Thailand Statistician
July 2011; 9(2) : 103-127
http://statassoc.or.th
Contributed paper

Generation of Uniform Designs in the k-dimensional Ball

Wipawan Laoarun [a], John J. Borkowski [b], and Kamon Budsaba [a]*

[a] Department of Mathematics and Statistics, Faculty of Science and Technology, Thammasat University, Pathum Thani 12120, Thailand.

[b] Department of Mathematical Sciences, Montana State University, Bozeman, MT 59717 USA

*Author for correspondence; e-mail: kamon@mathstat.sci.tu.ac.th

Received: 6 February 2011
Accepted: 16 May 2011

Abstract

In this paper, we describe the generation of a uniformly scattered set of points in the k-dimensional ball (B_k) based on the transformation from the set of points in the k-dimensional unit cube (C^k). We apply four number-theoretic methods, which are described in Fang and Wang (1994) for generation of uniform designs in C^k, to generate the uniform design in B_k. These four number-theoretic methods are the good lattice point (GLP) method, the power-modulo a prime (PMP) method, the square root sequence (SRS) method and the Hammersley (HAM) set method. We study k =2, 3, and 4 dimensions with the number of design points (N) = 29, 53 and 98 for each k. Three distance-based criteria (the root mean squared distance (RMSD), the average distance (AD), and the maximum distance (MD)), are used for choosing the best design among a set of points and the RMSD is the major criterion for choosing the best design. When the number of factors (k) is small, the GLP is suitable for generation of a uniform design in B_k. The HAM and SRS methods are suitable when N and k are large.

Keywords: distance-based criteria, number-theoretic method, uniform design.

1. Introduction

When there are a large number of factors, the required number of runs in a full factorial design is very large. For example, if there are 10 factors with 6 levels for each factor, then the number of all possible runs is 6^{10} which is very large and, therefore, cannot actually be run. If a fractional-factorial design or an orthogonal design is used, the number of runs is at least 6^m for $m = 2, 3, \ldots, 9$. For small m, the model must be simple and interactions will be confounded with main effects. For larger m, the design size is again impractically large. In recent years, there has been rapid growth in the use of computer experimentation. Computer experiments have been used to study complex processes, and they require computer code to simulate part or all of a process that usually has many input variables. Inputting values for these variables will generate an "observation" or "response" from the computer code, but unlike the classical experiments on physical processes, computer experiments are deterministic. That is, every time the code is run with the same inputs, the outputted observations will be identical. Thus, there is no random error in the model. In addition, a larger experimental space (or domain) can be employed to explore complicated nonlinear functions. For the details of physical and computer experiments, see Fang, Li, and Sudjianto [1] and Santer, Williams and Notz [2].

Sacks, et al. [3] presented examples of computer experiments and concluded three primary objectives of computer experiments are: (i) predict the response at untried inputs, (ii) optimize a function of the response and (iii) tune the computer model (code) based on physical data. Because there is no random error, an additional assumption made in a computer experiment is that the true response surface (the relationship between the inputs and outputs) is unknown. When few or no details on the functional behavior of the response parameters are available, it is important to be able to obtain information from the entire experimental design space. For this reason, the methods for computer experiments tend to place samples representing settings of the input variables on the interior of the experimental design space in what is often termed a *space-filling* set of samples. Sampling in the interior of the experimental design space can reduce *bias error* resulting from assuming a model that does not adequately represent the true relationship between the input and output variables. *Space-filling* is one criterion that a design for computer experiments should adequately satisfy. Therefore, design points should be *evenly* or *uniformly* spread throughout the entire region (Husslage et al. [4]). A design which has its points placed throughout the space is called a *space-filling* design.

McKay, Beckman and Conover [5] were the first to explicitly consider experimental designs applied to a deterministic computer model. They proposed *Latin*

Hypercube Sampling (LHS), an extension of stratified sampling, which ensures that each of the input variables (factors) has all segments within its range represented. A design using a Latin hypercube sample can be computationally cheap to run because it can include relatively few combinations of input values and it can also cope with experiments having many input variables.

Fang [6] and Wang and Fang [7] proposed the *uniform design (UD)* and provided tables for certain classes of Uniform designs. Uniform design is another kind of the space-filling design. The Uniform design concept is to generate a set of experimental points using quasi-Monte Carlo methods or number-theoretic methods such that the points are uniformly scattered throughout the experimental space with low discrepancy. Uniform designs were initially applied in China for the design of new products in the textile and metallurgical industries, and in engineering and agriculture applications. See Liang, Fang and Xu [8] for a history of Uniform design applications.

Number-theoretic methods (NTMs) or quasi-Monte Carlo methods are special methods which represent a combination of number theory and numerical analysis. The methods have been applied to numerical integration techniques, but more recently have been applied in many applications in statistics. For example, Fang [6] and Wang and Fang [7] were the first to apply NTMs for generating a set of design points when the experimental space is a k-dimensional unit cube, denoted C^k , such that these points are uniformly scattered in C^k. In addition, Fang and Wang [9] gave a general outline regarding the transformation of the design points in C^k to other experimental design spaces. In particular, they discussed transformations from the unit cube to the ball and to the sphere, and to the simplex in mixture designs. The most common measure of assessing the uniformity in their research is the *discrepancy* and its various versions. A lower discrepancy value implies a better design based on uniformity of scatter of the points in the design space. See Hua and Wang [10], Niederreiter [11] and Hickernell [12] for the details of discrepancy and its versions.

Borkowski and Piepel [13] discussed two number-theoretic methods for generating uniform designs for single and multiple component constraints in the mixture designs. The two methods are called the one-pass exchange algorithm and the power-modulo-a-prime method. They used three distance-based criteria (the root mean squared distance (RMSD), the average distance (AD), and the maximum distance (MD)), for choosing the best design among a set of number-theoretic mixture designs (NTMDs).

The article is organized as follows: In Section 2, we describe the four NTMs for generation of uniformly scattered sets of points in C^k, which can be transformed to other

experimental spaces, in particular, the k-dimensional ball, denoted B_k. The four NTMs are the good lattice point (GLP) method, the power-modulo a prime (PMP) method, the square root sequence (SRS) method, and the Hammersley set (HAM) method. In Section 3, we describe the transformation method from the points in C^k to the uniformly scattered points in B_k. In Section 4, the three distance-based criteria for assessing the uniformity in this study are presented. Section 5 presents a comparison study of generation the uniform design in B_k, using 4 NTMs for low dimensions k such that k = 2, 3, and 4 dimensions, and for each k, the number of design points N can be considered in small (N = 29), medium (N = 53) and large (N = 98). The first two values of N are primes and the last is written in the form $N = 2p^l$, where p is an odd prime and $l \geq 1$, these are examples of number of design points which can be considered for all 4 NTMs. The conclusions and discussion are given in Section 6.

2. Methods for Generation uniformly Scattered Sets of Points in C^k

There are several methods for the generation of uniformly scattered sets of points in k- dimensional unit cube C^k. Fang and Wang [9] classified the set of points from these methods into 3 types; (i) good lattice point (GLP) sets generated using the good lattice point method or the power-modulo a prime method; (ii) good point (GP) sets generated using the square root sequence method, the power of the $(k+1)^{st}$ root method, or the cyclotomic field method; (iii) H-sets which are generated using the Halton set method or the Hammersley set method. In this paper, we will study only the following four methods:

I. Good Lattice Point (GLP) Method

This method was described by Wang and Fang [7]. For a given number of design points N and number of factors k, let $\boldsymbol{h} = (h_1, h_2, \ldots, h_k)$ be a vector of positive integers such that

a) the greatest common divisor of N and h_i (gcd (N, h_i)) = 1 for i = 1, 2, …, k and

b) $1 \leq h_1 < h_2 < \ldots < h_k < N$

These GLP sets can be generated as follows:

i) For each h_i ; i = 1, 2, …, k and for j = 1, 2, …, N, calculate

$$q_{ji} = j\, h_i \pmod N \qquad (1)$$

If $q_{ji} = 0$, then set $q_{ji} = N$. The values of q_{ji} form an $N \times k$ matrix $\boldsymbol{L}$ and because gcd (N, h_i) = 1, each column of $\boldsymbol{L}$ is equivalent to a permutation of 1, 2, …, N. The matrix $\boldsymbol{L}$ is called a *lattice point* matrix of integers.

ii) Create a matrix $\boldsymbol{C}$ by transforming each q_{ji} in $\boldsymbol{L}$ to

$$x_{ji} = \left\{ \frac{2q_{ji} - 1}{2N} \right\} \qquad (2)$$

Hence each column of $\boldsymbol{C}$ is equivalent to a permutation of $\frac{1}{2N}, \frac{3}{2N}, \ldots, \frac{2N-3}{2N}, \frac{2N-1}{2N}$.

