Designing Computer Experiments with Multiple Types of Factors: The MaxPro Approach

V. Roshan Joseph

H. Milton Stewart School of Industrial and Systems Engineering,

Georgia Institute of Technology, Atlanta, GA 30332-0205

Evren Gul

Loyalty One, Chicago, IL

Shan Ba

Data & Modeling Sciences,

The Procter & Gamble Company, Cincinnati, OH 45040

Abstract

Computer experiments may involve not only continuous input factors but also nominal factors, discrete numeric factors, and ordinal factors. Most existing literature in designing computer experiments focus only on continuous factors. Some works have further considered nominal factors, but the cases that also contain discrete numeric or ordinal factors are almost overlooked. In this work, we propose a new optimal design criterion that can accommodate all these types of factors. The proposed design is flexible in run size and number of factors, and can also achieve good space-filling properties in the full design space and in all possible low-dimensional projections.

Keywords: Space-filling designs, Quantitative factors, Qualitative factors, Latin hypercube designs.

Introduction

Modern computing technology enables the use of computers to simulate sophisticated physical systems with high fidelity, which are widely used for scientific discovery, technology innovation, and quality improvement. Although much cheaper than experimenting with a real physical system, each run of a computer simulation can take hours or days to complete. Due to this high computational cost, it is often necessary to employ experimental design techniques to strategically choose the input factor settings for running the simulation so that maximum information can be learned from the system with minimum number of runs (Sacks, Welch, Mitchell and Wynn 1989, Santner, Williams and Notz 2003).

Computer experiments can involve both qualitative and quantitative input factors. The qualitative factors can be nominal or ordinal and the quantitative factors can be continuous or discrete numeric. As an example consider a solid end milling process, which can be simulated using the Production Module software of Third Wave Systems. The software accepts several parameter specifications such as the cutting tool parameters shown in Figure 1. These parameters: rake angle, relief angle, helix angle, and corner radius are quantitative factors which are continuous. On the other hand, we can also specify the number of flutes for the end mill cutting tool, which can take values 2, 3, or 4. This is a discrete numeric-type quantitative factor. We can also specify the type of workpiece material such as high-speed steel or Titanium alloy, which is a nominal-type qualitative factors. Although not currently available in this particular software, one can also imagine other types of factors such as the condition of the tool rated as poor, fair, or good. This is a qualitative factor, but is of ordinal-type.

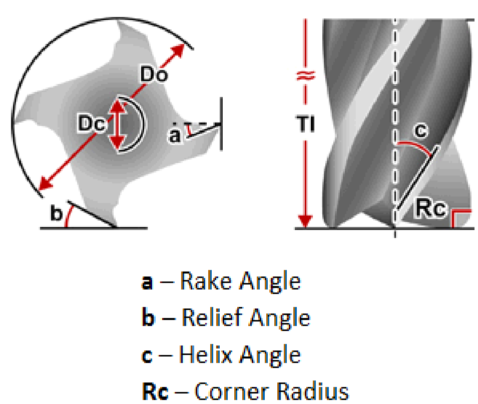


Figure 1: Solid end milling cutting tool parameters

Most existing literature on designing computer experiments focus only on continuous input factors, and some recent works have further considered the case of nominal factors. When the computer experiments also involve discrete numeric and ordinal factors, however, the optimal design strategy remains largely unresolved. Practitioners often have to ignore the ordinal information and treat them as nominal factors. Sometimes the discrete numeric factors are also handled by either treating them as nominal factors, or as continuous factors and then rounding to the nearest integer values, which can lead to sub-optimal designs. In this work, we propose an optimal design criterion that can distinguish and incorporate all these different types of input factors and produce much better designs.

Existing Works and Limitations

The majority of literature in computer experiments assume all input factors to be continuous. *Space-filling designs*, which spreads out design points evenly throughout the input space, are widely used for such factors. See Joseph (2016) for a recent review of space-filling designs. Let $\boldsymbol{D} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$ denote an n-run experimental design, where each point $\boldsymbol{x}_i \in \mathcal{X} = [0, 1]^p$. A popular space-filling criterion is the *maximin distance criterion* (Johnson, Moore and Ylvisaker 1990), which tries to maximize the minimum inter-point distance between design points:

$$\max_{\boldsymbol{D}} \min_{\boldsymbol{x}_i, \boldsymbol{x}_j \in \boldsymbol{D}} d(\boldsymbol{x}_i, \boldsymbol{x}_j), \tag{1}$$

where $d(\boldsymbol{x}_i, \boldsymbol{x}_j)$ is the Euclidean distance between points $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$.

Among the many input factors in a computer experiment, usually only a few of them are important. This is often called the *effect sparsity principle* in the literature (Wu and Hamada 2009). Since computer experiment outputs are deterministic, any replicated design points would lead to wastage of computational resources. As a result, a good design for computer experiments also needs to be *non-collapsing*, which requires the projections of design points onto any lower-dimensional subspace to be non-overlapping. The most popular non-collapsing design in computer experiments is the *Latin hypercube design* (LHD) (McKay, Beckman and Conover 1979), whose projections onto each input dimension always have n distinct levels. However, non-collapsing designs are not necessarily space-filling. To improve the space-filling property of a randomly generated LHD, Morris and Mitchell (1995) proposed to construct the maximin

LHD (Mm LHD), which is obtained by minimizing

$$\phi_k(\boldsymbol{D}) = \left\{ \frac{1}{\binom{n}{2}} \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \frac{1}{d^k(\boldsymbol{x}_i, \boldsymbol{x}_j)} \right\}^{1/k}, \tag{2}$$

where k is nonnegative and usually a large value.

The maximin LHD ensures space-fillingness in the full dimensional space and also uniformity in all one-dimensional projections. However, its projection properties for subspaces with dimensions $2, \dots, p-1$ are not optimized. To remedy this problem, Joseph, Gul and Ba (2015) proposed the following *maximum projection* (MaxPro) criterion:

$$\min_{\boldsymbol{D}} \quad \psi(\boldsymbol{D}) = \left\{ \frac{1}{\binom{n}{2}} \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \frac{1}{\prod_{l=1}^{p} (x_{il} - x_{jl})^2} \right\}^{1/p}, \tag{3}$$

which can simultaneously optimize the space-filling properties of the design points with respect to all possible subsets of factors. Whenever any two points have the same coordinate in a single dimension, (3) would become ∞, and thus the MaxPro criterion automatically justifies the necessity of having n distinct level for each factor of the design, as required by the LHD structure.

When the computer experiments also involve nominal input factors, Qian (2012) proposed the *Sliced Latin hypercube design* (SLHD). Suppose t is the number of all possible combinations of levels of the nominal factors, then an n-run SLHD is defined as a special type of LHD that can be partitioned into t slices with each slice being an LHD of size n/t runs which is used as the design for continuous factors corresponding to each level combination of the nominal factors. Obviously, as the number of nominal factors increases, the value of t would increase exponentially. As a result, the SLHD can only accommodate a small number of nominal factors. Recently, Deng, Hung and Lin (2015) proposed the *marginally coupled design* (MCD) which can accommodate a larger number of nominal factors. The MCD structure combines an orthogonal array (OA) (see, e.g., Wu and Hamada 2009) for nominal factors with an LHD for continuous factors in such a way that the LHD can be partitioned by the levels of any single nominal factor into multiple smaller LHDs. In other words, the LHD for continuous factors combined with any single nominal factor column in an MCD forms an SLHD. Because the MCD uses an OA instead of a full factorial combination for the nominal factors, it can achieve more economic run size than the SLHD.

Nevertheless, the existing condition of an MCD is restrictive : it requires not only the existence of an OA of a given size, but also the existence of a special LHD which can form an SLHD when paired with each single column of the OA.

