Integer Functions Suitable for Homomorphic Encryption over Finite Fields

Ilia Iliashenko[1], Christophe Nègre[2,3], and Vincent Zucca[2,3]

[1] imec-COSIC, KU Leuven, Belgium
[2] DALI, Univ Perpignan Via Domitia, France
[3] LIRMM, Univ Montpellier, France
ilia@esat.kuleuven.be,
christophe.negre@{upvd.fr, lirmm.fr},
vincent.zucca@{upvd.fr, lirmm.fr}

Abstract. Fully Homomorphic Encryption (FHE) gives the ability to evaluate any function over encrypted data. However, despite numerous improvements during the last decade, the computational overhead caused by homomorphic computations is still very important. As a consequence, optimizing the way of performing the computations homomorphically remains fundamental. Several popular FHE schemes such as BGV and BFV encode their data, and thus perform their computations, in finite fields. In this work, we study and exploit algebraic relations occurring in prime characteristic allowing to speed-up the homomorphic evaluation of several functions over prime fields.
More specifically we give several examples of unary functions: "modulo", "is power of b" and "Hamming weight" and "Mod2" whose homomorphic evaluation complexity over $\mathbb{F}_p$ can be reduced from the generic bound $\sqrt{2p}+\mathcal{O}(\log(p))$ homomorphic multiplications, to $\sqrt{p}+\mathcal{O}(\log(p))$, $\mathcal{O}(\log(p))$, $\mathcal{O}(\sqrt{p/\log(p)})$ and $\mathcal{O}(\sqrt{p/(\log(p))})$ respectively. Additionally we provide a proof of a recent claim regarding the structure of the polynomial interpolation of the "less-than" bivariate function which confirms that this function can be evaluated in $2p-6$ homomorphic multiplications instead of $3p-5$ over $\mathbb{F}_p$ for $p \geq 5$.

1 Introduction

FHE allows to perform any kind of computations directly over encrypted data offering therefore natural solutions for privacy-preserving techniques. The first theoretical construction of FHE provided by Gentry in 2009 [10] has drawn an important attention from the cryptographic community which has resulted in numerous improvements in the following years. Although, the computational overhead of the first schemes was too important to consider using them in practice, the technology is now mature enough to be used in several practical scenarios such as genome analysis and is currently going through a standardization process. Nonetheless, the efficiency of the current schemes is not satisfactory yet and, as so, improving their efficiency is still an active research area.

Every FHE scheme follows Gentry's blueprint: a Somewhat Homomorphic Encryption (SHE) scheme allowing to perform a limited number of operations endowed with a *bootstrapping* procedure permitting to refresh the homomorphic capacity of a ciphertext. Current FHE schemes can be classified into three main categories depending on the kind of data they perform computations on. Schemes of the first category encode their data bit-wise and can evaluate boolean circuits efficiently [8,**?**]. The second category of schemes encodes their data word-wise as element of a finite field and can be used for efficient integer computations [1,**?**]. The third and last category of schemes supports computations directly over complex, and thus real, numbers but in an approximated manner [5].

Each category has its pros and cons and the choice of the scheme mainly depends on the targeted application. Schemes of the first category have a very efficient bootstrapping procedure but only operates at the bit level. On the other hand, schemes belonging to the second/third categories can operate directly on integers/complex numbers but are usually used as SHE schemes because their bootstrapping procedure is very slow. An interesting feature of the schemes of the second category

is that since their data are encoded as finite-field elements, computations on these data occur in prime characteristic which might simplify some computations.

1.1 Contributions

This work focuses on the study of specific functions which have a particular structure when interpolated over $\mathbb{F}_p$ for an odd prime p. This structure allows to speed-up their homomorphic evaluation by schemes of the second category making them more interesting for possible future applications.

Our first contribution is the study of several unary functions: "modulo", "is power of b", "Hamming weight" and "Mod2". The particular structure of these functions permits to reduce the complexity of their evaluation from the generic bound $\sqrt{2p} + \mathcal{O}(\log(p))$ homomorphic multiplications with the Paterson-Stockmeyer algorithm [17] to $\sqrt{p} + \mathcal{O}(\log p)$, $\mathcal{O}(\log(p))$, $\mathcal{O}(\sqrt{p/\log p})$ and $\mathcal{O}(\sqrt{p/\log p})$ respectively.

Our second contribution is the study of the interpolation polynomial of the bivariate less-than function: $x < y$ over $\mathbb{F}_p$. We prove the recent claim of Iliashenko and Zucca [14] that this polynomial has a structure which can be exploited to evaluate it using only $2p - 6$ homomorphic multiplications instead of $3p - 5$ by evaluating each monomial separately [19] when $p \geq 5$.

Note however that since our improvements do not affect the multiplicative depth required to evaluate the different functions, the parameters required to evaluate these functions homomorphically remain unchanged. As a consequence the size of the ciphertexts is not affected by our work. Nonetheless similarly to the work of [14], since homomorphic multiplication is by far the bottleneck of FHE/SHE schemes, we expect our improvements to result in speed-up proportional to the number of homomorphic multiplications saved. Eventually since our improvements only affect the number of homomorphic multiplications, the gain in practice will be agnostic to the chosen scheme.

1.2 Related art

The running time complexity of computing certain functions using homomorphic encryption is a versatile topic due to the variety of plaintext spaces used by homomorphic schemes.

In approximate homomorphic encryption [5], the plaintext space consists of complex numbers. Hence, non-arithmetic functions are approximated by complex- or real-valued polynomials. At the moment, there have been published numerous papers on evaluation of periodic functions such as modulo and sine functions (e.g. [4,2,13]). Recently, researchers focused on comparison operations including maximum/minimum and less-than functions [7,6].

For FHE/SHE schemes with a plaintext space $\mathbb{Z}_{p^e}$ where p is prime, the current studies are limited to digit removal polynomials [12,3], i.e. functions that remove the least significant digits of an input number in base p. In particular, Chen and Han showed that a digit removal polynomial has a surprisingly low degree at most $(e-1)(p-1)+1$.

For FHE/SHE scheme with a plaintext space $\mathbb{F}_p$, Kaji et al. [15] showed how to compute max and argmax functions. However, their result is not optimal as it requires quadratic number of ciphertext-ciphertext multiplications in p. Iliashenko and Zucca [14] improved this complexity by proving that the less-than and the maximum functions have a total degree p and can be computed in $\mathcal{O}(p)$ ciphertext-ciphertext multiplications. Furthermore, they demonstrated that this function can be computed in $\mathcal{O}(\sqrt{p})$ multiplications at the cost of less efficient plaintext encoding.

2 Preliminaries

2.1 Basic notation

Vectors are written column-wise and denoted by boldface lower-case letters. For some non-negative ℓ and k, we denote the set of integers $\{\ell, \ldots, k\}$ by $[\ell, k]$.

We denote the set of integer residue classes modulo an integer m by $\mathbb{Z}_m$. By default, we assume that the class representatives of $\mathbb{Z}_m$ are taken from the interval $[0, m-1]$. The modulo operation is denoted by $|\cdot|_m$, i.e. $a \mod m = |a|_m$.

For a non-negative integer a, we denote its binary representation by a_2. The Hamming weight of a_2 is denoted by $\mathrm{Hwt}(a)$.

2.2 Finite fields

Let $\mathbb{F}_p$ be a prime finite field where p is an odd prime. In this work, we will use the following standard facts about $\mathbb{F}_p$.

