Affordable experimental design with tens of variables

Yaileen M. Méndez-Vázquez, Kasandra L. Ramirez-Rojas, Hecny Pérez-Candelario and Mauricio Cabrera-Ríos*

Industrial Engineering Department, The Applied Optimization Group, University of Puerto Rico at Mayagüez, Mayagüez, PR 00681-9000, USA

(Received 28 May 2014; accepted 18 August 2014)

Simulation models have importantly expanded the analysis capabilities in engineering designs. With larger computing power, more variables can be modeled to estimate their effect in ever larger number of performance measures. Statistical experimental designs, however, are still somewhat focused on the variation of less than about a dozen variables. In this work, an effort to identify strategies to deal with tens of variables is undertaken. The aim is to be able to generate designs capable to estimate full quadratic models using simply a personal computer. Quadratic models are interesting because they can support statistical testing, provide competitive approximating models, and make optimization problems tractable. Several strategies are contrasted: (1) generate designs with random numbers, (2) use designs already available in the literature, (3) generate designs under a clustering strategy, and (4) generate designs using random-walk methods. The first strategy is an easy way to generate a design, although the statistical properties cannot be controlled. The second strategy does focus on statistical properties, but some of the designs become rapidly inconvenient to generate when increasing the number of variables. The third and fourth strategies are investigated as novel possibilities to generate designs in a convenient manner.

Keywords: large experimental designs; full quadratic regression model; clustering designs

1. Introduction

Systems in engineering and sciences are affected by multiple factors simultaneously. Understanding how these factors affect key performance indicators is important for design, control, and optimization purposes. Moreover, achieving an appropriate understanding level must commonly be carried out while being mindful of resource consumption. Assessing the effects of multiple factors on multiple performance measures has been made a lot more convenient with the development of computer simulation.

A somewhat standard approach to understand variation through experimental means is the use of a regression model. Of special interest to this work is the situation in which curvature is suspected in the experimental response of interest; thus, a full quadratic regression model is sought. There seems to be an imbalance between the increasing capability of simulation models to relate large numbers of variables to similar numbers in performance measures and the restricted focus of statistical experimental designs in dealing with a low number of variables. Furthermore, if the idea is to generate a design to explore the space of tens of variables, the options are to recur to

*Corresponding author. Email: mauricio.cabrera1@upr.edu

© 2014 The Author(s). Published by Taylor & Francis.
This is an Open Access article distributed under the terms of the Creative Commons Attribution License http://creativecommons.org/licenses/by/3.0/, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. The moral rights of the named author(s) have been asserted.
commercially available software or computers with large number-crunching capability. The interest here is to generate designs for tens of variables using the everyday personal computer without incurring in significant additional costs.

This work attempts to bring attention to the imbalance described previously and foster the generation of designs to investigate tens of variables at a time. A more effective use of simulation models is possible with developments in this area including a more powerful capability of simulation optimization.

This work is organized as follows: first, an exploration of the literature in terms of number of variables and designs included in several studies is presented followed by a proposed strategy based on clustering to approach the study of tens of variables, as well as two other alternatives based on random-walk-like processes. The different strategies are then compared for 10, 20, and 50 variables on their ability to estimate a full quadratic model: (i) generate a random design; (ii) generate a design from techniques already established in the literature (full factorial design (FFD), central composite design (CCD), D-optimal design (D-OPT)); (iii) generate designs under a clustering strategy (CD); and (iv) generate designs using random-walk methods (LCG and MT). A case study in simulation optimization is later presented to demonstrate the usefulness of the proposed work. Finally, the results of the comparison are discussed and future directions are outlined.

2. Literature review

As technology and computational capacity increase, the possibility to analyze and simulate systems that are affected by multiple factors simultaneously is more attractive and feasible. For many experimental designs, the number of variables being investigated; however, is less than a dozen; in fact, in most cases, it is only three or four factors (Alkhatib, Muyibi, & Amode, 2011; Anotai, Thuptimdang, Su, & Lu 2012; Christin et al., 2008; Job, Sukumaran, & Jayachandran 2010; Laferriere et al., 2011; Larentis, Sampaio, Martins, Rodrigues, & Alves, 2011; Marwa et al., 2011; Rigas, Papadopoulou, Philippoussis, Papadopoulo, & Chatzipavlidis, 2009; Sudheer, Varakumar, & Reddy, 2010; Vishwantha, AppuRao, & Singh, 2010).

For many cases, the intention is to predict the parameters for the characterization of the system with a second-order model (Anotai et al., 2012; Cabrera-Rios, Mount-Campbell, & Irani, 2002; Larentis et al., 2011; Laferriere et al., 2011, Mahapatra, 2009; Marwa et al., 2011; Nobuyuki, Mello, Melo Santa Anna, & Pereira, 2010; Rigas et al., 2009; Sudhankar & Nagarajan 2011; Vaithanomsat et al., 2011). The experimental design used for these systems are usually the fractional factorial design, the central composite design, or the Box-Behnken design (Anotai et al., 2012; Cabrera-Rios et al., 2002; Job et al., 2010; Laferriere et al., 2011; Marwa et al., 2011; Nair et al., 2008).

It is desirable to analyze tens of factors that significantly affect a system simultaneously to be more realistic when trying to characterize and model it. In the literature, it is common to find mostly works that analyze fewer than 10 variables simultaneously; however, the number of variables tends to be even smaller when the objective is to build a full quadratic model since a minimum of three levels per variable is necessary to estimate pure quadratic effects (Cabrera-Rios et al., 2002; Christin et al., 2008; Sudheer et al., 2010).

