

A Methodological Study on the Impact of Several Latin Hypercube Designs in Constructing Ordinary Kriging Models

Zainab Abdulateef Rasheed^{1*}, Younus Al-Taweel², Shaymaa M. Younus³

Department of Mathematics, College of Education for Pure Science, University of Mosul, Iraq. Email: zainab.abdulateef@uomosul.edu.iq

Department of Mathematics, College of Education for Pure Science, University of Mosul, Iraq. Email: younus.altaweel@uomosul.edu.iq

Department of Mathematics, College of Education for Pure Science, University of Mosul, Iraq. Email: shaymaa.mohammed@uomosul.edu.iq

ARTICLE INFO

Article history:

Received: 02 /04/2026

Revised form: 24 /04/2026

Accepted : 26 /04/2026

Available online: 30/09/2026

Keywords:

Ordinary Kriging models,
Space filling design,
Latin hypercube,
computer code.

ABSTRACT

Kriging models (KMs) are utilized in several scientific domains as replacements for physical systems. The KM interprets the output from the computer code as a Gaussian process. A well-designed component is crucial in building the KMs. The design and analysis utilized to create (KMs) for computer experiments are the focus of our study. We are interested in assessing the impact of design on the accuracy of KM predictions. KMs are frequently constructed using Latin hypercube designs (LHDs). We examine and evaluate a variety of LHD types. In this study, we aim to construct several KMs based on various LHD kinds, and we compared each KMs performance. Several measures that assess the KM performance are used to make the comparison. The constructed KMs are applied to a real computer code represented by the Piston Simulation function.

MSC..

<https://doi.org/10.29304/jqcm.2026.18.32778>

1. Introduction

In recent years, physical experiments that are deemed too costly, impractical, or unfeasible have been replaced more often with computer experiments. To perform a computer experiment (CE), a computer code (CC) must be created. A CE using CC software requires many input values. These CEs typically demand many input points and a significant amount of execution time. Kriging models (KMs) are essential in this type of experiment because they save time, effort, and costs, as they reduce computational expenses and generate a large number of results quickly. The KMs should quantify the uncertainty of the computer code as they are often fitted using a small number of data points. The GP model would benefit from interpolating the observations in CEs with predictable outcomes.

The KM has been extensively employed for characterizing measurements from computational experiments in numerous research disciplines. [1] introduced a KM as a proxy model as a replacement of a CC. Various academic domains have implemented KMs for representing observations from simulations. Some recent studies are; in engineering, [2] attained numerical estimations by a KM as a surrogate of a CC for power cylinders. [3] devised a KM

*Corresponding author: Zainab Abdulateef Rasheed

Email addresses: zainab.abdulateef@uomosul.edu.iq

Communicated by 'sub editor'

in order to evaluate wind turbines. In the climate field, computationally generated observations were derived by [4] on soil moisture using a CC and a KM. In CEs, [5] contrasted two distinct approaches for constructing KM. [6] introduced multiple approaches for KM verification. [7] outlined a non-decomposable variance-covariance structure for producing multi-response KM for multiple-response CC. [8] recommended multiple criteria for verifying multivariate KM.

Each independent simulation instance of a numerical modeling procedure may necessitate multiple hours or considerably more. As a result, it might not be feasible to construct CEs directly using a CC. Thus, choosing an appropriate selection of input points is essential when making inferences based on a CC. In the literature, this rapidly developing field is referred to as design for CEs. Space-filling is becoming more common in CEs. Space-filling designs encompass a variety of types, such as Latin hypercube designs (LHD) and random sequence based designs. The majority of these configurations possess homogeneous coverage characteristics, which reveal that the samples are distributed evenly across the entire input space. The use of LHDs in creating KMs for CEs has grown significantly in recent years. Thus, a variety of LHDs have been proposed. The purpose of this work is to study the behavior of several LHD types. We first construct numerous KMs, each based on a certain kind of LHD, and then we use some suggested measures to compare the KMs predictions.

