SIAM J. SCI. COMPUT.
Vol. 30, No. 5, pp. 2635–2654

CONSTRUCTING SOBOL′ SEQUENCES WITH BETTER TWO-DIMENSIONAL PROJECTIONS*

STEPHEN JOE† AND FRANCES Y. KUO‡

Abstract. *Direction numbers* for generating Sobol′ sequences that satisfy the so-called Property A in up to 1111 dimensions have previously been given in Joe and Kuo [*ACM Trans. Math. Software*, 29 (2003), pp. 49–57]. However, these Sobol′ sequences may have poor two-dimensional projections. Here we provide a new set of direction numbers alleviating this problem. These are obtained by treating Sobol′ sequences in d dimensions as (t, d)-sequences and then optimizing the *t-values* of the two-dimensional projections. Our target dimension is 21201.

Key words. Sobol′ sequences, two-dimensional projections, digital nets and sequences, numerical integration, quasi-Monte Carlo methods

AMS subject classifications. 65D30, 65D32

DOI. 10.1137/070709359

1. Introduction.

A popular technique for approximating integrals over the d-dimensional unit cube is to make use of Sobol′ sequences; that is, we approximate the integral

$$\int_{[0,1]^d} f(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x} \qquad \text{by} \qquad \frac{1}{n}\sum_{i=0}^{n-1} f(\boldsymbol{x}_i),$$

where $\boldsymbol{x}_0, \ldots, \boldsymbol{x}_{n-1}$ are n points in $[0,1]^d$ generated according to the method proposed by Sobol′ [23]. A computer implementation of a Sobol′ sequence generator in Fortran 77 was given by Bratley and Fox [2] as Algorithm 659. This implementation allowed the approximation of integrals for dimension d up to 40. It was extended by Joe and Kuo [9] to allow d to go up to 1111 dimensions by having more *primitive polynomials* and more so-called *direction numbers*; these are the main ingredients for generating Sobol′ sequences.

The direction numbers in [9] are such that they satisfy the extra uniformity condition known as *Property A*, introduced by Sobol′ [24]. Geometrically, if the cube $[0,1]^d$ is divided by the planes $x_j = 1/2$ into 2^d equally-sized subcubes, then a sequence of points belonging to $[0,1]^d$ possesses Property A if, after dividing the sequence into consecutive blocks of 2^d points, each one of the points in any block belongs to a different subcube.

Property A is not so useful to have for large d because this uniformity property is based on 2^d points, and it is simply not feasible computationally to approximate an integral using so many points. Also, Property A is not enough to ensure that there are no bad correlations between pairs of dimensions. This issue was raised by Morokoff and Caflisch [13]. In that article they gave an example of a bad pairing of

*Received by the editors November 26, 2007; accepted for publication (in revised form) April 28, 2008; published electronically August 1, 2008.
http://www.siam.org/journals/sisc/30-5/70935.html
†Department of Mathematics, University of Waikato, Private Bag 3105, Hamilton, New Zealand (stephenj@math.waikato.ac.nz).
‡School of Mathematics and Statistics, University of New South Wales, Sydney NSW 2052, Australia (f.kuo@unsw.edu.au). This author's work was supported by the Australian Research Council Queen Elizabeth II Research Fellowship.

dimensions for the initial $2^{12} = 4096$ Sobol′ points; see Figure 1.1. (The parameters for this plot are discussed in section 2.3.) One can see a clear pattern of wiggly strips of points and blank regions with no points inside. When we double the total number of points to 8192, we find that the appearance does not improve much. It is true that, in this case when we add another 8192 points, these additional points fall only where the gaps are and lead to a nice uniform plot of 16384 points, but in other unfortunate cases we may require a huge and impractical number of points to eventually fill in the gaps.

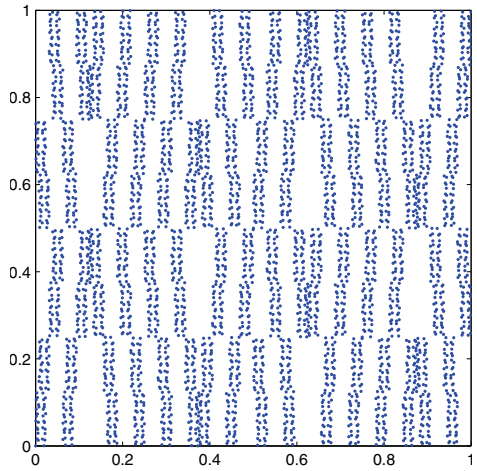

FIG. 1.1. *Example of a bad pairing of dimensions for* 4096 *Sobol′ points* (*t-value is* 6).

The aim of this paper is to provide direction numbers for Sobol′ sequences which do not have bad correlations between pairs of variables. Key to our approach is to make use of the fact that a Sobol′ sequence may be considered to be a (t, d)-*sequence in base* 2, in which (after being divided into consecutive blocks of 2^m points) every block of 2^m points forms a (t, m, d)-*net*. The general theory was developed by Niederreiter (see [14, Chapter 4]); more details are given in the next section. It suffices to say here that the *t-value* is a quality parameter which measures the uniformity of the point sets. For example, the t-value for the plot of $2^m = 2^{12}$ points in Figure 1.1 is 6, which means the following: if we partition the unit square using $2^{m-t} = 2^{12-6} = 2^6$ identical rectangles, then each rectangle contains exactly $2^t = 2^6$ points. Note that the rectangles can vary in shape from narrow strips of size 1 by $1/2^6$ to squares of size $1/2^3$ by $1/2^3$. The smaller the t-value is, the finer the partition can be, and the more uniformly distributed the points are. See the survey article by Schmid [17] for a discussion on the quality of the projections of nets and sequences.

Our approach in this paper is to choose the direction numbers so that (i) Property A holds up to 1111 dimensions (for consistency with [9]), and (ii) the t-values of the two-dimensional (2D) projections of the point sets are minimized in some sense (to be made explicit in section 3). Our target dimension is 21201. The difficulty in our approach lies in determining an appropriate search criterion. We have $\binom{d}{2}$ 2D projections to consider up to dimension d, which is a huge number if d is large. Furthermore, there is the complication of varying m when we consider 2^m points at a time. Before we say more about our search criterion, we first provide some motivation for improving the 2D projections.

There may well be bad higher-dimensional correlations which are difficult to detect. But they may not matter much, as it is often the low-dimensional projections that are important in certain applications. By now it is widely believed that the success of *quasi-Monte Carlo methods* (of which Sobol′ sequences are classic examples) lies in their superior uniformity in the low-dimensional projections, especially the one-dimensional and 2D projections. This is due to the observation that many practical problems, in particular, problems in mathematical finance, have low *effective dimension*—a notion introduced by Caflisch, Morokoff, and Owen [3]. Loosely speaking, this means either that the integrand depends mostly on the initial handful of variables, or that the integrand can be well approximated by a sum of functions with each depending on only a small number of variables at a time. See, for example, [25, 26, 27] and the papers cited therein for some recent literature on this topic.

One way to model the relative importance between variables or groups of variables is to introduce *weights* following Sloan and Woźniakowski [22] (see [21] for a more general setting). With a chosen set of weights, the *generating vectors* for *lattice rules* (a family of quasi-Monte Carlo methods) can be constructed *component-by-component* (the component for the dth dimension is chosen while keeping the components for the previous $d-1$ dimensions held fixed), tailoring to the specific model. See, for example, [5, 6, 10, 15, 20] and the papers cited therein for recent developments on the construction of lattice rules.

Unlike lattice rules which require educated tuning in determining the right parameters for a given model, Sobol′ sequences have always been used as a universal tool regardless of the given problem. The "one size fits all" property of Sobol′ sequences makes them extremely popular for practitioners. However, since there is actually scope in choosing the direction numbers and since many practical applications do have low effective dimension, it makes sense to aim for Sobol′ sequences which do not have bad correlations between pairs of variables.

Returning now to our approach for choosing direction numbers, we take a component-by-component approach focusing on one dimension at a time, thus considering only $d-1$ 2D projections in dimension d, and we aim for small t-values across all 2D projections of 2^m points with m restricted to a finite range. We introduce weights in our search criterion to emphasize the importance of earlier dimensions, and we further tweak our search criterion to take into account our belief that, as m varies, it is important to have $m-t$ as large as possible.

Our search criterion is chosen based on experimental trial and error and borrows recent ideas and techniques from lattice rules. We do not claim to have eliminated all the bad 2D projections. Since we take a "minimax" optimization approach, we simply eliminate the worst ones! Another way of interpreting the results is that, by using a component-by-component approach and by introducing weights, we push the bad 2D projections further along to later dimensions. Of course, other ways of obtaining the direction numbers are also possible; see, for example, [4, 8, 11, 16, 18].

In section 2 we provide some background material on the generation of Sobol′ sequences, the definition of digital nets, the t-values of 2D projections of Sobol′ sequences, and the mathematical formulation of Property A. In section 3, we give full details of the approach used to obtain the new direction numbers. A summary is given in section 4.

