Random polynomials in Legendre sequences

Katalin Gyarmati and Károly Müllner

Abstract

It is crucial in pseudorandomness cryptographic applications that the smaller key used as a seed can be generated at random. Thus, if the Legendre sequence based on a polynomial (as proposed by Hoffstein and Lieman) is used, that is

$$\left\{\left(\frac{f(1)}{p}\right),\left(\frac{f(2)}{p}\right),\left(\frac{f(3)}{p}\right),\dots,\left(\frac{f(p)}{p}\right)\right\},$$

it is important to choose the polynomial f at random. Goubin, Mauduit, and Sárközy presented some non-restrictive conditions on the polynomial f, but these conditions may not be satisfied if we choose a truly random polynomial. However, how can it be ensured that the sequence's pseudorandom measures are always low for nearly "random" polynomials? These semirandom polynomials will be constructed with as few modifications as necessary from a truly random polynomial.

1 Introduction

Mauduit and Sárközy [9] proposed new quantitative techniques to study binary sequence pseudorandomness in 1997. They came up with the following new measures:

2020 Mathematics Subject Classification: Primary: 11K45, Secondary: 11C08.

Keywords and phrases: pseudorandomness, random polynomial

Research supported by Hungarian National Research Development and Innovation Funds KKP133819 and K119528

Definition 1 *For a binary sequence*

$$E_N = \{e_1, \ldots, e_N\} \in \{-1, +1\}^N,$$

define the well-distribution measure of E_N *as*

$$W(E_N) = \max_{a,b,t} \left| \sum_{j=1}^{t} e_{a+jb} \right|,$$

where the maximum is taken over all a, b, t *such that* $a \in \mathbb{Z}$, $b, t \in \mathbb{N}$ *and* $1 \le a \le a+tb \le N$, *while the correlation measure of order* ℓ *of* E_N *is defined as*

$$C_\ell(E_N) = \max_{M,D} \left| \sum_{n=1}^{M} e_{n+d_1} \ldots e_{n+d_\ell} \right|,$$

where the maximum is taken over all $D = (d_1, \ldots, d_\ell)$ *and* M *such that* $0 \le d_1 < \cdots < d_\ell < M + d_\ell \le N$.

The Legendre sequence was the first to be studied using the measures defined above. This:

$$E_{p-1} = \left\{ \left(\frac{1}{p}\right), \left(\frac{2}{p}\right), \left(\frac{3}{p}\right), \ldots, \left(\frac{p-1}{p}\right) \right\}.$$

Sárközy and Mauduit [9] proved that:

$$\begin{aligned} W(E_{p-1}) &\ll p^{1/2} \log p, \\ C_\ell(E_{p-1}) &\ll \ell p^{1/2} \log p. \end{aligned} \tag{1}$$

Since these estimates are significantly sharper than the trivial estimate, we may claim that the Legendre sequence has strong pseudorandom measures.

What estimates may be given for W and C_ℓ for an average sequence was a key topic in developing the theory of pseudorandomness. Cassaigne, Mauduit, and Sárközy [4] proved that for nearly every binary sequence of length N,

$$\sqrt{N} \ll W(E_N) \ll \sqrt{N \log N}$$

and

$$\sqrt{N} \ll C_\ell(E_N) \ll \sqrt{\ell N \log N}.$$

Alon, Kohayakawa, Mauduit, Moreira and Rödl [2] sharpened the lower estimate with a $\sqrt{\log N}$ and the upper estimate with a constant factor, giving the exact expected magnitude of these measures. Based on these results, it is safe to conclude that a sequence has very strong pseudorandom properties if

$$W(E_N),\ C_\ell(E_N) \ll \sqrt{N}\,(\log N)^c\,.$$

However, in practical applications, to have the estimates

$$W(E_N),\ C_\ell(E_N) \ll N^{1-\varepsilon}$$

with a positive constant ε (as $N \to \infty$) is sufficient. It should be noted that in practical applications, a lower estimate is not required at all. As Alon, Kohayakawa, Mauduit, Moreira and Rödl [1] demonstrated for even-order correlation measure we always have

$$C_{2\ell}(E_N) \gg \sqrt{N}.$$

Although the odd-order correlation can be very small, even one, it is clear from Gyarmati's [6] and later Anantharam's [3] and Gyarmati and Mauduit's [7] estimates that requiring a lower estimate is completely unnecessary.

