{((i,j) \to (P(i), (j+1) \mod r)), (i,j) \in V\}$ and $E_F = \{((2i,j) \to (P(2i+1), (j+1) \mod r)), (2i,j) \in V\}$.

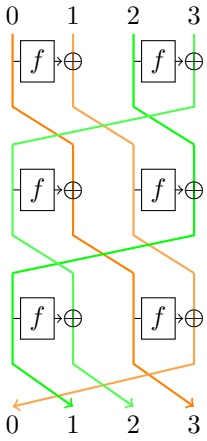
An example of a cyclic Feistel Graph is drawn in Figure 3. Let us consider an r-round Feistel network. Each vertex of the graph corresponds to a branch at the beginning of a round. There exists an edge between (i, r_0) and $(j, r_0 + 1)$ if and only if the i-th branch of round r_0 propagates to the j-th branch of the next round. After r rounds, for an r-cyclic Feistel, the edges loop back to the first round, modelling the conjugacy behaviour of the permutation.

These graphs are interesting because we can reduce the equivalence of permutations to an instance of graph isomorphism.

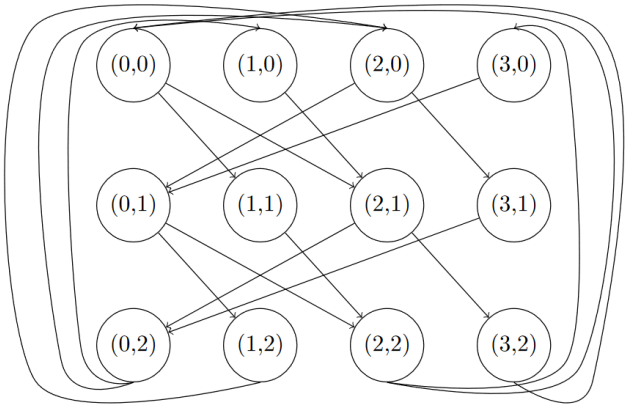
Property 3. Two even-odd permutations P and Q are r-cyclic-equivalent if and only if $G_F^c(P, r)$ and $G_F^c(Q, r)$ are isomorphic.

Proof. This result is rather intuitive as soon as the graphs are drawn. Indeed, one can group all the vertices of the same round by only looking at edges[3]. These sets are preserved

[3]This is only true when the graph is connected *i.e.* the diffusion of the permutation is finite. Otherwise, one can look at the connected components of the graph separately and repeat the argument.



(a) Three rounds of the GFN associated with P



(b) The graph $G_F^c(P, 3)$

Figure 3: An exemple of a GFN and its associated graph for 3 rounds ($P = [1, 2, 3, 0]$)

by any isomorphism (up to a cyclic renumbering of the rounds, but it has no importance because of the cyclic construction of the graph). So we can decompose the re-labelling of vertices of the isomorphism in r sub-permutations Π_i, each dealing with vertices of one round. It remains to show that these permutations are permutations of pairs of vertices and this is done in Subsection A.1. □

3.4 Fundamental characterisation for even-odd permutations

The previous characterisation helps to understand what it means for two permutations to be equivalent. However, there can be three permutations P, Q, R in a same expanded equivalence class, such that P and Q are r-cyclic-equivalent and Q and R are r'-cyclic-equivalent with $r \neq r'$. That is why we need a more fundamental characterisation, to capture the structure of the whole equivalence class.

One element of such a structure could be for example an invariant of the class. With conjugacy-based equivalence classes, we know that all the even-odd permutations from the same class had conjugated left-branches permutations and conjugated right-branch permutations. However, the example of $\texttt{Cl}(\Pi_{\mathrm{Id},\mathrm{Id}})$ in Section 3.1 shows that the cyclic structure of the left and right permutations are no more class invariants for expanded equivalence. However, all the permutations $\Pi_{P,P}$ of this class share the same quotient $R^{-1}L = P^{-1}P = \mathrm{Id}$ value. More generally, we observe that the cycle structure of the quotient $R^{-1}L$ of the left-branches permutation L by the right-branches permutation R is now a class invariant. Indeed, let $P = \Pi_{L,R}$ and $Q = \Pi_{L',R'}$ be two expanded-equivalent even-odd permutations. Then there exist $A = \Phi_{a,a}, B = \Phi_{b,b} \in \mathcal{S}_k^p$, such that $QAP^{-1} = B$ and thus $Q = BPA^{-1} = \Pi_{bLa^{-1},bRa^{-1}}$. Then $R'^{-1}L'^{-1} = (bRa^{-1})^{-1}bLa^{-1} = aR^{-1}La^{-1}$: it is a conjugate of $R^{-1}L$. Moreover, an even-odd permutation $\Pi_{L,R}$ can be uniquely defined by the two permutations R and $\alpha := R^{-1}L$. We denote this representation $\Psi_R^\alpha := \Pi_{R\alpha,R} = \Pi_{L,R}$ and it permits us to give the following characterisation of expanded equivalence.

Property 4 (Second characterisation)**.** Two even-odd permutations $P = \Psi_R^\alpha$ and Q are expanded-equivalent if and only there exist two permutations of pairs $A = \Phi_{a,a}$ and $B = \Phi_{b,b}$ such that $Q = ABPA^{-1}$ and $b \in \texttt{Centr}(\{R^{-i}\alpha R^i\}_{i\geq 0})$.

The proof, which relies on showing that for all i, P^iBP^{-i} is a permutation of pairs and

thus α commutes with $R^i b R^{-i}$, is given in Subsection A.2. This characterisation induces that expanded equivalence classes are the union of several conjugacy-based equivalence classes for some multiple $\Phi_{b,b}P$ of P with b being in $\mathtt{Centr}(\{R^{-i}\alpha R^i\}_{i\geq 0})$, *i.e.* b commutes with $R^{-i}\alpha R^i$ for all $i \geq 0$. Actually, $\mathtt{Centr}(\{R^{-i}\alpha R^i\}_{i\geq 0})$ is the centraliser of the permutations associated with i repetitions of L in one direction and i repetitions of R in the other direction.

Property 5. Let $L, R \in \mathcal{S}_k$ and $\alpha := R^{-1}L$. Then,

$$\mathtt{Centr}(\{R^{-i}\alpha R^i\}_{i\geq 0}) = \mathtt{Centr}(\{R^{-i}L^i\}_{i\geq 0}).$$

Proof. Let $C_i = R^{-i}\alpha R^i$, $D_i = R^{-i}L^i$. Then $C_0 = D_1$ and $D_{i+1} = R^{-i-1}L^{i+1} = R^{-i}\alpha L^i = R^{-i}\alpha R^i R^{-i}L^i = C_i D_i$ □

This set is also associated with the following property which brings about new class invariants: for all i, the cycle structure of $R^{-i}L^i$ is invariant in a class.

Property 6. Let $L, R, L', R' \in \mathcal{S}_k$, $\alpha := R^{-1}L$ and $b \in \mathtt{Centr}(\{R^{-i}\alpha R^i\}_{i\geq 0})$. Let $i \geq 0$. We have $(bR)^i\alpha(bR)^{-i} = R^i\alpha R^{-i}$ and $(bR)^{-i}(bL)^i = R^{-i}L^i$. Moreover, if $\Pi_{L,R}$ is expanded-equivalent to $\Pi_{L',R'}$, then $R^{-i}L^i$ and $R'^{-i}L'^i$ are conjugates.

Proof. Let us prove the first part by induction. It is trivial for $i = 0$, so let us suppose, we have i such that $(bR)^i\alpha(bR)^{-i} = R^i\alpha R^{-i}$ and $(bR)^{-i}(bL)^i = R^{-i}L^i$. Then, by the induction hypothesis,

$$(bR)^{i+1}\alpha(bR)^{-(i+1)} = bR(bR)^i\alpha(bR)^{-i}(bR)^{-1} = bRR^i\alpha R^{-i}R^{-1}b^{-1}$$

Moreover, b commutes with $R^{i+1}\alpha R^{-i-1}$ thus $(bR)^{i+1}\alpha(bR)^{-(i+1)} = R^{i+1}\alpha R^{-(i+1)}$. Similarly, using that $(bR)^{-i-1}(bL)^{i+1} = R^{-1}b^{-1}R^{-i}L^i bL$ and b commutes with $R^{-i}L^i$, we show that $(bR)^{-i-1}(bL)^{i+1} = R^{-i-1}L^{i+1}$.

If $\Pi_{L,R}$ and $\Pi_{L',R'}$ are expanded-equivalent, then there exist $A = \Phi_{a,a} \in \mathcal{S}_k^p$ and $B = \Phi_{b,b}$ with $b \in \mathtt{Centr}(\{R^{-i}\alpha R^i\}_{i\geq 0})$ such that $\Pi_{L',R'} = AB\Pi_{L,R}A^{-1} = \Pi_{abLa^{-1},abRa^{-1}}$. Then $R'^{-i}L'^i = a(bR)^{-i}(bL)^i a^{-1} = aR^{-i}L^i a^{-1}$. □

Let us now discuss another easy example of equivalence classes: let us consider the case of $\mathtt{Centr}(\{R^{-i}\alpha R^i\}_{i\geq 0}) = \{\mathrm{Id}\}$. In that case, by Property 4, $P := \Psi_R^\alpha$ is only equivalent to its conjugates via a permutation of pairs. Moreover, two conjugates via a permutation of pairs $\Phi_{a,a}$ are equal if and only if a commutes with α and with R. But $\mathtt{Centr}(\{R, \alpha\})$ is a subset of $\mathtt{Centr}(\{R^{-i}\alpha R^i\}_{i\geq 0})$ so there is no non-trivial element in this set. Thus, all the $k!$ conjugates of P are different. Hence, the expanded equivalence class of P has exactly $k!$ elements. Besides, it is true for all the expanded equivalence classes:

Theorem 1. *There exist $k!$ classes of expanded equivalence of even-odd GFNs with $2k$ branches. Each of these classes contains exactly $k!$ GFNs.*

The proof of this theorem is provided in Subsection A.3. It works in two steps: first, showing that the expanded equivalence class of Ψ_R^α can be partitioned in $k!/|\mathtt{Centr}(\alpha)|$ subsets and secondly, proving that all of these subsets are of size $|\mathtt{Centr}(\alpha)|$.