Lemma 1. *For any prime number $p > 2$ and $e \in [0, p-2]$, it holds*

$$\sum_{a \in \mathbb{F}_p} a^e = 0 \bmod p.$$

Proof. If $e = 0$, the lemma trivially holds. Let g be a primitive element of $\mathbb{F}_p$, i.e. $\mathbb{F}_p^\times = \langle g \rangle$. Thus, we can write

$$\sum_{a \in \mathbb{F}_p} a^e = \sum_{a \in \mathbb{F}_p^\times} a^e = \sum_{i=1}^{p-1} g^{ei}.$$

Since $p > 2$ and $e \in [1, p-2]$, we have $g^e \neq 1$. Thus,

$$\sum_{i=1}^{p-1} g^{ei} = \frac{(g^e)^p - g^e}{g^e - 1} = 0.$$

Lemma 2. *Any pair $(a, b) \in \mathbb{F}_p^2$ satisfies*

$$(a-b)^{p-1} = \sum_{i=0}^{p-1} a^i b^{p-1-i} \bmod p.$$

Proof. The binomial theorem yields

$$(a-b)^{p-1} = \sum_{i=0}^{p-1} \binom{p-1}{i} a^i (-b)^{p-1-i}.$$

Note that the binomial coefficient in this expression satisfies the following equality

$$\begin{aligned} \binom{p-1}{i} &= \frac{(p-1)!}{i!(p-1-i)!} \\ &= \frac{(p-1)(p-2)\ldots(i+1)}{1 \cdot 2 \ldots (p-(1+i))} \\ &= (-1)^{p-1-i} \mod p\,. \end{aligned}$$

Hence, $(a-b)^{p-1} = \sum_{i=0}^{p-1} a^i b^{p-1-i} \bmod p$.

2.3 Somewhat homomorphic encryption and its model of computation

An SHE scheme is an encryption scheme that can compute arithmetic circuits of bounded multiplicative depth on encrypted messages without knowing the secret key. The most efficient SHE schemes [9,1,11] are based on the LWE [18] and RLWE [16] problems that inject *noise* components into ciphertexts. This noise grows after homomorphic operations but must remain small enough to guarantee the correctness of decryption. Hence, the encryption parameters of an SHE scheme are defined not only by the targeted security level, but also by the family of circuits to be computed on ciphertexts.

The presence of noise implies that the complexity of homomorphic operations should be assessed with relation to their running time and the amount of noise they introduce. The noise growth depends on the parameters of LWE and RLWE, namely a ciphertext modulus q, a dimension n and a plaintext modulus p. This leads to a special model of computation similar to arithmetic circuits over the plaintext space of an SHE scheme. In this work, we focus on the cases where this plaintext space is a prime field $\mathbb{F}_p$.

The basic operations of our homomorphic model of computation over $\mathbb{F}_p$ are the field binary operations - addition and multiplication. Unlike in classic arithmetic circuits, these operations have different costs depending on whether their inputs are solely ciphertexts or plaintexts and ciphertexts.

Plaintext-ciphertext and ciphertext-ciphertext homomorphic addition are the simplest operations that take $\mathcal{O}(n)$ additions in $\mathbb{Z}_q$. The noise of an addition output is the sum of input noises plus a small $\mathcal{O}(p)$ factor.

Plaintext-ciphertext (or scalar) multiplication requires $\mathcal{O}(n)$ multiplications in $\mathbb{Z}_q$ and increases the noise of an input ciphertext by a factor of $\mathcal{O}(p)$.

The costliest homomorphic operation is ciphertext-ciphertext (or non-scalar) multiplication that takes $\mathcal{O}(n \log n + n \log q)$ multiplications in $\mathbb{Z}_q$. The output noise $\mathcal{O}(n \cdot p \cdot \max(E_1, E_2))$ where E_i is the noise of the ith input ciphertext.

Given the above complexities, the arithmetic circuits presented in this work are analyzed with relation to their non-scalar complexity, i.e. the number of non-scalar multiplications and their depth.

The main tool we exploit to estimate the non-scalar complexity of a given circuit is the following theorem due to Paterson and Stockmeyer [17].

Theorem 1 ([17]). *Any polynomial of degree d over a ring can be evaluated using $\mathcal{O}(\sqrt{d})$ non-scalar multiplications and $\lceil \log_2 d \rceil + 1$ multiplicative levels.*

Remark 1. If the ring in the above theorem is a finite field, the hidden constant in the above theorem is $\sqrt{2}$. Thus, approximately $\sqrt{2d}$ non-scalar multiplications are needed to compute a polynomial of degree d.

2.4 Interpolation over finite fields

Using Fermat little theorem, the equality function can be evaluated very simply over $\mathbb{F}_p^2$ as

$$EQ(x, y) = 1 - (x - y)^{p-1} = \begin{cases} 1 \text{ if } x = y \\ 0 \text{ otherwise.} \end{cases}.$$

This evaluation only requires $\mathcal{O}(\log(p))$ non-scalar multiplications using the square-and-multiply exponentiation algorithm.

A direct consequence of the above result is that any function defined over $\mathbb{F}_p^n$ can be interpolated by a polynomial according to the following well-known lemma.

Lemma 3. *Every function $f : \mathbb{F}_p^n \to \mathbb{F}_p$ is a polynomial function represented by a unique polynomial $P_f(X_1, \ldots, X_n)$ of degree at most $p-1$ in each variable. In particular,*

$$P_f(X_1, \ldots, X_n) = \sum_{\mathbf{a} \in \mathbb{F}_p^n} f(\mathbf{a}) \prod_{i=1}^{n} \left(1 - (X_i - a_i)^{p-1}\right) .$$

where a_i is the ith coordinate of vector $\mathbf{a}$.

3 Unary functions

Let $f : \mathbb{F}_p \to \mathbb{F}_p$ be a function. Using Lemma 2, one can check that the interpolation polynomial from Lemma 3 turns into

$$\begin{aligned} P_f(X) &= \sum_{a=0}^{p-1} f(a)(1 - (X-a)^{p-1}) && (1)\\ &= \sum_{a=0}^{p-1} f(a) \left(1 - \sum_{i=0}^{p-1} X^i a^{p-1-i}\right) \\ &= \sum_{a=0}^{p-1} f(a) - \sum_{a=0}^{p-1} \sum_{i=0}^{p-1} f(a) X^i a^{p-1-i} \\ &= \sum_{a=0}^{p-1} f(a) - \sum_{a=0}^{p-1} f(a) a^{p-1} - \sum_{i=1}^{p-1} \sum_{a=0}^{p-1} f(a) X^i a^{p-1-i} \\ &= f(0) - \sum_{i=1}^{p-1} X^i \sum_{a=0}^{p-1} f(a) a^{p-1-i}. && (2) \end{aligned}$$

From these two representations, we obtain that the non-scalar complexity of evaluating $P_f(X)$ is equal either to $\mathcal{O}(|\operatorname{supp}(f)| \cdot \log(p-1))$ (Eq. 1) or at most $\mathcal{O}(\sqrt{p-1})$ (Eq. 2) if the Paterson-Stockmeyer algorithm is used (Theorem 1).

For example, consider the function $f(x)$ that returns 1 only if $x = 2$. Hence, it has the following interpolation polynomial with two representations $P_f(X) = 1 - (X-2)^{p-1} = -\sum_{i=1}^{p-1} X^i 2^{p-1-i}$. Since $\operatorname{supp}(f)$ contains only one element, the former representation is so simple and results in a $\mathcal{O}(\log(p-1))$ non-scalar complexity. The latter representation can be computed with the Paterson-Stockmeyer algorithm that results in a worse non-scalar complexity in $\mathcal{O}(\sqrt{p-1})$.

We are interested in functions f with a support of size $\omega(1)$ that have a non-scalar complexity better than $\mathcal{O}(|\operatorname{supp}(f)| \cdot \log(p-1))$ and less than $\sqrt{2(p-1)}$ non-scalar multiplications. In particular, we focus on functions f that have many zero coefficients in $P_f(X)$. Thus, we need to study under what circumstances $\sum_{a=0}^{p-1} f(a) a^{p-1-i}$ is zero.

General observations. From Lemma 1, we know that $\sum_{a=0}^{p-1} a^{p-1-i} = 0$ for $i \neq 0$. It implies the trivial assertion that if f is constant, all the coefficients of $P_f(X)$ will be zero except for the constant term. More peculiar is the following property.

Lemma 4. *Let $\mathbb{F}_p$ be a prime finite field. Let γ be a primitve kth root of unity with $k > 0$ dividing $p-1$. Let $\mathcal{S}_0, \mathcal{S}_1, \ldots, \mathcal{S}_{k-1}$ be k disjoint subsets of $\mathbb{F}_p^\times$ satisfying*

$$\begin{aligned} \mathcal{S}_j &= \gamma^j \mathcal{S}_0 \text{ for } 0 \leq j < k\,, \\ \mathbb{F}_p^\times &= \mathcal{S}_0 \cup \mathcal{S}_1 \cup \cdots \cup \mathcal{S}_{k-1}. \end{aligned}$$

Then the following statements are true for any $i \in [1, p-2]$ such that $k | i$.