The Legendre sequence possesses very strong pseudorandom measures, according to the work of Sárközy and Mauduit [9], see (1). This construction has one major drawback: it only provides one sequence for each prime. Hoffstein and Liemann [8] aided in this with a clever idea. Their construction was as follows:

Construction 1 (Hoffstein, Liemann) *Let p be a prime, $f(x) \in \mathbb{F}_p[x]$ be a polynomial of degree k. Define $E_p = (e_1, \ldots, e_p)$ by:*

$$e_n = \begin{cases} \left(\frac{f(n)}{p}\right) & \text{for } (f(n), p) = 1, \\ +1 & \text{for } p \mid f(n). \end{cases}$$

Hoffstein and Lieman, on the other hand, did not prove anything about the sequence's pseudorandomness; they only observed that it possesses strong pseudorandom properties. However, Goubin, Mauduit, and Sárközy [5] thoroughly studied Construction 1 and proved the following:

Theorem 1 (Goubin, Mauduit, Sárközy) *Let p be a prime, $f(x) \in \mathbb{F}_p[x]$ be a polynomial of degree k, which is not of the form $cg(x)^2$, where $c \in \mathbb{F}_p$, $g(x) \in \mathbb{F}_p[x]$. Define $E_p = (e_1, \ldots, e_p)$ by Construction 1. Then*

$$W(E_p) \ll kp^{1/2} \log p.$$

Assume that one of the following three conditions for ℓ, which is the order of the correlation, holds true:
(i) $\ell = 2$;
(ii) $\ell < p$ and 2 is a primitive root modulo p;
(iii) $(4k)^\ell < p$.
Then:

$$C_\ell(E_p) \ll k\ell p^{1/2} \log p.$$

Goubin, Mauduit, and Sárközy [5] also provided polynomials f such that the associated sequences have a high correlation. As a result, there is a real need for the conditions they offer, or conditions similar to those, for the related sequence to have strong pseudorandom properties.

As a result, the pseudorandom measures of the sequences given in Construction 1 are optimal, the sequence elements can be generated quickly, and the construction is natural. It is without a doubt one of the most effective pseudorandom generators ever developed.

If the prime p and the coefficients of the polynomial f are given, the sequence in Construction 1 can be quickly programmed. When a sequence in Construction 1 is used as a secret key in cryptographic systems, the polynomial f must be chosen at random. This is significant because, for example,

if the polynomial coefficients in

$$f(x) = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0$$

are consecutive integers, the value of the other coefficients is immediately derived from the value a_n. Thus, simply looking at the cases $a_n = 1, 2, 3, ..., p-1$, the value of the sequence used as the secret key becomes decipherable. Similar problem can happen when f only has a few non-zero coefficients, in which case the secret key can be decrypted again using brute force. Consider how substantial the problem was when the primes p and q in the $N = pq$ modulus in RSA were consecutive primes rather than primes chosen at random. In this case, the RSA was immediately decipherable. In general, if there is a chance that the secret key will be decrypted within a reasonable time after trying all possible cases, then our encryption system is unacceptable.

Returning to the sequences in Construction 1, it is critical that the sequences' starting point, the polynomial f, be chosen at random. It is also critical that the degree of the polynomial f, k not be too small. Since all sequences based on a polynomial of degree k or less can be programmed in a reasonable amount of time in the case of a small degree, and then our key sequence is no longer secret. As a result, in Construction 1, users need to choose the degree of the polynomial f for at least p^{ε} for some small positive constant ε. We believe that $\varepsilon = 0.1$ is ideal for applications, for example. However, if the degree is large, condition (iii) of Theorem 1 does not apply. Furthermore, if 2 is not a primitive root mod p, condition (ii) does not hold, and a high-order correlation measure can be very large. Although Artin's conjecture states that 2 is the primitive root of an infinite number of primes, no one has yet proven it.