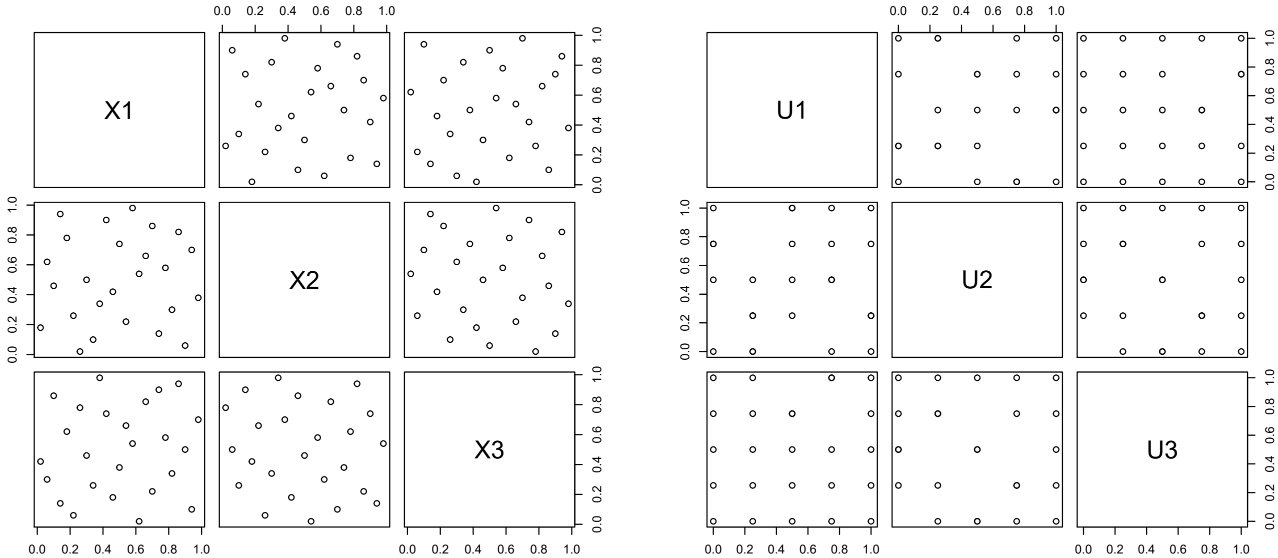
The SLHD and MCD structures can guarantee maximum uniformity in any one-dimensional projection. However, they are not optimized for the overall space-filling properties or for the projection properties in more than one dimensional subspace. Ba, Brenneman and Myers (2015) proposed to generate an optimal SLHD by modifying the maximin criterion in (2) as follows:

$$\min_{\boldsymbol{D}} \quad \phi_{Mm}(\boldsymbol{D}) = \frac{1}{2}\left\{\phi_r(\boldsymbol{D}) + \frac{1}{t}\sum_{i=1}^{t}\phi_r(\boldsymbol{D}_i)\right\}, \tag{4}$$

where $\boldsymbol{D}_i$ represents the ith slice of $\boldsymbol{D}$. Although it is overall space-filling, the optimal SLHD still cannot accommodate a large number of nominal factors and does not guarantee good space-filling properties when projected onto lower dimensional subspaces other than the single dimensions. Improvements to MCD are proposed in He, Lin, and Sun (2017) and He, Lin, Sun, and Lv (2017), but the flexibility of run size and number of factors is still a concern. In the next section, we will present a new optimal design criterion that can overcome these problems.

How to efficiently design computer experiments that contain not only continuous and nominal factors but also discrete numeric and ordinal factors is a challenging problem. The LHD structure cannot accommodate discrete numeric factors since it requires each factor to have n distinct levels. Using an SLHD to accommodate discrete numeric factors would ignore the quantitative distance between their factor levels. As discussed earlier, when engineers encounter discrete numeric factors in practice, they often apply a trick by first treating them as continuous factors to generate a space-filling LHD and then collapsing their levels to the nearest discrete numeric levels. This trick may not work well always. For example, Figure (2a) illustrates a 25-run space-filling LHD with three continuous factors. The design after collapsing the 25 levels in each of its columns to five equally spaced levels is shown in Figure (2b). We can see that it creates unnecessary gaps in the design space. Thus, better methods are needed to incorporate discrete numeric as well as ordinal factors in the experimental design.

Even in the realm of physical experiments, optimal designs with multiple types of factors is rarely



(a) (b)

Figure 2: Illustration of the level-collapsing approach: (a) A 25-level space-filling LHD with three continuous factors, and (b) Collapsing the space-filling design from (a) into a five-level design.

studied. Most existing works in physical experimental designs uses combinatorial constructions which are suitable only for nominal-type factors. Cheng and Ye (2004) tried to extend factorial designs to include continuous factors, but their method cannot handle a mix of continuous and nominal factors. Joseph, Ai, and Wu (2009) have considered a mix of continuous and nominal factors, but only for the case of two and four-level fractional factorial designs. On the other hand, most response surface designs deal with only continuous factors. A few have considered the case of nominal factors (Draper and John 1988, Wu and Ding 1998), but their construction methods are limited in terms of number of factors and levels. Moreover, we have not seen any works that considers ordinal and discrete numeric factors and a mix of all of these different types of factors. Thus, although here we focus on computer experimental designs, there is a potential to extend the proposed ideas to physical experimentation.

Maximum Projection Criterion for Multiple Types of Factors

In this section, we will develop a new criterion for designing computer experiments, which extends the MaxPro criterion in (3) to accommodate continuous, nominal, discrete numeric, and ordinal types of factors. First we convert the ordinal factors into discrete numeric factors through the well-known *scoring* *method* (see, e.g., Wu and Hamada 2009). That is, if the ordinal factor has levels, "poor", "fair" and "good", then depending on the nature of the classification, the experimenter can choose some discrete numeric levels such as 1, 4, and 5 to represent the three ordinal levels. Thus, for the rest of the paper, we will only consider continuous, nominal, and discrete numeric factors assuming that the ordinal factors are already converted to discrete numeric factors through scoring.

MaxPro criterion has an optimality connection with Gaussian process (GP) modeling (Joseph, Gul, and Ba 2015). We exploit this connection to extend the criterion to include continuous, nominal, and discrete numeric factors. First consider the case of continuous factors. Let there are p_1 of them. The GP model is defined as $Y(\boldsymbol{x}) = \mu + Z(\boldsymbol{x})$, where μ is the overall mean and $Z(\boldsymbol{x})$ is a Gaussian process with mean zero and covariance function $\sigma^2 R(\cdot)$ (Santner, Williams and Notz 2003). A popular choice for $R(\cdot)$ is the Gaussian correlation function given by

$$R(\boldsymbol{x}_i - \boldsymbol{x}_j; \boldsymbol{\alpha}) = \exp\{-\sum_{l=1}^{p_1} \alpha_l (x_{il} - x_{jl})^2\}, \tag{5}$$

where $\alpha_l \in (0, \infty)$ for $l = 1, \ldots, p_1$ are the correlation parameters. Suppose $\boldsymbol{R}(\boldsymbol{\alpha})$ is the $n \times n$ correlation matrix whose (i, j)th element is $R(\boldsymbol{x}_i - \boldsymbol{x}_j; \boldsymbol{\alpha})$. Assuming a noninformative prior distribution for the correlation parameters, Joseph, Gul and Ba (2015) showed that the MaxPro design minimizes the expected sum of off-diagonal elements of the correlation matrix given by $\mathbb{E}\{\sum_{i=1}^{n} \sum_{j \neq i} \boldsymbol{R}_{ij}(\boldsymbol{\alpha})\}$. Minimizing the off-diagonal elements of the correlation matrix tends to increase the determinant of the correlation matrix. Thus, a MaxPro design is expected to perform well under the maximum entropy criterion (Shewry and Wynn, 1987) as well.

