Calculation of the Quality Parameter of Digital Nets and Application to Their Construction

Gottlieb Pirsic

Institute of Discrete Mathematics, Austrian Academy of Sciences, Sonnenfelsgasse 19, A-1010 Vienna, Austria
E-mail: Gottlieb.Pirsic@oeaw.ac.at

and

Wolfgang Ch. Schmid

Department of Mathematics, University of Salzburg, Hellbrunnerstr. 34,

Received February 17, 2000; accepted June 9, 2000; published online September 14, 2001

In quasi-Monte Carlo methods, point sets of low discrepancy are crucial for accurate results. A class of point sets with low theoretic upper bounds of discrepancy are the digital point sets known as digital $(t, m, s)$-nets which can be implemented very efficiently. The parameter $t$ is indicative of the quality; i.e., small values of $t$ lead to small upper bounds of the discrepancy. We introduce an effective way to establish this quality parameter $t$ for digital nets constructed over arbitrary finite fields and give an application to the construction of digital nets of high quality.

Key Words: quasi-Monte Carlo methods; low-discrepancy point sets; implementation.

1. INTRODUCTION

A central issue in quasi-Monte Carlo methods is the effective construction of low-discrepancy point sets and sequences. The most powerful
current methods are based on the concepts of \((t, m, s)\)-nets and \((t, s)\)-sequences. A detailed theory was developed in Niederreiter [6] (see also Chapter 4 of Niederreiter [8] for a survey of this theory).

These \((t, m, s)\)-nets (resp. \((t, s)\)-sequences) in a base \(b\) provide point sets of \(b^m\) points (resp. infinite sequences) in the half-open \(s\)-dimensional unit cube \(I^s := [0, 1)^s, s \geq 1\). They are extremely well distributed if the quality parameters \(t \in \mathbb{N}_0\) are "small." We follow Niederreiter [8] in our basic notation and terminology.

**Definition 1.1.** Let \(b \geq 2\), \(s \geq 1\), and \(0 \leq t \leq m\) be integers. Then a point set consisting of \(b^m\) points of \(I^s\) forms a \((t, m, s)\)-net in base \(b\) if every subinterval \(J = \prod_{i=1}^s [a_i b^{-di}, (a_i + 1) b^{-di})\) of \(I^s\) with integers \(d_i \geq 0\) and \(0 \leq a_i < b^{di}\) for \(1 \leq i \leq s\) and of volume \(b^{t-m}\) contains exactly \(b^t\) points of the point set.

Until now all construction methods which are relevant for applications in quasi-Monte Carlo methods are the so-called digital methods. Another reason for the importance of the digital method is that digital nets behave extremely well if they are used for the numerical integration of multivariate Walsh series (for example cf. [4]).

To avoid technicalities, in the following we restrict ourselves to digital point sets defined over a finite field \(\mathbb{F}_q\) of prime power order \(q\). For a more general definition (over arbitrary finite commutative rings) see for example Niederreiter [8] or Larcher, Niederreiter, and Schmid [3].

**Definition 1.2.** Let \(q\) be a prime power and let \(s \geq 1\) and \(m \geq 1\) be integers. Let \(C^{(1)}, \ldots, C^{(s)}\) be \(m \times m\) matrices over \(\mathbb{F}_q\). For \(0 \leq n < q^m\) let \(n = \sum_{k=0}^{m-1} a_k q^k\) be the \(q\)-adic representation of \(n\) in base \(q\). Consider an arbitrary bijection \(\varphi: \{0, \ldots, q-1\} \to \mathbb{F}_q\). Let

\[
(y_1^{(i)}(n), \ldots, y_m^{(i)}(n))^T := C^{(i)} \cdot (\varphi(a_0), \ldots, \varphi(a_{m-1}))^T
\]

for \(i = 1, \ldots, s\) and

\[
x_n := (x_n^{(1)}, \ldots, x_n^{(s)}) \in I^s \quad \text{with} \quad x_n^{(i)} := \sum_{k=1}^m \varphi^{-1}(y_k^{(i)}(n)) q^k.
\]

The point set \(x_n, n = 0, 1, \ldots, q^m-1\), is called a digital net constructed over \(\mathbb{F}_q\).

