Proceedings of the 2014 Winter Simulation Conference
A. Tolk, S. Y. Diallo, I. O. Ryzhov, L. Yilmaz, S. Buckley, and J. A. Miller, eds.

ENABLING SIMHEURISTICS THROUGH DESIGNS FOR TENS OF VARIABLES: COSTING MODELS AND ONLINE AVAILABILITY

Yaileen M. Méndez-Vázquez
Kasandra L. Ramírez-Rojas
Hecny Pérez-Candelario
Mauricio Cabrera-Ríos

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

ABSTRACT

Experiments are key to characterize, model and optimize engineering systems. The use of computer models and hence computer simulations, have allowed engineers to predict the effect of dozen and sometimes hundreds of variables at a specific time in a particular system. The combinatorial explosion that results from using classical techniques to generate experimental designs, however, has hampered such capability. Many analysis tasks, such as simulation optimization and simheuristics, will be importantly enhanced with the possibility of dealing with dozens of variables at a time in a convenient manner. In previous work we identified a series strategies to this end. The objective of the present study is to propose a costing approach to compare these strategies. In addition, designs for 10, 20 or 50 variables and their assessment are made readily available online to different users interested in simulation-optimization based on experimental design, as illustrated here with 50 variables.

1 INTRODUCTION

The use of computer models and, hence, of computer simulation, has allowed engineering to predict the effect of dozen and sometimes hundreds of variables at a time in a particular system. The combinational explosion that results from using classical techniques to generate experimental designs to analyze engineering systems, however, has hampered such capability.

Planning, executing and analyzing experiments are activities that belong to the field of statistical design of experiments. Planning an experiment is carried out with the intention to obtain the best statistical 'signal' in the presence of noise at the minimum cost. The traditional cost of an experiment entails resources such as energy, materials, time and money. In this work, however, a cost to generate the actual design is assessed, thus attempting an economics-based approach. In precedent work, our research group devised different strategies to generate experimental designs for tens of variables with the intention to obtain a full quadratic model with the least possible number of experimental runs (Méndez-Vázquez, Ramírez-Rojas, and Cabrera-Ríos 2013). The experience showed that, as the number of variables increases, different strategies become infeasible or highly inconvenient to meet the stated objective. From this effort, designs for 10, 20 and 50 variables are currently available. These designs were compared in terms of many statistical properties, including Mean Square Error and trace and determinant of the information matrix. The number of necessary experimental runs; and the capability to approach a simulation-optimization problem -a simheuristic- were also assessed.

The present work is organized as follows: first, the different strategies devised in the previous work to generate experimental designs are explained. Then, a costing approach to compare the resulting designs of the previous work is proposed. Finally, an illustrative example with an optimization simheuristic 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 50 variables. The latter effectively ties the idea of simheuristics to experimental design for tens of variables.

2 LITERATURE REVIEW

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 1, for 10, 20 and 50 variables. As the number of variables increases, many of these strategies become unfeasible. The designs for 50 variables are the focus of analysis here due to the potential they offer for system characterization, modeling and optimization.