We employ the following strategy: we select a random polynomial f. Find a quadratic non-residue n for which $f(x)$ has no irreducible factor of the form $(x + c)^2 - n$, where $c \in \mathbb{F}_p$. We also select a random $a \in \mathbb{F}_p$. Then, instead

of f in Construction 1, we use the polynomial $g(x) = ((x+a)^2 - n)f(x)$.

Since f is a random polynomial, we can say that g is semirandom. The only thing we know about it is that it has a quadratic irreducible factor. We will show that the pseudorandom measures of the sequences in Construction 1 based on the new polynomial g are optimally small. Then we create a fast (polynomial-time) algorithm for calculating an appropriate n quadratic non-residue. Although our algorithm will be probabilistic rather than deterministic, it will not be bemusing in practice. The probability of never finding a suitable quadratic non-residue n after running the Step 1-Step 6 of the algorithm 200 times is $< \frac{1}{2^{100}}$. This is even less likely than winning the lottery three times in a row.

First, we prove the following:

Theorem 2 *Let p be a prime, $a \in \mathbb{F}_p$, n be a quadratic non-residue modulo p, and $f(x) \in \mathbb{F}_p[x]$ be a polynomial of degree k. If $f(x)$ has no irreducible factor of the form $(x+c)^2 - n$, where $c \in \mathbb{F}_p$, define the polynomial $g(x)$ as follows:*

$$g(x) = ((x+a)^2 - n)f(x).$$

Furthermore, the sequence $E_p = \{e_1, e_2, e_3, \ldots, e_p\}$ is defined in the same way as in Construction 1, but with $g(x)$ inside the Legendre symbol:

$$e_n = \begin{cases} \left(\frac{g(n)}{p}\right) & \text{for } (g(n), p) = 1, \\ +1 & \text{for } p \mid g(n). \end{cases}$$

Then:

$$W(E_n) \ll kp^{1/2} \log p,$$
$$C_\ell(E_n) \ll k\ell p^{1/2} \log p.$$

Proof of Theorem 2. Goubin, Mauduit, and Sárközy [5] proposed the following equivalence relation: The polynomials φ and $\psi \in \mathbb{F}_p[x]$ are equivalent if $c \in \mathbb{F}_p$ exists for which

$$\varphi(x) = \psi(x+c).$$

Since n is a quadratic non-residue modulo p, the polynomial $g(x)$ in Theorem 2 does have the irreducible factor $(x+a)^2 - n$. There is no other irreducible factor that is equivalent to $(x+a)^2 - n$. Thus $g(x)$ is not of the form $c'g'(x)^2$ with $c' \in \mathbb{F}_p$ and $g'(x) \in \mathbb{F}_p[x]$. Furthermore, we know that for $1 \leq d_1 < d_2 < \cdots < d_\ell \leq p$, the polynomial $g(x+d_1)g(x+d_2)\cdots g(x+d_k)$ is not of the form $c'g'(x)^2$ with $c' \in \mathbb{F}_p$ and $g'(x) \in \mathbb{F}_p[x]$, since the factors equivalent to $(x+a)^2 - n$ in this product are:

$$(x+a+d_1)^2 - n, (x+a+d_2)^2 - n, \ldots, (x+a+d_\ell)^2 - n.$$

Each of the above irreducible factors appears exactly once in the decomposition of the polynomial $g(x+d_1)g(x+d_2)\cdots g(x+d_k)$ into irreducible factors.