There are different methods to generate experimental designs. In the literature, the full factorial is a highly popular one. This design enumerates all possible combinations of the levels of all variables involved. Needless to say, this strategy becomes impractical
rapidly with a small number of variables. For example, for 10 variables at three levels, the full factorial requires $3^{10} = 59,049$ runs. Due to this combinatorial explosion, a prevalent strategy is to run a fraction of the full factorial design, that is, to run a fractional factorial design. In contrast to two-level fractional factorial designs, three-level fractional factorial designs have not been favored in the literature due to complex aliasing structures (Sanchez, Lucas, Sanchez, Nannini, & Wan, 2012).

Non-linearity is more the rule than the exception in real systems; it is therefore important to characterize it adequately. Second-order regression models afford this possibility through the inclusion of pure quadratic terms – which model curvature due to each independent variable separately – as well as interaction terms, which try to capture the joint behavior of paired variables. When screening variables for modeling significance, however, an important interaction implies that both independent variables involved must remain in the model. Significant interactions can then preclude models to be reduced to a manageable size when dealing with a large number of variables. Competitive and succinct estimation of all terms in a full quadratic model is, then, a key capability in an experimental design to represent non-linearity correctly.

One of the best-known practices when fitting a full quadratic model is to use a central composite design which entails the use of either a two-level full factorial or fractional factorial design, plus two axial runs per variable involved, plus a defined number of center runs. This design capitalizes on the use of a fraction of the full factorial to keep the number of runs low while providing a stable and minimal variance in the model’s coefficients. Because the fractions of the full factorial used to build the central composite design result from the expression $2^{k-p}$, where $k$ is the number of variables and $p$ is the number of fraction generators, the resulting number of runs might be a lot more than those necessary for a full quadratic model (Alkhatib et al., 2011; Montgomery, 2009; Sayara et al., 2010; Wass, 2011; Zambare, 2011; Zhou, He, Niu, & Zhong, 2010).

Another possibility to obtain a full quadratic model is the use of a D-optimal design, with which one can decide upon the number of experimental runs a priori. This strategy, as coded in many commercial and open-source software packages, uses a full factorial enumeration from which the predefined number of runs is chosen with the objective to provide a minimum variance across all regression coefficients (Langner, 2003).

Furthermore, if simplicity is important, a naïve way to generate a design is using a probability mass function to prescribe a desired number of experimental combinations. This strategy is considered here due to its feasibility to explore several tens of variables simultaneously, although no control over variance or any other statistical properties can be exercised in this instance. Along with these strategies, this work explores generating designs through clustering procedures as well as random-walk procedures as explained next.

The methods presented in this work are geared towards the exploration of an experimental region with a single response in mind. The possibility of obtaining a full quadratic model, however, is extensive to multiple simultaneous responses. A case can be made on using partitioning experimental methods to make modeling of multiple responses more efficient, as discussed in Koch (1997). Additionally, a rich body of literature specialized on screening variables and integrating systems of interdisciplinary subsystems can be found related to multidisciplinary design optimization (Balling & Wilkinson, 1997; Cramer, Dennis, Frank, Lewis, & Shubin, 1994; Sobiesczanski-Sobieski & Haftka, 1996). These approaches, however, go beyond the scope of this work and will be explored in the future.
3. Cluster design method

3.1. Initial version

The cluster design method was investigated as an alternative to generate designs in our group. The strategy was as follows in its initial form: (i) generate a full factorial design as an initial enumeration; (ii) add a column with uniformly distributed random numbers to the full factorial design; (iii) generate $k$ clusters with the $k$-means algorithm, with $k$ being the number of necessary regression coefficients plus one; (iv) retrieve the $k$-medoids associated to the $k$ clusters; (v) delete the values associated to the column with the random numbers, and (vi) present the experimental design.

The rationale behind step (i) is to provide orthogonal design points. A random dummy variable is introduced as a means to add a controlled perturbation in step (ii). This is necessary because clustering by equally spaced orthogonal points results in very similar clusters, and thus to very similar centroids (i.e. not independent) in the next step.

The $k$-means algorithm is the most basic of the clustering techniques. It iteratively forms a user-defined number $k$ of exclusive clusters with each cluster organized around its average location or centroid. As proposed here, $k$ is set to the number of necessary regression coefficients to fit a full quadratic model plus one in step (iii). The number of coefficients for $v$ variables of interest can be calculated as:

$$1 + 2v + \left(\frac{v}{2}\right)$$  \hspace{1cm} (1)

From step (iii), then, $k$ clusters result. In step (iv), the medoid of each cluster is obtained. The medoids, which are data points in the center of a cluster, are intended as the $k$ runs in the resulting cluster design. In this work, an approximate medoid is computed for each cluster by using the median of each of the values of the $v$ variables of interest within the cluster under analysis. Steps (v) and (vi) of the method are self-explanatory.

![Figure 1. Growth in the number of runs as a function of the number of variables for the cluster design and the full factorial design.](image)
Equation (1) is useful also to show the growth of the intended method when increasing the number of variables, as shown in Figure 1, where this growth is contrasted with that of the full factorial design. Looking at Figure 1, it is clear that – if feasible – the cluster design would be convenient to explore tens of variables. However, a limitation also becomes apparent. The first step of the initial version of the method requires a full factorial enumeration; thus, it would become computationally inconvenient at some point. This observation, corroborated by a series of tests, leads to the following modified version of the method.