This paper is organised as follows. In Section 2, we present the process of building KMs as replacements for CCs. In Section 3, the effect of design for CEs is discussed, and several LHD types are presented. In Section 4, we propose some measures for validating KMs that are based on the KM predictions. Section 5 presents many KMs that are constructed based on several LHDs and conducted to a real CC. In Section 6, the conclusion is given.

2. Ordinary Kriging Models

The KM can be used as a surrogate of the CC to obtain CC predictions.

$$\hat{y} = \beta + z(\mathbf{x}) \quad (1)$$

where β are the unknown parameters and $z(\mathbf{x})$ has a Gaussian process with zero mean and the unknown variance σ^2

$$z(\mathbf{x}) \sim GP(0, \sigma^2 C(\mathbf{x}_i - \mathbf{x}_j)) \quad (2)$$

For the correlation function, the Matérn function is used in our study:

$$C(\mathbf{x}_i - \mathbf{x}_j) = \prod_{i=1}^p \left(\frac{1 + \sqrt{5}|x_i - x_j|}{\psi_i} + \frac{5|x_i - x_j|^2}{3\psi_i^2} \right) \exp\left(-\frac{\sqrt{5}|x_i - x_j|}{\psi_i}\right) \quad (3)$$

where $\psi_i \geq 0$ is the correlation parameters. Suppose $y = \eta(\mathbf{x})$ is the output at the input $\mathbf{x}$. For n inputs, $\mathbf{X} = (\mathbf{x}_1, \dots, \mathbf{x}_n)$, the outputs are $\mathbf{y} = (y_1 = \eta(\mathbf{x}_1), \dots, y_n = \eta(\mathbf{x}_n))$. For a new input, $\mathbf{x}_0$, the linear predictor is given by

$$\hat{y}(\mathbf{x}_0) = \lambda^T(\mathbf{x}_0)\mathbf{y} \quad (4)$$

The mean squared error (MSE) of the CC prediction is

$$\text{MSE}(\hat{y}(\mathbf{x}_0)) = E[\lambda^T(\mathbf{x}_0)\mathbf{y} - y(\mathbf{x}_0)]^2 \quad (5)$$

where

$$y(\mathbf{x}_0) = \eta(\mathbf{x}_0) \quad (6)$$

The unbiased best linear predictor can be obtained by minimizing $\text{MSE}(\hat{y}(\mathbf{x}_0))$. Therefore, the prediction of the new outputs will have a multivariate normal given by

$$\hat{\mathbf{y}}(\mathbf{x}^*) = \hat{\beta}\mathbf{1} + \mathbf{1}_n^T \mathbf{C}^{-1}(\mathbf{y} - \hat{\beta}\mathbf{1}) \quad (7)$$

$$\sigma_{y^*}^2 = \hat{\sigma}^2[1 - \mathbf{c}^T \mathbf{C}^{-1} \mathbf{c} + \mathbf{u}^T (\mathbf{1}_n^T \mathbf{R}^{-1} \mathbf{1}_n)^{-1} \mathbf{u}] \quad (8)$$

where

$$\hat{\beta} = (\mathbf{1}^T \mathbf{R}^{-1} \mathbf{1})^{-1} \mathbf{1}^T \mathbf{C}^{-1} \mathbf{y} \quad (9)$$

$$\mathbf{c} = \{C(\mathbf{x}^*, \mathbf{x}_1), \dots, C(\mathbf{x}^*, \mathbf{x}_n)\}^T$$

$$\mathbf{u} = \mathbf{1}^T \mathbf{R}^{-1} \mathbf{c} - \mathbf{y}$$

where, $\mathbf{C}$ is the correlation matrix for $\mathbf{x} = (\mathbf{x}_1, \dots, \mathbf{x}_n)$.