If for some integer \(t\) with \(0 \leq t \leq m\) this point set is a \((t, m, s)\)-net in base \(q\), then it is called a digital \((t, m, s)\)-net constructed over \(\mathbb{F}_q\).
For digital nets, the quality parameter $t$ can also be determined in terms of linear algebra. Sometimes it is convenient to consider the matrices $C^{(1)}, \ldots, C^{(s)}$ as a two-parameter system of vectors in $\mathbb{F}_q^n$.

**Definition 1.3.** Let $d$ be an integer with $0 \leq d \leq m$. The system $\{c^{(i)}_j \in \mathbb{F}_q^n : 1 \leq j \leq m, 1 \leq i \leq s\}$ of vectors is called a $(d, m, s)$-system over $\mathbb{F}_q$ if for any nonnegative integers $d_1, \ldots, d_s$ with $\sum_{i=1}^s d_i = d$ the vectors $c^{(i)}_j$, $1 \leq j \leq d_i$, $1 \leq i \leq s$, are linearly independent over $\mathbb{F}_q$. (The empty set is considered linearly independent.)

Then we arrive at the following characterization of digital nets (see [10, Lemma 3]).

**Lemma 1.1.** The $m \times m$ matrices $C^{(1)}, \ldots, C^{(s)}$ provide a digital $(m-d, m, s)$-net constructed over $\mathbb{F}_q$ if and only if the system $\{c^{(i)}_j : 1 \leq j \leq m, 1 \leq i \leq s\}$ of their row vectors is a $(d, m, s)$-system over $\mathbb{F}_q$.

In Section 2 we propose an algorithm which determines the quality parameter of digital nets. This algorithm improves and generalizes an earlier version for the binary field to arbitrary finite fields.

In Section 3 we apply this algorithm to generalize the work in [2] and [13] and to construct digital nets of high quality.

## 2. CALCULATION OF THE QUALITY PARAMETER

There are several reasons why the knowledge of the exact value of the parameter $t$ is of interest. First of all, the discrepancy depends strongly on $t$; that is, we have the following bound (cf. [8, Theorem 4.10]).

**Theorem 2.1.** The star-discrepancy $D^*_b(P)$ of a $(t, m, s)$-net $P$ in base $b$ with $m > 0$ satisfies

$$ND^*_b(P) \leq B(s, b) \cdot b^t \cdot (\log N)^{-1} + O(b^t(\log N)^{-2}),$$

with a known constant $B(s, b)$ and the implied constant of the Landau symbol also depending only on $s$ and $b$.

Of course, the discrepancy (and hence $t$) is relevant to the theory of equidistribution and its application in quasi-Monte Carlo methods. By the Koksma–Hlawka inequality (see for example [8, Theorem 2.11]) the discrepancy is directly related to the error in numerical integration. Therefore it is important to know how small $t$ is for a particular $(t, m, s)$-net in order to know how large the integration error may be.
Furthermore, \((t, m, s)\)-nets are inherently objects of a combinatorial nature. There are relations to other combinatorial objects such as linear codes or orthogonal arrays, and their respective parameters are closely linked to \(t\), i.e., if a good \((t, m, s)\)-net exists, other combinatorial objects with good parameters exist.

For various construction methods of \((t, m, s)\)-nets there are only theoretical upper bounds on \(t\) that may be too pessimistic. The calculation of the exact value of \(t\) may indicate how accurate the theoretical bounds are.

An earlier approach to calculate \(t\) in the binary case can be found in [12], where the algorithm was applied to investigate certain digital sequences. It also was used as a subroutine in the “shift nets” construction method [11] and for improvements of the “Salzburg Tables” [13]. The latter will also be considered in more detail in the next section.

Here we generalize this previous algorithm to arbitrary finite fields \(\mathbb{F}_q\). Its key idea is the relation of \((t, m, s)\)-nets to \((d, m, s)\)-systems as stated in Lemma 1.1. We also build upon that and use the basic structure of the previous algorithm.