3.2. Modified version

The first step of the original method required the generation of a full factorial enumeration, which would become computationally inconvenient at some point as shown previously. A slower growing enumeration would help alleviate this situation. The following modification was then introduced:

1. Generate a cluster design D1 of moderate size, say one to explore \( v = 10 \) variables, using the original version of the method. D1 will have \( n \) runs.
2. Generate a second cluster design D2 as in the previous step. This second design will be different due to the random realization in step (ii). D2 will also have \( n \) runs. Concatenate every run in D1 together with every run in D2. The resulting enumeration contains \( n^2 \) runs with \( 2v \) variables.

With this new enumeration in place, steps (ii) through (vi) can then be applied to generate a design for up to \( 2v \) variables. Figure 2 shows the enumeration growth compared with the cluster design and the full factorial design. This is, indeed, an important point that results from this work as evidenced later: a smart initial enumeration can make a significant difference in the feasibility and the performance of the designs with tens of variables.

Figure 2. Growth on number of runs of the modified version enumeration compared with the full factorial enumeration.
4. Random-walk designs

After considering a randomly generated design, where no control can be exerted over the resulting statistical properties, the idea of creating a middle point between convenience and control could arise from designs elicited through random-walk procedures. This was especially necessary when approaching a case with 50 variables. The linear congruential generator (LCG) and the Mersenne Twister (MT) algorithm were used to this end. The idea behind the random-walk generator is a path that initiates in a known point and jumps in a determined direction with a given probability. Two random-walk variants are presented below.

4.1. LCG method

The linear congruential method is a pseudo-random number generator calculated with a linear equation as shown below:

\[ Z_i = (aZ_{i-1} + c) (mod(m)) \] (2)

where \( a \) and \( c \) are the multiplier and increment parameters, respectively, \( m \) is the module, and \( Z_i \) is the remnant integer from the ratio in the right-hand side of Equation (2).

To generate an experimental design, each variable in the design was initialized setting \( Z_0 \) as a random integer number in a range from 1 to 3, where each value has a probability of 1/3. Multiplier parameter \( a \) and modulus \( m \) were set to values of 1 and 3, respectively. The increment parameter \( c \) was defined as a function of the random number generated (\( rng \)) as follows: \( c = (0 \text{ if } rng < 1/3, \text{ 1 if } rng \geq 1/3 \text{ and } rng < 2/3, \text{ or 2 if } rng \geq 2/3) + 1 \). A series of numbers were then generated to match the number of necessary regression coefficients to fit a full quadratic model plus one. A balanced design – with as many columns as design variables and as many rows as regression coefficients plus one – is generated with this method.

4.2. Mersenne Twister method

The Mersenne Twister is derived from the generalized feedback shift register generator (Matsumoto & Nishimura, 1998). This algorithm has excellent statistical properties, including independence, uniformity, and competitive equidistribution (Matsumoto & Nishimura, 1998). It also has a large period length of \( 2^{19,937} - 1 \). This algorithm generates uniform random numbers in the range of \([0,1]\), and has been programmed in many software packages, including R-Project, which is of free distribution.

The generation of the experimental design was carried out in R-Project using the package called ‘rngSetSeedas’, where the initial seed was set to a value of 5. The idea of this method is to focus on repeatability. When the same seeds are selected, the resulting designs will be identical.

5. Comparison

The comparison among all competing strategies to generate experimental designs for tens of variables was carried out by artificially building a response through the addition of a known function and a random error. The known function was a full quadratic model, in the first case for 10 variables, 20 variables for a second case, and 50 variables for a third case, with all regression coefficients arbitrarily set equal to 10. The random error came from a normal distribution with 0 mean and standard deviation of 1.5 units.
The idea behind having an artificial response is to provide a controllable expected value and a random noise around it. The idea is focused on verification: if true experimental replicated data can be effectively modeled with a full quadratic regression model, it will look very similar to our artificial response. If we control the artificial response, then we can measure the performance of the designs when approaching it.

Experimental designs from each strategy (i) random design (RD), (ii) full factorial, central composite design and D-Optimal design, and (iii) the proposed clustering design, LCG Design, and the Mersenne Twister design, were used to sample and then to estimate the artificial response described previously. The following indicators were measured: (M1) number of runs, (M2) mean square error, (M3) number of regression coefficients estimated, (M4) the trace of \((XX)^{-1}\), that is, the trace of the inverse of the so-called design information matrix, which is proportional to the covariance of the regression coefficients, and (M5) the determinant of \((XX)^{-1}\) (Montgomery, 2009).

Residual analysis was also considered in this comparison to assess the assumptions of normality, independence, and constant variance. This is carried out mostly through hypothesis testing. The residual is computed for the \(i^{th}\) data point in \(n\) data points as 
\[e_i = Y_i - \hat{Y}_i; \ i = [1, 2 \ldots n]\], where \(Y_i\) is an actual observation and \(\hat{Y}_i\) is the corresponding fitted value from the regression model (Montgomery, 2009; Montgomery & Runger, 2007).

A design with the lowest possible number of runs, the lowest MSE, capable to estimate all regression coefficients, with the highest adjusted \(R^2\) value, the lowest value of the trace and determinant of \((XX)^{-1}\), and which complies with the residuals assumptions, would clearly dominate any other option.