2. Background.

2.1. Generating Sobol′ sequences. The algorithm for generating Sobol′ sequences is clearly explained in [2]. Here we give a brief outline of the details. To

generate the jth component of the points in a Sobol′ sequence, we need to choose a primitive polynomial of some degree s_j over the field $\mathbb{Z}_2$,

$$x^{s_j} + a_{1,j}\, x^{s_j-1} + a_{2,j}\, x^{s_j-2} + \cdots + a_{s_j-1,j}\, x + 1, \tag{2.1}$$

where the coefficients $a_{1,j}, a_{2,j}, \ldots, a_{s_j-1,j}$ are either 0 or 1. We define a sequence of positive integers $\{m_{1,j}, m_{2,j}, \ldots\}$ by the recurrence relation

$$\begin{aligned} m_{k,j} := {} & 2a_{1,j}\, m_{k-1,j} \oplus 2^2 a_{2,j}\, m_{k-2,j} \oplus \cdots \oplus 2^{s_j-1} a_{s_j-1,j}\, m_{k-s_j+1,j} \\ & \oplus 2^{s_j} m_{k-s_j,j} \oplus m_{k-s_j,j}, \end{aligned} \tag{2.2}$$

where $\oplus$ is the bit-by-bit exclusive-or operator. The initial values $m_{1,j}, m_{2,j}, \ldots, m_{s_j,j}$ can be chosen freely provided that each $m_{k,j}$, $1 \le k \le s_j$, is odd and less than 2^k. The so-called *direction numbers* $\{v_{1,j}, v_{2,j}, \ldots\}$ are defined by

$$v_{k,j} := \frac{m_{k,j}}{2^k}. \tag{2.3}$$

(With a slight abuse of terminology, we also refer to the numbers $m_{k,j}$ as direction numbers.) Then $x_{i,j}$, the jth component of the ith point in a Sobol′ sequence, is given by

$$x_{i,j} := i_1\, v_{1,j} \oplus i_2\, v_{2,j} \oplus \cdots, \tag{2.4}$$

where i_k is the kth binary digit of $i = (\ldots i_3 i_2 i_1)_2$. Here and elsewhere in the paper, we use the notation $(\cdot)_2$ to denote the binary representation of numbers.

See [2] for a numeric example illustrating the steps for generating Sobol′ sequences. The formula (2.4) corresponds to the original implementation of Sobol′. A more efficient Gray code implementation proposed by Antonov and Saleev [1] can be used in practice.

2.2. Digital nets and sequences. The theory of (t, m, d)-nets and (t, d)-sequences was developed by Niederreiter; see [14, Chapter 4]. Here we review only the key points needed for this paper.

DEFINITION 2.1. *Let $d \ge 1$, $b \ge 2$, and $0 \le t \le m$ be integers. A point set $\mathcal{P}$ consisting of b^m points in $[0,1)^d$ forms a (t, m, d)-net in base b if every subinterval $\prod_{j=1}^{d} [\alpha_j b^{-\beta_j}, (\alpha_j + 1) b^{-\beta_j}) \in [0,1)^d$ of volume b^{t-m}, with integers $\beta_j \ge 0$ and $0 \le \alpha_j < b^{\beta_j}$ for $1 \le j \le d$, contains exactly b^t points of $\mathcal{P}$.*

In less formal language, if there is a point set of b^m points and $[0,1]^d$ is partitioned into b^{m-t} identical rectangular boxes, then this point set forms a (t, m, d)-net if each rectangular box contains exactly b^t points.

Clearly any (t, m, d)-net is a (t', m, d)-net for all t' satisfying $t \le t' \le m$. When talking about the "t-value" of a (t, m, d)-net, we typically refer to the smallest value of t for which this is true, even though we do not say so explicitly.

The t-value is a quality parameter which appears in bounds on the so-called *star discrepancy*. It suffices to say here that, for a (t, m, d)-net with b^m points, the bound depends on $1/b^{m-t}$. Thus for fixed m, the smaller the t-value is, the more uniformly distributed the points are. The technical details may be found in [14, section 4.1].

DEFINITION 2.2. *Let $d \ge 1$, $b \ge 2$, and $t \ge 0$ be integers. A sequence $\{\boldsymbol{x}_i : i \ge 0\}$ of points in $[0,1)^d$ is a (t, d)-sequence in base b if every block of b^m points, $\boldsymbol{x}_{\ell b^m}, \ldots, \boldsymbol{x}_{(\ell+1)b^m - 1}$ for $\ell \ge 0$ and $m \ge t$, forms a (t, m, d)-net in base b.*

Note that the t-value of a (t,d)-sequence provides an upper bound on the t-values of all nets embedded in the sequence. More precisely, if we have a (t,m,d)-net which is one block of b^m points from a (t,d)-sequence, then its t-value is at most m and is smaller than or equal to the t-value of the sequence.

In this framework, a Sobol′ sequence in d dimensions is a (t,d)-sequence in base $b=2$. Providing that the primitive polynomials are distinct, the t-value of the sequence is given by (see, for example, [17])

$$t = \sum_{j=1}^{d} (s_j - 1), \tag{2.5}$$

where, as in the previous subsection, s_j is the degree of the primitive polynomial in dimension j. This suggests that we should use primitive polynomials with as low a degree as possible.

In practice all concrete constructions of (t,m,d)-nets, including nets associated with Sobol′ sequences, are based on the general construction scheme of *digital nets*. For simplicity we restrict ourselves to b being prime in the definition below.

DEFINITION 2.3. *Let b be a prime number and let $d \geq 1$ and $m \geq 1$ be integers. Let $C_{m,1}, \ldots, C_{m,d}$ be $m \times m$ matrices over the finite field $\mathbb{Z}_b$. For each $0 \leq i < b^m$ with base-b representation $i = \sum_{k=1}^{m} i_k\, b^{k-1}$, and for each $1 \leq j \leq d$, define*

$$(x_{i,j,1}, \ldots, x_{i,j,m})^\top := C_{m,j}\,(i_1, \ldots, i_m)^\top,$$

with all arithmetic carried out in $\mathbb{Z}_b$, and set

$$x_{i,j} := \frac{x_{i,j,1}}{b} + \cdots + \frac{x_{i,j,m}}{b^m}.$$

If the point set $\{\boldsymbol{x}_i = (x_{i,1}, \ldots, x_{i,d}) : 0 \leq i < b^m\}$ is a (t,m,d)-net in base b for some integer t with $0 \leq t \leq m$, then it is called a digital (t,m,d)-net over $\mathbb{Z}_b$.

The t-value (remember that we always refer to the smallest value of t) of a digital net is the smallest value of t such that for every possible choice of the integers $r_1, r_2, \ldots, r_d$ satisfying

$$m - t = \sum_{j=1}^{d} r_j, \quad \text{with} \quad 0 \leq r_j \leq m - t,$$

the system of vectors, obtained by taking the first r_1 rows of $C_{m,1}$, the first r_2 rows of $C_{m,2}, \ldots$, and the first r_d rows of $C_{m,d}$, is linearly independent.

From the description for generating Sobol′ sequences in the previous subsection, it is not hard to see that the first 2^m Sobol′ points correspond to a digital net generated by the $m \times m$ matrices $C_{m,j}$ whose kth column contains the binary digits of the direction number $v_{k,j} = (0.v_{k,j,1}v_{k,j,2}\ldots)_2$. More precisely, we have

$$C_{m,j} = \begin{bmatrix} 1 & v_{2,j,1} & v_{3,j,1} & \cdots & v_{m,j,1} \\ 0 & 1 & v_{3,j,2} & \cdots & v_{m,j,2} \\ 0 & 0 & 1 & \cdots & v_{m,j,3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix}. \tag{2.6}$$

The fact that they are upper triangular matrices with 1's down their main diagonals can be easily derived from the definition of the direction numbers. Note that the matrices $C_{m',j}$ for $m' < m$ are embedded in the upper-left corners of the matrices $C_{m,j}$.

The first dimension is a special case in which $m_{k,1} = 1$ for all k. Thus $v_{k,1} = 1/2^k = (0.00\ldots01)_2$ with the 1 in the kth position after the binary point, so that the matrix $C_{m,1}$ is the $m \times m$ identity matrix.

2.3. Two-dimensional projections. We showed in Figure 1.1 an example of a bad 2D projection from the first 4096 points of a Sobol′ sequence. The same plot appeared in [3, 13]. It corresponds to the degree-7 primitive polynomials $x^7 + x^5 + x^4 + x^3 + 1$ and $x^7 + x^5 + x^4 + x^3 + x^2 + x + 1$, together with the two sets of initial direction numbers $(1, 3, 5, 11, 3, 3, 35)$ and $(1, 1, 7, 5, 11, 59, 113)$, respectively. These two polynomials were associated with dimensions 27 and 28 in [13], but with dimensions 27 and 32 in [3], due to a different ordering of the primitive polynomials. Note that by changing any of these initial direction numbers, we may improve the quality of this particular 2D projection dramatically. However, as a result we may introduce other bad 2D projections.

In order to alleviate this problem of bad 2D projections for Sobol′ sequences, we make use of the fact that the quality of a 2D projection is intimately related to the t-value of the digital net corresponding to the particular projection: the smaller the t-value is, the more uniformly distributed the points are.