1.

$$\sum_{a\in\mathcal{S}_0} a^{p-1-i} = 0$$

2. If f is a function constant on each S_j, namely

$$f(a) = c_j \text{ for all } a \in \mathcal{S}_j,$$

then the coefficients of $P_f(X)$ of degree i are all zeros.

Proof. 1. Since $k|p-1$ and $k|i$, then for each $j \in [0, k-1]$ we have $(\gamma^j)^{p-1-i} = 1$.
Thus, we can write

$$\begin{aligned} k \cdot \textstyle\sum_{a\in\mathcal{S}_0} a^{p-1-i} &= \textstyle\sum_{a\in\mathcal{S}_0} a^{p-1-i} + \sum_{a\in\mathcal{S}_0} a^{p-1-i} + \cdots \\ &\qquad \cdots + \textstyle\sum_{a\in\mathcal{S}_0} a^{p-1-i} \\ &= \textstyle\sum_{a\in\mathcal{S}_0} a^{p-1-i} + \sum_{a\in\mathcal{S}_0} \gamma^{p-1-i} a^{p-1-i} + \cdots \\ &\qquad + \textstyle\sum_{a\in\mathcal{S}_0} (\gamma^{k-1})^{p-1-i} a^{p-1-i} \\ &= \textstyle\sum_{a\in\mathcal{S}_0} a^{p-1-i} + \sum_{a\in\mathcal{S}_1} a^{p-1-i} + \cdots \\ &\qquad + \textstyle\sum_{a\in\mathcal{S}_{k-1}} a^{p-1-i} \\ &= \textstyle\sum_{a=1}^{p-1} a^{p-1-i} = 0 \end{aligned}$$

The last equality holds due to Lemma 1 and since $p-1-i \in [1, p-2]$. This leads to $\sum_{a\in\mathcal{S}_0} a^{p-1-i} = 0$ as $k \neq 0 \mod p$.

2. From Equation (2), the ith coefficient of $P_f(X)$ satisfies the following

$$\begin{aligned} \textstyle\sum_{a=0}^{p-1} f(a) a^{p-1-i} &= \textstyle\sum_{j=0}^{k-1} c_j \sum_{a\in S_j} a^{p-1-i} \\ &= \textstyle\sum_{j=0}^{k-1} c_j \sum_{a\in S_0} (\gamma^j a)^{p-1-i} \\ &= (c_0 + c_1 + \cdots + c_{k-1}) \textstyle\sum_{a\in S_0} a^{p-1-i} \\ &= 0 \bmod p. \end{aligned}$$

Hence, the ith coefficient is zero.

Example 1. Consider, $\mathcal{S}_0 = [1, (p-1)/2]$ or $\mathcal{S}_0 = \{a \text{ is even}, a \in [2, p-1]\}$ and $k = 2$. The above lemma holds for the following functions

- the parity function $|x|_2$ that returns 1 on odd numbers from $[0, p-1]$ and 0 on even numbers (note that if a is odd, then $p - a$ is even for any odd prime p);
- $x < 0$ returns 1 if $x \in [-(p-1)/2, -1]$ and 0 otherwise.

Lemma 4 states that $P_f(X)$ of any of the above functions has only odd coefficients plus the constant and the leading terms. Hence, it can be presented in the form $P_f(X) = f_0 + f_{p-1}X^{p-1} + Xg(X^2)$ where $g(X)$ is a polynomial of degree $(p-3)/2$. Hence, $P_f(X)$ can be computed in approximately $\sqrt{p-3}$ non-scalar multiplications using the Paterson-Stockmeyer algorithm.

3.1 Modulo functions

Let us consider a function $f_m(x) = |x|_m$. This function has m outputs $\{0, 1, \ldots, m-1\}$, which implies that $\mathbb{F}_p$ splits into m subsets $\mathcal{S}_0, \mathcal{S}_1, \ldots, \mathcal{S}_{m-1}$ such that $f(a) = k$ for every $a \in S_k$. Then, the ith coefficient of the interpolation polynomial of f is equal to

$$\sum_{a\in\mathcal{S}_1} a^{p-1-i} + 2\sum_{a\in\mathcal{S}_2} a^{p-1-i} + \cdots + (m-1)\sum_{a\in\mathcal{S}_{m-1}} a^{p-1-i}. \tag{3}$$

If $p \equiv m-1 \mod m$ and $|a|_m = k$, then $|p-a|_m = m-1-k$. This is equivalent to the fact that if $a \in S_k$, then $p-a \in S_{m-1-k}$. Thus, it follows for any even $i \notin \{0, p-1\}$

$$\sum_{a \in \mathcal{S}_k} a^{p-1-i} = \sum_{a \in \mathcal{S}_{m-1-k}} a^{p-1-i}.$$

As a result, we obtain

$$\begin{aligned} k \sum_{a \in \mathcal{S}_k} a^{p-1-i} + (m-1-k) \sum_{a \in \mathcal{S}_{m-1-k}} a^{p-1-i} & \\ = (m-1) \sum_{a \in \mathcal{S}_k} a^{p-1-i} & \\ = (m-1) \sum_{a \in \mathcal{S}_{m-1-k}} a^{p-1-i} & \quad (4) \\ = \left\lfloor \frac{m-1}{2} \right\rfloor \sum_{a \in \mathcal{S}_k} a^{p-1-i} + \left\lceil \frac{m-1}{2} \right\rceil \sum_{a \in \mathcal{S}_{m-1-k}} a^{p-1-i} & \quad (5) \end{aligned}$$

If m is odd, then using (5) we can rewrite Eq. 3 as follows

$$\begin{aligned} \frac{m-1}{2} \left(\sum_{a \in \mathcal{S}_0} a^{p-1-i} + \sum_{a \in \mathcal{S}_1} a^{p-1-i} + \cdots + \sum_{a \in \mathcal{S}_{m-1}} a^{p-1-i} \right) & \\ = \frac{m-1}{2} \sum_{a \in \mathbb{F}_p} a^{p-1-i} = 0. & \end{aligned}$$

If m is even, then for every odd k we have even $m-1-k$. Using (4), we obtain from Eq. 3 the following expression

$$\begin{aligned} \sum_{a \in \mathcal{S}_1} a^{p-1-i} + 2 \sum_{a \in \mathcal{S}_2} a^{p-1-i} + \cdots + (m-1) \sum_{a \in \mathcal{S}_{m-1}} a^{p-1-i} = & \\ \sum_{a \in \mathcal{S}_{m-2}} a^{p-1-i} + 2 \sum_{a \in \mathcal{S}_2} a^{p-1-i} \cdots + (m-3) \sum_{a \in \mathcal{S}_2} a^{p-1-i} & \\ + (m-2) \sum_{a \in \mathcal{S}_{m-2}} a^{p-1-i} + (m-1) \sum_{a \in \mathcal{S}_0} a^{p-1-i} = & \\ (m-1) \left(\sum_{a \in \mathcal{S}_0} a^{p-1-i} + \sum_{a \in \mathcal{S}_2} a^{p-1-i} + \cdots + \sum_{a \in \mathcal{S}_{m-2}} a^{p-1-i} \right). & \end{aligned}$$

Note that if $a \in [0, p-1]$ and m are even, then $|a|_m$ is also even. It implies that $\cup_{i=0}^{m/2-1} \mathcal{S}_{2i} = \mathcal{S} = \{a \text{ is even}, a \in [2, p-1]\}$ and the above expression is equal to

$$(m-1) \sum_{a \in \mathcal{S}} a^{p-1-i},$$

which is zero as shown in Example 1.