Furthermore, it was important to assess how easy was to generate a design under each strategy. This last was done qualitatively by necessity. Finally, it was decided to tabulate the frequency of the coefficients by their percentual deviation from the target value. The results of the comparison are shown next for 10, 20, and 50 variables.

6. Results for 10 variables and the initial version of the cluster design method

In this experiment, only the initial version of the cluster design was included. Table 1 summarizes the comparative results for M1–M5. The D-optimal design seems to be an overall robust and sensitive alternative according to these results, with a minimum number of runs, the second lowest MSE, the capability to estimate all coefficients, and performing well in goodness-of-fit. The full factorial and the central composite designs,

| Experimental runs | Full factorial design | Central composite design | D-optimal design | Random design | Clustering design | Mersenne Twister design | LCG design |
|-------------------|-----------------------|--------------------------|------------------|---------------|------------------|------------------------|-----------|
| MSE               | 59,049                | 158                      | 71               | 71            | 71               | 71                     | 71        |
| Estimated coefficients | 2.2558               | 1.2351                   | .0774            | .0902         | .0944            | .069698                | .096793   |
| Trace of \((XX)^{-1}\) | .065                 | 86.24                    | 140.56           | 2854.7        | 670.1            | 1462.53                | 3366.64   |
| Determinant of \((XX)^{-1}\) | .00                  | 6.33E-124                | 1.495E-96        | 3.65E-89      | 5.39E-66         | 3.59E-68               | 5.84E-63  |

Table 1. Comparative results for different experimental designs for 10 variables.
even at 10 variables, start to seem impractical in terms of number of runs. This behavior was expected to be more drastic with larger numbers of variables. Looking at Table 2, it is evident that at this number of variables, the central composite design and the D-optimal design are the most competitive options.

Qualitatively speaking, the easiest options to generate (random design, Mersenne Twister design, and LCG design) show low values of MSE; although, the cost seems to come in terms of coefficient variance. At a competitive number of runs and an adequate performance in coefficient variance, the proposed clustering design at this point seemed like it could be improved to become a competitive option for larger numbers of variables. From running this comparison, it was experienced that both solving for the D-optimal design as well as carrying out the clustering procedure can be consuming in terms of computing resources. Devising a way to use a more efficient clustering procedure as well as to reduce the dependency on a complete enumeration as a starting point would help to importantly improve the proposed strategy.

Table 3 shows the results of the residual analysis for all designs under comparison. Normality was assessed with the Kolmogorov-Smirnov test, and independence with the Signs test. Variance homogeneity was assessed graphically and by measuring the percentage of residuals falling within a distance of two standard deviations of the estimated mean. Regarding the residuals’ normality test, the Random design, the Mersenne Twister design, and the LCG design showed varying degrees of deviation from normality, while independence did not seem a concern for any design.

### Table 2. Comparative results for the coefficients estimation by the different experimental designs for 10 variables.

|                 | Full factorial design | Central composite design | D-optimal design | Random design | Clustering design | Mersenne Twister design | LCG design |
|-----------------|-----------------------|--------------------------|-----------------|---------------|------------------|------------------------|-----------|
| ±5%             | 66                    | 52                       | 42              | 9             | 22               | 10                     | 12        |
| (5–10%)         | 0                     | 6                        | 8               | 13            | 18               | 17                     | 14        |
| (10–15%)        | 0                     | 2                        | 5               | 11            | 5                | 15                     | 10        |
| (15–20%)        | 0                     | 1                        | 3               | 8             | 2                | 10                     | 6         |
| >20%            | 0                     | 5                        | 8               | 25            | 19               | 14                     | 24        |

### Table 3. Comparative results of the residual analysis for different experimental designs for 10 variables.

|                        | Central composite design | D-optimal design | Random design | Clustering design | Mersenne Twister design | LCG design |
|------------------------|--------------------------|-----------------|---------------|------------------|------------------------|-----------|
| P-value of Kolmogorov  | >.15                     | >.15            | .046          | .138             | <.010                  | <.010     |
| Smirnov                |                          |                 |               |                  |                        |           |
| P-value of Runs Test   | .305                     | .137            | .261          | .274             | .780                   | .183      |
| Standard deviation     | 1.1149                   | .2802           | .3024         | .3094            | .2659                  | .3133     |
| μ – 2σ < v < μ + 2σ    | 149/158; 69/71;          | 66/71;          | 67/71;        | 69/71;           | 67/71;                 | 67/71;    |
|                        | 94%                      | 97%             | 93%           | 94%             | 97%                    | 94%       |
7. Results for 20 variables with the modified version of the cluster design method and a shortened enumeration for the D-optimal design

In this second set of results, the treatment of 20 variables was attempted. The modified version of the cluster design was included in this experiment. Also, the D-optimal design was generated in two ways: one with R-Project and an initial enumeration as in the modified clustering design, and the other with the commercially available statistical software JMP. Table 4 summarizes the comparative results for M1–M5 and Table 5 shows the distribution of the percentual deviation from the intended regression coefficients’ values. In this case, the random design presented the lowest MSE and has the capability to estimate all coefficients, but the precision for the estimates of the coefficient is lower than the D-optimal design using R-Project and the commercial software JMP (Table 4). The clustering design has a competitive value of MSE, and has the capability to estimate all coefficients, but the precision for the estimates of the coefficient is less than the obtained by the random design and the D-optimal design. The LCG design and the Mersenne Twister design do not fare well in terms of MSE; they are capable to estimate all regression coefficients but precision is still a challenge.

