IMPACT OF INTEGRAL DIMENSIONS ON HALTON AND SOBOL SEQUENCES PERFORMANCE

NADIA AKRAM MOHAMMED[*] and SHAFIKA SULTAN ABDULLAH[**]
[*]Dept. of Mathematics, College of Basic Education, University of Duhok, Kurdistan Region-Iraq
[**]Dept. of Highways and Bridges Engineering, College of Technical Engineering,
Duhok Polytechnic University, Kurdistan Region-Iraq

ABSTRACT

Halton and Sobol sequences are two of the most popular number sets used in quasi-Monte Carlo methods. These sequences are effectively used instead of pseudo random numbers in the evaluation of integrals. In this paper, the two sequences are compared in terms of the size of the number sets and dimensionality. The comparison is implemented with Matlab programming for evaluating numerical integrals. The absolute error values of the investigated integral tests, with constant number of points (n) show that the optimum performance of Sobol is better than Halton at dimension 2 of test 1 and at dimension 2 of test 2. Performance of sequences have been analysed with different dimensions and (n)s. The practical results show that, except for the first dimension, Sobol sequence is better than Halton sequence with output values more stable and low-discrepancy feature when the dimension value is increased while this feature is deteriorated with Halton sequence.

KEYWORDS: Monte Carlo, Quasi-Monte Carlo, Halton sequence, Sobol sequence

1. INTRODUCTION

Monte Carlo (MC) method is a numerical method for solving mathematical problems by the simulation of random numbers. The most common applications of MC method are the evaluation of integrals, mathematical finance, tree search, and simulation in several branches of science (Levy 2010) (Landau & Binder 2005).

In numerical integral application, integrals which are difficult to be solved by analytic techniques are solved easily by MC methods. The MC methods are based on computer generation sequences of pseudo-random numbers. These sequences behave as if they are truly random although they are produced by deterministic algorithms. The convergence rate of approximating an integral with MC methods using a set of random samples of size n is $O(n^{-1/2})$ (Serre 2010).

Quasi-Monte Carlo (QMC) methods, also called low-discrepancy methods, use algorithms that reduce the discrepancy of generated random numbers. The low-discrepancy sequences cover the space better than pseudo-random sequences by reducing gaps between the points (Vajargah&Chayjani 2015). In computer programming, the generated sequences are matrices of size n-by-s, where n is the number of points and s is the dimension of the hypercube being sampled. Quasi-random numbers appear to cover the area more uniformly than a pseudo-random numbers. The advantage of using low-discrepancy sequences is a faster rate of convergence, it achieves a convergence rate of $O(n^{-1}(log\ n)^s)$ (Serre 2010). There are several high-dimensional sequences for use in QMC: Halton, Faure, and Sobol sequences. Other important sequences are Niederreiter and generalized Faure sequences (Krykova 2003).

There are several publications that compare MC and QMC methods (Owen 2008). QMC method have been successfully used for multivariate integration of high dimensions, and were significantly more efficient than MC method (Sloan & Woźniakowski 1998) (Kuo & Sloan 2013). One disadvantage of QMC integration is that its error estimation is difficult to be estimated, unlike MC integration (Tuffin 2008) (Jank 2005). Also it is stated that the problem of QMC methods is that their convergence is not independent of dimensionality (Frey 2008). Therefore, the

nadia.mohammed@uod.ac, shafika.sultan@dpu.edu.krd

accuracy of high-dimensional computations will not be guaranteed to be better than the MC. Lemieux mentioned within the drawbacks of QMC that the dimension needs to be small and the number of elements needs to be large in order to enhance the efficiency of QMC over the regular MC (Lemieux 2009). Morokoff and Caflisch remark that the advantage of the QMC method is better if the integrand is smooth, and the number of dimensions *s* of the integral is small (Morokoff & Caflisch 1995).

The aim of this paper is to implement a practical comparison between two of the most popular sequences used in QMC methods, Halton and Sobol sequences. The comparison concentrates on the effect of the size of random number sets and the number of dimensions on the accuracy of the integral evaluation. The approach is to use Matlab programming language in the application of two high-dimensional sequences, *Haltonset* and *Sobolset*, for generating the two sequences and apply them on a simulation of mathematical integrals. To show the contribution of the dimensions in the error of the numerical integration, 20 dimensions are used separately in evaluating the dimensional integral functions.