If $i = p-1$, then

$$\begin{aligned}\sum_{a=0}^{p-1} |a|_m \, a^{p-1-i} &= \sum_{a=0}^{p-1} |a|_m = \sum_{a=0}^{p} |a|_m - |p|_m \\ &= \frac{p+1}{m} \cdot \frac{m(m-1)}{2} - (m-1) \\ &= \frac{(p+1)(m-1)}{2} - (m-1) \\ &= \frac{(p-1)(m-1)}{2}\end{aligned}$$

Therefore we have proved the following

Proposition 1. *Let $m > 1$ be an integer and p an odd prime such that $p \equiv m-1 \bmod m$. The interpolation polynomial of the modulo m function f_m over $\mathbb{F}_p$ is equal to*

$$P_{f_m}(X) = \frac{(p+1)(m-1)}{2} X^{p-1} - \sum_{i=1}^{\frac{p-1}{2}} X^{2i-1} \sum_{a=1}^{p-1} [a]_m \, a^{p-2i}.$$

Remark 2. Similarly to Example 1, this polynomial has only the leading term and odd coefficients which are non-zero. Thus, it can be computed with approximately $\sqrt{p-3}$ non-scalar multiplications due to Remark 1.

Example 2. For $p = 11$ and $m = 6$ we have $P_{f_m}(X) = 8X^{10} + 9X^9 + 3X^7 + 4X^5 + 4X^3 + 6X$.

3.2 "Is power of b" functions and all-one polynomials

Let b be an integer bigger than 1. We define a function $f_b(x)$ on the set $[0, p-1]$ that outputs 1 if its input is a power of b and 0 otherwise. Let $\ell = \lfloor \log_b p \rfloor$. Using the interpolation formula (1), we obtain that this function is defined over $\mathbb{F}_p$ by the following polynomial

$$P_{f_b}(X) = \sum_{a=0}^{\ell} (1 - (X - b^a)^{p-1}),$$

which can be computed in $O(\ell \log(p-1)) = O((\log p)^2)$ non-scalar multiplications using the square-and-multiply method. However, a logarithmic evaluation complexity can be achieved if $p = (b^r - 1)/k$ for some integers $k < b$ and an integer $r \geq 1$.

Let p be such that $b^{\ell+1} = 1 \bmod p$. Notice that since $b^i < p$ for every $0 \leq i \leq \ell$, $\ell + 1$ is the order of b modulo p. Therefore, since $\ell + 1 \mid p-1$, $p - 1 - i \equiv 0 \ [\ell+1]$ is equivalent to $i \equiv 0 \ [\ell+1]$.

Now, from (2), we obtain

$$P_{f_b}(X) = -\sum_{i=1}^{p-1} X^i \sum_{a=0}^{\ell} (b^a)^{p-1-i}.$$

- if $i \equiv 0 \ [\ell+1]$, then $p - 1 - i \equiv 0 \ [\ell+1]$ and therefore the ith coefficient of P_{f_b} is equal to $-(\ell+1) = p - (\ell+1) \bmod p$.
- if $i \not\equiv 0 \ [\ell+1]$ then $p - 1 - i \not\equiv 0 \ [\ell+1]$ and therefore the ith coefficient of P_{f_b} is equal to

$$-\sum_{a=0}^{\ell} (b^a)^{p-1-i} = -\frac{b^{(p-1-i)(\ell+1)} - 1}{b^{p-1-i} - 1} = 0 \bmod p.$$

Eventually, notice that since $b^\ell < p$, then $b^{\ell+1} = 1 + kp < pb$ and therefore $p(k-b) < -1$, which implies that $k < b$. Reciprocally if $p = (b^r - 1)/k$ for some integers $r \geq 1$ and $k < b$, then $b^{r-1} < 1/b + p$ and thus $b^{r-1} < p < b^r$ therefore $r = \lceil \log_b(p) \rceil$ is the order of b modulo p. Overall, we have obtained the following result

Proposition 2. *If $p = (b^r - 1)/k$ for some integer $k < b$ and $r \geq 1$ with $k < b$, then*

$$P_{f_b}(X) = (p-r) \sum_{i=1}^{(p-1)/(\ell+1)} X^{i(\ell+1)}.$$

Example 3. $p = 31$ and $b = 2$ we have $p = (b^5 - 1)/1$ and

$$P_{f_b}(X) = 26\left(X^{30} + X^{25} + X^{20} + X^{15} + X^{10} + X^5\right)$$

To simplify the non-scalar complexity analysis, we denote $Y = X^r$ and $e = (p-1)/r$. Now, to find the non-scalar complexity of evaluating $P_{f_b}(X)$, we should count the number of non-scalar multiplications to evaluate the all-one polynomial $\sum_{i=1}^{e} Y^i$. Using the ideas of the Paterson-Stockmeyer method [17], we design the following evaluation scheme. Given an input $y \in \mathbb{F}_p$, we precompute the powers $y^2, \ldots, y^{2^k}$ where $k = \lfloor \log_2 e \rfloor$. This step requires k non-scalar multiplications. The last power is computed with multiplicative depth k. Then, we compute the following k products

$$\begin{aligned} S_1 &= (y + y^2)(1 + y^2), \\ S_2 &= (y + y^2)(1 + y^2)(1 + y^4), \\ &\ldots \\ S_{k-1} &= (y + y^2) \prod_{i=1}^{k-1} (1 + y^{2^i}) \end{aligned}$$

that can be done with $k-1$ non-scalar multiplications. The multiplicative depth of computing S_i from y is equal to $i+1$. Notice that $S_i = \sum_{j=1}^{2^{i+1}} y^j$ for any $i \in [1, k-1]$. To compute $\sum_{i=1}^{e} y^i$ with $e \geq 4$, we split it into two parts as follows

$$\begin{aligned} \sum_{i=1}^{e} y^i &= y + y^2 + \cdots + y^{2^k} + y^{2^k}(y + y^2 + \ldots y^{e-2^k}) \\ &= S_{k-1} + y^{2^k} \left(\sum_{i=1}^{e-2^k} y^i \right) \end{aligned} \tag{6}$$

If $e - 2^k = 0$, then we assume $\sum_{i=1}^{e-2^k} y^i = 0$. Since S_{k-1} and y^{2^k} are precomputed, the non-scalar complexity of computing an all-one polynomial of degree e is equal to the non-scalar complexity of computing an all-one polynomial of degree $e - 2^{\lfloor \log_2 e \rfloor}$ (i.e. e without its top bit) plus one non-scalar multiplication. If $e < 4$, then computing $\sum_{i=1}^{e} y^i$ requires at most one non-scalar product to compute y^3. As a result, at most $\mathrm{Hwt}(e) - 1$ non-scalar multiplications are necessary to compute (6) recursively. Together with the cost of precomputation done using the square-and-multiply method, we obtain that $P_{f_b}(X)$ can be computed in

$$\lfloor \log_2 r \rfloor + \mathrm{Hwt}(r) - 1 + 2k + \mathrm{Hwt}(e) - 2 \in \mathcal{O}(\log r + \log e)$$

non-scalar multiplications. Since $e = (p-1)/r$, this complexity turns into $\mathcal{O}(\log p - 1)$ non-scalar multiplications. The non-scalar multiplicative depth is $\lceil \log_2 e \rceil + \lceil \log_2 r \rceil$.

Remark 3. Similar non-scalar complexity of evaluating $P_{f_b}(X)$ can be obtained by a circuit of multiplicative depth $\lceil \log_2 \ell + 1 \rceil + \lceil \log_2(p-1) \rceil$, see Appendix A.1.

We can also generalize f_b for any b that generates a subgroup in $\mathbb{F}_p^\times$. See Appendix A.1 for more details.

3.3 Hamming weight functions

Recall that the Hamming weight $\mathrm{Hwt}(a)$ of an integer a is the Hamming weight of its binary decomposition. Obviously, the function Hwt is defined on a set $[0, p-1]$ with p prime. Its interpolation polynomial over $\mathbb{F}_p$ is equal to

$$P_{\mathrm{Hwt}}(X) = -\sum_{i=1}^{p-1} X^i \sum_{a=0}^{p-1} \mathrm{Hwt}(a) a^{p-1-i}.$$

As above, let us focus on the ith coefficient of this expression

$$\begin{aligned}\sum_{a=0}^{p-1} \mathrm{Hwt}(a) \cdot a^{p-1-i} &= \sum_{a=1}^{p-1} \mathrm{Hwt}(a) \cdot a^{p-1-i} \\ &= 1^{p-1-i} + 2^{p-1-i} + \cdots + \mathrm{Hwt}(p-1) \cdot (p-1)^{p-1-i}. \end{aligned} \quad (7)$$

Assume that p is a Mersenne prime, i.e. $p = 2^q - 1$ for some prime integer q. This means that 2 has order q, which implies that q divides $p-1$. Let G be a subgroup of $\mathbb{F}_p^\times$ generated by 2 and $H = \mathbb{F}_p^\times / G$.