As a consequence, all the expanded equivalence classes contain $k!$ even-odd permutations with $2k$ branches. Since there are $(k!)^2$ even-odd permutations with $2k$ branches, we obtain that there are $k!$ expanded equivalence classes of even-odd permutations with $2k$-branches.

4 Improvements on the exhaustive search of [CGT19]

In [CGT19], the authors suggested an algorithm to enumerate at least one element per conjugacy-based equivalence class. We improve this algorithm for extended conjugacy-based equivalence classes. We also suggest an algorithm to enumerate exactly one element per equivalence class, both for conjugacy-based equivalence classes and expanded equivalence classes.

4.1 Previous work

Let us create $\mathcal{A}_k$ a subset of S_k with a representative per conjugacy class, *i.e.* $\{T_P, P \in \mathcal{A}_k\} = \{T_P, P \in \mathcal{S}_k\}$. We denote by $\mathcal{N}_k$ the size of $\mathcal{A}_k$ (which is exactly the number of decomposition into cycles of a permutation acting on k elements). The exhaustive search for conjugacy-based equivalence classes of even-odd permutations of [CGT19] is based on two properties:

1. For any permutation L, there exists $\tau \in \mathcal{A}_k$ such that L and τ are conjugates.

2. For any permutation of pairs A, $\Pi_{ALA^{-1},R}$ and $\Pi_{L,A^{-1}RA}$ are equivalent.

Therefore, any permutation $\Pi_{L,R}$ is equivalent to another permutation $\Pi_{\tau,R'}$ with $\tau \in \mathcal{A}_k$ and $R' \in \mathcal{S}_k$. Finally, considering the set $\mathcal{U}_k = \{\Pi_{\tau,R}, \tau \in \mathcal{A}_k, R \in \mathcal{S}_k\}$ is enough to cover all equivalence classes. This gives an enumeration of $\mathcal{N}_k k!$ elements.

4.2 Generalisation to extended conjugacy-based equivalence

The former argumentation can be adapted to extended conjugacy-based equivalence classes. Let L and R be two permutations of k elements. Then $\Pi_{L,R}$ and $\Pi_{L,R}^{-1} = \Pi_{R^{-1},L^{-1}}$ are in the same extended conjugacy-based equivalence class. Let us notice that if L and R^{-1} do not have the same cycle decomposition, this class will appear at least twice in the former enumeration. In order to prevent this, let us label the $\mathcal{N}_k$ cycle types: $\mathcal{T} := [t_0, \ldots, t_{N_k-1}]$. This enables us to filter out classes whose inverses have already been treated by imposing that the cycle type of the left-branches permutation L has a greater or equal index in $\mathcal{T}$ than the one of the right-branches permutation R, which we denote as "$L \geq R$".

More formally, it means that considering $\mathcal{E}_k = \{\Pi_{\tau,R}, \tau \in \mathcal{A}_k, R \in \mathcal{S}_k \text{ with } \tau \geq R\}$ is enough to enumerate all the extended conjugacy-based equivalence classes.

Moreover,

$$|\mathcal{E}_k| = \sum_{i=0}^{\mathcal{N}_k-1} \sum_{j=0}^{i} \gamma_{k,t_j} = \sum_{j=0}^{\mathcal{N}_k-1} (j+1)\gamma_{k,t_j} < \mathcal{N}_k \sum_{j=0}^{\mathcal{N}_k-1} \gamma_{k,t_j} = \mathcal{N}_k k!.$$

It is worth noticing that the labels of the cycle types impact greatly the size of $\mathcal{E}_k$. The optimal choice is therefore to label the cycle types such that $i \leq j$ implies $\gamma_{k,t_i} \leq \gamma_{k,t_j}$. This choice leads to a significant improvement of the former enumeration: for GFNs with 16 branches, this enumeration leads to 3.6 times fewer candidates, as reported in Table 2. Moreover, this ratio improves with the size of the GFNs. For example, $|\mathcal{U}_{25}|/|\mathcal{E}_{25}| \simeq 15$ and $|\mathcal{U}_{50}|/|\mathcal{E}_{50}| \simeq 83$.

Furthermore, let us call $\phi(k)$ (resp. $\psi(k)$) the number of conjugacy-based equivalence classes (resp. extended conjugacy-based equivalence classes). These numbers were formerly given on an experimental basis by testing the equivalence relation among the outputs of the enumeration of [CGT19] for small values of k. As a side note, we give here an explicit formula for $\phi(k)$: One can notice that beyond GFN, $\phi(k)$ describes the number

of equivalence classes of $\mathcal{S}_k \times \mathcal{S}_k$ acted on by $\mathcal{S}_k$. Consequently, we can apply Burnside's lemma (as in [Ove]) which leads to

$$\phi(k) = \sum_{t \in A_k} |S_k|/\gamma_{k,T_t} = \sum_{t \in A_k} |\texttt{Centr}(t)| = \sum_{t \in A_k, T_t = \{\ell : n_\ell\}} \prod n_\ell! \ell^{n_\ell}.$$

Table 2: Enumeration sizes compared to the number of classes.

	k	4	5	6	7	8	9
[SM10]	$(k!)^2$	576	14400	518400	25401600	1625702400	13×10^{10}
[CGT19]	$\lvert\mathcal{U}_k\rvert = \mathcal{N}_k k!$	120	840	7920	75600	887040	10886400
	$\phi(k)$	43	161	901	5579	43206	378360
Section 4	$\lvert\mathcal{E}_k\rvert$	55	360	2720	24030	244370	2721517
	$\Phi(k)$	28	96	495	2919	22024	190585
Gain	$\lvert\mathcal{U}_k\rvert/\lvert\mathcal{E}_k\rvert$	2.2	2.3	2.9	3.1	3.6	4.0

4.3 An algorithm exhausting conjugacy-based equivalence classes

The enumeration of [CGT19] gives multiple representatives for the same conjugacy-based equivalence class. Indeed, $\Pi_{t,R}$ and $\Pi_{t,R'}$ are equivalent by conjugacy if and only if there is a permutation A such that $t = AtA^{-1}$ and $R = AR'A^{-1}$. Therefore, the set of representatives of the conjugacy-based equivalence class of $\Pi_{t,R}$ returned by the algorithm of [CGT19] is exactly the set $\{\Pi_{t,ARA^{-1}}, A \in \texttt{Centr}(t)\}$.

Algorithm 1 Enumeration of conjugacy-based equivalence classes.

```
Initialise a set of representative of classes classes = {}.
for t ∈ A_k do
    Initialise a set S = S_k.
    while S is not empty do
        Pick R in S.
        Add Π_{t,R} to classes.
        Remove {ARA^{-1}, A ∈ Centr(t)} from S.
    end while
end for
Return classes.
```

To exhaust conjugacy classes without repetitions, one can use Algorithm 1. This algorithm has a time complexity $O(\mathcal{N}_k k!)$ steps and memory complexity of $O(k!)$ (to store the values of S). Experimentally, our Python implementation shows that this algorithm is 1.5 times slower than the enumeration from [CGT19]. The exact timings of our implementations are given in Appendix B. However, when dealing with large GFNs, the complexity of enumerating classes is negligible compared to the complexity of evaluating the resistance to differential attacks (which may take hours for one permutation) and thus avoiding redundancy is crucial for this step. We provide examples in Section 5 of the differences of time between the steps.

This algorithm can also be combined with the ideas from Subsection 4.2 in order to give one representative per extended-conjugacy-based class.

4.4 Enumeration of expanded equivalence classes

Similarly, we can use the characterisation from Property 4 to define an algorithm giving exactly one representative per expanded equivalence class. This algorithm is described in Algorithm 2.

Algorithm 2 Enumeration of expanded equivalence classes for $2k$-branch even-odd GFNs.

Initialise a set of representative of classes $classes = \{\}$.
for $\alpha \in \mathcal{A}_k$ **do**
  Initialise a set $S = S_k$.
  while S is not empty **do**
    Pick R in S.
    Add Ψ_α^R to $classes$.
    for B in $\texttt{Centr}(\{R^{-i}\alpha R^i\}_{i\geq 0})$ **do**
      Remove $\{ABRA^{-1}, A \in \texttt{Centr}(\alpha)\}$ from S.
    end for
  end while
end for
Return $classes$.