Now consider the case of nominal and discrete numeric factors. Let there are p_2 discrete numeric factors and p_3 nominal factors. The key to incorporate them in the design criterion is to properly define a correlation function for them in the GP model. Qian, Wu and Wu (2008) proposed several correlation construction schemes along with a general framework for building GP models with quantitative and nominal-type qualitative factors. Here we assume the following exchangeable correlation structure for the nominal factor which was also used by Joseph and Delaney (2007):

$$R(\boldsymbol{w}_i - \boldsymbol{w}_j; \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}) = \exp\left\{-\sum_{l=1}^{p_1} \alpha_l |x_{il} - x_{jl}| - \sum_{k=1}^{p_2} \beta_k |u_{ik} - u_{jk}| - \sum_{h=1}^{p_3} \gamma_h I(v_{ih} \neq v_{jh})\right\}, \tag{6}$$

where $\boldsymbol{w}_i = (\boldsymbol{x}_i, \boldsymbol{u}_i, \boldsymbol{v}_i)$, $\boldsymbol{x}$ represent the p_1 continuous factors, $\boldsymbol{u}$ represent the p_2 discrete numeric factors (including ordinal factors), $\boldsymbol{v}$ represent the p_3 nominal factors, and $I(v_{ih} \neq v_{jh})$ is the indicator function that takes values 1 if $v_{ih} \neq v_{jh}$ and 0 otherwise. Assume that the discrete numeric factors are also scaled in $[0, 1]$ so that the correlation parameters of the three types of factors are in the same scale. Different from Joseph, Gul, and Ba (2015), we use an exponential correlation function for the continuous factors instead of a Gaussian correlation function. We will show that using an informative prior on the correlation parameters of the exponential correlation function, we can derive the same MaxPro criterion as with the Gaussian correlation function. This modification using an informative prior is crucial for the discrete numeric factor because they cannot have n levels in the design.

Assume the following priors for the correlation parameters:

$$\begin{aligned}
\alpha_l &\sim^{iid} \quad Gamma(2, \bar{\alpha}_l), l = 1, \dots, p_1, \\
\beta_k &\sim^{iid} \quad Gamma(2, \bar{\beta}_k), k = 1, \dots, p_2, \\
\gamma_h &\sim^{iid} \quad Gamma(2, \bar{\gamma}_h), h = 1, \dots, p_3.
\end{aligned}$$

Then, it is easy to show that:

$$\begin{aligned}
&\mathbb{E}\{\sum_{i=1}^{n} \sum_{j \neq i} \boldsymbol{R}_{ij}(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma})\} \\
= &\int\int\int \sum_{i=1}^{n} \sum_{j \neq i} R(\boldsymbol{w}_i - \boldsymbol{w}_j; \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}) \prod_{l=1}^{p_1} \bar{\alpha}_l^2 \alpha_l e^{-\bar{\alpha}_l \alpha_l} \prod_{k=1}^{p_2} \bar{\beta}_k^2 \beta_k e^{-\bar{\beta}_k \beta_k} \prod_{h=1}^{p_3} \bar{\gamma}_h^2 \gamma_h e^{-\bar{\gamma}_h \gamma_h} \, d\boldsymbol{\alpha} \, d\boldsymbol{\beta} \, d\boldsymbol{\gamma} \\
= &\sum_{i=1}^{n} \sum_{j \neq i} \prod_{l=1}^{p_1} \frac{\bar{\alpha}_l^2}{\{|x_{il} - x_{jl}| + \bar{\alpha}_l\}^2} \prod_{k=1}^{p_2} \frac{\bar{\beta}_k^2}{\{|u_{ik} - u_{jk}| + \bar{\beta}_k\}^2} \prod_{h=1}^{p_3} \frac{\bar{\gamma}_h^2}{\{I(v_{ih} \neq v_{jh}) + \bar{\gamma}_h\}^2}. \qquad (7)
\end{aligned}$$

This gives a new design criterion, which is to minimize

$$\frac{1}{\binom{n}{2}} \sum_{i=1}^{n} \sum_{j \neq i} \frac{1}{\prod_{l=1}^{p_1} \{|x_{il} - x_{jl}| + \bar{\alpha}_l\}^2 \prod_{k=1}^{p_2} \{|u_{ik} - u_{jk}| + \bar{\beta}_k\}^2 \prod_{h=1}^{p_3} \{I(v_{ih} \neq v_{jh}) + \bar{\gamma}_h\}^2}. \qquad (8)$$

It is preferable to choose smaller values for $\bar{\alpha}_l$, $\bar{\beta}_k$ and $\bar{\gamma}_h$ in (8) in order to increase its sensitivity in comparing different designs. For continuous factors, we can simply set $\bar{\alpha}_l = 0$ $(l = 1, \dots, p_1)$ because this would force each continuous factor to have n distinct levels. For nominal and discrete numeric factors, however, the total available number of levels is less than n for each factor. Therefore, we must have $\bar{\beta}_k > 0$ $(k = 1, \ldots, p_2)$ and $\bar{\gamma}_h > 0$ $(h = 1, \ldots, p_3)$ to prevent the denominator in (8) to become zero when $u_{ik} = u_{jk}$ or $v_{ih} = v_{jh}$ for some (i, j) values. Let L_h denote the number of distinct levels of the hth nominal factor, $h = 1, \ldots, p_3$. Because $P\{I(v_{ih} \neq v_{jh}) = 0\} = 1/L_h$, we have $E\{I(v_{ih} \neq v_{jh})\} = 1 - 1/L_h$ for $h = 1, \ldots, p_3$. This suggests choosing $\bar{\gamma}_h = 1/L_h$, which is the smallest possible value of $\bar{\gamma}_h$ to make $E\{I(v_{ih} \neq v_{jh}) + \bar{\gamma}_h\}$ independent of L_h. Similarly, for discrete numeric factors, we can also set $\bar{\beta}_k = 1/m_k$, where m_k is the number of levels of the kth discrete numeric factor. It can be seen that as m_k increases, the corresponding discrete numeric factor behaves more like a continuous factor, and the corresponding $\bar{\beta}_k \to 0$ as desired.

With the foregoing choices of the hyperparameter values, the new optimal design criterion for continuous factors $\boldsymbol{x}_l$ $(l = 1, \ldots, p_1)$, discrete numeric factors (including ordinal factors) $\boldsymbol{u}_k$ $(k = 1, \ldots, p_2)$ and nominal factors $\boldsymbol{v}_h$ $(h = 1, \ldots, p_3)$ can be formally defined as to minimize

$$\psi(\boldsymbol{D}) = \left\{ \frac{1}{\binom{n}{2}} \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \frac{1}{\prod_{l=1}^{p_1} (x_{il} - x_{jl})^2 \prod_{k=1}^{p_2} \{|u_{ik} - u_{jk}| + \frac{1}{m_k}\}^2 \prod_{h=1}^{p_3} \{I(v_{ih} \neq v_{jh}) + \frac{1}{L_h}\}^2} \right\}^{\frac{1}{p_1+p_2+p_3}} . \tag{9}$$

This new criterion contains the MaxPro criterion in (3) as a special case when there are only continuous factors $(p_1 > 0, p_2 = p_3 = 0)$. As a result, it possesses all the desirable properties of a MaxPro design and maximizes the space-filling properties not only in the full design space but also in all possible projections to sub-dimensional spaces. For nominal factors, this criterion also shares similar ideas as the J_2-Optimality criterion proposed by Xu (2002) in constructing mixed-level OAs and near OAs through maximizing the dissimilarity of rows, which is defined as $\min J_2(\boldsymbol{D}) = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \{\sum_{h=1}^{p_3} I(v_{ih} = v_{jh})\}^2$. Furthermore, if we have only one nominal factor with t levels and p continuous factors, then as shown in the Appendix, the criterion in (9) is closely related to the criterion in (4) for finding optimal SLHDs.