Let $m \geq 1$ and, for the time being, suppose it is a fixed value. Also, for $j \neq d$, let

$$t(j, d; m)$$

denote the t-value of the digital net corresponding to the (j, d)-projection (that is, the 2D projection of dimensions j and d) of the first 2^m Sobol′ points. Clearly this quantity depends on the choice of the primitive polynomials and the initial direction numbers in dimensions j and d (and hence on $C_{m,j}$ and $C_{m,d}$). There is a natural upper bound on $t(j, d; m)$ given by

$$t(j, d; m) \leq \min(m, s_j + s_d - 2), \tag{2.7}$$

where the second bound holds since the t-value of the net can be no greater than the t-value of the entire 2D Sobol′ sequence, which is $s_j + s_d - 2$ in this case; see (2.5).

There are only two $m \times m$ matrices to consider, namely, $C_{m,j}$ and $C_{m,d}$. We know that $t(j, d; m)$ is the smallest possible value of t such that, under all possible choices of the nonnegative integers r_j and r_d satisfying $m - t = r_j + r_d$, the first r_j rows of $C_{m,j}$ and the first r_d rows of $C_{m,d}$ form a system of linearly independent vectors. A simple pseudocode to calculate the quantity $t(j, d; m)$ is given in Figure 2.1. Note that determining linear independence (or whether the matrix is of full rank) may be done by row reduction in $\mathbb{Z}_2$, with the elementary row operations carried out using bit-by-bit exclusive-or operations. The actual implementation should also make use of the fact that the matrices $C_{m,j}$ and $C_{m,d}$ are upper-triangular and have 1's down their main diagonals.

As we said in the introduction, the t-value is 6 for the plot in Figure 1.1. In Table 2.1 we present the t-values for all 2D projections of the first $2^{12} = 4096$ Sobol′ points up to dimension 28 with the primitive polynomials and direction numbers from [9].

We see that there are 20 occurrences of t-value 5, seven occurrences of t-value 6, and even a t-value of 7. Figure 2.2(a) gives a plot of 4096 points for the

```
function t_value(j, d, m) {
   for t from 0 to m do
     for r_j from 0 to m − t do
       r_d = m − t − r_j
       form a matrix by taking the first r_j rows of C_{m,j}
                           and the first r_d rows of C_{m,d}
       row-reduce the matrix using binary operations
       if <matrix is of full rank> then
         if <end of inner loop reached> then return t_value= t
         else continue inner loop
       else break out of inner loop
 }
```

FIG. 2.1. *Pseudocode for computing* $t(j, d; m)$.

TABLE 2.1
Values of $t(j, d; 12)$ *for Sobol′ points from* [9].

$d\backslash j$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
2	0																										
3	1	1																									
4	2	2	2																								
5	2	1	2	2																							
6	2	1	2	2	2																						
7	3	2	1	2	2	1																					
8	2	3	3	2	3	2	3																				
9	2	1	1	1	2	2	2	3																			
10	2	3	2	1	3	**5**	4	3	**6**																		
11	3	2	4	2	3	3	3	**6**	3	3																	
12	1	1	1	2	3	2	3	3	2	3	3																
13	1	2	1	2	2	2	**5**	4	**5**	2	0	2															
14	3	2	2	2	3	3	4	2	1	3	3	2	3														
15	1	4	2	2	3	3	2	1	2	4	2	4	4	3													
16	2	1	1	3	**5**	**5**	3	2	2	2	3	**5**	3	2	**5**												
17	3	2	2	3	2	2	3	2	2	2	**5**	2	2	2	3	3											
18	3	2	2	1	3	**5**	2	1	2	3	2	3	3	2	1	2	2										
19	1	2	2	2	2	4	3	2	3	3	3	3	1	3	3	1	2	2									
20	2	2	3	2	2	2	1	3	3	1	2	2	2	**5**	2	3	3	2	**6**								
21	3	2	3	3	3	1	3	4	3	2	2	2	1	2	3	2	3	2	2	3							
22	1	2	3	3	**7**	3	3	1	2	**5**	2	2	4	2	3	2	2	3	4	**6**	3						
23	2	2	3	2	**5**	2	2	2	3	3	1	1	1	2	2	**5**	2	4	2	1	3	2					
24	3	2	3	**5**	4	2	2	4	2	3	3	4	4	2	2	**6**	2	**5**	2	3	3	2	3				
25	3	4	2	3	1	2	4	2	4	2	3	4	2	3	1	2	4	2	2	3	3	2	4	2			
26	2	2	3	**5**	1	1	4	1	3	4	2	1	1	2	1	1	2	4	**5**	3	2	2	2	3	**5**		
27	2	2	3	2	2	2	3	2	4	3	3	4	2	3	2	2	3	4	1	2	3	3	2	1	3	2	
28	3	**5**	3	2	3	2	3	**6**	3	1	2	1	1	3	2	2	3	4	**6**	4	2	3	3	2	3	2	**5**

$(19, 28)$-projection in which $t(19, 28; 12) = 6$, while Figure 2.2(b) contains a similar plot for the $(5, 22)$-projection in which $t(5, 22; 12) = 7$. Clearly there are some even worse 2D projections out there than are shown in Figure 1.1! To provide a comparison, we include in Figures 2.2(c) and 2.2(d) the plots for the $(10, 28)$-projection in which $t(10, 28; 12) = 1$ and the $(26, 27)$-projection in which $t(26, 27; 12) = 2$.

We assumed in the preceding discussion that the total number of points, 2^m, is fixed. However, in reality we want our Sobol′ sequence to have good 2D projections for all values of m. Thus the aim in this paper is to choose the direction numbers so that the t-values of all the 2D projections are as small as possible across a range of

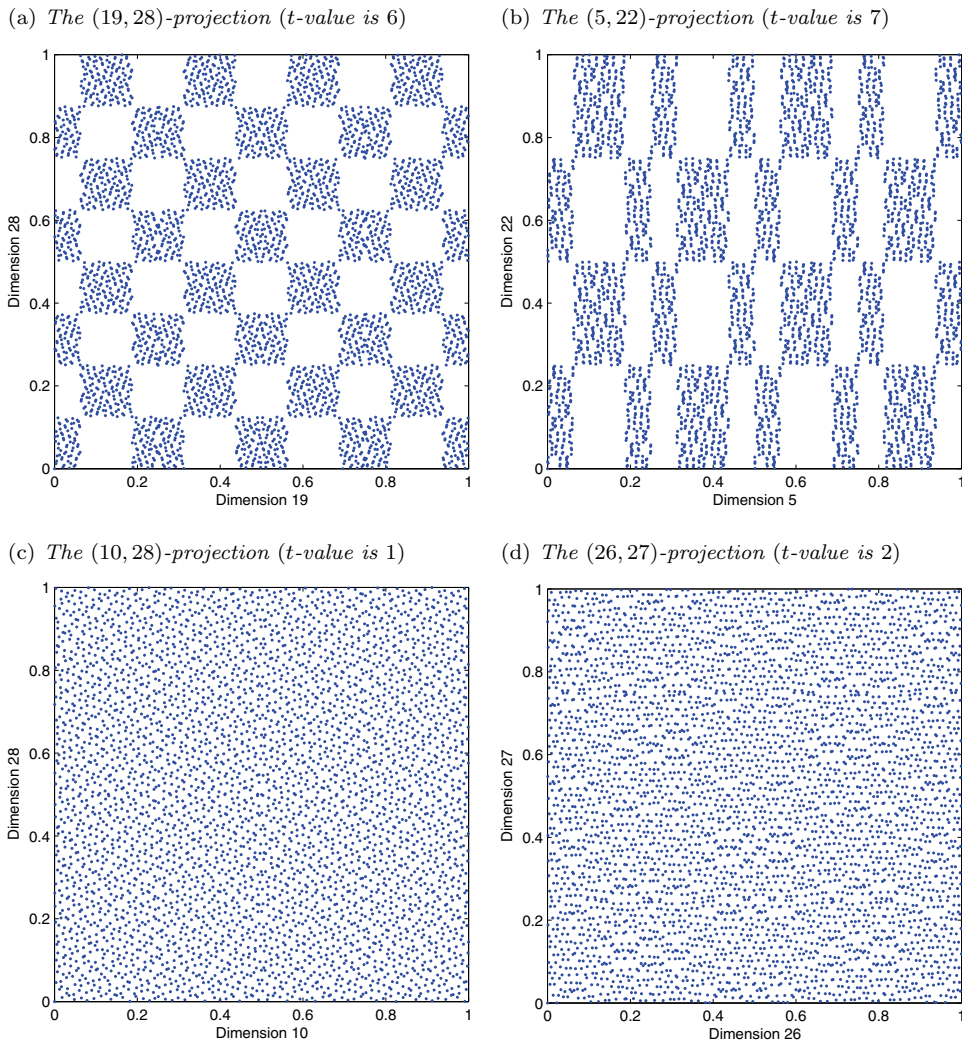

FIG. 2.2. *Some* 2*D projections of* 4096 *Sobol′ points from* [9].

practical values for m. This becomes an optimization problem, and we discuss our approach in section 3.