Applying some fine-tuning allowed us to reduce the complexity of the critical steps. Indeed, the time needed to remove elements from s can be reduced by keeping S sorted or at least by grouping conjugated permutations inside S. Moreover, $\texttt{Centr}(\{R^{-i}\alpha R^i\}_{i\geq 0})$ may be a large set and thus it should be constructed if and only if the corresponding elements have not yet been removed from S. A solution is to test for each B whether BR still belongs to S, and thus whether its conjugacy class has already been removed. Finally, the case $\alpha = \mathrm{Id}$ takes a particularly long time, as the $\mathcal{N}_k$ conjugacy-based equivalence classes are grouped in a single expanded equivalence class. Since we already know that it corresponds to only one class, we can treat this case apart and only add $\Pi_{\mathrm{Id},\mathrm{Id}}$ to the set $classes$.

The complexity of the algorithm depends on the complexity of computing the centraliser of a set of permutations, which itself depends on the complexity of computing a base and a strong generating set[4] of a permutation group. These algorithms are well studied in the computational group theory [Sim70] but go far beyond this article.

We were able to run this algorithm in less than half an hour on a laptop for k up to 9. For k up to 8, we give the lists of representatives of the $k!$ classes of equivalence of even-odd $2k$-branch GFNs with their associated security properties in Appendix C. We recall that we only used a Python implementation. Without doubt, an implementation using optimized libraries (e.g. GAP) would allow to go further.

4.5 A new family of GFNs with good diffusion properties

When generating all the optimal permutations (regarding diffusion of the associated GFNs) for a small number of branches, we observe that some of them has a strong property: both α and the group generated by the $R^i\alpha R^{-i}$ are of the same order that the number of pairs of branches k of the GFNs. For instance, this is the case for

$$\Psi_{[0,1]}^{[1,0]}, \Psi_{[0,2,1]}^{[1,2,0]}, \Psi_{[1,0,3,2]}^{[1,2,3,0]}, \Psi_{[4,3,2,1,0]}^{[1,2,3,4,0]}, \Psi_{[0,5,4,3,2,1]}^{[1,2,3,4,5,0]} \text{ and } \Psi_{[1,3,5,0,2,4,6]}^{[1,2,3,4,5,6,0]}.$$

Let us generalise these examples to a wider family of permutations. We first focus on prime k. Indeed, in that case, the group generated by any permutation α of order k is of

[4] A set permitting to efficiently go through a group of permutations [HEO05].

order k. Therefore, the group generated by the $R^i \alpha R^{-i}$ is of order k if and only all the $R^i \alpha R^{-i}$ are powers of α, *i.e.* cyclic permutations or the identity. If α is merely a rotation (that is $\alpha : x \mapsto x + 1 \mod k$), it implies that there exist j and b smaller than k, such that for all $x, R(x) = jx + b \mod k$. Conversely, if R is of this shape, then for all positive integer i, for all x, $R^i \alpha R^{-i}(x) \equiv x + j^i \equiv \alpha^{j^i}(x) \mod k$. Furthermore if $j > 1$, we can consider $\ell = b(j-1)^{-1} \mod k$ and notice that $\alpha^\ell R \alpha^{-\ell}(x) = jx + b + \ell - j\ell \equiv jx \mod k$ and thus $\Psi_\alpha^{x \mapsto jx}$ and Ψ_α^R are conjugacy-based equivalent. If $j = 1$, then α commutes with $R = \alpha^b$ and thus $\Psi_\alpha^{x \mapsto jx} = \Phi_{\alpha^{-b}, \alpha^{-b}} \Psi_\alpha^R$ and Ψ_α^R are expanded equivalent. In both cases, we conclude that there is no need to consider b.

If k is not prime, positive integers j are not always invertible modulo k, and thus defining $R : x \mapsto jx$ does not always lead to a permutation. Therefore, to ensure that R is a permutation, we generalise the previous structure in the following way:

Definition 7 (Pseudo-cyclic permutations)**.** Let k be a positive integer and j be a positive integer smaller than k. Let $i = k/\gcd(j,k)$. Let α be the cyclic permutation of order k defined as $\alpha : x \mapsto x + 1 \mod k$.

We call *pseudo-cyclic permutation* the permutation $\Psi_{R_j}^\alpha$ with R_j the permutation of $\mathcal{S}_k$ such that $R_j(x) = jx + \left\lfloor \frac{x}{i} \right\rfloor \mod k$.

There are $k-1$ pseudo-cyclic permutations with $2k$ elements. Hence, it is easy to evaluate the diffusion round of all the pseudo-cyclic permutations for k relatively large. We reported the minimal diffusion round of pseudo-cyclic permutations for k up to 150 in Figure 4. In this figure, we distinguished the case where k is prime, since in this case, the diffusion round is really close to its lower bound: experimentally, we observe that it is never more than 2 rounds distant from the Fibonacci lower bound. For $k \in \{11, 14, 29, 59, 61, 101, 145, 149\}$, the Fibonacci lower bound is even reached, which is a surprising result as this lower bound is not tight for smaller values of k: From [SM10, CGT19, DFLM19, DDGP22], we know that this lower bound cannot be reached for $k \in \{5, 6, 7, 8, 10, 12, 13\}$.

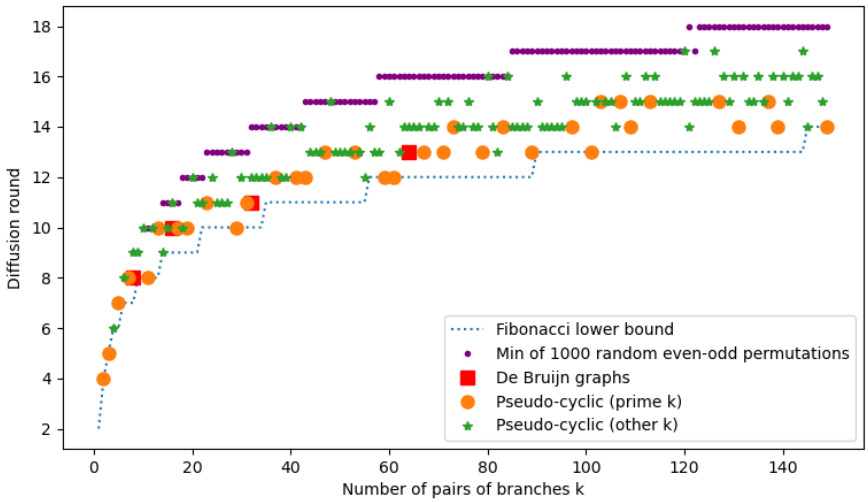


Figure 4: Comparison between the Fibonacci lower bound and the two known families of GFNs with good diffusion properties. Experimental results about random even-odd permutations are also given for comparison.

In the Figure 4, we compare the pseudo-cyclic permutations with permutations obtained from De Bruijn graphs. Indeed in [SM10, CGT19], the authors showed that for k a power

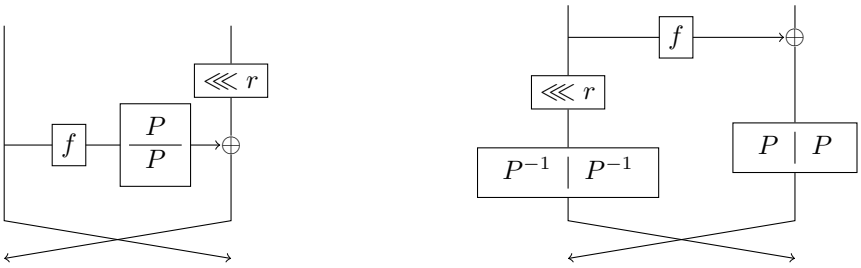


Figure 5: 2 representations of the GFN used in WARP.

of 2, permutations obtained from colouring of De Bruijn graphs are good GFN candidates regarding diffusion. One can notice that by construction, such permutations have α corresponding to the rotation of $k/2$ elements (which have order 2). Moreover, for all such good permutations exhibited by [CGT19], we computed the group generated by the $R^i \alpha R^{-i}$ for all the positive i and observed it was always of order k.

These two families are therefore complementary: Even if one deals with smooth k (powers of 2) while the other behaves better with prime k, both have good diffusion properties and more surprisingly both have the group generated by the $R^i \alpha R^{-i}$ of the same order as the number of pairs of branches. One may wonder whether there is something deeper behind this property.

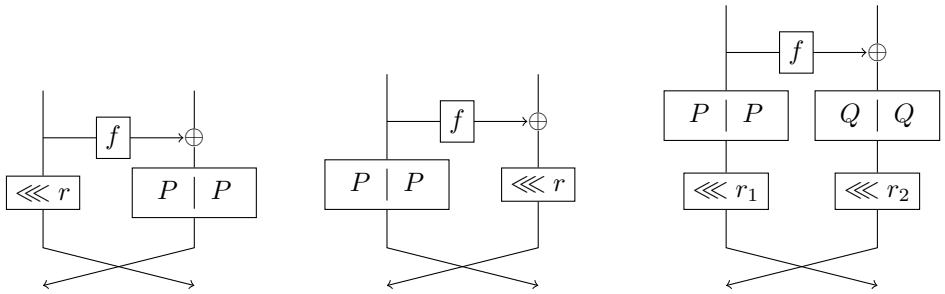
As a side note, let us observe that the α permutation of pseudo-cyclic permutations is the same as the α permutation from the original type-II GFNs with a circular shift. However, in the latter case, the diffusion round was approximately the number of branches. The same α leads therefore to GFNs with slow or fast diffusion.