Optimal Design Construction

In this section, we will discuss how to generate the optimal designs for computer experiments based on the proposed MaxPro criterion in (9) for continuous, nominal, discrete numeric, and ordinal types of factors.

When the computer experiments only involve continuous factors $(p_1 > 0, p_2 = p_3 = 0)$, (9) reduces to the original MaxPro criterion in (3). Thus, algorithms discussed in Joseph, Gul and Ba (2015) for generating the MaxPro designs can be directly used.

When the computer experiments involve continuous factors and discrete numeric factors (ordinal factors) $(p_1 > 0, p_2 > 0, p_3 = 0)$, we can first randomly initialize a design matrix by generating a $n \times (p_1 + p_2)$ random LHD and collapsing the n levels in each of the last p_2 columns to the nearest given discrete numeric levels. Then, this initial design can be optimized with respect to criterion (9) using a version of the simulated annealing algorithm (Morris and Mitchell 1995), which iteratively searches in the design space in such a way that in each step two randomly chosen elements within a randomly selected column in the design matrix are interchanged. Because the optimization steps only permute the order of the existing levels in each column of the initial design matrix, the final optimal design still guarantees to consist n levels for each of the continuous factors and m_k levels for each of the discrete numeric factors ($k = 1, \ldots, p_2$).

When nominal factors are also present in the computer experiments $(p_1 > 0, p_2 \geq 0, p_3 > 0)$, criterion (9) can similarly be used to optimize the columns of nominal factors from a random initial design. However, directly searching for the optimal design with all three types of factors is challenging. Because vast amounts of literature have already been available in physical experiments for generating optimal designs for nominal factors (e.g., Wu and Hamada 2009), we propose to leverage these existing results and choose the optimal design matrix for nominal factors from the well-studied fractional factorial designs, OAs, near OAs, D-optimal Designs, or I-optimal designs in the literature. This can save considerable amounts of computational time and also give the practitioners the greatest flexibility in selecting the design with the most suitable properties. The criterion (9) is then used to optimize the columns for continuous factors and discrete numeric factors and also to optimize how these columns are joined with the fixed design matrix for the nominal factors.

Our proposed design construction algorithm can be formally stated as follows:

I. Initialization Stage:

Step 1. If $p_1 > 0$, generate a $n \times p_1$ random LHD $\boldsymbol{D}_x$ for continuous factors.

Step 2. If $p_2 > 0$, generate a $n \times p_2$ random LHD $\boldsymbol{D}_u$ and collapse the n levels in each column to the nearest given discrete numeric levels of each factor.

Step 3. If $p_3 > 0$, choose a $n \times p_3$ optimal design for nominal factors $\boldsymbol{D}_v$ from the existing physical experiments' literature or using a commercial statistical software such as JMP.

Step 4. Form the $n \times (p_1 + p_2 + p_3)$ initial design matrix $\boldsymbol{D} = [\boldsymbol{D}_x, \boldsymbol{D}_u, \boldsymbol{D}_v]$, which consists n levels for each of the p_1 continuous factors, m_k levels for the kth discrete numeric factor ($k = 1, \ldots, p_2$), and L_h levels for the hth nominal factor ($h = 1, \ldots, p_3$).

II. Optimization Stage: Iteratively search in the design space to optimize the criterion (9) using a version of the simulated annealing algorithm (Morris and Mitchell 1995).

Step 5. Denote the current design matrix as $\boldsymbol{D} = [\boldsymbol{D}_x, \boldsymbol{D}_u, \boldsymbol{D}_v]$. Randomly choose a column from the $[\boldsymbol{D}_x, \boldsymbol{D}_u]$ components, and interchange two randomly chosen elements within the selected column. Denote the new design matrix as $\boldsymbol{D}_{try}$.

Step 6. If $\boldsymbol{D}_{try} = \boldsymbol{D}$, repeat Step (5).

Step 7. If $\psi(\boldsymbol{D}_{try}) < \psi(\boldsymbol{D})$, replace the current design $\boldsymbol{D}$ with $\boldsymbol{D}_{try}$; otherwise, replace the current design $\boldsymbol{D}$ with $\boldsymbol{D}_{try}$ with probability $\pi = \exp\{-[\psi(\boldsymbol{D}_{try}) - \psi(\boldsymbol{D})]/T\}$, where T is a preset parameter known as "temperature".

Step 8. Repeat Step (5) to Step (7) until some convergence requirements are met. Report the design matrix with the smallest $\psi(\boldsymbol{D})$ value as the optimal design with respect to criterion (9).

In the above algorithm, design for the continuous factors $\boldsymbol{D}_x$ is initialized using a random LHD, and design for the discrete numeric factors $\boldsymbol{D}_u$ is initialized using the level-collapsing method. The design matrix for the nominal factors $\boldsymbol{D}_v$ is chosen from the existing optimal design literature or statistical software and is fixed in the algorithm. The optimization steps permute the order of existing levels in columns of the $\boldsymbol{D}_x$ and $\boldsymbol{D}_u$ matrices as well as the order of their rows when joined with the fixed $\boldsymbol{D}_v$ matrix. The simulated annealing parameters can be set similarly as in Lundy and Mees (1986) for which convergence is already established. Also, similar to the computational shortcut used by Jin, Chen and Sudjianto (2005), an updating formula can be used to compute $\psi(\boldsymbol{D}_{try})$ based on the existing $\psi(\boldsymbol{D})$ value of the preceding design, which avoids re-computing all the summation terms in criterion (9).

In a special case when $p_1 > 0, p_2 = 0, p_3 = 1$, the optimal design found by the proposed algorithm may not have an exact SLHD structure, but our numerical study in the next section shows it can have similar one-dimensional uniformity properties. In addition, the proposed algorithm can generate an optimal design even when n is not a multiple of L_h for which an SLHD does not exist. When $p_1 > 0, p_2 = 0, p_3 > 1$, similar to the MCD structure, the proposed design can also accommodate a large number of nominal factors with economic run size. Nevertheless, the new design does not have the strict run size restrictions as the MCD does and can achieve much superior space-filling properties as we will illustrate in the next section. When all $p_1 > 0, p_2 > 0, p_3 > 0$, the proposed algorithm becomes the only solution so far in the literature to generate optimal designs suitable for computer experiments.

The continuous factors constructed above have an LHD structure with n equally spaced levels. As discussed in Joseph, Gul and Ba (2015), although criterion (9) justifies using n distinct levels for each continuous factor, their levels do not have to be equally spaced. After obtaining the optimal design from the above algorithm, we can improve it further through a local optimization of the continuous factors. The gradient of the objective function in (9) with respect to the continuous factor is

$$\frac{\partial \psi^{(p_1+p_2+p_3)}(\boldsymbol{D})}{\partial x_{rs}} = \frac{2}{\binom{n}{2}} \sum_{i \neq r} \frac{1}{\prod_{l=1}^{p_1}(x_{il}-x_{rl})^2 \prod_{k=1}^{p_2}\{|u_{ik}-u_{rk}| + \frac{1}{m_k}\}^2 \prod_{h=1}^{p_3}\{I(v_{ih} \neq v_{rh}) + \frac{1}{L_h}\}^2} \frac{1}{(x_{is}-x_{rs})}, \tag{10}$$

which can be used to implement a fast derivative-based algorithm to optimize the n levels of the continuous factors.