3. Efficient designs for Building KMs

The primary aim of KM construction is to obtain CC predictions for unknown inputs. Thus, in a CE design selection, the source of sound optimization criteria should be based on prediction error. A good design is expected to generate dense points in every variable projection and also in the complete design space. LHDs are built using projections that are evenly spaced over each input's points. Unfortunately, these designs might not fit the hypercube entirely. To assist in selecting a good design from a class of LHDs, several optimality criteria are provided.

Since the CC is deterministic, replications at a particular input set should be avoided. LHDs are therefore useful since they accommodate accurate KMs. The prediction error is found by comparing untested point positions that are observed in the design space. Thus, KM predictions will be less accurate when the untested positions fall in a sub-space with a few points. Many concepts have recently been suggested as ways to improve LHDs. The LHD and a few designs based on it are reviewed in the ensuing subsections.

3.1. Latin Hypercube Design

The [9] proposed the Latin Hypercube Design (LHD) as a design that and the observations are apportioned evenly across the entire space with all the variables. For generating n points $\mathbf{x}_1, \dots, \mathbf{x}_n$ where $\mathbf{x}_i = x_{i1}, \dots, x_{ip}$, suppose the columns of the $n \times p$ matrix are random of $\{1, \dots, n\}$. Permutations. Suppose also that U_{jk} is a matrix whose elements are uniformly distributed on $(0,1)$. LHDs are therefore generated by

$$X_{jk} = \frac{\Lambda_{ik} - U_{jk}}{n} \quad (10)$$

3.2. Improved Latin Hypercube Design

Improved Latin Hypercube Design (ILHD) was proposed by [10] to be a design that intends to use particular mathematical techniques to improve the distribution of points within the hypercube in a multi-dimensional region. The ILHD algorithm is based on the selection of the most suitable samples, one after another, to give the most optimal set. The reason is that the decisions are not independent and that the decision made influences subsequent decisions. The aforementioned premise is not totally accurate when the set is limited.

The concept of equal volume representation of the points is advanced by ILHD by letting the representation of the points be projected on the hypercube; the coefficient of variance of the minimum distance was considered to be well-distributed. As the constraint of the hypercube applied to the edges and surfaces would reduce the uncertainty, the advantage would be increased provided that the same can be applied to the volume of the hypercube. As the LHD does not take into account hypercube surface distributions, the ILHD does not take into account hypercube volume distributions, either. Consequently, ILHD will maintain the intrinsic stochasticity of the system for error estimates while maintaining a much more ordered sample. The optimal value is calculated based on

$$d_{opt} = \frac{n}{m \sqrt[n]{n}} \quad (11)$$

The m th root of the volume per point is used. In order to reduce variation and increase the accuracy of results in a variety of applications, the ILHD seeks to guarantee the observations across all space of the hypercube.

3.3. Maximin Latin Hypercube Design

Maximin Latin Hypercube Design (MLHD) depends on measuring the distance between the design observations, and it was suggested by [11]. For a given design $\mathbf{X}$, and a metric d , the MLHD can be obtained by calculating the shortest distance between design points [12].

$$X = \max_{\mathbf{x} \in \Omega} \min_{(x, x') \in \mathbf{x}} d(x_i, x_j) \quad (12)$$

3.4. Genetic Latin Hypercube Design

Genetic Latin Hypercube Design (GLHD) was proposed by [13] to work on a variety of problems in optimization. The GLHD can be obtained as follows [13] and [14]:

1. Create initial set of points, a random LHD population of size N .
2. Optimal $\frac{N}{2}$ LHD are chosen to be "survivors," while the rest are ignored.
3. In this stage, N of the $N/2$ are produced. First, numbers 1 and $\frac{N}{2} + 1$ are given the best LH out of all the children. Next, the optimal LH is matched by the k th ($k \in [2, \frac{N}{2}]$). A pair of offspring will be produced by the best and the k th. The 1st child is found by having the optimal LH. The number k is set to child who outperforms the rest. Take k th LH and replace an arbitrary column by the associated column in optimal LH to obtain a second child. The resultant child is set $\frac{N}{2} + k$.
4. With the exception of the first child, best LH, all children of the former generation are mutated. Two randomized items in either column are swapped in case the randomized column is under a given p .