Note that any integer a can be represented as $a = 2^r \cdot s$ with non-negative r and odd s. Moreover, $\mathrm{Hwt}(a) = \mathrm{Hwt}(s)$ and $\mathrm{Hwt}(a) = \mathrm{Hwt}(a \cdot 2^k \mod p)$ for any positive k. This implies that for any $g \in G$, it holds $\mathrm{Hwt}(gh) = \mathrm{Hwt}(h)$ for any $h \in H$. Then Eq. (7) can be rewritten as follows

$$\sum_{h \in H} \sum_{g \in G} \mathrm{Hwt}(hg) \cdot (hg)^{p-1-i} = \sum_{h \in H} \mathrm{Hwt}(h) \cdot h^{p-1-i} \sum_{j=0}^{q-1} 2^{j(p-1-i)}$$

If $2^{p-1-i} \neq 1 \mod p$, i.e q does not divide $p-1-i$, we can write $\sum_{j=0}^{q-1} 2^{j(p-1-i)} = \frac{2^{q(p-1-i)}-1}{2^{p-1-i}-1} \equiv 0 \mod p$. Thus, the ith coefficient (7) is zero.

To find other zero coefficients, we use the fact that $\mathrm{Hwt}(a) = q - \mathrm{Hwt}(p-a)$ since p is a Mersenne prime. If i is even and $i \neq 0 \mod (p-1)$, then the ith coefficient is equal to

$$\sum_{a=0}^{p-1} \mathrm{Hwt}(a) \cdot a^{p-1-i} = q \sum_{a=1}^{\frac{p-1}{2}} a^{p-1-i} = 0.$$

If $i = p-1$, then

$$\sum_{a=0}^{p-1} \mathrm{Hwt}(a) \cdot a^{p-1-i} = \sum_{a=1}^{p-1} \mathrm{Hwt}(a) = \sum_{a=1}^{\frac{p-1}{2}} q = \frac{q(p-1)}{2}.$$

To summarize, we have the following result

Proposition 3. *The interpolation polynomial of* $\mathrm{Hwt}(x)$ *modulo a Mersenne prime* $p = 2^q - 1$ *is equal to*

$$P_{\mathrm{Hwt}}(X) = \frac{q(p+1)}{2} X^{p-1} - \sum_{i=1}^{\frac{p-1}{2q}} X^{(2i-1)q} \sum_{a=1}^{p-1} \mathrm{Hwt}(a) \cdot a^{p-1-q(2i-1)}.$$

To evaluate this polynomial at x, we can precompute x^q in $\mathcal{O}(\log q)$ non-scalar multiplications. By replacing X^q by Y, we obtain that $P_{\mathrm{Hwt}}(Y) = \frac{q(p+1)}{2} Y^{\frac{p-1}{q}} + Y g(Y^2)$ where g has degree $\frac{p-1}{2q} - 1$. Hence, the evaluation of $P_{\mathrm{Hwt}}(X)$ takes

$$\mathcal{O}\left(\log q + \sqrt{\frac{p-1}{q} - 2}\right) = \mathcal{O}\left(\sqrt{\frac{p-1}{\log(p-1)} - 2}\right)$$

non-scalar multiplications using both the square-and-multiply exponentiation method and the Paterson-Stockmeyer algorithm.

Example 4. For $p = 31$, we have $P_{\mathrm{Hwt}}(X) = 18X^{30} + 22X^{25} + 15X^{15} + 8X^5$. If $p = 127$, then $P_{\mathrm{Hwt}}(X) = 67X^{126} + 63X^{119} + 65X^{105} + 37X^{91} + 113X^{77} + 35X^{63} + 58X^{49} + 64X^{35} + 90X^{21} + 44X^7$.

3.4 Mod2 function

We assume that p is a Mersenne prime $p = 2^q - 1$ with q prime. We consider the Mod2 function which, given $x \in [0, p-1]$ with binary expression $x = (x_{q-1}, \ldots, x_0)_2$, outputs

$$\mathrm{Mod2}(x) = \left(\bigoplus_{i=0}^{q-1} x_i\right) \oplus 1.$$

From (2) the interpolation polynomial $P_{\mathrm{Mod2}}(X)$ of the Mod2 function is as follows

$$\begin{aligned} P_{\mathrm{Mod2}}(X) &= 1 - \textstyle\sum_{i=1}^{p-1} X^i \sum_{a=0}^{p-1} \mathrm{Mod2}(a) a^{p-1-i} \\ &= 1 - X^{p-1}\left(\textstyle\sum_{a=0}^{p-1} \mathrm{Mod2}(a)\right) - \sum_{i=1}^{p-2} X^i \sum_{a=0}^{p-1} \mathrm{Mod2}(a) a^{p-1-i} \\ &= 1 - \tfrac{p+1}{2} X^{p-1} - \textstyle\sum_{i=1}^{p-2} X^i \sum_{a=0}^{p-1} \mathrm{Mod2}(a) a^{p-1-i}. \end{aligned} \tag{8}$$

Then using the property $\mathrm{Mod2}(2^k x) = \mathrm{Mod2}(x)$ for any x and any k, we can arrange the right most sum on a in (8). We split the sum using the expression of $x = h \times 2^i$ for some i and $h \in H = \mathbb{F}_p^\times / G$. We obtain the followings which shows that the coefficients of degree i for $i \not\equiv (p-1) \mod q = (2^q - 2) \mod q = 0 \mod q$ are equal to zero:

$$\begin{aligned} \textstyle\sum_{a=0}^{p-1} \mathrm{Mod2}(a) a^{p-1} &= \textstyle\sum_{h \in H} \sum_{k=0}^{q-1} \mathrm{Mod2}(2^k h)(2^k h)^{p-1-i} \\ &= \textstyle\sum_{h \in H} \sum_{k=0}^{q-1} \mathrm{Mod2}(h)(2^{k(p-1-i)})(h)^{p-1-i} \\ &= \textstyle\sum_{h \in H} \mathrm{Mod2}(h)(h)^{p-1-i} \left(\sum_{k=0}^{q-1} 2^{k(p-1-i)}\right) \\ &= \textstyle\sum_{h \in H} \mathrm{Mod2}(h)(h)^{p-1-i} \frac{(2^q)^{(p-1-i)} - 1}{2^{p-1-i} - 1} \\ &= 0. \end{aligned} \tag{9}$$

The last equality comes from $2^q \equiv 1 \mod p$.

Now we consider the sets $\mathcal{S}_0 = \{x \in \mathbb{F}_p^\times \text{ s.t. } \mathrm{Mod2}(x) = 0\}$ and $\mathcal{S}_1 = \{x \in \mathbb{F}_p^* \text{ s.t. } \mathrm{Mod2}(x) = 1\}$ which are disjoints and satisfy $\mathcal{S}_0 \cup \mathcal{S}_1 = \mathbb{F}_p^*$. If we denote $\gamma = -1$, the Mod2 function satisfies:

$$\mathrm{Mod2}(\gamma x) = \mathrm{Mod2}(p - x) = \mathrm{Mod2}(2^q - 1 - x) = 1 - \mathrm{Mod2}(x).$$

This implies that $\gamma \mathcal{S}_0 = \mathcal{S}_1$ and we can then apply Lemma 4 which tells us that the coefficients of $P_{\mathrm{Mod2}}(X)$ are zero for degree $i \equiv (p-1) \mod 2 = 0 \mod 2$.

Then the coefficients p_i with $i \notin \{0, p-1\}$ of P_{Mod2} are non-zero if $i \equiv 0 \mod q$ and $i \equiv 1 \mod 2$, which yields $i \equiv q \mod 2q$. Therefore we have the following result

Proposition 4. *Let $p = 2^q - 1$ be a Mersenne prime, the interpolation polynomial of the* Mod2 *function modulo p is given by*

$$P_{Mod2}(X) = 1 - \frac{p+1}{2}X^{p-1} + \sum_{j=0}^{(p-1)/2q} p_j X^{q(2j+1)}.$$

Evaluating P_{Mod2} at x requires $\mathcal{O}(\log(p)) + \mathcal{O}(\sqrt{(p-1)/q}) = \mathcal{O}(\sqrt{(p-1)/q}) = \mathcal{O}(\sqrt{p/\log p})$ non-scalar multiplications.