Numerical Studies

The proposed MaxPro design generated by the algorithm from the previous section imposes no constraint on the structure of the continuous factors and thus the design exists for any run size and number of factors. On the other hand, as discussed before, the MCD has strict existence conditions because it further requires that for each level of any nominal factors, the corresponding design points for the continuous factors form a small LHD. When there is only one nominal factor, the MCD contains SLHD as a special case. In this section, we use numerical examples to show that although the proposed MaxPro design does not impose the restricted MCD structure, through optimizing the criterion (9), it can still approximately

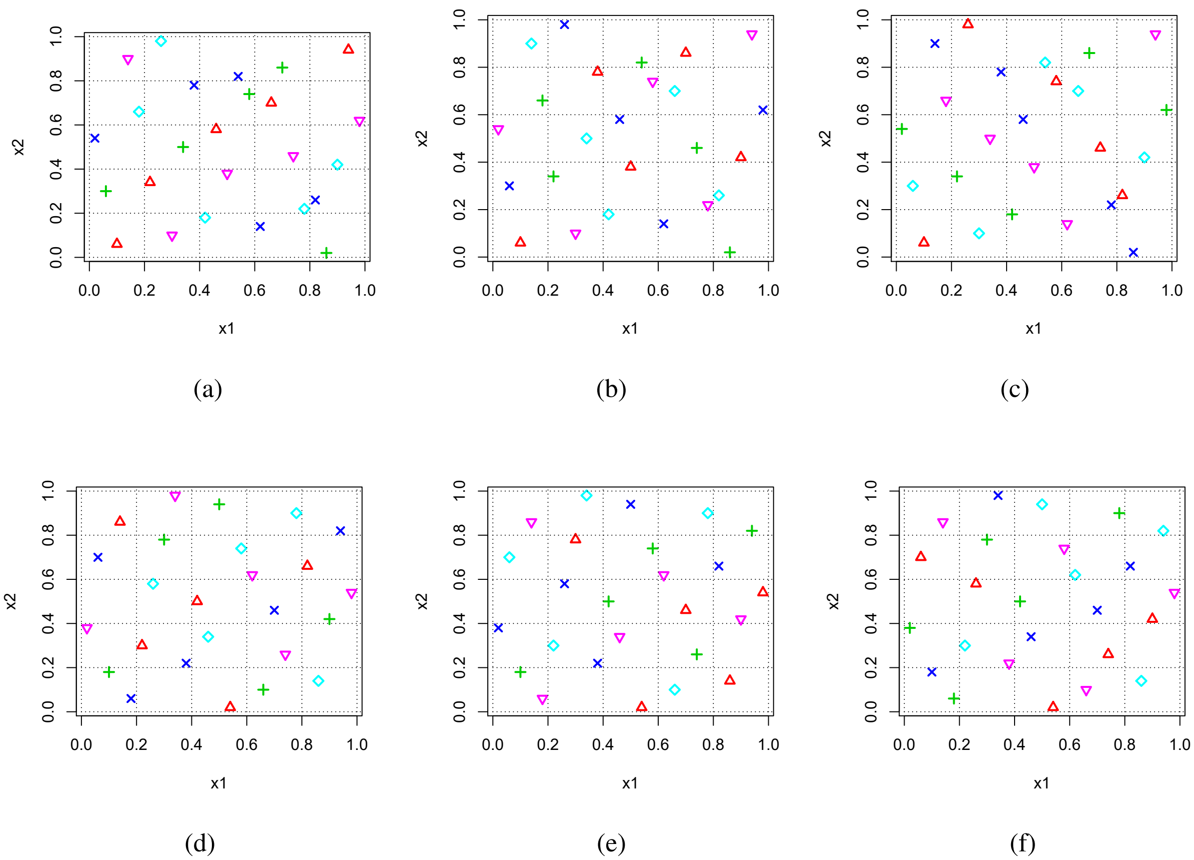


Figure 3: Compare the MaxPro design with MCD in Example 1. (a)-(c): Scatter plots of x_1 versus x_2 of the 25-run MCD. The five types of symbols represent the five different levels of: (a) the first nominal factor; (b) the second nominal factor; (c) the third nominal factor. (d)-(f): Scatter plots of x_1 versus x_2 of the 25-run MaxPro design. The five types of symbols represent the five different levels of: (d) the first nominal factor; (e) the second nominal factor; (f) the third nominal factor.

achieve similar one-dimensional uniformity when the columns for continuous factors are sliced by the levels of any nominal factor. Moreover, we will also show that the proposed MaxPro design can have much better space-filing properties than the MCD.

Example 1: 25-run design with two continuous factors and three nominal factors each with five levels $(p_1 = 2, p_2 = 0, p_3 = 3)$.

To illustrate that the MaxPro design can have similar one-dimensional properties as an MCD, we consider this simple example with only two continuous factors. As discussed in Deng, Hung and Lin (2015), a 25-run OA with six five-level factors can be used to construct the MCD, where three columns from the OA are directly used for the $p_3 = 3$ nominal factors and another two columns from the OA are used to form an 25-run OA-based LHD (Tang 1993) for the $p_1 = 2$ continuous factors. Scatter plots of the two continuous factors (x_1 versus x_2) of the resulting MCD are shown in Figure 3 (a)-(c), where the symbols in each scatter plot are determined by the levels of one of the three nominal factors. Because the original 25-run OA has strength two (every two columns have all 25 possible combinations of levels appearing exactly once), it can be seen in Figure 3 (a), (b) or (c) that only one point appears in each of the 5×5 grids and the projection of the 25 design points onto any dimension have exactly one type of symbol in each of the five equally-spaced bins.

Compared to the MCD, Figure 3 (d)-(f) show a MaxPro design in which the same three columns are used for the three nominal factors but the two continuous factors are generated purely by optimizing the criterion in (9) instead of leveraging other OA columns. Although not using an OA-based LHD structure (exactly one point in each of the 5×5 grids), the MaxPro design points appear to be similarly space-filling in the scatter plot of x_1 versus x_2 (the minimum inter-point distance gets even better than the MCD). In addition, for each level of a nominal factor, projections of the corresponding MaxPro design points onto any single dimension in the scatter plots are also well-spread out. In other words, without imposing any structural constraints on the continuous factors, the proposed MaxPro design optimized by criterion (9) can approximate an MCD in achieving one-dimensional uniformities under each level of a nominal factor. In the next example, we will show that when the dimension of input factors becomes higher, the MaxPro design would perform much better than the MCD, since the MCD is not optimized for space-filling properties higher than two dimensions.

Example 2: 49-run design with five continuous factors and three nominal factors each with seven levels $(p_1 = 5, p_2 = 0, p_3 = 3)$.

In this example, an MCD can be constructed based on a 49-run strength-two OA with eight seven-level factors. The first five columns in the OA can generate a 49-run OA-based LHD through random level expansions (Tang 1993) and used for the $p_1 = 5$ continuous factors of the MCD, and the last three columns in the OA are directly used for the $p_3 = 3$ nominal factors. Keeping the same three columns for the nominal factors, a MaxPro design can be created through the proposed algorithm which generates five new columns for the continuous factors. To demonstrate the effectiveness of criterion (9), we also combine the same three columns for the nominal factors with a random LHD of five columns for the continuous factors. Space-filling properties of the MCD, random LHD and the MaxPro design are compared in Figure 4 and 5.