The program consists of an outer loop running through all \(d\) from 1 to \(m\) which tests whether the given matrices form a \((d, m, s)\)-system. This is achieved by an inner loop that runs through all partitions \(d_1, \ldots, d_s\) of \(d\) into \(s\) parts. Within that loop, the first \(d_1\) row vectors of the first matrix, the first \(d_2\) of the second, etc. are collected in a set of vectors that is checked for linear independence. If it is linearly dependent, the matrices do not form a \((d, m, s)\)-system and the maximum value \(d'\) such that they form a \((d', m, s)\)-system is \(d-1\). Then \(t := m - (d-1)\) is the minimum value of \(t'\) such that the matrices generate a \((t', m, s)\)-net and \(t\) is returned to the main program. (See Algorithm 1 for a description in pseudo-code.)

**Algorithm 1** (The basic algorithmic structure).

1. read matrices \(C^{(1)}, \ldots, C^{(s)}\) from file
2. FOR \(d = 1\) to \(m\)
3. FOR ALL partitions of \(d\) in \(s\) parts
4. compose subset \(\{v_1, \ldots, v_d\}\) of vectors
5. IF the subset is linearly dependent
6. exit and return \(t := m - d + 1\)
7. ENDIF
8. ENDFOR
9. ENDFOR
In the version for \(q = 2\) the linear independence was checked by traversing all linear combinations of the \(d\) vectors using the Gray code. This is a very efficient method of running through all linear combinations as only one vector has to be added at each step.

In the generalized version we extended this Gray code method to arbitrary prime bases. As in the case of \(q = 2\), care is taken that as \(d\) increases, no linear combination is checked more than once; i.e., we verify that the \(d_1\)th vector taken from the first matrix, the \(d_2\)th of the second, etc. has a non-zero coefficient in the linear combination.

We also used another method (which works in arbitrary prime power base \(q\)) to check for linear independence; namely, reduction of the matrix of the selected subset \(\{v_1, \ldots, v_d\}\) of vectors to row echelon form by Gauss elimination and subsequent determination of its rank. This proved to be significantly more efficient for higher bases. Furthermore, this allowed for the desired generalization to arbitrary finite fields: Once the basic vector operations are implemented, the elimination subroutine does not change substantially, whereas the Gray code method would have called for additional calculations.

The respective computational worst case complexities of the employed methods are presented in the following theorem.

**Theorem 2.2.** The number of vector additions (\(\oplus\)) and scalar multiplications (\(\odot\)) in \(\mathbb{F}_q^m\) required for the determination of the quality parameter \(t\) of a digital \((t, m, s)\)-net over \(\mathbb{F}_q\) are at most:

1. **Gray code method (\(\oplus\)):**
   
   \[
   \sum_{d=1}^{m-t} \sum_{r=1}^{d} \binom{s}{r} \binom{d-1}{r-1} (q-1)^r q^{d-r} = O((1+q)^{m-t+s-1}).
   \]

2. **Gauss elimination method (\(\oplus, \odot\)):**
   
   \[
   \sum_{d=1}^{m-t} \binom{d+s-1}{d} \frac{d(d-1)}{2} = O(m^2 q^{m-t+s-1}).
   \]

**Proof.** The sum in (i) is the number of linear combinations to investigate and for each of those just one vector addition is necessary (the details are given in [12, Lemma 2]).

The estimate in (ii) follows in a similar way: The number of partitions of \(d\) into \(s\) parts, which is the number of times the inner loop of the algorithm is run through to perform the test for linear independence, is \(\binom{d+s-1}{d}\). In the linear independence test, the number of vector additions and scalar multiplications for reduction of \(d\) vectors to row echelon form is at most
\( \frac{d(d-1)}{2} \) for both operations. Therefore, summation over \( d \) leads to the given total number of operations.