The full factorial and the central composite designs were not used in this comparison since at 20 variables, they are not practical. The full factorial design, for 20 variables at three levels each, requires $3^{20} = 3,486,784,401$ runs. The central composite design requires 1,048,617 experimental runs in its worst case. An important result is that of the D-optimal paired with the shortened initial enumeration as proposed in this work becomes feasible and is a competitive alternative for larger number of variables.

Table 4. Comparative results for different experimental designs of 20 variables.

| Experimental runs | D-optimal R-Project | D-optimal (JMP) | Random design | Clustering design | Mersenne Twister design | LCG design |
|-------------------|---------------------|-----------------|---------------|-------------------|------------------------|-----------|
| MSE               | .0453               | .0108           | .0036         | .0142             | .0924                  | .0383     |
| Estimates         | 231/231             | 231/231         | 231/231       | 231/231           | 231/231                | 231/231   |
| Trace of $(XX')^{-1}$ | 5100.7             | 251.1           | 15867.9       | 50480.9           | 153902.2               | 13148.6   |
| Determinant of $(XX')^{-1}$ | 0                  | 0               | 0             | 0                 | 0                      | 0         |

Table 5. Comparative results for the coefficients estimation by the different experimental designs for 20 variables.

| ±5%    | D-optimal R-Project | D-optimal (JMP) | Random design | Clustering design | Mersenne Twister design | LCG design |
|--------|---------------------|-----------------|---------------|-------------------|------------------------|-----------|
| (5–10%)| 145                 | 194             | 81            | 32                | 4                      | 17        |
| (10–15%)| 37                  | 9               | 65            | 26                | 3                      | 19        |
| (15–20%)| 12                  | 8               | 38            | 32                | 5                      | 10        |
| >20%   | 6                   | 4               | 23            | 25                | 5                      | 14        |
As in the previous case, a residual analysis was carried out. Table 6 shows the results of the hypothesis tests and the assessment of the variance. The D-optimal design (JMP) and the Mersenne Twister design showed some deviation in terms of normality. The D-optimal design (R-Project) and the LCG Design showed problems with independence.

8. Results for 50 variables

In this third set of results, the treatment of 50 variables was attempted. Table 7 summarizes the comparative results for M1–M6, and Table 8 shows the distribution of the percentage deviation from the intended regression coefficients’ values for experimental designs. For the development of experimental design for 50 variables, the modified version of the clustering method was used in combination with the D-optimal design; however, it was not possible to complete the design due to lack of independence in the resulting enumeration, as detected by the software.

In this case, the random design presented the lowest MSE and had the capability to estimate all coefficients, but the precision for the estimates of the coefficient is lower than the D-optimal design using (JMP), MT, and the LCG Design (Table 7). The D-optimal design (JMP) has an intermediate value of MSE, it has the capability to estimate all coefficients, and has the lowest value of the trace of $(XX)^{-1}$. It also has the best precision for the estimates of the coefficient.

As in the previous case, a residual analysis was carried out. Table 9 shows the results of the hypothesis tests and the assessment of the variance. The clustering design

### Table 6. Comparative results of the residual analysis for different experimental design for 20 variables.

| Design                  | D-optimal (R-Project) | D-optimal (JMP) | Random design | Clustering design | Mersenne Twister design | LCG design |
|-------------------------|-----------------------|-----------------|---------------|-------------------|--------------------------|------------|
| P-value of Kolmogorov-Smirnov | .133                  | .027            | >.15          | >.15              | <.010                    | >.15       |
| P-value of Runs Test    | .000                  | .795            | .895          | .595              | .480                     | .026       |
| Standard deviation      | .2130                 | .09910          | .0591         | .1195             | .62.81                   | .1961      |
| $\mu - 2\sigma < \mu < \mu + 2\sigma$ | 221/232; 221/232; 202/232; 220/232; 221/232; 222/232; | 95% 95% 87% 95% 95% 96% |

### Table 7. Comparative results for different experimental design for 50 variables.

| Design                  | D-optimal (JMP) | Random design | Clustering design | Mersenne Twister design | LCG design |
|-------------------------|-----------------|---------------|-------------------|--------------------------|------------|
| Experimental runs       | 1327            | 1327          | 1327              | 1327                     | 1327       |
| MSE                     | .0023           | .0002         | 8522329.079       | .0060                    | .0010      |
| estimated coefficients  | 1326/1326       | 1326/1326     | 1273/1326         | 1326/1326                | 1326/1326  |
| Trace of $(XX)^{-1}$    | 382.55          | 104,474.89    | -1.65598E+13      | 121,595.42               | 36,945.58  |
| Determinant of $(XX)^{-1}$ | 0                 | 0             | NA                | 0                        | 0          |
is the only that showed some deviation in terms of normality, while independence did not seem a concern for any design.

9. Illustrative example: production line with 50 workstations

The capability of dealing with tens of variables simultaneously opens important analysis possibilities ranging from statistical characterization to optimization. This section illustrates how a 50-variable simulation-optimization problem can be addressed aided by an experimental design with such capability. The strategies identified to generate experimental designs capable to analyze tens of variables at a time using a full quadratic regression model with the minimum number of necessary runs are shown in Figure 3, for 10, 20 and 50 variables.

