IACR Transactions on Symmetric Cryptology
ISSN 2519-173X, Vol. 2023, No. 2, pp. 94–131. DOI:10.46586/tosc.v2023.i2.94-131

Bounded Surjective Quadratic Functions over $\mathbb{F}_p^n$ for MPC-/ZK-/FHE-Friendly Symmetric Primitives

Lorenzo Grassi

Ruhr University Bochum, Bochum, Germany
Lorenzo.Grassi@ruhr-uni-bochum.de

Abstract. Motivated by new applications such as secure Multi-Party Computation (MPC), Fully Homomorphic Encryption (FHE), and Zero-Knowledge proofs (ZK), many MPC-, FHE- and ZK-friendly symmetric-key primitives that minimize the number of multiplications over $\mathbb{F}_p$ for a large prime p have been recently proposed in the literature. These symmetric primitives are usually defined via invertible functions, including (i) Feistel and Lai-Massey schemes and (ii) SPN constructions instantiated with invertible non-linear S-Boxes. However, the "invertibility" property is actually never required in any of the mentioned applications.

In this paper, we discuss the possibility to set up MPC-/FHE-/ZK-friendly symmetric primitives instantiated with non-invertible *bounded surjective* functions. In contrast to one-to-one functions, each output of a l-bounded surjective function admits at most l pre-images. The simplest example is the square map $x \mapsto x^2$ over $\mathbb{F}_p$ for a prime $p \geq 3$, which is (obviously) 2-bounded surjective. When working over $\mathbb{F}_p^n$ for $n \geq 2$, we set up bounded surjective functions by re-considering the recent results proposed by Grassi, Onofri, Pedicini and Sozzi at FSE/ToSC 2022 as starting points. Given a quadratic local map $F : \mathbb{F}_p^m \to \mathbb{F}_p$ for $m \in \{1, 2, 3\}$, they proved that the shift-invariant non-linear function over $\mathbb{F}_p^n$ defined as $\mathcal{S}_F(x_0, x_1, \ldots, x_{n-1}) = y_0 \| y_1 \| \ldots \| y_{n-1}$ where $y_i := F(x_i, x_{i+1})$ is never invertible for any $n \geq 2 \cdot m - 1$. Here, we prove that

- the quadratic function $F : \mathbb{F}_p^m \to \mathbb{F}_p$ for $m \in \{1, 2\}$ that minimizes the probability of having a collision for $\mathcal{S}_F$ over $\mathbb{F}_p^n$ is of the form $F(x_0, x_1) = x_0^2 + x_1$ (or equivalent);
- the function $\mathcal{S}_F$ over $\mathbb{F}_p^n$ defined as before via $F(x_0, x_1) = x_0^2 + x_1$ (or equivalent) is 2^n-bounded surjective.

As concrete applications, we propose modified versions of the MPC-friendly schemes MiMC, HadesMiMC, and (partially of) Hydra, and of the FHE-friendly schemes Masta, Pasta, and `Rubato`. By instantiating them with the bounded surjective quadratic functions proposed in this paper, we are able to improve the security and/or the performances in the target applications/protocols.

Keywords: Bounded Surjective Functions · Local Maps · Quadratic Functions

1 Introduction

Almost all the symmetric primitives published in the literature – including ciphers, Pseudo-Random Functions/Permutations (PRFs/PRPs), hash functions – are typically designed by iterating an efficiently implementable round function a sufficient number of times such that the resulting composition satisfies the security requirements. Even if not strictly necessary in many scenarios (e.g., stream ciphers, hash functions, and so on), the round function is usually *invertible*, that is, it is instantiated either via invertible components, or

Received: 2022-11-23 Revised: 2023-03-01 Accepted: 2023-05-01 Published: 2023-06-16

Table 1: Invertible non-linear round functions that instantiate MPC-/FHE-/ZK-friendly symmetric primitives over $\mathbb{F}_p^n$ proposed in the literature. Some primitives are instantiated via several non-linear functions ("Others" include the `Horst` scheme and look-up tables).

Symmetric Primitive	Invertible Power Map(s)	Non-Invertible Function(s) in Feistel/Lai-Massey Scheme	Others
MiMC [AGR+16]	✓		
GMiMC [AGP+19]	✓		
HadesMiMC [GLR+20]	✓		
Rescue [AAB+20]	✓		
Poseidon [GKR+21]	✓		
Ciminion [DGGK21]		✓	
Grendel [Sze21]	✓		
Pasta [DGH+21]	✓	✓	
`Reinforced Concrete` [GKL+22]	✓		✓
Neptune [GOPS22]	✓	✓	
Griffin [GHR+22]	✓		✓
Chaghri [AMT22]	✓		
Hydra [GØSW23]	✓	✓	
Anemoi [BBC+22]	✓	✓	

in such a way that, even if the components are not invertible by their own, the overall round function is invertible (as in the case of Feistel and/or Lai-Massey schemes).

In many cases, this choice is crucial for guaranteeing the security (or/and for simplifying the security analysis). Consider the case of a Substitution-Permutation Network (SPN), in which the non-linear layer is instantiated via a concatenation of independent S-Boxes, i.e., $(x_0, x_1, \ldots, x_{n-1}) \mapsto (S(x_0), S(x_1), \ldots, S(x_{n-1}))$ for a certain non-linear function S over a field $\mathbb{F}_q$ for a *small* $q = p^s$ (usually, $q \leq 2^8$). If S is not invertible, finding a collision at the output of any single function S could potentially allow the attacker to break the entire scheme. As an example, the hash function SHA-3/Keccak [BDPA13] instantiated with 5-bit *non-invertible* S-Boxes may be easily broken by looking for a collision at the output of the first rounds, set up via input messages that activate a single (or few) S-Box(es).

The scenario is potentially different in the case of symmetric primitives defined over a field $\mathbb{F}_q$ for a *large* q, as for the symmetric primitives designed for being efficient in Secure Multi-Party Computation (MPC), Fully Homomorphic Encryption (FHE), and/or Zero-Knowledge proofs (ZK). Unlike in the case of "traditional" cipher design, the size of the field over which these MPC-/FHE-/ZK-friendly schemes are defined has basically *no impact* on the performance of such applications/protocols. As a result, these MPC-/FHE-/ZK-friendly schemes – listed in Table 1 – operate over $\mathbb{F}_p^n$ for a large prime $p \gg 3$, e.g., p is of order 2^{64}, 2^{128} or even bigger. In such a scenario:

- the *invertibility property is required* neither in MPC-/FHE-applications nor in ZK protocols. Indeed, MPC and FHE applications require a PRF scheme (which is not invertible in general), while ZK protocols requires a hash function (which is not invertible by definition);
- due to the large size of the integer q, finding collisions could be much more expensive than the maximum data and/or complexity cost allowed for setting up an attack (as we will show in the following).

Moreover, to the best of our knowledge, the inverse of all the ciphers listed in Table 1 is never used in practice. Consider the case of the block ciphers MiMC and HadesMiMC, both instantiated with invertible power maps $x \mapsto x^d$ defined over $\mathbb{F}_p$ for $d \geq 3$ such that $\gcd(d, p-1) = 1$. The inverse of such power maps are again power maps of the form $x \mapsto x^{d'}$, where $d' \geq 3$ is the smallest integer for which $d \cdot d' - 1$ is a multiple of $p - 1$.[1]

[1] We recall that $x^{p-1} = 1$ for each $x \in \mathbb{F}_p \setminus \{0\}$ due to *Fermat's little theorem*.

For small values of d, the exponent d' is of the same order of magnitude as p, i.e., much bigger than d, which implies that decrypting is much more expensive than encrypting, a property that is in general *not* desirable in practical use cases. For this reason, MiMC's and HadesMiMC's designers suggest to use such schemes in a mode of operation in which the inverse is not needed: "*[...] decryption is much more expensive than encryption. Using modes where the inverse is not needed is thus advisable*" (see [AGR+16, Sect. 1]).

Having said that, almost all the MPC-/FHE-/ZK-friendly symmetric cryptographic primitives that have been recently proposed in the literature for minimizing the number of field non-linear operations in their natural algorithmic description – often referred to as the *multiplicative complexity* – are instantiated via invertible round functions only,[2] as summarized in Table 1 (to the best of our knowledge, the only scheme instantiated via non-invertible components is the FHE-friendly scheme Masta [HKC+20], which is based on the Rasta design strategy discussed in Sect. 7). Hence, natural questions arise: since the invertibility property is not required in MPC-/FHE-/ZK-applications, what happens when considering a symmetric primitive instantiated with non-invertible round functions? Can we decrease the multiplicative complexity without affecting its security?

In order to answer these questions, in this paper we start a research regarding quadratic *non-invertible* functions over $\mathbb{F}_p^n$ that can be used as building blocks in MPC-/FHE-/ZK-friendly symmetric primitives.

1.1 Bounded Surjective Functions constructed via Local Maps

When considering non-invertible functions, an estimation of the probability of the collision event is paramount. From this point of view, it is desirable that the used non-invertible function admits a *(fixed) maximum number of pre-images* for each possible output.

"Bounded Surjective" Functions. For this reason, in Sect. 3, we introduce the concept of *"bounded surjective" functions*. Given $l \geq 1$, each output of a l-bounded surjective function admits *at most* l distinct pre-images. Thus, a bijective function is a 1-bounded surjective function, since each output admits exactly one pre-image, while the square function $x \mapsto x^2$ is a 2-bounded surjective function.

Bounded Surjective Functions over $\mathbb{F}_p^n$. When working over $\mathbb{F}_p^n$ for $n \geq 2$, one can potentially set up bounded surjective functions by concatenating bounded surjective functions over $\mathbb{F}_p$. For example, it is not hard to check that the function $(x_0, x_1, \ldots, x_{n-1}) \mapsto (x_0^2, x_1^2, \ldots, x_{n-1}^2)$ is 2^n-bound surjective, since $x \mapsto x^2$ is 2-bounded surjective, and the square map operates independently on each word.

Our starting point for setting up a l-bound surjective function with l smaller than 2^n are the recent results proposed by Grassi et al. [GOPS22] at FSE/ToSC 2022 regarding the case of SI-lifting functions.

Definition 1. Let $p \geq 3$ be a prime integer. Let $1 \leq m \leq n$, and let $F : \mathbb{F}_p^m \to \mathbb{F}_p$ be a non-linear function defined as

$$F(x_0, x_1, \ldots, x_{m-1}) := \sum_{0 \leq i_0 + i_1 + \ldots + i_{m-1} \leq d} \alpha_{i_0, i_1, \ldots, i_{m-1}} \cdot \prod_{j=0}^{m-1} x_j^{i_j} \,,$$

where $d \geq 1$, $i_0, i_1, \ldots, i_{m-1} \geq 0$ are integers, and $\alpha_{i_0, i_1, \ldots, i_{m-1}} \in \mathbb{F}_p$. The Shift-Invariant (m, n)-lifting (for simplicity, SI-lifting) function $\mathcal{S}_F$ over $\mathbb{F}_p^n$ induced by F is defined as

[2] In this paper, we use the term "$\mathbb{F}_p$-multiplication" – or simply, "multiplication" – to refer to the field non-linear operation $(x, y) \in \mathbb{F}_p^2 \mapsto x \cdot y \in \mathbb{F}_p$. Moreover, we do not make any distinction between a $\mathbb{F}_p$-multiplication and a $\mathbb{F}_p$-square operation, since – to the best of our knowledge – they have the same cost in the considered applications/protocols.

$\mathcal{S}_F(x_0, x_1, \ldots, x_{n-1}) \coloneqq y_0 \| y_1 \| \ldots \| y_{n-1}$ where

$$\forall i \in \{0, 1, \ldots, n-1\} : \qquad y_i \coloneqq F(x_i, x_{i+1}, \ldots, x_{i+m-1}) \,, \tag{1}$$

where the sub-indexes are taken modulo n.

In [GOPS22], authors proved that, given any quadratic function $F : \mathbb{F}_p^2 \to \mathbb{F}_p$, the corresponding function $\mathcal{S}_F$ over $\mathbb{F}_p^n$ for $n \geq 3$ as defined in Def. 1 is never invertible. An equivalent similar result holds when considering quadratic functions $F : \mathbb{F}_p^3 \to \mathbb{F}_p$ and the corresponding function $\mathcal{S}_F$ over $\mathbb{F}_p^n$ for $n \geq 5$.

Due to the possibility to compute several of these non-invertible functions $\mathcal{S}_F$ over $\mathbb{F}_p^n$ via n multiplications only, these functions seem to be optimal candidates for our goals. By re-analyzing them, in Sect. 5 we prove that, among *all* quadratic functions $F : \mathbb{F}_p^m \to \mathbb{F}_p$ for $m \in \{1, 2\}$ for which $\mathcal{S}_F$ can be computed via n multiplications only, the function $\mathcal{S}_F$ over $\mathbb{F}_p^n$ for $n \geq 3$ induced by $F(x_0, x_1) = x_0^2 + \alpha_{1,0} \cdot x_0 + \alpha_{0,1} \cdot x_1$ with $\alpha_{0,1} \neq 0$ (or equivalently by $F(x_0, x_1) = x_1^2 + \alpha_{1,0} \cdot x_0 + \alpha_{0,1} \cdot x_1$ with $\alpha_{1,0} \neq 0$):

- minimizes the probability that a collision occurs for $\mathcal{S}_F$ over $\mathbb{F}_p^n$ for $n \geq 3$, which is upper bounded by p^{-n};
- is a 2^n-bounded surjective function.

Compared to $(x_0, x_1, \ldots, x_{n-1}) \mapsto (x_0^2, x_1^2, \ldots, x_{n-1}^2)$, we show that (i) the probability that a collision occurs for $\mathcal{S}_F$ just defined is much smaller (approximately by a factor $2^n - 1$), and that (ii) a collision $\mathcal{S}_F(x) = \mathcal{S}_F(y)$ can occur only in the case in which $x_i \neq y_i$ for each $i \in \{0, 1, \ldots, n-1\}$.

Open Problem. The problem to set up a l-bounded surjective function over $\mathbb{F}_p^n$ with (i) $l < 2^n$ and (ii) that can be computed via n multiplications, is left open for future work.

1.2 Impact on MPC- and FHE-Friendly Schemes

Even if the function $\mathcal{S}_F$ over $\mathbb{F}_p^n$ induced by $F(x_0, x_1) = x_0^2 + \alpha_{1,0} \cdot x_0 + \alpha_{0,1} \cdot x_1$ with $\alpha_{0,1} \neq 0$ (or $F(x_0, x_1) = x_0^2 + x_1$ for simplicity) is not invertible, it can be suitable for instantiating a non-invertible symmetric primitive, due to its several beneficial properties just listed. In order to show concrete evidences of this fact, we re-consider some MPC-/FHE-friendly symmetric primitives proposed in the literature, and we propose some variants of them instantiated with the bounded surjective function just proposed with the goal of getting better results in terms of performance and/or security.

1.2.1 MPC-Friendly Schemes: `MiMC++`, Pluto, and Hydra++

MPC protocols allow several parties to jointly compute a function over their inputs, without exposing these inputs. In the most common case in which MPC protocols are evaluated via linearly homomorphic secret sharing schemes, multiplications require communication between the parties, while affine operations can be evaluated locally. In such a case, the MPC cost metric is related to the number of multiplications needed to evaluate the symmetric scheme.

`MiMC++`. MiMC is an iterative Even-Mansour scheme, whose round function is instantiated via the invertible power map $x \mapsto x^d$. As a simple concrete example of a scheme instantiated with a 2-bounded surjective function, in Sect. 4 we propose the PRF `MiMC++`, a version of MiMC in which the invertible power map is replaced by the square one $x \mapsto x^2$. In order to guarantee the same security level of MiMC, the size p' of the field $\mathbb{F}_{p'}$ over which `MiMC++` operates must be triple with respect to the one used in MiMC, that is, $p' \approx p^3$

(where p is the prime number that defines the field of MiMC). At the same time, replacing $x \mapsto x^d$ with $x \mapsto x^2$ allows to decrease the multiplicative complexity, e.g., by a factor 27.5% for a security level of 128 bits (where $p \approx 2^{128}$ and $p' \approx 2^{384}$).

Pluto and Hydra++. The security of `MiMC++` strictly depends on the fact that the prime p' is much larger than the security level. This requirement is not necessary in the case of block ciphers defined over $\mathbb{F}_p^n$ such as HadesMiMC: given a certain security level and a fixed prime p, the security can be achieved by choosing an appropriate value of n.

The main characteristic of the Hades design strategy [GLR+20] is the uneven distribution of the S-Boxes through the rounds. The external rounds are instantiated with a full S-Box layer that provides security against the statistical attacks, while the internal rounds are instantiated with a partial S-Box layer, which increases the overall degree of the scheme while minimizing the cost. This strategy has been recently generalized in `Reinforced Concrete`, Neptune, and Hydra's body. Instead of having just an uneven distribution of the S-Boxes as in HadesMiMC, these symmetric primitives are instantiated with two different round functions, one for the internal rounds and one for the external ones. We refer to Sect. 2 for more details about HadesMiMC and Hydra's body. At the current state of the art, Hydra is the symmetric primitive that offers the *best* performance in MPC applications/protocols (equivalently, the PRF with the smallest multiplicative complexity), improving e.g. upon Ciminion especially in the case in which the symmetric encryption key is shared among all participating parties – see [GØSW23] for more details.

The new PRF Pluto proposed in Sect. 6 takes inspiration of the body of Hydra, but the power maps in the external rounds are replaced by the SI-lifting function $\mathcal{S}_F$ over $\mathbb{F}_p^n$ induced by $F(x_0, x_1) = x_0^2 + x_1$. As we are going to show, this modified version achieves (much) better performances with respect to HadesMiMC in term of multiplicative complexity, especially in the case of large $n \gg 1$. In a similar way, when replacing the body of Hydra with the keyed PRF Pluto, the multiplicative complexity of the modified PRF Hydra++ is (slightly) reduced. Besides performance improvements, in Sect. 6.3 we also analyze the *security advantages* of instantiating Pluto and Hydra++ with the non-invertible non-linear $\mathcal{S}_F$ defined as before with respect to e.g. a similar invertible non-linear layer defined via a Type-III Feistel function [ZMI90, Nyb96].

1.2.2 Masta, Pasta, and `Rubato` for FHE Protocols

FHE protocols allow a user to operate on encrypted data without decrypting them. Differing from MPC applications, the cost metric in FHE applications is related to the multiplicative depth. The FHE-friendly PRFs Masta, Pasta, and `Rubato` minimize their multiplicative depth by adapting the same design strategy initially proposed for the FHE-friendly PRF Rasta, that is, (i) they are instantiated via new randomly generated affine layers for each new block to encrypt (for preventing statistical attacks), and (ii) their states have large size (for preventing linearization attacks without increasing the number of rounds, and so the depth). Their non-linear layers are instantiated via quadratic functions, including (i) the SI-lifting function $\mathcal{S}_\chi$ over $\mathbb{F}_p^n$ defined via the local quadratic chi-map $\chi : \mathbb{F}_p^3 \to \mathbb{F}_p$ introduced in [Wol85, DGV91] and adapted to the prime case, and (ii) the Type-III Feistel scheme instantiated via a quadratic map.

In Sect. 7, we show that it is potentially possible to increase the security and/or the performance of such schemes by replacing such non-linear layers with the quadratic SI-lifting function $\mathcal{S}_F$ over $\mathbb{F}_p^n$ induced by $F(x_0, x_1) = x_0^2 + x_1$.

2 Preliminary: MiMC, HadesMiMC, and Hydra for MPC Applications

In this section, we briefly recall some MPC-friendly schemes analyzed in the following.

Remark 1. For a fair comparison among all the MPC-friendly schemes, in the entire paper we assume a data limit of $2^{\kappa/2}$ texts available for the attacker (where κ is the security level in bits). We refer to [GRR+16] for more details regarding the *data limit* in MPC applications.

MiMC. MiMC [AGR+16] over $\mathbb{F}_p$ is an iterated Even-Mansour cipher, whose round function is defined as $F(x) = (x + \mathtt{K} + \gamma)^d$, where $\mathtt{K} \in \mathbb{F}_p$ is the secret master key, $\gamma \in \mathbb{F}_p$ is a random round constant and $d \geq 3$ is the smallest integer that satisfies $\gcd(d, p-1) = 1$ (in order to guarantee invertibility). A final key is added. For a security level of $\kappa \approx \log_2(p)$ bits with a data-limit of $2^{\kappa/2} \approx p^{1/2}$ texts available for the attack, the number of rounds is $R = 1 + \lceil(\kappa - 2 \cdot \log_2(\kappa)) \cdot \log_d(2)\rceil$.[3]

HadesMiMC. The Hades design strategy [GLR+20] allows to design SPN schemes over $\mathbb{F}_q^n$ that aim to reduce the overall multiplicative complexity. In order to guarantee security and maximize the efficiency:

- the external rounds at the beginning and at the end of the primitive are instantiated with full S-Box layers (that is, n S-Boxes in each non-linear layer) for ensuring security against statistical attacks, besides masking the internal rounds;
- the internal rounds instantiated with partial S-Box layers (that is, 1 S-Box and $n-1$ identity functions) aim to increase the overall degree of the scheme ensuring security against algebraic attacks, besides being cheaper to evaluate.

