Construction of Space-Filling Designs for Three Input Variables Computer Experiments

Kazeem A. Osuolale, Waheed B. Yahya, Babatunde L. Adeleke

***Abstract*—**Latin hypercube designs (LHDs) have been applied in many computer experiments among the space-filling designs found in the literature. A LHD can be randomly generated but a randomly chosen LHD may have bad properties and thus act poorly in estimation and prediction. There is a connection between Latin squares and orthogonal arrays (OAs). A Latin square of order s involves an arrangement of s symbols in s rows and s columns, such that every symbol occurs once in each row and once in each column and this exists for every non-negative integer s. In this paper, a computer program was written to construct orthogonal array-based Latin hypercube designs (OA-LHDs). Orthogonal arrays (OAs) were constructed from Latin square of order s and the OAs constructed were afterward used to construct the desired Latin hypercube designs for three input variables for use in computer experiments. The LHDs constructed have better space-filling properties and they can be used in computer experiments that involve only three input factors. MATLAB 2012a computer package (www.mathworks.com/) was used for the development of the program that constructs the designs.

***Keywords*—**Computer Experiments, Latin Squares, Latin Hypercube Designs, Orthogonal Array, Space-filling Designs.

I. Introduction

DETERMINISTIC computer experiments are becoming more frequently used in many areas of science and engineering. This is mainly because the underlying physical experiments are too time consuming, expensive, or even impossible to perform.

Rapid growth in computer power has made it possible to perform deterministic experiments on simulators. Since the emergence of the first computer experiment conducted by Enrico Fermi and colleagues [13] in Los Alamos in 1953, scientists in diverse areas such as engineering, cosmology, particle physics and aircraft design have turned to computer experiments as an effective tool to understand their respective processes. For instance, in the design of a vehicle, computer experiments are used to study the effect of a collision of the vehicle with a barrier before manufacturing the prototype of the vehicle, [1]. Space-filling design like Latin hypercube design (LHD) is popularly used in designing computer experiments. In this paper we are particularly interested in constructing orthogonal array-based LHDs. Space-filling designs are designs that spread points evenly throughout the experimental region. A Latin square of order s is an s × s array of s symbols, each appearing s times, once in each row and once in each column.

The s rows and s columns are labelled by the same set of symbols and are used to form an array in which each row contains the row-block symbol, the column-block symbol, and the Latin square treatment symbol for a cell in the orthogonal array constructed; this is simply an OA (s^2, 3, s, 2). A Latin square of order s where s is a prime power therefore produces OA (s^2, 3, s, 2).

Kazeem Osuolale is a Ph.D research student with the University of Ilorin, Kwara State Nigeria (phone: +2348066688932; e- mail: whereisqosimadewale@gmail.com.)

Waheed Yahya and Babatunde Adeleke are with the University of Ilorin, Kwara State, Nigeria (e-mail: wbyahya@unilorin.edu.ng, bladeleke@unilorin.edu.ng).

II. Orthogonal Arrays

Orthogonal arrays (OAs) were introduced by [11] followed by [2]. Orthogonal arrays (OAs) are essential in statistics and they are used in computer science and cryptography. In statistics they are primarily used in designing experiments which simply means that they are immensely important in all areas of human investigation. Orthogonal arrays are used in medicine, agriculture and manufacturing. Your automobile lasts longer today because of orthogonal arrays [3]. Pharmaceutical companies use orthogonal arrays to investigate the stability and shelf life of drugs which commonly involve many different factors.

The use of orthogonal arrays can lead to more economical tests and provide better statistical information. An orthogonal array of N runs, k factors, s levels and strength $t \geq 2$ is an N –by- k matrix with entries from a set of s levels, usually taken as 0. . . s-1, such that for every N-by- t submatrix, each of the s^t level combinations occurs the same number of times. Such an array is denoted by OA (N, k, s, t). The number $\lambda = N/s^t$ is called the index of the array. The rows of the array represent the experiments or tests to be performed, the columns correspond to the different variables whose effects are being analyzed and the entries in the array specify the levels at which the variables are to be applied. The construction of OA-based Latin hypercube designs depends on the existence of orthogonal arrays. For an excellent review for the existence results of orthogonal arrays see, for example, [3] and [8]. A library of orthogonal arrays is also available on the N. J. A. Sloane website and the MktEx macro using the software SAS [5]. Another important problem in the study of OAs is to determine the minimal number of rows n in any OA (N, k, s, t) for given values k, s and t.