Define $\boldsymbol{x}_j = (x_{j1}, x_{j2}, \ldots, x_{jk})$ to be j^{th} row of $\boldsymbol{C}$ and $\boldsymbol{P}_N$ be the set $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N\}$ formed from the N rows of $\boldsymbol{C}$. $\boldsymbol{P}_N$ is called the *lattice point set* of the *generating vector* (N;h_1, h_2, …, h_k). Wang and Fang [7] suggested that the number of factors k needs to be less than or equal to the *Euler function* $\Phi(N)$, which denotes the number of integers m satisfying $1 \le m < N$, and gcd(m , N) = 1. Suppose $N = p_1^{l_1} \cdot p_2^{l_2} \cdots p_r^{l_r}$ and $p_1, p_2, \ldots p_r$ are different primes, then $\Phi(N) = N\left(1-\frac{1}{p_1}\right)\cdots\left(1-\frac{1}{p_r}\right)$ and $\Phi(N) = N\text{-}1$ when N is prime. For example, take $N = 7$ points and $k = 3$ factors. There are $\Phi(7) = 6$ h_i values {1, 2, …, 6} which satisfy the condition gcd(h_i , 7) = 1. These h_i values yield the generating vector $\boldsymbol{h} = (1, h_2, h_3)$ where $1 < h_2 < h_3 < 7$. Suppose the generating vector is $\boldsymbol{h} = (1, 4, 5)$. The value of $h_1 = 1$, $h_2 = 4$ and $h_3 = 5$ are used to calculate q_{j1}, q_{j2} and q_{j3} for $j = 1, 2, \ldots, 7$ as in (1) to form a 7×3 matrix $\boldsymbol{L}$ and transforming each q_{ji} in $\boldsymbol{L}$ to x_{ji} as in (2) to form a 7×3 matrix $\boldsymbol{C}$. Hence, for $\boldsymbol{h} = (1, 4, 5)$, the matrices $\boldsymbol{L}$ and $\boldsymbol{C}$ are

$$\boldsymbol{L} = \begin{pmatrix} 1 & 4 & 5 \\ 2 & 1 & 3 \\ 3 & 5 & 1 \\ 4 & 2 & 6 \\ 5 & 6 & 4 \\ 6 & 3 & 2 \\ 7 & 7 & 7 \end{pmatrix}, \text{ and } \boldsymbol{C} = \begin{pmatrix} 1/14 & 7/14 & 9/14 \\ 3/14 & 1/14 & 7/14 \\ 5/14 & 9/14 & 1/14 \\ 7/14 & 3/14 & 11/14 \\ 9/14 & 11/14 & 7/14 \\ 11/14 & 5/14 & 3/14 \\ 13/14 & 13/14 & 13/14 \end{pmatrix}.$$

II. Power-modulo a prime (PMP) method

When the number of design points is a prime number p. Korobov [14] suggested the generating vector

$$\boldsymbol{h} = (h_1, h_2, \ldots, h_k) \equiv (1, a, a^2, \ldots, a^{k-1}) \pmod{p} \qquad (3)$$

where $1 < a < p$ and a can be chosen among the *primitive roots modulo p*. That is a satisfies

$$a^i \neq a^j \pmod{p} \quad ; 1 \leq i < j < p \tag{4}$$

The condition of gcd (N, h_i) = 1 is still needed. Note that this method depends on the set of primitive roots modulo p. If design size N is not a prime number, but N can be written in the form $N = 2, 4, p^l$ or $2p^l$, where p is an odd prime and $l \geq 1$, then a primitive root modulo N will exist. The number of primitive roots modulo N is $\Phi(\Phi(N))$. To find the primitive roots, we will apply the following definition from Niven and Zuckerman [15], the theorem and corollary from Rosen (1985, chapter 8)[16], and the condition in (4).

Definition A reduced residue set modulo N is a set of integers r_i such that gcd(r_i, N) = 1, $r_i \neq r_j \pmod{N}$ if $i \neq j$, and such that every x primes to N is congruent modulo N to some member r_i of the set.

Theorem If a and N are relatively prime positive integers and if a is a primitive root modulo N, then the integers $a^1, a^2, \ldots, a^{\Phi(N)}$ form a reduced residue set modulo N.

It is clear that $a^1, a^2, \ldots, a^{\Phi(N)}$ form a reduced residue set modulo N if and only if gcd(a^i, N) = 1 for $i = 1, 2, \ldots, \Phi(N)$ and $a^1, a^2, \ldots, a^{\Phi(N)}$ are incongruent in pairs modulo N.

Corollary Let a be a primitive root modulo m. Then a^u is a primitive root modulo m if and only if gcd(u, $\Phi(m)$) = 1.

In practice, we use the smallest primitive root for generating other primitive roots. The next step is to use the generating vector $\boldsymbol{h}$ in (3) to generate the q_{ji} as in (1) and then transform each q_{ji} to x_{ji} as in (2). This will yield the matrix $\boldsymbol{C}$ and the set $\boldsymbol{P}_N$ formed by the rows of $\boldsymbol{C}$. For the PMP method, $\boldsymbol{P}_N$ is called the *lattice point set*. For example, consider $N = 7$ points and $k = 3$ factors. The number of primitive roots modulo 7 is 2, and the smallest primitive root modulo 7 is 3. Hence the generating vector $\boldsymbol{h} = (1, 3, 3^2) \bmod 7 = (1, 3, 2)$, and then it is used to generate q_{ji} as in (1) and transforming each q_{ji} in $\boldsymbol{L}$ to x_{ji} as in (2).

III. Square root sequence (SRS) method

This method uses a *good point vector* $\boldsymbol{\gamma}$ which contains the square root of *k* different prime numbers $p_1, p_2, \ldots, p_k$ to generate a set of *N* points $\boldsymbol{P}_N$. Specifically

$$\boldsymbol{\gamma} = \left(\sqrt{p_1},\ldots,\sqrt{p_k}\right) \tag{5}$$

and $$\boldsymbol{P}_N = \left\{\left(\left\{\gamma_1 \cdot j\right\},\ldots,\left\{\gamma_k \cdot j\right\}\right)\right\} \quad ; j = 1, 2, \ldots, N \tag{6}$$

and $\left\{\gamma_i \cdot j\right\}$ stands for the fractional part of $\gamma_i \cdot j$. For example, consider *N* = 7 points and *k* = 3 factors. Using three different prime numbers p_1=*2,* p_2 =*3,* and p_3 = 5 to generate a good point vector $\boldsymbol{\gamma}$. A good point vector $\boldsymbol{\gamma}$ is $\boldsymbol{\gamma} = (\sqrt{2},\sqrt{3},\sqrt{5})$. Hence, a set of 7 points $\boldsymbol{P}_7$ is

$$\boldsymbol{P_7} = \begin{pmatrix} 0.4142 & 0.7321 & 0.2361 \\ 0.8284 & 0.4641 & 0.4721 \\ 0.2426 & 0.1962 & 0.7082 \\ 0.6569 & 0.9282 & 0.9443 \\ 0.0711 & 0.6603 & 0.1803 \\ 0.4853 & 0.3923 & 0.4164 \\ 0.8995 & 0.1244 & 0.6525 \end{pmatrix}.$$