Let $p > 2^{63}$ (or equivalently, $\lceil\log_2(p)\rceil \geq 64$), and let $n \geq 2$. Let $\mathtt{K} \in \mathbb{F}_p^n$ be the secret master key. Let κ be the security level such that $2^{80} \leq 2^\kappa \leq \min\{p^2, 2^{256}\}$. Let $d \geq 3$ be the smallest integer such that $\gcd(d, p-1) = 1$. The block cipher HadesMiMC $\mathcal{H}_\mathtt{K} : \mathbb{F}_p^n \to \mathbb{F}_p^n$ is defined as[4]

$$\mathcal{H}_\mathtt{K}(x) = \underbrace{\mathcal{E}_{R_f+R'_f-1} \circ \cdots \circ \mathcal{E}_{R_f}}_{=R'_f \text{ rounds}} \circ \underbrace{\mathcal{I}_{R_P-1} \circ \cdots \circ \mathcal{I}_0}_{=R_P \text{ rounds}} \circ \underbrace{\mathcal{E}_{R_f-1} \circ \cdots \circ \mathcal{E}_0}_{=R_f \text{ rounds}}(x + \mathtt{K})$$

where

$$\begin{aligned} \forall i \in \{0, 1, \ldots, R_f + R'_f - 1\} : &\qquad \mathcal{E}_i(x) = k_i + M_\mathcal{E} \times \mathcal{S}_\mathcal{E}(x)\,, \\ \forall j \in \{0, 1, \ldots, R_\mathcal{I} - 1\} : &\qquad \mathcal{I}_j(x) = k_j + M_\mathcal{I} \times S_\mathcal{I}(x)\,, \end{aligned}$$

such that

- the external non-linear layer is defined as $\mathcal{S}_\mathcal{E}(x_0, x_1, \ldots, x_{n-1}) = x_0^d \| x_1^d \| \ldots \| x_{n-1}^d$;
- the internal non-linear layer is defined as $\mathcal{S}_\mathcal{I}(x_0, x_1, \ldots, x_{n-1}) = x_0^d \| x_1 \| \ldots \| x_{n-1}$ (that is, the power map is applied only to the first component);

[3]For completeness, we point out that MiMC in [AGR+16] is proposed only for $d = 3$, assuming $p = 2 \bmod 3$. Here, we simply generalized it for a generic $d \geq 3$ such that $\gcd(d, p-1) = 1$, by re-using the same argument proposed in [AGR+16] and by noting that currently no attack [ACG+19, EGL+20] applies on the full version of MiMC defined over a prime field.

[4]In [GLR+20], authors use the nomenclature "Full" and "Partial" rounds for referring respectively to the "External" and the "Internal" rounds. This new nomenclature has been introduced in [GOPS22].

- $M_\mathcal{E} = M_\mathcal{I} \in \mathbb{F}_p^{n\times n}$ is a MDS matrix that prevents the existence of invariant subspace trails for the internal rounds – we refer to [GRS21] for a detailed description on how to choose such matrices;
- $k_i, k_j \in \mathbb{F}_p^n$ are the round sub-keys derived from the master key via an affine key-schedule of the form

$$k_i = (M_\mathcal{K})^i \times \mathtt{K} + \varphi_i\,, \qquad k_j = (M'_\mathcal{K})^j \times \mathtt{K} + \varphi'_j \tag{2}$$

for invertible matrices $M_\mathcal{K}, M'_\mathcal{K} \in \mathbb{F}_p^{n\times n}$ and for random round constants $\varphi_i, \varphi'_j \in \mathbb{F}_p^n$ – we refer to [GLR+20, Sect. 3] for all details.

Let $2^{40} \le 2^{\kappa/2} \le \min\{p, 2^{128}\}$ be the data limit available for the attack. The number of rounds are given by $R_f = R'_f = 3$ and $R_P = 4 + \left\lceil \frac{\kappa}{2\cdot\log_2(d)} \right\rceil + \lceil \log_d(n) \rceil$.[5]

Hydra's Body. The PRF Hydra is based on the Megafono mode of operation recently introduced in [GØSW23], a modified version of the Farfalle mode of operation [BDH+17] suitable for MPC applications. A scheme based on the Megafono mode of operation is composed of two phases, that is, (1st) an initial phase in which the input is mixed with the secret key via a PRP, and (2nd) an expansion phase in which the state is expanded until the desired state size is reached. We refer to [GØSW23] for more details.

For the goal of this paper, we focus on the initial phase only. The primitive that instantiates the initial phase – called body – is an Even-Mansour construction of the form

$$x \mapsto \mathtt{K} + \mathcal{B}(x + \mathtt{K})\,, \tag{3}$$

where $\mathtt{K}$ is the secret key, and $\mathcal{B}$ is an *unkeyed* permutation. In the case in which *the body is indistinguishable from a PRP*, the security of the entire Megafono construction can be heavily simplified due to the fact that only few attacks apply, as a consequence of the facts that (i) the attacker does not have access to the internal states of the construction, as the inputs of the expansion phase (besides not being able to choose the outputs for e.g. setting up a chosen ciphertext attacks), and (ii) the inputs of the expansion phase (equivalently, the outputs of the initial phase) do not have any algebraic and statistical structure.

In the case of Hydra, the permutation $\mathcal{B}$ that instantiates the initial phase is based on the Hades design strategy, but differs from HadesMiMC on the following points:

- the body of Hydra is defined over $\mathbb{F}_p^4$ (that is, $n = 4$ fixed and not variable);
- the non-linear layer $\mathcal{S}_\mathcal{I}$ of the internal rounds are instantiated via a degree 4 Lai-Massey scheme that can be computed via only 2 $\mathbb{F}_p$-multiplications, that is, $\mathcal{S}_\mathcal{I}(x_0, x_1, x_2, x_3) = y_0\|y_1\|y_2\|y_3$ where

$$\forall j \in \{0,1,2,3\}: \qquad y_j = x_j + \left(\left(\sum_{i=0}^{3} \lambda_i^{(0)} \cdot x_i\right)^2 + \left(\sum_{i=0}^{3} \lambda_i^{(1)} \cdot x_i\right)\right)^2 \tag{4}$$

such that $(\lambda_0^{(0)}, \dots, \lambda_3^{(0)}), (\lambda_0^{(1)}, \dots, \lambda_3^{(1)}) \in (\mathbb{F}_p \setminus \{0\})^4$ are linearly independent and satisfy $\sum_{i=0}^3 \lambda_i^{(0)} = \sum_{i=0}^3 \lambda_i^{(1)} = 0$;
- $M_\mathcal{I} \in \mathbb{F}_p^{n\times n}$ is an invertible matrix that aims for destroying the invariant subspace trails of the Lai-Massey construction – we refer to [GRS21, GØSW23] for all details regarding how to construct/choose the matrix $M_\mathcal{I}$;

[5] In [GLR+20], the number of rounds are provided only for the cases in which either $p \approx 2^\kappa$ or $p^n \approx 2^\kappa$. The number of rounds given here is a simple generalization of the number of rounds given in the original paper [GLR+20].

- the round sub-keys are replaced by random round constants.

As in HadesMiMC, the external non-linear layer is defined as $\mathcal{S}_\mathcal{E}(x_0, x_1, \ldots, x_{n-1}) = x_0^d \| x_1^d \| \ldots \| x_{n-1}^d$, and $M_\mathcal{E} \in \mathbb{F}_p^{n\times n}$ is a MDS matrix.

The number of rounds of Hydra's body are given by $R_f = 2$, $R_{f'} = 4$, and $R_P = \lceil 1.125 \cdot \lceil \frac{\kappa}{4} + 6 - \log_2(d) \rceil \rceil$, including a security margin of 12.5% for the internal rounds.

3 "Bounded Surjective" Functions

In this section, we introduce the concept of bounded surjective functions. First, we recall the well-known definition of surjective/injective functions.

Definition 2 (Bijective)**.** Let $\mathtt{X}, \mathtt{Y}$ be two sets, and let $\mathcal{F} : \mathtt{X} \to \mathtt{Y}$ be a function. The function $\mathcal{F}$ is bijective if it is both surjective and injective, where (i) *surjective* means that for any element $y \in \mathtt{Y}$, there exists $x \in \mathtt{X}$ such that $\mathcal{F}(x) = y$, and (ii) *injective* means that $\mathcal{F}(x) = \mathcal{F}(x')$ implies $x = x'$ (for $x, x' \in \mathtt{X}$).

By the definition of a surjective function, for each $y \in \mathtt{Y}$, there exists *at least* one pre-image $x \in \mathtt{X}$ such that $\mathcal{F}(x) = y$. However, more pre-images can potentially exist. From a practical point of view, we are interested in the case in which each output admits a fixed maximum number of pre-images.

Definition 3 (l-Bounded Surjective)**.** Let $\mathtt{X}, \mathtt{Y}$ be two sets, and let $l \geq 1$ be an integer. A function $\mathcal{F} : \mathtt{X} \to \mathtt{Y}$ is "*l-bounded surjective*" if any element $y \in \mathtt{Y}$ admits at most l pre-images in $\mathtt{X}$, that is, if there exist *at most l distinct* elements $x_0, x_1, \ldots, x_{l-1} \in \mathtt{X}$ such that $\mathcal{F}(x_0) = \mathcal{F}(x_1) = \ldots = \mathcal{F}(x_{l-1}) = y$ (and $\mathcal{F}(z) \neq y$ for each $z \notin \{x_0, x_1, \ldots, x_{l-1}\}$).

Given such definition, we list some useful properties of l-bounded surjective functions, pointing out that a l-bounded surjective is *not* surjective in general.

Lemma 1. *(1st) Every function $\mathcal{F} : \mathtt{X} \to \mathtt{Y}$ is $|\mathtt{X}|$-bounded surjective (where $|\cdot|$ denotes the cardinality of the set). (2nd) Every bijective function is 1-bounded surjective. (3rd) If $|\mathtt{X}| = |\mathtt{Y}|$, then every 1-bounded surjective function $\mathcal{F} : \mathtt{X} \to \mathtt{Y}$ is also bijective.*

The proof follows immediately by the definition of l-bounded surjective function. We point out that the last point is false without the assumption $|\mathtt{X}| = |\mathtt{Y}|$ (e.g., $\mathcal{F} : \mathbb{Z}_q \to \mathbb{Z}_{2\cdot q}$ defined as $\mathcal{F}(x) = x$ is 1-bounded surjective but not bijective).

Next, we propose the following result regarding the composition of two bounded surjective functions.

Lemma 2. *Let $\mathtt{X}, \mathtt{Y}, \mathtt{Z}$ be three sets, and let $\mathcal{F} : \mathtt{X} \to \mathtt{Y}$ and $\mathcal{G} : \mathtt{Y} \to \mathtt{Z}$ be two functions. Assume that $\mathcal{F}$ is a f-bounded surjective function, and $\mathcal{G}$ is a g-bounded surjective function. Then, $\mathcal{H} := \mathcal{G} \circ \mathcal{F} : \mathtt{X} \to \mathtt{Z}$ is a $(f \cdot g)$-bounded surjective function.*

The proof follows immediately from the facts that (i) each output of $\mathcal{G}$ admits at most g pre-images, and (ii) each output of $\mathcal{F}$ admits at most f pre-images.

Finally, we evaluate the probability that a collision occurs (i.e., the collision probability) at the output of a l-bounded surjective function.

Lemma 3. *Let $\mathtt{X}, \mathtt{Y}$ be two sets, and let $l \geq 1$ be an integer. Let $\mathcal{F} : \mathtt{X} \to \mathtt{Y}$ be a l-bounded surjective function. The probability that a collision occurs at the output of $\mathcal{F}$ is at most $(l-1)/(|\mathtt{X}| - 1) \approx l/|\mathtt{X}|$.*

This follows from the fact that each output element admits at most l pre-images.

4 The PRF MiMC++: Reducing the Multiplicative Complexity of MiMC via the Square Map

MiMC is an iterated Even-Mansour cipher proposed at Asiacrypt 2016 for MPC [GRR+16] applications. Here, we show how to reduce its multiplicative complexity by instantiating it with the square map. We refer to this modified version of MiMC as `MiMC++`.

4.1 The PRF MiMC++

As we already mentioned in the introduction, MiMC's designers suggest to use it in a mode of operation in which the inverse is not needed. Hence, a natural question arises: if the inverse is not needed, why not implement it with a non-invertible function? In such a case, the most natural choice is the quadratic map $x \mapsto x^2$ over $\mathbb{F}_p$. Given $x \mapsto x^2$, a collision $x^2 = y^2$ can occur if and only if $x = \pm y$, which implies that the probability of having a collision is approximately p^{-1}.

Based on this simple observation, we propose the PRF `MiMC++` defined as follows. Let κ be the security level, and let p be *any* prime number such that $p \geq 2^{3\cdot\kappa}$.[6] Given a secret key $\mathtt{K} \in \mathbb{F}_p$, we define the PRF `MiMC++` as the iterated Even-Mansour scheme whose round function is define as by $F'(x) = (x + \mathtt{K} + \gamma)^2$, where $\gamma \in \mathbb{F}_p$ is a random round constant. Again, a final key is added. Since the PRF `MiMC++` is not invertible, it must be used in a mode in which the inverse is not needed, e.g., the Counter (CTR) one

$$(x, N) \mapsto (x + \mathtt{MiMC++}_{\mathtt{K}}(N), N) \tag{5}$$

where $N \in \mathbb{F}_p$ is a nonce. For a security level of κ bits and assuming a data-limit of $2^{\kappa/2}$ texts available for the attack, the number of rounds to provide security is given by

$$R_{\mathtt{MiMC++}} = 3 + \lceil \kappa - 2 \cdot \log_2(\kappa) \rceil .$$

4.2 Security Analysis for MiMC++

Here we justify the number of rounds $R_{\mathtt{MiMC++}}$ just given for `MiMC++`. Since the only differences between MiMC and `MiMC++` regard the invertibility and the degree of the power map $x \mapsto x^d$, we limit ourselves to adapt the security analysis of MiMC to `MiMC++`. In that sense, we only focus on the main attack vectors that are affected by these differences, including the differential one, the interpolation one, the GCD one, and the linearization one. As in the case of MiMC, all other attacks do not outperform the ones just listed (see [AGR+16] for more details). Moreover, we explicitly state that we do **not** make any security claim in the related-key setting.

4.2.1 Statistical Attacks

Differential Attack. Given pairs of inputs with some fixed input differences, differential cryptanalysis [BS90,BS93] considers the probability distribution of the corresponding output differences produced by the cryptographic primitive. Let $\Delta_I, \Delta_O \in \mathbb{F}_p^n$ be respectively the input and the output differences through a function $\mathcal{F}$ over $\mathbb{F}_p^n$. The differential probability (DP) of having a certain output difference Δ_O given a particular input difference Δ_I is equal to

$$\mathrm{Prob}_{\mathcal{F}}(\Delta_I \to \Delta_O) = \frac{|\{x \in \mathbb{F}_p^n \mid \mathcal{F}(x + \Delta_I) - \mathcal{F}(x) = \Delta_O\}|}{p^n} .$$

For any non-zero input and output differences $\Delta_I, \Delta_O \in \mathbb{F}_p \setminus \{0\}$, the equality $(x + \Delta_I)^2 - x^2 = \Delta_O$ admits a single solution, that is, $x = (\Delta_O - \Delta_I^2)/(2\Delta_O)$. Hence, the maximum

[6] Remember that the restriction $\gcd(d, p-1) = 1$ holds for guaranteeing the invertibility of MiMC.

differential probability for 1-round MiMC++ is $1/p$. As for MiMC, few rounds are sufficient for preventing differential attacks based on trails with non-zero differences.

However, since $x \mapsto x^2$ is not invertible, a collision can occur at every round. Based on Lemma 2, the polynomial function corresponding to the PRF MiMC++ is a $2^{R_{\texttt{MiMC++}}}$-bounded subject function. Given an uniformly randomly chosen pair of distinct input values, the overall probability that a collision occurs at the output of MiMC++ – given in Lemma 3 – is upper bounded by

$$\frac{2^{R_{\texttt{MiMC++}}}-1}{p-1} = \frac{2^{3+\lceil \kappa - 2\cdot\log_2(\kappa)\rceil}-1}{p-1} \approx \frac{8\cdot 2^{\kappa}}{p\cdot \kappa^2} \le \frac{8}{\kappa^2}\cdot 2^{-2\cdot\kappa} < 2^{-2\cdot\kappa},$$

where the inequalities follow from the facts that (i) $p \ge 2^{3\cdot\kappa}$ and (ii) $\kappa^2 > 8$. Since at most $2^{\kappa/2}$ texts are available for the attack, the attacker can construct at most $\binom{2^{\kappa/2}}{2} \approx 2^{\kappa-1}$ different pairs of texts, which implies that observing a collision is unrealistic for large κ.

Other Statistical Attacks. As in the case of MiMC, few rounds of MiMC++ are sufficient for preventing other statistical attacks such as the linear one [Mat93], the truncated differential one [Knu94], the impossible differential one [BBS99], the boomerang one [Wag99], and the invariant subspace attack [LAAZ11,LMR15], among others.

We limit ourselves to recall that $\mathbb{F}_p$ does not admit any non-trivial subspace. Moreover, we point out that the set $\texttt{R} = \{x^2 \in \mathbb{F}_p \mid x \in \mathbb{F}_p\}$ is *not* closed with respect to the addition. Indeed, let $x \ge 2$ be the smallest integer that is a quadratic non-residue modulo p (that is, $x \neq y^2$ for each $y \in \mathbb{F}_p$). Note that $0 = 0^2$ and $1 = (\pm 1)^2$ are always quadratic residue. By definition of x, $x-1$ is a quadratic residue, that is, there exists y such that $x - 1 = y^2$. Equivalently, $x = (\pm 1)^2 + y^2$ and $x \cdot z^2 = (\pm z)^2 + (z\cdot y)^2$ for each $z \in \mathbb{F}_p \setminus \{0\}$, where obviously $x \cdot z^2$ is a quadratic non-residue modulo p. Hence, the sum of two elements in R does not belong to such set in general.

4.2.2 Algebraic Attacks

Non-Invertibility: Local Inverse Functions with High Degree. In order to prevent MitM algebraic attacks, it is crucial to understand the algebraic expression/properties of MiMC++ in the backward direction. For this, we define the set $\texttt{R} = \{x^2 \in \mathbb{F}_p \mid x \in \mathbb{F}_p\}$ as before.

Since $x \mapsto F(x) := x^2$ is not invertible, it is only possible to define "local" inverses $F_+^{-1} : \texttt{R} \to \texttt{X}_+$ and $F_-^{-1} : \texttt{R} \to \texttt{X}_-$ such that the following points are satisfied:

1. the sets $\texttt{X}_-, \texttt{X}_+$ satisfy: $\texttt{X}_- \cap \texttt{X}_+ = \{0\}$ and $\texttt{X}_- \cup \texttt{X}_+ = \{0, 1, 2, \ldots, p-1\}$;
2. for each $x \in \texttt{X}_+ \setminus \{0\}$: $-x \in \texttt{X}_-$ and $-x \notin \texttt{X}_+$;
3. $F(F_+^{-1}(x)) = x$ and $F(F_-^{-1}(x)) = x$ for each $x \in \texttt{R}$.

Given $F_\pm^{-1}$ and $\texttt{X}_\pm$, other equivalent local inverses can be set up, (1st) by carefully swapping elements of $\texttt{X}_+$ and $\texttt{X}_-$, and (2nd) by adapting the definitions of F_+^{-1} and F_-^{-1}.

The algebraic representations of the functions F_+^{-1} and F_-^{-1} obviously depend on the sets $\texttt{X}_+$ and $\texttt{X}_-$. In general, we expect that the degrees of the functions F_+^{-1} and F_-^{-1} are of the same order as p, due to Fermat's little theorem. For example, if $p = 3 \mod 4$, then $F_\pm^{1/2}$ can be defined as $x \mapsto \pm x^{\frac{p+1}{4}}$ over certain sets $\texttt{X}_\pm$, since $\pm x^{\frac{p+1}{4}}$ are the square roots of the quadratic residue x.

To summarize, even if the overall construction is not invertible, we note that:

- the inverse functions can only be set up locally;
- in general, such local inverse functions have maximum degree after two rounds.

Interpolation Attack. The interpolation attack [JK97] aims to construct an interpolation polynomial that describes the scheme. Such polynomial can be exploited in order to set up a distinguisher and/or a forgery/key-recovery attack on the symmetric scheme. The interpolation polynomial cannot be constructed if the number of unknown monomials is larger than the data available for the attack.