We then use Weil's theorem [10] for the prime p and the Legendre symbol character:

Lemma 1 (Weil) *Suppose that $\mathbb{F}_q$ is a finite fields, χ is a non-principal character of order d over it, $f \in \mathbb{F}_q[x]$ has s distinct roots in $\overline{\mathbb{F}}_q$, and it is not a constant multiple of the d-th power of a polynomial over $\mathbb{F}_q$. Then:*

$$\left| \sum_{n \in \mathbb{F}_q} \chi(f(n)) \right| \leq (s-1)q^{1/2}.$$

From here, the argument is the same as in the paper of Goubin, Mauduit, and Sárközy [5]; it is based on Weil's theorem above, and the estimates for $W(E_p)$ and $C_\ell(E_p)$ are obtained immediately. This concludes Theorem 2's proof.

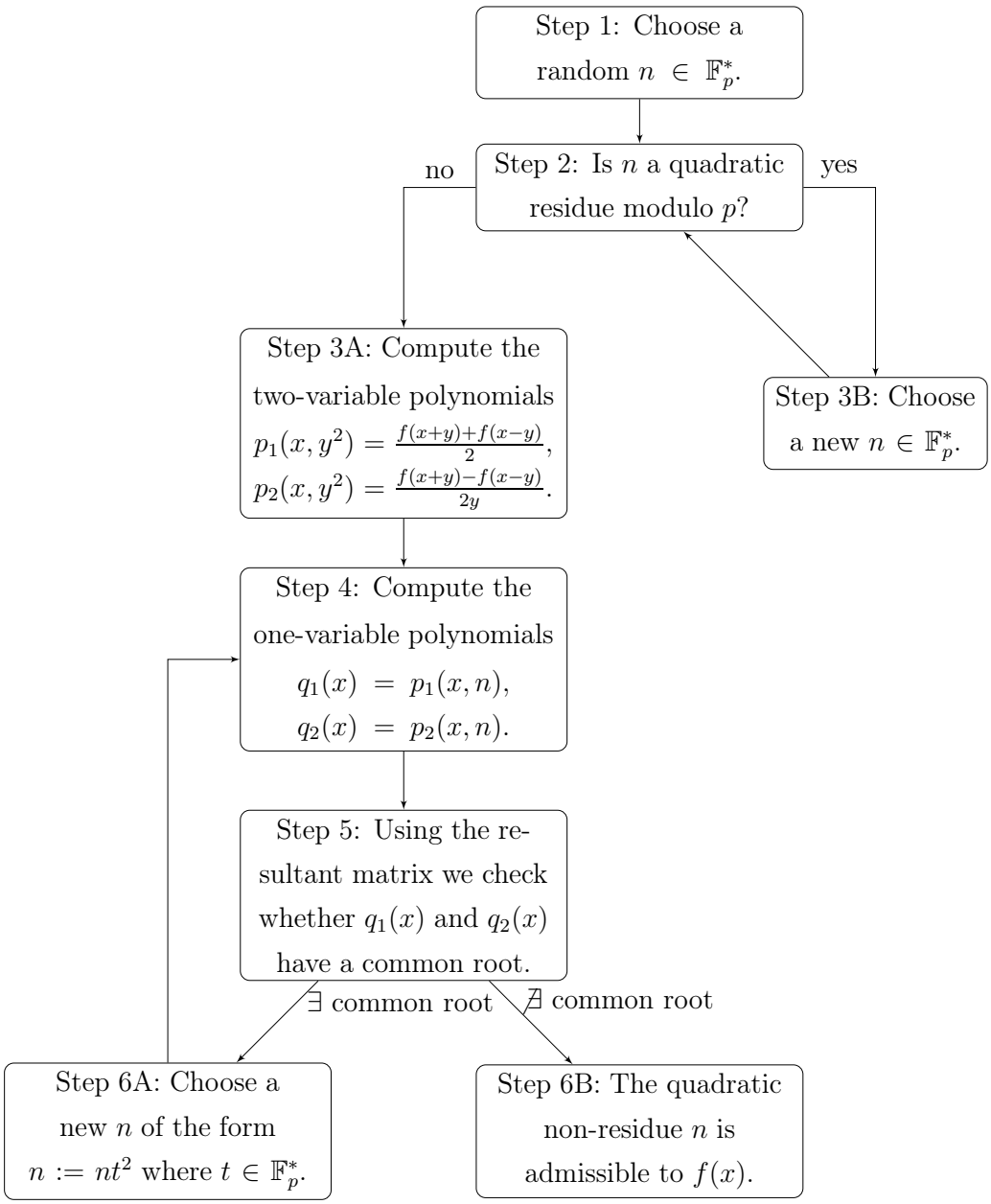
2 Admissible quadratic non-residues and the algorithm