2.4. Property A. It is shown in [24] that a d-dimensional Sobol′ sequence possesses Property A if and only if

$$\det(V_d) \equiv 1 \pmod 2,$$

where V_d is the $d \times d$ binary matrix defined by

$$V_d := \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 1 & v_{2,2,1} & v_{3,2,1} & \cdots & v_{d,2,1} \\ 1 & v_{2,3,1} & v_{3,3,1} & \cdots & v_{d,3,1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & v_{2,d,1} & v_{3,d,1} & \cdots & v_{d,d,1} \end{bmatrix},$$

with $v_{k,j,1}$ denoting the first digit after the binary point of the direction number $v_{k,j} = (0.v_{k,j,1}v_{k,j,2}\ldots)_2$.

Note that Property A is preserved in dimension d if we reorder the primitive polynomials and corresponding direction numbers within the first d dimensions. (Permuting the rows in a binary matrix does not change its determinant modulo 2.) It is also worth noting that Property A holding in dimension d does not imply that Property A holds for any lower-dimensional projections. More specifically, if Property A holds in all dimensions up to d, that is, $\det(V_{d'}) \equiv 1 \pmod 2$ for all $d' \le d$, and we reorder the primitive polynomials and corresponding direction numbers within the first d dimensions, then the condition $\det(V_d) \equiv 1 \pmod 2$ still holds, but there is no guarantee that $\det(V_{d'}) \equiv 1 \pmod 2$ holds for $d' < d$.

The determinant of V_d can be evaluated by doing row reduction using bit-by-bit exclusive-or operations. More details regarding this calculation that are specific to our approach are discussed in section 3.5.

3. Our approach.

3.1. Ordering the primitive polynomials. The error bounds for Sobol′ sequences given in [23] indicate that we should use primitive polynomials of as low a degree as possible. Our discussion in the previous section regarding the t-values of Sobol′ sequences viewed as digital nets also leads to the same conclusion.

The total number of primitive polynomials of degree s is $\phi(2^s - 1)/s$, where ϕ is Euler's totient function. Following the convention established in [2], we identify the coefficients of a primitive polynomial (2.1) with an integer

$$a_j := (a_{1,j}a_{2,j}\ldots a_{s_j-1,j})_2,$$

so that each primitive polynomial is uniquely specified by its degree s_j together with the number a_j. For example, from $s_j = 7$ and $a_j = 28 = (011100)_2$ we obtain the polynomial $x^7 + x^5 + x^4 + x^3 + 1$; the pair $s_j = 7$ and $a_j = 31 = (011111)_2$ leads to $x^7 + x^5 + x^4 + x^3 + x^2 + x + 1$. These are the two polynomials associated with Figure 1.1.

We arrange the primitive polynomials in increasing order of their degrees, and for those with the same degree we systematically arrange them in increasing order of the numbers a_j. Leaving aside the special case for the first dimension where all the $m_{k,1}$ are 1, we assign one primitive polynomial for each dimension starting from dimension 2. Thus we arrive at dimension 1111 after using all primitive polynomials up to degree 13; this is as far as [9] went. Using all primitive polynomials up to degree 18 gives us 21201 dimensions; this is the target dimension of the present paper.

Note that our ordering in the first 46 dimensions is different from that of [9], which followed the historical ordering in [2]. See the appendix for a list of the primitive polynomials and corresponding direction numbers for the first 100 dimensions.

3.2. Reducing the search space. In dimension j, we need to choose the first s_j values of $m_{k,j}$, with each $m_{k,j}$ odd and less than 2^k. This leads to a total of $2^{s_j(s_j-1)/2}$ different sets of direction numbers for dimension j. The number of possibilities grows extremely fast: with the degree-6 primitive polynomials we have $2^{15} = 32768$ choices; with the degree-7 polynomials we have $2^{21} = 2097152$ choices. Clearly an exhaustive search based on any kind of criterion is practically impossible.

To reduce the search space, we take a "component-by-component" approach (borrowing the idea from lattice rules; see [20]); that is, once the direction numbers up to

dimension $d-1$ are chosen, we keep those fixed while choosing the direction numbers for dimension d. In a nutshell, our algorithm goes like this:

> *in dimension d, we sieve through various choices of direction numbers and eliminate those which do not satisfy Property* A *if $d \le 1111$; for the remaining choices we check the quality of all (j,d)-projections for $j = 1, 2, \ldots, d-1$, and find the set of direction numbers which gives the best* $2D$ *projections overall.*

The precise criterion for deciding the overall quality of the 2D projections is discussed in the next subsection.

For each of the first 19 dimensions, a full search through all sets of direction numbers is feasible, since there are at most 32768 choices in each dimension. Our results suggest that, most of the time, exactly half of the choices satisfy Property A, although occasionally we do get none or all of the choices satisfying Property A. We have no explanation for this phenomenon, which appears to be a side effect of having a component-by-component algorithm. If we indeed end up with all choices failing Property A, then we abandon the search, adjust our search criterion, and start again. Fortunately this does not happen very often.

From dimension 20 onward, we only search through a number of randomly generated sets of direction numbers. Since the cost of the algorithm increases linearly with dimension (due to the number of 2D projections we have to check in each step, together with other computational aspects regarding Property A to be discussed later), we restrict ourselves to

$$w_d := \frac{2000000}{d}$$

random choices satisfying Property A in dimension d if $20 \le d \le 1111$, or simply w_d random choices when $d > 1111$. Thus we have 100000 choices in dimension 20 and down to 100 choices in dimension 20000. Our results suggest that, most of the time, we need to generate roughly $2w_d$ choices to get w_d choices satisfying Property A. We abandon the search if after $2w_d$ choices we still have none satisfying Property A when $d \le 1111$.

We remark that Property A is important in low dimensions, say, for $d \le 19$. As the dimension d increases, it becomes less and less meaningful. Nevertheless, we try to retain Property A up to dimension 1111 for consistency with the Sobol′ points from [9].

3.3. Defining the search criterion. Assume for the moment that we already have the direction numbers up to dimension $d-1$, and suppose for now that m is given and fixed. Our aim is to choose direction numbers for dimension d so that we have small values of $t(j,d;m)$ for all $j = 1, 2, \ldots, d-1$. For fixed d, the upper bound (2.7) suggests that the values of $t(j,d;m)$ can potentially be higher for larger values of j.

To get an idea of how these t-values are distributed empirically, we take the primitive polynomials and direction numbers from [9] up to dimension 17, and we generate all 32768 sets of direction numbers for dimension 18. It turns out that exactly half of these direction numbers satisfy Property A. With different values of m, we compute $t(j,18;m)$ for all $j = 1, 2, \ldots, 17$ for each set of direction numbers satisfying Property A. A frequency table for the case $m = 12$ is presented in Table 3.1.

The jth column in the table contains the frequency of the t-values for the $(j,18)$-projection for those direction numbers satisfying Property A. In this case the sum of each column is 16384. The entries with no number indicate that the corresponding t-values are not possible due to the upper bound (2.7). For example, the t-value of

TABLE 3.1
Frequency of $t(j, 18; 12)$ for Sobol′ points from [9].

$t\backslash j$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
0	32	22	57	0	34	0	0	26	36	0	0	45	0	0	24	29	67
1	2252	3212	2095	1980	2290	752	1869	2882	1916	978	1956	2749	1580	1712	2028	1783	2033
2	5932	6014	6696	7532	5844	5262	4971	5804	6048	5924	6190	5718	6676	5760	7468	7596	8116
3	5672	5152	5168	3928	5784	5154	4456	5240	4992	4778	5294	3008	7104	4624	4304	3904	4184
4	1472	1984	2368	2944	2432	3232	3104	2432	2368	3168	1920	2816	1024	4288	1536	2048	1984
5	1024	0	0	0	0	1984	1984	0	1024	512	1024	2048	0	0	1024	1024	0
6			0	0	0	0	0	0	0	1024	0	0	0	0	0	0	0
7				0	0	0	0	0	0	0	0	0	0	0	0	0	0
8						0	0	0	0	0	0	0	0	0	0	0	0
9								0	0	0	0	0	0	0	0	0	0
10														0	0	0	0
11																	

TABLE 3.2
Frequency of $t(j, 200; 18)$ for Sobol′ points from [9].

$t\backslash j$	11	22	33	44	55	66	77	88	99	110	121	132	143	154	165	176	187	198
0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	289	344	264	0	0	0	218	207	0	269	0	0	0	0	72	305	0	0
2	2304	2411	2177	1416	0	0	1969	1986	1423	2369	1015	0	0	1594	1445	2166	1526	1339
3	3239	3174	3425	3087	0	3047	3198	3212	3210	3163	3155	0	0	3046	3080	3654	3271	2811
4	2149	2616	2248	2638	0	3293	2511	2446	2854	2411	2660	3836	4346	3022	2857	2062	2742	3016
5	1361	844	1142	1773	4432	1954	1221	1249	1455	1150	1679	2847	2696	1309	1415	1102	1470	1466
6	405	267	417	593	2789	882	581	597	614	392	851	1734	1535	557	597	487	581	745
7	154	183	186	258	1399	465	176	180	272	147	354	829	762	271	321	137	232	318
8	67	86	91	136	680	165	72	67	101	88	157	409	337	89	102	75	105	223
9	10	74	27	64	340	99	36	37	37	11	75	192	164	72	60	12	45	47
10	21	0	23	22	180	45	18	19	21	0	46	76	63	22	51	0	19	20
11	0	0	0	13	102	18	0	0	13	0	8	36	97	18	0	0	9	15
12	0	0	0	0	37	8	0	0	0	0	0	27	0	0	0	0	0	0
13	0	0	0	0	14	24	0	0	0	0	0	14	0	0	0	0	0	0
14	0	0	0	0	7	0	0	0	0	0	0	0	0	0	0	0	0	0
15		0	0	0	20	0	0	0	0	0	0	0	0	0	0	0	0	0
16		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18					0	0	0	0	0	0	0	0	0	0	0	0	0	0
19																		

the $(6, 18)$-projection is at most $\min(12, 4 + 6 - 2) = 8$. Note there is no guarantee that a particular set of direction numbers which yields a small value of $t(j, d; m)$ for some j will also yield a small value $t(j', d; m)$ for $j' \neq j$.