The degree of `MiMC++` after $R_{\texttt{MiMC++}}$ rounds is $\min\{2^{R_{\texttt{MiMC++}}}, p-1\}$, and the number of monomials is upper bounded by $2^{R_{\texttt{MiMC++}}}+1$. Assuming that this upper bound is reasonably tight (note that the polynomial representation of `MiMC++` has the same density of the one of MiMC over $\mathbb{F}_p$),[7] and since the data limit for the attack is $2^{\kappa/2}$, then the scheme is secure against the interpolation polynomial if $2^{R_{\texttt{MiMC++}}} \geq 2^{\kappa/2}$, that is, $R_{\texttt{MiMC++}} \geq \kappa/2$. One more round is added for preventing key-guessing. Based on the previous argument, we finally also add two more rounds for preventing interpolation attacks that make used of the Meet-in-the-Middle (MitM) approach.

GCD Attack. A dedicated attack proposed for MiMC is the GCD attack. Let us denote by $E(k,x)$ the encryption of x under the key k. Given two inputs/outputs pairs (p_0,c_0) and (p_1,c_1), it is easy to check that the secret key is a zero of $\gcd(E(k,p_0)-c_0, E(k,p_1)-c_1)$, which has in general low degree. The cost of computing the GCD of two polynomials of degree (at most) d is $\mathcal{O}(d\cdot\log^2(d))$. The cost in our case is $\mathcal{O}(2^{R_{\texttt{MiMC++}}}\cdot\log^2(2^{R_{\texttt{MiMC++}}}))$, which implies that the number of rounds $R_{\texttt{MiMC++}}$ necessary for preventing such attack must satisfy

$$2^{R_{\texttt{MiMC++}}}\cdot\log^2(2^{R_{\texttt{MiMC++}}}) \geq 2^{\kappa} \qquad \longrightarrow \qquad R_{\texttt{MiMC++}} \geq \kappa - 2\cdot\log_2(\kappa)+1$$

(see also [AGR⁺16, Sect. 4.2] for more details). Due to the same argument given for the interpolation attack, we conjecture that two more rounds are sufficient for preventing the meet-in-the-middle version of the attack.

Linearization and Other Algebraic Attacks. Linearization [KS99] is a well-known technique to solve multivariate polynomial systems of equations. Given a system of polynomial equations, the idea is to turn it into a system of linear equations by adding new variables that replace all the monomials of the system whose degree is strictly greater than 1. This linear system of equations can be solved using linear algebra if the number of *linearly independent* equations is at least equal to (or bigger than) the number of variables after linearization. The most straightforward way to linearize algebraic expressions in t unknowns of degree limited by d is just by introducing a new variable for every monomial. As it is well known, the number of monomials in t variables of degree at most d is at most equal to

$$\#(d,t) \coloneqq \binom{t+d}{d}.$$

Based on this, the computation cost of such attacks is of $\mathcal{O}(\#(d,t)^{\omega})$ operations (for $2<\omega\leq 3$), besides a memory cost of $\mathcal{O}(\#(d,t)^2)$ for storing the linear equations.

In the case of `MiMC++`, assuming as before that the upper bound on the number of monomials is reasonably tight, the cost of the attack is given by $\binom{1+2^{R_{\texttt{MiMC++}}}}{2^{R_{\texttt{MiMC++}}}}^{\omega} \geq 2^{2\cdot R_{\texttt{MiMC++}}}$, which is similar to the one of the interpolation attack. Hence, security against the interpolation attack implies security against the linearization attack as well.

As in the case of MiMC, the same conclusion holds for other algebraic attacks, including the higher-order differential one [Lai94, Knu94, BCD⁺20] and the factorization attack (where the complexity of factorizing a polynomial over $\mathbb{F}_{p^t}$ of degree d is $\mathcal{O}(d^3\cdot t^2+d\cdot t^3)$ – see [Gen07] for details).

[7] We point out that the results recently proposed by Bouvier et al. [BCP23] do not apply to MiMC over prime fields, but only to the version of MiMC defined over binary fields $\mathbb{F}_{2^n}$ (see e.g. [BCP23, Footnote 1]).

4.3 Multiplicative Complexity: MiMC vs. MiMC++

Regarding the performances, the number of $\mathbb{F}_p$-multiplications required for evaluating the PRF MiMC++ is smaller than the corresponding one required for MiMC, even if MiMC++ requires a larger prime – approximately, triple size – with respect to the one used in MiMC for the same security level. (We emphasize again that the size of the prime has basically no impact on the MPC performances, as discussed in the introduction).

For comparing the performances of MiMC and of MiMC++, we first recall that evaluating $x \mapsto x^d$ costs $\lfloor \log_2(d) \rfloor + \text{hw}(d) - 1$ $\mathbb{F}_p$-multiplications, as shown e.g. in [GOPS22]. Based on this, the number of multiplications required for evaluating MiMC corresponds to

$$(1 + \lceil (\kappa - 2 \cdot \log_2(\kappa)) \cdot \log_d(2) \rceil) \cdot (\lfloor \log_2(d) \rfloor + \text{hw}(d) - 1) \, ,$$

which satisfies

$$\begin{aligned} &(1 + \lceil (\kappa - 2 \cdot \log_2(\kappa)) \cdot \log_d(2) \rceil) \cdot (\lfloor \log_2(d) \rfloor + \text{hw}(d) - 1) \\ \geq &(\lfloor \log_2(d) \rfloor + \text{hw}(d) - 1) + \lceil (\kappa - 2 \cdot \log_2(\kappa)) \cdot \underbrace{\log_d(2) \cdot (\lfloor \log_2(d) \rfloor + \text{hw}(d) - 1)}_{>1} \rceil \\ \geq &(\lfloor \log_2(d) \rfloor + \text{hw}(d) - 1) + \lceil \kappa - 2 \cdot \log_2(\kappa) \rceil \; \geq \; 2 + \lceil \kappa - 2 \cdot \log_2(\kappa) \rceil \, . \end{aligned}$$

In particular, note that $\log_d(2) \cdot (\lfloor \log_2(d) \rfloor + \text{hw}(d) - 1) > 1$ if and only if $\lfloor \log_2(d) \rfloor + \text{hw}(d) > 1 + \log_2(d)$. Since $\lfloor \log_2(d) \rfloor > \log_2(d) - 1$, this last inequality is satisfied if and only if $\text{hw}(d) \geq 2$, i.e., odd $d \geq 3$.

As a result, the number of multiplications required to evaluate MiMC is always bigger than or equal to $2 + \lceil \kappa - 2 \cdot \log_2(\kappa) \rceil$, which is almost the number of multiplications required for evaluating MiMC++, corresponding to $3 + \lceil \kappa - 2 \cdot \log_2(\kappa) \rceil$. As a concrete example, consider $\kappa = 128$: MiMC (with $p \approx 2^{128}$) requires 79 rounds and 158 $\mathbb{F}_p$-multiplications, while the PRF MiMC++ (with $p' \approx 2^{384}$) requires 126 rounds and 126 $\mathbb{F}_{p'}$-multiplications, that is, approximately 27.5% less multiplications.

Remark 2. Even if MiMC++ has a smaller multiplicative complexity than MiMC, it is not competitive with respect to other MPC-friendly schemes as HadesMiMC, Ciminion, or Hydra. As a result, MiMC++ is mainly of theoretical interest rather than of practical use.

5 Bounded-Surjective Quadratic SI-Lifting Functions over $\mathbb{F}_p^n$ via $F : \mathbb{F}_p^m \to \mathbb{F}_p$ for $m \in \{1, 2\}$

The main drawback of MiMC++ regards the fact that one is forced to work with a state size p that is much larger than the security level κ in order to guarantee security. Even if the performance in MPC applications does not depend on the size of the prime, it could be possible that some restrictions on the size/choice of p exist in certain applications. As we already mentioned in the introduction, this requirement/problem does not arise when working with a scheme that is defined over $\mathbb{F}_p^n$, as HadesMiMC and Hydra. In such a case, it is possible to guarantee security for a proper choice of n, keeping p fixed. For this reason, in this section we start an analysis of bounded surjective quadratic functions over $\mathbb{F}_p^n$ to use as possible building blocks for setting up variants of the MPC-friendly PRFs HadesMiMC and Hydra.

5.1 Our Results and Related Works

As it is well known, no quadratic function F over $\mathbb{F}_p$ is invertible, which (obviously) implies that no SI-lifting function $\mathcal{S}_F$ over $\mathbb{F}_p^n$ induced by $F(x) = x^2 + \alpha_1 \cdot x + \alpha_0$ can be invertible as well. Recently, at FSE/ToSC 2022, Grassi et al. [GOPS22] proved that, given any quadratic function $F : \mathbb{F}_p^2 \to \mathbb{F}_p$, the corresponding SI-lifting function $\mathcal{S}_F$ over $\mathbb{F}_p^n$ for $n \geq 3$ as defined in Def. 1 is never invertible.

For all such functions that can be computed via n multiplications only, here we analyze

- the probability that a collision occurs, namely, the probability that $\mathcal{S}_F(x) = \mathcal{S}_F(y)$ given $x, y \in \mathbb{F}_p^n$ such that $x \neq y$;
- the details of the inputs $x, y \in \mathbb{F}_p^n$ for which $\mathcal{S}_F(x) = \mathcal{S}_F(y)$.

Remark 3. Even if the analysis proposed in the following and the one proposed in [GOPS22] are very similar, we point out the following. In [GOPS22], authors proved that any SI-lifting function $\mathcal{S}_F$ over $\mathbb{F}_p^n$ for $n \geq 3$ induced by $F : \mathbb{F}_p^2 \to \mathbb{F}_p$ of degree 2 is not invertible by finding (at least) *one* collision for $\mathcal{S}_F$. In our current work, we aim to estimate the probability that a collision occurs, which requires to find *all* possible collisions of $\mathcal{S}_F$.

Motivation. While the motivation regarding the analysis of the probability that a collision occurs is clear, here we explain – via concrete examples – why we are also interested in the details of the inputs $x, y \in \mathbb{F}_p^n$ for which $\mathcal{S}_F(x) = \mathcal{S}_F(y)$.

First of all, consider a sponge hash function [BDPA08] instantiated via an iterative scheme, whose round function is of the form $x \mapsto \gamma + M \times \mathcal{S}(x)$, where $\mathcal{S} : \mathbb{F}_p^n \to \mathbb{F}_p^n$ is a non-linear layer instantiated by $\mathcal{S}_F$ as defined in Def. 1 for a certain $F : \mathbb{F}_p^m \to \mathbb{F}_p$, $M \in \mathbb{F}_p^{n \times n}$ is an invertible matrix, and $\gamma \in \mathbb{F}_p^n$ is a round constant. Let r, c be respectively the rate and the capacity of the sponge hash function, where $c + r = n$. Assume that $\mathcal{S}_F$ is not invertible, and assume there exist different $x, y \in \mathbb{F}_p^n$ such that (1st) $\mathcal{S}_F(x) = \mathcal{S}_F(y)$ and such that (2nd) $x_i = y_i$ for each $i \in \{r, r+1, \ldots, n-1\}$ (that is, x and y are equal in the part corresponding to the inner part of the sponge hash function). In such a case, since the attacker has full control of the inputs of the hash function and since no secret material is involved, a collision at the output of the sponge hash function can be obviously constructed, independently of the probability that a collision occurs for $\mathcal{S}_F$.[8]

Secondly, consider the case of a keyed SPN cipher, whose round function is of the form $x \mapsto \gamma + M \times \mathcal{S}_F(x)$ as before. Assume that $\mathcal{S}_F$ is not invertible, and assume that a collision $\mathcal{S}_F(x) = \mathcal{S}_F(y)$ for different $x \neq y$ cannot occur if some particular relation between x and y hold (e.g., $\mathcal{S}_F(x) = \mathcal{S}_F(y)$ only if $x_i \neq y_i$ for each $i \in \{0, 1, \ldots, n-1\}$). This fact can be exploited e.g. in an impossible differential attack for discarding (partially) wrongly guessed key candidates.

Our Results. As main results, we prove that:

1. given $F(x_0, x_1) = x_1^2 + \alpha_{1,0} \cdot x_0 + \alpha_{0,1} \cdot x_1$ with $\alpha_{0,1} \neq 0$ (equivalently, $F(x_0, x_1) = x_0^2 + \alpha_{1,0} \cdot x_1 + \alpha_{0,1} \cdot x_0$ with $\alpha_{1,0} \neq 0$), the probability that a collision occurs at the output of $\mathcal{S}_F$ is $\frac{(p-1)^n}{p^n \cdot (p^n - 1)} \leq p^{-n}$. In particular, we show that if a collision $\mathcal{S}_F(x) = \mathcal{S}_F(y)$ occurs, then $x_i \neq y_i$ for each $i \in \{0, 1, \ldots, n-1\}$. We emphasize that *such probability is much smaller than the upper bound obtained via Lemma 3*, which is only based on the fact that such function is 2^n-bounded surjective;
2. given any other quadratic function $F : \mathbb{F}_p^m \to \mathbb{F}_p$ for $m \in \{1, 2\}$ such that $\mathcal{S}_F$ can be computed via n multiplications *independently of the value of* p, the probability that a collision occurs in $\mathcal{S}_F$ is never smaller than the one corresponding to $F(x_0, x_1) = x_1^2 + \alpha_{1,0} \cdot x_0 + \alpha_{0,1} \cdot x_1$ as before (or equivalent);
3. the SI-lifting function $\mathcal{S}_F$ over $\mathbb{F}_p^n$ induced by $F(x_0, x_1) = x_1^2 + \alpha_{1,0} \cdot x_0 + \alpha_{0,1} \cdot x_1$ as before (or equivalent) is 2^n-bounded surjective.

[8]For completeness, we limit ourselves to mention that, depending on the details of M, a similar result can be achieved even if the linear layer M is applied before the non-linear $\mathcal{S}_F$.

In particular, we emphasize that both the SI-lifting functions $\mathcal{S}_F$ over $\mathbb{F}_p^n$ induced (i) by $F(x_0, x_1) = x_1^2 + \alpha_{1,0} \cdot x_0 + \alpha_{0,1} \cdot x_1$ with $\alpha_{0,1} \neq 0$ (or equivalent) and (ii) by $F(x) = x^2 + \alpha_1 \cdot x$ are 2^n-bounded surjective. However, the probability that a collision occurs in the first case is much smaller than in the second one, approximately by a factor $2^n - 1$. Moreover, a collision $x_0^2 \| x_1^2 \| \ldots \| x_{n-1}^2 = y_0^2 \| y_1^2 \| \ldots \| y_{n-1}^2$ can occur also in the case in which $x_i = y_i$ for some $i \in \{0, 1, \ldots, n-1\}$, while this is not possible for $\mathcal{S}_F$ induced by $F(x_0, x_1) = x_1^2 + \alpha_{1,0} \cdot x_0 + \alpha_{0,1} \cdot x_1$ with $\alpha_{0,1} \neq 0$ (or equivalent).

Organization of the Section. In the following, we propose a complete analysis of the following cases:

$$\begin{aligned} F(x_0, x_1) &= x_1^2 + \alpha_{1,0} \cdot x_0 + \alpha_{0,1} \cdot x_1 \, , \\ F(x) &= x^2 + \alpha_1 \cdot x \, , \\ F(x_0, x_1) &= x_0 \cdot x_1 + \alpha_{1,0} \cdot x_0 + \alpha_{0,1} \cdot x_1 \, . \end{aligned}$$

The analogous analysis of the other cases (including $F(x_0, x_1) = x_0^2 + \alpha_{0,2} \cdot x_1^2 + \alpha_{1,0} \cdot x_0 + \alpha_{0,1} \cdot x_1$, $F(x_0, x_1) = x_0 \cdot x_1 + \alpha_{2,0} \cdot x_0^2 + \alpha_{1,0} \cdot x_0 + \alpha_{0,1} \cdot x_1$, and $F(x_0, x_1) = \alpha_{2,0} \cdot x_0^2 + x_0 \cdot x_1 + \alpha_{0,2} \cdot x_1^2 + \alpha_{1,0} \cdot x_0 + \alpha_{0,1} \cdot x_1$) is given in App. A.

5.2 $F(x_0, x_1) = x_1^2 + \alpha_{1,0} \cdot x_0 + \alpha_{0,1} \cdot x_1$

We start by analyzing the case $F(x_0, x_1) = \alpha_{0,2} \cdot x_1^2 + \alpha_{1,0} \cdot x_0 + \alpha_{0,1} \cdot x_1$ with $\alpha_{0,2}, \alpha_{1,0} \neq 0$ (equivalently, $F(x_0, x_1) = \alpha_{2,0} \cdot x_0^2 + \alpha_{1,0} \cdot x_0 + \alpha_{0,1} \cdot x_1$ with $\alpha_{2,0}, \alpha_{0,1} \neq 0$). Without loss of generality (W.l.o.g.), we assume $\alpha_{0,2} = 1$. Indeed, note that $\mathcal{S}_F$ over $\mathbb{F}_p^n$ induced by $F(x_0, x_1) = \alpha_{0,2} \cdot x_1^2 + \alpha_{1,0} \cdot x_0 + \alpha_{0,1} \cdot x_1$ is equivalent to $\alpha_{0,2} \cdot \mathcal{S}_{F'}$ induced by $F'(x_0, x_1) = x_1^2 + \alpha'_{1,0} \cdot x_0 + \alpha'_{0,1} \cdot x_1$ where $\alpha'_{1,0} = \alpha_{1,0}/\alpha_{0,2}$ and $\alpha'_{0,1} = \alpha_{0,1}/\alpha_{0,2}$.

The collision $\mathcal{S}_F(x) = \mathcal{S}_F(y)$ occurs if and only if

$$\begin{aligned} \forall i \in \{0, \ldots, n-1\} : \qquad & x_{i+1}^2 + \alpha_{1,0} \cdot x_i + \alpha_{0,1} \cdot x_{i+1} = y_{i+1}^2 + \alpha_{1,0} \cdot y_i + \alpha_{0,1} \cdot y_{i+1} \\ \longleftrightarrow \qquad & x_{i+1}^2 - y_{i+1}^2 = -\alpha_{1,0} \cdot x_i - \alpha_{0,1} \cdot x_{i+1} + \alpha_{1,0} \cdot y_i + \alpha_{0,1} \cdot y_{i+1} \\ \longleftrightarrow \qquad & (x_{i+1} - y_{i+1}) \cdot (x_{i+1} + y_{i+1}) = -\alpha_{1,0} \cdot (x_i - y_i) - \alpha_{0,1} \cdot (x_{i+1} - y_{i+1}) \, . \end{aligned}$$

Via the change of variables

$$d_i \coloneqq x_i - y_i \qquad \text{and} \qquad s_i \coloneqq x_i + y_i, \tag{6}$$

where $x_i = (s_i + d_i)/2$ and $y_i = (s_i - d_i)/2$, the collision $\mathcal{S}_F(x) = \mathcal{S}_F(y)$ occurs if and only if

$$\begin{bmatrix} 0 & d_1 & 0 & \ldots & 0 \\ 0 & 0 & d_2 & \ldots & 0 \\ \vdots & & & \ddots & \vdots \\ 0 & 0 & 0 & \ldots & d_{n-1} \\ d_0 & 0 & 0 & \ldots & 0 \end{bmatrix} \times \begin{bmatrix} s_0 \\ s_1 \\ s_2 \\ \vdots \\ s_{n-1} \end{bmatrix} = - \begin{bmatrix} \alpha_{1,0} \cdot d_0 + \alpha_{0,1} \cdot d_1 \\ \alpha_{1,0} \cdot d_1 + \alpha_{0,1} \cdot d_2 \\ \alpha_{1,0} \cdot d_2 + \alpha_{0,1} \cdot d_3 \\ \vdots \\ \alpha_{1,0} \cdot d_{n-1} + \alpha_{0,1} \cdot d_0 \end{bmatrix} . \tag{7}$$

The determinant of the left-hand side (l.h.s.) matrix is $-(-1)^n \cdot \prod_{i=0}^{n-1} d_i$:

- if $d_i \neq 0$ for all $i \in \{0, 1, \ldots, n-1\}$, then the system admits a solution for each given $s_0, s_1, \ldots, s_{n-1}$, which corresponds to a collision;
- if $d_i = 0$ for a certain $i \in \{0, 1, \ldots, n-1\}$, e.g. $d_1 = 0$, then the condition $d_1 \cdot s_0 = -(\alpha_{1,0} \cdot d_0 + \alpha_{0,1} \cdot d_1)$ is satisfied only by $d_0 = 0$. Working iteratively, we get that if at least one d_i is zero, then the system admits a solution if and only if all d_i are zero, which corresponds to $x = y$.