IV. Hammersley set (HAM) method

This method suggested by Hammersley [17] is based on forming an H-set from the Halton set method. For *k*-1 different prime numbers $p_1, \ldots, p_{k-1}$, the Hammersley set is defined by the set of points $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N\}$ where

$$\boldsymbol{x_j} = \left(\frac{2j-1}{2N}, y_{p_1}(j),\ldots,y_{p_{k-1}}(j)\right) \quad ; j = 1, 2, \ldots, N. \tag{7}$$

and $$y_{p_i}(j) = b_0 p_i^{-1} + b_1 p_i^{-2} + \ldots + b_r p_i^{-r-1} \tag{8}$$

and the natural number *j* has a p_i *–digit representation*

$$j = b_0 + b_1 p_i + b_2 p_i^2 + \ldots + b_r p_i^r \tag{9}$$

such that $0 \le b_l \le p_i - 1$; $l = 0,1,2,\ldots,r$ and $i = 1, 2, \ldots, k$. From (9), *j* can be written in base-p_i notation $j = b_r b_{r-1} \ldots b_1 b_0$. In (8), the value of $y_{p_i}(j)$ is between 0 and 1, and $y_{p_i}(j)$ is called the *radical inverse of j with base* p_i. For example, take *N* = 7 points and *k* = 3 factors using 2 different prime numbers base p_1 = 2 and p_2 = 3. For $j = 1, 2, \ldots, 7$ each *j* can be written in the binary and tertiary representations, hence, the values of b_l for $y_2(j)$ and $y_3(j)$ are:

$p_1 = 2$					$p_2 = 3$				
j	b_0	b_1	b_2	$y_2(j)$	j	b_0	b_1	b_2	$y_3(j)$
$1 = 1+0(2)$	1			0.5	$1 = 1+0(3)$	1			0.3333
$2 = 0+1(2)$	0	1		0.25	$2 = 2+0(3)$	2			0.6667
$3 = 1+1(2)$	1	1		0.75	$3 = 0+1(3)$	0	1		0.1111
$4 = 0+0(2)+1(2)^2$	0	0	1	0.125	$4 = 1+1(3)$	1	1		0.4444
$5 = 1+0(2)+1(2)^2$	1	0	1	0.625	$5 = 2+1(3)$	2	1		0.7778
$6 = 0+1(2)+1(2)^2$	0	1	1	0.325	$6 = 0+2(3)$	0	2		0.2222
$7 = 1+1(2)+1(2)^2$	1	1	1	0.875	$7 = 1+2(3)$	1	2		0.5556

$y_2(j)$ and $y_3(j)$ yield $\boldsymbol{x}_j$ as in (7) and then the set of points is a 7×3 matrix $\boldsymbol{X}$, where

$$\boldsymbol{X} = \begin{pmatrix} 0.0714 & 0.5000 & 0.3333 \\ 0.2143 & 0.2500 & 0.6667 \\ 0.3571 & 0.7500 & 0.1111 \\ 0.5000 & 0.1250 & 0.4444 \\ 0.6429 & 0.6250 & 0.7778 \\ 0.7857 & 0.3750 & 0.2222 \\ 0.9286 & 0.8750 & 0.5556 \end{pmatrix}$$

For the details of these 4 methods and other methods, see Fang and Wang [9]. In the next section, we present details for the generation NT-nets in the k-dimensional Ball (B_k)

3. Methods for Generation of Uniformly Scattered Sets of Points in B_k

Assume the experimental space is the k- dimensional Ball B defined as

$$B_k = \{(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_k) : \sum_{i=1}^{k} x_i^2 \le 1 \}.$$

The following method for generation the uniformly scattered sets of points in B_k is due to Wang and Fang [18] which is also contained in Fang and Wang [9]. The approach is to generate an N- point NT- net in $\boldsymbol{C}^k$, and then map its points into B_k.

Let $\{\boldsymbol{c}_j ; j = 1, 2, \ldots, N\}$ be an NT-net in $\boldsymbol{C}^k$. Each element c_{ji} is generated by one of the NT methods described earlier. Let $\boldsymbol{x}_j$ be an NT-net in B_k. each element x_{ji} is computed by

$$x_{ji} = b_{j1} \prod_{m=2}^{i} S_{jm} C_{j,i+1} \quad \text{for } i = 1, 2, \ldots, k\text{-}1 \text{ and } j = 1, 2, \ldots, N.$$

$$x_{jk} = b_{j1} \prod_{m=2}^{k} S_{jm} \qquad (10)$$

where

$S_{jm} = \sin(\pi b_{jm})$, $C_{jm} = \cos(\pi b_{jm})$, for $m = 1, 2, \ldots, k\text{-}1$ and $j = 1, 2, \ldots, N$.

$S_{jk} = \sin(2\pi b_{jk})$, $\boldsymbol{C_{jk}} = \cos(2\pi b_{jk})$

$b_{j1} = c_{j1}^{1/k}$, $b_{jm} = F_m^{-1}(c_{jm})$, for $m = 1, 2, \ldots, k$

This method for generating uniformly scattered sets of points $\{ \boldsymbol{X}_1, \boldsymbol{X}_2, \ldots, \boldsymbol{X}_N \}$ is based on the following theory of transformations.

Let vector $\boldsymbol{X} = (\boldsymbol{X}_1, \boldsymbol{X}_2, \ldots, \boldsymbol{X}_k)$ be uniformly distributed in B_k. Using the spherical coordinate transformation, this set of uniformly distributed points in B_k has the form

$$\boldsymbol{X}_i = \phi_1 S_2 \cdots S_i C_{i+1}, \quad \text{for } i = 1, 2, \ldots, k\text{-}1 \qquad (11)$$

$$\boldsymbol{X}_k = \phi_1 S_2 \cdots S_{k-1} S_k$$

where

$S_m = \sin(\pi\phi_m)$, $C_m = \cos(\pi\phi_m)$, for $m = 1,2, \ldots, k\text{-}1$

$S_k = \sin(2\pi\phi_k)$, $C_k = \cos(2\pi\phi_k)$,

and $\boldsymbol{\phi} = (\phi_1, \phi_2, \ldots, \phi_k) \in C^k$ is a $1 \times k$ vector such that

1. $\phi_1, \phi_2, \ldots, \phi_k$ are mutually independent and

2. ϕ_i has p.d.f. $p_i(\phi) = \begin{cases} k\phi^{k-1} & , \text{if } i = 1 \\ \dfrac{\pi(\sin(\pi\phi))^{k-i}}{B\left(\frac{1}{2}, \frac{k-i+1}{2}\right)} & , \text{if } i = 2, \ldots, k \end{cases} \qquad (12)$

where $0 \le \phi \le 1$ and $B(a,b)$ denotes the beta function. Note that $p_k(\phi) = 1$ which shows that $\phi_i \sim U(0,1)$. The c.d.f. of ϕ_i is

$$F_i(\phi) = \begin{cases} \phi^k & , \text{if } i = 1 \\ \dfrac{\pi}{B\left(\frac{1}{2}, \frac{k-i+1}{2}\right)} \displaystyle\int_0^{\phi} (\sin(\pi\phi))^{k-i}\, dx & , \text{if } i = 2, \ldots, k \end{cases} \qquad (13)$$

By applying the inverse transform method, we obtain the set of ϕ_i and then use the values of ϕ_i to compute the values of x_{ji} as in (10). Hence, we will obtain the uniformly scattered set in B_k. We give two examples of the generation of obtain the uniformly scattered sets in B_k for k = 2 and 3 factors in the Appendix.

4. Design Comparison Criteria

A uniform design is a design whose points are uniformly scattered over the design space. In the beginning of the study, the design space C^k is considered. Uniformity of scatter may be achieved by minimizing a discrepancy criterion. There are many versions of discrepancy used for assessing the uniformity. The *F*- discrepancy, the star L_p – discrepancy which has been used in quasi-Monte Carlo methods (Hua and Wang [10] and Niederreiter [11]). The modified L_2 - discrepancy proposed by Hickernell [12], which includes L_2 – discrepancies of projections of the design point in all lower dimensional subspace of C^k. The centered L_2 – discrepancy which contains all L_2 - discrepancies each calculated using one of the 2^k vertices of C^k as the origin. The wrap-around L_2 - discrepancy is calculated after wrapping around each one-dimensional subspace of C^k into a closed loop. The definitions and properties of these discrepancies can be found in Hickernell [12] and [19], Fang et al. [20], Fang, Li, and Sudjianto [1]. In 2009, Borkowski and Piepel [13] developed two number-theoretic methods for generating uniform designs for constrained mixture experiments. They used three distance-based measures: the root mean squared distance (rmsd (***X***)), the average distance (ad (***X***)), and the maximum distance (md (***X***)) to evaluate a design's uniformity of scatter.