Consider a production line with 50 workstations simulated with the software package SIMIO. The simulation is run for 8 h per day with 10 replicates. The simulation parameters of interest were the mean process time on each of the workstations (WSi). The process time in each workstation was assumed to follow a normal distribution with a mean that varied in three levels and a constant standard deviation of .25 min. It is further assumed that the nominal process time can be chosen by a particular user. The response of interest was the system time defined as the period of time elapsed since a raw part to be processed enters the system until it exits as a finished product.

A simulation optimization method based on design of experiments and metamodeling techniques was used (Villarreal-Marroquín, Castro, Chacón-Modragón, & Cabrera-Ríos, 2013). The method starts with an initial experimental design, which for 50 variables has 1327 experimental runs. Figure 4 shows the ranges of values to be explored with the objective to minimize the system time per unit.

| Table 8. Comparative results for the coefficients estimation by the different experimental designs for 50 variables. |
| --- |
| | D-optimal (JMP) | Random design | Clustering design | Mersenne Twister design | LCG design |
| ±5% | 1255 | 211 | 352 | 389 | 338 |
| (5–10%) | 26 | 206 | 98 | 311 | 285 |
| (10–15%) | 14 | 206 | 17 | 228 | 213 |
| (15–20%) | 3 | 172 | 4 | 151 | 187 |
| >20% | 28 | 531 | 802 | 247 | 303 |

| Table 9. Comparative results of the residual analysis for different experimental designs for 50 variables. |
| --- |
| | D-optimal (JMP) | Random design | Clustering design | Mersenne Twister design | LCG design |
| $P$-value of Kolmogorov Smirnov | >.15 | >.15 | <.010 | >.15 | >.15 |
| $P$-value of Runs test | .764 | .672 | .745 | .391 | .799 |
| Standard deviation | .0337 | .013 | 1392.76 | .078 | .032 |
| $\mu - 2\sigma < \epsilon < \mu + 2\sigma$ | 1276/1327; 1269/1327; 1285/1327; 1271/1327; 1257/1327; | 96% | 96% | 97% | 96% | 95% |
The minimum value for the average cycle time in the experimental design was identified and selected as the first best solution (first incumbent solution) (I-1). I-1 corresponds to a value of 312.09 min for the D-optimal design (JMP), 317.16 min for the LCG Design, 317.16 min for the Random Design, and 316.82 min for the Mersenne Twister design (Table 10). With the initial experimental design, a full quadratic regression meta-model was built, and used as the objective function to be minimized to obtain a predicted competitive solution. A generalized reduced gradient optimization procedure along with a multistart strategy was used for this purpose.

Using the process times prescribed for each workstation by the first predicted competitive solution, a simulation was performed and the simulated values were compared with the incumbent solution (I-1) for updating purposes. Each iteration of the algorithm follows a similar structure until either a solution that has already been visited is predicted, or a user-defined maximum number of iterations are met. For this example, a maximum of 40 iterations was used. The algorithm was stopped once it maxed out the allowed number of iterations. The best solution corresponded to a system time of 278.80 min for the D-optimal design (JMP), 304.72 min for the LCG, 295.61 min for the random design, and 303.5 min for the Mersenne Twister (Table 10).
When a comparison between the initial incumbent solution (I-1) with the final one (I-4) was performed, the result was that the system time decreased in 33.09 min for the D-optimal design (JMP), 12 min using the LCG, 21.5 min for the random design, and in 13.33 for the Mersenne Twister design (Figure 5). This represents a reduction of 10.67, 3.9, 6.8, and 4.21%, respectively, in the system time per unit in the simulated production system.

Although many aspects are interesting in this example, it is important to emphasize that it was possible to run this simulation-optimization procedure because there existed an experimental design capable to build a full quadratic regression model for 50 variables with a low number of runs.

### Conclusion

This study contrasts the performance of different strategies to generate experimental designs, aiming to devise feasible options to explore tens of variables simultaneously in the future. An emphasis was made in using only a personal computer for the generation of the design. It was learnt that a more efficient initial enumeration would improve the generation of the D-optimal design. It was also learnt that the clustering design might be improved in terms of coefficient variance for it to be a competitor to the D-optimal
design. Furthermore, at least the designs included in this preliminary comparison could be kept as benchmarks for future developments.

In these studies, at 10 variables, the traditional design of experiment techniques such as the full factorial design and the central composite designs are the most competitive options. These, however, are already difficult to generate at 20 variables, where computer generated designs such as the D-optimal become competitive. At 50 variables, designs that require a large and well-crafted initial enumeration such as the D-optimal and the proposed clustering design become difficult to approach with a personal computer, although their generation is still possible and competitive. It is remarkable, however, that designs simple to generate such as the random design and the random-walk-like methods become convenient options at such number of variables due to their overall feasibility. Further research on how to control the resulting designs from the latter seems promising in its own right.

An illustrative example with simulation optimization was used to show how important analysis possibilities open when having an experimental design that can be used to obtain a full quadratic model with the least possible number of experimental runs for tens of variables. This encourages further research into the matter. In addition, it is envisioned that the designs resulting from this work be tested in real systems in the future.

**Funding**

This material is based upon work supported by the National Science Foundation (NSF) under [grant number HRD 0833112] (CREST program); the National Institutes of Health (NIH) MARC [grant number 5T36GM095335–02] ‘Bioinformatics Programs at Minority Institutions’.