It follows that

Proposition 1. *Let $p \geq 3$ be a prime and let $n \geq 2$. Let $F : \mathbb{F}_p^2 \to \mathbb{F}_p$ be defined as $F(x_0, x_1) = x_1^2 + \alpha_{1,0} \cdot x_0 + \alpha_{0,1} \cdot x_1$ where $\alpha_{1,0} \neq 0$. Let $\mathcal{S}_F$ over $\mathbb{F}_p^n$ be defined as in Def. 1. The probability of a collision $\mathcal{S}_F(x) = \mathcal{S}_F(y)$ for $x, y \in \mathbb{F}_p^n$ such that $x \neq y$ is*

$$\frac{(p-1)^n}{p^n \cdot (p^n - 1)} < \frac{p^n - 1}{p^n \cdot (p^n - 1)} = p^{-n} .$$

*Moreover, if $x, y \in \mathbb{F}_p^n$ such that $x \neq y$ and $\mathcal{S}_F(x) = \mathcal{S}_F(y)$, then $x_i \neq y_i$ **for each** $i \in \{0, 1, \ldots, n-1\}$.*

(Note that, for any $p \geq 3$ and any $n \geq 2$, $p^n - 1 > (p-1)^n$ since $\sum_{i=1}^{n-1} \binom{n}{i} \cdot p^i > 0$.)

The following Lemma provides the details of the collisions:

Lemma 4. *Let $p \geq 3$ be a prime and let $n \geq 2$. Let $F : \mathbb{F}_p^2 \to \mathbb{F}_p$ be defined as $F(x_0, x_1) = x_1^2 + \alpha_{1,0} \cdot x_0 + \alpha_{0,1} \cdot x_1$ where $\alpha_{1,0} \neq 0$. Let $\mathcal{S}_F$ over $\mathbb{F}_p^n$ be defined as in Def. 1. Two distinct inputs $x, y \in \mathbb{F}_p^n$ form a collision $\mathcal{S}_F(x) = \mathcal{S}_F(y)$ if and only*

$$x_i = \frac{\alpha_{0,1}}{2} + \frac{d_i}{2} \cdot \left(\frac{\alpha_{1,0}}{d_{i+1}} + 1 \right) \qquad \text{and} \qquad y_i = \frac{\alpha_{0,1}}{2} + \frac{d_i}{2} \cdot \left(\frac{\alpha_{1,0}}{d_{i+1}} - 1 \right) = x_i + d_i ,$$

for each $i \in \{0, 1, \ldots, n-1\}$, where $d_0, d_1, \ldots, d_{n-1} \in \mathbb{F}_p^n \setminus \{0\}$.

In order to prove the result, it is sufficient to invert the l.h.s. diagonal matrix given in (7), and to make used of the definition of d, s given in (6). Given a difference $d \in \mathbb{F}_p^n$ and a sum $s \in \mathbb{F}_p^n$ that correspond to a collision, that is, $\mathcal{S}_F((s+d)/2) = \mathcal{S}_F((s-d)/2)$, we point out that $\mathcal{S}_F((s + \omega \cdot d)/2) = \mathcal{S}_F((s - \omega \cdot d)/2)$ for each $\omega \in \mathbb{F}_p$.

The Function $\mathcal{S}_F$ over $\mathbb{F}_p^n$ via $F(x_0, x_1) = x_1^2 + \alpha_{1,0} \cdot x_0 + \alpha_{0,1} \cdot x_1$ is 2^n-Bounded Surjective. As final step, we prove the following result.

Proposition 2. *Let $p \geq 3$ be a prime and let $n \geq 2$. Let $F : \mathbb{F}_p^2 \to \mathbb{F}_p$ be defined as $F(x_0, x_1) = x_1^2 + \alpha_{1,0} \cdot x_0 + \alpha_{0,1} \cdot x_1$ where $\alpha_{1,0} \neq 0$. The SI-lifting function $\mathcal{S}_F$ over $\mathbb{F}_p^n$ defined as in Def. 1 is 2^n-bounded surjective.*

Proof. By definition of 2^n-bounded surjective, we aim to prove that each output y of $\mathcal{S}_F$ admits at most 2^n pre-images. W.l.o.g., we focus on $\alpha_{1,0} = 0$ and $\alpha_{0,1} = 1$ (the following proof is equivalent for the other cases). By definition of F:

$$\forall i \in \{0, 1, \ldots, n-1\} : \qquad y_i = x_i + x_{i+1}^2 \qquad \longrightarrow \qquad x_i = G_{y_i}(x_{i+1}) := y_i - x_{i+1}^2 , \tag{8}$$

for $G_y : \mathbb{F}_p \to \mathbb{F}_p$ given $y \in \mathbb{F}_p$. Working iteratively, we have that

$$x_0 = G_{y_0}(x_1) = G_{y_0} \circ G_{y_1}(x_2) = \ldots = G_{y_0} \circ G_{y_1} \circ \ldots \circ G_{y_{n-1}}(x_0) .$$

That is, given $y_0, y_1, \ldots, y_{n-1} \in \mathbb{F}_p$, there exists a function $H_{y_0, y_1, \ldots, y_{n-1}}$ over $\mathbb{F}_p$ of degree 2^n such that

$$H_{y_0, y_1, \ldots, y_{n-1}}(x_0) := G_{y_0} \circ G_{y_1} \circ \ldots \circ G_{y_{n-1}}(x_0) - x_0 = 0 . \tag{9}$$

Such function admits at most 2^n distinct solutions in $x_0 \in \mathbb{F}_p$. For each one of the 2^n solutions x_0, the values $x_1, x_2, \ldots, x_{n-1}$ are fixed and defined iteratively by $x_i = G_{y_i}(x_{i+1})$ for each $i \in \{1, 2, \ldots, n-1\}$. This means that each output of $\mathcal{S}_F$ admits at most 2^n distinct pre-images. $\square$

As a concrete example, consider the case $p = 7$ and $n = 2$. The function $\mathcal{S}_F(x_0, x_1) = x_0^2 + x_1 \| x_1^2 + x_0$ over $\mathbb{F}_p^2$ is "strictly" 4-bounded surjective, since there exist outputs in $\mathbb{F}_p^2$ with four distinct pre-images, as $\mathcal{S}_F(0,0) = \mathcal{S}_F(3,5) = \mathcal{S}_F(5,3) = \mathcal{S}_F(6,6) = (0,0)$.

Before going on, we point out that the collision probability given in Prop. 1 is (much) smaller than the corresponding probability obtained by combining Lemma 3 with the fact that $\mathcal{S}_F$ is 2^n-bounded surjective, that is,

$$\underbrace{\frac{(p-1)^n}{p^n \cdot (p^n - 1)}}_{\leq p^{-n}} \ll \underbrace{\frac{2^n - 1}{p^n - 1}}_{\approx 2^n \cdot p^{-n}} .$$

5.3 $F(x) = x^2 + \alpha_1 \cdot x$

Next, we compare the result just obtained with the collision probability of the SI-lifting function $\mathcal{S}_F$ over $\mathbb{F}_p^n$ induced by $F(x) = \alpha_2 \cdot x^2 + \alpha_1 \cdot x$ for $\alpha_2, \alpha_1 \in \mathbb{F}_p$. W.l.o.g., we assume as before $\alpha_2 = 1$. The collision $\mathcal{S}_F(x) = \mathcal{S}_F(y)$ occurs if and only if

$$\forall i \in \{0, \ldots, n-1\} : \qquad (x_{i+1} - y_{i+1}) \cdot (x_{i+1} + y_{i+1} + \alpha_1) = 0 \quad \rightarrow \quad d_i \cdot (s_i + \alpha_1) = 0 ,$$

via the change of variables $d_i \coloneqq x_i - y_i$ and $s_i \coloneqq x_i + y_i$ given in (6). Obviously, the i-th equation is satisfied if and only if (i) $d_i = 0$ and/or (ii) $s_i = -\alpha_1$, for a total of $2 \cdot p - 1$ possible values (d_i, s_i) for each i. Hence, the following result holds (note that the term $-p^n$ is due to the case $d_0 = d_1 = \ldots = d_{n-1} = 0$).

Lemma 5. *Let $p \geq 3$ be a prime and let $n \geq 2$. Let $F : \mathbb{F}_p \to \mathbb{F}_p$ be defined as $F(x) = x^2 + \alpha_1 \cdot x$. Let $\mathcal{S}_F$ over $\mathbb{F}_p^n$ be defined as in Def. 1. The probability of a collision $\mathcal{S}_F(x) = \mathcal{S}_F(y)$ for $x, y \in \mathbb{F}_p^n$ such that $x \neq y$ is*

$$\frac{(2 \cdot p - 1)^n - p^n}{p^n \cdot (p^n - 1)} \approx \frac{2^n - 1}{p^n - 1} ,$$

where the approximation holds for large $p \gg 1$. Moreover, the function $\mathcal{S}_F$ over $\mathbb{F}_p^n$ induced by $F(x) = x^2 + \alpha_1 \cdot x$ is 2^n-bounded surjective.

Note that $\mathcal{S}_F(x_0, x_1, \ldots, x_{n-1})$ is 2^n-bounded surjective since (i) the n components of $\mathcal{S}_F$ are *independent*, and (ii) the function $x \mapsto x^2$ is 2-bounded surjective.

5.4 $F(x_0, x_1) = x_0 \cdot x_1 + \alpha_{1,0} \cdot x_0 + \alpha_{0,1} \cdot x_1$

Next, we analyze $\mathcal{S}_F$ induced by $F(x_0, x_1) = \alpha_{1,1} \cdot x_0 \cdot x_1 + \alpha_{1,0} \cdot x_0 + \alpha_{0,1} \cdot x_1$. W.l.o.g., we fix $\alpha_{1,1} = 2$ (this allows for a simpler description when using the variables s_i and d_i). Given $F(x_0, x_1) = 2 \cdot x_0 \cdot x_1 + \alpha_{1,0} \cdot x_0 + \alpha_{0,1} \cdot x_1$, the system of equations that defines the collision $\mathcal{S}_F(x) = \mathcal{S}_F(y)$ via the variables $d_i \coloneqq x_i - y_i$ and $s_i \coloneqq x_i + y_i$ introduced in Eq. (6) is

$$\begin{bmatrix} d_1 & d_0 & 0 & 0 & \ldots & 0 \\ 0 & d_2 & d_1 & 0 & \ldots & 0 \\ 0 & 0 & d_3 & d_2 & \ldots & 0 \\ \vdots & & & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \ldots & d_{n-1} & d_{n-2} \\ d_{n-1} & 0 & 0 & \ldots & 0 & d_0 \end{bmatrix} \times \begin{bmatrix} s_0 \\ s_1 \\ s_2 \\ \vdots \\ s_{n-2} \\ s_{n-1} \end{bmatrix} = - \begin{bmatrix} \alpha_{1,0} \cdot d_0 + \alpha_{0,1} \cdot d_1 \\ \alpha_{1,0} \cdot d_1 + \alpha_{0,1} \cdot d_2 \\ \alpha_{1,0} \cdot d_2 + \alpha_{0,1} \cdot d_3 \\ \vdots \\ \alpha_{1,0} \cdot d_{n-2} + \alpha_{0,1} \cdot d_{n-1} \\ \alpha_{1,0} \cdot d_{n-1} + \alpha_{0,1} \cdot d_0 \end{bmatrix} . \tag{10}$$

The determinant of the l.h.s. matrix is

$$(1 - (-1)^n) \cdot \prod_{i=0}^{n-1} d_i = \begin{cases} 2 \cdot \prod_{i=0}^{n-1} d_i & \text{if } n \text{ odd,} \\ 0 & \text{otherwise (if } n \text{ even).} \end{cases}$$

As we are going to show:

1. the probability that a collision occurs is strictly higher than $\frac{(p-1)^n}{p^n\cdot(p^n-1)}$, which corresponds to the probability of having a collision for $F(x_0, x_1) = x_1^2 + x_0$ (and equivalent functions) as given in Prop. 1;

2. a collision can occur also in the case in which $n-1$ input differences d_i are equal to zero.

Analysis of $x, y \in \mathbb{F}_p^n$ such that $\mathcal{S}_F(x) = \mathcal{S}_F(y)$. About this second point, consider the case $d_i \in \mathbb{F}_p \setminus \{0\}$ and $d_j = 0$ for each $j \neq i$, for which the system of equation reduces to

$$d_i \cdot s_{i-1} = -\alpha_{0,1} \cdot d_i \qquad \text{and} \qquad d_i \cdot s_{i+1} = -\alpha_{1,0} \cdot d_i \,.$$

The solution of it corresponds to $s_{i-1} = -\alpha_{0,1}$ and $s_{i+2} = -\alpha_{1,0}$ (no conditions on the others s_l for $l \notin \{0, 2\}$).

Collision Probability for n odd. First of all, note that if $d_i \neq 0$ for each $i \in \{0, 1, \ldots, n-1\}$, then a collision can occur. Indeed, the determinant is different from zero, which means that there exist $s_0, s_1, \ldots, s_{n-1}$ that satisfy the required condition for having a collision.

Consider the case in which $n-1$ differences d_i are equal to zero (note that there are n different cases). This case is obviously not included in the previous one, since now the determinant is equal to zero. As pointed out in the previous paragraph, a collision can occur if s_{i-1} and s_{i+1} satisfy some particular equalities, while no condition is imposed on the other s_j. As a result, the probability of having a collision is *at least* equal

$$\frac{(p-1)^n + n \cdot p^{n-2} \cdot (p-1)}{p^n \cdot (p^n - 1)} > \frac{(p-1)^n}{p^n \cdot (p^n-1)},$$

which is strictly bigger than the probability given in Prop. 1.

Collision Probability for n even. Since the determinant of the matrix is always equal to zero, there is a linear combination of its rows that is equal to zero. Assuming such linear combination is defined via $\lambda_0, \lambda_1, \ldots, \lambda_{n-1} \in \mathbb{F}_p$, a collision can occur if the right-hand side (r.h.s.) of (10) satisfies the same linear relation, that is, if $\sum_{i=0}^{n-1} \lambda_i \cdot (\alpha_{1,0} \cdot d_i + \alpha_{0,1} \cdot d_{i+1}) = 0$. In such a case, this implies that one difference d_i is fixed. W.l.o.g., assuming that d_1 satisfies such linear relation, the collision takes place if

$$\begin{bmatrix} d_2 & d_1 & 0 & \ldots & 0 \\ 0 & d_3 & d_2 & \ldots & 0 \\ \vdots & & \ddots & \ddots & \vdots \\ 0 & 0 & \ldots & d_{n-1} & d_{n-2} \\ 0 & 0 & \ldots & 0 & d_0 \end{bmatrix} \times \begin{bmatrix} s_1 \\ s_2 \\ \vdots \\ s_{n-2} \\ s_{n-1} \end{bmatrix} = - \begin{bmatrix} \alpha_{1,0} \cdot d_1 + \alpha_{0,1} \cdot d_2 \\ \alpha_{1,0} \cdot d_2 + \alpha_{0,1} \cdot d_3 \\ \vdots \\ \alpha_{1,0} \cdot d_{n-2} + \alpha_{0,1} \cdot d_{n-1} \\ (\alpha_{1,0} + s_0) \cdot d_{n-1} + \alpha_{0,1} \cdot d_0 \end{bmatrix},$$

where d_1 is fixed and where no condition holds on s_0. The determinant of the l.h.s. matrix is equal to $d_0 \cdot \prod_{i=2}^{n-1} d_i$. As before, a collision can occur if $d_0, d_2, d_3, \ldots, d_{n-1} \neq 0$, since in such a case the determinant of the matrix is different from zero. This is sufficient for concluding that the probability of having a collision is *at least* equal to

$$\frac{p \cdot (p-1)^{n-1}}{p^n \cdot (p^n - 1)} > \frac{(p-1)^n}{p^n \cdot (p^n - 1)}\,,$$

which is strictly bigger than the probability given in Prop. 1.

6 The MPC-Friendly PRFs Pluto and Hydra++

In this section, we propose the PRF PLUTO,[9] a modified version of HYDRA's body over $\mathbb{F}_p^n$ in which the external rounds are instantiated via the quadratic SI-lifting function $\mathcal{S}_F$ over $\mathbb{F}_p^n$ proposed in the previous section. HYDRA++ is simply defined as the PRF HYDRA whose body is replaced with PLUTO. As we are going to show, PLUTO and HYDRA++ improve respectively HADESMiMC and HYDRA from the multiplicative complexity point of view (for the same security level and for the same size of the prime p). Besides that, we will discuss some concrete security advantages of instantiating PLUTO and HYDRA++ with a non-invertible non-linear with respect to e.g. a similar invertible non-linear layer defined via a Type-III Feistel.

6.1 The PRFs Pluto and Hydra++

The PRF Pluto. Let $p > 2^{63}$ (or equivalently, $\lceil \log_2(p) \rceil \geq 64$), and let $n \geq 4$. Let $\mathtt{K} \in \mathbb{F}_p^n$ be the secret master key. Let κ be the security level such that

$$2^{80} \leq 2^{\kappa} \leq \min\left\{p^2, 2^{256}, \frac{1}{2} \cdot \left(\frac{p}{2^8}\right)^{n/2}\right\} .$$

The keyed PRF PLUTO $\mathcal{P}_{\mathtt{K}} : \mathbb{F}_p^n \to \mathbb{F}_p^n$ is defined as

$$\mathcal{P}_{\mathtt{K}}(x) = \underbrace{\mathcal{E}_{R_{\mathcal{E}}+R_{\mathcal{E}'}-1} \circ \cdots \circ \mathcal{E}_{R_{\mathcal{E}}}}_{=R_{\mathcal{E}'} \text{ rounds}} \circ \underbrace{\mathcal{I}_{R_{\mathcal{I}}-1} \circ \cdots \circ \mathcal{I}_0}_{=R_{\mathcal{I}} \text{ rounds}} \circ \underbrace{\mathcal{E}_{R_{\mathcal{E}}-1} \circ \cdots \circ \mathcal{E}_0}_{=R_{\mathcal{E}} \text{ rounds}}(x + \mathtt{K})$$

where

$$\begin{aligned} \forall i \in \{0, 1, \dots, R_{\mathcal{E}} + R_{\mathcal{E}'} - 1\} : \qquad & \mathcal{E}_i(x) = k_i + M_{\mathcal{E}} \times \mathcal{S}_{\mathcal{E}}(x) , \\ \forall j \in \{0, 1, \dots, R_{\mathcal{I}} - 1\} : \qquad & \mathcal{I}_j(x) = k_j + M_{\mathcal{I}} \times S_{\mathcal{I}}(x) , \end{aligned}$$

such that

- the external non-linear layer $\mathcal{S}_{\mathcal{E}}$ is instantiated by the SI-lifting function $\mathcal{S}_F$ over $\mathbb{F}_p^n$ induced by $F(x_0, x_1) = x_0^2 + x_1$;
- similar to HYDRA's body, the internal non-linear layer $\mathcal{S}_{\mathcal{I}}$ of degree 4 is instantiated via the Lai-Massey function defined in (4) generalized over $\mathbb{F}_p^n$, that is, $\mathcal{S}_{\mathcal{I}}(x_0, x_1, \dots, x_{n-1}) = x_0 + z \| x_1 + z \| \dots \| x_{n-1} + z$ where

  $$z = \left(\left(\sum_{i=0}^{n-1} \lambda_i^{(0)} \cdot x_i\right)^2 + \left(\sum_{i=0}^{n-1} \lambda_i^{(1)} \cdot x_i\right)\right)^2$$

  such that $(\lambda_0^{(0)}, \dots, \lambda_{n-1}^{(0)}), (\lambda_0^{(1)}, \dots, \lambda_{n-1}^{(1)}) \in (\mathbb{F}_p \setminus \{0\})^n$ are linearly independent and satisfy $\sum_{i=0}^{n-1} \lambda_i^{(0)} = \sum_{i=0}^{n-1} \lambda_i^{(1)} = 0$;
- similar to HYDRA's body, $M_{\mathcal{E}} \in \mathbb{F}_p^{n \times n}$ is a MDS matrix, while $M_{\mathcal{I}} \in \mathbb{F}_p^{n \times n}$ is an invertible matrix that aims at destroying the invariant subspace trails of the Lai-Massey construction. We refer to [GRS21, GØSW23] for all details on how choosing/constructing the matrix $M_{\mathcal{I}}$;
- k_i, k_j are the round sub-keys defined as $k_i = \mathtt{K} + \varphi_i$ and $k_j = \mathtt{K} + \varphi'_j$ for random round constants $\varphi_i, \varphi'_j \in \mathbb{F}_p^n$.

[9] In [GOPS22], the updated modified version of POSEIDON is called NEPTUNE. Following such approach, here we decided to call this new version as PLUTO, which is the Roman name of the Greek god Hades.

Obviously, this construction is not invertible anymore, but it can be used as a stream cipher for encryption purpose, exactly as in the case of `MiMC++` – see (5). The number of rounds are given by

$$R_{\mathcal{E}} = R_{\mathcal{E}'} = 4 \qquad \text{and} \qquad R_{\mathcal{I}} = \left\lceil 1.125 \cdot \left\lceil \frac{\kappa}{4} + \frac{n}{2} + \log_2(n) + 1 \right\rceil \right\rceil ,$$

where we add an arbitrary security margin of 12.5% for the internal rounds, which corresponds to the same security margin for the internal/partial rounds of Poseidon and Hydra.

The PRF Hydra++. The keyed PRF Hydra++ is defined as the PRF Hydra whose body is replaced with the keyed PRF Pluto just defined over $\mathbb{F}_p^4$.