The asymptotic formula of (i) follows by

\[
\sum_{d=1}^{m-t} \sum_{r=1}^{d} \binom{s}{r} \left( \binom{d-1}{r-1} \right) (q-1)^{q^{d-r}} \leq \sum_{d=1}^{m-t} q^d \sum_{r=1}^{d} \binom{s}{r} \left( \binom{d-1}{r-1} \right) = \sum_{d=1}^{m-t} \left( \frac{d+s-1}{d} \right) q^d = O \left( \sum_{d=1}^{m-t+s-1} \binom{m-t+s-1}{d} q^d \right) = O(1+q)^{m-t+s-1},
\]

and likewise for (ii)

\[
\sum_{d=1}^{m-t} \binom{d+s-1}{d}^{\binom{d}{2}} = O \left( \sum_{d=1}^{m-t+s-1} \binom{m-t+s-1}{d} \right) = O(m^{-t+s-1}).
\]

The above complexity results are also mirrored in the runtimes: For small \( q \) and \( m \) sometimes the Gray code method still was ahead, whereas for larger \( q \) the elimination method outperformed it considerably, needing about the same amount of time regardless of the base. In Table I, we list some runtimes that illustrate these effects. In the first column, the parameters \((t, m, s)\) are given and the subscript indicates the base \( q \). The second column shows the time needed for the algorithm using Gray code, and the third for the algorithm using Gauss elimination.

In the runtimes of the algorithm using Gray code we observe a strong dependency on the base \( q \). In the other algorithm the dependency is stronger on \( m-t \), since not only the number of partitions but also the number of operations in the Gauss elimination is dependent on \( m-t \).

<table>
<thead>
<tr>
<th>Gray</th>
<th>Gauss</th>
</tr>
</thead>
<tbody>
<tr>
<td>(9, 14, 14) (_q)</td>
<td>0.07</td>
</tr>
<tr>
<td>(4, 16, 7) (_q)</td>
<td>6.29</td>
</tr>
<tr>
<td>(5, 12, 9) (_q)</td>
<td>10.31</td>
</tr>
<tr>
<td>(0, 8, 5) (_q)</td>
<td>66.21</td>
</tr>
</tbody>
</table>
This algorithm can be used to examine previously constructed nets or to compare different construction methods, with respect to the actual value of $t$ and in comparison with its theoretical bounds. A simple modification of the algorithm allows to make investigations of lower-dimensional projections of a digital net (see for example [14]). We can also employ the algorithm in extensive computer searches for nets of high quality, similar to those for the “Salzburg Tables.”

3. CONSTRUCTION OF DIGITAL NETS OVER $\mathbb{F}_q$

We first describe the special family of digital $(t, m, s)$-nets that was introduced in Niederreiter [7]. Our notation follows [8, Section 4.4].

Let $\mathbb{F}_q((x^{-1}))$ be the field of formal Laurent series over $\mathbb{F}_q$ in the variable $x^{-1}$. Thus, the elements of $\mathbb{F}_q((x^{-1}))$ have the form $\sum_{k=-w}^{\infty} b_k x^{-k}$, where $w$ is an arbitrary integer and all $b_k \in \mathbb{F}_q$.

For a given dimension $s \geq 2$, let $f \in \mathbb{F}_q[x]$ with $\deg(f) = m$ and $g = (g_1, \ldots, g_s) \in \mathbb{R}^s_{q,m}$, where $\mathbb{R}^s_{q,m}$ is the set of all polynomials $g \in \mathbb{F}_q[x]$ with $\deg(g) < m$. (Here and in the following we use the convention $\deg(0) = -1$.) Then we consider the expansions

$$g_i(x) = \frac{g_i(x)}{f(x)} = \sum_{k=1}^{\infty} u^{(i)}_k x^{-k} \in \mathbb{F}_q((x^{-1}))$$

for $1 \leq i \leq s$, and define the elements $c^{(i)}_{jr}$ of the matrices $C^{(i)}$ in Definition 1.2 by

$$c^{(i)}_{jr} = u^{(i)}_{j+r} \in \mathbb{F}_q$$

for $1 \leq i \leq s$, $1 \leq j \leq m$, $0 \leq r < m$.