We produce a similar table for dimension 200 (see Table 3.2), where we take the primitive polynomials and direction numbers from [9] up to dimension 199 and tabulate the frequency of $t(j, 200; m)$ from 10000 randomly generated sets of direction numbers satisfying Property A, with $m = 18$ and with j increasing in steps of 11.

It appears from these frequency tables (and many others that we constructed) that the upper bound (2.7) does not play a significant role in the empirical distribution of $t(j, d; m)$ as j varies. Rather, the spread of $t(j, d; m)$ remains much the same for increasing values of j. This observation leads us to define the quantity

$$T(d; m) := \max_{1 \leq j \leq d-1} t(j, d; m),$$

which, for the digital net of 2^m Sobol′ points in dimension d, corresponds to the highest t-value of the 2D projections formed by dimension d and the earlier dimensions. In other words, it is essentially the largest entry in the dth row of a table such as Table 2.1. For example, we see that $T(28; 12) = 6$ and $T(22; 12) = 7$ for the Sobol′ points from [9].

TABLE 3.3
Frequency of $T(18;m)$ for Sobol′ points from [9].

$T\backslash m$	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	448	1216	590	315	1230	1148	256	70	31	47	0	100	0	18	0	0	0	18
4	5696	6272	5282	3925	5438	10676	6976	3794	2207	2033	4512	4540	2736	1516	976	396	1376	2982
5	10240	4800	6992	5120	5364	3024	8128	8680	7234	7768	6016	8800	11152	6482	5472	7604	5720	9096
6	0	4096	3520	4528	1856	1536	1024	3840	5888	4168	4320	1920	1472	5872	5712	4960	6920	3264
7		0	0	2496	1472	0	0	0	1024	2368	1024	1024	1024	1472	3200	2400	2368	1024
8			0	0	1024	0	0	0	0	0	512	0	0	1024	1024	1024	0	0
9				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10					0	0	0	0	0	0	0	0	0	0	0	0	0	0
11																		

TABLE 3.4
Frequency of $T(200;m)$ for Sobol′ points from [9].

$T\backslash m$	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	83	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	2212	1017	207	63	5	2	0	0	0	0	0	0	0	0	0	0	0	0
7	7705	3819	3015	1451	673	262	118	27	25	18	16	0	61	0	98	20	6	0
8	0	5164	3719	4144	3021	2250	1503	921	766	664	655	733	1374	1597	1803	925	515	297
9		0	3059	2877	3951	3246	3212	2774	2610	2415	2490	2693	3155	3832	4220	3459	2717	2285
10			0	1465	1589	2856	2556	3076	3038	3015	2901	2923	2684	2434	2766	3409	3435	3365
11				0	761	860	1838	1674	1964	1958	2043	1837	1507	1185	730	1620	2061	2352
12					0	524	518	1043	828	1082	1023	1037	667	547	260	397	925	1052
13						0	255	313	521	432	487	415	301	233	105	126	252	517
14							0	172	161	282	222	188	120	78	18	44	73	115
15								0	87	66	123	107	62	42	0	0	16	17
16									0	68	13	60	46	21	0	0	0	0
17										0	27	7	23	8	0	0	0	0
18											0	0	0	23	0	0	0	0
19												0	0	0	0	0	0	0
20													0	0	0	0	0	0
21																		

Of course, we do not want a set of direction numbers to be restricted to just one value of m (recall that we are considering the first 2^m points of the Sobol′ sequence). Ideally, we want our direction numbers to be good for a large range of m. Again we see from the bound (2.7) that a larger value of m could potentially mean higher values of $T(d;m)$. To see how these numbers are distributed empirically, in Table 3.3 we use the primitive polynomials and direction numbers from [9] up to dimension 17 and tabulate the frequency of $T(18;m)$ for $m = 6, 7, \ldots, 23$ from all 16384 sets of direction numbers satisfying Property A in dimension 18. Similar data is presented in Table 3.4 for $m = 8, 9, \ldots, 25$ for dimension 200 from 10000 randomly generated sets of direction numbers satisfying Property A.

We see that, once again, the bound (2.7) does not play a significant role in the empirical distribution of the numbers $T(d;m)$. Larger values of m do not necessarily mean higher values of $T(d;m)$. This leads us to define our "vanilla" search criterion

$$\mathcal{D}^{(0)}(d;m_{\min},m_{\max}) := \max_{m_{\min}\le m\le m_{\max}} T(d;m) = \max_{\substack{m_{\min}\le m\le m_{\max}\\ 1\le j\le d-1}} t(j,d;m),$$

which is precisely the largest t-value among all 2D projections of 2^m points up to dimension d with m between $m_{\min}$ and $m_{\max}$. (The idea to restrict m to a finite

range in the search criterion was recently used in [5] in the construction of embedded or extensible lattice rules.)

The vanilla search criterion has two drawbacks. First, it implicitly assumes that all 2D projections are equally important. However, in many practical applications the earlier dimensions are more important than the later ones. In other words, among all (j,d)-projections for $j = 1, 2, \ldots, d-1$, we are willing to allow the t-values to be higher for larger values of j in the hope of reducing the t-values for smaller values of j. Our attempt to achieve this is by introducing weights as follows (borrowing the concept of weights from [22]):

$$\hat{T}(d;m) := \max_{1\le j\le d-1} \left[t(j,d;m) \times 0.9999^{j-1}\right].$$

The choice of weights 0.9999^{j-1} is clearly arbitrary. We have $0.9999^{20} \approx 0.998$, $0.9999^{200} \approx 0.980$, $0.9999^{2000} \approx 0.819$, and $0.9999^{20000} \approx 0.135$. In other words, the effect of the weights is very minor for small values of j, but it does make a difference when the dimension is really high.[1]

The second drawback of the vanilla search criterion is that it does not take into account the difference between m and t. For fixed m, we should minimize the t-value, but with m now varying, one could argue that it is more important to have $m - t$ as large as possible, since it corresponds to how fine the subdivisions are in the definition of nets. This leads us to consider a compromise and define the modified search criterion

$$\mathcal{D}^{(q)}(d; m_{\min}, m_{\max}) := \max_{m_{\min}\le m\le m_{\max}} \frac{[\hat{T}(d;m)]^q}{m - \hat{T}(d;m) + 1}, \qquad q > 0.$$

The parameter q acts as a balance between the importance of t being small and $m-t$ being large.

We focus on a range of m, say $m_{\min} = 1$ and $m_{\max} = 31$ (which gives 2^{31} points). With the direction numbers for all dimensions up to $d-1$ already chosen and fixed, our algorithm then searches through different sets of direction numbers in dimension d, eliminating those which do not satisfy Property A (for d up to 1111), and finally choosing the set which gives the smallest value of $\mathcal{D}^{(q)}(d; m_{\min}, m_{\max})$. Recall that we search through all possible sets of direction numbers when $d \le 19$, but we consider only a randomly generated selection of w_d sets when $d \ge 20$.

3.4. Choosing new direction numbers. Since we are unsure about which search criterion $\mathcal{D}^{(q)}(d; 1, 31)$ is the best, we try $q = 1, 2, \ldots, 9$, among which $q = 2$, 4, 8, 9 fails Property A in dimensions 56, 63, 16, 16, respectively. The best of the remaining choices appears to be $q = 6$, although $q = 5, 7$ do exhibit similar quality. We also try out the vanilla search criterion $\mathcal{D}^{(0)}(d; 1, 31)$. Although the vanilla criterion appears to be better than $\mathcal{D}^{(6)}(d; 1, 31)$ for larger values of m, it is worse for smaller values of m, which is undesirable. This is completely within our expectation since the vanilla criterion does not take into account the difference between m and t.

In Table 3.5 we present the values of $t(j, 28; 12)$ for $j = 1, 2, \ldots, 27$ with direction numbers obtained using the search criterion $\mathcal{D}^{(6)}(d; 1, 31)$. This table should be

[1] This choice of weights may not be suitable for the type of problems where the interactions between successive variables are the most important ones (e.g., in simulation problems). In that case, a different choice of weights should be used in $\hat{T}$ for the search criterion, thus leading to a different set of direction numbers.

TABLE 3.5
Values of $t(j,d;12)$ for Sobol′ points obtained based on $\mathcal{D}^{(6)}(d;1,31)$.