Remark 4. We point out that the body of the PRF Hydra is instantiated via an Even-Mansour construction $x \mapsto \mathtt{K} + \mathcal{B}(x + \mathtt{K})$, where $\mathcal{B}$ is a permutation that is independent of the secret key. In the case of Hydra++, its body is instantiated with a keyed iterated PRF in which the key addition takes place in each round. We are not aware of any attack on Hydra++ that exploits such difference.

6.2 Security Analysis of Pluto

Here we justify the number of rounds just given for Pluto (and Hydra++). The main differences between Pluto and Hydra's body regard (i) the fact that Pluto is defined over $\mathbb{F}_p^n$ for any $n \geq 4$, while Hydra's body is defined only over $\mathbb{F}_p^4$, and (ii) the invertibility and the degree of the non-linear layer of the external rounds. For this reason, and as already done for `MiMC++` before, here we limit ourselves to adapt the security analysis of Hydra's body (which resembles the one of HadesMiMC) to Pluto, focusing only on the main attack vectors that are affected by these differences, including the differential one, the interpolation one, the linearization one, and the Gröbner basis attack (see Sect. 4.2 for a generic description of the attacks analyzed here). As in the case of HadesMiMC and Hydra's body, all other attacks do not outperform the ones just listed (see [GLR+20, GØSW23] for more details). Moreover, we explicitly state that we do **not** make any security claim in the related-key setting.

About the Key-Schedule of Pluto. Before going on, we clarify the choice of the key-schedule of Pluto compared to the one of HadesMiMC. Assume that the attacker partially guesses one sub-key k_i. In the case of HadesMiMC, due to its linear key-schedule (2), the attacker only partially knows the relation between the entries of other sub-keys k_j for $j \neq i$, but not the exact values of its entry (in general). This choice aims to frustrate attacks in which the attacker partially guesses multiple sub-keys. Due to the conditions $p > 2^{63}$, $n \geq 4$, and $\kappa \leq \min\{2 \cdot \lfloor \log_2(p) \rfloor, 256\}$, here we claim that a simpler key-schedule (defined via random round constants addition) is sufficient for reaching the same goal.

6.2.1 Statistical Attacks

Let $\mathcal{A}$ over $\mathbb{F}_p^n$ be an invertible affine transformation. As in the case of HadesMiMC, our goal is to show that

$$x \mapsto \underbrace{\mathcal{E}_7 \circ \cdots \circ \mathcal{E}_4}_{=4 \text{ rounds}} \circ \mathcal{A} \circ \underbrace{\mathcal{E}_3 \circ \cdots \circ \mathcal{E}_0}_{=4 \text{ rounds}}(x) \tag{11}$$

is secure against statistical attacks. The security of Pluto follows from the fact that the security of this "weaker" scheme (11) is not reduced when $\mathcal{A}$ is replaced by internal invertible non-linear rounds (note that the internal rounds of Pluto are invertible). As in the case of HadesMiMC, two of the eight external rounds of Pluto aim to frustrate partial key-guessing attacks.

Differential Attack. First of all, we analyze the probability that a collision occurs in PLUTO, keeping in mind that $\mathcal{S}_F$ is a 2^n-bounded surjective function. Since every bijective function is a 1-bounded surjective function (see Lemma 1), the polynomial function corresponding to the PRF PLUTO is $2^{8\cdot n}$-bounded surjective due to Lemma 2 (remember that the number of external rounds is eight). Based on the result proposed in Lemma 3, given an uniformly randomly chosen pair of distinct input values, the probability that a collision occurs at the output of PLUTO is upper bounded by

$$\frac{2^{8\cdot n}-1}{p^n-1} \approx \left(\frac{2^8}{p}\right)^n < 2^{-2\cdot\kappa},$$

where the last inequality holds due to the assumption on κ. Since at most $2^{\kappa/2}$ texts are available for the attack, the attacker can construct at most $\binom{2^{\kappa/2}}{2} \approx 2^{\kappa-1}$ different pairs of texts, which implies that observing a collision is unrealistic for large κ.

Next, consider the case of a differential characteristic without collision. Let $\Delta^I, \Delta^O \in \mathbb{F}_p^n \setminus \{0\}$ be respectively an input/output (non-null) difference. The system of equations $\mathcal{S}_F(x+\Delta^I) - \mathcal{S}_F(x) = \Delta^O$ is satisfied by $x = (s-\Delta^I)/2 \in \mathbb{F}_p^n$ where

$$\begin{bmatrix} 0 & \Delta_1^I & 0 & \dots & 0 \\ 0 & 0 & \Delta_2^I & \dots & 0 \\ \vdots & & & \ddots & \vdots \\ 0 & 0 & 0 & \dots & \Delta_{n-1}^I \\ \Delta_0^I & 0 & 0 & \dots & 0 \end{bmatrix} \times \begin{bmatrix} s_0 \\ s_1 \\ s_2 \\ \vdots \\ s_{n-1} \end{bmatrix} = - \begin{bmatrix} \alpha_{1,0}\cdot\Delta_0^I + \alpha_{0,1}\cdot\Delta_1^I - \Delta_0^O \\ \alpha_{1,0}\cdot\Delta_1^I + \alpha_{0,1}\cdot\Delta_2^I - \Delta_1^O \\ \alpha_{1,0}\cdot\Delta_2^I + \alpha_{0,1}\cdot\Delta_3 - \Delta_2^O \\ \vdots \\ \alpha_{1,0}\cdot\Delta_{n-1}^I + \alpha_{0,1}\cdot\Delta_0^I - \Delta_{n-1}^O \end{bmatrix}.$$

Since $\Delta^O \neq 0 \in \mathbb{F}_p^n$, for each $i \in \{0,1,\dots,n-1\}$:

- if $\Delta_i^I = 0$, then the i-th equality is satisfied if and only if $\alpha_{1,0}\cdot\Delta_{i-1}^I = \Delta_i^O$;
- if $\Delta_i^I \neq 0$, then the i-th equality is satisfied if $s_i = -(\alpha_{1,0}\cdot\Delta_{i-1}^I + \alpha_{0,1}\cdot\Delta_i^I - \Delta_i^O)/\Delta_i^I$.

Hence, the number of different solutions of $\mathcal{S}_F(x+\Delta^I) - \mathcal{S}_F(x) = \Delta^O$ is *at most* equal to p^z, where $0 \leq z \leq n-1$ is the number of zero $\mathbb{F}_p$-components of Δ^I.

Let us now consider two consecutive rounds, and let us introduce:

- $\mathtt{a}_0 :=$ number of active (i.e., non-zero) $\mathbb{F}_p$-components at the *input* of the *first* non-linear layer $\mathcal{S}_F$;
- $\mathtt{a}_1 :=$ number of active (i.e., non-zero) $\mathbb{F}_p$-components at the *output* of the *first* non-linear layer $\mathcal{S}_F$;
- $\mathtt{a}_2 :=$ number of active (i.e., non-zero) $\mathbb{F}_p$-components at the *input* of the *second* non-linear layer $\mathcal{S}_F$.

Given $\mathtt{a}_0 \geq 1$ active inputs $\mathbb{F}_p$-components, then at most $\mathtt{a}_1 \leq \min\{2\cdot\mathtt{a}_0, n\}$ $\mathbb{F}_p$-components are active at the output of $\mathcal{S}_F$. In particular, note that if the $\mathtt{a}_0 \leq n/2$ active input $\mathbb{F}_p$-components are not in consecutive positions, then at most $2\cdot\mathtt{a}_0$ are active in output. Since $M_{\mathcal{E}} \in \mathbb{F}_p^{n\times n}$ is a MDS matrix (hence, its branch number is $n+1$ – see e.g. [DR01,DR02] for details), then at least $\mathtt{a}_2 \geq n+1-\mathtt{a}_1$ $\mathbb{F}_p$-components are active at the input of the second round. Over two consecutive rounds, the probability of a differential trail is approximately given by

$$\begin{aligned} \frac{1}{p^{2\cdot n}} \cdot \underbrace{p^{n-\mathtt{a}_0}}_{\text{1st round}} \cdot \underbrace{p^{n-\mathtt{a}_2}}_{\text{2nd round}} &= p^{-\mathtt{a}_0-\mathtt{a}_2} \leq p^{-\mathtt{a}_0+\mathtt{a}_1-n-1} \\ &\leq p^{-\mathtt{a}_0+\min\{2\mathtt{a}_0,n\}-n-1} = p^{\min\{\mathtt{a}_0-n-1,-1-\mathtt{a}_0\}}, \end{aligned}$$

since $\mathtt{a}_2 \geq n+1-\mathtt{a}_1 \geq 1$ and since $\mathtt{a}_1 \leq \min\{2 \cdot \mathtt{a}_0, n\}$. It follows that the probability over two rounds is upper bounded by

$$\max_{1\leq \mathtt{a}_0 \leq n} p^{\min\{\mathtt{a}_0-n-1,-1-\mathtt{a}_0\}} = \max\left\{ \max_{1\leq \mathtt{a}_0\leq \lfloor n/2\rfloor} p^{\mathtt{a}_0-n-1}, \max_{\lceil n/2\rceil\leq \mathtt{a}_0\leq n} p^{-1-\mathtt{a}_0} \right\} = p^{-\lceil n/2\rceil -1}\,.$$

Given two consecutive rounds per two times (equivalently to four external rounds only), the probability of any differential trail based on trails with non-zero differences is at most equal to

$$\left(p^{-\lceil n/2\rceil-1}\right)^2 \leq p^{-2}\cdot 2^{-\frac{2}{4}\cdot n\cdot\kappa} \leq p^{-2}\cdot 2^{-2\cdot\kappa}$$

since $2^\kappa \leq p^2$ and since $n \geq 4$, which is much smaller than the security level. Moreover, besides the external rounds $\mathcal{E}$, the internal rounds $\mathcal{I}$ guarantee security against classical differential attack as well, as pointed out in e.g. [KR21]. In particular, since no invariant subspace with no active non-linear function can cover $R_\mathcal{I}/2$ (or more) internal rounds (see [GØSW23] for details), then the probability of each characteristic of four external rounds and $R_\mathcal{I}$ internal rounds is upper bounded by

$$p^{-2}\cdot 2^{-2\cdot\kappa}\cdot\left(\frac{3}{p}\right)^{\lfloor R_\mathcal{I}/2\rfloor}\,,$$

where $3/p$ is the maximum differential probability of $\mathcal{S}_\mathcal{I}$, as proved in [GØSW23, App. H]. Based on this, we conclude that Pluto instantiated with eight external rounds is secure against classical differential attacks with a data limit of $2^{\kappa/2}$ texts available for the attacker.

Other Statistical Attacks. As in the case of HadesMiMC, eight external rounds are sufficient for preventing other statistical attacks, including the linear one [Mat93], the truncated differential one [Knu94], the impossible differential [Knu98, BBS99], the boomerang attack [Wag99], the integral one [DKR97], the multiple-of-n/mixture differential [GRR17, Gra18], among others. This follows from the fact that no truncated differential with probability 1 can cover more than a single external round, due to the facts that (1st) $M_\mathcal{E}$ is an MDS matrix and (2nd) $\mathcal{S}_F$ is a full non-linear layer.

6.2.2 Algebraic Attacks

As before, we assume that two of the eight external rounds of Pluto aim to frustrate partial key-guessing attacks.

Non-Invertibility: Local Inverse Functions with High Degree. While the internal rounds are invertible, the function $\mathcal{S}_F$ (and so the external rounds $\mathcal{E}_i$) is not invertible. Let $\mathtt{R} = \{\mathcal{S}_F(x) \in \mathbb{F}_p^n \mid x \in \mathbb{F}_p\}$. Working as in Sect. 4.2.2, we can define three sets $\mathtt{X}_+, \mathtt{X}_-, \mathtt{Z} \subset \mathbb{F}_p^n$ and "local" inverses $\mathcal{Z}_-^{-1} : \mathtt{R} \to \mathtt{X}_+ \cup \mathtt{Z}$ and $\mathcal{Z}_-^{-1} : \mathtt{R} \to \mathtt{X}_- \cup \mathtt{Z}$ that satisfy the following:

1. $\mathtt{X}_+ \cup \mathtt{X}_- \cup \mathtt{Z} = \mathbb{F}_p^n$;
2. for each $x, x' \in \mathbb{F}_p^n \setminus \mathtt{Z}$ such that $\mathcal{S}_F(x) = \mathcal{S}_F(x')$ and $x \neq x'$, then (i) $x \in \mathtt{X}_+$ and $x' \in \mathtt{X}_-$, and (ii) $x \notin \mathtt{X}_-$ and $x' \notin \mathtt{X}_+$ (or vice-versa);
3. given $z \in \mathtt{Z}$, then $\mathcal{S}_F(y) \neq \mathcal{S}_F(z)$ for each $y \in \mathbb{F}_p^n \setminus \{z\}$;
4. $\mathcal{S}_F(\mathcal{Z}_+^{-1}(x)) = x$ and $\mathcal{S}_F(\mathcal{Z}_-^{-1}(x)) = x$ for each $x \in \mathtt{R}$.

The first three points imply that $\mathtt{X}_+ \cap \mathtt{X}_- = \mathtt{X}_+ \cap \mathtt{Z} = \mathtt{X}_- \cap \mathtt{Z} = \emptyset$. Since a collision can occur if and only if $x_i \neq x'_i$ for each $i \in \{0, 1, \ldots, n-1\}$, we have that

$$|\mathtt{X}_+| = |\mathtt{X}_-| = \frac{(p-1)^n}{2} \approx \frac{p^{n-1} \cdot (p-n)}{2} \qquad \text{and} \qquad |\mathtt{Z}| = p^n - (p-1)^n \approx n \cdot p^{n-1}\,.$$

Analogous to the analysis proposed for `MiMC++`, while $\mathtt{Z}$ is uniquely defined, there are several equivalent representations of $\mathtt{X}_+$ and $\mathtt{X}_-$ (by carefully swapping two elements x and x', as in the case of $x \mapsto x^2$).

The algebraic expression of $\mathcal{Z}_+$ and $\mathcal{Z}_-$ obviously depend on the details of the sets $\mathtt{X}_+, \mathtt{X}_-, \mathtt{Z}$. However, we expect that such local inverses have high degree in general, due to the argument given in the proof of Prop. 2 adapted to this case. Given $y = \mathcal{S}_F(x)$, let $x_i = G_{y_i}(x_{i+1}) := y_i - x_{i+1}^2$ for each $i \in \{0, 1, \ldots, n-1\}$, where $G_{y_i} : \mathbb{F}_p \to \mathbb{F}_p$ is defined as in (8). In order to compute the pre-image of a given y, one has to invert the polynomial equation $H_{y_0, y_1, \ldots, y_{n-1}}(x_0) := G_{y_0} \circ G_{y_1} \circ \ldots \circ G_{y_{n-1}}(x_0) - x_0 = 0$ as in Eq. (9). Since such equation has degree $2^n \geq 16$ (for $n \geq 4$), we expect the inverse formula for x_0 to have high degree in $y_0, y_1, \ldots, y_{n-1}$, similar to what happens for the local inverses of $x \mapsto x^2$. Once x_0 is given, x_i can be found via the formula $x_i = G_{y_i} \circ G_{y_{i-1}} \circ \ldots \circ G_{y_1}(x_0)$ for each $i \geq 1$. Since each function G_y has degree 2, it follows that the algebraic expression for finding x_i has degree 2^i in x_0 (and so high degree in $y_0, y_1, \ldots, y_{n-1}$).

Interpolation Attack. As explained in Sect. 4.2.2, a primitive can be considered secure against the interpolation attack [JK97] if the number of unknown monomials that defines the scheme is larger than the data available to the attacker.

Based on the previous analysis, we conjecture that the last three external rounds $\mathcal{E}$ are sufficient for stopping backward interpolation attacks, due to the facts that (i) only local inverses exist in the backward direction, (ii) such local inverses have high degree in general, and (iii) combining local inverses over multiple rounds is in general hard. Regarding the forward direction,[10] the degree growth is

$$\underbrace{2^2}_{\text{due to external rounds}} \cdot \underbrace{4^{R_{\mathcal{I}} - n/2 - \log_2(n) - 2}}_{\text{due to internal rounds}}$$

until the reduction takes place, where we discount

- one external round and $\log_2(n) + 1$ internal rounds in order to destroy possible relations existing between the coefficients of the monomials (and so, to ensure full diffusion);[11]
- $n/2 + 1$ extra internal rounds, since an attacker can cover at most $n/2$ internal rounds via an invariant subspace without activating any function of $\mathcal{I}$ (see [GØSW23] for more details);
- three external rounds in order to prevent backward and MitM attacks.

As a result, the number of rounds $R_{\mathcal{I}}$ necessary for preventing MitM interpolation attacks (without key-guessing) must satisfy

$$\forall x \in \{1, 2, \ldots, n\} : \qquad \binom{2^2 \cdot 4^{R_{\mathcal{I}} - n/2 - \log_2(n) - 2} + x}{x} \geq \min\{2^{\kappa/2}, x\}$$

(under the assumption that the upper bound on the number of monomials is tight), where x is the number of variables in which the interpolation polynomial is constructed, and 2^x is the maximum number of data available for the attacker. This implies

$$2^2 \cdot 4^{R_{\mathcal{I}} - n/2 - \log_2(n) - 2} \geq 2^{\kappa/2} \qquad \to \qquad R_{\mathcal{I}} \geq \frac{\kappa}{4} + \frac{n}{2} + \log_2(n) + 1\,.$$

[10] Remember that $\deg(\mathcal{E}_i) = \deg(\mathcal{S}_F) = 2$ and that $\deg(\mathcal{I}_j) = \deg(\mathcal{S}_{\mathcal{I}}) = 4$.

[11] These are exactly the same numbers given in the security analysis of Hydra's body.

Other Algebraic Attacks. As in the case of HadesMiMC/Hydra's body and based on the same argument proposed in Sect. 4.2.2, the security against the MitM interpolation attack implies security against higher-order differential attack [BCD+20], the linearization attack [KS99] and the Gröbner basis one [Buc76]. Without going into the details, the cost of a Gröbner basis attack depends on several factors, including (i) the number of non-linear equations that compose the system to solve, (ii) the number of independent variables, and (iii) the degree of each equations to solve. In particular, such cost depends on the considered representative of the system of equations to solve. Due to the analogous Gröbner basis attacks on HadesMiMC and Hydra given in [GLR+20, Sect. 4] and in [GØSW23, Sect. 7], the two main strategies for setting up such attack on Pluto are as follows:

(1st) working with a system of equations that involve only the inputs/outputs of the entire cipher;

(2nd) considering a system of equations defined at round level.

In the first case, the number of variables is fixed, and the attacker can collect more equations than the number of possible monomials. In such a case, a Gröbner basis attack reduces to a linearization attack, which does not outperform the interpolation attack just described (see the analogous analysis proposed in Sect. 4.2.2). In the second case, the number of variables is proportional to the number of rounds. Due to the analogous result proposed for Hydra's body and HadesMiMC, we conclude that the cost of such strategy is higher than the security level. Other approaches do not seem to be as competitive as the ones just discussed.

6.3 Design Rationale: Advantages of Non-Invertible Non-Linear Layers

Based on the previous security analysis, here we highlight some *security advantages* of the SI-lifting function $\mathcal{S}_F(x) = y$ over $\mathbb{F}_p^n$ defined as

$$(y_0, y_1, \ldots, y_{n-2}, y_{n-1}) := (x_0^2 + x_1, x_1^2 + x_2, \ldots, x_{n-2}^2 + x_{n-1}, \mathbf{x_{n-1}^2} + x_0)$$

with respect to a Type-III Feistel scheme $\mathcal{F}_{\text{III}}(x) = y$ instantiated with the square map $x \mapsto x^2$ defined as

$$(y_0, y_1, \ldots, y_{n-2}, y_{n-1}) := (x_0^2 + x_1, x_1^2 + x_2, \ldots, x_{n-2}^2 + x_{n-1}, x_0)\,.$$

Note that the only difference between the two functions (highlighted in bold) is in the last component, that is, $y_{n-1} = x_{n-1}^2 + x_0$ for $\mathcal{S}_F$ versus $y_{n-1} = x_0$ for $\mathcal{F}_{\text{III}}$. As well known, this difference is crucial for the (non-)invertibility of the two functions.

Let us compare the security of Pluto instantiated with $\mathcal{S}_F$ with respect to a version of Pluto instantiated with the Type-III Feistel scheme $\mathcal{F}_{\text{III}}$, denoted as $\text{Pluto}_{\text{Feistel}}$. Obviously, the main advantage of this second version is being invertible, hence, no collision at its output can occur. Still, in our scenario, the probability that a collision occurs is so small that it does not represent a concrete threat to the security of our design. Having said that, the proposed Pluto has several security advantages with respect to $\text{Pluto}_{\text{Feistel}}$:

- our security analysis with respect to the differential attack relies on counting the minimum number of active square maps over multiple external rounds. This number is strictly smaller for $\text{Pluto}_{\text{Feistel}}$ than for Pluto, since one component of the Type-III Feistel scheme is linear in the input (namely, the one that depends on x_{n-1}). Hence, given the same number of rounds, the security level – given by our analysis – with respect to the differential attack is higher for Pluto than for $\text{Pluto}_{\text{Feistel}}$ (indeed, such component contributes with probability 1 instead of $1/p$);[12]

[12] We do not exclude the possibility to refine and/or even close this gap via a more dedicate analysis or/and tools.