In this study, three distance-based criteria for assessing the uniformity are used to compare the designs. These criteria are the rmsd (***X***), the ad (***X***), and the md (***X***) as in Borkowski and Piepel [13]. Let ***X*** be an *N*- point design and $\boldsymbol{u}_l$ be a random sample point in the set of evaluation points in B_k for l = 1,2, …, M. The rmsd, ad, and md distance criteria for a design with N points $\boldsymbol{P}_N = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N\}$ are defined as

$$\text{rmsd } (\boldsymbol{X}) = \sqrt{\frac{1}{M}\sum_{l=1}^{M}(d(\boldsymbol{u}_l, \boldsymbol{X}))^2} \tag{14}$$

$$\text{ad } (\boldsymbol{X}) = \frac{1}{M}\sum_{l=1}^{M} d(\boldsymbol{u}_l, \boldsymbol{X}) \tag{15}$$

$$\text{md } (\boldsymbol{X}) = \max_{l} d(\boldsymbol{u}_l, \boldsymbol{X}) \tag{16}$$

where $d(\boldsymbol{u}_l, \boldsymbol{X}) = \min_j d(\boldsymbol{u}_l, \boldsymbol{x}_j) = \min_j [(\boldsymbol{u}_l - \boldsymbol{x}_j)(\boldsymbol{u}_l - \boldsymbol{x}_j)']^{1/2}$ is the minimum of the Euclidean distance between evaluation set point $\boldsymbol{u}_l$ in B_k and the j^{th} design point $\boldsymbol{x}_j$ of $\boldsymbol{X}$. When comparing two designs, if the values of these criteria are smaller for the first design than the second design, then the implication is that the points in the first design are more uniformly scattered than the points in the other design.

5. Results of the Study

In this study, we consider the generation of NT-nets in the *k*- dimension ball (B_k) for *k* = 2, 3, and 4 dimensions, and for each k, we consider *N* = 29, 53 and 98 points. The methods used for generating NT-nets are the GLP, PMP, SRS and HAM methods. For each study, a random sample of 5,000 points in B_k was used to evaluate the three criteria for assessing the uniformity of scatter.

a) Results for *k* = 2 factors

i) When *N* = 29 points

Because *N* = 29 is prime number, there are 28 h_i values satisfying the condition gcd(*N*, h_i) = 1. Table 1 contains the three distance criteria values and the vectors generating NT-nets by the GLP method. When we compare designs based on the rmsd and ad values, the results are similar. The design generated by $\boldsymbol{h}$ = (1,8) is slightly better than the design generated by $\boldsymbol{h}$ = (1,21). But when we compare designs under the md criterion, the best design is generated by $\boldsymbol{h}$ = (1,12) with md = 0.270217. The worst design under all three criteria is generated by $\boldsymbol{h}$ = (1,28). Figures 1(a), 1(b), 1(c) and 1(d) are plots of the 29 points in B_2 resulting from generators $\boldsymbol{h}$ = (1,8) , $\boldsymbol{h}$ = (1,21), $\boldsymbol{h}$ = (1,12) and $\boldsymbol{h}$ = (1,28), respectively. Visually, we see that the design in Figures 1(a), 1(b) and 1(c) are similar with respect to the uniformity of the scatter but the design points in Figure 1(d) are not uniformly scattered.

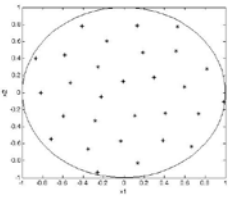
For the PMP method, there are $\Phi(\Phi(29)) = \Phi(28) = 12$ primitive roots modulo 29. The residue set modulo 29 for the least primitive root 2 is $\{2^1,2^3,2^5,2^9,2^{11},2^{13},2^{15},2^{17},2^{19},2^{23},2^{25}, 2^{27}\} \bmod(29) = \{2,8,3,19,18,14,27,21,26,10,11,15\} \bmod(29)$. Thus, the primitive roots modulo 29 are a_i = 2, 3, 8, 10, 11, 14, 15, 18, 19, 21, 26, and 27 for *i* = 1,2, …, 12, respectively. In the rightmost columns of Table 1, the three distance criteria values are sorted by the rmsd value for all possible generators $\boldsymbol{h}$ = (1, a_i) from the PMP method. The PMP method

results are same as the GLP having the same generator ***h***, but the worst design from the PMP method is generated from ***h*** = (1,15).

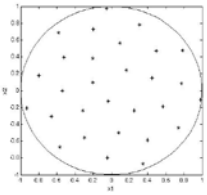
Table 1. Distance Criteria Values and the GLP and PMP method generators of NT-nets in B_2 for N = 29 points. These values are sorted by ascending rmsd (**X**) values.

GLP				PMP			
rmsd (**X**)	ad (**X**)	md (**X**)	Generator (***h***)	rmsd (**X**)	ad (**X**)	md (**X**)	Generator (***h***)
0.136439	**0.127265**	0.324344	(1,8)	**0.136439**	**0.127265**	0.324344	(1,8)
0.136690	0.127682	0.290507	(1,21)	0.136690	0.127682	**0.290507**	(1,21)
0.138037	0.128722	**0.270217**	(1,12)	0.141608	0.131478	0.302936	(1,18)
0.138540	0.129259	0.305499	(1,17)	0.142169	0.131760	0.330997	(1,11)
0.141608	0.131478	0.302936	(1,18)	0.144120	0.132605	0.298439	(1,3)
0.141694	0.131475	0.306881	(1,16)	0.146366	0.134783	0.305104	(1,26)
0.142169	0.131760	0.330997	(1,11)	0.161889	0.146306	0.334711	(1,2)
0.142257	0.131398	0.314182	(1,9)	0.163815	0.148114	0.337405	(1,27)
0.142508	0.131951	0.290091	(1,13)	0.190136	0.168927	0.427550	(1,10)
0.143230	0.132665	0.327854	(1,20)	0.190725	0.169198	0.467031	(1,19)
0.144120	0.132605	0.298439	(1,3)	0.214354	0.188895	0.516992	(1,14)
0.146366	0.134783	0.305104	(1,26)	0.215669	0.189886	0.501395	(1,15)
0.146987	0.135164	0.321314	(1,4)				
0.149088	0.137184	0.323464	(1,25)				
0.151711	0.138212	0.367389	(1,5)				
0.153432	0.140542	0.343198	(1,24)				
0.160957	0.146471	0.386431	(1,6)				
0.161486	0.146568	0.374005	(1,23)				
0.161889	0.146306	0.334711	(1,2)				
0.163815	0.148114	0.337405	(1,27)				
0.172043	0.154074	0.445655	(1,7)				
0.173880	0.156433	0.391837	(1,22)				
0.190136	0.168927	0.427550	(1,10)				
0.190725	0.169198	0.467031	(1,19)				
0.214354	0.188895	0.516992	(1,14)				
0.215669	0.189886	0.501395	(1,15)				
0.249531	0.218608	0.594201	(1,28)				

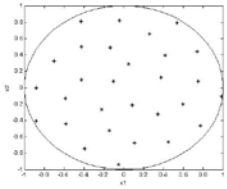
(a) NT-nets with $\boldsymbol{h}$ = (1,8)



(b) NT-nets with $\boldsymbol{h}$ = (1,21)



(c) NT-nets with $\boldsymbol{h}$ = (1,12)



(d) NT-nets with $\boldsymbol{h}$ = (1,28)

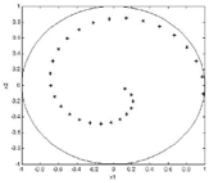


Figure 1. Plot of 29-point NT-nets in B_2 generated by the GLP method.

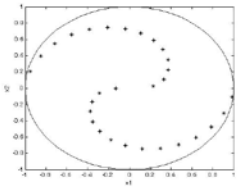


Figure 2. Plot of 29-point NT-nets in B_2 with the PMP method generator $\boldsymbol{h}$ = (1,15).