$d\backslash j$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
2	0																										
3	1	1																t-value 0 occurred 1 time								
4	1	2	2															t-value 1 occurred 69 times								
5	2	2	2	2														t-value 2 occurred 139 times								
6	3	1	2	3	2													t-value 3 occurred 109 times								
7	2	3	2	2	1	1												t-value 4 occurred 46 times								
8	2	1	2	2	3	2	3											t-value 5 occurred 14 times								
9	1	2	2	2	2	1	1	2																			
10	3	3	3	2	1	4	2	2	2																		
11	2	3	3	2	1	3	3	1	3	3																	
12	1	2	3	2	2	2	2	2	3	2	2																
13	1	3	2	2	2	2	2	1	3	2	2	2															
14	2	3	2	2	3	3	4	4	2	3	1	4	1														
15	1	3	2	2	2	3	2	3	3	2	3	2	2	1													
16	4	3	1	4	2	**5**	3	3	2	2	3	3	3	1	3												
17	3	3	2	3	3	1	4	4	4	3	1	3	3	1	3	2											
18	2	2	1	3	3	2	1	3	2	3	2	4	2	3	1	2	3										
19	1	3	3	3	4	2	2	3	4	1	2	1	3	1	2	3	1	3									
20	1	4	1	4	3	4	3	**5**	4	2	2	1	2	4	**5**	2	4	3	4								
21	2	1	1	2	1	3	2	2	2	2	1	3	3	**5**	1	2	2	2	1	3							
22	4	4	2	3	2	1	1	1	3	2	3	2	3	3	4	3	4	4	3	4	3						
23	2	3	2	3	2	3	2	3	4	**5**	3	2	4	4	3	4	1	2	2	4	2	2					
24	1	3	2	3	2	2	2	3	2	1	2	2	4	3	3	2	1	1	2	2	3	2	1				
25	2	2	3	4	2	3	**5**	**5**	3	1	2	2	3	2	3	1	2	**5**	3	2	2	1	**5**	2			
26	2	3	3	4	2	4	2	3	2	**5**	3	1	2	3	4	3	2	4	1	3	4	3	1	3	1		
27	2	4	3	2	1	1	1	3	2	3	3	2	4	1	2	2	2	2	2	3	2	3	2	1	1	1	
28	3	3	4	4	3	1	2	4	1	4	3	2	2	**5**	3	1	2	**5**	1	2	3	**5**	**5**	4	4	2	4

compared with Table 2.1, keeping in mind that our primitive polynomials are in a different order. (If we reorder the primitive polynomials and direction numbers from [9] to our new ordering in which the a_j are in increasing order, then the updated Table 2.1 has counts 22, 9, 1 for t-values 5, 6, 7, respectively, which is even worse than before. Furthermore, Property A fails in dimension 17 with the new ordering.) Notice that we have successfully eliminated all occurrences of t-values 6 and 7. Although we have not eliminated the t-value 5, we have reduced the total number from 20 (or 22 for the new ordering) to 14. Similar improvements are observed in many other tables that we produced.

In Tables 3.6 and 3.7 we compare the overall quality up to dimension 28 between the Sobol′ points obtained using the search criterion $\mathcal{D}^{(6)}(d;1,31)$ and those obtained in [9]. The (d,m)th entry in the table corresponds to the largest t-value among all 2D projections between dimension d and the previous dimensions for 2^m Sobol′ points. At the end of each row/column we show the maximum entry in each row/column as well as the average value (to one decimal place) of each row/column. It can be seen that the criterion $\mathcal{D}^{(6)}(d;1,31)$ nicely pushes bigger t-values toward higher dimensions and larger values of m. The improvement over the original direction numbers from [9] is clearly noticeable.

In Table 3.8 we extend our comparison in the previous paragraph to higher dimensions. For $m = 10$, 12, 14, 16, 18 and for t-values from 0 up to m, we list the dimension at which each t-value first occurs. (At press, the search reached dimension 8300.) Obviously we want bigger t-values to occur as late in the dimension as possible, and thus the larger the entries in the table, the better. The results clearly indicate that our search criterion $\mathcal{D}^{(6)}(d;1,31)$ is the winner.

TABLE 3.6
Values of $T(d;m)$ for Sobol′ points obtained based on $\mathcal{D}^{(6)}(d;1,31)$.

$d\backslash m$	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	Max	Avg
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.0
3	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1.0
4	2	1	2	2	1	2	2	2	2	2	2	2	3	2	2	2	2	2	2	2	2	3	2.0
5	2	3	1	2	2	3	2	2	2	2	3	2	2	2	2	3	2	2	3	3	2	3	2.2
6	2	2	3	3	3	3	3	3	2	3	3	3	3	2	3	3	4	3	3	4	4	4	3.0
7	2	2	2	3	3	2	3	3	3	3	3	3	4	3	4	3	3	3	3	3	3	4	2.9
8	3	4	2	3	4	3	4	3	4	4	4	4	3	4	5	3	4	4	3	4	3	5	3.6
9	3	3	3	4	3	4	4	2	3	3	4	4	3	4	5	4	5	4	4	4	4	5	3.7
10	3	3	3	3	3	3	3	4	3	3	4	3	4	4	3	4	4	4	4	4	3	4	3.4
11	4	3	3	4	4	4	3	3	4	3	3	4	4	5	4	4	3	4	4	5	3	5	3.7
12	4	3	4	3	4	4	3	3	4	5	4	4	4	5	4	4	4	5	4	5	4	5	4.0
13	2	3	4	4	3	4	4	3	4	4	5	4	5	5	4	4	3	4	4	4	4	5	3.9
14	3	3	3	4	4	3	3	4	5	4	4	5	4	5	4	5	4	4	4	5	4	5	4.0
15	4	3	4	3	4	3	3	3	4	4	5	6	5	6	6	6	4	5	4	5	6	6	4.4
16	4	4	4	5	5	5	4	5	4	5	5	4	5	4	4	4	5	6	4	5	5	6	4.6
17	3	3	4	3	4	4	4	4	5	4	4	5	4	5	4	5	4	5	6	5	4	6	4.2
18	4	4	4	4	5	4	5	4	4	5	5	4	4	4	5	6	5	4	4	5	6	6	4.5
19	4	4	4	5	5	4	5	4	4	4	4	5	5	4	5	5	4	5	5	4	4	5	4.4
20	4	4	5	5	4	5	5	5	5	5	4	5	5	5	5	5	5	5	6	6	5	6	4.9
21	4	4	5	4	4	3	4	5	4	5	4	4	4	4	5	4	5	6	6	4	5	6	4.4
22	4	4	4	4	4	4	4	4	5	6	5	5	5	5	6	5	6	6	6	4	5	6	4.8
23	4	5	4	4	4	4	5	5	5	5	6	6	6	5	4	5	5	6	5	5	4	6	4.9
24	4	5	4	5	4	4	5	4	4	5	4	4	5	6	5	5	5	5	6	5	6	6	4.8
25	4	4	4	5	5	5	6	5	6	5	6	6	5	5	6	6	7	5	5	5	6	7	5.3
26	4	5	5	4	4	5	5	5	5	6	5	5	6	6	5	5	6	7	7	6	7	7	5.4
27	3	4	5	5	5	5	5	4	5	4	5	6	6	6	6	6	5	6	7	6	6	7	5.2
28	3	4	3	4	5	5	4	5	5	4	5	5	6	5	5	6	5	6	5	6	6	6	4.9
Max	4	5	5	5	5	5	6	5	6	6	6	6	6	6	6	6	7	7	7	6	7		
Avg	3.1	3.3	3.3	3.6	3.6	3.6	3.7	3.5	3.8	3.9	4.0	4.0	4.1	4.1	4.1	4.2	4.1	4.3	4.3	4.3	4.1		

TABLE 3.7
Values of $T(d;m)$ for Sobol′ points from [9].