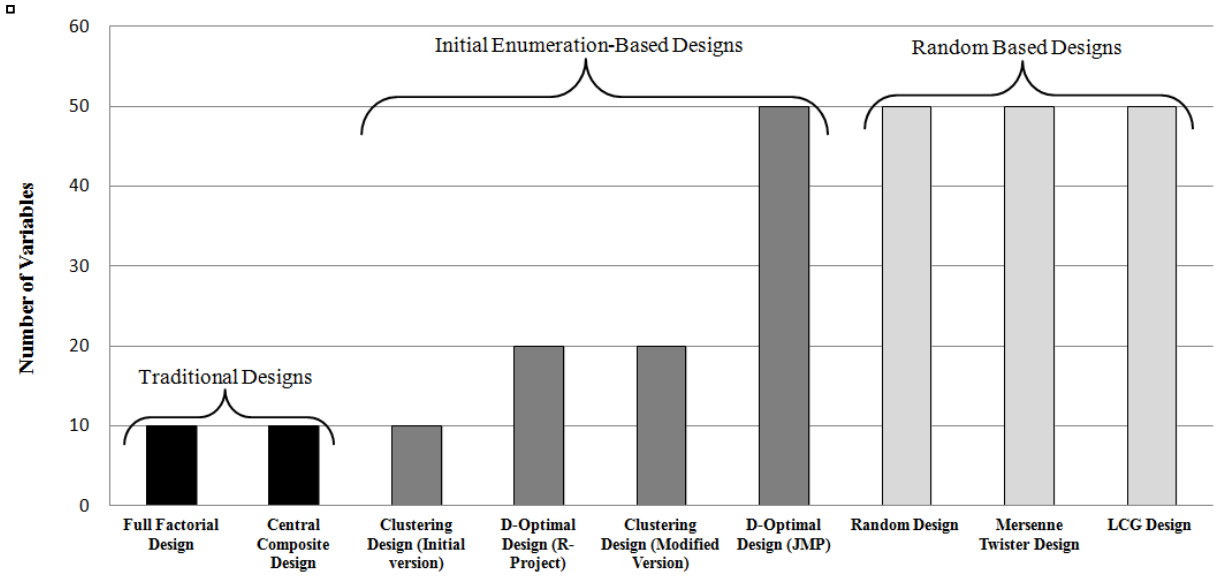

Figure 1: Resulting experimental designs for 10, 20 and 50 variables.

At 50 variables, it can be noticed that two categories remain in this work: Initial Enumeration Based Designs and Random Based Designs. In the D-optimal design, the sole remaining representation of the initial enumeration based designs at 50 variables, one can decide upon the number of experimental runs that are chosen from an initial enumeration. The objective of this strategy is to generate designs that result in the targeted regression model's parameters with the lowest variance. The strategy, then, minimizes the determinant of the inverse of the so-called design information matrix, $|(X'X)^{-1}|$; where X is the design matrix (Meyer and Nachtsheim 1995, Montgomery 2009).

This method, as coded in several commercial and open-source software packages, uses an initial enumeration in which the predefined number of runs is chosen - either by the use of the software- with the objective to meet the optimality criterion (Meyer and Nachtsheim 1995). Exchange algorithms are habitually used to generate this design (Meyer and Nachtsheim 1995). Two software packages were used in this work: R and JMP. In R, up to the time when this study was conducted, it was necessary that the experimenter provided the initial enumeration, while in JMP the enumeration is internally generated.

On the other hand, the designs that resulted from random based strategies, include the completely random design, and two designs generated using random walk methods. The random strategy is a simple

way to generate a design using a probability mass function to prescribe the sampling levels of the variables under study in a predetermined 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.

For the designs using random walk methods, the idea is to define a path starting in a known point and jumping in a determined direction with a given probability. Two strategies were identified to generate designs using this method: the Linear Congruential Generator (LCG) and the Mersenne Twister (MT) Algorithm. LCG is a pseudo random number generator calculated with a linear equation as shown below:

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

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 the equation. This method generates a balanced design with as many columns as design variables and as many rows as regression coefficients plus one. This design can be conveniently generated in an electronic spreadsheet.

The other strategy used in this work to generate a design is the MT algorithm, which is derived from the generalized feedback shift register (GFSR) generator (Matsumoto and Nishimura 1998). This algorithm generates uniform random numbers in the range of [0,1] and has been programmed in many software packages, including the freely distributed R-Project. The resulting random numbers can be translated into the levels of variables using a step function. MT sequences have excellent statistical properties, including independence, uniformity and competitive equidistribution (Matsumoto and Nishimura 1998).

3 COSTING MODEL

In this work a costing approach is proposed to compare the different strategies previously described. The costing approach was developed based on computing time and the cost associated to the purchase of necessary software to generate the design. The discussion at this point is limited to software available to the authors at the time of performing the comparison. The idea is to use the cost model to evaluate other alternative combinations as they become available. Furthermore, the cost model includes computational time for accounting precision purposes. As discussed previously, the resulting designs were classified in two categories: (1) those based in an initial enumeration and (2) those based on random processes. Table 1 shows the estimated associated costs. It is important to note that, in the second category, it is possible to reduce the software cost to 0 with the use of freely-distributed electronic spreadsheets, such as those included in LibreOffice and OpenOffice.