The results for the SRS method are contained in Table 2. Table 2 contains each *good point vector* $\boldsymbol{\gamma}$ having the form given in (5). The points in the NT-nets $\boldsymbol{P}_N$ corresponding to each $\boldsymbol{\gamma}$ are defined in (6). The design rankings based on the rmsd, ad and md values are the same. The design generated by $\boldsymbol{\gamma} = (\sqrt{2}, \sqrt{3})$ has the three smallest criteria values which indicates that its points are more uniformly scattered in B_2 than the points from the other designs. Figures 3(a), 3(b) and 3(c) are plots of the 29-point NT-nets for $\boldsymbol{\gamma} = (\sqrt{2}, \sqrt{3})$, $\boldsymbol{\gamma} = (\sqrt{3}, \sqrt{5})$, and $\boldsymbol{\gamma} = (\sqrt{2}, \sqrt{5})$, respectively.

Table 2. Distance Criteria Values and the SRS method good point generators of NT-nets in B_2 for N = 29 points. These values are sorted by ascending rmsd (**X**) values.

rmsd (**X**)	ad (**X**)	md (**X**)	Good Point (γ)
0.147757	**0.135413**	**0.350237**	$(\sqrt{2},\sqrt{3})$
0.159425	0.144816	0.394857	$(\sqrt{3},\sqrt{5})$
0.168435	0.154761	0.373294	$(\sqrt{2},\sqrt{5})$

(a) NT-nets with $\gamma = (\sqrt{2},\sqrt{3})$ (b) NT-nets with $\gamma = (\sqrt{3},\sqrt{5})$ (c) NT-nets with $\gamma = (\sqrt{2},\sqrt{5})$

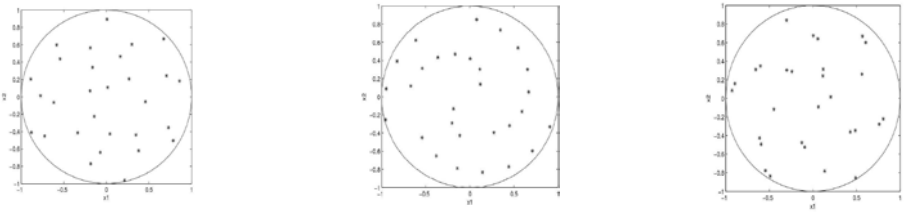


Figure 3. Plots of 29-point NT-nets in B_2 generated by the SRS method.

The results from the HAM method are contained in Table 3. For k = 2 , the NT-nets given by the form in (7) depend on one small prime number p. For N = 29, the possible values are p = 2, 3. The design of base p = 3 is slightly more uniformly scattered than the design of base p = 2. Plots of the points from these two designs are shown in Figures 4(a), and 4(b).

Table 3. Distance Criteria Values and the HAM method base (***p***) generators of NT-nets in B_2 for N = 29 points. These values are sorted by ascending rmsd (**X**) values.

rmsd (**X**)	ad (**X**)	md (**X**)	Base (***p***)
0.141230	**0.130585**	**0.301331**	3
0.147183	0.135514	0.332841	2

(a) NT-nets with ***p*** = 3 (b) NT-nets with ***p*** = 2

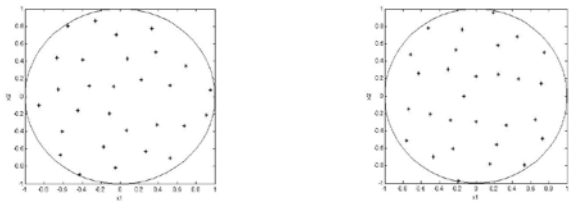


Figure 4. Plots of 29-point NT-nets in B_2 generated by the HAM method.

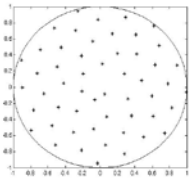
ii) When N = 53 points

The results for this case are summarized in Tables 4 to 6 and the plots in Figures 5 to 8. Table 4 gives the results of the best 24 designs generated by the GLP method and the results of all 24 possible designs generated by the PMP method. The value of h_2 in generating vector $\boldsymbol{h}$ is a primitive root modulo 53. For these two methods, the design rankings are similar based on the rmsd and ad values. The first 4 designs are the same based on rmsd and ad criteria with the best design generated from $\boldsymbol{h}$ = (1,45) which is slightly better than the design generated from $\boldsymbol{h}$ = (1, 12). However, when based on md values, the best design is generated from $\boldsymbol{h}$ = (1,8). For the GLP method, the worst design has generator $\boldsymbol{h}$ = (1, 52) with rmsd = 0.243044, ad = . 0.210891 and md = 0.590174 while the worst design from the PMP method is generated from $\boldsymbol{h}$ = (1, 27). Figures 5(a) to 5(d) and Figure 6 contain plots of the design points which correspond to the results from Table 4.

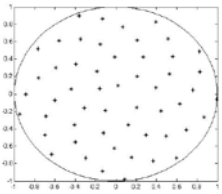
Table 4. Distance Criteria Values and the GLP and PMP method generators of NT-nets in B_2 for N = 53 points. These values are sorted by ascending rmsd (**X**) values.

GLP				PMP			
rmsd (**X**)	ad (**X**)	md (**X**)	Generator (***h***)	rmsd (**X**)	ad (**X**)	md (**X**)	Generator (***h***)
0.100843	**0.094405**	0.198189	(1,45)	**0.100843**	**0.094405**	0.198189	(1,45)
0.101587	0.094733	0.194988	(1,12)	0.101587	0.094733	0.194988	(1,12)
0.101700	0.095103	**0.190312**	(1,8)	0.101700	0.095103	**0.190312**	(1,8)
0.101792	0.094889	0.230793	(1,41)	0.101792	0.094889	0.230793	(1,41)
0.101969	0.094904	0.202372	(1,46)	0.103740	0.096558	0.197536	(1,39)
0.102450	0.095572	0.202592	(1,7)	0.103791	0.096183	0.201239	(1,14)
0.103005	0.096108	0.193654	(1,43)	0.104363	0.097188	0.231850	(1,34)
0.103420	0.096302	0.212518	(1,10)	0.104774	0.097517	0.213700	(1,19)
0.103740	0.096558	0.197536	(1,39)	0.104869	0.097259	0.219878	(1,22)
0.103791	0.096183	0.201239	(1,14)	0.105069	0.097246	0.220347	(1,48)
0.104363	0.097188	0.231850	(1,34)	0.105164	0.097635	0.220440	(1,5)
0.104774	0.097517	0.213700	(1,19)	0.105406	0.097473	0.243484	(1,31)
0.104869	0.097259	0.219878	(1,22)	0.118208	0.106382	0.266881	(1,3)
0.105069	0.097246	0.220347	(1,48)	0.119473	0.108128	0.271986	(1,33)
0.105164	0.097635	0.220440	(1,5)	0.120030	0.108776	0.306769	(1,20)
0.105406	0.097473	0.243484	(1,31)	0.120464	0.108189	0.268394	(1,50)
0.105546	0.097751	0.211633	(1,30)	0.147730	0.129581	0.318320	(1,2)
0.105668	0.097958	0.229091	(1,23)	0.148007	0.130965	0.359310	(1,32)
0.107387	0.099056	0.231176	(1,29)	0.148276	0.130759	0.344577	(1,21)
0.107918	0.099540	0.249192	(1,24)	0.149203	0.131149	0.320462	(1,51)
0.110060	0.100927	0.253969	(1,16)	0.182262	0.158611	0.421621	(1,18)
0.111105	0.102137	0.279052	(1,37)	0.183017	0.159085	0.458986	(1,35)
0.111194	0.101785	0.242294	(1,4)	0.207773	0.180227	0.497676	(1,26)
0.111981	0.102333	0.237632	(1,28)	0.209652	0.181813	0.485231	(1,27)

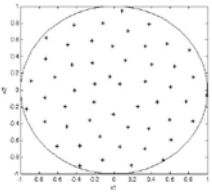
(a) NT-nets with ***h*** = (1,45)



(b) NT-nets with ***h*** = (1,12)



(c) NT-nets with ***h*** = (1,8)



(d) NT-nets with ***h*** = (1,52)

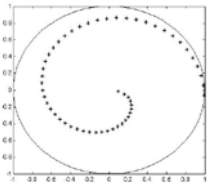


Figure 5. Plots of 53-point NT-nets in B_2 generated by the GLP method.