2. MONTE CARLO INTEGRATION

Monte Carlo approximation transforms the problem of integration into numerical method by calculating the average of the functions over random numbers. This is one of MC methods for evaluating integrals which is called sample-mean method. To find the integral of a function over an s-dimensional unit cube, this method approximates the solution to the average of the function at a randomly set of points $x_1, ..., x_n$:

$$\int_{[0,1]^s} f(u)du \approx \frac{1}{n}\sum_{i=1}^{n} f(x_i)$$

Where each x_i is a vector of *s* elements. Many trials are implemented for testing Haltonset and Sobolset sequences with different integrals and different sets of *n* and *s*. To explain the results, the integrals are evaluated with *n*=100000 for each value of *s*, where *s*=1, 2, …, 20. The multi-dimensional testing integrals 1 and 2 shown below are applied for the testing of the investigated QMC (Weir & Hass 2010) :

$$\int_1^e \int_1^e \dots \int_1^e \frac{1}{x_1 x_2 \dots x_s} dx_1 dx_2 \dots dx_s \qquad Test\ (1)$$
$$Exact\ Solution = 1$$

$$\left(\frac{2}{\sqrt{\pi}}\right)^s \int_0^2 \int_0^2 \dots \int_0^2 e^{-x_1^2 - x_2^2 \dots - x_s^2}\ dx_1 dx_2 \dots dx_s \qquad Test(2)$$

Exact Solution =
erf(2)^d

The general form of the integral is:
$$I = \int_a^b f(x)\, dx$$

While in the sample-mean method, the n random numbers are taken from $x \in [0,1]$ and the integral is estimated as shown as below:
$$I_{est} = (b-a)\frac{1}{n}\sum_1^n f(x_i)$$

If the interval of integration is taken between a=0 and b=1, the integral is simplified to:
$$I_{est} = \frac{1}{n}\sum_1^n f(x_i)$$

To illustrate the results of evaluating the integrals, Matlab programming is used to simulate the MC sample-mean method in two successive phases. In the first phase, the *Haltonset* and *Sobolset* sequences are used to generate 20 dimensions for evaluating integral tests 1 and 2 in the interval [0, 1]. The absolute error, which represents the absolute difference between the exact solution of the integral I_{exc}, and the estimated solution I_{est}, is used to evaluate the efficiency of each of the investigated sequences with a fixed value of *n*=100000 for each of the 20 generated dimensions, It is given by:
$$AsoluteError = |I_{est} - I_{exc}|$$

For the second phase, The number of points (n) are tested with different escalating values: 10000, 20000, 30000, ….., 100000, for each of the

twenty dimensions, the Mean Absolute Error (MAE) and Root of Mean Squared Error (RMSE) coefficients have been adopted for evaluating the performance of sequences to identify the effect of "n" on the accuracy of integral results (Chai &Draxler 2014) .

$$e = I_{est} - I_{exc}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|e_i|$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{1}^{n}e_i^2}$$

3. RESULTS AND DISCUSSION

The multi-dimensional integrals (test 1 and 2) are evaluated with each of the twenty generated dimensions using Halton and Sobol sequences, as illustrated in figures (1) and (2) respectively. The absolute error of estimated integrals is used as an indication of determining the efficiency of the two targeted sequences in relation with variation of dimensions.
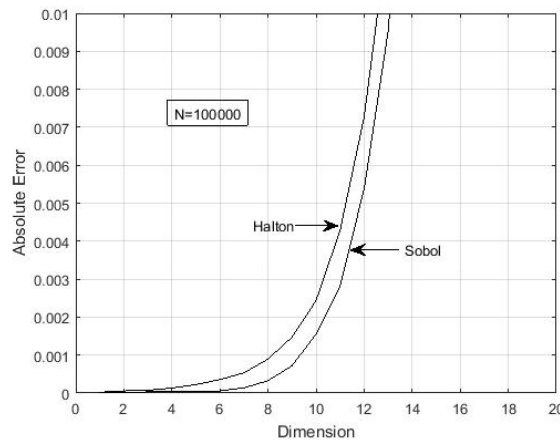