**References**

Alkhatib, M. F., Muyibi, S. A., & Amode, J. O. (2011). Optimization of activated carbon production from empty fruit bunch fibers in one-step steam pyrolysis for cadmium removal from aqueous solution. *Environmentalist Journal, 31*, 349–357.

Anotai, J., Thuptimdang, P., Su, C. C., & Lu, M. C. (2012). Degradation of O-Toluidine by fluidized-bed Fenton process: Statistical and kinetic study. *Environmental Science and Pollution Research, 19*, 169–176.

Balling, R., & Wilkinson, C. (1997). Execution of multidisciplinary design optimization approaches on common test problems. *AIAA Journal, 35*, 178–186.

Cabrera-Ríos, M., Mount-Campbell, C. A., & Irani, S. A. (2002). An approach to the design of a manufacturing cell under economic considerations. *International Journal of Production Economics, 78*, 223–237.

Christin, C., Smilde, A. K., Hoeftsoot, H. C., Suits, F., Bischoff, R., & Horvatovich, P. L. (2008). Optimized time alignment algorithm for LC-MS data: Correlation optimized warping using component detection algorithm-selected mass chromatograms. *Analytical Chemistry, 80*, 7012–7021.

Cramer, E. J., Dennis, J. J. E., Frank, P. D., Lewis, R. M., & Shubin, G. R. (1994). Problem formulation for multidisciplinary optimization. *SIAM Journal on Optimization, 4*, 754–776.

Job, J., Sukumaran, R. K., & Jayachandran, K. (2010). Production of a highly glucose tolerant B-glucosidase by pacelomycesvariottiing3: Optimization of fermentation conditions using Plackett Burman and Box-Behnken experimental designs. *World Journal of Microbiology & Biotechnology, 26*, 1382–1391.

Koch, P. N. (1997). *Hierarchical modeling and robust synthesis for the preliminary design of large scale, Complex systems*. Atlanta, GA: The G.W. Woodruff School of Mechanical Engineering.

Laferriere, C., Ravenscroft, N., Wilson, S., Combrink, J., Gordon, L., & Petre, J. (2011). Experimental design to optimize an Haemophilus influenzae type B conjugate vaccine made with hydrazide-derivatized tetanus toxoid. *Glycoconjugate Journal, 28*, 463–472.
Langner, H. W. (2003). Genetic algorithms for the construction of D-optimal design. *Journal of Quality Technology, 35*, 28–46.

Larentis, A. L., Sampaio, H. C., Martins, O. B., Rodrigues, M. I., & Alves, T. L. (2011). Influence of induction conditions on the expression of carbazoleoxyfenase components (CarAa, CarAc and CarAd) from *Pseudomonas Stutzeri* in recombinant *Escherichia Coli* using experimental design. *Journal of Industrial Microbiology and Biotechnology, 38*, 1045–1054.

Mahapatra, S. S. (2009). Modelling and analysis of erosion wear behavior of hybrid composites using Taguchi experimental design. *Journal of Engineering Tribology, 224*, 157–168.

Marwa, H. A., Sammour, A., El-ghamryHanaa, A., & El-nahasHanan, M. (2011). Optimizing proniosomes for controlled release of ketoprofen using Box-Behnken experimental design. *International Journal of Pharmaceutical Sciences and Research, 2*, 2195–2205.

Matsumoto, M., & Nishimura, T. (1998). Mersenne twister: A dimensionally equidistributed uniform pseudo-random number generator. *Journal ACM Transactions on Modeling and Computer Simulation, 8*, 3–30.

Montgomery, D. C. (2009). *Designs and analysis of experiments* (8th ed.). New York, NY: John Wiley.

Montgomery, D. C., & Runger, G. C. (2007). *Applied statistics and probability for engineers* (4th ed.). New York, NY: John Wiley.

Nair, V., Strecher, V., Fagerlin, A., Ubel, P., Resnicow, K., Murphy, S., … Zhang, A. (2008). Screening experiments and the use of fractional factorial designs in behavioral intervention research. *American Journal Public Health, 98*, 1354–1359.

Nobuyuki, R. M., Mello, P. S., Melo Santa Anna, L. M., & Pereira, N. (2010). Nitrogen source optimization for cellulase production by *penicilliumfuniculosum*, using a sequential experimental design methodology and the desirability function. *Applied Biochemistry and Biotechnology, 161*, 411–422.

Rigas, F., Papadopoulou, K., Philippoussis, A., Papadopoulou, M., & Chatzipavlidis, J. (2009). Bioremediation of lindane contaminated soil by pleurotusostreatus in non sterile condition using multilevel factorial design. *Journal of Water Air and Soil Pollution, 197*, 121–129.

Sanchez, S. M., Lucas, T. W., Sanchez, P. J., Nannini, C. J., & Wan, H. (2012). Designs for large-scale simulation experiments, with applications to defense and homeland security. In K. Hinkelmann (Ed.), *Design and analysis of experiments: Special designs and applications* (Vol. 3, pp. 413–441). Hoboken, NJ: Wiley.

Sayara, T., Sarra, M., & Sanchez, A. (2010). Optimization and enhancement of soil bioremediation by composting by using the experimental design technique. *Biodegradation Journal, 21*, 345–356.

Sobiesczanski-Sobieski, J., & Haftka, R. T. (1996). Multidisciplinary aerospace design optimization: Survey of recent developments. *Proceedings of 34th Aerospace Sciences Meeting and Exhibit*. Reno, NV: AIAA Paper AIAA-1996-0711.