Table 1: Cost estimates for the generation of experimental design for 50 variables.

	Initial Enumeration-based Design	***Random-based Designs***		
	D-Optimal Design	**Random Design**	**LCG Design**	**MT Design**
Software	$1,470	$1,495	$139.99	$139.99
Computational Time	10-15 minutes	0	0	20-25minutes
Computational Cost (1$/minute)	$15	0	0	$25
Total Cost	$1,485	$1,495	$139.99	$164.99

3.1 Designs Based on Initial Enumeration

In this category, at the 50 variables mark, the D-optimal design using JMP software is the only feasible alternative out of the strategies included in this study. To generate a design with this strategy, it is necessary to have specialized professional software, such as JMP in this case. The cost of the license of this software is $1,470 annually. The professional license has a cost of $14,900 annually (JMP Inc. 2014).

In terms of computing time, these designs are coded in the software, requiring that the user selects the number of variables and the levels for each of them, as well as the desired number of runs. The software internally runs the algorithm and presents the designs in a short-period of time. For 50 variables, the software generated the design in approximately 10-15 minutes. Assuming arbitrarily that the computational time has a cost of $1 per minute, the total cost associated to the generation of D-optimal design for 50 variables is $1,485 (Table 1). The designs were generated using a trial version of the software, and a computer with Intel CORE i5 processor, 64 bits. All designs were generated using the same computer.

3.2 Designs Based on Randomness

This category contains the experimental designs based on random methods. These designs are discussed below.

3.2.1 Random Design

The generation of this design is, as its name indicates, completely random. It is not possible to control the design's statistical properties. In this method it is necessary to create a matrix with the levels of the variables' to be investigated on the design and the probability for each level. The number of experimental runs is established by the user. This design was generated in this work using Minitab following a uniform discrete distribution. The perpetual license of this software has a cost of $1,495 without upgrades versions (Minitab Inc. 2014).

The computing time for the generation of this design, in 50 variables, can be considered negligible because it only takes a few seconds. Therefore, the total cost associated to the generation of the random design is $1,495 (Table 1).

3.2.2 Random Walks Methods

Two methods were tried in this strategy. The first one generated the MT Design using R and MS Excel. R is an open access software, so it does not have a cost. This is a well-known statistical software characterized by its computational capacity. The Microsoft Office suite that includes Excel, in its version "Office home and student 2013" has a cost of $139.99, while the "Office Home and Business 2013" has a cost of $219.99 (MS Office Store, 2014). Freely-distributed electronic spreadsheets, such as those included in LibreOffice and OpenOffice, could feasibly be used instead in further applications.

For this design the MT algorithm was coded in R. The initial step is setting the seed to then generate vectors of magnitude v, where v represents the number of necessary regression coefficients to fit a full quadratic model plus one. These vectors were copied into Excel to be translated into practical levels. For this design k vectors were generated, where k represents the number of variables to be investigated. For each vector, if is necessary to establish a new seed. The computer time to generate each vector is negligible, as it takes only a few seconds. The process of copying and adjusting each vector in Excel, with the translation to the selected levels, takes approximately 0.5 minutes. The expression of the total generation time is 0.5n. For 50 variables for example, the design generation time was calculated as 0.5(50) = 25 minutes approximately. Using this approximation, the total cost associated to the generation of this design is $164.99 (Table 1).

The second method was the LCG. To develop this design, it was only necessary to use an electronic spreadsheet. The associated cost for Excel was shown previously. For this design it is necessary to set the associated parameters in formula (2). The computer time is not significant for our analysis. The total cost associated to this design is $139.99 (Table 1).

4 EXAMPLE OF OPTIMIZATION WITH A SIMHEURISTIC: 50 VARIABLES

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 based simheuristic with such capability.