Fig (1): Results of integral test (1)

Table (1). Absolute Error values for the twenty dimensions of integral test (1)

Dimension	Absolute Error (Halton)	Absolute Error (Sobol)
1	0.000023	0.000023
2	0.000057	0.000024
3	0.000076	0.000030
4	0.000130	0.000040
5	0.000226	0.000052
6	0.000354	0.000057
7	0.000532	0.000137
8	0.000896	0.000326
9	0.001471	0.000720
10	0.002437	0.001552
11	0.004284	0.002824
12	0.007265	0.005372
13	0.012137	0.009528
14	0.020538	0.017182

15	0.035890	0.030987
16	0.063520	0.054730
17	0.110068	0.095317
18	0.188930	0.165847
19	0.324581	0.286468
20	0.557954	0.498075

Values of absolute errors shown in table 1 indicate the identical behavioure of the two sequences at the first dimension, the superiority of Sobol on Halton starts escalating and show the optimum performance at dimension 2, this continues till dimension 6, where it reaches about 84% better absolute error than the Halton due to the low discrepancy of Sobol in comparison with Halton. The deterioration in the performance of both sequences starts from dimension 7 upwards till they reach the worst performance at the optimal investigated dimension (20), indicating lower efficiency of both sequences with high dimensions, nevertheless the Sobol keeps ahead with a 10.8% superiority over the Halton at dimension 20.

Figure 2 presents the impact of increasing the dimensions of integral test (2) on the absolute error output values. The homogeneity of produced results is almost similar to that of test (1), Sobol is identical with Halton at the starting dimension, and later it proves a better performance through lower values of absolute errors.
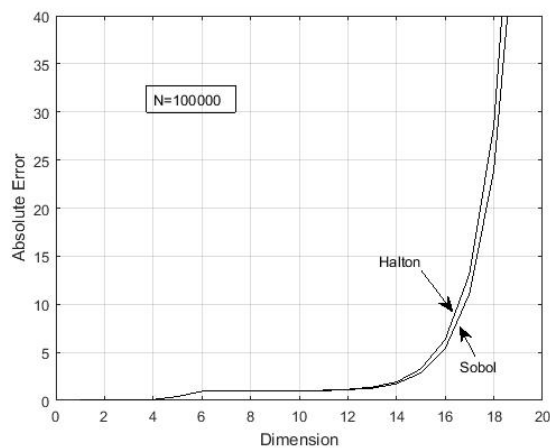

Fig. (2): Results of integral test (2)

Details of testing performance of both sequences in terms of absolue errors for integral test (2) are produced in table 2.

Table (2): Absolute Error values for the twenty dimensions of integral test (2)

Dimension	Absolute Error (Halton)	Absolute Error (Sobol)
1	0.00004	0.000046
2	0.00011	0.000048
3	0.00012	0.000049
4	0.00030	0.000066
5	0.00094	0.000067
6	0.00220	0.000420
7	0.00548	0.003078
8	0.01743	0.011050

9	0.03435	0.018189
10	0.05479	0.039505
11	0.10048	0.061206
12	0.20799	0.180819
13	0.42847	0.321062
14	0.98264	0.803258
15	2.34503	1.929460
16	5.40369	4.456555
17	12.2467	10.19418
18	27.6071	22.98396
19	62.4271	51.94117
20	141.1827	117.3588

Inspite of the fact revealed by results of table 2 which prove a relatively odd performance of both investigated sequences in the integration test (2) starting from dimension 6, reaching the worst results in dimension 20; nevertheless, Sobol performance is the best among the two sequences recordig about 54.54% higher than that of Halton at dimension 2, when n is maintained constant.