$d\backslash m$	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	Max	Avg
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.0
3	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1.0
4	2	3	1	2	2	2	2	2	2	3	2	2	2	2	3	2	2	2	3	2	2	3	2.1
5	2	1	2	2	2	3	2	2	2	3	2	2	3	2	2	2	2	2	3	2	2	3	2.1
6	3	3	2	2	3	3	4	2	3	2	3	3	3	2	3	3	3	4	3	4	3	4	2.9
7	3	2	3	3	2	3	3	3	3	2	3	3	3	4	3	3	2	3	3	3	3	4	2.9
8	2	3	2	3	3	4	4	3	4	3	4	5	4	4	4	4	3	4	4	4	5	5	3.6
9	3	4	4	5	3	4	5	3	4	4	5	4	3	4	5	6	4	3	4	4	4	6	4.0
10	3	3	4	4	4	4	5	6	7	5	4	5	6	5	6	3	3	4	5	4	4	7	4.5
11	4	3	4	5	3	4	5	6	4	4	4	5	3	4	5	3	4	4	5	4	3	6	4.1
12	3	4	5	4	4	4	3	3	4	3	4	5	5	6	4	5	6	4	5	5	4	6	4.3
13	3	3	3	4	4	4	4	5	6	5	6	7	4	5	6	6	5	4	5	5	4	7	4.7
14	3	3	4	4	4	4	3	4	4	4	5	5	4	5	6	6	6	6	5	6	6	6	4.6
15	3	4	4	3	4	5	3	4	5	5	5	6	4	5	6	6	7	8	5	5	6	8	4.9
16	3	3	4	4	4	5	4	5	6	4	5	6	7	5	5	6	7	7	8	5	6	8	5.2
17	4	4	5	4	5	3	4	5	6	7	5	6	5	4	5	5	5	5	4	5	5	7	4.8
18	4	4	5	6	7	8	5	5	4	5	6	6	4	4	5	6	7	7	5	5	6	8	5.4
19	4	4	5	5	4	5	4	4	5	5	6	7	4	5	6	5	6	6	7	4	5	7	5.0
20	3	4	5	5	4	4	5	6	4	5	5	6	6	6	7	5	5	6	7	6	6	7	5.2
21	4	5	5	6	5	6	4	4	4	5	4	5	6	7	5	6	7	5	5	6	7	7	5.3
22	4	4	5	6	4	5	6	7	6	7	5	5	5	5	6	6	7	6	7	5	6	7	5.6
23	4	5	4	5	6	7	5	5	5	4	4	5	6	7	5	6	6	5	6	7	8	8	5.5
24	4	4	5	6	6	6	5	6	4	4	5	5	6	5	6	5	6	7	6	5	5	7	5.3
25	4	3	4	5	4	4	5	4	5	6	6	7	8	9	6	5	5	6	6	7	8	9	5.6
26	4	4	5	5	5	6	5	5	6	7	8	6	6	6	6	7	8	7	8	7	6	8	6.0
27	4	5	5	5	6	7	6	4	4	5	6	7	7	6	7	5	5	6	7	6	7	7	5.7
28	4	4	4	5	6	6	6	6	6	5	6	6	7	8	7	6	5	6	6	7	8	8	5.9
Max	4	5	5	6	7	8	6	7	7	7	8	7	8	9	7	7	8	8	8	7	8		
Avg	3.1	3.3	3.7	4.0	3.9	4.3	4.0	4.1	4.2	4.1	4.4	4.8	4.5	4.7	4.8	4.6	4.7	4.7	4.9	4.6	4.8		

TABLE 3.8
Dimension at which each t-value first occurs.

	t-value																		
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
$m = 10$ $D^{(6)}$	2	3	4	5	9	16	32	76	167	431	>8300								
[9]	2	3	4	5	8	15	21	23	18	36	>1111								
$m = 12$ $D^{(6)}$	2	3	4	6	10	16	34	40	109	242	506	1049	>8300						
[9]	2	3	4	7	10	10	10	22	35	51	96	61	>1111						
$m = 14$ $D^{(6)}$	2	3	4	6	8	12	22	48	85	164	383	761	1298	1837	>8300				
[9]	2	3	4	5	9	10	25	17	40	55	67	67	131	61	>1111				
$m = 16$ $D^{(6)}$	2	3	4	6	8	14	15	35	80	159	280	525	926	1534	2116	2662	>8300		
[9]	2	3	4	6	9	8	15	13	32	58	69	74	102	95	447	167	>1111		
$m = 18$ $D^{(6)}$	2	3	4	7	8	11	15	35	70	108	220	393	701	1172	1669	2649	3282	3619	>8300
[9]	2	3	4	7	7	10	12	21	28	25	103	126	115	114	196	232	665	380	>1111

In the appendix we list the primitive polynomials and the direction numbers obtained using the search criterion $\mathcal{D}^{(6)}(d; 1, 31)$ for d up to 100.

3.5. Speeding up the computation. We implement our search algorithm in C++, where bit-by-bit operations such as exclusive-or are easy and quick to perform. Since an (unsigned) integer in C++ has 32 bits (i.e., 4 bytes), we assume that the total number of Sobol′ points needed is no greater than 2^{32}, that is, $m_{\max} \leq 32$. This is not an unreasonable assumption in practice, since $2^{32} \approx 4.3 \times 10^9$ function evaluations would take an enormous amount of time when d is large.

Recall that the matrices $C_{m',j}$ (see (2.6)) for $m' < m$ are embedded in the upper-left corners of the matrices $C_{m,j}$. Thus for each j, we compute the 32×32 matrix $C_{32,j}$ and store each row of the matrix as one 32-bit integer. We can then compute the t-values of the 2D projections (see the pseudocode in Figure 2.1) for any $m \leq 32$ by applying exclusive-or operations to these integers, operating on all 32 bits in one go.

We also need to form the matrix V_d, which grows as the dimension increases. Knowing in advance that we check Property A only up to dimension 1111, it is advantageous to consider V_d to be a matrix of size $d \times 1111$; that is, there are always 1111 entries in each row. Note that the first row of the matrix $C_{32,j}$ gives the first 32 entries in the jth row of V_d; the remaining entries in the jth row can be obtained using the recurrence

$$v_{k,j,1} := a_{1,j}\, v_{k-1,j,1} \oplus a_{2,j}\, v_{k-2,j,1} \oplus \cdots \oplus a_{s_j-1,j}\, v_{k-s_j+1,j,1} \oplus v_{k-s_j,j,1},$$

which can be derived from (2.2) and (2.3). Since we take a component-by-component approach, we build up the matrix V_d by adding one new row at a time and then performing row reduction on the new row to maintain an upper-triangular form for the $d \times d$ submatrix with 1's down the main diagonal. We store the reduced form of the rows of V_d as integers, which requires $\lceil 1111/32 \rceil = 35$ integers per row. Assuming that the first $d-1$ rows of the matrix are stored in their row-reduced form, the cost for evaluating the determinant of V_d grows only linearly with d.

4. Summary. In our previous paper [9] we gave the primitive polynomials and direction numbers for generating Sobol′ sequences up to 1111 dimensions. Those parameters have been used by practitioners in areas such as mathematical finance, statistics, and even theoretical biology; see, for example, [7, 12, 19]. Recently there is news that the commercial Numerical Algorithms Group is planning to incorporate our parameters into its software package, and the open source QuantLib project (see

http://www.quantlib.org) is also interested in adding the parameters to its library. There is a huge amount of interest in having good Sobol′ sequences for still higher dimensions.

The Sobol′ sequence from [9] satisfies the so-called Property A, but it does not guarantee that there are no bad correlations between pairs of dimensions. We have shown through analyzing the t-values of the 2D projections of 2^m Sobol′ points that bad correlations do exist. This led us to find new direction numbers using a search criterion based on optimizing the t-values across a range of values for m. The new Sobol′ sequence obtained still satisfies Property A up to dimension 1111, and the problem of bad 2D projections is alleviated in the sense that we systematically pushed bigger t-values toward higher dimensions and larger values of m.

From a theoretical point of view in terms of t-values for the 2D projections, our new Sobol′ sequence beats our old one in [9] (and a number of other implementations). How it performs in practice remains to be seen. We carried out some preliminary calculations for a number of finance models and found that the new Sobol′ sequence gives better results in some cases and is at worst comparable to the old one. More comprehensive investigation is required and is left for future work.

The primitive polynomials and direction numbers obtained based on various search criteria can be downloaded as text files from our web page

http://www.maths.unsw.edu.au/~fkuo/sobol/.

The files will be updated frequently as the parameters for higher dimensions become available.

Appendix.