4. Measures for Validating Ordinary KMs

The KMs are constructed based on several assumptions. Therefore, it is necessary to check the KM. We use here two different methods. The 1st measure is the relative absolute error (RAE) given by

$$\text{RAE} = \frac{\sum_{i=1}^n (\hat{y}_i - y_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \quad (13)$$

where $\bar{y}$ is the mean of $\mathbf{y}$ [15]. The RAE displays the discrepancy between the true and predicted CC values. The predictions will be more accurate if the RAE approaches to zero. The individual standardized errors (ISE) is the 2nd measure that is obtained by:

$$\text{ISE} = \frac{y(x_0) - \hat{y}(x_0)}{\sqrt{\sigma_{y^*}^2}} \quad (14)$$

where $\sigma_{y^*}^2$ is the root of the predictive variance equation (8). If the GPR is valid, the ISE values are expected to be between -2 and 2. As a result, if extreme values of ISE are observed, the GPR predictions will be inaccurate, and the GPR will be invalid. The ISE is not determined only by the differences between observations of the computer model and GPR predictions but also by the uncertainty about GPR predictions. This is due to the square root of the GPR variance being used to standardize the variances [16].

5. Illustrative Example

We employ in this section the Piston Simulation (PS) function as an application of a computer model. [17] created the Piston Simulation function to mimic piston action inside a cylinder. The piston's cycle time, expressed in seconds, is the output C . It is provided by:

$$C(\mathbf{x}) = 2\pi \sqrt{\frac{M}{k + S^2 \frac{P_0 V_0 T_a}{T_0 V^2}}} \quad (15)$$

where

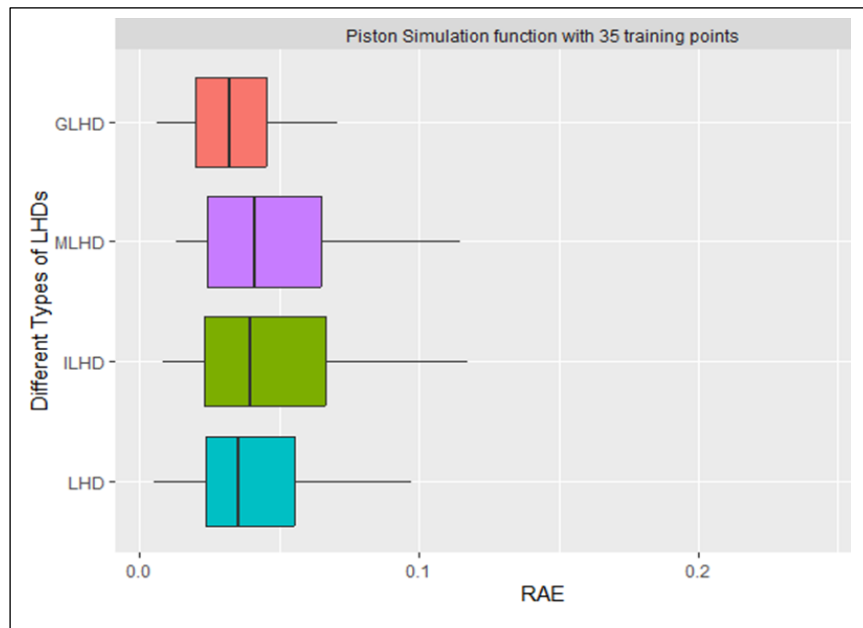
$$V = \frac{S}{2k} \left(\sqrt{A^2 + 4k \frac{P_0 V_0}{T_0} T_a} - A \right)$$

$$A = P_0 S + 19.62M - \frac{kV_0}{S}$$

The PS function has 7 input variables, which are as follows $\mathbf{x} = (M, S, V_0, k, P_0, T_a, T_0)$ where $M \in [30, 60]$ is piston weight (kg); $S \in [0.005, 0.020]$ is piston surface area (m^2); $V_0 \in [0.002, 0.010]$ is initial

gas volume (m^3); $k \in [1000, 5000]$ is spring coefficient (N/m); $P_0 \in [90000, 110000]$ is atmospheric pressure (N/m^2); $T_a \in [290, 296]$ is ambient temperature (K); $T_0 \in [340, 360]$ is filling gas temperature (K).