However, what conditions on the quadratic non-residue n are required to ensure that the polynomial $f(x)$ does not have an irreducible $(x+a)^2 - n$-

shaped factor, where $a \in \mathbb{F}_p$? At first glance, this may appear to be a difficult question, but it is much easier to say that a polynomial does not have a specific type of root than it is to find one of the roots. We will now present a fast (polynomial-time) algorithm for finding an appropriate n quadratic non-residue. But first, a definition will be provided.

Definition 2 *Let p be an odd prime, $f(x) \in \mathbb{F}_p[x]$ be a polynomial and n be a quadratic non-residue modulo p. If $f(x)$ has no irreducible factor of the form $(x+c)^2 - n$, with $c \in \mathbb{F}_p$, then the quadratic non-residue n is said to be admissible to $f(x)$.*

Following that, we describe our algorithm for determining an admissible n to $f(x)$. The algorithm is first illustrated in a figure, followed by step-by-step instructions for each step of the algorithm.



This algorithm is totally right if it states that n is admissible to f, however there are many n's for which the algorithm cannot determine whether or not n is admissible, in which case a new n must be picked. The algorithm is extremely efficient and finds an admissible n in polynomial time. The fact that it is never stated that a particular n is not admissible is immaterial

because the primary goal of the technique is to generate admissible n's. Following this, we will study the algorithm in detail, including its speed, storage, and probabilistic nature.

We would like to begin the algorithm by generating a quadratic non-residue n. This is accomplished by using random techniques. In Step 1, a random $n \in \mathbb{F}_p$ is chosen. Since the number of quadratic residues and quadratic non-residues are both $\frac{p-1}{2}$, n has a 50% probability of being a quadratic residue and a 50% probability of being a quadratic non-residue. If n happens to be a quadratic residue, we choose a new $n \in \mathbb{F}_p^*$. The process is repeated until a quadratic non-residue is found. The probability that we will always find a quadratic residue n in 100 trials is quite low: $\frac{1}{2^{100}}$. As a result, we can undoubtedly find a quadratic non-residue in a very short time.

Next we try to find a n quadratic non-residue that is admissible to f. To do this, calculate the two-variables polynomials p_1 and p_2 given in Step 3A. For p_1 and p_2 to be well defined, all the coefficients of the polynomials

$$\frac{f(x+y)+f(x-y)}{2}$$

and

$$\frac{f(x+y)-f(x-y)}{2y}$$

must be in $\mathbb{F}_p$ and the exponent of every power of y must be even in these polynomials. The first statement is obvious, but to prove the second, we write $f(x)$ in the form

$$f(x) = a_n x^n + a_{n-1}x^{n-1} + a_{n-2}x^{n-2} + \cdots + a_1 x + a_0.$$

Then

$$\frac{f(x+y)+f(x-y)}{2} = \sum_{i=0}^{n} a_i \frac{(x+y)^i + (x-y^i)}{2}, \tag{2}$$

$$\frac{f(x+y)+f(x-y)}{2y} = \sum_{i=0}^{n} a_i \frac{(x+y)^i - (x-y^i)}{2y}, \tag{3}$$

$$\tag{4}$$

According to the binomial theorem, the exponent of y in both polynomials in (2) and (3) is even. That is, the polynomials p_1 and p_2 are well defined by the formulas given in Step 3A:

$$p_1(x, y^2) = \frac{f(x+y) + f(x-y)}{2},$$
$$p_2(x, y^2) = \frac{f(x+y) - f(x-y)}{2y}.$$