Theorem 1: (Rao's Inequalities) Rao's [12] inequalities serve as an upper bound on the number of columns that can be included in an OA of given N, s and t or a lower bound on the number of rows required for an OA of given k, s and t:

(i) $N \geq \sum_{i=0}^{u} \binom{K}{i} (s-1)^{i})^{i}$, if $t = 2u$ and

(ii) $N \geq \sum_{i=0}^{u}\binom{K}{i}(s-1)^i + \binom{k-1}{u}(s-1)^{u+1}$, if $t = 2u+1$ for $u \geq 0$

III. Latin Hypercube Designs

Latin hypercube designs were the first type of design proposed specifically for computer experiments [7]. LHDs do not have repeated runs. Latin hypercube designs have one-dimensional uniformity in that when projected on each dimension, each portion of the design region has a design point. However, random LHDs may not be a good choice with respect to some useful criteria such as maximin distance and orthogonality but can easily be constructed.

The maximin distance criterion first introduced by [4] maximizes the smallest distance between any two design points so that no two design points are too close. A maximin distance design spreads out its points evenly over the entire design region. There are numerous reasons for the Latin hypercube popularity. One possible good reason is that it allows the creation of experimental designs with as many points as needed or desired and that is why Latin hypercube designs are very well accepted, particularly in studying computer experiments because of flexibility in terms of data density and location, and in addition, non-collapsing and space-filling properties.

An N x k matrix D = (d_{ij}) is called a Latin hypercube of N runs for k factors if each column of D is a permutation of 1, . . . , N. There are two natural ways of generating design points in the unit cube $[0, 1]^k$ based on a given Latin hypercube. The first is through $L_{ij} = (d_{ij} - 0.5)/N$ with the N points given by (l_{i1}, . . . , l_{Nk}) with i = 1, . . . , N, j = 1, . . .,k. The other is through, $L_{ij} = (d_{ij} - u_{ij})/N$ with the N points given by (l_{i1}, . . . , l_{Nk}) with i = 1, . . . , N, j = 1, . . .,k where u_{ij} are independent random variables with a common uniform distribution on [0, 1]. The discrepancy between the two methods can be seen as follows. When projected onto each of the k variables, both methods have the property that one and only one of the N design points fall within each of the N small intervals defined by [0, 1/N), [1/N, 2/N), . . . , [(N − 1)/N, 1]. The first method gives the mid-points of these intervals while the second method gives the points that are uniformly distributed in their corresponding intervals. The second method was adopted in this work following the method of [10]. The Latin Hypercube method is a compromise between spread of points and uniform spacing. This method produces designs that mimic the uniform distribution. It is good to avoid giving the same name for (N, k, s, t) as we do for an orthogonal array because they do not all have an explicit statistical interpretation [14]. The most important one is s which does not refer to the number of levels of the design since the design has N levels. The other parameters N, k and t have similar interpretations.

IV. Orthogonal Array-Based Latin Hypercube Designs

Orthogonal array-based Latin hypercube designs were proposed by [15]. Reference [9] also presented a paper on the algorithm for constructing space-filling designs for Hadamard matrices of Orders 4λ and 8λ. These designs achieve better space-filling property. Orthogonal Array (OA) can generate a sample with better space-filling property than LHDs in that the former tend to place points both in the interior and on the boundary of the design space while the latter is more likely to have design points on the boundary. Orthogonal arrays are used to construct LHDs in this study to achieve better space-filling property. An OA based Latin hypercube design in the design space $[0, 1)^k$ can be generated. In addition to the univariate maximum stratification, an OA (N, k, s, t)-based Latin hypercube has the t-dimensional projection property that when projected onto any t columns, it has exactly λ points in each of the s^t cells. [16] provided a way to obtain OA-based Latin hypercubes based on single replicated full factorial designs and showed that if the underlying orthogonal array is optimal with respect to the maximin distance criterion, so is the corresponding OA-based Latin hypercube. Reference [6] considered searching for optimal OA-based Latin hypercubes that minimize

$$\sum_{i=1}^{N}\sum_{j\neq i}\frac{1}{d_{ij}^{2}}$$

where d_{ij} is the Euclidean distance, defined as

$$d_{ij} = \frac{l_{ij} + {}^{(N-1)}\!/_{2} + u_{ij}}{N}, i = 1,...N, j = 1,...k$$

with t = 2, between the ith and jth design points.