We constructed different KMs for different LHD types. We first generating $n = 5$ $p=35$ training points for each LHD type. Then, the PS outputs were obtained. We use $m = 2$ $p=14$ for the validation points. Each of the combinations of the training and validation points was duplicated 1000 times. Then, the RAE was calculated, equation (13). Box-plots RAE of KMs that are constructed with various LHDs are presented in Fig(1).



Fig(1): Box-plots of RAE for KMs on PS with $n = 35$.

Fig(1) shows that the RAE values are low in KMs. Thus, all LHDs appear to be accurate in terms of their performance. Nevertheless, the RAE values of KM, which are based on the GLHD, are the lowest.

Therefore, to obtain the optimal LHD behavior, the mean of the RAE values of all LHDs is presented in Table 1.

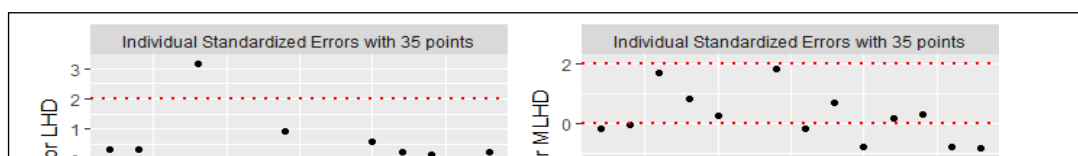
Table 1: The mean of RAE values with $n = 35$

	LHD	ILHD	MLHD	GLHD
RAE	0.0474	0.0558	0.0553	0.04245

The value in bold font is the best.

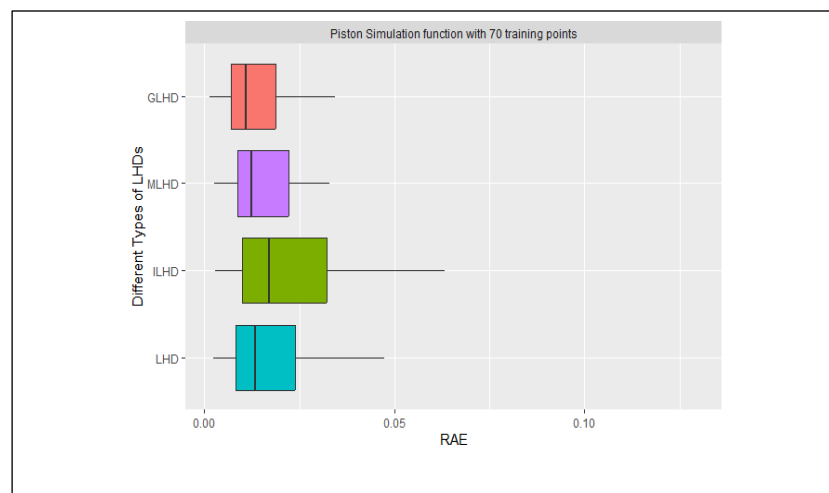
Table 1 shows that the GLHD is the one with lowest value of RAE mean.

We also calculated the ISE equation (14). Fig(2) shows plots of the ISE of KMs. We can see from Fig(2) that the ISE are small for all KMs for the LHDs. However, there is one error that lies outside the ± 2 interval for the KM that is based on the MLHD and two errors lie outside the ± 2 interval for the KM that is based on the LHD.