j	s_j	a_j	$m_{k,j}$
2	1	0	1
3	2	1	1 3
4	3	1	1 3 1
5	3	2	1 1 1
6	4	1	1 1 3 3
7	4	4	1 3 5 13
8	5	2	1 1 5 5 17
9	5	4	1 1 5 5 5
10	5	7	1 1 7 11 19
11	5	11	1 1 5 1 1
12	5	13	1 1 1 3 11
13	5	14	1 3 5 5 31
14	6	1	1 3 3 9 7 49
15	6	13	1 1 1 15 21 21
16	6	16	1 3 1 13 27 49
17	6	19	1 1 1 15 7 5
18	6	22	1 3 1 15 13 25
19	6	25	1 1 5 5 19 61
20	7	1	1 3 7 11 23 15 103
21	7	4	1 3 7 13 13 15 69
22	7	7	1 1 3 13 7 35 63
23	7	8	1 3 5 9 1 25 53
24	7	14	1 3 1 13 9 35 107
25	7	19	1 3 1 5 27 61 31
26	7	21	1 1 5 11 19 41 61
27	7	28	1 3 5 3 3 13 69
28	7	31	1 1 7 13 1 19 1
29	7	32	1 3 7 5 13 19 59
30	7	37	1 1 3 9 25 29 41
31	7	41	1 3 5 13 23 1 55
32	7	42	1 3 7 3 13 59 17
33	7	50	1 3 1 3 5 53 69
34	7	55	1 1 5 5 23 33 13
35	7	56	1 1 7 7 1 61 123
36	7	59	1 1 7 9 13 61 49
37	7	62	1 3 3 5 3 55 33
38	8	14	1 3 1 15 31 13 49 245
39	8	21	1 3 5 15 31 59 63 97
40	8	22	1 3 1 11 11 11 77 249
41	8	38	1 3 1 11 27 43 71 9
42	8	47	1 1 7 15 21 11 81 45
43	8	49	1 3 7 3 25 31 65 79
44	8	50	1 3 1 1 19 11 3 205
45	8	52	1 1 5 9 19 21 29 157
46	8	56	1 3 7 11 1 33 89 185
47	8	67	1 3 3 3 15 9 79 71
48	8	70	1 3 7 11 15 39 119 27
49	8	84	1 1 3 1 11 31 97 225
50	8	97	1 1 1 3 23 43 57 177
51	8	103	1 3 7 7 17 17 37 71
52	8	115	1 3 1 5 27 63 123 213
53	8	122	1 1 3 5 11 43 53 133
54	9	8	1 3 5 5 29 17 47 173 479
55	9	13	1 3 3 11 3 1 109 9 69
56	9	16	1 1 1 5 17 39 23 5 343
57	9	22	1 3 1 5 25 15 31 103 499
58	9	25	1 1 1 11 11 17 63 105 183
59	9	44	1 1 5 11 9 29 97 231 363
60	9	47	1 1 5 15 19 45 41 7 383
61	9	52	1 3 7 7 31 19 83 137 221
62	9	55	1 1 1 3 23 15 111 223 83
63	9	59	1 1 5 13 31 15 55 25 161
64	9	62	1 1 3 13 25 47 39 87 257
65	9	67	1 1 1 11 21 53 125 249 293
66	9	74	1 1 7 11 11 7 57 79 323
67	9	81	1 1 5 5 17 13 81 3 131
68	9	82	1 1 7 13 23 7 65 251 475
69	9	87	1 3 5 1 9 43 3 149 11
70	9	91	1 1 3 13 31 13 13 255 487
71	9	94	1 3 3 1 5 63 89 91 127
72	9	103	1 1 3 3 1 19 123 127 237
73	9	104	1 1 5 7 23 31 37 243 289
74	9	109	1 1 5 11 17 53 117 183 491
75	9	122	1 1 1 5 1 13 13 209 345
76	9	124	1 1 3 15 1 57 115 7 33
77	9	137	1 3 1 11 7 43 81 207 175
78	9	138	1 3 1 1 15 27 63 255 49
79	9	143	1 3 5 3 27 61 105 171 305
80	9	145	1 1 5 3 1 3 57 249 149
81	9	152	1 1 3 5 5 57 15 13 159
82	9	157	1 1 1 11 7 11 105 141 225
83	9	167	1 3 3 5 27 59 121 101 271
84	9	173	1 3 5 9 11 49 51 59 115
85	9	176	1 1 7 1 23 45 125 71 419
86	9	181	1 1 3 5 23 5 105 109 75
87	9	182	1 1 7 15 7 11 67 121 453
88	9	185	1 3 7 3 9 13 31 27 449
89	9	191	1 3 1 15 19 39 39 89 15
90	9	194	1 1 1 1 1 33 73 145 379
91	9	199	1 3 1 15 15 43 29 13 483
92	9	218	1 1 7 3 19 27 85 131 431
93	9	220	1 3 3 3 5 35 23 195 349
94	9	227	1 3 3 7 9 27 39 59 297
95	9	229	1 1 3 9 11 17 13 241 157
96	9	230	1 3 7 15 25 57 33 189 213
97	9	234	1 1 7 1 9 55 73 83 217
98	9	236	1 3 3 13 19 27 23 113 249
99	9	241	1 3 5 3 23 43 3 253 479
100	9	244	1 1 5 5 11 5 45 117 217

Acknowledgments. The authors thank Ian Sloan, Dirk Nuyens, and Rudolf Schürer for valuable comments and suggestions. Part of this work was done when the first author was a Visiting Research Fellow at the University of New South Wales.

REFERENCES

[1] I. A. Antonov and V. M. Saleev, *An economic method of computing LP_τ-sequences*, Zh. Vȳchisl. Mat. Mat. Fiz., 19 (1979), pp. 243–245 (in Russian); Comput. Maths. Math. Phys., 19 (1980), pp. 252–256 (in English).

[2] P. Bratley and B. L. Fox, *Algorithm* 659*: Implementing Sobol's quasirandom sequence generator*, ACM Trans. Math. Software, 14 (1988), pp. 88–100.

[3] R. E. Caflisch, W. Morokoff, and A. Owen, *Valuation of mortgage-backed securities using Brownian bridges to reduce effective dimension*, J. Comput. Finance, 1 (1997), pp. 27–46.

[4] J. Cheng and M. J. Druzdzel, *Computational investigation of low-discrepancy sequences in simulation algorithms for Bayesian networks*, in Proceedings of the 16th Annual Conference on Uncertainty in Artificial Intelligence, C. Boutilier and M. Goldszmidt eds., Morgan Kaufmann, San Francisco, 2000, pp. 72–81.

[5] R. Cools, F. Y. Kuo, and D. Nuyens, *Constructing embedded lattice rules for multivariate integration*, SIAM J. Sci. Comput., 28 (2006), pp. 2162–2188.

[6] J. Dick, F. Pillichshammer, and B. J. Waterhouse, *The construction of good extensible rank-*1 *lattices*, Math. Comp., 77 (2008), pp. 2345–2373.

[7] H.-K. Fung and L. K. Li, *Pricing discrete dynamic fund protections*, N. Am. Actuar. J., 7 (2003), pp. 23–31.

[8] P. Jäckel, *Monte Carlo Methods in Finance*, John Wiley and Sons, New York, 2002.

[9] S. Joe and F. Y. Kuo, *Remark on Algorithm* 659*: Implementing Sobol's quasirandom sequence generator*, ACM Trans. Math. Software, 29 (2003), pp. 49–57.

[10] F. Y. Kuo, *Component-by-component constructions achieve the optimal rate of convergence for multivariate integration in weighted Korobov and Sobolev spaces*, J. Complexity, 19 (2003), pp. 301–320.

[11] C. Lemieux, M. Cieslak, and K. Luttmer, *RandQMC User's Guide: A Package for Randomized Quasi-Monte Carlo Methods in C*, Technical report 2002-712-15, Department of Computer Science, University of Calgary, Calgary, AB, Canada, 2002.

[12] J. C. W. Locke, A. J. Millar, and M. S. Turner, *Modelling genetic networks with noisy and varied experimental data: The circadian clock in Arabidopsis thaliana*, J. Theor. Biol., 234 (2005), pp. 383–393.

[13] W. J. Morokoff and R. E. Caflisch, *Quasi-random sequences and their discrepancies*, SIAM J. Sci. Comput., 15 (1994), pp. 1251–1279.

[14] H. Niederreiter, *Random Number Generation and Quasi-Monte Carlo Methods*, SIAM, Philadelphia, 1992.

[15] D. Nuyens and R. Cools, *Fast algorithms for component-by-component construction of rank-*1 *lattice rules in shift-invariant reproducing kernel Hilbert spaces*, Math. Comp., 75 (2006), pp. 903–920.

[16] S. H. Paskov and J. F. Traub, *Faster valuation of financial derivatives*, J. Portfolio Manage., 22 (1995), pp. 113–120.

[17] W. Ch. Schmid, *Projections of digital nets and sequences*, Math. Comput. Simulation, 55 (2001), pp. 239–247.

[18] M. E. Silva and T. Barbe, *Quasi-Monte Carlo in finance: Extending for problems of high effective dimension*, Econ. Apl., 9 (2005), pp. 577–594.

[19] A. Singhee and R. A. Rutenbar, *From finance to flip flops: A study of fast quasi-Monte Carlo methods from computational finance applied to statistical circuit analysis*, in Proceedings of the 8th International Symposium on Quality Electronic Design (ISQED'07), IEEE Computer Society, Washington, DC, 2007, pp. 685–692.

[20] I. H. Sloan, F. Y. Kuo, and S. Joe, *Constructing randomly shifted lattice rules in weighted Sobolev spaces*, SIAM J. Numer. Anal., 40 (2002), pp. 1650–1665.

[21] I. H. Sloan, X. Wang, and H. Woźniakowski, *Finite-order weights imply tractability of multivariate integration*, J. Complexity, 20 (2004), pp. 46–74.

[22] I. H. Sloan and H. Woźniakowski, *When are quasi-Monte Carlo algorithms efficient for high dimensional integrals?*, J. Complexity, 14 (1998), pp. 1–33.

[23] I. M. Sobol′, *Distribution of points in a cube and approximate evaluation of integrals*, Zh. Vȳchisl. Mat. Mat. Fiz., 7 (1967), pp. 784–802 (in Russian); Comput. Maths. Math. Phys., 7 (1967), pp. 86–112 (in English).

[24] I. M. Sobol′, *Uniformly distributed sequences with an additional uniform property*, Zh. Vȳchisl. Mat. Mat. Fiz., 16 (1976), pp. 1332–1337 (in Russian); Comput. Maths. Math. Phys., 16 (1976), pp. 236–242 (in English).

[25] X. Wang and K. T. Fang, *The effective dimension and quasi-Monte Carlo integration*, J. Complexity, 19 (2003), pp. 101–124.
[26] X. Wang and I. H. Sloan, *Why are high-dimensional finance problems often of low effective dimension?*, SIAM J. Sci. Comput., 27 (2005), pp. 159–183.
[27] X. Wang and I. H. Sloan, *Low discrepancy sequences in high dimensions: How well are their projections distributed?*, J. Comput. Appl. Math., 213 (2008), pp. 366–386.