A Matlab simulation model is written to find MAE and RMSE for different values of *n* to show .

its effect on the accuracy of integral evaluation. The values of *n* are taken as 10000, 20000, …, 100000 with twenty different dimensions. These values are sufficient for revealing the effect of increasing *n* on the error with small and large values of the dimension. Results obtained from implementing integral test 1 and 2 using the Halton and Sobol sequences, in terms of MAE and RMSE outputs, are illustrated in figures 3 and 4 respectively
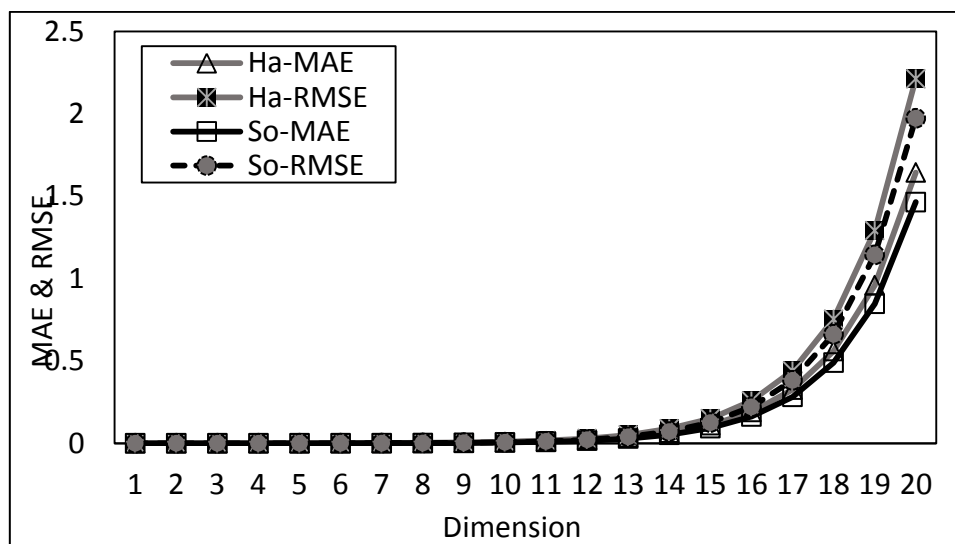

Fig.(3): Variation of MAE and RMSE with dimension using Halton and Sobol sequences for integral test (1)

Figure 3 shows that Sobol performance is better than that of Haltons in terms of MAE and RMSE, the superiority of Sobol comes clearer with higher dimension catagories.

The output values of MAE and RMSE shown in table 3 emphasize this conclusion, and give clearer vision of the actual performance of sequences.

Table (3): MAE and RMSE values for the twenty dimensions of integral test (1)

Dimension	Halton		Sobol	
	MAE	RMSE	MAE	RMSE
1	0.0001	0.0001	0.0001	0.0001
2	0.0001	0.0002	0.0001	0.0001
3	0.0002	0.0003	0.0001	0.0001
4	0.0004	0.0005	0.0001	0.0002
5	0.0006	0.0008	0.0002	0.0003
6	0.0009	0.0012	0.0003	0.0004
7	0.0014	0.0018	0.0005	0.0008
8	0.0024	0.0032	0.0011	0.0015
9	0.0040	0.0054	0.0024	0.0033
10	0.0071	0.0097	0.0046	0.0062
11	0.0126	0.0172	0.0084	0.0115
12	0.0216	0.0296	0.016	0.0217
13	0.0369	0.0506	0.0289	0.0394
14	0.0644	0.0880	0.0516	0.0709
15	0.1102	0.1496	0.0919	0.1254
16	0.1902	0.2572	0.1625	0.2209
17	0.3257	0.4396	0.2821	0.3821
18	0.5600	0.7537	0.4897	0.6616
19	0.9594	1.2913	0.8469	1.1434
20	1.6440	2.2123	1.4639	1.9727

From table 3, the performance of Halton in integral test 1 is good and keeps pace with Sobol on the first 3 dimensions, the results in dimension 7 show that Sobol in better by 67.3% of MAE and about 55.6% of RMSE than Halton, while in dimension 20 the performance of Sobol is nearer to that of halton with only 10.9% lower value of both MAE and RMSE, which is the worst output value, indicating the impact of dimension on the integral by both sequences.