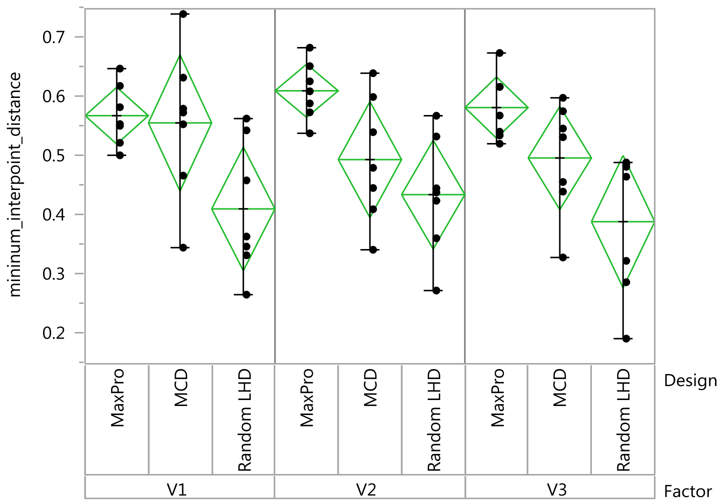
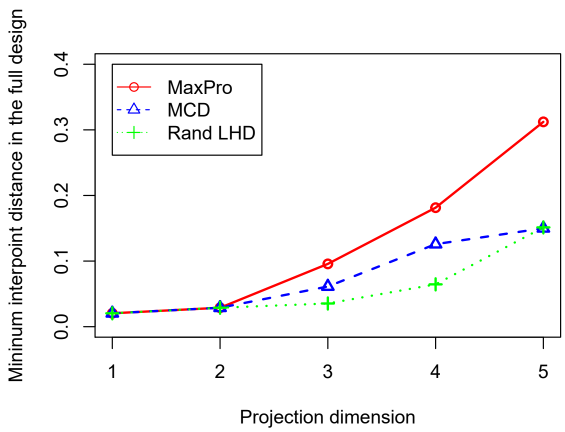


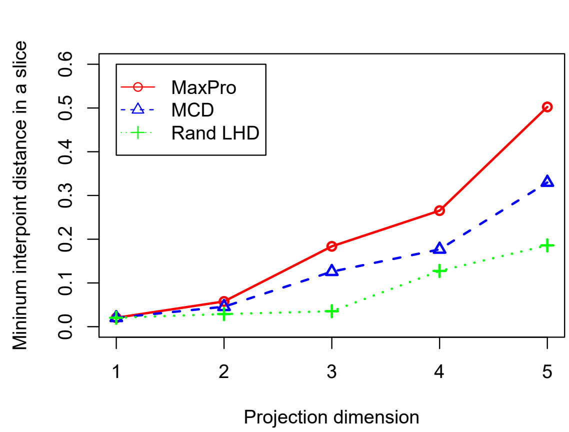
Figure 4: The minimum inter-point distances (larger-the-better) for each slice (block) of the MaxPro design, MCD and random LHD. Each time one nominal factor (V_1, V_2 or V_3) is used to partition the columns of continuous factors in each design into seven slices (blocks) according to its seven different levels.

In Figure 4, we partition the columns for continuous factors in each design into seven slices (blocks) according to the levels of one of the nominal factors (V_1, V_2 or V_3). The minimum inter-point distance of each slice is plotted in Figure 4, from which we can see that all slices of the MaxPro design are consistently space-filling and slices of the random LHD are always not good. Properties for the slices of the MCD, however, vary a lot: some of its slices are space-filling while some others can behave poorly. These results show that although the MCD structure guarantees one-dimensional uniformity when the columns of its continuous factors are partitioned by the levels of any nominal factor, it may not be space-filling in more than one dimensions.

In Figure 5(a), we compare the overall space-filling property and projection properties of the five continuous factors in each design. The minimum inter-point distances of the 49-run design with respect to all the five continuous factors, as well as projections to any four, three, two or one factors are shown. Because the MCD structure does not optimize the space-filing property other than ensuring the low-dimensional uniformity, we can clearly see that the MCD behaves substantially worse than the MaxPro design in more than two dimensions. Figure 5(b) further compares the projection properties of the three designs after partitioning the columns of their continuous factors into slices by each of their nominal factors. The comparison results are similar: the MaxPro design performs consistently better than the MCD for all projection



(a)



(b)

Figure 5: Plots of the minimum inter-point distances (larger-the-better) with respect to the continuous factors versus the projection dimensions: (a) the 49-run MaxPro design, MCD and random LHD; (b) the slices of the MaxPro design, MCD and random LHD, where the columns of continuous factors are partitioned into slices by levels of each nominal factors.

dimensions, and the random LHD is the worst.

Example: Solid End Milling Process

In this section, we illustrate the proposed MaxPro designs using a more realistic example. Consider the solid end milling process simulation described briefly in the introduction with six input factors. Three factors are continuous cutting tool parameters: rake angle (x_1), relief angle (x_2) and helix angle (x_3), which are shown in Figure 1. Ranges of these factors are chosen to be: $x_1 \in [3.5, 6.5]$ deg, $x_2 \in [21, 39]$ deg and $x_3 \in [7, 13]$ deg. The simulation also involves a discrete numeric factor, the number of flutes, which can only take values 2,3, or 4. Two nominal factors in the study are the titanium alloy (z_1) and the tool path optimization type (z_2). For them, six different types of titanium alloys that are commonly used in the aerospace industry and four available tool path optimization types are considered, whose details are given in Table 1. The simulation can be done on the Production Module software of Third Wave Systems (Minneapolis, MN). Here we discuss only the design aspects of the simulation. The outputs and their modeling using an experimental design for only the continuous factors are discussed in Gul et al. (2018).

Using a full factorial design for the two nominal factors, a 48-run MaxPro design is generated. The two-dimensional projections of the design are shown in Figure 6, from which we can see that the 48-run

Table 1: Nominal Factors and Levels

Level	Titanium Alloy	Tool Path Optimization
1	Ti-6AI-4V	None
2	Ti-6AI-2Sn-4Zr-6Mo	In-Cut
3	Ti-6AI-2Sn-4Zr-2Mo	Air-Cut
4	Ti-6AI-6V-2Sn	Both
5	Ti-4AI-4Mo-2Sn	
6	Ti-10V-2Fe-3AI	

MaxPro design is not only space-filling overall, but its design points for continuous factors are also space-filling when they are sliced by the types of titanium alloys, by the levels of tool path optimization type or by the levels of number of flutes. Under the settings of this study, an MCD does not exist because an OA of similar size which can accommodate a six-level nominal factor, a four-level nominal factor, a three-level discrete numeric factor, and three other continuous factors does not exist. An SLHD would also require a much larger run size in this example, since it needs to generate an LHD under each unique level combination of the qualitative factors. For example, even if we choose only five points for the small LHD in each slice, the SLHD would require $5 \times 6 \times 4 \times 3 = 360$ runs in total.

Another possible solution for this example is to use the Fast Flexible Filling (FFF) design (Leviketz and Jones 2015) from JMP, which employs a clustering-based algorithm to generate space-filling designs with both continuous and categorical factors. The design points for the continuous factors are selected from each cluster using the original MaxPro criterion and thus this would make a fair comparison with the proposed design. By treating the discrete numeric factor as nominal, we used JMP 13 to generate a 48-run FFF design with three continuous factors and three categorical factors containing three, six and four levels. As shown in Figure 7, the space-filling property of the FFF design is clearly not as good as that of the proposed MaxPro design in Figure 6. By using the more space-filling MaxPro design instead of the FFF design, we can expect much higher accuracy of the resulting surrogate model in predicting the tangential forces in the solid end milling process.