The illustrative example in this work follows a simulation-optimization algorithm developed in our research group and described in (Villarreal- Marroquín, Castro, Chacón-Modragón, and Cabrera-Ríos 2013). This algorithm resulted in high quality solutions that were achieved efficiently with a modest number of simulation runs, a performance comparable to that obtained through OptQuest. (Villarreal-Marroquín, Mulyana, Castro, and Cabrera Ríos 2011)

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, $\mathbf{x}^i = (x_1, x_2, x_3, \ldots, x_v)^i$, as well as their evaluations $f(\mathbf{x}^i)$, where i=1,2,…,n. If a replicated DOE is used, the value of $f(\mathbf{x}^i)$ 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

3. Update counter: $k = k+1$
4. 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.
5. 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, $\mathbf{x}_k$.
6. Simulate the new point: Estimate, via simulation, the value of $f(\mathbf{x}_k)$ 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.
7. Evaluate if the new point is better than the incumbent: In this case, evaluate if $\mathbf{x}_k$ has an objective value strictly lower than $\mathbf{x}_{(k\text{-}1)\text{-best}}$ i.e. if $f(\mathbf{x}_k) < f(\mathbf{x}_{(k\text{-}1)\text{-best}})$.
8. Update the incumbent: Update the incumbent according to the evaluation in the previous step. If $f(\mathbf{x}_k) < f(\mathbf{x}_{(k\text{-}1)\text{-best}})$, then the following is set $[\mathbf{x}_{k\text{-best}}, f(\mathbf{x}_{k\text{-best}})] := [\mathbf{x}_k, f(\mathbf{x}_k)]$, otherwise, the incumbent remains the same.
9. Evaluate the stopping criteria: Stop the algorithm if (i) $\mathbf{x}_k$ belongs to the initial DOE or is similar to any of the points generated on previous iterations; (ii) if the coefficient of determination, $R^2 \geq \varepsilon$ (where ε is defined by the user); or (iii) the maximum number of iterations has been reached. Both the ε 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, $\mathbf{x}_k$ 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). It has also been shown applicable in physics-based simulation (Villarreal-Marroquín, et al. 2011).

For the development of the simulation model, consider a production line with 50 workstations modeled with the software package SIMIO. The simulation is run for 8 hours per day with 10 replicates. The simulation parameters of interest were the mean process time on each of the workstations (WS_i). 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 0.25 minutes. 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.

The method starts with an initial experimental design, which for 50 variables has 1327 experimental runs. Figure 2 shows the ranges of values to be explored with the objective to minimize the system time per unit.

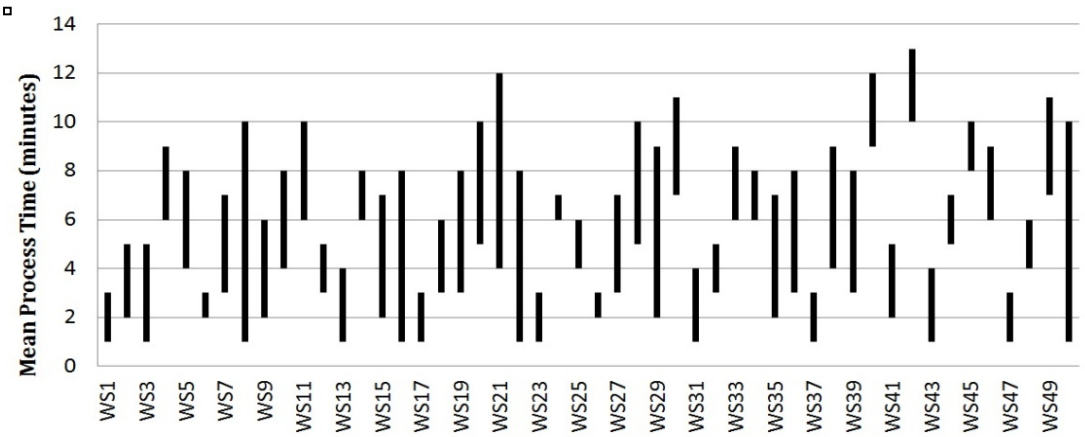

Figure 2: Range of values for the workstations' mean process time in simulation model.

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-1corresponded to a value of 312.09 minutes for the D-Optimal Design (JMP), 317.16 minutes for the LCG Design, 317.16 minutes for the Random Design and 316.82 minutes for the Mersenne Twister Design (Table 2)**.** With the initial experimental design, a full quadratic regression metamodel 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 multi-start strategy was used for this purpose.