A simple calculation shows that

$$f(x+y) = p_1(x, y^2) + yp_2(x, y^2), \tag{5}$$
$$f(x-y) = p_1(x, y^2) - yp_2(x, y^2). \tag{6}$$

Assume that for a fixed n quadratic non-residue, $f(x)$ has an irreducible factor of the form $(x+c)^2 - n$ with $c \in \mathbb{F}_p$. Now that n is a quadratic non-residue, F_{p^2} has an element θ for which

$$\theta^2 = n.$$

Then $\mathbb{F}_{p^2} = \mathbb{F}_p(\theta)$. Since $\theta \notin \mathbb{F}_p$, we get that in case

$$u + v\theta = 0, \qquad u, v \in \mathbb{F}_p$$

we have

$$u = 0, v = 0$$

Since $f(x)$ has the irreducible factor $(x+c)^2 - n$, we know that $-c + \theta$ and $-c - \theta$ are roots of $f(x)$. Then write $x = -c$ and $y = \theta$ in (5) and (6) to get the following result:

$$0 = f(-c+\theta) = p_1(-c, \theta^2) + \theta p_2(-c, \theta^2),$$
$$0 = f(-c-\theta) = p_1(-c, \theta^2) - \theta p_2(-c, \theta^2).$$

From this

$$p_1(-c, n) = 0 \qquad \text{and} \qquad p_2(-c, n) = 0.$$

That is, the polynomials q_1 and q_2 given in Step 4 have a common root in this case: $-c$. If the resultant of the two polynomials q_1 and q_2 is not zero in $\mathbb{F}_p$, the polynomials have no common root, and hence $f(x)$ cannot have an irreducible factor of the form $(x+c)^2-n$. That is, n is admissible to $f(x)$. If the resultant of the two polynomials q_1 and q_2 is zero in $\mathbb{F}_p$, the polynomials have a common root (but it is far from certain that it is in $\mathbb{F}_p$). Since we do not know if n is admissible in this case, we choose a new quadratic non-residue n. Since every quadratic non-residue have the form nt^2 where $t \in \mathbb{F}_p^*$, we do not need to use probabilistic methods to create this new quadratic non-residue; simply define the new n by $n := nt^2$ with a $t \in \mathbb{F}_p^*$. We return to Step 4 with this new n.

What is the probability that we will get at Step 6B after Step 5 for a randomly chosen quadratic non-residue n, i.e. the polynomials q_1 and q_2 do not have a common root? We know that if there is a common root, which we denote by a, then:

$$\begin{aligned} p_1(a,n) &= 0 \\ p_2(a,n) &= 0 \\ \theta^2 &= n \\ f(a+\theta) &= p_1(a,n) + \theta p_2(a,n) = 0 \\ f(a-\theta) &= p_1(a,n) - \theta p_2(a,n) = 0. \end{aligned}$$

That is, f has two roots, $\alpha = a+\theta$ and $\beta = a-\theta$ for which

$$\begin{aligned} \theta &= \frac{\alpha-\beta}{2}, \\ n &= \left(\frac{\alpha-\beta}{2}\right)^2. \end{aligned}$$

Then

$$n \in \mathcal{F} \stackrel{\text{def}}{=} \left\{ \left(\frac{\alpha-\beta}{2}\right)^2 : \ \alpha, \beta \text{ are different roots of } f \right\}.$$

Clearly,

$$|\mathcal{F}| \leq \binom{\deg f}{2} < \frac{k^2}{2}.$$

That is, if $n \notin \mathcal{F}$, the polynomials q_1 and q_2 have no common root, implying that the quadratic non-residue n is definitely admissible to f.

Since $\mathcal{F}$ has less than $k^2/2$ elements, thus if we do Step 4-Step 6 on at least $k^2/2$ different quadratic non-residues n (which is possible for $p > k^2$), then we will undoubtedly find one of them that is admissible to f. We also mention the probabilistic nature of these steps: by using the upper bound $k^2/2$ for the number of elements in $\mathcal{F}$, we can see that if $p > 2k^2$, the chance of getting from Step 5 to Step 6B for a randomly chosen quadratic non-residue n is more than $1/2$. That is, if we run this section of the algorithm 100 times, we will almost certainly find a quadratic non-residue that is admissible to f. Time required for the algorithm is $O(k^3(\log p)^2)$, storage required is $O(k^2 \log p)$.