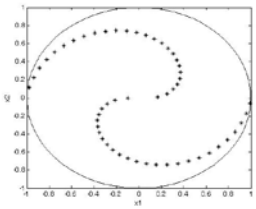


Figure 6. Plot of 53-point NT-nets in B_2 with the PMP method generator ***h*** = (1, 27).

Table 5 gives the results of three designs generated by the SRS method. For all three distance criteria, the best design is generated from $\boldsymbol{\gamma} = (\sqrt{2}, \sqrt{3})$. That is, the points of this design are more uniformly scattered than the points from the other two designs. This can be seen in Figure 7(a), 7(b), and 7(c).

Table 5. Distance Criteria Values and the SRS method good point generators of NT-nets in B_2 for N = 53 points. These values are sorted by ascending rmsd (**X**) values.

rmsd (**X**)	ad (**X**)	md (**X**)	Good Point ($\boldsymbol{\gamma}$)
0.117900	**0.108207**	**0.260046**	$(\sqrt{2}, \sqrt{3})$
0.120505	0.108567	0.283109	$(\sqrt{3}, \sqrt{5})$
0.151117	0.136365	0.373294	$(\sqrt{2}, \sqrt{5})$

(a) NT-nets with $\boldsymbol{\gamma} = (\sqrt{2}, \sqrt{3})$ (b) NT-nets with $\boldsymbol{\gamma} = (\sqrt{3}, \sqrt{5})$ (c) NT-nets with $\boldsymbol{\gamma} = (\sqrt{2}, \sqrt{5})$

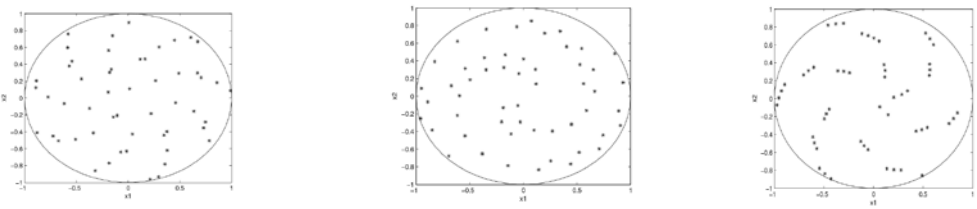


Figure 7. Plots of 53-point NT-nets in B_2 generated by the SRS method.

Table 6 contains the results of two designs generated by the HAM method. Small prime numbers p = 2, 3 were the base p generators. For all three distance criteria, the points of the base 3 design are more uniformly scattered than the points from the base 2 design. Plots of the points from these two designs are shown in Figures 8(a), and 8(b).

Table 6. Distance Criteria Values and the HAM method base (p) generators of NT-nets in B_2 for N = 53. These values are sorted by the ascending rmsd (**X**) values.

rmsd (**X**)	ad (**X**)	md (**X**)	Base (***p***)
0.103225	**0.095861**	**0.216255**	3
0.105922	0.097806	0.227075	2

(a) NT-nets with $\boldsymbol{p} = 3$ (b) NT-nets with $\boldsymbol{p} = 2$

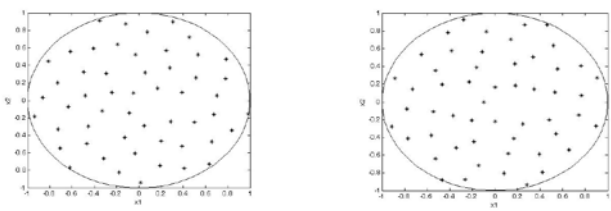


Figure 8. Plots of 53-point NT-nets **i**n B_2 generated by the HAM method.

iii) When N = 98 points

Since N = 98 is not a prime number but it is in the form of $2p^l$,where p = 7, then the number of primitive roots modulo 98 is equal to $\phi(\phi(98))$ (i.e,12). There are 12 possible primitive roots a_i = 3, 5, 17, 33, 45, 47, 59, 61, 67, 73, 75, 87, and 89 for the PMP method. Table 7 gives the results of the best 12 designs from all 41 possible

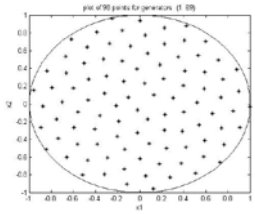
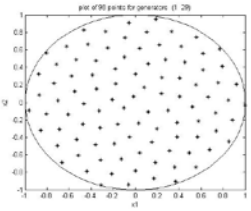
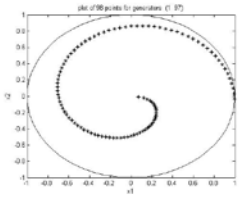
designs generated by the GLP method and the all 12 possible designs generated by the PMP method based on the rmsd.

Table 7. Distance Criteria Values and the GLP and PMP method generators of NT-nets in B_2 for N = 100 points. These values are sorted by ascending rmsd (**X**) values.

GLP				PMP			
rmsd (**X**)	ad (**X**)	md (**X**)	Gener-ator (***h***)	rmsd (**X**)	ad (**X**)	md (**X**)	Gener-ator (***h***)
0.072744	**0.068184**	**0.136632**	**(1,69)**	**0.074503**	**0.069308**	**0.151050**	**(1,45)**
0.073233	0.068612	0.143196	(1,29)	0.077076	0.071543	0.168998	(1,17)
0.073771	0.068868	0.153353	(1,71)	0.078335	0.072483	0.164642	(1,75)
0.073968	0.068964	0.155702	(1,43)	0.084080	0.076266	0.205485	(1,5)
0.074466	0.069396	0.164289	(1,55)	0.086681	0.078496	0.196285	(1,47)
0.074503	0.069308	0.151050	(1,45)	0.087142	0.079155	0.184846	(1,61)
0.074597	0.069662	0.149500	(1,27)	0.093301	0.084028	0.215441	(1,89)
0.075132	0.069904	0.171276	(1,53)	0.102685	0.090784	0.254789	(1,87)
0.076804	0.071301	0.166153	(1,81)	0.106462	0.092415	0.258674	(1,3)
0.077076	0.071543	0.168998	(1,17)	0.109496	0.096884	0.214207	(1,73)
0.078092	0.072340	0.159982	(1,57)	0.143176	0.123891	0.361536	(1,59)
0.078095	0.072250	0.156027	(1,23)	0.179240	0.153843	0.426044	(1,33)

The best design generated by the GLP method under all three distance criteria is generated from ***h*** = (1,69) which is slightly better than the designed generated from ***h*** = (1,29). The worst design under all three criteria is generated from ***h*** = (1,97) with rmsd = 0.239426, ad = 0.206785 and md = 0.597041 which is not shown in Table 7. Figures 9 (a), 9(b) and 9(c) are plots of the 98 points in B_2 resulting from generators ***h*** = (1,69) , ***h*** = (1,29) and ***h*** = (1,97), respectively. Visually, the designs in Figures 9(a) and 9(b) are similar with respect to the uniformity of the scatter while the design points in Figure 9(c) are not uniformly scattered. The best design generated by the PMP method is generated from ***h*** = (1,45) which is the sixth ranking of the design generated by the GLP method. The worst design by the PMP method is generated from ***h*** = (1,33). The 98 points plots in B_2 resulting from generators ***h*** = (1,45) and ***h*** = (1,33) are displayed in Figure 9(d) and 9(e).

(a) NT-nets with $\boldsymbol{h}$ = (1,69) (b) NT-nets with $\boldsymbol{h}$ = (1,29) (c) NT-nets with $\boldsymbol{h}$ = (1,97)

(d) NT-nets with $\boldsymbol{h}$ = (1,45) (e) NT-nets with $\boldsymbol{h}$ = (1,33)

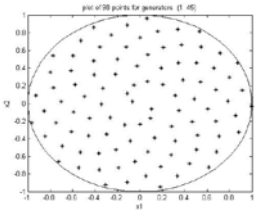
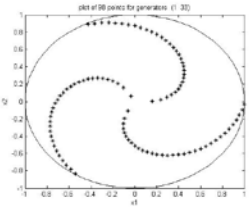



Figure 9. Plot of 98-point NT-nets in B_2 generated by the GLP method ((a)-(c)) and the PMP method ((d) and (e)).