**Definition 3.1.** The construction principle in Definition 1.2 yields a digital $(t, m, s)$-net over $\mathbb{F}_q$ which is denoted by $P(g, f)$. If $g$ is of the special form $g = (1, g^i, \ldots, g^{i-1})$ with $g \in \mathbb{R}^s_{q,m}$ (that is, $g_i \equiv g^{i-1} \pmod{f}$) we will denote the point set by $P^{(i)}(g, f)$.

**Remark 3.1.** The matrix

$$C_{g, f} := C^{(i)} = \begin{pmatrix} u^{(i)}_1 & u^{(i)}_2 & \cdots & u^{(i)}_{m-1} & u^{(i)}_m \\ u^{(i)}_2 & u^{(i)}_3 & \cdots & u^{(i)}_m & u^{(i)}_{m+1} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ u^{(i)}_{m-1} & u^{(i)}_m & \cdots & u^{(i)}_{2m-3} & u^{(i)}_{2m-2} \\ u^{(i)}_m & u^{(i)}_{m+1} & \cdots & u^{(i)}_{2m-2} & u^{(i)}_{2m-1} \end{pmatrix}$$

is called the Hankel matrix associated with the linear recurring sequence $(u^{(i)}_1, u^{(i)}_2, \ldots)$. If $f(x)$ is monic, it is the characteristic polynomial of the
sequence. When \( \gcd(g_i, f) = 1 \), \( f(x) \) is called the minimal polynomial of the linear recurring sequence (see for example [8, Appendix A]) and \( C_{g_i,f} \) is nonsingular (see [5, Theorem 6.75]).

The following quantity and the subsequent lemma play a crucial role.

**Definition 3.2.** The figure of merit \( \rho(g, f) \) is given by

\[
\rho(g, f) = s - 1 + \min_{i=1}^{s} \deg(h_i),
\]

where the minimum is extended over all nonzero \( s \)-tuples \( (h_1, \ldots, h_s) \in R_{q,m}^s \) for which \( f \) divides \( \sum_{i=1}^{s} g_i h_i \).

We note the following result of Niederreiter (cf. [8, Theorem 4.42]).

**Lemma 3.1.** The point set \( P(g, f) \) is a digital \((t, m, s)\)-net in base \( q \) with \( t = m - \rho(g, f) \).

Existence theorems for large figures of merit and the history of the search for binary “optimal polynomials” (polynomials \( g \in R_{2,m}^s \) providing point sets \( P^{(s)}(g, f) \) with a large figure of merit) are given in [13]. The following facts for prime power base \( q \) are similar to the binary case.

For \( f(x) = x^m + f_{m-1}x^{m-1} + \cdots + f_0 \) and \( g_i(x) = g_i^{(i)}x^{m-1} + \cdots + g_0^{(i)} \), the elements of \( C_{g_i,f} \) can be calculated by the following recursion:

\[
\begin{align*}
    u_1^{(i)} &= g_0^{(i)}, \\
    u_j^{(i)} &= g_{m-j}^{(i)} - \sum_{l=1}^{j-1} u_l^{(i)} f_{m+l-j} \text{ in } \mathbb{F}_q & \text{ for } j = 2, \ldots, m, \\
    u_j^{(i)} &= - \sum_{l=j-m}^{j-1} u_l^{(i)} f_{m+l-j} \text{ in } \mathbb{F}_q & \text{ for } j = m+1, \ldots, 2m-1. 
\end{align*}
\]

**Remark 3.2.** By generalizing [13, Lemmas 8–10] to arbitrary prime power bases \( q \) one can easily compute matrices \( (D_{g_1,f}, \ldots, D_{g_s,f}) \) of very simple form providing the same point set as \( (C_{g_1,f}, \ldots, C_{g_s,f}) \).