3 The algorithm's program code

The algorithm from previous section was implemented in Matlab [11]. Our program is able to find a quadratic non-residue n which is admissible to $f(x)$. We must first enter a prime number (p), and the code will determine whether or not it is truly prime. Then we must specify which polynomial will be used during the algorithm. At first we need the degree of polynomial, and save in a variable (m). When defining the polynomial, the coefficients are requested in separately and stored in a vector for later use.

After performing the required calculations with the given polynomial $(h_1(x))$ we can compute the two-variable polynomials (p_1, p_2). In both polynomials, the exponent of y is even. At this point, we ask the user to enter an integer n that is less than p but greater than 1 and we enter a 'while' loop to repeat when the condition is true. In this loop, we must determine whether n is a quadratic non-residue modulo p. We can use the jacobi(n,p) function from another M-file where it is implemented for this check.

In case of n is suitable let's compute the one-variable polynomials with $y^2 = n$ substitutions. Using the resulting matrix, we determine whether

these two (one-variable) polynomials have a common root. If yes, we must choose a new n; otherwise, the quadratic non-residue n is admissible to $f(x)$. The program also prints the runtime, which is affected by the choice of n, p and $f(x)$ polynomial.

```
clear;

prompt = "Give a prime: " ;
    p = input(prompt);

while isprime(p) == 0
    fprintf('Invalid input. Give a prime!: \n');
    p = input(prompt);
end

prompt2 = "Give the degree of the polynom: " ;
m = input(prompt2);

for j=0:m
    promptString = sprintf('Please enter the
        coefficient of x^%d : ', j);
    input2 = input(promptString);
    s(j+1) = input2;
end

syms f(x,y) g(x,y) h1(x)
f(x,y) = x+y;
g(x,y) = x-y;

```

```
h=sym(zeros(1, m+1));
for k=0:m
    h(k+1)= s(k+1)*x^(k);
end

h1(x)=0;
for i=1:length(s)
    h1(x)= h1(x)+h(i);
end

q1=h1(f(x,y));
q2=h1(g(x,y));
syms p1(x,y) p2(x,y)

p1(x,y)=(q1+q2)/2;
p2(x,y)=(q1-q2)/(2*y);

while 1
    promptString2 = sprintf('Please give n between 1
        and %d : ', p);
    n = input(promptString2);
    X=jacobi(n,p);  % Legendre symbole calculation

    while X ~= -1
        fprintf('%d is quadratic residue (mod p),
            choose another n: \n', n);
        n = input(promptString2);
        X=jacobi(n,p);
    end
```

```
51
52      n1=p1(x,sqrt(n));
53      n2=p2(x,sqrt(n));
54      R=resultant(n1,n2);
55
56      if R ~= 0 || mod(R,p) ~= 0
57          fprintf('The quadratic non-residue n(=%d) is
                admissible to f(x)\n ',n);
58          break;
59      else
60          fprintf('\n The quadratic non-residue n(=%d)
                does NOT admissible to f(x), Choose another
                n\n \n ',n);
61      end
62  end
```

4 Multiple layers of security

Assume we have a prime p and a polynomial f, and the sequence $E_p = \{e_1, e_2, \ldots, e_p\}$ is given by Construction 1, that is

$$e_n = \begin{cases} \left(\frac{f(n)}{p}\right) & \text{for } (f(n), p) = 1, \\ +1 & \text{for } p \mid f(n). \end{cases}$$

This sequence's correlation can be high, and we can help by finding the quadratic non-residue n that is admissible to f using the algorithm described in Chapter 2. The polynomial $g(x)$ is then defined by the formula

$$g(x) = ((x+a)^2 - n)f(x),$$

where a is any element of $\mathbb{F}_p$. Furthermore, the following formula defines the sequence $F_p = (f_1, f_2, \ldots, f_p)$:

$$f_n = \begin{cases} \left(\frac{g(n)}{p}\right) & \text{for } (g(n), p) = 1, \\ +1 & \text{for } p \mid g(n). \end{cases} \tag{7}$$