Example 5. For $p = 2^{127} - 1$ we have the following polynomial for Mod2:

$$\begin{aligned} P_{\text{Mod2}}(X) = 63X^{126} + 107X^{119} + 14X^{105} + 75X^{91} + 72X^{77} + 35X^{63} \\ +72X^{49} + 75X^{35} + 14X^{21} + 107X^7 + 1. \end{aligned}$$

4 Less than Function

Let $\mathcal{S}$ be a subset of $[0, p-1) \hookrightarrow \mathbb{F}_p$, the less than function $\texttt{LT}_\mathcal{S}$ is defined over $\mathcal{S}^2$ as

$$\texttt{LT}_\mathcal{S}(x, y) = \begin{cases} 1 \text{ if } x < y \\ 0 \text{ otherwise} \end{cases}$$

While the equality function can be computed very efficiently over finite fields, the less than function is more intricate. Considering $\mathcal{S} = [0, p-1)$, the interpolation polynomial of $\texttt{LT}_\mathcal{S}$ over $\mathcal{S}^2$ is equal to

$$P_{\texttt{LT}_\mathcal{S}}(X, Y) = \sum_{a=0}^{p-2} \left(1 - (X-a)^{p-1}\right) \sum_{b=a+1}^{p-1} \left(1 - (Y-b)^{p-1}\right).$$

It was shown in [14] that the total degree of $P_{\texttt{LT}_\mathcal{S}}(X, Y)$ is only p. The coefficients of the polynomial can be described more precisely by the following theorem.

Theorem 2 ([14]). *Let $p > 2$ be a prime number and $\mathcal{S} = [0, p-1]$, then the interpolation polynomial of $\texttt{LT}_\mathcal{S}$ over $\mathbb{F}_p$ has the following form*

$$P_{\texttt{LT}_\mathcal{S}}(X, Y) = Y^{p-1} - \frac{p-1}{2}(XY)^{\frac{p-1}{2}} + \sum_{\substack{i,j>0, \\ i\neq j, \\ i+j\leq p}} a_{ij} X^i Y^j$$

where $a_{ij} = \sum_{a=0}^{p-2} \sum_{b=a+1}^{p-1} a^{p-1-i} b^{p-1-j} \in \mathbb{F}_p$. The total degree of $P_{\texttt{LT}_\mathcal{S}}(X, Y)$ is p.

The following results were used in [14] but their proof were omitted due to space restriction. In this section we provide a proof of these results. The polynomial $\texttt{LT}_\mathcal{S}$ is composed of several factors given by the following lemma.

Lemma 5. *There exists a polynomial $f \in \mathbb{F}_p[X, Y]$ of total degree $p-3$ such that:*

$$P_{\texttt{LT}_\mathcal{S}}(X, Y) = Y(X-Y)(X+1)f(X, Y). \tag{10}$$

Proof. From the definition of $\mathtt{LT}_{\mathcal{S}}$ it is straightforward to notice that $P_{\mathtt{LT}_S}(X,0) = P_{\mathtt{LT}_S}(p-1,Y) = 0 \bmod p$ and thus Y and $X+1$ both divide $P_{\mathtt{LT}_S}(X,Y)$.

Now let us consider $P_{\mathtt{LT}_S}(X,X)$. For any $x \in \mathbb{F}_p$ we have $P_{\mathtt{LT}_S}(x,x) = 0$, which means $P_{\mathtt{LT}_S}(X,X)$ has p distinct roots. However, it follows from Theorem 2 that $P_{\mathtt{LT}_S}(X,X)$ could be of degree p. Let us show that it is actually of degree $p-1$, Theorem 2 states that

$$P_{\mathtt{LT}_S}(X,X) = X^{p-1} - \frac{p-1}{2}X^{p-1} + \sum_{\substack{i,j>0,\\ i\neq j,\\ i+j\leq p}} a_{i,j}X^{i+j}$$

So the coefficient of degree p is given by

$$\begin{aligned}
&\sum_{\substack{i,j>0,\\ i\neq j,\\ i+j=p}} \sum_{a=0}^{p-2} a^{p-1-i} \sum_{b=a+1}^{p-1} b^{p-1-j} = \sum_{i=1}^{p-1}\sum_{a=0}^{p-2} a^{p-1-i} \sum_{b=a+1}^{p-1} b^{i-1} \\
=&\sum_{a=0}^{p-2}\sum_{b=a+1}^{p-1}\sum_{i=1}^{p-1} a^{p-1-i}b^{i-1} = \sum_{a=0}^{p-2}\sum_{b=a+1}^{p-1} b^{p-2}\sum_{i=1}^{p-1} a^{p-1-i}b^{i+1-p} \\
=&\sum_{a=0}^{p-2}\sum_{b=a+1}^{p-1} b^{p-2}\sum_{i=1}^{p-1}(ab^{-1})^{p-1-i} = \sum_{a=0}^{p-2}\sum_{b=a+1}^{p-1} b^{p-2}\sum_{i=0}^{p-2}(ab^{-1})^{i} \\
=&\sum_{b=1}^{p-1} b^{p-2} + \sum_{a=1}^{p-2}\sum_{b=a+1}^{p-1} b^{p-2}\sum_{i=0}^{p-2}(ab^{-1})^{i} \\
=&0 + \sum_{a=1}^{p-2}\sum_{b=a+1}^{p-1} b^{p-2}\frac{(ab^{-1})^{p-1}-1}{ab^{-1}-1} = 0 \bmod p.
\end{aligned}$$

Therefore $P_{\mathtt{LT}_S}(X,X)$ is a polynomial of degree $p-1$ which has p distinct roots, thus it must be equal to 0 and so $X-Y$ divides $P_{\mathtt{LT}_S}(X,Y)$. Since $\mathbb{F}_p[X,Y]$ is a unique factorization domain and Y, $X+1$ and $X-Y$ are distinct irreducible elements, there exists $f(X,Y) \in \mathbb{F}_p[X,Y]$ of total degree $p-3$ such that

$$P_{\mathtt{LT}_{\mathcal{S}}}(X,Y) = Y(X-Y)(X+1)f(X,Y).$$

The following theorem describes the structure of $f(X,Y)$.

Theorem 3. *Let p be an odd prime and $\mathcal{S} = [0, p-1]$. Let $P_{\mathtt{LT}_{\mathcal{S}}}(X,Y)$ be the interpolation polynomial of $\mathtt{LT}_{\mathcal{S}}$ over $\mathbb{F}_p$ and $f(X,Y)$ such that $P_{\mathtt{LT}_{\mathcal{S}}}(X,Y) = Y(X-Y)(X+1)f(X,Y)$. We have*

$$f(X,X) = f(X,0) = f(p-1,X). \tag{11}$$

As a consequence, there exists $(p-1)/2$ polynomials $f_n(X)$ over $\mathbb{F}_p$, $0 \leq n \leq (p-3)/2$, such that:

$$f(X,Y) = \sum_{n=0}^{(p-3)/2} f_n(X)Z^n, \tag{12}$$

with $Z = Y(X-Y)$ and $\deg(f_n(X)) = p-3-2n$, or equivalently $(p-1)/2$ polynomials $f'_n(Y)$ over $\mathbb{F}_p$, $0 \leq i \leq (p-3)/2$, such that:

$$f(X,Y) = \sum_{n=0}^{(p-3)/2} f'_n(Y)Z'^n, \tag{13}$$

with $Z' = (X+1)(X-Y)$ *and* $\deg(f'_n(Y)) = p-3-2n$.

Proof. The idea of the proof is to show that we have $f(X,0) = f(X,X)$. Then since Y divides $g(X,Y) = f(X,Y) - f(X,0)$ and $X-Y$ divides $g'(X,Y) = f(X,Y) - f(X,X)$ we obtain that $Y(X-Y)$ divides $g(X,Y) = g'(X,Y)$. Then we re-apply the same procedure for $g(X,Y)$.