V. Materials and Methods

A computer program via MATLAB 2012a package was written to construct orthogonal array Latin hypercube designs (OALHDs). A Latin square of order s where s is a prime power produces OA (s^2, 3, s, 2) .

Orthogonal arrays (OAs) were constructed from Latin square of order s for two cases s= 3 and s= 7 and they are subsequently used to construct the desired orthogonal array-based Latin hypercube designs (OALHDs) using

$L_{ij} = (d_{ij} - U_{ij})/N$. The program sorted and ranked the entries in the array column by column ignoring ties and then applied the formula to arrive at the desired OALHDs. This works for three input factors computer experiment only. Upper case letters D and L stand for the orthogonal array and the corresponding Latin hypercube designs constructed from orthogonal array respectively.

VI. Results

1. Construction of OA (25, 3, 5, 2) LHD from Latin Square of Order 5

```
[D,L]=oa_test (3,5)
D =
    0    0    0
    0    1    1
    0    2    2
    0    3    3
```

0	4	4
1	0	1
1	1	2
1	2	3
1	3	4
1	4	0
2	0	2
2	1	3
2	2	4
2	3	0
2	4	1
3	0	3
3	1	4
3	2	0
3	3	1
3	4	2
4	0	4
4	1	0
4	2	1
4	3	2
4	4	3

L =

0.0200	0.0200	0.0200
0.0600	0.2200	0.2200
0.1000	0.4200	0.4200
0.1400	0.6200	0.6200
0.1800	0.8200	0.8200
0.2200	0.0600	0.2600
0.2600	0.2600	0.4600
0.3000	0.4600	0.6600
0.3400	0.6600	0.8600
0.3800	0.8600	0.0600
0.4200	0.1000	0.5000
0.4600	0.3000	0.7000
0.5000	0.5000	0.9000
0.5400	0.7000	0.1000
0.5800	0.9000	0.3000
0.6200	0.1400	0.7400
0.6600	0.3400	0.9400
0.7000	0.5400	0.1400
0.7400	0.7400	0.3400
0.7800	0.9400	0.5400
0.8200	0.1800	0.9800
0.8600	0.3800	0.1800
0.9000	0.5800	0.3800
0.9400	0.7800	0.5800
0.9800	0.9800	0.7800

2. Construction of OA (49, 3, 7, 2) LHD from Latin Square of Order 7

[D,L]=oa_test (3,7)

D =

0	0	0
0	1	1
0	2	2
0	3	3
0	4	4
0	5	5
0	6	6
1	0	1
1	1	2
1	2	3
1	3	4
1	4	5
1	5	6
1	6	0
2	0	2
2	1	3
2	2	4
2	3	5
2	4	6
2	5	0
2	6	1
3	0	3
3	1	4
3	2	5
3	3	6
3	4	0
3	5	1
3	6	2
4	0	4
4	1	5
4	2	6
4	3	0
4	4	1
4	5	2
4	6	3
5	0	5
5	1	6
5	2	0
5	3	1
5	4	2
5	5	3
5	6	4
6	0	6
6	1	0
6	2	1
6	3	2
6	4	3
6	5	4
6	6	5

L =

0.0102	0.0102	0.0102
0.0306	0.1531	0.1531
0.0510	0.2959	0.2959
0.0714	0.4388	0.4388
0.0918	0.5816	0.5816
0.1122	0.7245	0.7245
0.1327	0.8673	0.8673
0.1531	0.0306	0.1735
0.1735	0.1735	0.3163
0.1939	0.3163	0.4592
0.2143	0.4592	0.6020
0.2347	0.6020	0.7449
0.2551	0.7449	0.8878
0.2755	0.8878	0.0306
0.2959	0.0510	0.3367
0.3163	0.1939	0.4796
0.3367	0.3367	0.6224
0.3571	0.4796	0.7653
0.3776	0.6224	0.9082
0.3980	0.7653	0.0510
0.4184	0.9082	0.1939
0.4388	0.0714	0.5000
0.4592	0.2143	0.6429
0.4796	0.3571	0.7857
0.5000	0.5000	0.9286
0.5204	0.6429	0.0714
0.5408	0.7857	0.2143