Sudhankar, P., & Nagarajan, P. (2011). Optimization of chitinase production using statistics based experimental designs. *Journal of Chemical, Biological and Physical Sciences, 40*, 352–362.

Sudheer, K. Y., Varakumar, S., & Reddy, O. V. (2010). Production and optimization of polygalacturonase from mango (*Mangiferaindica L.*) peel using fusariummoniliforme solid state fermentation. *World Journal of Microbiology & Biotechnology, 26*, 1973–1980.

Vaithanomsat, P., Songpim, M., Malapant, T., Kosugi, A., Thanapase, W., & Mori, Y. (2011). Production of β-Glucosidase from a newly isolated Aspergillus species using response surface methodology. *International Journal of Microbiology, 2011*, 1–9.

Villarreal-Marroquin, M. G., Castro, J. M., Chacón-Modragón, O. L., & Cabrera-Ríos, M. (2013). Optimisation via simulation: A metamodelling-based method and a case study. *European J. Industrial Engineering, 7*, 275–294.

Vishwantha, K. S., AppuRao, A. G., & Singh, S. A. (2010). Acid protease production by solid-state fermentation using *Aspergillusoryzae* MTCC 5341: Optimization of process parameters. *Journal of Industrial Microbiology and Biotechnology, 37*, 129–138.

Wass, J. A. (2011). A further step in experimental design (III): The response surface. *Journal of Validation Technology, 17*, 54–62.

Zambare, V. P. (2011). Optimization of amylase production from *Bacillus Sp.* using statistics based experimental design. *Emirates Journal of Food and Agriculture, 23*, 37–47.

Zhou, W. W., He, Y. L., Niu, T. G., & Zhong, J. J. (2010). Optimization of fermentation condition for production of anti-TMV extracellular ribonuclease by *Bacillus Cereus* using response surface methodology. *Bioproces and Biosystems Engineering, 33*, 657–663.
Appendix A

The illustrative example in this work follows a simulation-optimization algorithm developed in our research group and described in Villarreal-Marroquín et al., 2013. This algorithm results in high-quality solutions that can be achieved efficiently with a modest number of simulation runs. The algorithm starts with an initial design of experiments (DOE) from which an incumbent solution is obtained. In each iteration, a metamodel is obtained using the available set of points and is used to generate a new attractive point where a simulation is performed. The simulated value of the new point is compared against the incumbent for updating purposes. A series of stopping criteria are evaluated and, if none is met, the new point is added to the existing set of points and a new iteration begins. Otherwise, the iteration stops. A more detailed description is presented next.

Initialization

1. **Initial DOE**: the initial DOE consists of \( n \) runs containing combinations of the \( v \) controllable variables of interest, \( \mathbf{x}' = (x_1, x_2, x_3, \ldots, x_v)' \), as well as their evaluations \( f(\mathbf{x}') \), where \( i = 1, 2, \ldots, n \). If a replicated DOE is used, the value of \( f(\mathbf{x}') \) will be the average across the replicates.

2. **Select incumbent**: considering a minimization instance, the DOE run with the minimum objective value is selected as the current best (incumbent) solution \( [x_{k\text{-best}}, f(x_{k\text{-best}})] \). An iteration counter is initialized here at \( k = 0 \).

Main Iteration

1. Update counter: \( k = k + 1 \)

2. **Obtain metamodel**: using the available points, build the \( k \)-th metamodel, \( f(\cdot)_k \). In case of having only few variables, a saturated metamodel is preferred i.e. one that uses all available degrees of freedom, in this case a regression model with \( (n + k - 1) \) coefficients.

3. **Optimize metamodel**: using the metamodel as objective function in the optimization problem under analysis, a multiple starting points heuristic is used along with a local optimizer to obtain an attractive solution, \( x_k \).

4. **Simulate the new point**: estimate, via simulation, the value of \( f(x_k) \) considering that if a replicated DOE was used, the same number of replicates is used for the new point and the mean value across them is reported.

5. **Evaluate if the new point is better than the incumbent**: in this case, evaluate if \( x_k \) has an objective value strictly lower than \( x_{(k-1)\text{-best}} \) i.e. if \( f(x_k) < f(x_{(k-1)\text{-best}}) \).

6. **Update the incumbent**: update the incumbent according to the evaluation in the previous step. If \( f(x_k) < f(x_{(k-1)\text{-best}}) \), then the following is set \( [x_{k\text{-best}}, f(x_{k\text{-best}})] = [x_k, f(x_k)] \), otherwise, the incumbent remains the same.

7. **Evaluate the stopping criteria**: stop the algorithm if (i) \( x_k \) belongs to the initial DOE or is similar to any of the points generated on previous iterations; (ii) if the coefficient of determination, \( R^2 \geq \epsilon \) (where \( \epsilon \) is defined by the user); or (iii) the maximum number of iterations has been reached. Both the \( \epsilon \) and the maximum number of iterations are defined by the user.

If any of the stopping criteria is met, the method stops and the incumbent is reported as the final output. Otherwise, \( x_k \) and its simulated objective function value are added to the set of points available to build a new metamodel, and the main iteration is repeated. This algorithm has been empirically shown to converge in a moderate number of iterations even in the presence of several variables using global optimization test functions (Villarreal-Marroquín et al., 2013).