5 Application to WARP

WARP [BBI+20] is a 128-bit block cipher designed with the goal of having a minimalist hardware footprint. It is based on a GFN with 32 branches. Its designers used the following strategy to search for a good 32-branch permutation:

> "We searched all permutations of LBlock-like structure that consists of one 16-branch permutation composed of two identical 8-branch permutations, and one rotation on 16 branches with an amount of rotation from 0 to 15 nibbles as shown in Fig. 3. The resulting search space has size $8! \times 16 \simeq 2^{19.3}$. The search over this space found 152 candidates of diffusion round 10.
>
> We conducted MILP-based differential AS-box counting for them. This evaluation requires about 2 days on computer equipped with 44 cores and 64 GB RAM. Among them, 21 candidates achieved AS-box of ≥ 64 (which is needed for security) at 19 rounds (and no candidates achieved it at 18 rounds), and 8 out of 21 achieved AS-box of 66, which was the largest among them. These 8 permutations are not isomorphic; however, as far as we investigated, the attack characteristics for other attacks (linear AS-box, impossible differential characteristics, etc.) are identical for all of them."

This paragraph led us to a few questions: the fact that the last eight candidates have exactly the same cryptographic properties seems to indicate a relation between them. However, they are not isomorphic, so they are maybe r-equivalent with $r \geq 2$. Moreover, as it took two days to evaluate the minimal number of active S-boxes of 152 candidates, maybe it is worth grouping the candidates by equivalence classes before it.



(a) Rotation on left branches. (b) Rotation on right branches. (c) More generic candidates.

Figure 6: Spaces explored by our program.

We reproduced the process and generated the 152 candidates of diffusion round 10. We noticed that they could be regrouped in 7 classes of extended-expanded equivalence: one class has 96 elements in it, one has 16 elements, two have 12 elements, one has 8 elements and two have 4 elements. Note that these classes are small subsets of the equivalence classes of even-odd 32-branch permutations. Indeed, the complete equivalence classes contain more than 16! permutations but only a few of them are LBlock-like permutations with one 16-branch sub-permutation being the concatenation of two identical 8-branch permutations. That is why they are not all of the same size.

It took 50 minutes on a 12-core laptop to compute the differential AS-box for these 7 candidates (and even less if we add a callback to remove candidates with strictly less than 64 active S-boxes). Exactly one class, which effectively has 8 elements, has 66 active S-boxes after 19 rounds.

Since our search for good candidates was significantly faster, we explored other spaces of 32-branch permutations in the hope of finding one permutation with minimal AS of at least 64 after only 18 rounds. We only studied permutations with diffusion round less than 10 and explored the following spaces:

- Figure 6a: Rotation on the left branches, 2 identical 8-branch permutations on the right branches.

  There are 8 such permutations with diffusion round 10 that are all in the same extended-expanded equivalence class. The minimal number of active S-boxes of this class is less than 63 in 18 rounds and reaches 66 in 19 rounds.

- Figure 6b: 2 identical 8-branch permutations on the left branches, rotation on the right branches. It is exactly the dual permutations of the previous space, and the result is similar: all the permutations with diffusion round 10 are in the same extended-expanded equivalence class. It reaches a minimal number of active S-boxes of 66 in 19 rounds.

- Figure 6c: 2 identical 8-branch permutations P followed by a rotation of an amount r_1 on the left branches, 2 identical 8-branch on the right branches Q followed by a rotation of an amount r_2 of the right branches. When $r_1 = r_2 = 0$ or $r_1 = r_2 = 8$, the GFN does not achieve full diffusion.

  Notice that when $r_1 \mod 8 = 0$ and $r_2 \mod 8 = 0$, the rotations are merely swaps of the two sub-parts. In that case, the expanded equivalence classes of $\Pi_{P,Q}$ lead to equivalence classes of the whole 32-branch Feistel. Therefore, we only enumerated with the (P,Q) generated by Algorithm 2. We extrapolated this search for the others r_i as well. Keeping only candidates with diffusion round less than or equal to 10, it gives us 184 candidates belonging to 68 extended-expanded classes. 7 classes reach a

minimal number of differentially active S-boxes of 64 or more in 18 rounds (*i.e.* one round less than the permutation of `WARP`).

The 7 classes are presented in Table 3. 5 classes have a good resistance both to differential and linear attacks, while the last two have good performance against differential attacks but worse performance against linear attacks. We recall that differential and linear properties can be swapped by using dual permutations.

To conclude, we found 5 extended-expanded equivalence classes of permutations which perform better than `WARP` in the truncated differential/linear cryptanalysis setting. Indeed, they need 18 rounds (when `WARP` needs 19 rounds) to have the guarantee that at least 64 S-Boxes are active in any differential/linear trail. They also have the same diffusion round as `WARP` which enables to keep the security arguments against Impossible Differential cryptanalysis and Meet-in-the-Middle attacks. Nevertheless, we did not evaluate the hardware footprints of our suggested permutation but we hope that among the $5 \times 16!$ equivalent permutations, there are some with a low footprint.

Table 3: Performances of good 32-branch even-odd permutations. The 7 first permutations are based on the structure of Figure 6c so the explicit values of P, Q, r_1 and r_2 are given. The value of each 32-branch even-odd permutation is also given and denoted as $\mathcal{P}$. The last permutation is the `WARP` permutation, presented here for the sake of comparison. Diff. and Lin. stand for the minimal number of differential and linear actives S-boxes across 18 rounds. DR is the diffusion round.

Type	Representative of the equivalence class	Diff.	Lin.	DR
Figure 6c	$\mathcal{P}$ =[23, 28, 27, 0, 17, 4, 25, 26, 15, 2, 21, 24, 29, 30, 19, 6, 7, 12, 11, 16, 1, 20, 9, 10, 31, 18, 5, 8, 13, 14, 3, 22] P =[4, 6, 1, 5, 0, 3, 7, 2] $r_1 = 7$ Q =[2, 4, 6, 1, 5, 0, 3, 7] $r_2 = 12$	66	64	10
Figure 6c	$\mathcal{P}$ =[25, 6, 27, 4, 29, 0, 15, 8, 19, 26, 17, 30, 23, 28, 21, 2, 9, 22, 11, 20, 13, 16, 31, 24, 3, 10, 1, 14, 7, 12, 5, 18] P =[5, 6, 7, 0, 2, 1, 4, 3] $r_1 = 7$ Q =[6, 5, 3, 7, 0, 2, 1, 4] $r_2 = 13$	65	64	10
Figure 6c	$\mathcal{P}$ =[9, 20, 5, 24, 11, 22, 7, 26, 15, 16, 1, 30, 3, 28, 13, 18, 25, 4, 21, 8, 27, 6, 23, 10, 31, 0, 17, 14, 19, 12, 29, 2] P =[4, 2, 5, 3, 7, 0, 1, 6] $r_1 = 0$ Q =[2, 4, 3, 5, 0, 7, 6, 1] $r_2 = 8$	64	64	10
Figure 6c	$\mathcal{P}$ =[9, 16, 15, 8, 21, 14, 17, 20, 11, 18, 7, 10, 13, 6, 19, 12, 25, 0, 31, 24, 5, 30, 1, 4, 27, 2, 23, 26, 29, 22, 3, 28] P =[6, 3, 7, 5, 1, 0, 2, 4] $r_1 = 1$ Q =[3, 6, 4, 7, 5, 1, 0, 2] $r_2 = 1$	64	64	10
Figure 6c	$\mathcal{P}$ =[15, 8, 9, 14, 17, 10, 13, 16, 5, 12, 3, 4, 7, 2, 11, 6, 31, 24, 25, 30, 1, 26, 29, 0, 21, 28, 19, 20, 23, 18, 27, 22] P =[1, 4, 7, 5, 2, 0, 3, 6] $r_1 = 3$ Q =[5, 1, 4, 7, 6, 2, 0, 3] $r_2 = 3$	64	64	10
Figure 6c	$\mathcal{P}$ =[13, 30, 11, 0, 19, 28, 23, 4, 21, 8, 25, 6, 17, 10, 15, 2, 29, 14, 27, 16, 3, 12, 7, 20, 5, 24, 9, 22, 1, 26, 31, 18] P =[1, 0, 4, 6, 5, 7, 3, 2] $r_1 = 5$ Q =[1, 2, 0, 4, 6, 5, 7, 3] $r_2 = 14$	66	<60	10
Figure 6c	$\mathcal{P}$ =[7, 22, 9, 16, 1, 24, 11, 20, 5, 26, 13, 30, 3, 28, 15, 18, 23, 6, 25, 0, 17, 8, 27, 4, 21, 10, 29, 14, 19, 12, 31, 2] P =[3, 4, 0, 5, 2, 6, 1, 7] $r_1 = 0$ Q =[3, 0, 4, 2, 5, 7, 6, 1] $r_2 = 8$	64	<60	10

Type	Representative of the equivalence class	Diff.	Lin.	DR
WARP	$\mathcal{P}$ = [31, 6, 29, 14, 1, 12, 21, 8, 27, 2, 3, 0, 25, 4, 23, 10, 15, 22, 13, 30, 17, 28, 5, 24, 11, 18, 19, 16, 9, 20, 7, 26]	61	61	10

6 TWINE and LBlock

6.1 Extended equivalence of the permutations

LBlock [WZ11] and TWINE [SMMK13] are two block-ciphers using a 16-branch Generalised Feistel network. TWINE is designed directly as a type II GFN, while LBlock can be rewritten as one (similarly to the representation in Figure 5). Since the publication of TWINE, it is known that their linear layers are equivalent to each other (for a definition that is not made explicit) and that they are both optimal in the sense that they have both minimal diffusion (8) and that among the permutations with 8-round diffusion, they have the highest possible minimal number of differential and linear active S-Boxes (achieving 32 active S-boxes for 15 rounds).