0.5612	0.9286	0.3571
0.5816	0.0918	0.6633
0.6020	0.2347	0.8061
0.6224	0.3776	0.9490
0.6429	0.5204	0.0918
0.6633	0.6633	0.2347
0.6837	0.8061	0.3776
0.7041	0.9490	0.5204
0.7245	0.1122	0.8265
0.7449	0.2551	0.9694
0.7653	0.3980	0.1122
0.7857	0.5408	0.2551
0.8061	0.6837	0.3980
0.8265	0.8265	0.5408
0.8469	0.9694	0.6837
0.8673	0.1327	0.9898
0.8878	0.2755	0.1327
0.9082	0.4184	0.2755
0.9286	0.5612	0.4184
0.9490	0.7041	0.5612
0.9694	0.8469	0.7041
0.9898	0.9898	0.8469

VII. Discussion

From our results, the following were observed in Sections I and II as discussed below. 1 shows the result of construction of OA (25, 3, 5, 2)-LHD for Latin square of order 5 and 2 contains the result of construction of OA (49, 3, 7, 2)-LHD for Latin square of order 7. OA (25, 3, 5, 2)-LHD contains 25 runs (rows) with 3 factors (columns) while OA (49, 3, 7, 2)-LHD has 49 runs with 3 factors. In both cases we have been able to construct OALHDs for only 3 factors and strength of 2.

The program was initialized with U = 0.5 to achieve the desired orthogonal array Latin hypercube designs. The two cases constructed OALHDs for the strength of 2 only for 3 factors. Table I summarizes the results.

TABLE I
Orthogonal Array Based Latin Hypercube Designs (N, K) Constructed from Latin Square of Order S

OA (N, k, s, t)	OALHD(N, k)
(25, 3, 5, 2)	(25, 3)
(49, 3, 7, 2)	(49, 3)

VIII. Conclusion

Many authors and researchers have paid particular attention to the construction of Orthogonal Array Based Latin Hypercube Designs (OALHDs). There are several techniques and criteria available for the construction of space-filling designs. These include the use of special permutations, difference matrices, Galois fields and orthogonal arrays among others.

This study employed a MATLAB program to construct orthogonal array based Latin hypercube designs from Latin square of order s for 3 factors computer experiment. A technique described by [3] was thereafter used, via the MATLAB program to obtain their corresponding orthogonal arrays which gave rise to OA (N, k, s, t)-LHD.

All the two OALHDs constructed have space-filling properties and they achieve uniformity in low dimension as depicted in Figs. 1 and 2. We have made the construction of OALHDs easier by writing a computer program that runs in few seconds to get a desired design. The program is optimised using the maximin distance criterion as default. In conclusion, the designs constructed in this paper are more appropriate designs for computer experiments. They are also useful in control engineering. Research work is in progress to apply this method and utilize the design constructed in performing a simple pendulum experiment as a demonstrative example for computer experiments. We are also working on the extension of the method to incorporate more than three input variables computer experiments for future research.

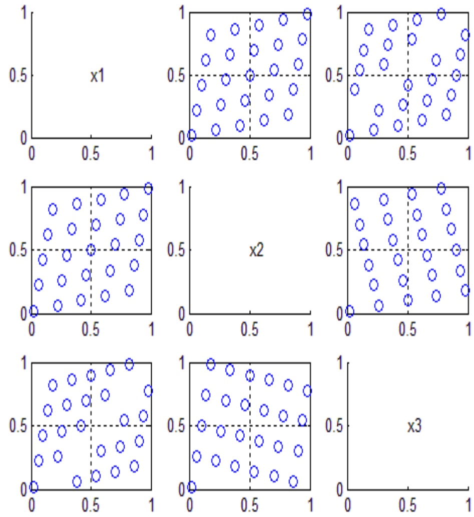

Fig. 1 The bivariate projections among x_1, x_2, x_3 for OA (25, 3, 5, 2) LHD showing the space-filling properties of the experimental design