**Theorem 3.1.**

(i) For irreducible \( f(x) \), we may restrict ourselves to \( s \)-tuples of the form \( g = (1, g_2, \ldots, g_s) \).
(ii) For \( \gcd(g_i, f) = 1 \), \( 2 \leq i \leq s \), the one-dimensional projection of \( P(g, f) \) to the \( i \)th coordinate is a digital \((0, m, 1)\)-net. In particular, this is true for all one-dimensional projections of \( P^{(s)}(g, f) \) when \( \gcd(g, f) = 1 \).

(iii) The quality parameter of \( P(g, f) \) satisfies \( t \geq m - 1 - \min_{2 \leq i \leq s} (\deg(g_i)) \).

**Proof.**

(i) We may assume \( g_1 \neq 0 \). Since \( \mathbb{F}_q[x]/(f) \) is a field, there exists a \( g_2 \) with \( g_1 \cdot g_2 \equiv 1 \pmod{f} \), and \( \rho(g, f) = \rho(g \cdot g_2, f) \) holds.

(ii) This is a consequence of the facts given in Remark 3.1 and of Lemma 1.1.

(iii) By (i) we can assume \( C^{(1)} = C_{1, f} \) which is a triangular matrix. For \( k = \deg(g_i) = \min_{2 \leq j \leq s} (\deg(g_j)) \) we have \( u_{m-1}^{(i)} = u_{m-2}^{(i)} = \cdots = u_{k+1}^{(i)} = 0 \). Therefore, the first row vector of \( C^{(i)} \) is a linear combination of the first \( k+1 \) row vectors of \( C^{(1)} \). By Lemma 1.1, \( P(g, f) \) is not a digital \((m-k-2, m, s)\)-net.

In our search for digital \((t, m, s)\)-nets over \( \mathbb{F}_q \) with small quality parameters \( t \) we first fixed an irreducible polynomial \( f \in \mathbb{F}_q[x] \) with \( \deg(f) = m \). We only considered point sets of the form \( P^{(i)}(g, f) \); that is, in every dimension we only searched among a certain selection of polynomials \( g(x) \in \mathbb{R}_{q, m} \). For small quality parameters \( t < k \leq m \), Theorem 3.1(iii) allows the consideration of polynomials \( g \) with \( \deg(g') \geq m-k \) for \( 1 \leq i < s \).

We constructed the matrices \( C_{g, f}^{(i)} \), \( 0 \leq i < s \) by the above recursion and then used the algorithm described in Section 2.

For each dimension \( 2 \leq s \leq 50 \), we calculated “optimal” polynomials in a search (for some small \( m \) we performed an exhaustive search) with \( m \leq 19 \).