The permutation of TWINE is $T = [5, 0, 1, 4, 7, 12, 3, 8, 13, 6, 9, 2, 15, 10, 11, 14]$. The rewritten permutation of LBlock is $L = [15, 2, 11, 6, 5, 0, 1, 4, 7, 10, 3, 14, 13, 8, 9, 12]$. With our test, we prove once again that L and T are not 1-cyclic equivalent but L and T^{-1} are.

6.2 Optimality of the permutations of TWINE and LBlock

The impact of relaxing the diffusion constraint on the minimal number of active S-boxes on a differential/linear path is still an open question. That is why, we decided to exhaust all the even-odd 16-branch permutations (according to Algorithm 2) and compute the minimal number of active S-boxes on 15 rounds of all the extended-expanded equivalence classes of permutation with diffusion round less than 12. None of them achieves 32 active S-boxes in 14 rounds, which justifies the optimality of TWINE and LBlock.

More precisely, among the 12 classes of permutations with a diffusion round of 8, there are 4 classes which achieve 32 active S-boxes in differential and linear trails in 15 rounds. These 4 classes are extended-expanded equivalent to their dual. Representatives of these classes are

$$[7, 0, 11, 4, 15, 2, 3, 8, 13, 10, 5, 12, 9, 14, 1, 6],$$
$$[13, 0, 9, 2, 15, 6, 11, 4, 5, 10, 1, 12, 7, 14, 3, 8],$$
$$[11, 2, 7, 0, 3, 6, 13, 8, 9, 4, 5, 12, 1, 14, 15, 10],$$
$$[9, 0, 7, 4, 13, 6, 11, 2, 5, 10, 3, 12, 1, 14, 15, 8].$$

Among the 950 classes of permutations with a diffusion round of 9, there are 2 classes which achieve 34 active S-boxes in differential trails in 15 rounds. The first (resp. second) one achieves 33 (resp. 30) active S-boxes in linear trails in 15 rounds. Representatives of these classes are

$$[13, 2, 9, 0, 15, 6, 11, 4, 7, 10, 1, 12, 5, 14, 3, 8],$$
$$[7, 0, 15, 4, 5, 6, 9, 2, 1, 10, 11, 12, 3, 14, 13, 8].$$

Among permutations with diffusion round 10 and 11, all the permutations achieve at most a minimal number of 33 active S-Boxes, so they are less interesting than the previous ones.

7 Application to previous results

The literature on GFNs is full of long lists of good permutation candidates. Thus, we wanted to check whether these lists could be shortened by only considering one element per equivalence class.

7.1 Optimal permutations from [SM10]

In [SM10], the authors introduced type-II GFN with generic permutations and give in the appendix optimal permutations for GFNs with 6,8,10,12,14 or 16 branches regarding diffusion. They claim to have removed isomorphic shuffles, without defining this notion. We verified it with our criterion and indeed all but two are in different classes for the extended-expanded equivalence.

The two remaining permutations are 2-cyclic-equivalent:

$$P = [5, 2, 11, 6, 13, 8, 15, 0, 3, 4, 9, 12, 1, 14, 7, 10],$$
$$Q = [1, 2, 11, 4, 9, 6, 7, 8, 15, 12, 5, 10, 3, 0, 13, 14].$$

Indeed $Q = A_1^{-1} P A_0$ and $Q^2 = A_0^{-1} P^2 A_0$ with:

$$A_0 = [6, 7, 10, 11, 2, 3, 14, 15, 0, 1, 4, 5, 12, 13, 8, 9] = \Phi_{[3,5,1,7,0,2,6,4],[3,5,1,7,0,2,6,4]},$$
$$A_1 = [14, 15, 0, 1, 12, 13, 6, 7, 10, 11, 8, 9, 2, 3, 4, 5] = \Phi_{[7,0,6,3,5,4,1,2],[7,0,6,3,5,4,1,2]}.$$

The associated GFNs are drawn in appendix in Figure 7.

7.2 Regrouping the permutations from [CGT19]

In [CGT19], the authors listed best-known permutations[5] up to 128 branches according to diffusion. They are claimed to be regrouped by "extended pair-equivalence classes" which we have called extended conjugacy-based equivalence in this article.

Indeed, permutations with less than 16 branches which are listed in that article are all in different extended conjugacy-based equivalence classes. However, this is not the case for permutations with more branches. We summarise the number of equivalence classes of permutations with more than 16 branches in Table 4.

Table 4: Equivalence classes of the permutations given in [CGT19].

Permutations with ... branches	16	20	22	24	32	64	128
Number of even-odd permutations listed	13	74	2	31	10	4	18
Number of (extended) conjugacy-based equivalence classes	13(13)	74(74)	2(2)	31(31)	7(4)	2(1)	16(9)
Number of (extended) expanded equivalence classes	12(12)	73(72)	2(2)	31(31)	5(3)	2(1)	9(6)

Most of the expanded equivalent permutations are 2-cyclic equivalent. Nonetheless, the cyclic behaviour takes up to 8 rounds to appear. For example, the following permutations are 8-cyclic equivalent (and not equivalent for fewer rounds):

[5] As a side note, we noticed a probable typo in [CGT19]. Indeed, among the 12 32-branch permutations denoted as even-odds, only 10 are actually even-odds

$P =$ [1, 2, 7, 4, 11, 8, 13, 14, 17, 18, 23, 20, 27, 24, 29, 30, 35, 32, 37, 38, 41, 42, 47, 44, 51, 48, 53, 54, 57, 58, 63, 60, 67, 64, 69, 70, 73, 74, 79, 76, 83, 80, 85, 86, 89, 90, 95, 92, 97, 98, 103, 100, 107, 104, 109, 110, 113, 114, 119, 116, 123, 120, 125, 126, 3, 0, 5, 6, 9, 10, 15, 12, 19, 16, 21, 22, 25, 26, 31, 28, 33, 34, 39, 36, 43, 40, 45, 46, 49, 50, 55, 52, 59, 56, 61, 62, 65, 66, 71, 68, 75, 72, 77, 78, 81, 82, 87, 84, 91, 88, 93, 94, 99, 96, 101, 102, 105, 106, 111, 108, 115, 112, 117, 118, 121, 122, 127, 124],

$Q =$ [3, 0, 5, 6, 9, 10, 15, 12, 19, 16, 21, 22, 25, 26, 31, 28, 33, 34, 39, 36, 43, 40, 45, 46, 49, 50, 55, 52, 59, 56, 61, 62, 65, 66, 71, 68, 75, 72, 77, 78, 81, 82, 87, 84, 91, 88, 93, 94, 99, 96, 101, 102, 105, 106, 111, 108, 115, 112, 117, 118, 121, 122, 127, 124, 1, 2, 7, 4, 11, 8, 13, 14, 17, 18, 23, 20, 27, 24, 29, 30, 35, 32, 37, 38, 41, 42, 47, 44, 51, 48, 53, 54, 57, 58, 63, 60, 67, 64, 69, 70, 73, 74, 79, 76, 83, 80, 85, 86, 89, 90, 95, 92, 97, 98, 103, 100, 107, 104, 109, 110, 113, 114, 119, 116, 123, 120, 125, 126].

7.3 Regrouping the permutations from [DFLM19]

In [DFLM19], the authors suggested some other permutations with better diffusion properties. This time, the permutations are claimed to be regrouped by conjugacy-based equivalence classes (*i.e.* 1-cylic equivalence classes), which matches our observations. However, they may be further regrouped considering extended and/or expanded equivalence.

Table 5: Equivalence classes of the permutations given in [DFLM19].

Permutations with ... branches	28	30	32	34
Number of even-odd permutations listed	9	2	4	4
Number of (extended)-1-cyclic equivalence classes	9(5)	2(2)	4(2)	4(2)
Number of (extended)-expanded equivalence classes	5(3)	2(2)	3(2)	2(2)

One can notice that, in this example, the extended-expanded equivalences classes are uniquely determined by their security evaluation listed in the paper: two permutations are equivalent as soon as they have the same value for the number of rounds for the longest Impossible Differential distinguisher, the same value for the minimal number of rounds to get a certain amount of differentially active S-boxes and the same minimal number of active S-boxes over 20 rounds.

7.4 Regrouping the permutations from [SSD+18]

In [SSD+18], alternative permutations are suggested to improve the resistance of `LBlock` and `TWINE` against Demirci-Selçuk Meet-in-the-Middle Attack.

For `LBlock`, the security against this attack of 40320 different GFN permutations is estimated and the 64 permutations giving optimal results are proposed as variants of `LBlock`. We noticed that these permutations belong to only four conjugacy-based equivalence classes and two expanded equivalence classes.

For `TWINE`, they tested all the 887040 even-odd 16-branch permutations given by the enumeration of [CGT19] and suggested 12 optimal variants of `TWINE`. We observe that they are all in the same conjugacy-based equivalence class.

These drastic reductions can be seen as an indicator that Demirci-Selçuk Meet-in-the-Middle distinguishers are preserved by expanded equivalence classes.

Moreover, in the case of `TWINE`, they could have used the enumeration of Algorithm 2. In that case, they would have had only to test $8! = 40320$ candidates and the computation would have been 22 times faster, going from two hours to a few minutes. It would have even been approximately twice faster if the candidates were regrouped by extended expanded equivalence.