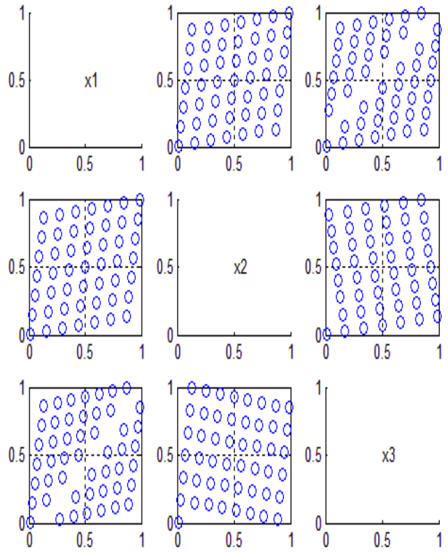

Fig. 2 The bivariate projections among x_1, x_2, x_3 OA (49, 3, 7, 2) LHD showing the space-filling properties of the experimental design

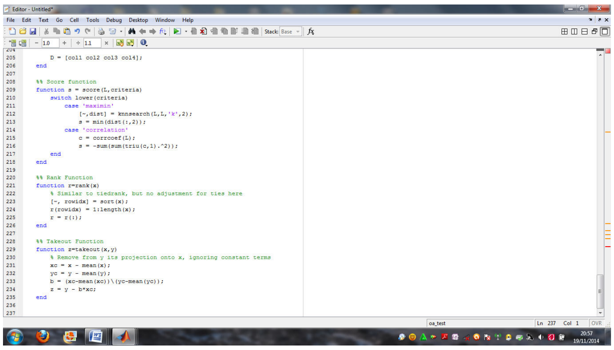

Fig. 3 MATLAB code for the construction of orthogonal array-based Latin hypercube designs

Acknowledgment

We appreciate Engr. Kunle Ifeta for supporting us in implementing the program in MATLAB.

References

[1] M. Bayarri, J. O. Berger, D. Higdon, M. Kennedy, A. Kottas, R. Paulo, J. Sacks, J. Cafeo, J. Cavendish and J. Tu (2002). A Framework for the Validation of Computer Models. Proceedings of the Workshop on Foundations for V&V in the 21st Century, D. Pace and S. Stevenson, eds., Society for Modelling and Simulation International.
[2] R. C. Bose and K. A. Bush (1952). Orthogonal Arrays of Strength Two and Three. Annals of Mathematical Statistics 23, 508–524.
[3] A. S. Hedayat, N. J. A. Sloane and J. Stufken (1999). Orthogonal Arrays: Theory and Applications. Springer-Verlag, New York.
[4] M. Johnson, L. Moore and D. Ylvisaker (1990). Minimax and Maximin Distance Design. J. Statist. Plann. Inference 26, 131-148.
[5] W. F. Kuhfeld (2009). Orthogonal Arrays. Website courtesy of SAS Institute.
[6] S. Leary, A. Bhaskar and A. Keane (2003). Optimal Orthogonal Aarray-Based Latin Hypercubes. J. Applied Statist. 30, 585-598.
[7] M. D. McKay, R. J. Beckman and W. J. Conover (1979),"A Comparison of Three Methods for selecting Values of Input Variables in the Analysis of Output from a Computer Code, Technometrics, 21, 239 -245.
[8] R. Mukerjee and C. F. Wu (2006). A Modern Theory of Factorial Designs. Springer Verlag.
[9] K. A. Osuolale, W. B. Yahya and B. L. Adeleke (2014). An Algorithm for constructing Space-Filling Designs for Hadamard Matrices of Orders 4λ and 8λ. Proceedings of the 49th Annual Conference of the Science Association of Nigeria, 27 th April - 1st May.
[10] A. B. Owen (1992a). A Central Limit Theorem for Latin Hypercube Sampling, Journal of the Royal Statistical Society. B54, 541-555.
[11] C. R. Rao (1946). Hypercube of Strength 'd' Leading to Confounded Designs in Factorial Experiments. Bull. Calcutta Math. Soc., 38, 67-78.
[12] C. R. Rao (1947). Factorial Experiments Derivable from Combinatorial Arrangements of Arrays. Journal of the Royal Statistical Society (Suppl.). 9, 128-139.
[13] S. Strogatz (2003). The Real Scientific Hero of 1953, New York Times. March 4, Editorial/Op-Ed.
[14] B. Tang (1991). Orthogonal Array-Based Latin Hypercubes. IIQP Research Report, RR-91-10.
[15] B. Tang (1993). Orthogonal Array-Based Latin Hypercubes. Journal of the American Statistical Association 88, 1392–1397.
[16] B. Tang (1994). A Theorem for selecting OA-Based Latin Hypercubes using a Distance Criterion. Comm. Statist. : A Theory and Methods 23, 2047-2058.