Fig(2): The ISE plots of KMs of PS that constructed using the various kinds of LHDs with 35 training points as well as 14 test points.

We made the training points to be $n = 10p = 70$. The 1000 RAE values of KMs built with various types of LHDs with varying $n = 10p = 70$ points and a varying $m = 2p = 14$ test points are box-plotted in Fig(3).



Fig(3): Box-plots of 1000 RAE for KMs on PS with $n = 70$ points.

We notice that RAE for KMs are very small for all LHDs, which indicates that the behavior of all KMs has been improved. RAE values for KM, which are also based on the GLHD, are the smallest. The mean of RAE for KMs is shown in Table 2.

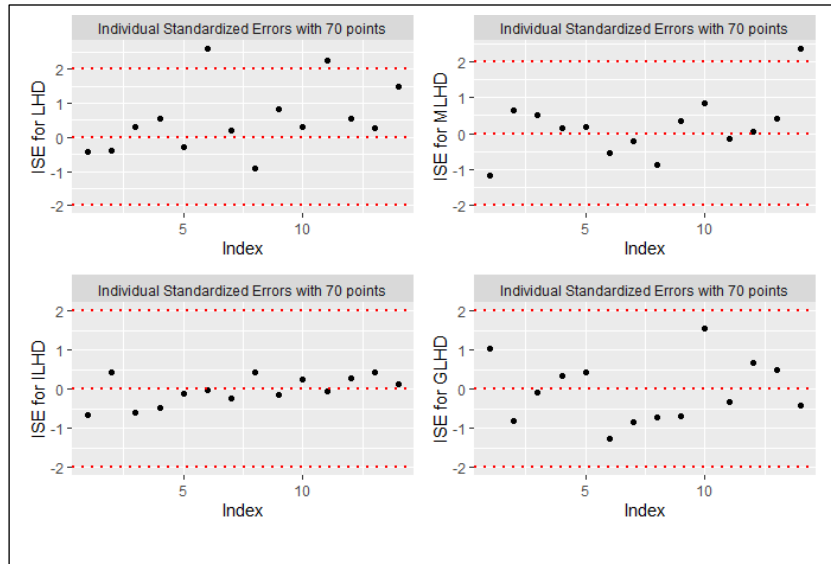
Table 2: The mean of RAE values with $n = 70$

	LHD	ILHD	MLHD	GLHD
RAE	0.0151	0.0199	0.0155	0.0134

The value in bold font is the best.

We notice the GLHD has also the lowest RAE mean.

We also calculated ISE values in equation (14). Fig(4) shows plots of the ISE of KMs.



Fig(4): The ISE plots of KMs of PS that constructed using the various kinds of LHDs with 70 training points as well as 14 test points.

We notice that ISE errors are also small for all KMs. However, there is also one error that lies outside the ± 2 interval for the KM that is based on the MLHD, and two errors lie outside the ± 2 interval for the KM that is based on the LHD.

We find out that the RAE values of the KMs of any type of LHDs have been reduced by the introduction of additions of training points. Moreover, according to the values of RAE, the behavior of all types of KMs of LHDs is very similar, though with a slight outperformance to GLHD. Conversely, the ISE values of KMs of any form of LHDs are small. In addition, all KMs of all LHDs also have relatively same performance, albeit a slight outperforming to ILHD and GLHD. This is perhaps because the ILHD and GLHD objectives to minimize the variance, which leads to smaller ISE values as the uncertainty of the KMs predictions are used by the ISE.

6. Conclusion

We investigated ordinary KMs in this work, which are popular methods for assessing and estimating complicated computer codes. When building ordinary Kriging models, CE design is essential since it can impact prediction accuracy.