### TABLE II

<table>
<thead>
<tr>
<th>Irreducible polynomial</th>
<th>( s = 28 )</th>
<th>( s = 29 )</th>
<th>( s = 30 )</th>
<th>( s = 31 )</th>
<th>( s = 32 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( m = 14 )</td>
<td>5579194</td>
<td>2398962</td>
<td>1538726</td>
<td>667496</td>
<td>2690426</td>
</tr>
<tr>
<td>( m = 15 )</td>
<td>19582546</td>
<td>8557930</td>
<td>6100671</td>
<td>12404723</td>
<td>1198478</td>
</tr>
<tr>
<td>( m = 16 )</td>
<td>76038598</td>
<td>3269952</td>
<td>36826732</td>
<td>24046410</td>
<td>5549267</td>
</tr>
<tr>
<td>( m = 17 )</td>
<td>168473041</td>
<td>27870947</td>
<td>69179318</td>
<td>127927207</td>
<td>125684131</td>
</tr>
<tr>
<td>( m = 18 )</td>
<td>442795727</td>
<td>104655530</td>
<td>193805785</td>
<td>248624063</td>
<td>7004447</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Irreducible polynomial</th>
<th>( s = 29 )</th>
<th>( s = 30 )</th>
<th>( s = 31 )</th>
<th>( s = 32 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( m = 14 )</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>( m = 15 )</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>9</td>
</tr>
<tr>
<td>( m = 16 )</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>9</td>
</tr>
<tr>
<td>( m = 17 )</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>( m = 18 )</td>
<td>10</td>
<td>11</td>
<td>11</td>
<td>11</td>
</tr>
<tr>
<td>( m = 19 )</td>
<td>10</td>
<td>11</td>
<td>11</td>
<td>11</td>
</tr>
<tr>
<td>( m = 20 )</td>
<td>10</td>
<td>11</td>
<td>11</td>
<td>11</td>
</tr>
<tr>
<td>( m = 21 )</td>
<td>10</td>
<td>11</td>
<td>11</td>
<td>11</td>
</tr>
<tr>
<td>( m = 22 )</td>
<td>10</td>
<td>11</td>
<td>11</td>
<td>11</td>
</tr>
<tr>
<td>( m = 23 )</td>
<td>10</td>
<td>11</td>
<td>11</td>
<td>11</td>
</tr>
<tr>
<td>( m = 24 )</td>
<td>10</td>
<td>11</td>
<td>11</td>
<td>11</td>
</tr>
<tr>
<td>( m = 25 )</td>
<td>10</td>
<td>11</td>
<td>11</td>
<td>11</td>
</tr>
<tr>
<td>( m = 26 )</td>
<td>10</td>
<td>11</td>
<td>11</td>
<td>11</td>
</tr>
<tr>
<td>( m = 27 )</td>
<td>10</td>
<td>11</td>
<td>11</td>
<td>11</td>
</tr>
<tr>
<td>( m = 28 )</td>
<td>10</td>
<td>11</td>
<td>11</td>
<td>11</td>
</tr>
<tr>
<td>( m = 29 )</td>
<td>10</td>
<td>11</td>
<td>11</td>
<td>11</td>
</tr>
<tr>
<td>( m = 30 )</td>
<td>10</td>
<td>11</td>
<td>11</td>
<td>11</td>
</tr>
<tr>
<td>( m = 31 )</td>
<td>10</td>
<td>11</td>
<td>11</td>
<td>11</td>
</tr>
<tr>
<td>( m = 32 )</td>
<td>10</td>
<td>11</td>
<td>11</td>
<td>11</td>
</tr>
</tbody>
</table>

**QUALITY AND CONSTRUCTION OF DIGITAL NETS**
(resp. \( m \leq 17 \)) in the case of \( F_3 \) (resp. \( F_5 \)). With such polynomials we are able to construct low-discrepancy point sets of \( 3^{19} \approx 10^9 \) resp. \( 5^{17} \approx 10^{12} \) points. For each \( m \) and \( s \) we have tabulated the smallest quality parameter \( t \) and the polynomial providing this value, as well as the irreducible polynomial \( f(x) \) used. The tables can be obtained from the authors. As an example, for the bases \( q = 3, 4, \) and \( 5 \), a small selection is given in Tables II–IV. The first value in each box is the value of \( t \). The second value in each box is the “polynomial number” \( g^* \) of the polynomial \( g \in R_{q,m} \) that gives the value of \( t \). If \( g(x) = \sum_{j=0}^{\infty} a_j x^j \in F_q[x] \), then the corresponding polynomial number is given by \( g^* = \sum_{j=0}^{\infty} \varphi^{-1}(a_j) q^j \), with \( \varphi \) as in Definition 1.2. For each \( m \) we have used the same irreducible polynomial \( f(x) \) for every dimension \( s \). Its polynomial number \( f^* \) is given at the begin of each row.

Subsequently we present two figures, one for base 3 (Fig. 1) and one for base 5 (Fig. 2), comparing the quality parameters of our new method with previously known best parameters as published in [1]. For our comparisons, we have updated these tables with the latest improvements deduced from Niederreiter–Xing sequences [9, Table 2].