Table 2: System Time for the incumbent solution for the production line with 50 workstations.

D-Optimal (JMP)			***LCG Design***			***Mersenne Twister Design***			***Random Design***		
Run	I-j	System Time (minutes)	Run	I-j	System Time (minutes)	Run	I-j	System Time (minutes)	Run	I-j	System Time (minutes)

166	I-1	312.09	**863**	I-1	317.16	**551**	I-1	316.82	**1168**	I-1	317.16
1328	I-2	291.10	**1341**	I-2	312.91	**1332**	I-2	312.32	**1329**	I-2	305.51
1329	I-3	284.27	**1346**	I-3	307.43	**1334**	I-3	311.94	**1334**	I-3	298.73
1331	I-4	281.72	**1355**	I-4	307.37	**1336**	I-4	307.47	**1356**	I-4	295.61
1335	I-5	281.24	**1361**	I-5	305.55	**1348**	I-5	304.74	-	-	-
1338	I-6	280.11	**1364**	I-6	304.72	**1354**	I-6	304.74	-	-	-
1341	I-7	279.55	-	-	-	**1356**	I-7	303.50	-	-	-
1364	I-8	278.80	-	-	-	-	-	-	-	-	-

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 is 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 minutes for the D-Optimal Design (JMP), 304.72 minutes for the LCG, 295.61 minutes for the Random Design and 303.5 minutes for the Mersenne Twister (Table 2).

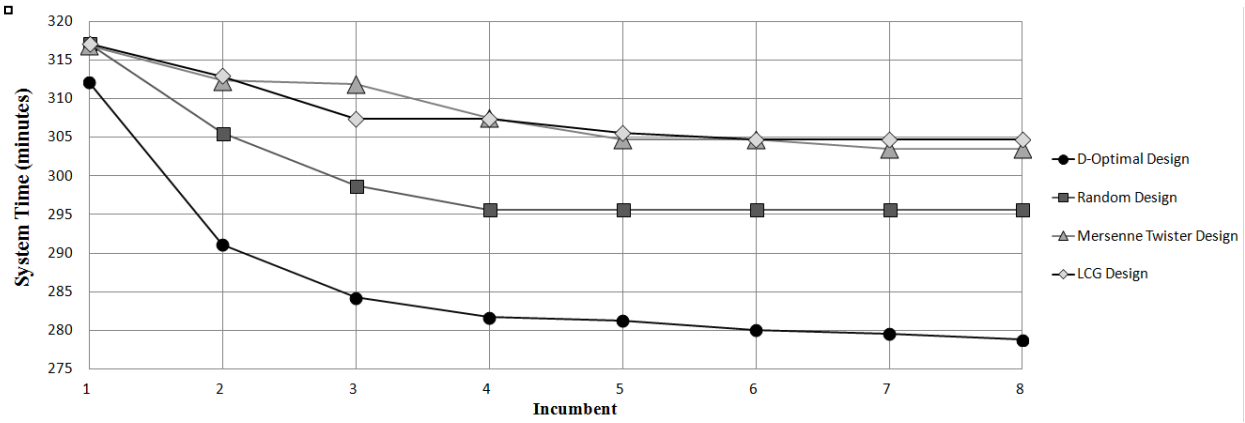

Figure 3: System time for the incumbent solutions of the simulation optimization method.

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 minutes for the D-Optimal Design(JMP), 12 minutes using the LCG, 21.5 minutes for the Random Design and in 13.33 for the Mersenne Twister Design (Figure 3). 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, the important part to emphasize here is 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.

5 ONLINE AVAILABILITY

Different experimental designs for 10, 20 and 50 variables and their assessment are made readily available online to those users interested in simulation-optimization based on experimental design. In this web page the designs, their statistical properties, the associated cost analysis, and the simulation-optimization example can be found. The website is in construction, and the URL is shown below:

6 CONCLUSION

Simheuristics will require data structures that facilitate their optimization work in the presence of tens of variables to remain realistic. This work provides designs of experiments, the largest of which can accommodate 50 variables with a relatively low number of runs and with the capability to completely estimate a full quadratic regression model. The most novel of these experiments are based on random walk processes and are rather inexpensive to generate. As demonstrated here, these designs can fully support a simulation-optimization process handling up to 50 variables. In addition, a website is currently under development to host these designs for distribution.