Since $f(X,Y)$ has total degree $p-3$, $f(X,0)$ and $f(X,X)$ also have degree smaller than $p-3$ therefore it is enough to show that $f(x,0) = f(x,x)$ on $p-2$ distinct values of x. Let $i \in [1, p-2]$, from the definition of $P_{\mathtt{LT}_\mathcal{S}}$ we have

$$\begin{aligned} P_{\mathtt{LT}_\mathcal{S}}(i,Y) &= \sum_{j=i+1}^{p-1} 1 - (Y-j)^{p-1} \\ &= p-1-i - \sum_{k=0}^{p-1} \left(\sum_{j=i+1}^{p-1} j^{p-1-k} \right) Y^k \\ &= -\sum_{k=1}^{p-1} \left(\sum_{j=i+1}^{p-1} j^{p-1-k} \right) Y^k \end{aligned} \tag{14}$$

but also

$$\begin{aligned} P_{\mathtt{LT}_\mathcal{S}}(i,Y) &= p-1-i - \sum_{j=i+1}^{p-1} (Y-j)^{p-1} \\ &= p-1-i - \sum_{j=1}^{p-1-i} (Y-i-j)^{p-1} \\ &= p-1-i - \sum_{k=0}^{p-1} \left(\sum_{j=1}^{p-1-i} j^{p-1-k} \right) (Y-i)^k \\ &= -\sum_{k=1}^{p-1} \left((-1)^k \sum_{j=1}^{p-1-i} j^{p-1-k} \right) (i-Y)^k \end{aligned} \tag{15}$$

So by dividing (14) and (15) by $Y(i-Y)(i+1)$ (for $i \neq p-1$) we have:

$$\begin{aligned} f(i,Y) &= -\sum_{k=1}^{p-1} \left(\underbrace{\frac{\sum_{j=i+1}^{p-1} j^{p-1-k}}{(i+1)(i-Y)}}_{a_{i,k}(Y)} \right) Y^{k-1} \\ &= -\sum_{k=1}^{p-1} \left(\underbrace{\frac{(-1)^k \sum_{j=1}^{p-1-i} j^{p-1-k}}{(i+1)Y}}_{b_{i,k}(Y)} \right) (i-Y)^{k-1} \end{aligned}$$

Now notice that for $i \in [1, p-1)$ and $k \in [1, p-1)$ we have

$$\begin{aligned} a_{i,k}(0) &= \frac{\sum_{j=i+1}^{p-1} j^{p-1-k}}{(i+1)i} = \frac{\sum_{j=1}^{p-1-i} (-j)^{p-1-k}}{(i+1)i} \\ &= \frac{(-1)^{p-1-k} \sum_{j=1}^{p-1-i} j^{p-1-k}}{(i+1)i} \\ &= \frac{(-1)^{k} \sum_{j=1}^{p-1-i} j^{p-1-k}}{(i+1)i} = b_{i,k}(i) \bmod p \end{aligned}$$

Proving that $f(i,0) = f(i,i)$ is exactly proving that $a_{i,1}(0) = b_{i,1}(i)$ in which case $f(X,0) = f(X,X)$ and $g(X,Y) = (f(X,Y) - f(X,0))/(Y(X-Y))$ will be defined as a polynomial. However, in order to prove Equation (12) we will need to prove that this decomposition works at higher order i.e. that $g(X,0) = g(X,X)$ so that $Y(X-Y)$ divides $h(X,Y) = g(X,Y) - g(X,0)$ and so on until we obtain a polynomial of degree 0.

The coefficient of degree 0 of $g(i,Y) = (f(i,Y) - a_{i,0}(0))/(Y(i-Y))$ in base Y is $a_{i,2}(0)/(i-0)$ and $b_{i,2}(i)/i$ in base $i - Y$. By induction, proving Equation (12) is exactly showing that the coefficients $a_{i,k}(0)/i^{k-1}$ and $b_{i,k}(i)/i^{k-1}$ in the above decompositions are equals or equivalently that $a_{i,k}(0) = b_{i,k}(i)$ for $p-2$ distinct values of i and every $k \in [1, p-1)$.

Hence, by defining $f_0(X,Y) = f(X,Y)$ and $f_{n+1}(X,Y) = (f_n(X,Y) - f(X,0))/(Y(X-Y))$ we have shown that $f_n(x,0) = f_n(x,x)$ for any $x \in [1, p-2]$ and $0 \leq n \leq (p-3)/2$ and therefore that the $f_n(X,Y)$ are well defined as polynomials.

Since $f_n(X,Y)$ has total degree smaller than $p-3-2n$ and that $f_n(X,0)$ and $f_n(X,X)$ are equals on $p-2$ distinct values, they must be equal as polynomials.

The second decomposition can be obtained in the same way by considering $P_{\mathtt{LT}_\mathcal{S}}(X,j)$ for any $j \in [1, p-1)$.

According to Theorem 3, in order to evaluate $P_{\mathtt{LT}_\mathcal{S}}$ one needs:

- 2 multiplications to compute $Z(X+1)$ with $Z = Y(X-Y)$;

and for $(p-3)/2 > 0$, i.e. $p \geq 5$

- $p-4$ multiplications to compute the X^is for $1 \leq i \leq p-3$;
- $(p-5)/2$ multiplications to compute the Z^js for $1 \leq j \leq (p-3)/2$;
- $(p-5)/2$ multiplications to compute the $f_n(X) \cdot Z^n$ for $1 \leq n \leq (p-3)/2$ since $f_{(p-3)/2}(X)$ has degree 0 and is thus a constant;
- 1 multiplication to compute $Z(X+1) \cdot f(X,Y)$.

Therefore, as explained in [14], for $p \geq 5$ at most $2p-6$ homomorphic multiplications are required to evaluate $P_{\mathtt{LT}_\mathcal{S}}$ in total.

5 Acknowledgments

The first author is supported by CyberSecurity Research Flanders with reference number VR20192203 and by a Junior Postdoctoral Fellowship from the Research Foundation – Flanders (FWO).

6 Conclusion

In this work, we proved that several integer functions have non-trivial polynomial interpolations over finite fields that facilitate faster evaluation of these functions using somewhat homomorphic encryption.

In particular, we described a family of function that have almost all evenly indexed coefficients equal to zero in any prime finite field $\mathbb{F}_p$. These functions include the parity function and the "is negative" function and generalize the result of Iliashenko and Zucca [14]. The same phenomenon occurs for the modulo function $|x|_m$ if $p \equiv -1 \mod m$. This implies that the above functions can be computed in approximately $\sqrt{p-3}$ non-scalar (ciphertext-ciphertext) multiplications.

We showed that unary functions with scaled all-one polynomial interpolations in $\mathbb{F}_p$ can be computed in $\mathcal{O}(\log p)$ non-scalar multiplications. These include 'is power of b' functions if $p = (b^r - 1)/k$ for some $k < b$ and a positive integer r.

We also proved that if $p = 2^q - 1$ is a Mersenne prime, then the Hamming weight and the Mod2 functions have only coefficients with odd indexes equal to multiples of q plus the leading and the constant terms. This allows to evaluate these functions with $\mathcal{O}\left(\sqrt{p/\log p}\right)$ non-scalar multiplications.

Finally, we proved the claim of Iliashenko and Zucca [14] that the polynomial interpolation of the less-than function is equal to

$$\begin{aligned} P_{\mathtt{LT}_S}(X,Y) &= (X+1)Z \sum_{i=0}^{(p-3)/2} f_i(X) Z^i \\ &= YZ' \sum_{i=0}^{(p-3)/2} f_i(Y) Z'^i \end{aligned}$$

where $Z = Y(X-Y)$ and $Z' = (X+1)(X-Y)$. This result substantiates that the non-scalar complexiy of evaluating this function is $2p-6$ as shown in [14] if $p \geq 5$.

Future work. This work demonstrates only a few families of integer functions with non-trivial non-scalar complexity. We strongly believe that other examples of such functions are yet to be found. In addition, we limited our search to the context of prime finite fields. The next logical step is to enlarge this context to extensions of finite fields. To the best of our knowledge, there are no works for interpolations over such fields in the context of SHE computation.

Another interesting research direction is to study interpolations over rings $\mathbb{Z}_{p^e}$. The current results [12,3] are limited to the function $f(x) = x - |x|_p$. However, such rings are exploited in practice to implement byte-wise arithmetic and logic. The non-scalar complexity of such operations is yet to be studied.