### Table III

Optimal Polynomials and Values of \( t \) in \( F_4 \)

<table>
<thead>
<tr>
<th>Irreducible polynomial ( m = 8 )</th>
<th>( s = 25 )</th>
<th>( s = 30 )</th>
<th>( s = 35 )</th>
<th>( s = 40 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>122451</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>( m = 12 )</td>
<td>6</td>
<td>7</td>
<td>?</td>
<td>?</td>
</tr>
<tr>
<td>29729585</td>
<td>10955956</td>
<td>16117311</td>
<td>163675</td>
<td>11381537</td>
</tr>
<tr>
<td>( m = 15 )</td>
<td>8</td>
<td>8</td>
<td>9</td>
<td>9</td>
</tr>
<tr>
<td>1468696943</td>
<td>148144545</td>
<td>1010757900</td>
<td>368224410</td>
<td>769342713</td>
</tr>
</tbody>
</table>

### Table IV

Optimal Polynomials and Values of \( t \) in \( F_5 \)

<table>
<thead>
<tr>
<th>Irreducible polynomial ( m = 14 )</th>
<th>( s = 28 )</th>
<th>( s = 29 )</th>
<th>( s = 30 )</th>
<th>( s = 31 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>7964010618</td>
<td>7</td>
<td>7</td>
<td>7</td>
<td>7</td>
</tr>
<tr>
<td>( m = 15 )</td>
<td>7</td>
<td>8</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>36306772733</td>
<td>202564256</td>
<td>128448745</td>
<td>128448745</td>
<td>4266757</td>
</tr>
<tr>
<td>( m = 16 )</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>159610613043</td>
<td>1903254357</td>
<td>1810160397</td>
<td>1922918640</td>
<td>976932671</td>
</tr>
<tr>
<td>( m = 17 )</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>9</td>
</tr>
<tr>
<td>1134064288192</td>
<td>202956538</td>
<td>348303940</td>
<td>1008956512</td>
<td>1337551289</td>
</tr>
</tbody>
</table>
FIG. 1. Improvements for ternary digital nets obtained by the new method.

FIG. 2. Improvements for quinary digital nets obtained by the new method.

TABLE V
Comparisons for Nets over $\mathbb{F}_q$ with Nets Deduced from Niederreiter–Xing Sequences

<table>
<thead>
<tr>
<th>$m$ ( \backslash ) s</th>
<th>5</th>
<th>10</th>
<th>15</th>
<th>20</th>
<th>25</th>
<th>30</th>
<th>35</th>
<th>40</th>
<th>45</th>
<th>50</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>12</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>6</td>
<td>7</td>
<td>7</td>
<td>7</td>
<td>7</td>
<td>7</td>
</tr>
<tr>
<td>15</td>
<td>4</td>
<td>6</td>
<td>7</td>
<td>7</td>
<td>8</td>
<td>8</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>10</td>
</tr>
<tr>
<td>NX, $m \geq t$</td>
<td>0</td>
<td>2</td>
<td>4</td>
<td>6</td>
<td>9</td>
<td>13</td>
<td>16</td>
<td>17</td>
<td>25</td>
<td>25</td>
</tr>
</tbody>
</table>
In both figures, black indicates an improvement of $t$ by 2, gray an improvement of $t$ by 1, and white indicates that no improvement occurred.

As mentioned in Section 2, the second algorithm also works in prime power bases. We have carried out several calculations over $\mathbb{F}_4$, $\mathbb{F}_8$, $\mathbb{F}_9$, and $\mathbb{F}_{27}$. Apart from some sporadic values there only exist the (excellent) upper bounds deduced from Niederreiter–Xing sequences. In Table V we compare the quality parameters obtained by our method for some selected values in base $q = 4$ with the net parameters deduced from Niederreiter–Xing sequences as they were given in [9, Table 3]. Note that, by [10, Lemma 2], a quality parameter for sequences in dimension $s$ holds for nets with $q^m$ points, $m \geq t$ arbitrary, in dimension $s + 1$.

ACKNOWLEDGMENTS

The calculations (implementation in C++) were carried out on an SGI Power Challenge GR of the RIST++ (Research Institute for Software Technology at the University of Salzburg). We are grateful to Peter Zinterhof, the head of the RIST++, for his support. In our implementation we made use of the following free software: the GNU C++ compiler and Victor Shoup’s number theoretic library NTL. We thank the respective authors for letting their programs run under public licenses. We also thank Mary Flahive for valuable discussion and proofreading.

REFERENCES