For constructing ordinary Kriging models, Latin hypercube designs have gained popularity. Therefore, by comparing the predictions of numerous ordinary KMs that are constructed by various LHD types, we assessed the performance of various Latin hypercube design types in this work. Some measures that assess the KM performance are used to make this comparison. We use ordinary Kriging models on an actual case. The behavior of various kinds of LHDs is rather similar, according to the relative absolute error findings of using ordinary Kriging models on an actual case. The performance of the Genetic Latin hypercube design was marginally superior. This is because the GLHD optimizes for space-filling properties and orthogonality simultaneously. The GLHD uses genetic algorithms to maximize the distance between design points to ensure no regions of the design space are left unexplored. Moreover, GLHD minimizes correlations between design points to prevent the points from collapsing onto each other when projected. The results of ISE show that the ILHD and the GLHD are slightly outperformed by other LHDs as they reduce the variation in of ordinary KM predictions.

References

- [1] J. Sacks, W. J. Welch, T. J. Mitchell, and H. P. Wynn, "Design and analysis of computer experiments," *Statistical science*, vol. 4, no. 4, pp. 409–423, 1989.
- [2] R. Li and A. Sudjianto, "Analysis of computer experiments using penalized likelihood in Gaussian Kriging models," *Technometrics*, vol. 47, no. 2, pp. 111–120, 2005.
- [3] I. Abdallah, C. Lataniotis, and B. Sudret, "Parametric hierarchical kriging for multi-fidelity aero-servo-elastic simulators—Application to extreme loads on wind turbines," *Probabilistic Engineering Mechanics*, vol. 55, pp. 67–77, 2019.
- [4] J. P. Fentanes et al., "Quasi-regression," *Journal of complexity*, vol. 17, no. 4, pp. 588–607, 2001.
- [5] Y. H. Al-Taweel and N. Sadeek, "A comparison of different methods for building Bayesian kriging models," *Pakistan Journal of Statistics and Operation Research*, pp. 73–82, 2020.
- [6] R. W. Al-Naser and Y. Al-Taweel, "Using Kriging models for approximating computer models and quantifying their uncertainty," in *AIP Conference Proceedings*, AIP Publishing, 2023.
- [7] J. P. Kleijnen, "Verification and validation of simulation models," *European journal of operational research*, vol. 82, no. 1, pp. 145–162, 1995.
- [8] Y. Al-Taweel, "Uncertainty quantification of multivariate Gaussian process regression for approximating multivariate computer codes," *TWMS Journal of Applied and Engineering Mathematics*, 2024.
- [9] M. D. McKay, R. J. Beckman, and W. J. Conover, "A comparison of three methods for selecting values of input variables in the analysis of output from a computer code," *Technometrics*, vol. 42, no. 1, pp. 55–61, 2000.
- [10] B. Beachkofski and R. Grandhi, "Improved distributed hypercube sampling," in *43rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, 2002, p. 1274.
- [11] M. D. Morris and T. J. Mitchell, "Exploratory designs for computational experiments," *Journal of statistical planning and inference*, vol. 43, no. 3, pp. 381–402, 1995.
- [12] K.-T. Fang, R. Li, and A. Sudjianto, *Design and modeling for computer experiments*. CRC Press, 2010.
- [13] T. W. Simpson, *A concept exploration method for product family design*. Georgia Institute of Technology, 1998.
- [14] M. Liefvendahl and R. Stocki, "A study on algorithms for optimization of Latin hypercubes," *Journal of statistical planning and inference*, vol. 136, no. 9, pp. 3231–3247, 2006.
- [15] A. M. Overstall and D. C. Woods, "Multivariate emulation of computer simulators: model selection and diagnostics with application to a humanitarian relief model," *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 2016.
- [16] L. S. Bastos and A. O'Hagan, "Diagnostics for Gaussian process emulators," *Technometrics*, vol. 51, no. 4, pp. 425–438, 2009.
- [17] R. Kenett and S. Zacks, *Modern Industrial Statistics: Design and Control of Quality and Reliability*. Duxbury Press, 1998. [Online]. Available: <https://books.google.co.uk/books?id=0zcfAQAAIAAJ>.