Figure 4 is illustrating evaluation coefficients MAE and RMSE as output values in correlation with variation of dimension of integral test 2, visual inspection reveals again that the performance of both sequences is efficient and close to each other in early stages. Thses results tends to vary with the increase of dimension.
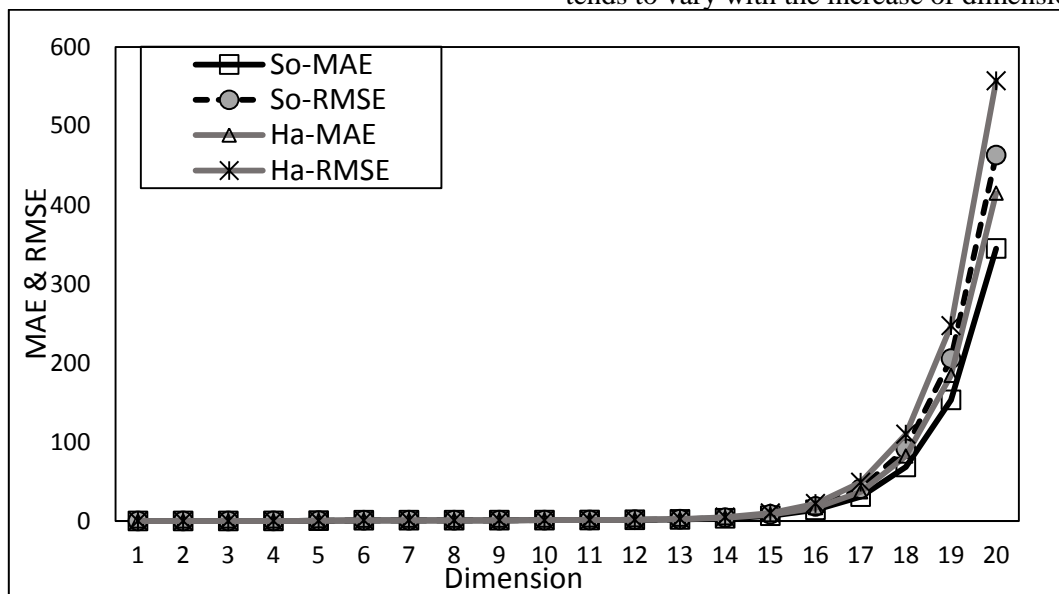

Fig. (4): Variation of MAE and RMSE with dimension using Halton and Sobol sequences for integral test (2)
Table 4 summarizes all output results of MAE and RMSE of integral test 2

Table (4): MAE and RMSE values for the twenty dimensions of integral test (2)

Dimension	Halton		Sobol	
	MAE	RMSE	MAE	RMSE
1	0.00012	0.00015	0.00012	0.00015
2	0.00029	0.00037	0.00015	0.00018
3	0.00036	0.00045	0.0002	0.00026
4	0.00088	0.00125	0.00038	0.00054
5	0.0026	0.0039	0.0005	0.0007
6	0.0054	0.0076	0.0008	0.0012
7	0.0061	0.0084	0.0048	0.0073
8	0.0331	0.0386	0.0239	0.0256
9	0.0683	0.0936	0.0472	0.0574
10	0.1497	0.2099	0.0879	0.1027
11	0.3036	0.4124	0.2248	0.2974
12	0.6209	0.8438	0.5442	0.7038
13	1.3449	1.8579	1.0404	1.4454
14	3.086	4.1991	2.4378	3.3471
15	7.0349	9.4968	5.7333	7.7614
16	15.8673	21.3789	13.1574	17.7329
17	35.948	48.3435	29.8706	40.1643
18	81.232	109.1832	67.4438	90.6912
19	183.3049	246.435	152.3147	204.7991
20	413.8803	556.2957	343.8953	462.2779

According to values of table 4, the impact of dimension on the performance of both sequences is very clear. Except the first dimenstion, all values in the table show that the performance of Sobol sequence is better than Halton sequence. However, the difference between the two sequences memains very small in dimensions 2-7, from dimension 8 and above, the difference increases with the increasing of the dimendions.