Table 8 gives the results of three designs generated by the SRS method. For all three distance criteria, the best design is generated from $\boldsymbol{\gamma} = (\sqrt{3}, \sqrt{5})$. That is, the points of this design are more uniformly scattered than the points from the other two designs. This can be seen in Figure 10(a), 10(b), and 10(c).

Table 8. Distance Criteria Values and the SRS method good point generators of NT-nets in B_2 for N = 98 points. These values are sorted by ascending rmsd (**X**) values.

rmsd (**X**)	ad (**X**)	md (**X**)	Good Point ($\boldsymbol{\gamma}$)
0.079946	**0.073352**	**0.170911**	$(\sqrt{3}, \sqrt{5})$
0.101628	0.092084	0.228960	$(\sqrt{2}, \sqrt{3})$
0.124850	0.110021	0.330541	$(\sqrt{2}, \sqrt{5})$

(a) NT-nets with $\boldsymbol{\gamma} = (\sqrt{3}, \sqrt{5})$ (b) NT-nets with $\boldsymbol{\gamma} = (\sqrt{2}, \sqrt{3})$ (c) NT-nets with $\boldsymbol{\gamma} = (\sqrt{2}, \sqrt{5})$

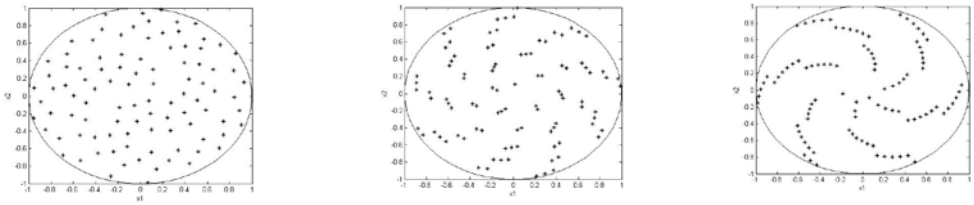


Figure 10. Plots of 98-point NT-nets in B_2 generated by the SRS method.

The results for the HAM method are presented in Table 9. Small prime numbers p = 2, 3 were the base p generators as the case N = 29 and 53. For all three distance criteria, the points of the base 3 design are more uniformly scattered than the points from the base 2 design. Plots of the points from these two designs are shown in Figures 11(a), and 11(b).

Table 9. Distance Criteria Values and the HAM method base (p) generators of NT-nets in B_2 for N = 100. These values are sorted by the ascending rmsd (**X**) values.

rmsd (**X**)	ad (**X**)	md (**X**)	Base (***p***)
0.075140	**0.069925**	**0.169136**	**3**
0.075964	0.070404	0.162888	2

(a) NT-nets with $\boldsymbol{p} = 3$ (b) NT-nets with $\boldsymbol{p} = 2$

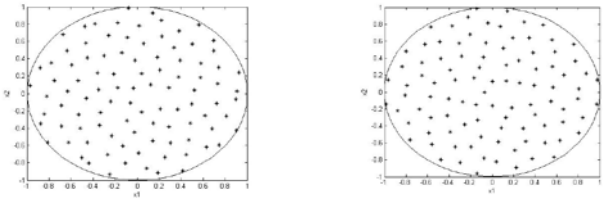


Figure 11. Plots of 98-point NT-nets **i**n B_2 generated by the HAM method.

b) Results for *k* = 3 and 4 factors

Tables 10 and 11 contain the results of the best design generated by each of the four methods for N = 29, 53 and 98 points based on the rmsd value criterion when k = 3 and 4 factors, respectively. For all cases of N, the GLP method is superior to other methods and it differs from the PMP method, while the SRS method gives the worst

results. As expected, when we consider each method, the three distance criteria values decrease as N increases.

Table 10. The best design (based on the rmsd value) generated by the GLP, PMP, SRS and HAM method for N = 29, 53 and 98 points when k = 3.

N	Method	rmsd (**X**)	ad (**X**)	md (**X**)	$\boldsymbol{h}$, $\boldsymbol{\gamma}$, or $\boldsymbol{p}$
29	GLP	0.274025	0.261650	0.528516	$\boldsymbol{h}$ = (1,16,18)
	PMP	0.276385	0.263915	0.560728	$\boldsymbol{h}$ = (1,21,6)
	SRS	0.289855	0.273437	0.662971	$\boldsymbol{\gamma}$ = $(\sqrt{2},\sqrt{3},\sqrt{7})$
	HAM	0.285972	0.271915	0.560395	$\boldsymbol{p}$ = 3, 5
53	GLP	0.223327	0.213482	0.420141	$\boldsymbol{h}$ = (1,33,38)
	PMP	0.225720	0.215369	0.433361	$\boldsymbol{h}$ = (1,41,38)
	SRS	0.234671	0.223105	0.507424	$\boldsymbol{\gamma}$ = $(\sqrt{2},\sqrt{3},\sqrt{7})$
	HAM	0.229844	0.218941	0.465015	$\boldsymbol{p}$ = 2, 3
98	GLP	0.179971	0.172159	0.328209	$\boldsymbol{h}$ = (1,57,87)
	PMP	0.186644	0.177738	0.358523	$\boldsymbol{h}$ = (1,17,93)
	SRS	0.193350	0.181937	0.376656	$\boldsymbol{\gamma}$ = $(\sqrt{2},\sqrt{3},\sqrt{5})$
	HAM	0.187132	0.178073	0.408362	$\boldsymbol{p}$ = 2, 3

Table 11. The best design (based on the rmsd value) generated by the GLP, PMP, SRS and HAM method for N = 29, 53 and 98 points when k = 4.

N	Method	rmsd (**X**)	ad (**X**)	md (**X**)	$\boldsymbol{h}$, $\boldsymbol{\gamma}$, or $\boldsymbol{p}$
29	GLP	0.402052	0.388903	0.696920	$\boldsymbol{h}$ = (1,5,7,16)
	PMP	0.407460	0.393178	0.759638	$\boldsymbol{h}$ = (1,8 6,19)
	SRS	0.415973	0.399717	0.779002	$\boldsymbol{\gamma}$ = $(\sqrt{2},\sqrt{3},\sqrt{5},\sqrt{7})$
	HAM	0.409966	0.395858	0.702240	$\boldsymbol{p}$ = 2,3,5
53	GLP	0.340157	0.329004	0.639111	$\boldsymbol{h}$ = (1,12,20,30)
	PMP	0.347659	0.335911	0.612382	$\boldsymbol{h}$ = (1,41 38,21)
	SRS	0.357600	0.344896	0.662651	$\boldsymbol{\gamma}$ = $(\sqrt{2},\sqrt{3},\sqrt{5},\sqrt{7})$
	HAM	0.352600	0.339672	0.660201	$\boldsymbol{p}$ = 2,3,7
98	GLP	0.288124	0.278573	0.522658	$\boldsymbol{h}$ = (1,53,83,93)
	PMP	0.295210	0.285214	0.524845	$\boldsymbol{h}$ = (1,77,29,33)
	SRS	0.301981	0.290668	0.628644	$\boldsymbol{\gamma}$ = $(\sqrt{2},\sqrt{5},\sqrt{7},\sqrt{11})$
	HAM	0.300306	0.289472	0.542292	$\boldsymbol{p}$ = 2,3,7

6. Conclusion and Discussion

In this study, we describe the generation of NT-nets on the k-dimensional ball (B_k) based on transformation the points from the k-dimensional unit cube (C^k) for k = 2, 3 and 4 dimensions. For each k, we studied primes N = 29 and 53, and the larger non-prime N = 98. The methods used for generating NT-nets are the GLP, PMP, SRS and

HAM methods. For each study, a random sample of 5,000 points in B_k was used to evaluate the three criteria for assessing the uniformity of scatter.