8 Conclusion and perspectives

This paper brings new perspectives on GFNs and their permutations by considering bigger equivalence classes: many GFNs which were previously considered as different are actually cryptographically equivalent for a set of classical attacks. From a designer perspective, it reduces the space of GFN candidates and thus shrinks drastically the amount of time to compare their properties.

Finally, many open questions remain: Is there another representation of the permutations for which the expanded equivalence is an easy construction? Is there a more efficient way to compute the conjugacy classes? Can we characterise the classes which lead to good cryptographic properties? What is the size of the classes if we consider also non even-odd permutations? It may also have some implications for a cryptanalyst: for a given GFN, is there any equivalent GFN which is vulnerable to the attacks not taken into account here? For instance, invariant attacks rely on symmetries of the state, which may vary in the same class. One may then try to find an equivalent representation of the GFN such that an invariant attack succeeds.

References

[BBI+20] Subhadeep Banik, Zhenzhen Bao, Takanori Isobe, Hiroyasu Kubo, Fukang Liu, Kazuhiko Minematsu, Kosei Sakamoto, Nao Shibata, and Maki Shigeri. WARP : Revisiting GFN for lightweight 128-bit block cipher. In Orr Dunkelman, Michael J. Jacobson Jr., and Colin O'Flynn, editors, *SAC 2020*, volume 12804 of *LNCS*, pages 535–564. Springer, Heidelberg, October 2020.

[BMT14] Thierry P. Berger, Marine Minier, and Gaël Thomas. Extended generalized Feistel networks using matrix representation. In Tanja Lange, Kristin Lauter, and Petr Lisonek, editors, *SAC 2013*, volume 8282 of *LNCS*, pages 289–305. Springer, Heidelberg, August 2014.

[BS13] Andrey Bogdanov and Kyoji Shibutani. Generalized feistel networks revisited. *Des. Codes Cryptogr.*, 66(1-3):75–97, 2013.

[CGT19] Victor Cauchois, Clément Gomez, and Gaël Thomas. General diffusion analysis: How to find optimal permutations for generalized type-II Feistel schemes. *IACR Trans. Symm. Cryptol.*, 2019(1):264–301, 2019.

[DDGP22] Stéphanie Delaune, Patrick Derbez, Arthur Gontier, and Charles Prud'homme. New algorithm for exhausting optimal permutations for generalized feistel networks. In *International Conference on Cryptology in India*, pages 103–124. Springer, 2022.

[DFLM19] Patrick Derbez, Pierre-Alain Fouque, Baptiste Lambin, and Victor Mollimard. Efficient search for optimal diffusion layers of generalized feistel networks. *IACR Trans. Symmetric Cryptol.*, 2019(2):218–240, 2019.

[HEO05] Derek Holt, Bettina Eick, and Eamonn O'Brien. *Handbook of Computational Group Theory*. Chapman and Hall/CRC, 2005.

[HSH+06] Deukjo Hong, Jaechul Sung, Seokhie Hong, Jongin Lim, Sangjin Lee, Bon-Seok Koo, Changhoon Lee, Donghoon Chang, Jesang Lee, Kitae Jeong, Hyun Kim, Jongsung Kim, and Seongtaek Chee. HIGHT: A new block cipher suitable for low-resource device. In Louis Goubin and Mitsuru Matsui, editors, *CHES 2006*, volume 4249 of *LNCS*, pages 46–59. Springer, Heidelberg, October 2006.

[LMR15] Gregor Leander, Brice Minaud, and Sondre Rønjom. A generic approach to invariant subspace attacks: Cryptanalysis of robin, iSCREAM and Zorro. In Elisabeth Oswald and Marc Fischlin, editors, *EUROCRYPT 2015, Part I*, volume 9056 of *LNCS*, pages 254–283. Springer, Heidelberg, April 2015.

[LR88] Michael Luby and Charles Rackoff. How to construct pseudorandom permutations from pseudorandom functions. *SIAM J. Comput.*, 17(2):373–386, 1988.

[Mat95] Mitsuru Matsui. On correlation between the order of S-boxes and the strength of DES. In Alfredo De Santis, editor, *EUROCRYPT'94*, volume 950 of *LNCS*, pages 366–375. Springer, Heidelberg, May 1995.

[Nyb96] Kaisa Nyberg. Generalized feistel networks. In Kwangjo Kim and Tsutomu Matsumoto, editors, *Advances in Cryptology - ASIACRYPT '96, International Conference on the Theory and Applications of Cryptology and Information Security, Kyongju, Korea, November 3-7, 1996, Proceedings*, volume 1163 of *Lecture Notes in Computer Science*, pages 91–104. Springer, 1996.

[Ove] Math Overflow. https://mathoverflow.net/questions/41337/a-general-formula-for-the-number-of-conjugacy-classes-of-mathbbs-n-times-m/.

[S+77] Data Encryption Standard et al. Data encryption standard. *Federal Information Processing Standards Publication 46*, 1977.

[Sim70] Charles C Sims. Computational methods in the study of permutation groups. In *Computational problems in abstract algebra*, pages 169–183. Elsevier, 1970.

[SM10] Tomoyasu Suzaki and Kazuhiko Minematsu. Improving the generalized Feistel. In Seokhie Hong and Tetsu Iwata, editors, *FSE 2010*, volume 6147 of *LNCS*, pages 19–39. Springer, Heidelberg, February 2010.

[SMMK13] Tomoyasu Suzaki, Kazuhiko Minematsu, Sumio Morioka, and Eita Kobayashi. TWINE : A lightweight block cipher for multiple platforms. In Lars R. Knudsen and Huapeng Wu, editors, *SAC 2012*, volume 7707 of *LNCS*, pages 339–354. Springer, Heidelberg, August 2013.

[SSA+07] Taizo Shirai, Kyoji Shibutani, Toru Akishita, Shiho Moriai, and Tetsu Iwata. The 128-bit blockcipher CLEFIA (extended abstract). In Alex Biryukov, editor, *FSE 2007*, volume 4593 of *LNCS*, pages 181–195. Springer, Heidelberg, March 2007.

[SSD+18] Danping Shi, Siwei Sun, Patrick Derbez, Yosuke Todo, Bing Sun, and Lei Hu. Programming the Demirci-Selçuk meet-in-the-middle attack with constraints. In Thomas Peyrin and Steven Galbraith, editors, *ASIACRYPT 2018, Part II*, volume 11273 of *LNCS*, pages 3–34. Springer, Heidelberg, December 2018.

[WZ11] Wenling Wu and Lei Zhang. LBlock: A lightweight block cipher. In Javier Lopez and Gene Tsudik, editors, *ACNS 11*, volume 6715 of *LNCS*, pages 327–344. Springer, Heidelberg, June 2011.

[ZMI90] Yuliang Zheng, Tsutomu Matsumoto, and Hideki Imai. On the construction of block ciphers provably secure and not relying on any unproved hypotheses. In Gilles Brassard, editor, *CRYPTO'89*, volume 435 of *LNCS*, pages 461–480. Springer, Heidelberg, August 1990.

A Proofs

A.1 Proof of Property 3

To complete the proof given in Section 3 we still have to show that the permutations obtained are permutations of pairs of vertices. Since P is an even-odd permutation, we can distinguish between nodes associated with left branches and those associated with right branches. Indeed, left branches have two input edges and two output edges, while right branches have one input edge and one output edge. Hence, one may recognise on the graphs the edges of E_P as their extremities are of different parities from edges of E_F which go from even vertices to even vertices.

Two graphs are isomorphic if and only if they are identical up to a re-labelling of the vertices. This implies that the re-labelling keeps the arity of the vertices. Thus, even (resp. odd) vertices are re-indexed as even (resp. odd) vertices. Therefore, the role of the edges is maintained by the re-labelling. Moreover, even vertices have two antecedents which by construction are consecutive (and can be distinguished between left antecedent and right antecedent) so the Π_i (easily linked with the A_i of the definition) are merely permutations of pairs of vertices.

A.2 Proof of Property 4

We give here the proof of Property 4: $P = \Psi_R^\alpha$ and Q are expanded-equivalent if and only there exist two permutations of pairs A and $B = \Phi_{b,b}$ such that $Q = ABPA^{-1}$ and $b \in \texttt{Centr}(\{R^{-i}\alpha R^i\}_{i\geq 0})$. The proof is based on the following observation with directly comes from the fact that $PBP^{-1} = \Phi_{R\alpha b\alpha^{-1}R^{-1},R^{-1}bR}$.

Lemma 1. *Let $P = \Psi_\alpha^R$ be an even-odd permutation and $B = \Phi_{b,b}$ a permutation of pair. Then $PBP^{-1} \in \mathcal{S}_k^p$ if and only if α and b commute. In this case, $PBP^{-1} = \Phi_{RbR^{-1},RbR^{-1}}$*

We can now prove the direct implication of Property 4. $P = \Psi_R^\alpha$ and Q are expanded-equivalent, so there exists $A \in \mathcal{S}_k^p$ such that $QAP^{-1} \in \mathcal{S}_k^p$. Then $B := A^{-1}QAP^{-1} \in \mathcal{S}_k^p$ verifies $Q = ABPA^{-1}$. Moreover, for all i,

$$\underbrace{Q^{i+1}AP^{-i-1}}_{\in\mathcal{S}_k^p} = Q^iQAP^{-1}P^{-i} = Q^iABP^{-i} = \underbrace{Q^iAP^{-i}}_{\in\mathcal{S}_k^p}P^iBP^{-i}$$

and thus for all i, $B_i := P^iBP^{-i} \in \mathcal{S}_k^p$.