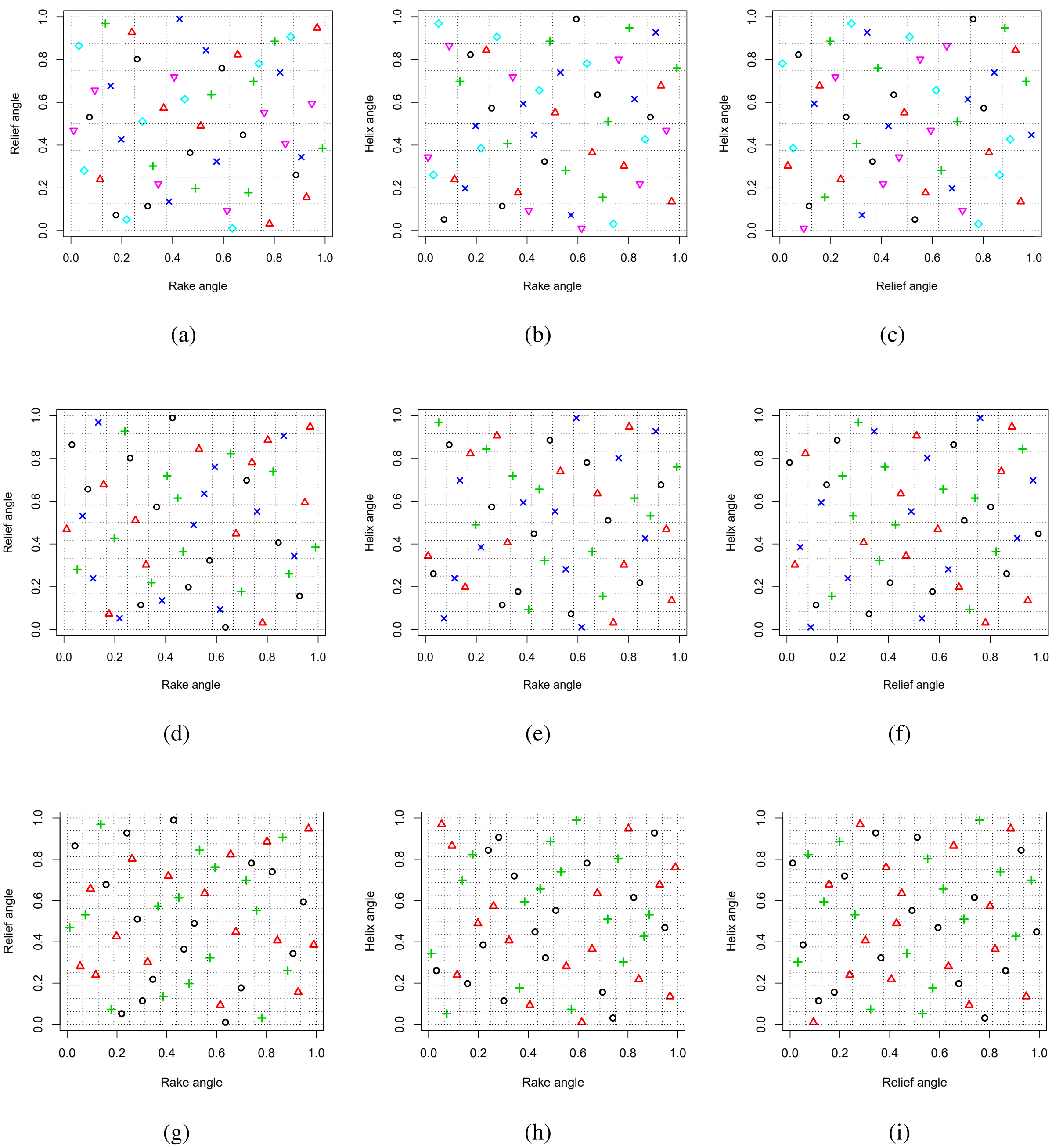


Figure 6: Pairwise scatter plots of the three continuous factors in the 48-run MaxPro design for the solid end milling process simulation (after standardizing each factor into the [0,1] unit region). (a)-(c): The six types of symbols correspond to the six different titanium alloy types in Table 1. (d)-(f): The four types of symbols represent the four different tool path optimization types in Table 1. (g)-(i): The three types of symbols represent the three different levels of number of flutes.

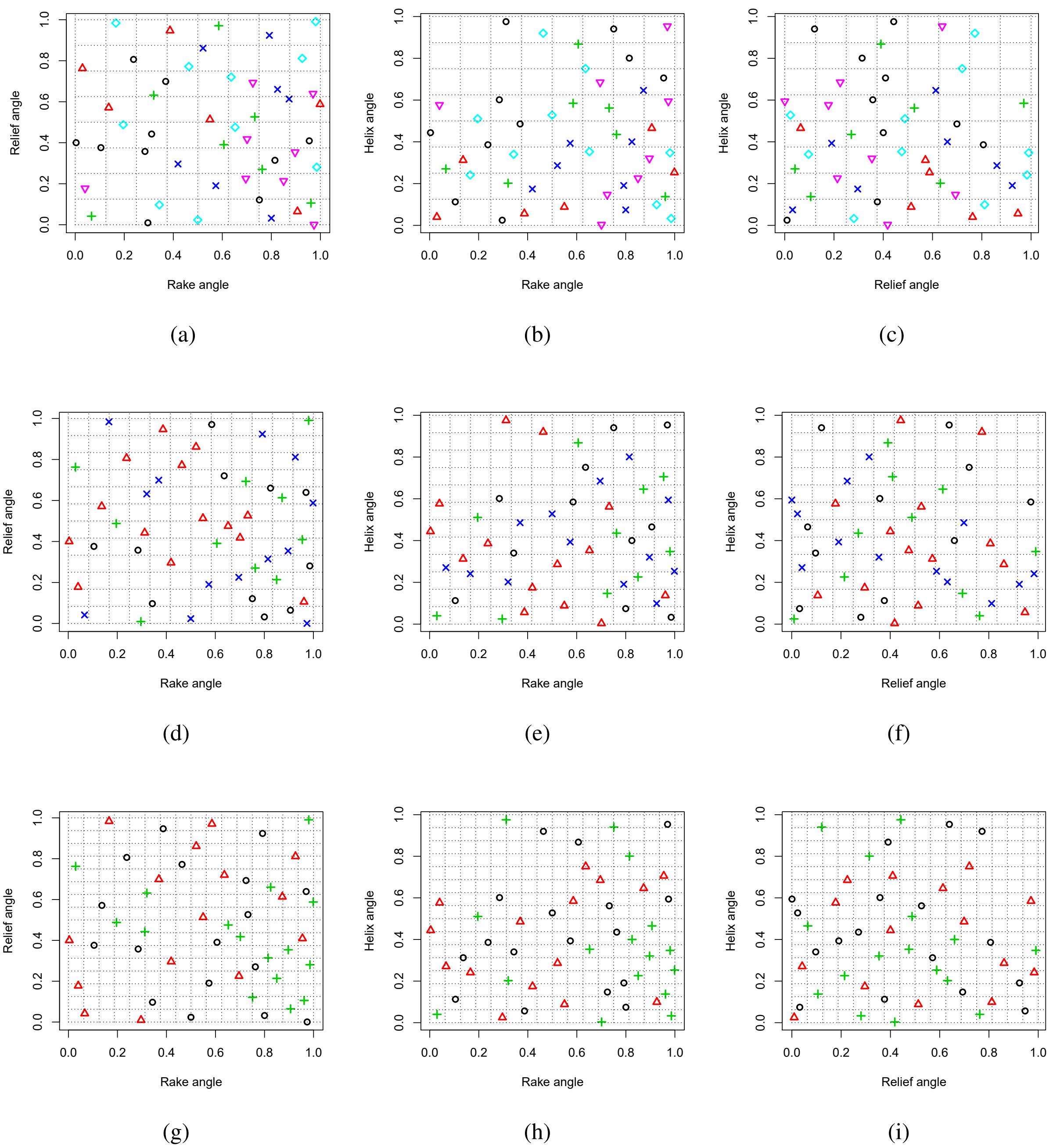


Figure 7: Pairwise scatter plots of the three continuous factors of the 48-run Fast Flexible Filling design from JMP 13 (after standardizing each factor into the [0,1] unit region). (a)-(c): The six types of symbols correspond to the six different titanium alloy types in Table 1. (d)-(f): The four types of symbols represent the four different tool path optimization types in Table 1. (g)-(i): The three types of symbols represent the three different levels of numbers of flutes.