The number of possible designs generated by the GLP method depends on the number of the integer h_i such that gcd(h_i, N) = 1 for i = 2,3, …., k. Hence when N is a prime number and k is large, the number of possible designs is also large and this requires extensive computing time. The number of possible designs generated by the PMP method depends on the number of the primitive roots modulo N which has fewer possible designs than the GLP method. However, when N and k are large, the PMP method may require long computing time. The number of possible designs generated by the SRS and HAM methods depends on k and the first k-1 prime numbers. Hence, these last two NTMs require less computing time than the GLP and PMP methods.

The results from this study suggest that the GLP method may, in general, generate the best uniform designs. The PMP, HAM and SRS rank second, third and fourth rank with respect to design generation based on the rmsd criterion for all N and k. When k = 2, the best design generators for the GLP and PMP methods were the same. However, for each N and k, the criterion values are slightly different. When we compare the distance criteria across N for each method and k, the three criteria values decrease as N increases.

When N and k are large, the SRS and HAM methods yield criteria values which are slightly greater than the GLP and PMP methods and these differences decrease with increasing N and k. However, these methods require less computing time than the GLP and PMP methods. Thus, the SRS and HAM method may be considered for construction of designs having uniformly scattered design points in B_k for large N and k.

Acknowledgements

We would like to thank Thammasat University for partial support of this research.

References

[1] Fang, K.Y., Li, R. and Sudjianto, A., *Design and Modeling for Computer Experiments,* London : Chapman & Hall / CRC, 2006.

[2] Santer, T.j., Williams, B.J. and Notz. W.I., *The Design and Analysis of Computer Experiments*, Springer, New York, 2003.

[3] Sacks, J., Welch, W.J. Mitchell, T.J. and Wynn, H.P., Design and analysis of computer experiments, *Statist. Sci.,*1989; 4: 409-435.

[4] Husslage, B., Rennen, G. van Dam, E.R., and den Hertog D. (2010). Space-filling Latin hypercube designs for computer experiments*, Optimization and Engineering*, 2010: 1-20. Available from URL: http:// www.sprinkerlink.com/content /85v0742814128352/fulltext.pdf.

[5] McKay, M.D., Beckman, R.J. and Canover, W.J., A comparison of three methods for selecting values of input variables in the analysis of output from a computer code, *Technometrics,* 1979; 21: 239-245.

[6] Fang, K.T., The Uniform Design : Application of Number-Theoretic Methods in Experimental Design, *Acta Mathematicae Applacatae, Sinica*, 1980; 3: 363-372.

[7] Wang, Y. and Fang, K.T., A note on Uniform Distribution and Experimental Design, *Kexue Tongbao,* 1981; 26: 485-489.

[8] Liang, Y.Z., Fang, K.T. and Xu, Q.S., Uniform design and its applications chemistry and chemical engineering, *Chemometrics and Intelligent Laboratory Systems,* 2001; 58: 43-57.

[9] Fang, K.T. and Wang, Y., *Number-Theoretic Methods in Statistics*, London : Chapman & Hall, 1994.

[10] Hua, L.K. and Wang, Y., *An Application of Number Theory to Numerical Analysis*, Springer and Science Press : Berlin and Beijing, 1981.

[11] Niederreider, H., *Random Number Generation and Quasi-Monte Carlo methods.* Applied Mathematics, SIAM CBMS-NSF Regional Conference, Philadelphia, 1992.

[12] Hickernell, F.J., A Generalized Discrepancy and Quadrature Error Bound. *Math. Comp.*, 1998a; 67: 299-322.

[13] Borkowski, J.J. and Piepel, G.F., Uniform Design for Highly Constrained Mixture Experiments, *Journal of Quality Technology,* 2009; 41: 35-47.

[14] Korobov, N.M., Computation of multiple integrals by the method of optimal Coefficients, *Vestnik Moskow Univ. Ser. Mat. Astr. Fiz. Him*., 1959; 19-25.

[15] Niven, I. and Zuckerman, H. S., *An Introduction to the Theory of Numbers*, New York: John Wiley and Sons, 1980.

[16] Rosen, K.H., *Elementary Number Theory and Its Applications*, USA: Addison-Wesley Publishing Company, 1985.

[17] Hammersley, J.M., Monte-Carlo methods for solving multivariable problems, *Ann. New York Acad. Sci.,* 1960; 86: 844-874.

[18] Wang, Y. and Fang, K.T., Number Theoretic Method in Applied Statistics, *Chin. Ann. Of math. Series. B.,* 1990; 11: 51-65.

[19] Hickernell, F.J., Lattice rules: How well do they measure up?, in P. Hellekalek and G. Larcher, *Random and Quasi-Random Point Sets.*, 1998b: 106-166, Springer-Verlag, New York.

[20] Fang, K.Y., Lin, P., Winker, Y. and Zhang, Y., Uniform Design: Theory and Application, *Technometrics,* 2000; 42: 237-248.

Appendix

Generation of uniformly Scattered Sets of Points in B_k

We will give the examples for generating the set of uniformly scattered points in B_k for k = 2 and 3 factors by using the c.d.f of ϕ_i for i = 1, 2, .., k as in (13) and applying the inverse transform method to generate ϕ_i for i = 1, 2, .., k.

1) For k = 2 factors

Let $\boldsymbol{U}_i$ be mutually independent with U(0,1) distribution for i = 1, 2, …, k. From (13), the

c.d.f of ϕ_1 and ϕ_2 are $F_1(\phi) = \phi^2$ and $F_2(\phi) = \phi$. Let $\boldsymbol{U}_1 = F_1(\phi) = \phi^2$, then $\phi = F_1^{-1}(u_1)$ $= U_1^{1/2}$. Suppose $\phi_1 = \phi$, hence $\phi_1 = U_1^{1/2}$. That is, for each element of ϕ_1 and $\boldsymbol{U}_1$, we obtain $\phi_{j1} = U_{j1}^{1/2} = b_{j1}$ in (10) for j = 1, 2, …, N. Let $\boldsymbol{U}_2 = F_2(\phi) = \phi$, then $\phi = F_2^{-1}(u_2)$ $= \boldsymbol{U}_2$. Suppose $\phi_2 = \phi$, hence $\phi_2 = \boldsymbol{U}_2$. That is, for each element of ϕ_2 and $\boldsymbol{U}_2$, we obtain $\phi_{j2} = U_{j2} = b_{j2}$ in (10). Substituting the values of b_{j1} and b_{j2} into(10), and by (11) it yields

$$\boldsymbol{X}_1 = \boldsymbol{U}_1^{1/2}\cos(2\pi\boldsymbol{U}_2) \quad \text{and} \quad \boldsymbol{X}_2 = \boldsymbol{U}_1^{1/2}\sin(2\pi\boldsymbol{U}_2),$$

from which the set of uniformly scattered points in B_2 $\{\boldsymbol{X}_1, \boldsymbol{X}_2\}$ will be obtained.

2) For k = 3 factors

From (13), the c.d.f of ϕ_1, ϕ_2 and ϕ_3 are $F_1(\phi) = \phi^2$, $F_2(\phi) = \frac{1}{2}(1-\cos(\pi\phi))$, and $F_3(\phi) = \phi$. By using the inverse transform method, the elements of ϕ_i for i = 1, 2, 3 are

$\phi_{j1} = U_{j1}^{1/3}$, $\phi_{j2} = \frac{1}{\pi}\arccos(1-2U_{j2})$, and $\phi_{j3} = U_{j3}$ for j = 1, 2, …, N. Hence, we obtain

$$\boldsymbol{X}_1 = U_1^{1/3}\,(1\text{-}2\boldsymbol{U}_2)\text{ ,}$$

$$\boldsymbol{X}_2 = 2\,U_1^{1/3}\,(U_2(1-U_2))^{1/2}\,\cos(2\pi\boldsymbol{U}_3)\text{ , and}$$

$$\boldsymbol{X}_3 = 2\,U_1^{1/3}\,(U_2(1-U_2))^{1/2}\,\sin(2\pi\boldsymbol{U}_3)\text{ ,}$$

from which the set of uniformly scattered points in B_3 {$\boldsymbol{X}_1$, $\boldsymbol{X}_2$, $\boldsymbol{X}_3$} will be obtained.