Then, one can use the lemma to prove by induction that for all i, $B_i = \Phi_{R^ibR^{-i},R^ibR^{-i}}$ and that α commutes with R^ibR^{-i} ie that b commutes with $R^i\alpha R^{-i}$.

Conversely, if $b \in \texttt{Centr}(\{R^{-i}\alpha R^i\}_{i\geq 0})$ then for all i, α commutes with R^ibR^{-i} and thus applying the lemma, one can show by induction that $P^iBP^{-i} \in \mathcal{S}_k^p$ and finally that $Q^iAP^{-i} \in \mathcal{S}_k^p$.

A.3 Proof of Theorem 1

We need two preliminary lemmas to prove this theorem.

Lemma 2. *Let $\alpha, R \in \mathcal{S}_k$. Let $G := \mathcal{S}_k/_{\texttt{Centr}(\alpha)}$. Then $\{g\alpha g^{-1}, g \in G\}$ is the set of conjugates of α without repetition.*

Moreover, $\texttt{Cl}(\Psi_R^\alpha)$ can be partitioned in the following way:

$$\texttt{Cl}(\Psi_R^\alpha) = \bigcup_{g\in G}\left\{\Psi_{gABRA^{-1}g^{-1}}^{g\alpha g^{-1}}, A \in \texttt{Centr}(\alpha), B \in \texttt{Centr}(\{R^{-i}\alpha R^i\}_{i\geq 0})\right\}$$

All of these subsets are of the same size, denoted $\mathcal{N}_{\alpha,R}$*, and thus* $|\mathtt{Cl}(\Psi_R^\alpha)| = |G|\mathcal{N}_{\alpha,R}$*, with*

$$\mathcal{N}_{\alpha,R} := \left|\left\{ABRA^{-1}, A \in \mathtt{Centr}(\alpha), B \in \mathtt{Centr}(\left\{R^{-i}\alpha R^i\right\}_{i\geq 0})\right\}\right|.$$

Proof. From the characterisation of expanded classes, we have,

$$\mathtt{Cl}(\Psi_R^\alpha) = \left\{\Psi_{ABRA^{-1}}^{A\alpha A^{-1}}, A \in \mathcal{S}_k, B \in \mathtt{Centr}(\left\{R^{-i}\alpha R^i\right\}_{i\geq 0})\right\}$$

which can be partitioned according to the value of $A\alpha A^{-1}$. □

Lemma 3. *Let* $\alpha, R \in \mathcal{S}_k$*, then* $\mathcal{N}_{\alpha,R} = |\mathtt{Centr}(\alpha)|$.

Proof. The idea of the proof is to show that for any permutation B in $\mathtt{Centr}(\left\{R^{-i}\alpha R^i\right\}_{i\geq 0})$ and any permutation A in $\mathtt{Centr}(\alpha)$, there exists a permutation $B' \in \mathtt{Centr}(\left\{R^{-i}\alpha R^i\right\}_{i\geq 0})$ such that $ABRA^{-1}$ and $B'R$ are equal if and only if $A \in \mathtt{Centr}(\left\{R^{-i}\alpha R^i\right\}_{i\geq 0})$. To do this, we will prove that $ABRA^{-1}R^{-1} \in \mathtt{Centr}(\left\{R^{-i}\alpha R^i\right\}_{i\geq 0})$ if and only if $A \in \mathtt{Centr}(\left\{R^{-i}\alpha R^i\right\}_{i\geq 0})$. The result then comes directly from the partition of the space on conjugacy classes of cosets.

Let us first prove the direct implication. We have $B, B' \in \mathtt{Centr}(\left\{R^{-i}\alpha R^i\right\}_{i\geq 0})$ such that $ABRA^{-1} = B'R$. Let us prove by induction that A commutes with $R^i\alpha R^{-i}$. The base case $i = 0$ is trivial, since by definition A commutes with α. Moreover, the induction works as follows: let us suppose that A commutes with $R^i\alpha R^{-i}$. Then, by Property 6, $AR^{i+1}\alpha R^{-i-1}A^{-1} = A(BR)^{i+1}\alpha(BR)^{-i-1}A^{-1}$ which is also equal to

$$\underbrace{ABRA^{-1}}_{(1)}\underbrace{A(BR)^i\alpha(BR)^{-i}A^{-1}}_{(2)}\underbrace{A(BR)^{-1}A^{-1}}_{(3)}.$$

(1) and (3): By hypothesis, $ABRA^{-1} = B'R$ and $A(BR)^{-1}A^{-1} = (B'R)^{-1}$

(2): By Property 6, $A(BR)^i\alpha(BR)^{-i}A^{-1} = AR^i\alpha R^{-i}A^{-1}$. By induction hypothesis, it is also equal to $R^i\alpha R^{-i}$.

Finally, $AR^{i+1}\alpha R^{-i-1}A^{-1} = B'RR^i\alpha R^{-i}(B'R)^{-1} = B'R^{i+1}\alpha R^{-i-1}B'^{-1}$ which concludes since $B \in \mathtt{Centr}(\left\{R^{-i}\alpha R^i\right\}_{i\geq 0})$.

Let us now prove the converse part: we have A and B in $\mathtt{Centr}(\left\{R^{-i}\alpha R^i\right\}_{i\geq 0})$ and we want to prove that $ABRA^{-1}R^{-1} \in \mathtt{Centr}(\left\{R^{-i}\alpha R^i\right\}_{i\geq 0})$ so it remains to prove that $RA^{-1}R^{-1} \in \mathtt{Centr}(\left\{R^{-i}\alpha R^i\right\}_{i\geq 0})$. Yet, the equation $(RA^{-1}R^{-1})(R^{-i}\alpha R^i) = (R^{-i}\alpha R^i)(RA^{-1}R^{-1})$ is equivalent to the equation $AR^{-i-1}\alpha R^{i+1} = R^{-i-1}\alpha R^{i+1}A$ *i.e.* to $A \in \mathtt{Centr}(R^{i+1}\alpha R^{-i-1})$. So we easily have $RA^{-1}R^{-1} \in \mathtt{Centr}(\left\{R^{-i}\alpha R^i\right\}_{i\geq 0})$.

□

Let us finally prove Theorem 1:
Let $\Pi_{L,R}$ an even-odd permutation with $2k$-branches.
Let $\alpha := R^{-1}L$ in order to have $\Pi_{L,R} = \Psi_R^\alpha$. Let $G := \mathcal{S}_k/\mathtt{Centr}(\alpha)$.
From Lemma 2 and 3, we have that $|\mathtt{Cl}(\Pi_{L,R})| = |\mathtt{Cl}(\Psi_R^\alpha)| = |G||\mathtt{Centr}(\alpha)|$.
Therefore, $|\mathtt{Cl}(\Pi_{L,R})| = \dfrac{k!}{|\mathtt{Centr}(\alpha)|}|\mathtt{Centr}(\alpha)| = k!$.

B Comparison of the timings of the different algorithms of Section 4

In Section 4, three algorithms are described to enumerate even-odd permutations with $2k$ branches. The first one provides at least one representative per conjugacy-based

equivalence class while the second one provides exactly one representative per conjugacy-based equivalence class. The last one provides one representative per expanded equivalence class.

Table 6: Timings of our Python implementations of the 3 algorithms described in Section 4

k	Enumeration from [CGT19]	Algorithm 1	Algorithm 2
5	0.04s	0.03s	0.10s
6	0.15s	0.22s	0.92s
7	1.54s	2.31s	9.12s
8	20.3 s	29.05s	104.77s
9	262.81s	388.77s	1306.89s

C Expanded equivalence classes for small GFNs

We give here one representative permutation P per expanded equivalence class for even-odd GFNs. DR stands for the maximum of the diffusion round of P and P^{-1}. Diff. and Lin. stand for the differential and linear minimal number of active S-boxes on 20 rounds. We also give the index of the class of the dual permutation and of the inverse permutation.