The new F_p sequence has low pseudorandom measures due to Theorem 2; nonetheless, is there a method to find a weak point in this new construction that may cause problems in applications? A skilled code breaker may be able to find the values of a and n in the definition of the polynomial $g(x)$ (for example, by going through all p^2 cases) and then observe that the sequence E_p is the same as

$$\left\{ \left(\frac{(1+a)^2 - n}{p}\right) f_1, \left(\frac{(2+a)^2 - n}{p}\right) f_2, \ldots, \left(\frac{(p+a)^2 - n}{p}\right) f_p. \right\}$$

So, while the sequence F_p's pseudorandom measures are optimal, a very simple operation can be used to derive back the original E_p sequence, which may have a high correlation. This is aided by producing more admissible quadratic non-residues $n_1, n_2, \ldots, n_r$ and defining the polynomial $g(x)$ by

$$g(x) = ((x + a_1)^2 - n_1)((x + a_2)^2 - n_2) \cdots ((x + a_r)^2 - n_r) f(x),$$

where $a_1, a_2, \ldots, a_t$ are arbitrary elements of $\mathbb{F}_p$. The sequence F_p is defined the same way as before by (7). All pairs (a_i, n_i) as $i = 1, 2, \ldots, r$ can take on too many values, making the strategy of the code breaker described in this chapter ineffective. In practice, we believe that the choice $r = 20$ is already safe.

References

[1] *Measures of Pseudorandomness for Finite Sequences: Minimal Values*, Combinatorics Probability and Computing 15 (1-2) (2006).

[2] N. Alon, Y. Kohayakawa, C. Mauduit, C. G. Moreira and V. Rödl, *Measures of pseudorandomness for finite sequences: typical values*, Proceedings of the London Mathematical Society 95 (3) (2007), 778-812,

[3] V. Anantharam, *A technique to study the correlation measures of binary sequences*, Discrete Math. 308, 24 (2008), 6203 -6209.

[4] J. Cassaigne, C. Mauduit and A. Sárközy *On finite pseudorandom binary sequences VII: The measures of pseudorandomness*, Acta Arith. 103 (2) (2001), 97-118.

[5] L. Goubin, C. Mauduit, A. Sárközy, *Construction of large families of pseudorandom binary sequences*, Journal of Number Theory 106 (1) (2004), 56-69

[6] K. Gyarmati, *On the correlation of binary sequences*, Studia Sci. Math. Hungar. 42 (2005), 59-75.

[7] K. Gyarmati and C. Mauduit, *On the correlation of binary sequences, II*, Discrete Math. 312 (2012), 811-818.

[8] J. Hoffstein, D. Lieman, *The distribution of the quadratic symbol in function fields and a faster mathematical stream cipher*, Progress in Computer Science and Applied Logic, Vol. 20, Birkhäuser, Verlag, Basel, 2001; pp. 59-68.

[9] C. Mauduit and A. Sárközy, *On finite pseudorandom binary sequences, I. Measures of pseudorandomness, the Legendre symbol*, Acta Arithmetica 82 (4) (1997), 365-377.

[10] A. Weil, *Sur les courbes algébriques et les variétés qui s'en déduisent*, Act. Sci. Ind. 1041, Hermann, Paris, 1948.

[11] MATLAB version 9.7.0. Natick, Massachusetts: The MathWorks Inc., 2018.

Katalin Gyarmati
Eötvös Loránd University, Institute of Mathematics,
H-1117 Budapest Pázmány Péter sétány 1/C, Hungary
Email: katalin.gyarmati@gmail.com

Károly Müllner
Eötvös Loránd University, Institute of Mathematics,
H-1117 Budapest Pázmány Péter sétány 1/C, Hungary
Email: mullni@student.elte.hu