Conclusions

In this work, we have extended the MaxPro criterion to accommodate not only continuous factors but also nominal, discrete numeric, and ordinal factors. Unlike the SLHD, the proposed MaxPro design can accommodate larger number of nominal factors with an economic run size. Compared to an MCD, the MaxPro design has better space-filling properties and has no restrictions on the run size and number of factors. The proposed MaxPro criterion seems to be the first optimal design criterion in computer experiments' literature that can incorporate all these different types of factors. In fact, if one wishes to use an SLHD or MCD, they can be optimized using the proposed criterion. A general algorithm to construct the proposed MaxPro design has been developed and is shown to work well using several examples. The implementation of the algorithm will be made available through the R package *MaxPro* (Ba and Joseph 2015).

Appendix: Equivalence of (4) and (9)

Here we show the equivalence of (4) and (9) for the case of a single nominal factor with t levels and p continuous factors. Consider the criterion in (9) without the power:

$$\psi(\boldsymbol{D}) = \frac{1}{\binom{n}{2}} \sum_{i=1}^{n} \sum_{j \neq i} \frac{1}{\prod_{l=1}^{p} (x_{il} - x_{jl})^2 \{I(z_i \neq z_j) + 1/t\}}.$$

Let, for $k \neq r$, $k, r = 1, ..., t$ and $m = n/t$,

$$\psi(\boldsymbol{D}_{x,k}, \boldsymbol{D}_{x,r}) = \frac{1}{m^2} \sum_{i=1}^{m} \sum_{j=1}^{m} \frac{1}{\prod_{l=1}^{p} (x_{il} - x_{jl})^2},$$

where $\boldsymbol{D}_{x,k}$ and $\boldsymbol{D}_{x,r}$ are the sub-designs of $\boldsymbol{D}_x$ corresponding to the qualitative factor level k and r, respectively. Let C is the set with 2-combinations of the set $1, \ldots, t$. Then,

$$\begin{aligned}
\psi(\boldsymbol{D}) &= \frac{1}{\binom{n}{2}} \left\{ \frac{m^2 t}{1+t} \sum_{(k,r) \in C} \psi(\boldsymbol{D}_{x,k}, \boldsymbol{D}_{x,r}) + t \binom{m}{2} \sum_{i=1}^{t} \psi(\boldsymbol{D}_{x,i}) \right\} \\
&= \frac{t}{\binom{n}{2}(1+t)} \left\{ m^2 \sum_{(k,r) \in C} \psi(\boldsymbol{D}_{x,k}, \boldsymbol{D}_{x,r}) + \binom{m}{2} \sum_{i=1}^{t} \psi(\boldsymbol{D}_{x,i}) + t \binom{m}{2} \sum_{i=1}^{t} \psi(\boldsymbol{D}_{x,i}) \right\} \\
&= \frac{t}{(1+t)} \left\{ \psi(\boldsymbol{D}_x) + \frac{(n-t)}{(n-1)} \frac{1}{t} \sum_{i=1}^{t} \psi(\boldsymbol{D}_{x,i}) \right\},
\end{aligned}$$

which is similar to (4) up to some scaling and with the maximin criterion $\phi(\cdot)$ replaced with the MaxPro criterion $\psi(\cdot)$.

Acknowledgements

This research is supported by a U.S. Army Research Office grant W911NF-17-1-0007 and a U.S. National Science Foundation grant DMS-1712642.

Bibliography

Ba, S. (2015). *SLHD: Maximin-Distance (Sliced) Latin Hypercube Designs*. R package version 2.1-1, http://CRAN.R-project.org/package=SLHD

Ba, S., and Joseph, V. R. (2015). *MaxPro: Maximum Projection Designs*. R package version 3.1-2, http://cran.r-project.org/web/packages/MaxPro

Ba, S., Brenneman, W. A., and Myers, W. R. (2015). Optimal Sliced Latin Hypercube Designs. *Technometrics* **57**(4), 479–487.

Cheng, S. W. and Ye, K. Q. (2004). Geormetric Isomorphism and Minimum Aberration for Factorial Designs with Quantitative Fcators. *The Annals of Statistics* **32**(5), 2168–2185.

Deng, X., Hung Y., and Lin C. D. (2015). Design for Computer Experiments with Qualitative and Quantitative Factors. *Statistica Sinica* **25**, 1567-1581.

Draper, N. R. and John, J. A. (1988). Response-Surface Designs for Quantitative and Qualitative Variables. *Technometrics* **30**, 423–428.

Gul, E., Joseph, V. R., Yan, H., and Melkote, S. N. (2018). Uncertainty Quantification in Machining Simulations Using In Situ Emulator. *Journal of Quality Technology*, to appear.

He, Y., Lin C. D., and Sun, F. (2017). On the Construction of Marginally Coupled Designs. *Statistica Sinica*, to appear.

He, Y., Lin C. D., Sun, F, and Lv, B. (2017). Marginally Coupled Designs for Two-Level Qualitative Factors. *Journal of Statistical Planning and Inference* **187**, 103-108.

Jin, R., Chen, and Sudjianto, A. (2005). An Efficient Algorithm for Constructing Optimal Design of Computer Experiments. *Journal of Statistical Planning and Inference* **134**, 268–287.

Johnson, M., Moore, L., and Ylvisaker, D. (1990). Minimax and Maximin Distance Designs. *Journal of Statistical Planning and Inference* **26**, 131–148.

Joseph, V. R., and Delaney, J. D. (2008). Functionally Induced Priors for the Analysis of Experiments. *Technometrics* **49**, 1–11.

Joseph, V. R., Ai, M., and Wu, C. F. J. (2009). Bayesian-Inspired Minimum Aberration Two- and Four-Level Designs. *Biometrika* **96**, 95–106.

Joseph, V. R., Gul, E., and Ba, S. (2015). Maximum Projection Designs for Computer Experiments. *Biometrika* **102**, 371–380.

Joseph, V. R. (2016). Space-Filling Designs for Computer Experiments: A Review. *Quality Engineering* **28**, 28–44.

Lekivetz, R., and Jones, B. (2015). Fast Flexible Space-Filling Designs for Non-rectangular Regions. *Quality and Reliability Engineering International* **31**(5), 829–837.

Lundy, M., and Mees, A. (1986). Convergence of an Annealing Algorithm. *Mathematical Programming* **34**, 111–124.

McKay, M. D., Beckman, R. J., and Conover, W. J. (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.

Morris, M. D. and Mitchell, T. J. (1995). Exploratory Designs for Computer Experiments. *Journal of Statistical Planning and Inference* **43**, 381–402.

Qian, P. Z. G., Wu, H., and Wu, C. F. J. (2008). Gaussian Process Models for Computer Experiments with Qualitative and Quantitative Factors. *Technometrics* **50**, 383–396.

Qian, P. Z. G. (2012). Sliced Lating Hypercube Designs. *Journal of the American Statistical Association* **107**(497), 393–399.

Sacks, J., Welch, W. J., Mitchell, T. J., and Wynn, H. P. (1989). Design and Analysis of Computer Experiments. *Statistical Science* **4**, 409–423.

Santner, T. J., Williams, B. J., and Notz, W. I. (2003). *The Design and Analysis of Computer Experiments*, New York: Springer.

Shewry, M. C., and Wynn, H. P. (1987). Maximum Entropy Sampling. *Journal of Applied Statistics* **14**, 165–170.

Tang, B. (1993). Orthogonal Array-Based Latin Hypercubes. *Journal of American Statistical Association* **88**, 1392–1397.

Wu, C. F. J. and Ding, Y. (1998), Construction of Response Surface Designs for Qualitative and Quantitative Factors. *Journal of Statistical Planning and Inference*, **71**, 331–348.

Wu, C. F. J., and Hamada, M. (2009), *Experiments: Planning, Analysis, and Parameter Design Optimization, 2nd Edition*, New York: Wiley.

Xu, H. (2002), An Algorithm for Constructing Orthogonal and Nearly-Orthogonal Arrays With Mixed Levels and Small Runs. *Technometrics* **44**, 356–368.