To find the reasons of the difference in accuracy of the results, it is important to see how the discrepancy is changed with changing

dimension values in both sequences. To do so, the sequence numbers of several pairs of dimensions are plotted and investigated. All the testing of outputs show that Sobol sequence fill the space better than Halton sequence, leading to improve the results of integral estimation. Figure 5 shows examples of the relation between pair of dimensions of the two sequences. In high dimensions, Halton sequence tends to be uniform in diagonal arrangement leaving large spaces in the square unit. This bad distribution will increase the error in estimating the value of the integral.
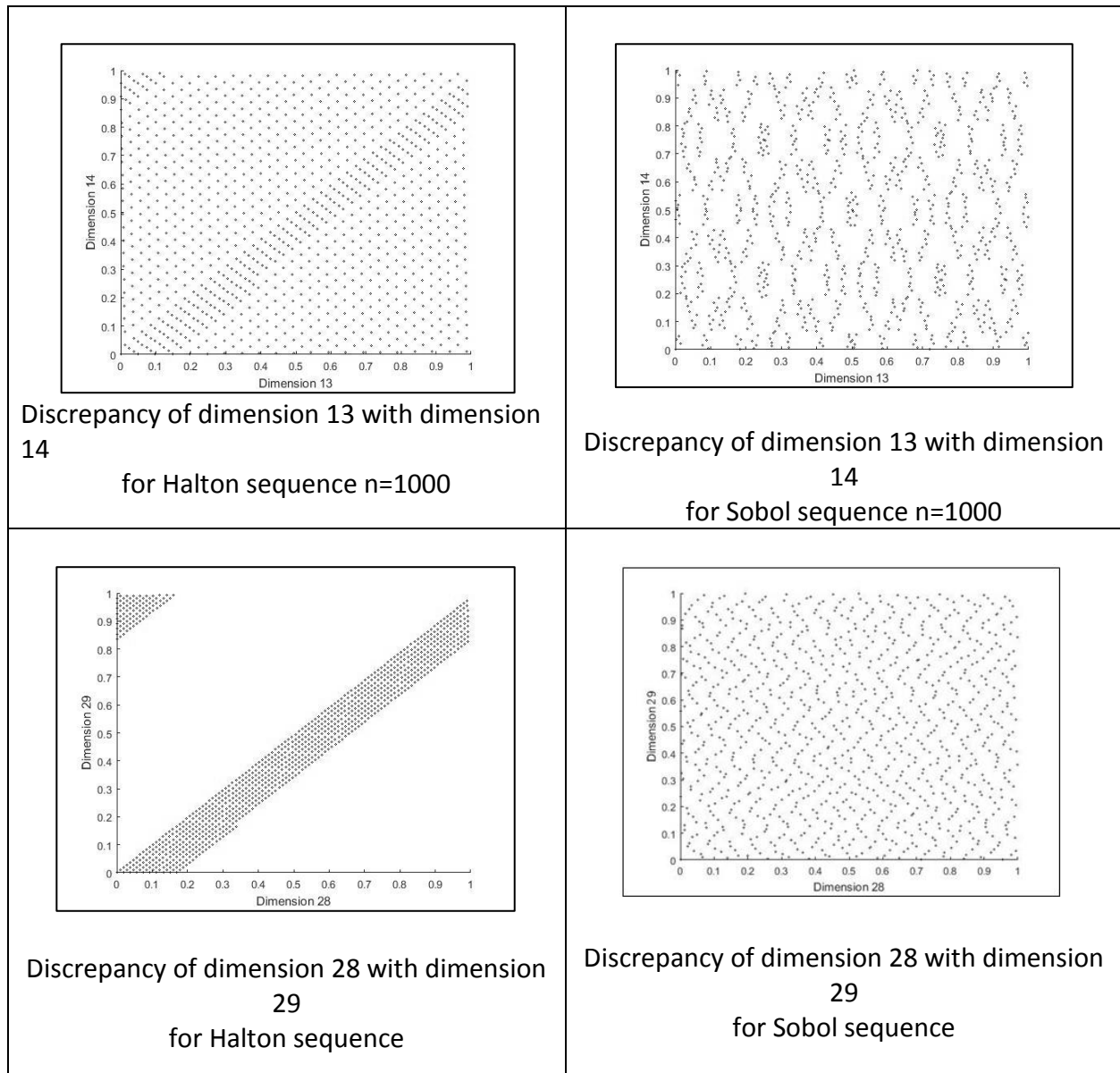
Sobol sequences maintains good distribution of the random number on all dimensions.