References

1. Brakerski, Z., Gentry, C., Vaikuntanathan, V.: (Leveled) Fully Homomorphic Encryption without Bootstrapping. In: Proceedings of the 3rd Innovations in Theoretical Computer Science Conference. p. 309–325. ITCS '12, Association for Computing Machinery, New York, NY, USA (2012), https://doi.org/10.1145/2090236.2090262
2. Chen, H., Chillotti, I., Song, Y.: Improved bootstrapping for approximate homomorphic encryption. In: Ishai, Y., Rijmen, V. (eds.) EUROCRYPT 2019, Part II. LNCS, vol. 11477, pp. 34–54. Springer, Heidelberg (May 2019)
3. Chen, H., Han, K.: Homomorphic lower digits removal and improved FHE bootstrapping. In: Nielsen, J.B., Rijmen, V. (eds.) EUROCRYPT 2018, Part I. LNCS, vol. 10820, pp. 315–337. Springer, Heidelberg (Apr / May 2018)

4. Cheon, J.H., Han, K., Kim, A., Kim, M., Song, Y.: Bootstrapping for approximate homomorphic encryption. In: Nielsen, J.B., Rijmen, V. (eds.) EUROCRYPT 2018, Part I. LNCS, vol. 10820, pp. 360–384. Springer, Heidelberg (Apr / May 2018)
5. Cheon, J.H., Kim, A., Kim, M., Song, Y.: Homomorphic encryption for arithmetic of approximate numbers. In: Takagi, T., Peyrin, T. (eds.) Advances in Cryptology – ASIACRYPT 2017. pp. 409–437. Springer International Publishing, Cham (2017)
6. Cheon, J.H., Kim, D., Kim, D.: Efficient homomorphic comparison methods with optimal complexity. In: ASIACRYPT 2020, Part II. pp. 221–256. LNCS, Springer, Heidelberg (Dec 2020)
7. Cheon, J.H., Kim, D., Kim, D., Lee, H.H., Lee, K.: Numerical method for comparison on homomorphically encrypted numbers. In: Galbraith, S.D., Moriai, S. (eds.) ASIACRYPT 2019, Part II. LNCS, vol. 11922, pp. 415–445. Springer, Heidelberg (Dec 2019)
8. Ducas, L., Micciancio, D.: FHEW: Bootstrapping homomorphic encryption in less than a second. In: Oswald, E., Fischlin, M. (eds.) EUROCRYPT 2015, Part I. LNCS, vol. 9056, pp. 617–640. Springer, Heidelberg (Apr 2015)
9. Fan, J., Vercauteren, F.: Somewhat Practical Fully Homomorphic Encryption. Cryptology ePrint Archive, Report 2012/144 (2012), `https://eprint.iacr.org/2012/144`
10. Gentry, C.: Fully homomorphic encryption using ideal lattices. In: Mitzenmacher, M. (ed.) 41st ACM STOC. pp. 169–178. ACM Press (May / Jun 2009)
11. Gentry, C., Sahai, A., Waters, B.: Homomorphic encryption from learning with errors: Conceptually-simpler, asymptotically-faster, attribute-based. In: CRYPTO (1). pp. 75–92 (2013)
12. Halevi, S., Shoup, V.: Bootstrapping for HElib. In: Oswald, E., Fischlin, M. (eds.) EUROCRYPT 2015, Part I. LNCS, vol. 9056, pp. 641–670. Springer, Heidelberg (Apr 2015)
13. Han, K., Ki, D.: Better bootstrapping for approximate homomorphic encryption. In: CT-RSA 2020. pp. 364–390. LNCS, Springer, Heidelberg (2020)
14. Iliashenko, I., Zucca, V.: Faster homomorphic comparison operations for BGV and BFV. PoPETs 2021(3), 246–264 (2021)
15. Kaji, S., Maeno, T., Nuida, K., Numata, Y.: Polynomial expressions of p-ary auction functions. Journal of Mathematical Cryptology 13(2), 69–80 (2019)
16. Lyubashevsky, V., Peikert, C., Regev, O.: On ideal lattices and learning with errors over rings. Journal of the ACM (JACM) 60(6), 43 (2013)
17. Paterson, M.S., Stockmeyer, L.J.: On the number of nonscalar multiplications necessary to evaluate polynomials. SIAM Journal on Computing 2(1), 60–66 (1973)
18. Regev, O.: On lattices, learning with errors, random linear codes, and cryptography. In: Proceedings of the 37th Annual ACM Symposium on Theory of Computing, Baltimore, MD, USA, May 22-24, 2005. pp. 84–93 (2005), `http://doi.acm.org/10.1145/1060590.1060603`
19. Tan, B.H.M., Lee, H.T., Wang, H., Ren, S.Q., Khin, A.M.M.: Efficient private comparison queries over encrypted databases using fully homomorphic encryption with finite fields. IEEE Transactions on Dependable and Secure Computing pp. 1–1 (2020)

A Remarks

A.1 Extensions of "is power of b" function

Alternative computation We have $b \in \mathbb{F}_p$ and $\ell = \lfloor \log_b(p) \rfloor$. The following polynomial

$$P'_f(X) = 1 - \left(\prod_{i=0}^{\ell} (X - b^i) \right)^{p-1}$$

satisfies

$$P'_f(x) = \begin{cases} 1 \text{ if } x = b^i \text{ for some } i \in \{0, \ldots, \ell\} \\ 0 \text{ otherwise} \end{cases}$$

$P'_f(x)$ can then be computed with ℓ multiplications for $(\prod_{i=0}(X - b^i))$ and $\leq 2\log_2(p)$ multiplications for the exponentiation to the power $p-1$, which leads to $\ell + 2\log_2(p) = O(\log(p))$ multiplications. Note that the multiplicative depth of this circuit is $\lceil \log \ell \rceil + \lceil \log(p-1) \rceil$.

Extension to b generator of a subgroup We consider the function $f_G(x)$ which, given a subgroup $G =< b >$ of $\mathbb{F}_p^\times$, equals 1 on G and 0 elsewhere:

$$f_G(x) = \begin{cases} 1 \text{ if } x \in G \\ 0 \text{ otherwise} \end{cases} \tag{16}$$

We assume now that b has order m and $p-1 = nm$ with $\gcd(n,m) = 1$ (in this case m might be different than $\log_b(p)$). In this case there exists $h \in \mathbb{F}_p^\times$ of order n such that:

$$\mathbb{F}_p^\times = \{h^i b^j \text{ for } i = 0, \dots n-1 \text{ and } j = 0, \dots, m-1\}.$$

The following polynomial satisfies Equation (16)

$$\begin{aligned} P_f(X) &= \textstyle\sum_{j=0}^{m-1}(1-(X-b^j)^{p-1}) \\ &= m - \textstyle\sum_{a=1}^{p-1} X^i \sum_{j=0}^{m-1}(b^j)^{p-1-i} \end{aligned}$$

When $p-1-i \not\equiv 0 \mod m$ we have $b^{p-1-i} \neq 1$ and then the coefficient of P_f of degree i becomes

$$\textstyle\sum_{j=0}^{m-1}(b^j)^{p-1-i} = \frac{(b^{p-1-i})^m - 1}{b^{p-1-i}-1} = 0$$

when $p-1-i \equiv 0 \mod m$ we have

$$\textstyle\sum_{j=0}^{m-1}(b^j)^{p-1-i} = m$$

In other words:

$$P_f(X) = m - m\textstyle\sum_{i=0}^{(p-1)/m} X^{im}$$

This function can be evaluated as in Section 3.2.

Alternative polynomial representation. An alternative polynomial expression of the function f_G is the following :

$$P'_{f_G}(X) = 1 - (1-X^m)^{p-1}$$

One can check that this polynomial satisfies (16) since for $x \in G$ we have $x^m = 1$ and the $1-(1-x^m)^{p-1} = 0$. For x not in G, $x^m \neq 1$, and then $(1-x^m)^{p-1} = 1$ as required.

$P'_{f_G}(X)$ can be evaluated at x in $2\log(m)) + 2\log(p)$ multiplications, but with depth $\lceil \log m \rceil + \lceil \log(p-1) \rceil$.