Table 2: Comparison between HadesMiMC (instantiated with $x \mapsto x^3$) and Pluto for the case $p \approx 2^{128}$, $\kappa = 128$, and several values of $n \in \{4, 8, 12, 16\}$.

	n	$R_f + R_{f'}$ & $R_{\mathcal{E}} + R_{\mathcal{E}'}$	R_P & R_I	Multiplicative Complexity
HadesMiMC ($d = 3$)	4	6	47	142 (+ 22.4 %)
Pluto	4	8	42	**116**
HadesMiMC ($d = 3$)	8	6	48	192 (+ 24.7 %)
Pluto	8	8	45	**154**
HadesMiMC ($d = 3$)	12	6	49	242 (+ 24.7 %)
Pluto	12	8	49	**194**
HadesMiMC ($d = 3$)	16	6	49	290 (+ 26.1 %)
Pluto	16	8	51	**230**

- the non-linear layer $\mathcal{S}_F$ is not invertible, and only local inverses with high degrees exist. For comparison, the Type-III Feistel scheme $\mathcal{F}_{\text{III}}$ is invertible, and the degree of its inverse is equal to 2^{n-1} (to be more precise, one component has degree 2^{n-1}, one has degree 2^{n-2}, and so on). In the case in which 2^{n-1} is much smaller than p (e.g., as in the case of Hydra++'s body where $n = 4$ and $p \geq 2^{64}$), the number of rounds for preventing algebraic attacks in the backward direction is much larger for $\text{Pluto}_{\text{Feistel}}$ than for Pluto.

As a result, this is a concrete example of a scenario in which using non-invertible non-linear layer can have concrete advantages with respect to invertible non-linear ones. Analogous advantages hold for the case of the FHE-friendly schemes Pasta and `Rubato`, as we will show in Sect. 7.2.

6.4 Multiplicative Complexity: HadesMiMC/Hydra vs. Pluto/Hydra++

Finally, we compare the multiplicative complexity of HadesMiMC/Hydra with the one of Pluto/Hydra++ respectively. In the following, we denote the size of the text to be encrypted by n.

Pluto vs. HadesMiMC. The number of multiplications required to evaluate Pluto is (approximately)

$$\mathbf{9.125 \cdot n} + \underbrace{2 \cdot \left\lceil 1.125 \cdot \left\lceil \frac{\kappa}{4} + \log_2(n) + 1 \right\rceil \right\rceil}_{\approx \text{ constant w.r.t. } n},$$

where the factor that multiplies n is fixed and (approximately) equal to 9 in Pluto (we recall that its external rounds are always instantiated with a quadratic function *independently* of p). For comparison, the number of multiplications required to evaluate HadesMiMC is

$$\underbrace{\mathbf{6 \cdot (hw(d) + \lfloor \log_2(d) \rfloor - 1)}}_{\geq \mathbf{12}} \cdot \mathbf{n} + \underbrace{(\text{hw}(d) + \lfloor \log_2(d) \rfloor - 1) \cdot \left(4 + \left\lceil \frac{\kappa}{2 \cdot \log_2(d)} \right\rceil + \lceil \log_d(n) \rceil \right)}_{\approx \text{ constant w.r.t. } n},$$

where the factor that multiplies n depends on the value of d (which depends on p) in HadesMiMC, and it is never smaller than 12.

As a result, we have been able to reduce the number of multiplications of HadesMiMC without decreasing its security. A concrete comparison between the two schemes for small values of n is shown in Table 2 for the most common case $p \approx 2^{128}$, $\kappa = 128$ and $d = 3$.

Hydra++ vs. Hydra and Ciminion. A similar conclusion holds when comparing the body of Hydra versus the body of Hydra++, for which we remember that $n = 4$ is fixed. In particular, let us consider the most common case for MPC applications, that is, $p \approx 2^{128}$ $\kappa = 128$ and $d = 3$. The number of $\mathbb{F}_p$-multiplications for computing Ciminion ("without" the key-schedule), Hydra, and Hydra++ are respectively given by

$$\begin{aligned} \text{Ciminion ("without" KS)} : \quad & 89 + 15 \cdot \left\lceil \frac{n}{2} \right\rceil \in \mathcal{O}(7.5 \cdot n)\,, \\ \text{Hydra} : \quad & 132 + 41 \cdot \left\lceil \frac{n}{8} \right\rceil \in \mathcal{O}(5 \cdot n)\,, \\ \text{Hydra++} : \quad & \mathit{116} + 41 \cdot \left\lceil \frac{n}{8} \right\rceil \in \mathcal{O}(5 \cdot n)\,. \end{aligned}$$

Hydra++'s body requires 116 $\mathbb{F}_p$-multiplications versus 132 $\mathbb{F}_p$-multiplications for Hydra's body (that is, 13.8% more). The gap growths for bigger values of d. As a result, this new variant of Hydra reduces the gap between the cost of Ciminion's body with respect to the one of Hydra's body.

Remark 5. We remark that Hydra (and so Hydra++) outperform Ciminion ("without" the key-schedule) for large values of n, while they all have similar performances for small values of n. At the same time, in the *common* scenario in which the secret symmetric key is shared among the parties, Hydra (and so Hydra++) are much more competitive than Ciminion, whose performance is significantly reduced due to the fact that its expensive key-schedule must also be computed in MPC for an extra/additional cost of $89 \cdot n$ $\mathbb{F}_p$-multiplications (see [GØSW23] for more details).

7 Implications of Our Results on FHE-friendly Symmetric Schemes

Masta [HKC+20], Pasta [DGH+21], and `Rubato` [HKL+22] are recent PRFs over $\mathbb{F}_p^n$ proposed for homomorphic encryption. For each one of these schemes, in the following we show that it could be possible to achieve better performance and/or security by modifying them with the non-linear layers proposed in this paper. Since all these schemes are inspired by Rasta [DEG+18], we first recall it for pointing out the main common design strategy of all these FHE-friendly PRFs.

Remark 6. In this section, we limit ourselves to present evidences that the security analysis proposed by the designers of the considered FHE-friendly schemes can be easily extend to the modified versions proposed here. For this reason, we limit ourselves to consider those FHE-friendly schemes that are instantiated via a degree-2 non-linear layers, excluding e.g. the FHE-friendly scheme `HERA` [CHK+21] (instantiated with power maps of degree at least 3). We encourage to perform a complete security analysis of the modified FHE-friendly schemes proposed here before using them in practical applications.

7.1 Preliminary: Rasta

Rasta is a family of FHE-friendly stream ciphers over $\mathbb{F}_2^n$ for odd n proposed at Crypto 2018. Given an input $x \in \mathbb{F}_2^n$ to encrypt, a public nonce $N \in \mathbb{F}_2^n$ and a public block index counter $i \in \mathbb{N}$, the ciphertext is generated as

$$(x, N) \mapsto (x + \mathtt{K} + \mathcal{P}_{N,i}(\mathtt{K}), N)$$

for a secret key $\mathtt{K} \in \mathbb{F}_2^n$. The public permutation $\mathcal{P}_{N,i} : \mathbb{F}_2^n \to \mathbb{F}_2^n$ consists of several rounds $R \geq 1$ of affine layers and non-linear layers of the form

$$\mathcal{P}_{N,i}(\cdot) = \mathcal{A}_{R,N,i} \circ \mathcal{S}_\chi \circ \ldots \circ \mathcal{A}_{1,N,i} \circ \mathcal{S}_\chi \circ \mathcal{A}_{0,N,i}(\cdot)\,, \tag{12}$$

where

- $\mathcal{S}_\chi$ over $\mathbb{F}_2^n$ is the SI-lifting function induced by the local map $\chi : \mathbb{F}_2^3 \to \mathbb{F}_2$ defined as $\chi(x_0, x_1, x_2) = x_0 + x_2 + x_1 \cdot x_2$. We recall that $\mathcal{S}_\chi$ is invertible for odd $n \geq 3$;

- for each $j \in \{0, 1, \ldots, r\}$, $\mathcal{A}_{j,N,i} = M_{j,N,i} \times x + c_{j,N,i}$ is an affine function over $\mathbb{F}_2^n$, where $M_{j,N,i} \in \mathbb{F}_2^{n \times n}$ is an invertible matrix and $c_{j,N,i} \in \mathbb{F}_2^n$.

In order to minimize the multiplicative depth, the design strategy adapted for Rasta is quite different from the one usually adopted by "traditional/classical" symmetric primitives. In general, given the size n and the security level κ, the number of rounds R of a symmetric primitive is chosen in order to guarantee security (e.g., so that no known attack published in the literature can break the scheme, besides a possible security margin). Exactly the opposite occurs for Rasta. In order to minimize the depth (note that each round has depth one, since $\mathcal{S}_\chi$ is a quadratic function and $\mathcal{A}_{j,N,i}$ is an affine function), given the number of rounds R and the security level κ, the size n is chosen in order to guarantee security, that is, in order to frustrate any possible attack on the scheme. This usually results in a much larger state size compared to the one of "traditional/classical" symmetric primitives.

Besides that, another crucial feature of Rasta regards the affine layers $\mathcal{A}_{j,N,i}$, which are *not* fixed. At each new encryption, new random invertible affine layers $\mathcal{A}_{0,N,i}, \ldots, \mathcal{A}_{R,N,i}$ are generated via a public XOF that takes as input the nonce N and the counter i. This fact has a crucial impact on the security against statistical attacks, such as linear or differential attacks. For a "traditional/classical" cipher, given a set of inputs and corresponding outputs encrypted via the same algorithm, the attacker performs statistical analysis on the output distribution in order to break the scheme. However, such strategy does not work in the case of Rasta, since each input is encrypted via a *different* encryption scheme.

As a result, the main attack vector against Rasta results the linearization one, whose cost – recalled in see Sect. 4.2.2 – is proportional to the number of monomials that define the analyzed function.[13]

7.2 Implications on Masta, Pasta, and Rubato

7.2.1 Masta

Masta can be seen as a direct translation of Rasta to $\mathbb{F}_p^n$ for a prime integer $p \geq 3$. Both in Rasta and in Masta, the non-linear layer is defined via the SI-lifting function $\mathcal{S}_\chi$ over the entire state $\mathbb{F}_q^n$ (where $q = 2$ for Rasta and $q = p$ for Masta) induced by the chi function $\chi : \mathbb{F}_q^3 \to \mathbb{F}_q$ defined as before. The main difference between Masta and Rasta regards the way in which the invertible matrices $M_{j,N,i}$ that define the affine layers are generated.

A Possible Modified Version of Masta based on Our Results. Our result proposed in Sect. 5.4 implies that $\mathcal{S}_\chi$ *over* $\mathbb{F}_p^n$ *for* $p \geq 3$ *and* $n \geq 3$ *is never invertible.* This fact can be easily proven by adapting the proof just given for such function. In the analyzed case, the equality given in (10) and corresponding to the collision $\mathcal{S}_\chi(x) = \mathcal{S}_\chi(y)$ re-written via the

[13] Here, we do not discuss other attacks on Rasta that have been recently proposed in the literature [DMRS20, LSMI21, LSMI22], since they exploit the details of the non-linear $\mathcal{S}_\chi$ over $\mathbb{F}_2^n$ and they (currently) do not apply to the prime field case.

variables $d, s \in \mathbb{F}_p^n$ becomes

$$\begin{bmatrix} 0 & d_2 & d_1 & 0 & \dots & 0 \\ 0 & 0 & d_3 & d_2 & \dots & 0 \\ \vdots & & & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & d_{n-1} & d_{n-2} \\ d_{n-1} & 0 & 0 & \dots & 0 & d_0 \\ d_1 & d_0 & 0 & \dots & 0 & 0 \end{bmatrix} \times \begin{bmatrix} s_0 \\ s_1 \\ s_2 \\ \vdots \\ s_{n-2} \\ s_{n-1} \end{bmatrix} = - \begin{bmatrix} d_0 + d_2 \\ d_1 + d_3 \\ d_2 + d_4 \\ \vdots \\ d_{n-2} + d_0 \\ d_{n-1} + d_1 \end{bmatrix} .$$

Note that the l.h.s. matrix in this equality corresponds to the l.h.s. matrix in (10) after a re-arrangement of the rows. Since the collision event $\mathcal{S}_\chi(x) = \mathcal{S}_\chi(y)$ only depends on the details of such matrix, the result follows immediately.

Hence, replacing $\mathcal{S}_\chi$ in Masta with the SI-lifting function $\mathcal{S}_F$ over $\mathbb{F}_p^n$ induced by $F(x_0, x_1) = x_0^2 + x_1$ implies the following advantages:

- the costs of $\mathcal{S}_\chi$ and of $\mathcal{S}_F$ in terms of multiplications is equal;
- the resistance of Masta against linearization attacks does not change, since the number of quadratic monomials of $\mathcal{S}_F$ and of $\mathcal{S}_\chi$ are equal;
- the probability that a collision occurs is smaller.

Regarding this last point, at the current state of the art, no attack on Masta based on the fact that $\mathcal{S}_\chi$ is not invertible has been proposed in the literature. As in the case of Rasta, this is related to the fact that statistical attacks are frustrated by the change of the affine layer at every encryption. At the same time, the proposed change allows to reduce the collision probability without any other counter-effect.

7.2.2 Pasta

With respect to Masta, Pasta is a variant of Rasta over $\mathbb{F}_p^n$ instantiated with invertible non-linear layers only, and in which the feed-forward is replaced by a final truncation. That is, given an input $x \in \mathbb{F}_p^n$ to encrypt, a public nonce $N \in \mathbb{F}_p^n$ and a public block index counter $i \in \mathbb{N}$, the ciphertext is generated as

$$(x, N) \mapsto (x + \mathcal{T}_{2n,n} \circ \mathcal{P}_{N,i}(\mathtt{K}), N)$$

for a secret key $\mathtt{K} \in \mathbb{F}_p^{2n}$, a public permutation $\mathcal{P}_{N,i} : \mathbb{F}_p^{2n} \to \mathbb{F}_p^{2n}$, and a truncation function $\mathcal{T}_{2n,n} : \mathbb{F}_p^{2n} \to \mathbb{F}_p^n$. As in the case of Rasta, the public permutation $\mathcal{P}_{N,i}$ consists of several rounds $R \geq 1$ of affine layers and non-linear layers of the form (12), where the non-linear layer in the first $R-1$ rounds is instantiated via two parallel (independent) Type-III Feistel schemes over $\mathbb{F}_p^n$ of the form

$$(x_0, \dots, x_{n-2}, x_{n-1}) \mapsto (x_0 + (x_1)^2, \dots, x_{n-2} + (x_{n-1})^2, x_{n-1}) \,, \tag{13}$$

while the last round is instantiated via power maps $(x_0, x_1, \dots, x_{2n-1}) \mapsto (x_0^d, x_1^d, \dots, x_{2n-1}^d)$ for an integer $d \geq 3$ such that $\gcd(d, p-1) = 1$.

A Possible Modified Version of Pasta based on Our Results. One of the selling points of Pasta compared to Masta regards the fact that no internal collision can occur due to the non-invertibility of the non-linear layer, see [DGH+21, Sect. 5.2]: "*the χ-function is in general* [actually, never] *no permutation when working over $\mathbb{F}_p^t$, which is why we consider some alternatives*". (Remember that a collision at the output can always occur, since Pasta as well as Masta and Rasta are not invertible due to the feed-forward/truncation construction). However, as we already pointed out, it seems hard that an internal collision can be extended into an attack on the entire scheme. Indeed, assume that a collision occurs at the $\tilde{R}$-th round for $\tilde{R} < R$, that is,

$$\mathcal{S}_\chi \circ \mathcal{A}_{\tilde{R},N,i} \circ \ldots \circ \mathcal{A}_{1,N,i} \circ \mathcal{S}_\chi \circ \mathcal{A}_{0,N,i}(x) = \mathcal{S}_\chi \circ \mathcal{A}_{\tilde{R},N',i'} \circ \ldots \circ \mathcal{A}_{1,N',i'} \circ \mathcal{S}_\chi \circ \mathcal{A}_{0,N',i'}(x)$$

for $N \neq N'$ and $i \neq i'$. Since $\mathcal{A}_{j,N,i}$ is (generally) different from $\mathcal{A}_{j,N',i'}$ for each $j \in \{\tilde{R}+1, \ldots, R\}$, such collision does not survive the next rounds. Hence, replacing the Type-III Feistel scheme as in (13) with the SI-lifting function $\mathcal{S}_F$ over $\mathbb{F}_2^n$ induced by $F(x_0, x_1) = x_0^2 + x_1$ implies the following advantages:

- the depth of the entire construction (and so the overall cost) does not change;
- the obtained construction would be (slightly) more resistant against the linearization attack, since the number of quadratic monomials in $\mathcal{S}_F$ is slightly bigger than in the Type-III Feistel scheme as in (13).

In particular, it is crucial to keep in mind that a collision occurs with probability approximately p^{-n}, which is much smaller than the security level due to the large size of Pasta (and of Rasta-like design schemes in general). For all these reasons, we claim that the advantages just proposed do not imply a smaller security level.

7.2.3 `Rubato`

`Rubato` is a family of *noisy* stream ciphers over $\mathbb{F}_p^n$ based on the Rasta design strategy, targeting the transciphering framework for approximate homomorphic encryption. The main difference with Pasta regards the way in which the encryption is performed. First of all, given a certain $m \geq 1$ and an input $x \in \mathbb{F}_p^n$ to encrypt and a public nonce $N \in \mathbb{F}_p^{n+m}$, the ciphertext is generated as

$$(x, N) \mapsto (x + \mathcal{N}_G + \mathcal{T}_{n+m,n} \circ \mathcal{E}_{\mathtt{K}}(N), N)$$

for a secret key $\mathtt{K} \in \mathbb{F}_p^{n+m}$, a cipher $\mathcal{E}_{\mathtt{K}} : \mathbb{F}_p^{n+m} \to \mathbb{F}_p^{n+m}$, and a Gaussian noise $\mathcal{N}_G \in \mathbb{F}_p^n$. With respect to the public permutation $\mathcal{P}_{N,i}$ used in Rasta, Masta, and Pasta:

- a round-key addition takes place at each round of $\mathcal{E}_{\mathtt{K}}$;
- the affine layers that define $\mathcal{E}_{\mathtt{K}}$ are *fixed*, that is, they do not change at each encryption;
- the round-keys are generated via an affine maps that change at every encryption (as before, such affine maps are generated via a public XOF that takes in input the nonce N).

In more details, the l-round sub-key $k_l \in \mathbb{F}_p^n$ for $l \in \{0, 1, \ldots, r\}$ of the i encryption for $i \geq 0$ is defined as $k_l = \mathcal{A}'_{l,N,i}(\mathtt{K})$ for an invertible affine layer $\mathcal{A}'_{l,N,i}$ over $\mathbb{F}_p^n$. We refer to [HKL+22] for more details. We emphasize that the noise addition does not play any role in the following argument.

A Possible Modified Version of `Rubato` based on Our Results. Having said that, since `Rubato` is instantiated with the same non-linear layers of Pasta, that is, the quadratic Type-III Feistel scheme given in (13), the observations just proposed for Pasta translate directly to `Rubato` as well.

8 Final Warning

We conclude with the following warning:

> *we discourage the use of non-bijective components for designing symmetric primitives in which the internal state is not obfuscated by a secret (e.g., a secret key).*

In particular, we discourage the use of non-bijective components for building sponge-based hash functions. Indeed, due to the fact that the internal state is known (equivalently, *not* "masked" by a secret key), the attacker can potentially use the full control they have over the inputs to ensure that they trigger a collision. Consider the case of a sponge over $\mathbb{F}_p^n$ instantiated with a low-degree non-bijective function, such as the SI-lifting function $\mathcal{S}_F$ induced by $F(x_0, x_1) = x_0^2 + x_1$. Due to the low degree of such function, an internal collision can be potentially set up after R rounds by solving a system of n equations of degree 2^R in $2 \cdot r$ variables (where r is the rate).

For comparison, a similar problem does not arise – in general – for symmetric primitives that depend on some secret key material, due to the fact that the concrete value of the internal state is unknown and "masked" by the secret key.

Acknowledgments

Author thanks anonymous ToSC reviewers for their valuable comments and suggestion. In particular, we thank them for (i) pointing out a mistake in the previous version of the paper (it was wrongly claimed that $\mathcal{S}_F$ over $\mathbb{F}_p^n$ instantiated with $F(x_0, x_1) = x_0^2 + x_1$ – or equivalent – was 2-bounded surjective) and for (ii) suggesting the "l-bounded surjective" terminology. Author also thanks Morten Øygarden for his valuable comments that helped to improve the quality of the paper. Finally, author thanks Tyge Tiessen for shepherding the final version of the paper. This work was supported by the German Research Foundation (DFG) within the framework of the Excellence Strategy of the Federal Government and the States – EXC 2092 CaSa – 39078197.