C.1 $k = 2$

No. of the class	P	DR	Diff.	Lin.	No. of the dual class	No. of the inverse class
0	[1,2,3,0]	4	19	19	0	0
1	[3,2,1,0]	100	13	13	1	1

C.2 $k = 3$

No. of the class	P	DR	Diff.	Lin.	No. of the dual class	No. of the inverse class
0	[3,4,1,2,5,0]	5	25	25	0	0
1	[1,4,3,0,5,2]	6	25	25	1	1
2	[5,4,3,0,1,2]	8	21	21	3	3
3	[5,4,1,2,3,0]	8	21	21	2	2
4	[1,0,5,2,3,4]	100	13	13	4	4
5	[5,4,1,0,3,2]	100	13	13	5	5

C.3 $k = 4$

No. of the class	P	DR	Diff.	Lin.	No. of the dual class	No. of the inverse class
0	[3,4,1,2,7,0,5,6]	6	30	30	0	0
1	[7,2,3,6,1,4,5,0]	6	26	26	1	1
2	[7,2,5,6,1,4,3,0]	7	28	28	3	3
3	[7,2,1,6,5,0,3,4]	7	28	28	2	2
4	[5,0,3,4,7,2,1,6]	7	28	28	5	5
5	[1,4,3,0,7,2,5,6]	7	28	28	4	4
6	[1,6,7,0,5,2,3,4]	7	25	25	6	6
7	[1,6,3,0,5,2,7,4]	8	27	27	7	7

No. of the class	P	DR	Diff.	Lin.	No. of the dual class	No. of the inverse class
8	[3,2,5,6,1,4,7,0]	8	24	24	8	8
9	[3,2,1,4,7,0,5,6]	8	24	24	10	10
10	[7,6,3,4,1,2,5,0]	8	24	24	9	9
11	[7,6,3,0,5,2,1,4]	10	19	19	12	12
12	[5,4,1,2,7,0,3,6]	10	19	19	11	11
13	[7,6,1,4,3,0,5,2]	10	13	13	13	13
14	[7,6,1,0,5,2,3,4]	10	13	13	15	15
15	[3,2,7,6,1,4,5,0]	10	13	13	14	14
16	[1,0,7,4,3,6,5,2]	100	13	13	16	16
17	[1,0,3,4,7,2,5,6]	100	13	13	17	17
18	[5,4,3,2,7,0,1,6]	100	13	13	19	19
19	[1,0,5,4,3,6,7,2]	100	13	13	18	18
20	[7,6,5,4,1,2,3,0]	100	18	18	20	20
21	[3,2,1,0,5,6,7,4]	100	13	13	21	21
22	[7,6,1,0,3,2,5,4]	100	13	13	22	22
23	[5,6,7,4,1,2,3,0]	100	19	19	23	23

C.4 $k = 5$

In order to keep the list reasonably short, we only print permutations with diffusion round less than 8. Full versions for k up to 8 are available in supplementary materials.

No. of the class	P	DR	Diff.	Lin.	No. of the dual class	No. of the inverse class
0	[9,0,7,8,5,6,3,4,1,2]	7	33	33	0	0
1	[7,2,1,6,5,0,9,4,3,8]	7	35	35	1	2
2	[7,0,3,6,9,2,5,8,1,4]	7	35	35	2	1
3	[7,0,1,6,9,4,3,8,5,2]	7	34	34	4	4
4	[9,2,3,8,5,6,1,4,7,0]	7	34	34	3	3
5	[5,8,1,4,3,0,7,2,9,6]	8	29	29	6	6
6	[3,0,7,2,9,6,5,8,1,4]	8	29	29	5	5
7	[3,4,9,2,1,8,7,0,5,6]	8	21	21	7	7
8	[1,2,5,0,9,4,7,8,3,6]	8	32	32	9	9
9	[9,0,3,8,7,2,5,6,1,4]	8	32	32	8	8
10	[7,0,5,6,9,4,3,8,1,2]	8	34	34	11	11
11	[9,4,3,8,1,2,7,0,5,6]	8	34	34	10	10
12	[5,8,3,4,1,2,7,0,9,6]	8	32	32	13	13
13	[7,0,5,6,3,4,9,2,1,8]	8	32	32	12	12
14	[7,2,1,6,9,0,5,8,3,4]	8	26	26	14	14
15	[7,0,3,6,5,2,9,4,1,8]	8	31	31	15	15
16	[9,8,1,4,3,0,7,2,5,6]	8	13	26	17	21
17	[7,6,1,4,3,0,9,2,5,8]	8	26	13	16	20
18	[9,8,3,4,1,2,7,0,5,6]	8	25	25	19	19
19	[5,4,9,2,7,8,1,6,3,0]	8	25	25	18	18
20	[9,8,1,6,5,0,3,4,7,2]	8	26	13	21	17
21	[3,2,9,4,7,8,1,6,5,0]	8	13	26	20	16
22	[5,4,9,2,3,8,7,0,1,6]	8	19	13	23	22
23	[7,6,5,0,1,4,3,8,9,2]	8	13	19	22	23
24	[3,4,5,2,1,8,7,0,9,6]	8	30	30	25	25
25	[5,2,3,4,7,0,9,6,1,8]	8	30	30	24	24

No. of the class	P	DR	Diff.	Lin.	No. of the dual class	No. of the inverse class
26	[7,8,9,6,3,0,5,2,1,4]	8	29	29	27	27
27	[5,6,7,4,9,0,3,8,1,2]	8	29	29	26	26
28	[9,0,1,8,5,6,3,4,7,2]	8	32	32	29	29
29	[1,6,7,0,9,4,3,8,5,2]	8	32	32	28	28
30	[7,2,3,6,9,0,5,8,1,4]	8	31	31	31	31
31	[9,0,1,8,3,6,5,2,7,4]	8	31	31	30	30
33	[7,2,3,6,5,0,9,4,1,8]	8	32	32	33	33
34	[7,0,1,6,5,2,9,4,3,8]	8	32	32	32	32

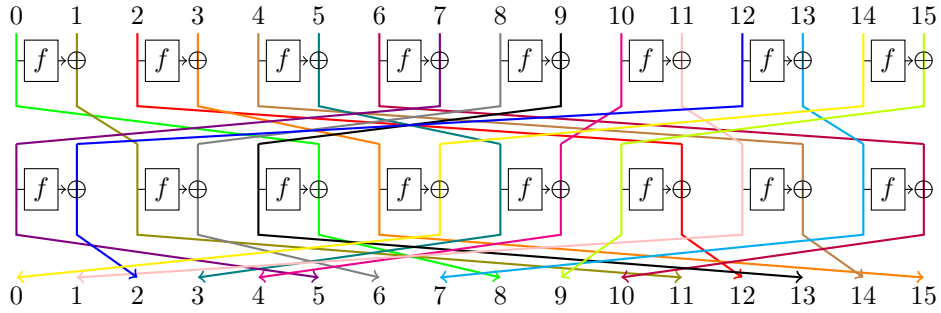
D Isomorphism between two 16-branch permutations

Figure 7 is a graphical representation of the 2-cyclic equivalence of

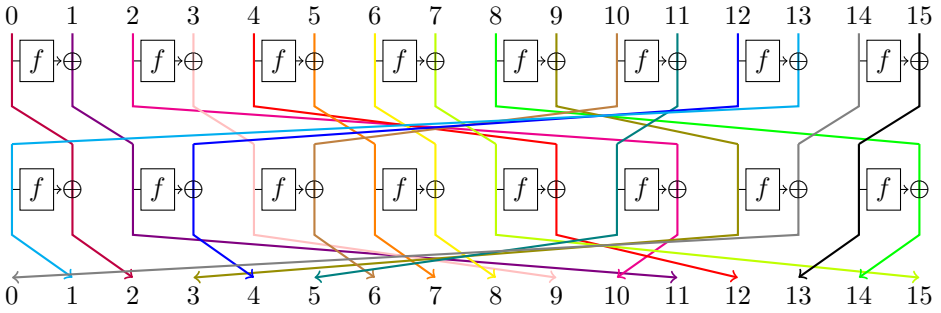
$$\begin{aligned} P &= [5, 2, 11, 6, 13, 8, 15, 0, 3, 4, 9, 12, 1, 14, 7, 10] \\ \text{and} \quad Q &= [1, 2, 11, 4, 9, 6, 7, 8, 15, 12, 5, 10, 3, 0, 13, 14]. \end{aligned}$$

They are linked by the relations $Q = A_1^{-1} P A_0$ and $Q^2 = A_0^{-1} P^2 A_0$ with:

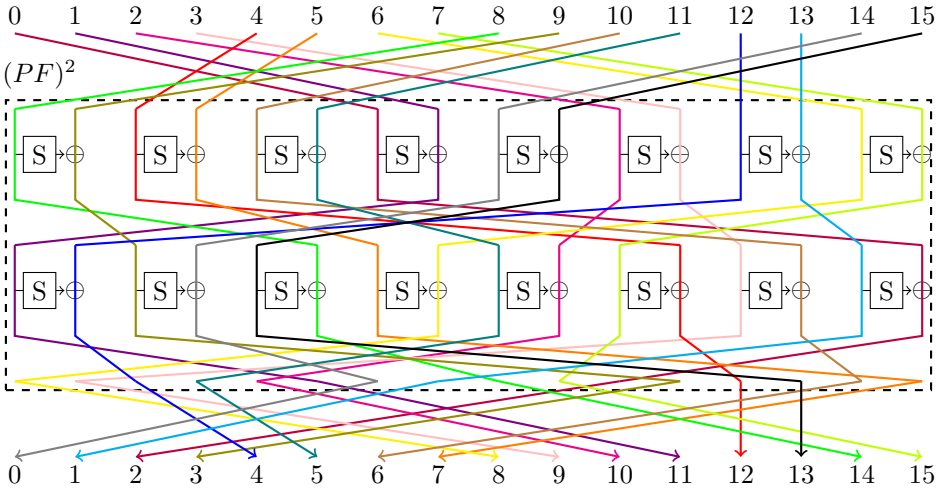
$$A_0 = \Phi_{[3,5,1,7,0,2,6,4],[3,5,1,7,0,2,6,4]} \text{ and } A_1 = \Phi_{[7,0,6,3,5,4,1,2],[7,0,6,3,5,4,1,2]}$$



(a) 2 rounds of the Feistel associated to $P : (PF)^2$.



(b) 2 rounds of the Feistel associated to Q : $(QF)^2$.



(c) 2-rounds of the Feistel associated to Q written as $A_0^{-1}(PF)^2A_0$.

Figure 7: Isomorphism between P and Q from [SM10].