Discrepancy of dimension 13 with dimension 14

for Halton sequence n=1000

Discrepancy of dimension 13 with dimension 14
for Sobol sequence n=1000

Discrepancy of dimension 28 with dimension 29
for Halton sequence

Discrepancy of dimension 28 with dimension 29
for Sobol sequence

Fig. (5): Discrepancy plot of Halton sequences (left) and Sobol sequences (right)

It is worth mentioning that there are other parameters, such as 'leap' and 'skip', which can be used with Haltonset and Sobolset in Matlab to change the properties of the produced sets of the numbers. These parameters are not used in this work.

CONCLUSIONS

Halton and Sobol sets are used in Matlab for the generation of quasi-Monte Carlo sequences. These sequences are compared with different size of set numbers and different dimensions. The comparison is based on evaluating numerical integrals for one-dimensional functions by Matlab programming by means of evaluation coefficients. The results show that the performance of Sobol and Halton sequences depends highly on the dimension, yet, the performance of Sobol is better and more stable than the that of Halton. The results also show that Sobol sequence maintains the feature of low-discrepancy while this feature is deteriorated with Halton sequence when the dimension value is increased. These differences in discrepancy affect the results of integral evaluations by increasing the error evaluation criteria with increasing dimension of Halton sequence for different number of points (n). In the future work, the properties that change the sequences, such as *'skip'* and *'leap'*, which are available in Matlab, can be used for further analysis

nadia.mohammed@uod.ac, shafika.sultan@dpu.edu.krd

REFERENCES

− Chai, T. & Draxler, R. R., 2014. Root mean square error (RMSE) or mean absolute error (MAE)? – Arguments against avoiding RMSE in the literature. Geoscintific Model Development, 7, 1247–1250.

− Frey, R., 2008. *Monte Carlo methods: with application to the pricing of interest rate derivatives*. University of St. Gallen.

− Jank, W., 2005. Quasi-Monte Carlo sampling to improve the efficiency of Monte Carlo EM. *Computational statistics & data analysis*, 48(4), pp.685–701.

− Krykova, I., 2003. *Evaluating of Path-Dependent Securities with Low Discrepancy Methods*. Worcester Polytechnic Institue.

− Kuo, F.Y. & Sloan, I.H., 2013. High-dimensional integration : The quasi-Monte Carlo way. , pp.133–288.

− Landau, D.P. & Binder, K., 2005. *A Guide to Monte Carlo Simulations in Statistical Physics* 2nd ed., cambridge university press.

− Lemieux, C., 2009. *Monte Carlo and Qusi-Monte Carlo Sampling*, Springer.

− Levy, G., 2010. An introduction to quasi-random numbers. *Transport*, pp.1–16.

− Morokoff, W.J. & Caflisch, R.E., 1995. Quasi-monte carlo integration. *Journal of computational physics*, 122(2), pp.218–230.

− Owen, A.B., 2008. Monte Carlo and Quasi-Monte Carlo for Statistics,. In *Monte Carlo and Quasi-Monte Carlo Methods*. Springer Berlin Heidelberg., pp. 3–18.

− Serre, L., 2010. *A Matlab Program for Testing Quasi-Monte Carlo Constructions*. University of Waterloo.

− Sloan, I.H. & Woźniakowski, H., 1998. When Are Quasi-Monte Carlo Algorithms Efficient for High Dimensional Integrals? *Journal of Complexity*, 14(1), pp.1–33.

− Tuffin, B., 2008. Randomization of Quasi-Monte Carlo methods for error estimation: survey and normal approximation. *Monte Carlo Methods and Applications*, 10(3–4), pp.617–628.

− Weir, M.D. & Hass, J., 2010. *Thomas' Calculas*. Metric ed., Twelfth ed., Addison-Wesley-Pearson

− Vajargah, B.F. & Chayjani, T.A., 2015. Improving Niederreiter Quasi Random Numbers Generator in Simulation and Estimating High-dimensional Integrals. World appl. programming, 5(3), pp. 60-67.