References

[AAB+20] Abdelrahaman Aly, Tomer Ashur, Eli Ben-Sasson, Siemen Dhooghe, and Alan Szepieniec. Design of Symmetric-Key Primitives for Advanced Cryptographic Protocols. *IACR Trans. Symmetric Cryptol.*, 2020(3):1–45, 2020.

[ACG+19] Martin R. Albrecht, Carlos Cid, Lorenzo Grassi, Dmitry Khovratovich, Reinhard Lüftenegger, Christian Rechberger, and Markus Schofnegger. Algebraic Cryptanalysis of STARK-Friendly Designs: Application to MARVELlous and MiMC. In *ASIACRYPT 2019*, volume 11923 of *LNCS*, pages 371–397, 2019.

[AGP+19] Martin R. Albrecht, Lorenzo Grassi, Léo Perrin, Sebastian Ramacher, Christian Rechberger, Dragos Rotaru, Arnab Roy, and Markus Schofnegger. Feistel Structures for MPC, and More. In *ESORICS 2019*, volume 11736 of *LNCS*, pages 151–171, 2019.

[AGR+16] Martin R. Albrecht, Lorenzo Grassi, Christian Rechberger, Arnab Roy, and Tyge Tiessen. MiMC: Efficient Encryption and Cryptographic Hashing with Minimal Multiplicative Complexity. In *ASIACRYPT 2016*, volume 10031 of *LNCS*, pages 191–219, 2016.

[AMT22] Tomer Ashur, Mohammad Mahzoun, and Dilara Toprakhisar. Chaghri - A FHE-friendly Block Cipher. In *Proceedings of the 2022 ACM SIGSAC*

Conference on Computer and Communications Security, CCS 2022, pages 139–150. ACM, 2022.

[BBC+22] Clémence Bouvier, Pierre Briaud, Pyrros Chaidos, Léo Perrin, and Vesselin Velichkov. Anemoi: Exploiting the Link between Arithmetization-Orientation and CCZ-Equivalence. Cryptology ePrint Archive, Paper 2022/840, 2022. https://eprint.iacr.org/2022/840.

[BBS99] Eli Biham, Alex Biryukov, and Adi Shamir. Cryptanalysis of Skipjack Reduced to 31 Rounds Using Impossible Differentials. In *EUROCRYPT 1999*, volume 1592 of *LNCS*, pages 12–23, 1999.

[BCD+20] Tim Beyne, Anne Canteaut, Itai Dinur, Maria Eichlseder, Gregor Leander, Gaëtan Leurent, María Naya-Plasencia, Léo Perrin, Yu Sasaki, Yosuke Todo, and Friedrich Wiemer. Out of Oddity - New Cryptanalytic Techniques Against Symmetric Primitives Optimized for Integrity Proof Systems. In *CRYPTO 2020*, volume 12172 of *LNCS*, pages 299–328, 2020.

[BCP23] Clémence Bouvier, Anne Canteaut, and Léo Perrin. On the algebraic degree of iterated power functions. *Des. Codes Cryptogr.*, 91(3):997–1033, 2023.

[BDH+17] Guido Bertoni, Joan Daemen, Seth Hoffert, Michaël Peeters, Gilles Van Assche, and Ronny Van Keer. Farfalle: parallel permutation-based cryptography. *IACR Trans. Symmetric Cryptol.*, 2017(4):1–38, 2017.

[BDPA08] Guido Bertoni, Joan Daemen, Michaël Peeters, and Gilles Van Assche. On the Indifferentiability of the Sponge Construction. In *EUROCRYPT 2008*, volume 4965 of *LNCS*, pages 181–197, 2008.

[BDPA13] Guido Bertoni, Joan Daemen, Michaël Peeters, and Gilles Van Assche. Keccak. In *EUROCRYPT 2013*, volume 7881 of *LNCS*, pages 313–314, 2013.

[BS90] Eli Biham and Adi Shamir. Differential Cryptanalysis of DES-like Cryptosystems. In *CRYPTO 1990*, volume 537 of *LNCS*, pages 2–21, 1990.

[BS93] Eli Biham and Adi Shamir. *Differential Cryptanalysis of the Data Encryption Standard*. Springer, 1993.

[Buc76] Bruno Buchberger. A theoretical basis for the reduction of polynomials to canonical forms. *SIGSAM Bull.*, 10(3):19–29, 1976.

[CHK+21] Jihoon Cho, Jincheol Ha, Seongkwang Kim, ByeongHak Lee, Joohee Lee, Jooyoung Lee, Dukjae Moon, and Hyojin Yoon. Transciphering Framework for Approximate Homomorphic Encryption. In *ASIACRYPT 2021*, volume 13092 of *LNCS*, pages 640–669, 2021.

[DEG+18] Christoph Dobraunig, Maria Eichlseder, Lorenzo Grassi, Virginie Lallemand, Gregor Leander, Eik List, Florian Mendel, and Christian Rechberger. Rasta: A Cipher with Low ANDdepth and Few ANDs per Bit. In *CRYPTO 2018*, volume 10991 of *LNCS*, pages 662–692, 2018.

[DGGK21] Christoph Dobraunig, Lorenzo Grassi, Anna Guinet, and Daniël Kuijsters. Ciminion: Symmetric Encryption Based on Toffoli-Gates over Large Finite Fields. In *EUROCRYPT 2021*, volume 12697 of *LNCS*, pages 3–34, 2021.

[DGH+21] Christoph Dobraunig, Lorenzo Grassi, Lukas Helminger, Christian Rechberger, Markus Schofnegger, and Roman Walch. Pasta: A Case for Hybrid Homomorphic Encryption. Cryptology ePrint Archive, Report 2021/731, 2021. https://ia.cr/2021/731.

[DGV91] J. Daemen, R. Govaerts, and J. Vandewalle. Efficient pseudorandom sequence generation by cellular automata. In *Proceedings of the 12th Symposium on Information Theory in the Benelux, Werkgemeenschap voor Informatie- en Communicatietheorie*, 1991.

[DKR97] Joan Daemen, Lars R. Knudsen, and Vincent Rijmen. The block cipher Square. In *FSE 1997*, volume 1267 of *LNCS*, pages 149–165, 1997.

[DMRS20] Christoph Dobraunig, Farokhlagha Moazami, Christian Rechberger, and Hadi Soleimany. Framework for faster key search using related-key higher-order differential properties: applications to Agrasta. *IET Inf. Secur.*, 14(2):202–209, 2020.

[DR01] Joan Daemen and Vincent Rijmen. The Wide Trail Design Strategy. In *IMA International Conference 2001*, volume 2260 of *LNCS*, pages 222–238, 2001.

[DR02] Joan Daemen and Vincent Rijmen. Security of a Wide Trail Design. In *INDOCRYPT 2002*, volume 2551 of *LNCS*, pages 1–11, 2002.

[EGL+20] Maria Eichlseder, Lorenzo Grassi, Reinhard Lüftenegger, Morten Øygarden, Christian Rechberger, Markus Schofnegger, and Qingju Wang. An Algebraic Attack on Ciphers with Low-Degree Round Functions: Application to Full MiMC. In *ASIACRYPT 2020*, volume 12491 of *LNCS*, pages 477–506, 2020.

[Gen07] Giulio Genovese. Improving the algorithms of berlekamp and niederreiter for factoring polynomials over finite fields. *J. Symb. Comput.*, 42(1-2):159–177, 2007.

[GHR+22] Lorenzo Grassi, Yonglin Hao, Christian Rechberger, Markus Schofnegger, Roman Walch, and Qingju Wang. Horst Meets *Fluid*-SPN: Griffin for Zero-Knowledge Applications. Cryptology ePrint Archive, Report 2022/403, 2022. https://ia.cr/2022/403.

[GKL+22] Lorenzo Grassi, Dmitry Khovratovich, Reinhard Lüftenegger, Christian Rechberger, Markus Schofnegger, and Roman Walch. Reinforced Concrete: A Fast Hash Function for Verifiable Computation. In *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security, CCS 2022*, pages 1323–1335. ACM, 2022.

[GKR+21] Lorenzo Grassi, Dmitry Khovratovich, Christian Rechberger, Arnab Roy, and Markus Schofnegger. Poseidon: A New Hash Function for Zero-Knowledge Proof Systems. In *USENIX Security 2021*. USENIX Association, 2021.

[GLR+20] Lorenzo Grassi, Reinhard Lüftenegger, Christian Rechberger, Dragos Rotaru, and Markus Schofnegger. On a Generalization of Substitution-Permutation Networks: The HADES Design Strategy. In *EUROCRYPT 2020*, volume 12106 of *LNCS*, pages 674–704, 2020.

[GOPS22] Lorenzo Grassi, Silvia Onofri, Marco Pedicini, and Luca Sozzi. Invertible Quadratic Non-Linear Layers for MPC-/FHE-/ZK-Friendly Schemes over $\mathbb{F}_p^n$: Application to Poseidon. *IACR Trans. Symmetric Cryptol.*, 2022(3):20–72, 2022.

[GØSW23] Lorenzo Grassi, Morten Øygarden, Markus Schofnegger, and Roman Walch. From Farfalle to Megafono via Ciminion: The PRF Hydra for MPC Applications. In *EUROCRYPT 2023*, volume 14007 of *LNCS*, pages 255–286, 2023.

[Gra18] Lorenzo Grassi. Mixture Differential Cryptanalysis: a New Approach to Distinguishers and Attacks on round-reduced AES. *IACR Trans. Symmetric Cryptol.*, 2018(2):133–160, 2018.

[GRR+16] Lorenzo Grassi, Christian Rechberger, Dragos Rotaru, Peter Scholl, and Nigel P. Smart. MPC-Friendly Symmetric Key Primitives. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, CCS 2016*, pages 430–443. ACM, 2016.

[GRR17] Lorenzo Grassi, Christian Rechberger, and Sondre Rønjom. A New Structural-Differential Property of 5-Round AES. In *EUROCRYPT 2017*, volume 10211 of *LNCS*, pages 289–317, 2017.

[GRS21] Lorenzo Grassi, Christian Rechberger, and Markus Schofnegger. Proving Resistance Against Infinitely Long Subspace Trails: How to Choose the Linear Layer. *IACR Trans. Symmetric Cryptol.*, 2021(2):314–352, 2021.

[HKC+20] Jincheol Ha, Seongkwang Kim, Wonseok Choi, Jooyoung Lee, Dukjae Moon, Hyojin Yoon, and Jihoon Cho. Masta: An HE-Friendly Cipher Using Modular Arithmetic. *IEEE Access*, 8:194741–194751, 2020.

[HKL+22] Jincheol Ha, Seongkwang Kim, ByeongHak Lee, Jooyoung Lee, and Mincheol Son. Rubato: Noisy Ciphers for Approximate Homomorphic Encryption. In *EUROCRYPT 2022*, volume 13275 of *LNCS*, pages 581–610, 2022.

[JK97] Thomas Jakobsen and Lars R. Knudsen. The Interpolation Attack on Block Ciphers. In *FSE 1997*, volume 1267 of *LNCS*, pages 28–40, 1997.

[Knu94] Lars R. Knudsen. Truncated and Higher Order Differentials. In *FSE 1994*, volume 1008 of *LNCS*, pages 196–211, 1994.

[Knu98] Lars R. Knudsen. DEAL - A 128-bit Block Cipher. *Technical Report, Department of Informatics, Bergen, Norway*, 1998.

[KR21] Nathan Keller and Asaf Rosemarin. Mind the Middle Layer: The HADES Design Strategy Revisited. In *EUROCRYPT 2021*, volume 12697 of *LNCS*, pages 35–63, 2021.

[KS99] Aviad Kipnis and Adi Shamir. Cryptanalysis of the HFE Public Key Cryptosystem by Relinearization. In *CRYPTO 1999*, volume 1666 of *LNCS*, pages 19–30, 1999.

[LAAZ11] Gregor Leander, Mohamed Ahmed Abdelraheem, Hoda AlKhzaimi, and Erik Zenner. A Cryptanalysis of PRINTcipher: The Invariant Subspace Attack. In *CRYPTO 2011*, volume 6841 of *LNCS*, pages 206–221, 2011.

[Lai94] X. Lai. Higher order derivatives and differential cryptanalysis. Communications and Cryptography: Two Sides of One Tapestry, 1994.

[LMR15] Gregor Leander, Brice Minaud, and Sondre Rønjom. A Generic Approach to Invariant Subspace Attacks: Cryptanalysis of Robin, iSCREAM and Zorro. In *EUROCRYPT 2015*, volume 9056 of *LNCS*, pages 254–283, 2015.

[LSMI21] Fukang Liu, Santanu Sarkar, Willi Meier, and Takanori Isobe. Algebraic Attacks on Rasta and Dasta Using Low-Degree Equations. In *ASIACRYPT 2021*, volume 13090 of *LNCS*, pages 214–240, 2021.

[LSMI22] Fukang Liu, Santanu Sarkar, Willi Meier, and Takanori Isobe. The Inverse of χ and Its Applications to Rasta-Like Ciphers. *J. Cryptol.*, 35(4):28, 2022.

[Mat93] Mitsuru Matsui. Linear Cryptanalysis Method for DES Cipher. In *EUROCRYPT 1993*, volume 765 of *LNCS*, pages 386–397, 1993.

[Nyb96] Kaisa Nyberg. Generalized Feistel Networks. In *ASIACRYPT 1996*, volume 1163 of *LNCS*, pages 91–104. Springer, 1996.

[Sze21] Alan Szepieniec. On the Use of the Legendre Symbol in Symmetric Cipher Design. Cryptology ePrint Archive, Report 2021/984, 2021. https://ia.cr/2021/984.

[Wag99] David A. Wagner. The boomerang attack. In *Fast Software Encryption – FSE 1999*, volume 1636 of *LNCS*, pages 156–170, 1999.

[Wol85] Stephen Wolfram. Cryptography with Cellular Automata. In *CRYPTO 1985*, volume 218 of *LNCS*, pages 429–432, 1985.

[ZMI90] Yuliang Zheng, Tsutomu Matsumoto, and Hideki Imai. On the Construction of Block Ciphers Provably Secure and Not Relying on Any Unproved Hypotheses. In *CRYPTO 1989*, volume 435 of *LNCS*, pages 461–480, 1990.

A Details of Sect. 5

A.1 $F(x_0, x_1) = x_0^2 + \alpha_{0,2} \cdot x_1^2 + \alpha_{1,0} \cdot x_0 + \alpha_{0,1} \cdot x_1$

Given $F(x_0, x_1) = x_0^2 + \alpha_{0,2} \cdot x_1^2 + \alpha_{1,0} \cdot x_0 + \alpha_{0,1} \cdot x_1$ where $\alpha_{0,2} \neq 0$ (w.l.o.g., we fixed $\alpha_{2,0} = 1$), the system of equations that defines the collision $\mathcal{S}_F(x) = \mathcal{S}_F(y)$ via the variables $d_i := x_i - y_i$ and $s_i := x_i + y_i$ introduced in Eq. (6) is

$$\begin{bmatrix} d_0 & \alpha_{0,2}\cdot d_1 & 0 & 0 & \dots & 0 \\ 0 & d_1 & \alpha_{0,2}\cdot d_2 & 0 & \dots & 0 \\ 0 & 0 & d_2 & \alpha_{0,2}\cdot d_3 & \dots & 0 \\ \vdots & & & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & d_{n-2} & \alpha_{0,2}\cdot d_{n-1} \\ \alpha_{0,2}\cdot d_0 & 0 & 0 & \dots & 0 & d_{n-1} \end{bmatrix} \times \begin{bmatrix} s_0 \\ s_1 \\ s_2 \\ \vdots \\ s_{n-2} \\ s_{n-1} \end{bmatrix} = \tag{14}$$
$$- \begin{bmatrix} \alpha_{1,0}\cdot d_0 + \alpha_{0,1}\cdot d_1 & \alpha_{1,0}\cdot d_1 + \alpha_{0,1}\cdot d_2 & \dots & \alpha_{1,0}\cdot d_{n-1} + \alpha_{0,1}\cdot d_0 \end{bmatrix}^T ,$$

where $\cdot^T$ denotes the transpose matrix/vector. The determinant of the l.h.s. matrix is equal to

$$(1 - (-\alpha_{0,2})^n) \cdot \prod_{i=0}^{n-1} d_i \,.$$

Hence, in order to give a lower bound on the probability of having a collision, we study separately the two cases: (1st) $1 \neq (-\alpha_{0,2})^n$ and (2nd) $1 = (-\alpha_{0,2})^n$. Before going on, we point out that $\mathcal{S}_F$ costs n multiplications by pre-computing $x_0^2, x_1^2, \dots, x_{n-1}^2$.

Analysis of $x, y \in \mathbb{F}_p^n$ such that $\mathcal{S}_F(x) = \mathcal{S}_F(y)$. As first step, we analyze the details of x, y such that $\mathcal{S}_F(x) = \mathcal{S}_F(y)$. In this case, a collision does not necessary occur if $n-1$ input differences d_i are equal to zero. E.g., in the case $d_1 \in \mathbb{F}_p \setminus \{0\}$ and $d_i = 0$ for each $i \neq 1$, the system of equations reduces to

$$\alpha_{0,2} \cdot d_1 \cdot s_1 = -\alpha_{0,1} \cdot d_1 \qquad \text{and} \qquad d_1 \cdot s_1 = -\alpha_{1,0} \cdot d_1 \,,$$

which is satisfied by $s_1 = -\alpha_{1,0}$ and $\alpha_{0,2} \cdot \alpha_{1,0} = \alpha_{0,1}$. If this second condition is not satisfied, then a collision cannot occur, independently of the choice of s_i. At the same time, if at least two differences are non-null (e.g., $d_1, d_2 \in \mathbb{F}_p \setminus \{0\}$), then it is always possible to have a collision (even if $\alpha_{0,1} \neq \alpha_{0,2} \cdot \alpha_{1,0}$). Indeed, in such a case, the system of equations reduces to

$$\begin{aligned} \alpha_{0,2} \cdot d_1 \cdot s_1 &= -\alpha_{0,1} \cdot d_1 \,, \\ d_1 \cdot s_1 + \alpha_{0,2} \cdot d_2 \cdot s_2 &= -\alpha_{1,0} \cdot d_1 - \alpha_{0,1} \cdot d_2 \,, \\ d_2 \cdot s_2 &= -\alpha_{1,0} \cdot d_2 \,, \end{aligned}$$

which is satisfied by $s_1 = -\alpha_{0,1}/\alpha_{0,2}$, $s_2 = -\alpha_{1,0}$ and by $d_1 = d_2 \cdot \alpha_{0,2}$.

Collision Probability for $1 - (-\alpha_{0,2})^n \neq 0$. First of all, if all d_i are non-zero, then the determinant of the l.h.s. matrix is non-zero, and a collision can occur by properly choosing $s_0, s_1, \ldots, s_{n-1}$. Consider the case in which only two differences d_i, d_{i+1} are non-null, and all the others are equal to zero (note that there are n different cases). As pointed out in the previous paragraph, given d_i, a collision can occur if s_i, s_{i+1}, d_{i+1} satisfy some particular relation (note that all the others s_j for $j \in \{0, 1, \ldots, n-1\} \setminus \{i, i+1\}$ are free to take any possible value). As a result, the probability of having a collision is *at least* equal to

$$\frac{(p-1)^n + n \cdot (p-1) \cdot p^{n-2}}{p^n \cdot (p^n - 1)} > \frac{(p-1)^n}{p^n \cdot (p^n - 1)} \,,$$

which is strictly bigger than the probability given in Prop. 1.

Collision Probability for $1 - (-\alpha_{0,2})^n = 0$. If $1 - (-\alpha_{0,2})^n = 0$, then the determinant of the l.h.s. matrix is equal to zero, which means that its rows satisfy a linear relation. Working as in Sect. 5.4, a collision can occur if the r.h.s. of (14) satisfies the same linear relation of the rows of the l.h.s. matrix, which implies that one difference d_i is fixed. W.l.o.g., assume d_0 is fixed. In such a case, the collision takes place if

$$\begin{bmatrix} d_1 & \alpha_{0,2} \cdot d_2 & 0 & \ldots & 0 \\ 0 & d_2 & \alpha_{0,2} \cdot d_3 & \ldots & 0 \\ \vdots & & \ddots & \ddots & \vdots \\ 0 & 0 & \ldots & d_{n-2} & \alpha_{0,2} \cdot d_{n-1} \\ 0 & 0 & \ldots & 0 & d_{n-1} \end{bmatrix} \times \begin{bmatrix} s_1 \\ s_2 \\ \vdots \\ s_{n-2} \\ s_{n-1} \end{bmatrix} = - \begin{bmatrix} \alpha_{1,0} \cdot d_1 + \alpha_{0,1} \cdot d_2 \\ \alpha_{1,0} \cdot d_2 + \alpha_{0,1} \cdot d_3 \\ \vdots \\ \alpha_{1,0} \cdot d_{n-2} + \alpha_{0,1} \cdot d_{n-1} \\ \alpha_{1,0} \cdot d_{n-1} + (\alpha_{0,1} + \alpha_{0,2} \cdot s_0) \cdot d_0 \end{bmatrix} ,$$

where no condition on $s_0 \in \mathbb{F}_p$ holds. The determinant of the l.h.s. matrix is equal to $\prod_{i=1}^{n-1} d_i$, which is different from zero if $d_1, d_2, \ldots, d_{n-1} \in \mathbb{F}_p \setminus \{0\}$. This is sufficient for concluding that the probability of having a collision is *at least* equal to

$$\frac{p \cdot (p-1)^{n-1}}{p^n \cdot (p^n - 1)} > \frac{(p-1)^n}{p^n \cdot (p^n - 1)} \,,$$

which is strictly bigger than the probability given in Prop. 1.