7 FUTURE WORK

Future work includes exploring cases with larger number of variables as well as improving the statistical properties of the random based designs to be inexpensive competitors to the D-Optimal Design. Furthermore, the possibility of incorporating partial information from the experimental design while is still running will be investigated as a means to speed up simulation optimization.

ACKNOWLEDGMENTS

This material is based upon work supported by the National Science Foundation (NSF) under Grant HRD 0833112 (CREST program), as well as the National Institutes of Health (NIH) MARC Grant 5T36GM095335-02 'Bioinformatics Programs at Minority Institutions'.

REFERENCES

Matsumoto, M., and T. Nishimura. 1998. "Mersenne Twister: A Dimensionally Equidistributed Uniform Pseudo-random Number Generator." *Journal ACM Transactions on Modeling and Computer Simulation* 8:3-30.

Méndez-Vázquez, Y., K. L. Ramírez-Rojas, and M. Cabrera-Ríos. 2013. "The Search for Experimental Design with Tens of Variables: Preliminary Results." In *Proceedings of the 2013 Winter Simulation Conference*, edited by R. Pasupathy, S. H. Kim, A. Tolk, R. Hill, and M. E. Kuhl, 2654–2665. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Meyer, R. K., and C. J. Nachtsheim. 1995. "The Coordinate-Exchange Algorithm for Constructing Exact Optimal Experimental Designs." *Technometrics* 37:60-69.

Microsoft Office Store. Accessed May 6, 2014. http://www.microsoftstore.com/store/msusa/en_US/cat/Office/categoryID.62684700?Icid=Office_suites_redirect_021314.

Minitab Inc. Minitab 17 Pricing. Accessed May 6, 2014. http://www.minitab.com/en-us/products/minitab/pricing/

Montgomery, D. C. 2009. *Designs and Analysis of Experiments*. 8th ed. New York: John Wiley & Sons, Inc.

SAS Institute Inc. JMP Statistical Discovery From SAS. Accessed May 6, 2014. http://word.tips.net/Pages/T000273_Numbering_Equations.html.

Villarreal-Marroquín, M. G., J. M. Castro, O. L. Chacón-Modragón, and M. Cabrera-Ríos. 2013. "Optimisation Via Simulation: A Metamodelling-Based Method and a Case Study." *European J. Industrial Engineering* 7:275-294.

Villareal-Marroquín, M. G., R. Mulyana, J. M. Castro, M. Cabrera-Ríos. 2011. "Selecting Process Parameters in Injection Molding via Simulation Optimization." *Journal of Polymer Engineering* 31:387-395.

AUTHOR BIOGRAPHIES

Yaileen M. Méndez-Vázquez obtained a BS degree and is currently pursuing her MS degree in the Industrial Engineering Department of University of Puerto Rico at Mayagüez. She is a graduate research assistant at the Applied Optimization Group at UPRM. Her research interests relate to experimental design and simulation optimization. Her email is yaileen.mendez@upr.edu.

Kasandra L. Ramírez-Rojas is currently pursuing her BS degree in the Industrial Engineering Department of University of Puerto Rico at Mayagüez. She is an undergraduate research assistant at the Applied Optimization Group at UPRM. Her research interests relate to experimental design and nonlinear optimization. Her email is kasandra.ramirez@upr.edu.

Hecny Pérez-Candelario is currently pursuing her BS degree in the Industrial Engineering Department at University of Puerto Rico at Mayagüez. She is an undergraduate research assistant at the Applied Optimization Group at UPRM. Her research interests relate to experimental design and high level programing languages. Her email is hecny.perez@upr.edu.

Mauricio Cabrera-Ríos obtained the M.S. and Ph.D. degrees in Industrial and Systems Engineering from The Ohio State University. He is an Associate Professor in the Industrial Engineering Department at University of Puerto Rico – Mayagüez. His research interests include manufacturing optimization and biological data analysis. His email is mauricio.cabrera1@upr.edu.