A.2 $F(x_0, x_1) = x_0 \cdot x_1 + \alpha_{2,0} \cdot x_0^2 + \alpha_{1,0} \cdot x_0 + \alpha_{0,1} \cdot x_1$

Consider $F(x_0, x_1) = \alpha_{1,1} \cdot x_0 \cdot x_1 + \alpha_{2,0} \cdot x_0^2 + \alpha_{1,0} \cdot x_0 + \alpha_{0,1} \cdot x_1$ where $\alpha_{1,1}, \alpha_{2,0} \neq 0$ (equivalently, $F(x_0, x_1) = \alpha_{1,1} \cdot x_0 \cdot x_1 + \alpha_{0,2} \cdot x_1^2 + \alpha_{1,0} \cdot x_0 + \alpha_{0,1} \cdot x_1$ where $\alpha_{1,1}, \alpha_{0,2} \neq 0$). W.l.o.g., we fix $\alpha_{1,1} = 2$. Given $F(x_0, x_1) = 2 \cdot x_0 \cdot x_1 + \alpha_{2,0} \cdot x_0^2 + \alpha_{1,0} \cdot x_0 + \alpha_{0,1} \cdot x_1$ where $\alpha_{2,0} \neq 0$, the system of equations that corresponds to the collision $\mathcal{S}_F(x) = \mathcal{S}_F(y)$ via the variables $d_i \coloneqq x_i - y_i$ and $s_i \coloneqq x_i + y_i$ introduced in (6) is[14]

[14] In the equivalent case $F(x_0, x_1) = \alpha_{1,1} \cdot x_0 \cdot x_1 + \alpha_{0,2} \cdot x_1^2 + \alpha_{1,0} \cdot x_0 + \alpha_{0,1} \cdot x_1$, the first and the second diagonals of the matrix are basically swapped. We point out that this does not influence the analysis proposed in this subsection, as e.g. the cases in which the determinant is equal to zero.

$$\begin{bmatrix} \alpha_{2,0}\cdot d_0 + d_1 & d_0 & 0 & 0 & \dots & 0 \\ 0 & \alpha_{2,0}\cdot d_1 + d_2 & d_1 & 0 & \dots & 0 \\ 0 & 0 & \alpha_{2,0}\cdot d_2 + d_3 & d_2 & \dots & 0 \\ \vdots & & & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & \alpha_{2,0}\cdot d_{n-2} + d_{n-1} & d_{n-2} \\ d_{n-1} & 0 & 0 & \dots & 0 & \alpha_{2,0}\cdot d_{n-1} + d_0 \end{bmatrix}$$
$$\times \begin{bmatrix} s_0 & s_1 & s_2 & \dots & s_{n-2} & s_{n-1} \end{bmatrix}^T =$$
$$- \begin{bmatrix} \alpha_{1,0}\cdot d_0 + \alpha_{0,1}\cdot d_1 & \alpha_{1,0}\cdot d_1 + \alpha_{0,1}\cdot d_2 & \dots & \alpha_{1,0}\cdot d_{n-1} + \alpha_{0,1}\cdot d_0 \end{bmatrix}^T .$$

Before going on, note that the function F can be computed via one multiplication only, by re-writing it as $F(x_0, x_1) = x_0 \cdot (\alpha_{1,1}\cdot x_1 + \alpha_{2,0}\cdot x_0) + \alpha_{1,0}\cdot x_0 + \alpha_{0,1}\cdot x_1$.

Analysis of $x, y \in \mathbb{F}_p^n$ such that $\mathcal{S}_F(x) = \mathcal{S}_F(y)$. In this case, a collision can occur even if $n-1$ differences d_i are equal to zero. E.g., if $d_1 \neq 0$ and $d_i = 0$ for each $i \neq 1$, the system of equations reduces to

$$d_1 \cdot s_0 = -\alpha_{0,1}\cdot d_1 \qquad \text{and} \qquad \alpha_{2,0}\cdot d_1 \cdot s_1 + d_1 \cdot s_2 = -\alpha_{1,0}\cdot d_1 ,$$

which is satisfied if $s_0 = -\alpha_{0,1}$ and $s_2 = -\alpha_{1,0} - \alpha_{2,0}\cdot s_1$, where $s_1, s_3, s_4, \dots, s_{n-1}$ are free to take any possible value.

Collision Probability. The determinant of the l.h.s. matrix is

$$\prod_{i=0}^{n-1} (\alpha_{2,0}\cdot d_i + d_{i+1}) - (-1)^n \cdot \prod_{i=0}^{n-1} d_i .$$

By re-writing it with respect to d_0, the determinant is equal to zero if and only if

$$\alpha_{2,0}\cdot\beta\cdot d_0^2 + \left(\alpha_{2,0}^2 \cdot \beta \cdot d_{n-1} + \beta\cdot d_1 - (-1)^n \cdot \prod_{i=1}^{n-1} d_i \right) \cdot d_0 + \alpha_{2,0}\cdot d_1 \cdot \beta \cdot d_{n-1} = 0 ,$$

where $\beta \coloneqq \prod_{i=1}^{n-2} (\alpha_{2,0}\cdot d_i + d_{i+1})$.

The case $\beta \neq 0$ holds if $d_{i-1} \neq -\alpha_{2,0}\cdot d_i$, i.e., $d_{n-1} \in \mathbb{F}_p$ and $d_i \in \mathbb{F}_p \setminus \{-\alpha_{2,0}\cdot d_{i+1}\}$ for each $i \in \{1, 2, \dots, n-2\}$. If $\beta \neq 0$, then the previous equation of degree two admits at most two solutions in d_0. This means that there are at least $p\cdot(p-1)^{n-2}\cdot(p-2)$ different values of $d_0, d_1, \dots, d_{n-1} \in \mathbb{F}_p$ for which the matrix is invertible, and so for which a collision occurs.

As pointed out in the previous paragraph, a collision can also occur if $n-1$ differences are equal to zero. E.g., If $d_i \neq 0$, this happens if s_{i-1} and s_{i+1} satisfy some particular relations given before. Note that this case is excluded from the previous case, since the determinant is equal to zero. This is sufficient for concluding that the probability of having a collision is *at least* equal to

$$\frac{p\cdot(p-1)^{n-2}\cdot(p-2) + n\cdot(p-1)\cdot p^{n-2}}{p^n\cdot(p^n-1)} = \frac{p\cdot(p-1)\cdot((p-1)^{n-3}\cdot(p-2) + n\cdot p^{n-3})}{p^n\cdot(p^n-1)} .$$

Since $(p-1)^{n-3}\cdot(p-2) + n\cdot p^{n-3} \geq (p-1)^{n-2}$ if and only if $n\cdot p^{n-3} \geq (p-1)^{n-3}$ (which is always satisfied), then we conclude that such probability is strictly bigger than the probability given in Prop. 1.

A.3 $F(x_0, x_1) = \alpha_{2,0} \cdot x_0^2 + x_0 \cdot x_1 + \alpha_{0,2} \cdot x_1^2 + \alpha_{1,0} \cdot x_0 + \alpha_{0,1} \cdot x_1$

Given $F(x_0, x_1) = \alpha_{2,0} \cdot x_0^2 + 2 \cdot x_0 \cdot x_1 + \alpha_{0,2} \cdot x_1^2 + \alpha_{1,0} \cdot x_0 + \alpha_{0,1} \cdot x_1$ for $\alpha_{2,0}, \alpha_{0,2} \in \mathbb{F}_p \setminus \{0\}$ (w.l.o.g., we fixed $\alpha_{1,1} = 2$), the system of equations that corresponds to the collision $\mathcal{S}_F(x) = \mathcal{S}_F(y)$ via the variables $d_i := x_i - y_i$ and $s_i := x_i + y_i$ introduced in Def. 6 is

$$\begin{bmatrix} \alpha_{2,0} \cdot d_0 + d_1 & d_0 + \alpha_{0,2} \cdot d_1 & 0 & \dots & 0 & 0 \\ 0 & \alpha_{2,0} \cdot d_1 + d_2 & d_2 + \alpha_{0,2} \cdot d_3 & \dots & 0 & 0 \\ \vdots & & & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & \alpha_{2,0} \cdot d_{n-2} + d_{n-1} & d_{n-2} + \alpha_{0,2} \cdot d_{n-1} \\ d_{n-1} + \alpha_{0,2} \cdot d_0 & 0 & 0 & \dots & 0 & \alpha_{2,0} \cdot d_{n-1} + d_0 \end{bmatrix}$$
$$\times \begin{bmatrix} s_0 & s_1 & s_2 & \dots & s_{n-2} & s_{n-1} \end{bmatrix}^T =$$
$$- \begin{bmatrix} \alpha_{1,0} \cdot d_0 + \alpha_{0,1} \cdot d_1 & \alpha_{1,0} \cdot d_1 + \alpha_{0,1} \cdot d_2 & \dots & \alpha_{1,0} \cdot d_{n-1} + \alpha_{0,1} \cdot d_0 \end{bmatrix}^T . \tag{15}$$

Multiplicative Complexity for Computing $\mathcal{S}_F$. As first thing, we discuss the cost of computing $\mathcal{S}_F$, keeping in mind that our goal is to consider only quadratic non-linear layers over $\mathbb{F}_p^n$ that cost n multiplications. In general, computing $\mathcal{S}_F$, costs $2 \cdot n$ multiplications, since one has to compute both $x_0^2, x_1^2, \dots, x_{n-1}^2$ and $x_0 \cdot x_1, x_1 \cdot x_2, \dots, x_{n-1} \cdot x_0$. However, if F can be re-written as

$$\begin{aligned} F(x_0, x_1) &= (\varphi_0 \cdot x_0 + \varphi_1 \cdot x_1) \cdot (\psi_0 \cdot x_0 + \psi_1 \cdot x_1) + \alpha_{1,0} \cdot x_0 + \alpha_{0,1} \cdot x_1 = \\ &= \varphi_0 \cdot \psi_0 \cdot x_0^2 + (\varphi_0 \cdot \psi_1 + \varphi_1 \cdot \psi_0) \cdot x_0 \cdot x_1 + \varphi_1 \cdot \psi_1 \cdot x_1^2 + \alpha_{1,0} \cdot x_0 + \alpha_{0,1} \cdot x_1 \end{aligned}$$

for certain $\varphi_0, \varphi_1, \psi_0, \psi_1 \in \mathbb{F}_p \setminus \{0\}$, then the cost decreases to exactly n multiplications for $\mathcal{S}_F$. This case occurs if the following equality are all satisfied:

$$\alpha_{2,0} = \varphi_0 \cdot \psi_0, \qquad \alpha_{1,1} = \varphi_0 \cdot \psi_1 + \varphi_1 \cdot \psi_0, \qquad \alpha_{0,2} = \varphi_1 \cdot \psi_1 .$$

Given $\alpha_{2,0}, \alpha_{0,2} \in \mathbb{F}_p \setminus \{0\}$ and $\alpha_{1,1} = 2$, these three equality are satisfied if

$$\alpha_{0,2} \cdot \psi_1^2 - 2\psi_0 \cdot \psi_1 + \alpha_{2,0} \cdot \psi_0 = 0 \qquad \longrightarrow \qquad \psi_1 = \frac{\psi_0 \cdot (1 \pm \sqrt{1 - \alpha_{0,2} \cdot \alpha_{2,0}})}{\alpha_{0,2}} .$$

The only case in which the square root exists *independently of the value of* p is $\alpha_{0,2} \cdot \alpha_{2,0} = 1$. For this reason, we limit ourselves to work with $\alpha_{0,2} \cdot \alpha_{2,0} = 1$ in the following.

Analysis of $x, y \in \mathbb{F}_p^n$ such that $\mathcal{S}_F(x) = \mathcal{S}_F(y)$. First of all, we notice that a collision can occur even if $n - 1$ differences d_i are equal to zero. E.g., if $d_1 \neq 0$ and $d_i = 0$ for each $i \neq 1$, we have

$$d_1 \cdot s_0 + \alpha_{0,2} \cdot d_1 \cdot s_1 = -\alpha_{0,1} \cdot d_1 \qquad \text{and} \qquad \alpha_{2,0} \cdot d_1 \cdot s_1 + d_1 \cdot s_2 = -\alpha_{1,0} \cdot d_1$$

which is satisfied if $s_0 = -\alpha_{0,2} \cdot s_1 - \alpha_{0,1}$ and $s_2 = -\alpha_{2,0} \cdot s_1 - \alpha_{1,0}$, where $s_1, s_3, s_4, \dots, s_{n-1}$ can take any possible value in $\mathbb{F}_p$.

Collision Probability. As before, our goal is to show that the probability of having a collision is strictly bigger than $\frac{(p-1)^n}{p^n \cdot (p^n - 1)}$. By simple computation, the determinant of the matrix is equal to

$$\prod_{i=0}^{n-1} (\alpha_{2,0} \cdot d_i + d_{i+1}) - (-1)^n \cdot \prod_{i=0}^{n-1} (d_i + \alpha_{0,2} \cdot d_{i+1}) .$$

Following the strategy proposed in App. A.2 and by re-writing the determinant with respect to d_0, it is equal to zero if and only if

$$\begin{aligned} & d_0^2 \cdot (\alpha_{2,0} \cdot \beta + \alpha_{0,2} \cdot \gamma) + d_0 \cdot (\alpha_{2,0}^2 \cdot \beta \cdot d_{n-1} + \beta \cdot d_1 + \alpha_{0,2}^2 \cdot d_1 \cdot \gamma + d_{n-1} \cdot \gamma) \\ & + (\alpha_{2,0} \cdot \beta + \alpha_{0,2} \cdot \gamma) \cdot d_1 \cdot d_{n-1} = 0 \,, \end{aligned} \tag{16}$$

where

$$\beta := \prod_{i=1}^{n-2} (\alpha_{2,0} \cdot d_i + d_{i+1}) \qquad \text{and} \qquad \gamma := -(-1)^n \cdot \prod_{i=1}^{n-2} (d_i + \alpha_{0,2} \cdot d_{i+1}) \,.$$

By limiting ourselves to focus on $\alpha_{0,2} \cdot \alpha_{2,0} = 1$, note that

$$\beta = \prod_{i=1}^{n-2} \left(\frac{1}{\alpha_{0,2}} \cdot d_i + d_{i+1} \right) = \left(\frac{1}{\alpha_{0,2}} \right)^{n-2} \cdot \prod_{i=1}^{n-2} (d_i + \alpha_{0,2} \cdot d_{i+1}) = - \left(-\frac{1}{\alpha_{0,2}} \right)^{n-2} \cdot \gamma \,. \tag{17}$$

This implies that $\gamma = 0$ if and only if $\beta = 0$. Moreover, the coefficient $\alpha_{2,0} \cdot \beta + \alpha_{0,2} \cdot \gamma$ of d_0^2 in (16) can be re-written as

$$\alpha_{2,0} \cdot \beta + \alpha_{0,2} \cdot \gamma = \gamma \cdot \alpha_{0,2} \cdot \left(1 - \left(-\frac{1}{\alpha_{0,2}} \right)^n \right) \,,$$

which implies that

- if $(-\alpha_{0,2})^n \neq 1$, then the coefficient $\alpha_{2,0} \cdot \beta + \alpha_{0,2} \cdot \gamma$ of d_0^2 is equal to zero if and only if $\beta = \gamma = 0$;
- if $(-\alpha_{0,2})^n = 1$, then the coefficient of d_0^2 in Eq. (16) is always equal to zero.

Case: $(-\alpha_{0,2})^n \neq 1$. As we have just seen, the coefficient $\alpha_{2,0} \cdot \beta + \alpha_{0,2} \cdot \gamma$ of d_0^2 is equal to zero if and only if $\beta = \gamma = 0$. Working as in App. A.2, by choosing $d_1, \ldots, d_{n-1} \in \mathbb{F}_p$ such that $d_i \neq -\alpha_{2,0} \cdot d_{i-1}$ for each $i \in \{2, 3, \ldots, n-1\}$ (where e.g. d_1 is free to take any possible value), then $\beta, \gamma \neq 0$. In such a case, there are at most two values of d_0 that satisfies Eq. (16). In other words, there are $p \cdot (p-1)^{n-2} \cdot (p-2)$ different values of $d_0, d_1, \ldots, d_{n-1}$ for which the determinant is different from zero.

As pointed out in the previous paragraph, a collision can occur even if $n-1$ differences d_i are equal to zero (note that this case is obviously excluded from the previous one, since the determinant would be zero). If d_i is not null, s_{i-1}, s_{i+1} would be fixed, while all other s_j are free to take any possible. This is sufficient for concluding that the probability of having a collision is *at least* equal to

$$\frac{p \cdot (p-1)^{n-2} \cdot (p-2) + n \cdot (p-1) \cdot p^{n-2}}{p^n \cdot (p^n - 1)} = \frac{p \cdot (p-1) \cdot ((p-1)^{n-3} \cdot (p-2) + n \cdot p^{n-3})}{p^n \cdot (p^n - 1)} \,,$$

which is strictly bigger than $\frac{(p-1)^n}{p^n \cdot (p^n - 1)}$, that is, the probability given in Prop. 1.

Case: $(-\alpha_{0,2})^n = 1$. As we already pointed out, $\alpha_{2,0} \cdot \beta + \alpha_{0,2} \cdot \gamma$ is equal to zero in this case, which implies that Eq. (16) reduces to

$$d_0 \cdot (\alpha_{2,0}^2 \cdot \beta \cdot d_{n-1} + \beta \cdot d_1 + \alpha_{0,2}^2 \cdot d_1 \cdot \gamma + d_{n-1} \cdot \gamma) = 0 \,.$$

Since $\alpha_{0,2} \cdot \alpha_{2,0} = 1$ and since $\gamma = -\alpha_{2,0}^2 \cdot \beta$ due to Eq. (17), such equation is always satisfied for each d_0, d_1, β. It follows that the determinant is always equal to zero, and so that the rows of the r.h.s. vector in Eq. (15) must satisfy the same linear relation of the rows of the l.h.s. matrix. This implies that one difference d_i is fixed. W.l.o.g., we assume d_0 satisfies such linear relation. In such a case, a collision takes place if

$$\begin{bmatrix} \alpha_{2,0} \cdot d_1 + d_2 & d_1 + \alpha_{0,2} \cdot d_2 & 0 & \dots & 0 & 0 \\ 0 & \alpha_{2,0} \cdot d_2 + d_3 & d_2 + \alpha_{0,2} \cdot d_3 & \dots & 0 & 0 \\ \vdots & & \ddots & \ddots & & \vdots \\ 0 & 0 & 0 & \dots & \alpha_{2,0} \cdot d_{n-2} + d_{n-1} & d_{n-2} + \alpha_{0,2} \cdot d_{n-1} \\ 0 & 0 & 0 & \dots & 0 & \alpha_{2,0} \cdot d_{n-1} + d_0 \end{bmatrix}$$
$$\times \begin{bmatrix} s_1 & s_2 & \dots & s_{n-2} & s_{n-1} \end{bmatrix}^T =$$
$$- \begin{bmatrix} \alpha_{1,0} \cdot d_1 + \alpha_{0,1} \cdot d_2 & \dots & \alpha_{1,0} \cdot d_{n-2} + \alpha_{0,1} \cdot d_{n-1} & \alpha_{1,0} \cdot d_{n-1} + \alpha_{0,1} \cdot d_0 + s_0 \cdot (d_{n-1} + \alpha_{0,2} \cdot d_0) \end{bmatrix}^T ,$$

where s_0 can take any possible value in $\mathbb{F}_p$. The determinant of the l.h.s. matrix is equal to $\prod_{i=1}^{n-1}(\alpha_{2,0} \cdot d_i + d_{i+1})$. Since d_0 is fixed, there are $(p-1)^{n-1}$ values of $d_1, d_2, \dots, d_{n-1}$ for which such matrix is invertible, and so, for which a collision can occur. This is sufficient for concluding that the probability of having a collision is *at least* equal to

$$\frac{p \cdot (p-1)^{n-1}}{p^n \cdot (p^n - 1)} > \frac{(p-1)^n}{p^n \cdot (p^n - 1)} ,$$

which is strictly bigger than the probability given in Prop. 1.